



Documento anónimo

lab1Arqui.pdf

Laboratorio 1 resuelto



3º Arquitectura de Computadores



Grado en Ingeniería Informática



Escuela Politécnica Superior
Universidad Carlos III de Madrid



Descarga la APP de Wuolah.
Ya disponible para el móvil y la tablet.





**KEEP
CALM
AND
ESTUDIA
UN POQUITO**

PRÁCTICA 1 ARQUITECTURA COMPUTADORES

¡¡SOLO se entrega el test!!

Simulador, bajar Jar de la página, copiar el código en un fichero de texto y lo guardamos en extensión .s

Comprobar los datos tras el desenrollado

Guardar estadísticas de ejecución

Numero de pasos puede ponerse. Ver cronograma de ejecución.

Hay que hacer la ejecución de todo antes de realizar el test.

Dale a Exceute- RUN y guardamos los datos. Fijarse un poco en los riesgos.

Copiar datos en un word para ver.

Pipeline

Test: 3 decimales eme los cálculos.

ESTADÍSTICAS

Ejercicio 1:

```
.data
v: .space 800
w: .space 800
```

```
.text
addi R1, R0, 800
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

```

Estadísticas:

- Sin forwarding.

```

Execution
1311 Cycles
705 Instructions
1.859 CPI (Cycles per instruction)
Stalls
503 RAW Stalls
0 WAW Stalls
0 WAR Stalls
0 Structural Stalls (Divider not available)
0 Structural Stalls (Memory not available)
0 Branch Taken Stalls
0 Branch Misprediction Stalls
Code size
48 Bytes
Floating point unit
FCSR register
      FCC      Cause EnablFlag RM
7654321 0      VZOUIVZOUIVZOUI
000000000000000000000000111100000001

```

- Con forwarding.

```

Execution
808 Cycles
705 Instructions
1.146 CPI (Cycles per instruction)
Stalls
0 RAW Stalls
0 WAW Stalls
0 WAR Stalls
0 Structural Stalls (Divider not available)
0 Structural Stalls (Memory not available)
0 Branch Taken Stalls
0 Branch Misprediction Stalls
Code size
48 Bytes
Floating point unit
FCSR register
      FCC      Cause EnablFlag RM
7654321 0      VZOUIVZOUIVZOUI
000000000000000000000000111100000001

```

Cálculo del speed up:

$$s = \text{tiempo formato original} / \text{tiempo mejorada} = 1311 / 808 =$$

Lo que importa es el tiempo de ejecución total.

Las mejoras de speed up pueden ser porque ejecutes las instrucciones más rápidos o que se ejecuten más instrucciones.

El speed up se calcula con ciclos o CPIS.

En el ejercicio 1 tienen el mismo número de instrucciones pero lo que cambia es el ciclo por instrucción de modo que:

$$\text{Speed up} = \frac{1,859 \text{ CPI} \times 705 \text{ instrucciones}}{1,146 \text{ CPI} \times 705 \text{ instrucciones}} = 1,622$$

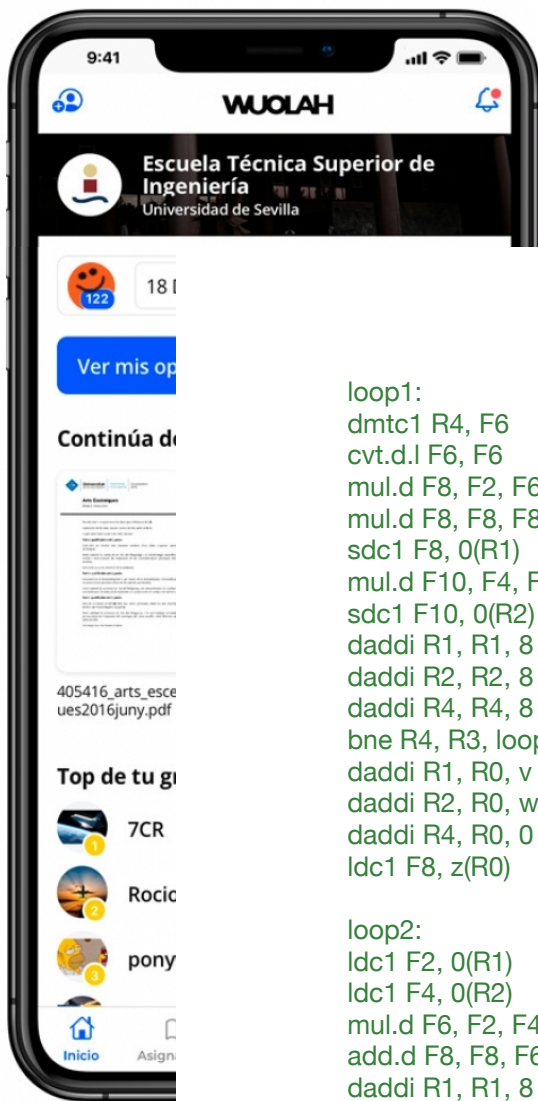
Cuidado

Nota: en el ejercicio 3 con el desenrollado el número de ciclo por instrucción no tiene por que mejorar pero si que ejecutan menos instrucciones.

