

UNIVERSIDAD CARLOS III DE MADRID

# Tema 0. Paralelismo y Organización de un Computador

---

Departamento de Ingeniería de Sistemas y  
Automática

RAÚL PÉRULA MARTÍNEZ  
LUIS ENRIQUE MORENO LORENTE  
ALBERTO BRUNETE GONZALEZ  
CESAR AUGUSTO ARISMENDI GUTIERREZ  
DOMINGO MIGUEL GUINEA GARCIA ALEGRE  
JOSÉ CARLOS CASTILLO MONTOYA



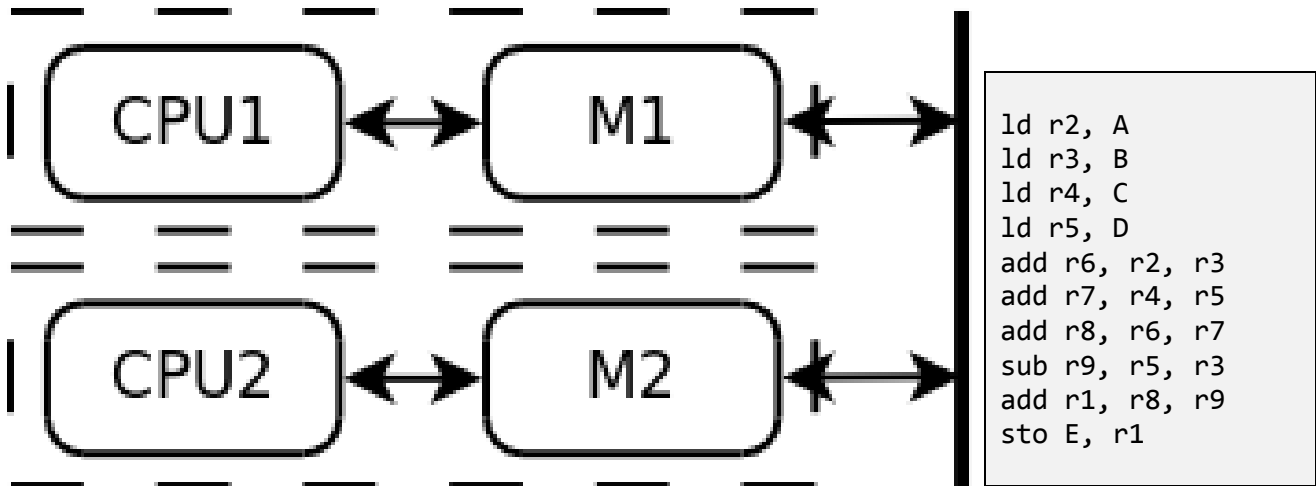
Universidad  
Carlos III de Madrid



Esta obra se publica bajo una licencia Creative Commons Reconocimiento-NoComercial-CompartidIgual 3.0  
España.

## Ejercicio 1

Dados la siguiente arquitectura y código en ensamblador:



1. Suponiendo que el programa se ejecuta en la CPU1, cuál sería la diferencia en número de ciclos si el acceso a la memoria se puede hacer en 1 o 2 ciclos.
2. Suponiendo que el tiempo de acceso del procesador 1 a la memoria 2 es de 3 ciclos (igual para el P2 – M1), determinar si hay alguna paralelización que mejore el resultado de ejecutarlo secuencialmente sobre un procesador, suponiendo que las lecturas fueran de 1 ciclo o de 2.

**Solución**

1. Acceso en 1 ciclo.

*c*

*o*

Ciclo	CPU1
1	ld r2
2	ld r3
3	ld r4
4	ld r5
5	add r6
6	add r7
7	add r8
8	sub r9
9	add r1
10	sto E

Total 10 ciclos.

*I l u*

$$CP = \frac{10}{10} = 1 \text{ cic /instr ón}$$

Acceso en 2 ciclos.

*c*

*o*

Ciclo	CPU1
1	ld r2
2	
3	ld r3
4	
5	ld r4
6	
7	ld r5
8	
9	add r6
10	add r7
11	add r8
12	sub r9
13	add r1
14	sto E
15	

Total 15 ciclos.

*I l u*

$$CP = \frac{15}{10} = 1.5 \text{ cic /instr ón}$$

2. Suponer el caso 1, de acceso a memoria M1 en 1 ciclo y a la otra en 1 ciclo también.

El programa solo accede a memoria para las lecturas de datos de entrada y salida, por lo que para pasar los datos de registro al otro procesador hay que escribirlos en alguna dirección de memoria que pueda ser leída por el otro procesador.

