

Laboratorio de paralelismo a nivel de instrucción

J. Daniel García Sánchez (coordinador)
Alberto Cascajo
Elías del Pozo
Christian Martínez

Arquitectura de Computadores
Grupo ARCOS
Departamento de Informática
Universidad Carlos III de Madrid

1. Objetivo

Esta práctica tiene como objetivo fundamental mejorar la comprensión por parte del estudiante de los conceptos relacionados con el paralelismo a nivel de instrucción.

2. Descripción de la práctica

En esta práctica se realizará la evaluación de la ejecución de dos pequeños programas sobre el simulador **EduMIPS64**. El simulador se puede obtener en la página <http://www.edumips.org/>. EduMIPS64 es un simulador escrito en Java que permite visualizar la ejecución de programas escritos en MIPS en un pipeline simplificado.

Desde la página mencionada se puede obtener:

- El simulador como un único archivo JAR.
- El manual de uso del simulador.
- El código fuente (disponible como un repositorio en GitHub).

El primer código a evaluar es el siguiente:

```
.data
v: .space 400
w: .space 400

.text
addi R1, R0, 400
addi R2, R1, -8
addi R3, R0, 0
addi R5, R0, -1

loop:
sw R3, v(R3)
addi R4, R3, 1
add R4, R4, R5
sw R4, w(R2)
```

```
addi R3, R3, 8
addi R2, R2, -8
bne R3, R1, loop
syscall 0
```

Listing 1: Ejercicio 1

El programa utiliza dos vectores (v y w) cada uno de los cuales puede almacenar 100 valores en doble precisión.

El segundo código a evaluar es el siguiente:

```
.data
v: .space 256
w: .space 256
x: .double 2.0
y: .double 1.5
z: .double 0.0
.text
daddi R1, R0, v
daddi R2, R0, w
daddi R3, R0, 256

ldc1 F2, x(R0)
ldc1 F4, y(R0)
daddi R4, R0, 0

loop1:
dmtc1 R4, F6
cvt.d.l F6, F6
mul.d F8, F2, F6
mul.d F8, F8, F8
sdc1 F8, 0(R1)
mul.d F10, F4, F6
sdc1 F10, 0(R2)
daddi R1, R1, 8
daddi R2, R2, 8
daddi R4, R4, 8
bne R4, R3, loop1

daddi R1, R0, v
daddi R2, R0, w
daddi R4, R0, 0
ldc1 F8, z(R0)

loop2:
ldc1 F2, 0(R1)
ldc1 F4, 0(R2)
mul.d F6, F2, F4
add.d F8, F8, F6
daddi R1, R1, 8
daddi R2, R2, 8
daddi R4, R4, 8
bne R4, R3, loop2
sdc1 F8, z(R0)
syscall 0
```

Listing 2: Ejercicio 2

NOTA: Tenga cuidado al copiar y pegar del documento PDF ya que algunos caracteres podrían cambiar. Compruebe la corrección del código antes de su ejecución.

3. Tareas

3.1. Tarea 1: Evaluación del ejemplo 1

Simule la ejecución del ejemplo 1 sin *forwarding*. Identifique las detenciones que se producen y la causa a la que se deben.

Active la opción de *forwarding* y vuelva a simular el mismo ejemplo. Determine el *speedup* que se produce debido al uso de *forwarding*.

3.2. Tarea 2: Evaluación del ejemplo 2

Simule la ejecución del ejemplo 2 sin *forwarding* y con *forwarding*. Identifique las detenciones que se producen en cada caso y las causas que los provocan.

Calcule el *speedup* que se consigue en el ejemplo, debida al uso de *forwarding*.

3.3. Tarea 3: Desenrollado del primer bucle

NOTA: Para todas las evaluaciones de este apartado mantenga la opción de *forwarding* activada.

Para el primer bucle (*loop1*) del ejemplo número 2, realice un desenrollado con factores de 2 y 4, pero sin desenrollar el segundo bucle.