Ejercicio 2:

```
.data
v: .space 128
w: .space 128
x: .double 2.0
y: .double 1.5
z: .double 0.0
.text
daddi R1, R0, v
daddi R2, R0, w
daddi R3, R0, 128
```

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



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.



```

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

```

Estadísticas:

· Sin forwarding.

```

Execution
1040 Cycles
316 Instructions
3.291 CPI (Cycles per instruction)
Stalls
674 RAW Stalls
0 WAW Stalls
0 WAR Stalls
0 Structural Stalls (Divider not available)
16 Structural Stalls (Memory not available)
0 Branch Taken Stalls
0 Branch Misprediction Stalls
Code size
124 Bytes
Floating point unit
FCSR register
      FCC      Cause EnablFlag RM
7654321 0      VZOUIVZOUIVZOUI
000000000000000000000000111100000001

```

· Con forwarding.

Execution

798 Cycles

316 Instructions

2.525 CPI (Cycles per instruction)

Stalls

400 RAW Stalls

32 WAW Stalls

0 WAR Stalls

0 Structural Stalls (Divider not available)

16 Structural Stalls (Memory not available)

0 Branch Taken Stalls

0 Branch Misprediction Stalls

Code size

124 Bytes

Floating point unit

FCSR register

FCC Cause EnablFlag RM

7654321 0 VZOUIVZOUIVZUI

00000000000000000000111100000001

$$\text{Speed up} = \frac{3,291 \text{ CPI} \times 316 \text{ instrucciones}}{2,525 \text{ CPI} \times 316 \text{ instrucciones}} = 1,303$$

Ejercicio 3: Ejemplo: Unrolling / Desenrollado.

```
for ( i = 0; i < 10; i++ ){  
    v[ i ] = v[ i + 1 ] + j;  
    j++;  
}
```

$$1 + 4 \times 10 = 41$$

Hacemos el desenrollado con un factor de 2.

```
for ( i = 0; i < 10; i = i + 2 ){  
    v[ i ] = v[ i + 1 ] + j;  
  
    v[ i + 1] = v[ i + 2 ] + j + 1;  
  
    j = j + 2;  
}
```

$1 + 5 \times 5 = 26$

duplicar instrucciones de memoria e incrementos

EJERCICIO 3: desenrollado loop 1 ej 2 con forwarding

```
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
```



```

loop1:
    dmtc1 R4, F6
    daddi R4, R4, 8
    dmtc1 R4, F12

    cvt.d.l F6, F6
    cvt.d.l F12, F12

    mul.d F8, F2, F6
    mul.d F8, F8, F8
    mul.d F13, F2, F12
    mul.d F13, F13, F13
    sdc1 F8, 0(R1)
    sdc1 F13, 8(R1)
    mul.d F10, F4, F6
    mul.d F14, F4, F12
    sdc1 F10, 0(R2)
    sdc1 F14, 8(R2)

    daddi R1, R1, 16
    daddi R2, R2, 16
    daddi R4, R4, 8
    bne R4, R3, loop1

```

Execution

646 Cycles
 292 Instructions
 2.212 CPI (Cycles per instruction)

Stalls

288 RAW Stalls
 16 WAW Stalls
 0 WAR Stalls
 0 Structural Stalls (Divider not available)
 24 Structural Stalls (Memory not available)
 0 Branch Taken Stalls
 0 Branch Misprediction Stalls