Ciclo	P1 / M1	P2 / M2
1	ld r2, A	ld r4, C
2	ld r3, B	ld r5, D
3	add r6, r2, r3	add r7, r4, r5
4	sto F, r3	sto 6, r7
5	ld r7, 6	ld r3, F
6	add r8, r6, r7	sub r9, r5, r3
7	... espera ...	sto 4, r9
8	ld r9, 4	
9	add r1, r8, r9	
10	sto E, r1	

Se puede observar que incluso en el caso de que el acceso del procesador 1 a la memoria M2 sea tan rápido como a la M1, no hay ganancia en lo que se refiere a tiempo de ejecución.

- En el caso de que los accesos a memoria de otro procesador tengan una penalización de 3 ciclos:

Si los accesos a la propia memoria tienen una penalización de 1 ciclo:

Ciclo	P1/M1	P2/M2
1	ld r2	ld r4
2	ld r3	ld r5
3	add r6	add r7
4	sto G, r3	sto F, r7
5	ld r7, F	ld r3, G
6	"	"
7	"	"
8	add r8	sub r9
9	---	sto H, r9
10	ld r9, H	
11	"	
12	"	
13	add r1, r8, r9	
14	sto E, r1	

Si los accesos dentro de la propia memoria tienen una penalización de 2 ciclos:

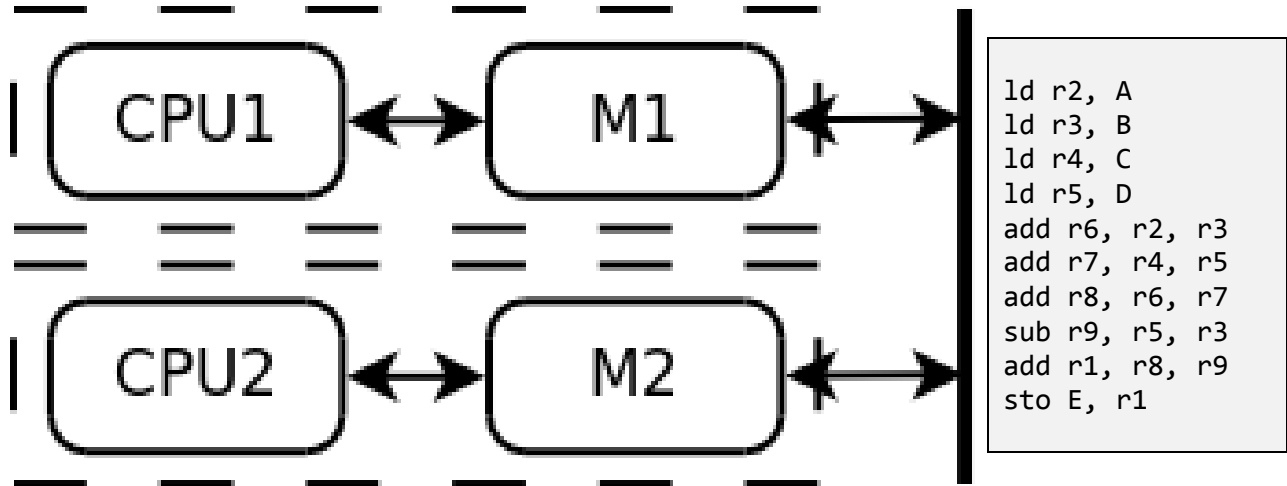
Ciclo	P1/M1	P2/M2
1	ld r2	ld r4
2	"	"
3	ld r3	ld r5
4	"	"
5	add r6	add r7
6	sto G, r3	sto F, r7
7	"	"
8	ld r7, F	ld r3, G
9	"	"
10	"	"



11	add r8	sub r9
12	---	sto H, r9
13		"
14	ld r9, H	
15	"	
16	"	
17	add r1, r8, r9	
18	sto E, r1	
19	"	

## Ejercicio 2

Dados la siguiente arquitectura y código en ensamblador:



1. Suponiendo que los accesos a memoria son todos de 1 ciclo y que las operaciones aritméticas requieren 2 ciclos:
  - a. Determinar el tiempo requerido para ejecutar en un procesador.
  - b. Determinar el tiempo requerido en una paralelización sobre 2 procesadores y razonar si es eficaz.



**Solución**

1.

a. P1 / M1

Ciclo	CPU1
1	ld r2
2	ld r3
3	ld r4
4	ld r5
5	add r6
6	
7	add r7
8	
9	add r8
10	
11	sub r9
12	
13	add r1
14	
15	sto E

$$CP = \frac{15 \text{ ciclos}}{10 \text{ instrucciones}} = 1.5 \text{ cic/instrucción}$$

b. Usando, por ejemplo, la paralelización.

