



Estructura de Computadores

Tema 5: Mejora del rendimiento con la segmentación

LECTURAS OBLIGATORIAS:

D. A. Patterson, J.L. Hennessy, “Estructura y Diseño de Computadores”:Capítulo 6.

BIBLIOGRAFÍA DE CONSULTA DEL TEMA:

D. A. Patterson, J.L. Hennessy, “Estructura y Diseño de Computadores”: Capítulo 6.

1. El siguiente fragmento de código se ejecuta en un MIPS con segmentación:

```
SUB R1,R2,R3
ADD R4,R5,R6
SUB R5,R4,R8
ADD R7,R2,R3
ADD R9,R7,R3
LW R1,10(R6)
ADD R3,R1,R4
SUB R6,R7,R8
```

Suponiendo que un dato se puede escribir en un banco de registros y leer su nuevo valor en el mismo ciclo:

- Calcular el número de ciclos necesarios para ejecutar el código si no existe posibilidad de adelantar operandos ni de reordenar el código.
- Calcular el número de ciclos necesarios si existe adelantamiento de operandos.
- Si se permite reordenar el código para que el número de ciclos sea mínimo, ¿cuántos ciclos son necesarios en este caso?

a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
SUB R1,R2,R3	IF																	
ADD R4,R5,R6		ID																
SUB R5,R4,R8		IF	EX															
ADD R7,R2,R3			ID	ME														
ADD R9, R7,R3			IF	EX	WB													
LW R1,10(R6)				ID	ME	WB												
ADD R3,R1,R4				IF	ID	ID	EX	ME	WB									
SUB R6,R7,R8					IF	IF	ID	EX	ME	WB	EX	ME	WB					
							IF	IF	IF	IF	ID	EX	ME	WB				
											IF	ID	ID	ID	EX	ME	WB	
												IF	IF	IF	ID	EX	ME	WB
																		WB

Contando:

18 ciclos

Otra forma:

 $4 \text{ ciclos llenado} + 8 \text{ instrucciones} + 3 \text{ RAW} * 2 \text{ ciclos paradas} = 18$

Otra forma:

 $4 \text{ ciclos llenado} + 8 \text{ instrucciones} + 6 \text{ ciclos en los que la etapa WB no se ejecuta} = 18$

b)

	1	2	3	4	5	6	7	8	9	10	11	12	13
SUB R1,R2,R3	IF												
ADD R4,R5,R6		ID											
SUB R5,R4,R8		IF											
ADD R7,R2,R3			EX										
ADD R9, R7,R3			ID	ME									
LW R1,10(R6)				EX	WB								
ADD R3,R1,R4				ID	ME	WB							
SUB R6,R7,R8				IF	EX	ME	WB						
					IF	ID	EX	ME	WB				
						IF	ID	EX	ME	WB			
							IF	ID	EX	ME	WB		
								IF	ID	EX	ME	WB	
									IF	ID	EX	ME	WB

Los cortocircuitos están sombreados.

Contando:

13 ciclos

Otra forma:

$4 \text{ ciclos llenado} + 8 \text{ instrucciones} + 1 \text{ RAW} * 1 \text{ ciclos paradas} = 13$

Otra forma:

$4 \text{ ciclos llenado} + 8 \text{ instrucciones} + 1 \text{ ciclos en los que la etapa WB no se ejecuta} = 13$

c) Intercambiamos las dos ultimas instrucciones, así la dependencia desaparece y el código tarda 12 ciclos.

2. En un computador MIPS64 ejecutamos la siguiente secuencia de código.:

```
SUB R1,R2,R3  
ADD R4,R5,R6  
OR R5,R1,R4  
SW R5,10(R6)  
LW R1,20(R8)  
AND R3,R8,R5  
ADD R3,R1,R7
```

Considerando que el banco de registros permite leer los datos en el flanco de bajada del reloj y se escriben en el de subida:

- Mostrar el diagrama temporal de ejecución del código cuando no se permite adelantamiento de operandos.
- Mostrar este mismo diagrama si se permite adelantamiento.

a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
SUB R1,R2,R3	IF	ID	EX	ME	WB											
ADD R4,R5,R6		IF	ID	EX	ME	WB										
OR R5,R1,R4			IF	ID	ID	ID	EX	ME	WB							
SW R5,10(R6)				IF	IF	IF	ID	ID	ID	EX	ME	WB				
LW R1,20(R8)							IF	IF	IF	ID	EX	ME	WB			
AND R3,R8,R5										IF	ID	EX	ME	WB		
ADD R3,R1,R7											IF	ID	ID	EX	ME	WB

Contando:

16 ciclos

Otra forma:

$$4 \text{ ciclos llenado} + 7 \text{ instrucciones} + 2 \text{ RAW} * 2 \text{ ciclos paradas} + 1 \text{ RAW} * 1 \text{ ciclo parada} = 16$$

Otra forma:

$$4 \text{ ciclos llenado} + 7 \text{ instrucciones} + 5 \text{ ciclos en los que la etapa WB no se ejecuta} = 16$$

b)

