



Parcial Global 2025 – 24 Junio 2025

TEMA B

Apellido y Nombres:

L.U.:

Resolver los ejercicios en hojas separadas, indicando Apellido y Nombre – LU y número de hoja en cada una.

Resuelva el parcial considerando la siguiente tabla de latencias:

Operación	Latencia en ciclos	Cantidad de Unidades funcionales	Unidad funcional estructurada en pipeline
+, -, XOR	1	1	no
*, /	3	1	no

Figura 1

Ejercicio 1

Seleccione la respuesta correcta. En caso de seleccionar otro, justificar su respuesta.

En un pipeline de 5 etapas, ¿qué tipo de *hazard* puede ocurrir si no se usa *forwarding*?

- a) Data hazard
- b) Control hazard
- c) Structural hazard
- d) Cache miss
- e) Otro: _____

Dadas las siguientes instrucciones, suponiendo que se cuenta con un procesador de 5 etapas (F, D, E, M, WB) en orden CON forwarding:

$R2 = R3 * R4$ (4 ciclos en EX)
 $R5 = R2 + R6$ (1 ciclo en EX)

¿Cuántos ciclos de *stall* se necesitan?

- a) 0
- b) 1
- c) 2

R2=R3*R4	F	D	E	E	E	M	WB											
R5=M[R2]		F	D	DS	DS	E	M	WB										

- d) 4
- e) Otro: _____

¿Cuáles son las principales ventajas y desventajas de la arquitectura VLIW?

- a) El hardware es más complejo, pero permite una gran compatibilidad binaria.
- b) Permite ejecutar instrucciones de manera secuencial y sin control de flujo.
- c) Simplifica el hardware y mejora el rendimiento, pero requiere compiladores sofisticados y tiene problemas de



compatibilidad binaria.

d) Mejora la compatibilidad entre diferentes versiones de procesador y reduce el uso de registros.

Teniendo en cuenta las siguientes instrucciones y la tabla de latencias especificada en la Figura 1, desarrollar los siguientes incisos considerando un procesador fuera de orden que implementa el algoritmo de Tomasulo con pipeline de instrucciones de 6 etapas sin forwarding (F,DR,DS,E,WR,C). Completar el siguiente diagrama de Gantt.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
I0: R4=R1 x R4	F	DR	DS → E1	E2	E3	WR	C											
I1: R6=M[R4]		F	DR	DS	w	w	w → E	E	WR	c								
I2: R5=R4 x R6			F	DR	DS	w	w	w	w	w → E1	E2	E3	WR	c				

- a) ¿En qué ciclo la instrucción I2 reserva la estación de reservación? 4
- b) ¿En qué ciclo la instrucción I1 libera el buffer de Load/Store? 11

En un procesador superescalar $W=3$, si un programa tiene muchas instrucciones secuenciales con dependencias de datos, ¿qué comportamiento se espera?

- a) El procesador alcanzará el rendimiento teórico máximo de 3 instrucciones por ciclo.
- b) El procesador podrá ejecutar hasta 3 instrucciones por ciclo mediante reordenamiento.
- c)** El paralelismo disminuirá y se ejecutará menos de 3 instrucciones por ciclo.
- d) El rendimiento será idéntico al de un procesador RISC clásico.



Ejercicio 2:

Suponiendo que se cuenta con un procesador de 5 etapas (F, D, E, M, WB) en orden **con forwarding**. Asumir que para Load y Store la dirección de memoria se calcula en la etapa *Execute* en un sumador dedicado exclusivamente al cálculo de direcciones efectivas que tiene 1 ciclo de latencia para las unidades aritmético-lógicas. El acceso a memoria se realiza en la etapa *Memory* e *insume 1 ciclo*. Considerando que las escrituras de write-back se realizan en la primera mitad del ciclo, mientras que la etapa *decode* lee en el banco de registros en la segunda mitad del ciclo. Además, suponiendo que los valores de los registros: R9 = 2 y R10 = 7, y que el pipeline cuenta con predicción **tomado** y en la etapa *Memory* resuelve el salto. Completar el diagrama de Gantt de la *Figura 2* considerando las latencias mostradas en la *Figura 1*.

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
I1	R3 = R4 × R1	F	D	E1	E2	E3	M	WB															
I2	R5 = R3 × R10		F	D	DS	DS	E1	E2	E3	M	WB												
I3	JNZ R5, Eti			F	FS	FS	D	DS	E	M	WB												
I4	R3 = R3 + 7					F	-----																
I5	R5 = R7 + R3																						
I6	M[R3] = R5																						
I7	R5 = R5 + R0																						
I8 Eti	R9 = R3 + R6						F	D	E	M	WB												
I9	R3 = R9 × R5							F	D	E1	E2	E3	M	WB									

PRED TOMADO EN ETAPA MEMORY RESUELVE SALTO. VALOR R10=2, R4=1, R1=2. El salto SI se va a realizar

Ejercicio 3:

Un procesador cuenta con una memoria caché que implementa una organización 4-way set associative que se direcciona a nivel de byte. Considerando que la dirección tiene el formato que se muestra en la *Figura 3*, con 8 bits para el Index y 6 bits para el Offset. Además, suponiendo que la memoria principal tiene una capacidad de 32 MB, **determinar el tamaño del campo de TAG.**



Figura 3

- Offset = 6 bits. Para direccionar $2^6 = 64$ bytes (tamaño de la línea).
- Tengo que direccionar 32 MB, $33554432 \text{ bytes } (32 * 1024 * 1024) = 2^{25}$ o sea que necesitaremos 25 para la dirección de memoria. $25 - (8 + 6) = 11 \text{ bits para el tag.}$