Ciclo	P1 / M1	P2 / M2
1	ld r2, A	ld r4, C
2	ld r3, B	ld r5, D
3	add r6, r2, r3	add r7, r4, r5
4	sto F, r3	sto 6, r7
5	ld r7, 6	ld r3, F
6	add r8, r6, r7	sub r9, r5, r3
7	-----	sto 4, r9
8	ld r9, 4	
9	add r1, r8, r9	
10	sto E, r1	

Además, en lo que se refiere al tiempo necesario para ejecutarlo.

Ciclo	P1 / M1	P2 / M2
1	ld r2, A	ld r4, C
2	ld r3, B	ld r5, D
3	add r6, r2, r3	add r7, r4, r5
4		
5	sto F, r3	sto 6, r7
6	ld r7, 6	ld r3, F
7	add r8, r6, r7	sub r9, r5, r3
8		
9	... espera ...	sto 4, r9
10	ld r9, 4	
11	add r1, r8, r9	
12		
13	sto E, r1	

Se ganarían 2 ciclos, en lo que se refiere al tiempo total de ejecución, es decir, el tiempo que se tardaría sería un 13% menor.

$$\frac{13}{15} = 0.87$$

Se puede observar alguna mejora:

Ciclo	P1 / M1	P2 / M2
1	ld r2, A	ld r4, C
2	ld r3, B	ld r5, D
3	add r6, r2, r3	add r7, r4, r5
4		sto 6, r7
5	ld r7, 6	ld r3, B
6	add r8, r6, r7	sub r9, r5, r3
7		sto 4, r9
8	ld r9, 4	
9	add r1, r8, r9	
10	sto E, r1	

En tiempo será:

Ciclo	P1 / M1	P2 / M2
1	ld r2, A	ld r4, C
2	ld r3, B	ld r5, D
3	add r6, r2, r3	add r7, r4, r5
4		
5	//////////	sto 6, r7
6	ld r7, 6	ld r3, B
7	add r8, r6, r7	sub r9, r5, r3
8		
9	//////////	sto 4, r9
10	ld r9, 4	
11	add r1, r8, r9	
12		
13	sto E, r1	



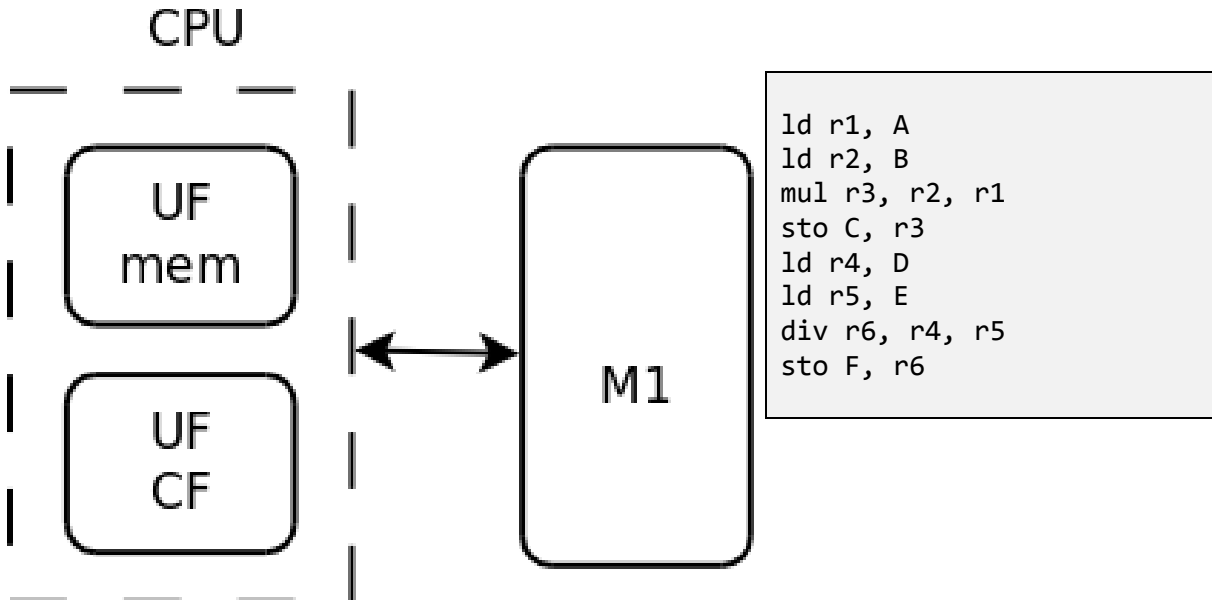


Universidad Carlos III de Madrid  
Departamento de Ingeniería de Sistemas y Automática

Que como se observa es de nuevo de 13 ciclos.

