

## **Notas**

LUI: (Load Upper Immediate)

ORI: (OR Immediate)

LWC1: Load Word Coprocessor 1 (floating point)

El color que muestra el simulador depende del tipo de instrucción.

### **Paso 3)**

Riesgos

cvt.d.s \$f0, \$f0 -> 1 ciclos de parada (ciclo 15)

cvt.d.s \$f2, \$f2 -> 1 ciclos de parada (ciclo 18)

#### **BUCLE**

add.d \$f8, \$f8, \$f6 -> 6 ciclos de parada (ciclo 25)

div.d \$f8, \$f8, \$f2 -> 3 ciclos de parada (ciclo 32)

swc1 \$f8, 0(\$t3) -> 9 ciclos de parada (ciclo 36)

bne \$t1, \$t6, loop -> 1 ciclos de parada del salto

#### **BUCLE**

### **Inst. Primera Iteración (desde la etiqueta de loop:)**

Instrucciones = 13

Total ciclos de parada = 18 datos + 1 control

Ciclos del bucle = 13 (1 instrucción por ciclo) + 19 (ciclos de parada) + 4 (rellenar el pipeline) = 36

### **Paso 4)**

Inicialización (fuera del bucle) + Iteraciones (CUIDADO PREDICTOR) + finalización

Instrucciones totales = 15 + 13\*100 + 2 (contando syscall) = 1317

Total ciclos de parada = 2 + [ (99\*19) + 18 ] + 0 = 1901

Ciclos totales: Instrucciones totales + Ciclos de parada totales + 4 (4 es porque la primera instrucción tarda 4 ciclos extra más el suyo propio en salir del pipeline. → 1317 + 1901 + 4 = 3222

*Nota:* El simulador no cuenta el `syscall` como instrucción, con lo que aparecen 1316 en las estadísticas. Sin embargo, sí que lo cuenta para el total de ciclos.

CPI = ciclos / instrucciones → 3222/1316 = 2.448 CPI

### Paso 5)

### Código optimizado

[illegible]

Ciclos: 2820  
CPI: 2.14285

$$\text{Speedup} = ((2.5243 / 2.14285) - 1) * 100 = 17,8159 \%$$

### **Paso 6)**

Ciclos: 547 | Instrucciones: 259

$$\text{CPI: } 547/259 = 2,111969112$$

Ciclos de parada | - Estructurales: 210  
| - Datos: 64  
| - Control: 9

### **Paso 7)**

Ciclos: 387 | Instrucciones: 259

$$\text{CPI: } 387/259 = 1,494208494$$

Ciclos de parada | - Estructurales: 50  
| - Datos: 64  
| - Control: 9

Mejoran los riesgos estructurales, que son los únicos que deben verse afectados por aumentar las unidades funcionales.

### **Paso 8)**

Unidades mínimas: 4 multiplicadores y 2 sumadores.

Pero, ¿por qué no 3 sumadores? → Existe un riesgo de datos que hace que el riesgo estructural del tercer add (add.d \$f20,\$f16,\$f18) nunca llegue a ocurrir.

CPI:  $337/259 = 1,301158301$

	- Estructurales: 0
Ciclos de parada	- Datos: 64
	- Control: 9

Estructurales : 0 → Las unidades están segmentadas y disponibles en el ciclo siguiente. Como sólo se lanza una instrucción no pueden tener riesgos estructurales.

Datos: 3 de la suma (add.d \$f20,\$f16,\$f18) y 3 del store (swc1 \$f20, -8(\$t5)).

Control: 1 ciclo.

Ciclos totales: Instrucciones totales + Ciclos de parada totales + 4 (4 es porque la primera instrucción tarda 4 ciclos extra más el suyo propio en salir del pipeline.

Ciclos totales:  $259 + [ 4 + (9*7) + 6 ] + 4 = 336$

*Nota:* 4 ciclos de parada fuera del bucle + 9 iteraciones con 6 ciclos de parada por datos y 1 de control + 1 iteración sólo con ciclos de parada por datos + 4 ciclos de rellenar el pipeline.

CPI :  $336/259 = 1.306$