



UNIVERSIDADE FEDERAL DE ALAGOAS – UFAL
INSTITUTO DE COMPUTAÇÃO - IC

Disciplina: ORGANIZAÇÃO E ARQUITETURA DE COMPUTADORES

Curso: Engenharia e Ciência da computação

Docente: Erick de Andrade Barboza

Lista – Teórica: Pipeline e Superescalar

1. Considere um sistema de computação que possui um CPU com palavras de 32 bits, cujo repertório é o do processador MIPS. Considere inicialmente que a CPU foi implementada com a técnica multi-ciclo e que a quantidade de ciclos de cada instrução é dada na tabela abaixo. Qual o tempo de execução (em ciclos de clock) e o CPI para uma implementação multi-ciclo executar o programa abaixo? Suponha que o registrador \$t4 possui o valor 100 e os conteúdos das posições de memória 100 e 104 são respectivamente 5 e 10.

Tabela 1 – Ciclos por instrução para implementação multi-ciclo

Instrução	Nr. de ciclos
Aritméticas e deslocamento	4
Load word – lw	5
Store word - sw	4
Jump	3
Beq	4
Lui	3
Jal e Jr	4

```
Lw $t1, 4($t4)
Lw $t3, 8($t4)
Lw $t2, 12($t4)
Lw $t0, 0($t4)
Bne $t0, $t1, endX
Sub $t0, $t0, $t1
Sub $t3, $t3, $t2
addi $t0, $t0, 1
addi $t2, $t2, 1
sub $t2, $t2, $t1
sub $t1, $t1, $t3
j endY
```

```
endX:
    sub $t3, $t3, $t2
    add $t0, $t0, $t1
    addi $t3, $t3, -1
    addi $t2, $t2, -1
    add $t2, $t2, $t1
    add $t1, $t1, $t3
endY:
    sw $t0, 0($t4)
    sw $t1, 4($t4)
    sw $t2, 12($t4)
```



2. Considere agora que a CPU foi implementada com um pipeline de 5 estágios conforme mostrado na Figura 1 (abaixo) e deve executar o mesmo programa da questão anterior.

- Calcule o tempo de execução (em ciclos de clock) e o CPI no pipeline da Figura 1 considerando que NOP's são inseridos na ocorrência de conflito de dados e de controle. Assuma o tempo de execução das instruções nos estágios do pipeline conforme dado na Figura 1. Qual o speed-up da implementação em pipeline em comparação com uma implementação multi-ciclo?
- Aplique otimizações para resolver todos os conflitos de dados e de controle. Qual(is) a(s) otimização(ões) que você sugere para melhorar o desempenho do pipeline? Calcule o tempo de execução e o CPI com a(s) otimização(ões) sugerida(s). Qual o speed-up em comparação com a implementação multi-ciclo e com a implementação em pipeline sem otimizações?

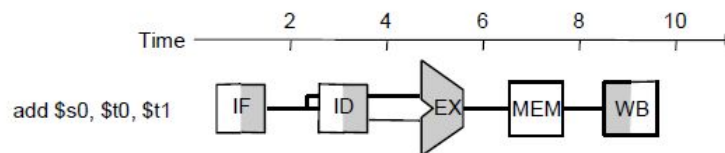


Figura 1. Pipeline de 5 estágios.

3. Um problema sério na implementação em pipeline é a ocorrência de conflitos de dados. Descreva 1 técnica de software e 1 de hardware que resolvem este tipo de conflito detalhando se o desempenho é degradado ou não e se algum suporte adicional (arquitetura ou hardware) se faz necessário.

4. Considere agora o programa em linguagem de montagem do MIPS descrito a seguir. Os vetores A, B e C possuem dimensão igual a 2. A localização das variáveis na memória pode ser visualizada na Figura 2, a qual contém os valores numéricos e simbólicos de cada endereço, bem como o correspondente conteúdo de memória. Por exemplo, a variável i possui o endereço numérico 128 e o seu valor inicial é 0. O vetor A possui o endereço inicial igual a 136 e o valor de A(0) é 5.

```
lui $3, 128
srl $3, $3, 16
lw $1, 0($3)
lui $2, 1
srl $2, $2, 16
beq $2, $1, fim
loop: lw $1, 8($3)
      lw $4, 24($3)
      beq $1, $4, oper_b j oper_c
oper_b: lw $4, 16 ($3)
oper_c: jal swap
add $5, $5, $2
```

```
addi $3, $3, 4
slti $6, $5, 2
bne $6, $0, loop
fim: break
swap: xor $1, $1, $4
xor $1, $1, $4
xor $4, $1, $4
xor $4, $1, $4
sw $1, 8($3)
jr $31
```



Conteúdo memória		
End. numérico		End.simbólico
	...	
128	0	i
132		
136	5	$A(0)$
140	2	$A(1)$
144	1	$B(0)$
148	3	$B(1)$
152	-3	$C(0)$
156	2	$C(1)$
160		

Figura 2. Localização e valores iniciais das variáveis na memória.

- Calcule o tempo de execução e o CPI do programa acima para uma implementação multi-ciclo da CPU
 - Calcule o tempo de execução e o CPI do programa acima no pipeline da Figura 1. Para resolver os conflitos de dados e de controle insira NOP's. Qual o speed-up em comparação com a implementação multi-ciclo
 - O tempo calculado acima pode ser reduzido? Qual(is) a(s) técnicas de resolução de conflito que você sugere? Calcule o tempo de execução e o CPI com a(s) otimização(ões) sugerida(s). Qual o speed-up obtido com as otimizações?
- Considerando novamente o programa MIPS da questão 4 responda as questões a seguir. Geralmente se pode identificar 3 tipos de dependências em programas. Explique essas dependências e encontre pelo menos um exemplo de cada no programa.
 - Duas técnicas usadas pelos compiladores são: escalonamento estático de instruções e loop unrolling. Mostre como o loop do programa da questão 4 pode ser desenrolado e como as instruções podem ser escalonadas. No novo código mostre as mudanças realizadas e descreva como e porque tais mudanças podem melhorar o desempenho.
 - Os processadores superescalares usam várias técnicas para explorar paralelismo de instrução ILP. Considerando tais técnicas responda às questões a seguir. Explique como funciona a técnica de previsão dinâmica de desvio baseada em preditores de 2 bits. Use o programa da questão 4 para mostrar como a previsão funciona. Quais as vantagens e desvantagens desta técnica?
 - Explique como os conflitos do tipo RAW, WAR e WAW são resolvidos pelo algoritmo básico de escalonamento dinâmico (Tomasulo).
 - De que maneira o reorder buffer ajuda no escalonamento dinâmico com especulação? Quais as principais modificações devem ser feitas em um processador para se introduzir suporte para especulação?