Code size

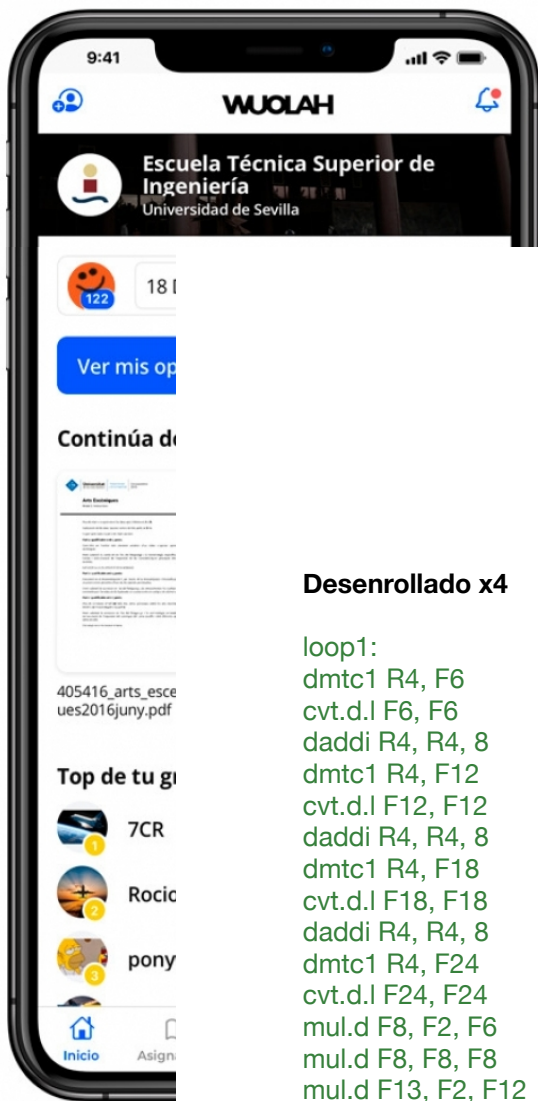
156 Bytes

Floating point unit

FCSR register

FCC	Cause	EnablFlag	RM
7654321 0	VZ0UIVZ0UIVZ0UI		
000000000000000000000000111100000001			

$$\text{La aceleración de desenrollar el bucle} = \frac{2,525 \text{ CPI} \times 316}{2,212 \text{ CPI} \times 292} = 1,235$$



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.



Desenrollado x4

```

loop1:
dmtc1 R4, F6
cvt.d.l F6, F6
daddi R4, R4, 8
dmtc1 R4, F12
cvt.d.l F12, F12
daddi R4, R4, 8
dmtc1 R4, F18
cvt.d.l F18, F18
daddi R4, R4, 8
dmtc1 R4, F24
cvt.d.l F24, F24
mul.d F8, F2, F6
mul.d F8, F8, F8
mul.d F13, F2, F12
mul.d F13, F13, F13
mul.d F19, F2, F18
mul.d F19, F19, F19
mul.d F25, F2, F24
mul.d F25, F25, F25
sdc1 F8, 0(R1)
sdc1 F13, 8(R1)
sdc1 F19, 16(R1)
sdc1 F25, 32(R1)
mul.d F10, F4, F6
mul.d F14, F4, F12
mul.d F20, F4, F18
mul.d F26, F4, F24
sdc1 F10, 0(R2)
sdc1 F14, 8(R2)
sdc1 F20, 16(R2)
sdc1 F26, 32(R2)
daddi R1, R1, 32
daddi R2, R2, 32
daddi R4, R4, 8
bne R4, R3, loop1
  
```

Execution
 594 Cycles
 280 Instructions
 2.121 CPI (Cycles per instruction)
Stalls
 232 RAW Stalls
 32 WAW Stalls
 0 WAR Stalls
 0 Structural Stalls (Divider not available)
 28 Structural Stalls (Memory not available)
 0 Branch Taken Stalls
 0 Branch Misprediction Stalls
Code size
 220 Bytes
Floating point unit
 FCSR register
 FCC Cause EnablFlag RM
 7654321 0 VZOUIVZOUIVZOUI
 000000000000000000000000111100000001

$$\text{La aceleración de desenrollar el bucle} = \frac{2,525 \text{ CPI} \times 316}{2,121 \text{ CPI} \times 280} = 1,343$$

Ejercicio 4

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

Desenrollado x2

loop2:

```

ldc1 F2, 0(R1)
ldc1 F4, 0(R2)
ldc1 F10, 8(R1)
ldc1 F11, 8(R2)

```

```

mul.d F6, F2, F4
mul.d F14, F10, F11
add.d F8, F8, F6
add.d F8, F8, F14
daddi R1, R1, 16
daddi R2, R2, 16
daddi R4, R4, 16

bne R4, R3, loop2

```

Execution

```

710 Cycles
284 Instructions
2.5 CPI (Cycles per instruction)

```

Stalls

```

352 RAW Stalls
40 WAW Stalls
0 WAR Stalls
0 Structural Stalls (Divider not available)
8 Structural Stalls (Memory not available)
0 Branch Taken Stalls
0 Branch Misprediction Stalls

```

Code size

```
140 Bytes
```

Floating point unit

```
FCSR register
```

```

      FCC      Cause EnablFlag RM
7654321 0      VZOUIVZOUIVZUI
000000000000000000000000111100000001

```

$$\text{Speed up} = \frac{2,525 \text{ CPI} \times 316 \text{ instrucciones}}{2,5 \text{ CPI} \times 284} = 1,123$$

Desenrollado x4

```
loop2:
```

```

ldc1 F2, 0(R1)
ldc1 F4, 0(R2)
ldc1 F16, 8(R1)
ldc1 F17, 8(R2)

