



ARQUITECTURA DE COMPUTADORAS

Recuperatorio Global

Primer Cuatrimestre de 2024

Apellido y Nombres: L.U.:
D.N.I.: Cantidad de Hojas Entregadas (sin enunciado):

Resolver los ejercicios en hojas separadas, poniendo nombre y número a cada una.

Ejercicio 1

Dado el siguiente fragmento de código:

Suponiendo que se cuenta con un procesador de 5 etapas (F, D, E, M, WB) en orden, **sin** forwarding. Asumir que para Load y Store la dirección de memoria se calcula en la etapa Decode y 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 primer mitad del ciclo, mientras que la etapa decode lee en el banco de registros en la segunda mitad del ciclo. Suponiendo que los valores de los registros: $R_9 = 2$ y $R_{10} = 7$, y que el pipeline cuenta con predicción tomado. Desarrolle los siguientes incisos considerando las latencias mostradas en la Figura 1.

- Identifique todas las dependencias de datos para el fragmento de código dado.
- Desarrolle el diagrama de Gantt, calcule la cantidad de ciclos por instrucción promedio.
- Indique en qué ciclo se despacha la instrucción I_1 , justifique su respuesta además indique las consecuencias para el resto de las instrucciones.

Ejercicio 2

Teniendo en cuenta el código y la tabla de latencias especificados en el *Ejercicio 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 diagrama de Gantt correspondiente a la evolución de la secuencia de instrucciones para las instrucciones I_4, I_5, I_6, I_7 , además asumir que se cuenta con los valores disponibles de los registros R_0, R_3, R_6 y R_7 al inicio de la ejecución de I_4 .

Complete las tablas correspondiente al ROB y demás estructuras de la Figura 4.

- Indique en qué ciclo se actualiza la estación de reservación asignada a la instrucción I_5 con el valor del registro de R_3 calculado por I_4 . **Justifique su respuesta.**
- Indique en qué ciclo la instrucción I_6 accede a memoria, **justifique su respuesta.**
- Indique en qué ciclo puede comenzar a ejecutar la instrucción I_7 en la ALU, **justifique su respuesta.**

Ejercicio 3

Considere la secuencia de instrucciones del Ejercicio 1 desde la instrucción *I4* hasta *I9* y las latencias indicadas en la Figura 2, resolver los siguientes incisos asumiendo que la arquitectura cuenta con un procesador superescalares con **ejecución en orden**, el cual posee dos unidades de ejecución, es decir que se pueden despachar dos instrucciones (de cualquier tipo) por ciclo ($w=2$).

Para Load y Store asumir que se calcula la dirección de memoria efectiva en etapa Decode y se accede a memoria en etapa Memory. Ambas etapas demoran 1 ciclo.

- a. Esquematizar en un *diagrama de Gantt* para las instrucciones desde *I4* hasta *I9*. En los ciclos que se produzcan conflictos de datos o estructurales, identificar el conflicto e indicar los motivos que lo producen y el modo en que se resuelven según esta implementación.
- b. Calcule la cantidad de instrucciones por ciclo promedio para este caso.

Ejercicio 4

Dada una memoria caché de 32 KB (1 KB = 1024 bytes) con una organización 4-way set associative, que consta de 256 líneas de datos. Considerar que cada línea tiene una capacidad de 128 bytes y la memoria se direcciona a nivel de byte. Además, se cuenta con una memoria principal con capacidad de 2 Megabytes (1 MB = 1024 KB).

Indicar el formato de dirección de memoria principal en términos de los campos TAG, INDEX y OFFSET, indicando el tamaño de cada campo y justificando cómo se obtienen estos valores.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
I0 R2 = R5 * R8	F	D	E1	E2	E3	M	WB										
I1 M[R2] = R7		F	D	DS	DS	E	M	WB									
I2 R3 = R8 * R4																	
I3 R5 = R9 + 8																	
I4 R4 = M[R5]																	
I5 R9 = R8 * R3																	
I6 R8 = R3 + #8									F	D	E	M	WB				
I7 R10 = R9 - R3										F	D	DS	E	M	WB		
I8 R11 = R8 - R9											F	FS	D	E	M	WB	
I9 R12 = R8 + R4													F	D	E	M	WB

Figura 1: Gantt Ejercicio 1

Operación	Latencia en ciclos	Cantidad de unidades funcionales	Unidad funcional estructurada en pipeline
+, -, XOR	1	1	no
x, /	3	1	si

Figura 2: Tabla de latencias

Operación	Latencia en ciclos	Cantidad de unidades funcionales	Unidad funcional estructurada en pipeline
+, -, XOR	1	2	no
x, /	3	2	no

Figura 3: Tabla de latencias superescalar

Apellido y Nombres: L.U.:

- I4 R3 ← R3 + 7
- I5 Etí R5 ← R7 + R3
- I6 M[R3] ← R5
- I7 R5 ← R5 + R0

I	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
I ₄																				
I ₅																				
I ₆																				
I ₇																				

Register Map

Index	Map

ROB

T.I	Busy	TAG/Valor?	Data	Reg Log Dest	Ready_address	Ready_bit

BUFFER LOAD/STORE

Operación	TAG/Valor?	Dato	Offset	Ready_dir	Dirección	Destino

ER SUMA Y RESTA

Operando 1			Operando 2		
Operación	TAG/Valor?	Dato	TAG/Valor?	Dato	destino



ER MULTIPLICACIÓN Y DIVISIÓN

Operando 1			Operando 2		
Operación	TAG/Valor?	Dato	TAG/Valor?	Dato	destino



Figura 4: Tablas Tomasulo