### Ejercicio 3

Dados la siguiente arquitectura y código en ensamblador:



1. Dado el programa anterior y un procesador con dos unidades funcionales de procesamiento, una especializada en los accesos a memoria y otra en las operaciones en coma flotante, determinar:
  - a. Cronograma de ejecución aproximada suponiendo que los *ld/sto* duran 1 ciclo, las operaciones aritméticas 4 ciclos y no hay solapamiento en el tiempo.
  - b. Cronograma de ejecución suponiendo que puede haber solapamiento temporal.
  - c. Cronograma suponiendo que existen 2 UF para la aritmética en coma flotante.
  - d. Cronograma suponiendo que existen 2 UF de acceso a memoria y 2 UF de aritmética en coma flotante.

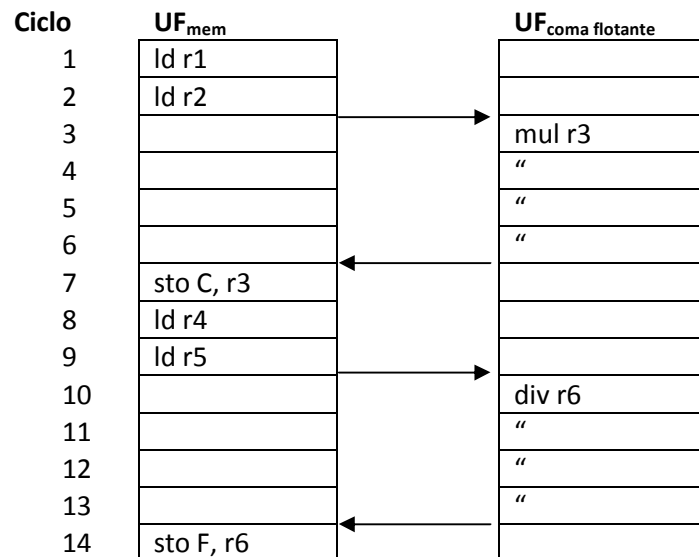


**Solución**

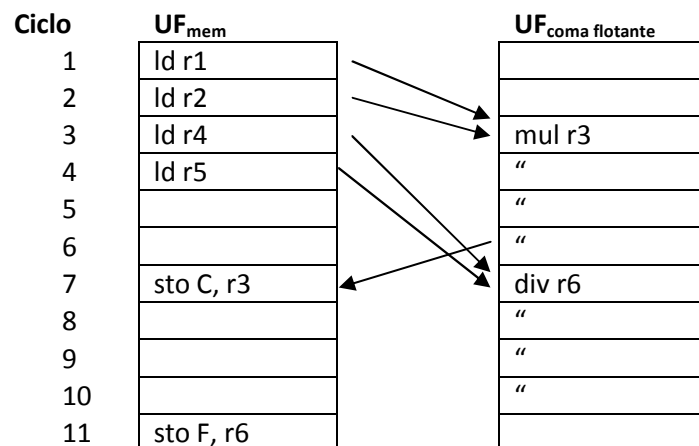
1.

a. Cronograma de ejecución aproximado.

Suponiendo que las instrucciones ld/sto duran 1 ciclo y las operaciones aritméticas 4 ciclos.  
 No hay solapamiento en el tiempo.

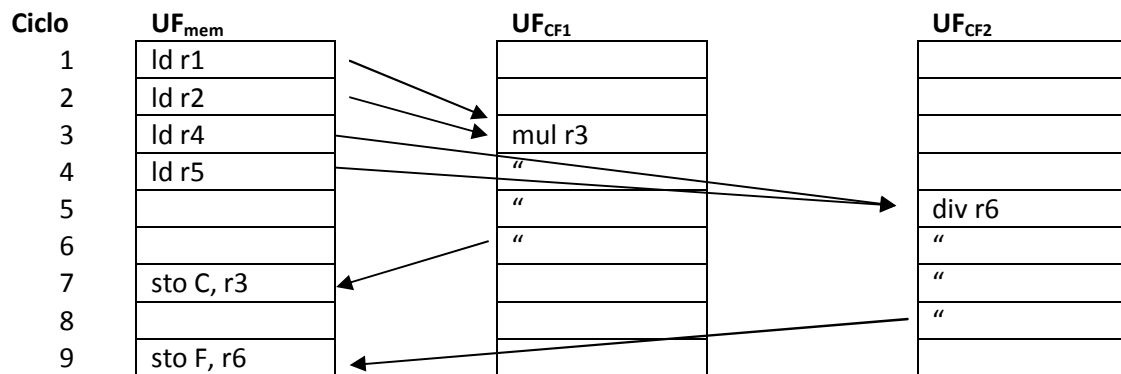


b. Cronograma de ejecución suponiendo que puede haber solapamiento temporal.





c. Cronograma suponiendo que existen 2 UF para la aritmética en coma flotante.



d. Cronograma suponiendo 2 UF de acceso a memoria y dos UF aritméticas (coma flotante).