	1	2	3	4	5	6	7	8	9	10	11
SUB R1,R2,R3	IF										
ADD R4,R5,R6		ID									
OR R5,R1,R4		IF									
SW R5,10(R6)			EX								
LW R1,20(R8)			ID	EX	ME	WB					
AND R3,R8,R5			IF	ID	EX	ME	WB				
ADD R3,R1,R7				IF	ID	EX	ME	WB			
							IF	ID	EX	ME	WB

Contando:

11 ciclos

Otra forma:

4 ciclos llenado + 7 instrucciones = 11

Otra forma:

4 ciclos llenado + 7 instrucciones + 0 ciclos en los que la etapa WB no se ejecuta = 11

6. Se ejecuta el siguiente código en el MIPS:

```
eti1 :  LW R2,v(R1)
        BEQ R2,R0,eti2
        LW R3,W(R1)
        SW v(R1),R3
        SW w(R1),R2
eti2 :  SUBI R1,R1,#4
        BNE R1,R0,eti1
```

El bucle se aplica a vectores con un 80% de componentes iguales a cero y nos piden:

- Calcular el CPI medio de este código cuando se ejecuta un gran número de veces.
- Reordenarlo para minimizar su tiempo de ejecución y calcular el CPI en este caso.

a) Cuando R2 no es 0, se ejecuta el siguiente código:

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
eti1:	LW R2,v(R1)	IF	ID	EX	ME	WB										
	BEQ R2,R0,eti2		IF	ID	ID	ID	EX	ME	WB							
	LW R3,w(R1)			-	-	-	IF	ID	EX	ME	WB					
	SW V(R1),R3						IF	ID	EX	ME	WB					
	SW W(R1),R2							IF	ID	EX	ME	WB				
eti2:	SUBI R1,R1,#4									IF	ID	EX	ME	WB		
	BNE R1,R0,eti1										IF	ID	EX	ME	WB	
												-	-	IF	ID	EX

Los saltos se computan en la etapa ID por eso el BEQ R2,R0,eti2 tiene que esperar hasta que el dato del LW sale de memoria para poder usarlo en su fase ID.

$$CPI = \frac{\text{ciclos}}{\text{instrucciones}} = \frac{7 \text{ instrucciones} + 3 \text{ por datos} + 2 \text{ ciclos por saltos}}{7} = 1,71$$

Cuando R2 es 0, se ejecuta el siguiente código:

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
eti1:	LW R2,v(R1)	IF	ID	EX	ME	WB										
	BEQ R2,R0,eti2		IF	ID	ID	ID	EX	ME	WB							
eti2:	SUBI R1,R1,#4			-	-	-	IF	ID	EX	ME	WB					
	BNE R1,R0,eti1							IF	ID	EX	ME	WB				
									-	-	IF	ID	EX	ME		

$$CPI = \frac{\text{ciclos}}{\text{instrucciones}} = \frac{4 \text{ instrucciones} + 3 \text{ por datos} + 2 \text{ ciclos por saltos}}{4} = 2,25$$

El resultado es:

$$CPI = 0,8 * 2,25 + 0,2 * 1,71 = 2,14$$

b) Reordenando el código para rellenar la ranura de salto y evitar al máximo dependencias de datos nos queda:

Cuando R2 no es 0, se ejecuta el siguiente código:

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
eti1:	LW R2,v(R1)	IF	ID	EX	ME	WB										
	BEQ R2,R0,eti2		IF	ID	ID	ID	EX	ME	WB							
	[SUBI R1,R1,#4]			IF	IF	IF	ID	EX	ME	WB						
	LW R3,W(R1)						IF	ID	EX	ME	WB					
	SW V(R1),R3							ID	EX	ME	WB					
	SW W(R1),R2								IF	ID	EX	ME	WB			
eti2:	BNE R1,R0,eti1									IF	ID	EX	ME	WB		
	[LW R2,v(R1)]										IF	ID	EX	ME	WB	WB

$$CPI = \frac{\text{ciclos}}{\text{instrucciones}} = \frac{7 \text{ instrucciones} + 2 \text{ por datos} + 0 \text{ ciclos por saltos}}{7} = 1,28$$

Cuando R2 es 0, se ejecuta el siguiente código:

		1	2	3	4	5	6	7	8	9	10	11
	LW R2,v(R1)	IF	ID	EX	ME	WB						
eti1:	BEQ R2,R0,eti2		IF	ID	ID	ID	EX	ME	WB			
	[SUBI R1,R1,#4]			IF	IF	IF	ID	EX	ME	WB		
eti2:	BNE R1,R0,eti1						IF	ID	ID	EX	ME	WB

$$CPI = \frac{\text{ciclos}}{\text{instrucciones}} = \frac{4 \text{ instrucciones} + 3 \text{ por datos}}{4} = 2,33$$

El resultado es:

$$CPI = 0,8 * 2,33 + 0,2 * 1,28 = 1,66$$