Calcule las aceleraciones que se consiguen en ambos casos con respecto al código original.

IMPORTANTE: Además de desenrollar el bucle debe planificar las instrucciones para obtener la mejor aceleración posible.

3.4. Tarea 4: Desenrollado del segundo bucle

NOTA: Para todas las evaluaciones de este apartado mantenga la opción de *forwarding* activada.

Para el segundo bucle (*loop2*) del ejemplo número 2, realice un desenrollado con factores de 2 y 4, pero sin desenrollar el primer bucle.

Calcule las aceleraciones que se consiguen en ambos casos con respecto al código original.

IMPORTANTE: Además de desenrollar el bucle debe planificar las instrucciones para obtener la mejor aceleración posible.

3.5. Tarea 5: Desenrollado total

NOTA: Para todas las evaluaciones de este apartado mantenga la opción de *forwarding* activada.

Combine los desenrollados de las tareas 3 y 4.

Calcule las aceleraciones que se consiguen en ambos casos con respecto al código original.

IMPORTANTE: Además de desenrollar el bucle debe planificar las instrucciones para obtener la mejor aceleración posible.

4. Entrega

La fecha límite para la entrega de los resultados de este laboratorio finaliza a las 24 horas de la finalización de la clase de prácticas en el grupo en el que el estudiante esté oficialmente matriculado.

Se seguirán las siguientes reglas:

- Todas las entregas se realizarán a través de aula global.
- El único formato admisible para la entrega será rellenando el cuestionario a través de Aula Global.

- La entrega y realización de los cuestionarios se hará de forma individual incluso aunque el ejercicio se haya desarrollado en parejas. Tenga en cuenta que las preguntas a contestar podrán ser distintas.
- Una vez iniciado un cuestionario el estudiante dispondrá de un tiempo máximo de 10 minutos para completarlo.
- Cada estudiante dispondrá de un único intento para completar el cuestionario.
- El número máximo de cuestiones que tendrá que contestar cada estudiante será de 10.
- Se recomienda que el estudiante tenga preparadas en distintos archivos separados las soluciones a distintas tareas así como los datos asociados (detenciones, ciclos...), puesto que se le podrá requerir que entregue estos archivos o que conteste alguna cuestión sobre la solución.

3.1)

Sin forwarding: 661 ciclos 1,861CPI 355 instrucciones

Con forwarding: 408 ciclos 1,149CPI 355 inst.

$$S = \frac{661}{408} = 1.620$$

CPI · Inst.

RAW i2 → i1 Espera R1, en el 5º ciclo ya lo lee

i3: El IF espera a que i2 libere el D, en el 5º ciclo

i4: No empieza hasta que i3 puede entrar, el 6º ciclo

RAW i3 → i5 i5 Espera a que el i3 escriba el R3, 1 ciclo en el 10º

i6 El IF dura 2 ciclos hasta el 10º.

i7 Empieza en el 11º, ID dura 3 ciclos hasta 14º

i8 El IF dura hasta el 13º, ID dura 3 hasta 16º

i9 Empieza en el 14º y dura 3 hasta 16º

i10 Empieza en el 17º

i11 Comienza en el 18º a captar syscall o, probando
descodifica hace el salto.

3.2)

Sin : 2064 ciclos 3329 CPI 640 inst.

Con : 1582 ciclos 2551 CPI 620 inst.

$$S = \frac{2064}{1582} = 1'304$$

3.3) Desenrollado x2

$$S = \frac{1582}{990} = 1'597$$

Desenrollado x4

$$S = \frac{1582}{894} = 1'769$$

3.4) loop 2 x2

$$S = \frac{1582}{1263} = 1'252$$

loop 2 x4

$$S = \frac{1582}{1234} = 1'285$$

3.5) x2

$$S = \frac{1582}{671} = 2'357$$

x4

$$S = \frac{1582}{543} = 2'913$$