```

```

ldc1 F18, 16(R1)
ldc1 F19, 16(R2)
ldc1 F20, 24(R1)
ldc1 F21, 24(R2)
mul.d F6, F2, F4
mul.d F24, F16, F17
mul.d F25, F18, F19
mul.d F26, F20, F21
add.d F8, F8, F6
add.d F8, F8, F24
add.d F8, F8, F25
add.d F8, F8, F26
daddi R1, R1, 32
daddi R2, R2, 32
daddi R4, R4, 32
bne R4, R3, loop2

```

Execution

674 Cycles
 268 Instructions
 2.514 CPI (Cycles per instruction)

Stalls

336 RAW Stalls
 44 WAW Stalls
 0 WAR Stalls
 0 Structural Stalls (Divider not available)
 4 Structural Stalls (Memory not available)
 0 Branch Taken Stalls
 0 Branch Misprediction Stalls

Code size

172 Bytes

Floating point unit

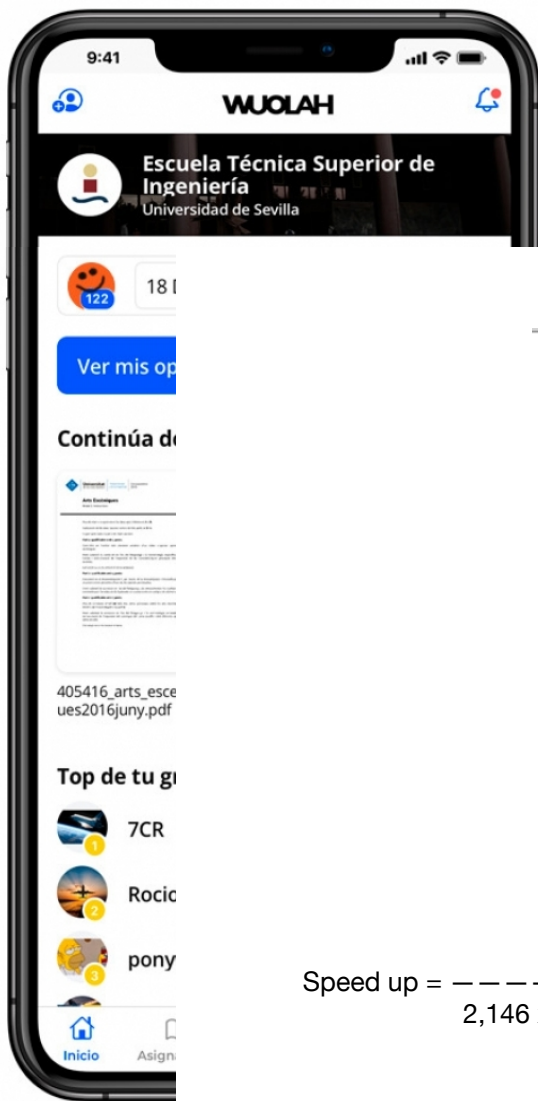
FCSR register

FCC	Cause	EnablFlag	RM
7654321 0	VZOUIVZOUIVZOUI		
000000000000000000000000111100000001			

$$\text{Speed up} = \frac{2,525 \text{ CPI} \times 316 \text{ instrucciones}}{2,514 \text{ CPI} \times 268} = 1,184$$

Ejercicio 5

Desenrollado x2:



Descarga la APP de Wuolah.

Ya disponible para el móvil y la tablet.



```

Execution
558 Cycles
260 Instructions
2.146 CPI (Cycles per instruction)
Stalls
240 RAW Stalls
24 WAW Stalls
0 WAR Stalls
0 Structural Stalls (Divider not available)
16 Structural Stalls (Memory not available)
0 Branch Taken Stalls
0 Branch Misprediction Stalls
Code size
172 Bytes
Floating point unit
FCSR register
      FCC      Cause EnablFlag RM
7654321 0      VZOUIVZOUIVZOUI
000000000000000000000000111100000001
  
```

$$\text{Speed up} = \frac{2, 525 \text{ CPI} \times 316 \text{ instrucciones}}{2,146 \times 260} = 1,43$$

Desenrolladox4:

```

Execution
470 Cycles
232 Instructions
2.025 CPI (Cycles per instruction)
Stalls
168 RAW Stalls
44 WAW Stalls
0 WAR Stalls
0 Structural Stalls (Divider not available)
16 Structural Stalls (Memory not available)
0 Branch Taken Stalls
0 Branch Misprediction Stalls
Code size
268 Bytes
Floating point unit
FCSR register
      FCC      Cause EnablFlag RM
7654321 0      VZOUIVZOUIVZOUI
000000000000000000000000111100000001
  
```

$$\text{Speed up} = \frac{2, 525 \text{ CPI} \times 316 \text{ instrucciones}}{2,025 \times 232} = 1,698$$