

TEST DISEÑO DEL PROCESADOR: RENDIMIENTO

1. **Si el procesador A tiene una frecuencia de 1GHz y el procesador B una frecuencia de 2GHz, entonces el rendimiento de A es menor que el de B. Indica si es verdadero o falso y razona tu respuesta.**

Falso. Que un procesador tenga una mayor frecuencia no significa que vaya más rápido ya que hay más factores en juego, por ejemplo, las distintas arquitecturas de los procesadores, el repertorio de instrucciones, la cantidad de núcleos que tenga, etc.

2. **Si el procesador A tiene una frecuencia de 1GHz y el procesador B una frecuencia de 2GHz, y ambos tienen el mismo repertorio de instrucciones, entonces el rendimiento de A es menor que el de B. Indica si es verdadero o falso y razona tu respuesta.**

Falso. Como he comentado en la anterior respuesta también influyen en el rendimiento las diferentes arquitecturas de los procesadores y la cantidad de núcleos, por ejemplo.

3. **Si el procesador A tiene una frecuencia de 1GHz y el procesador B una frecuencia de 2GHz, y ambos tienen la misma microarquitectura, entonces el rendimiento de A es menor que el de B. Indica si es verdadero o falso y razona tu respuesta.**

Falso. Complementando a las respuestas anteriormente dadas, el tamaño de la memoria cache y el TDP (Thermal Design Power, es decir, la máxima potencia que puede usar) también pueden influir en el rendimiento del procesador.

TEST DISEÑO DEL PROCESADOR: OPERACIONES MULTICICLO

4. Dado un procesador segmentado en 5 etapas como el visto en clase con las siguientes características:

- Los datos se pueden leer y escribir en el banco de registros en el mismo ciclo de reloj
- Incorpora la técnica de forwarding o cortocircuito para la anticipación de operandos
- Dispone de un sumador/restador en punto flotante no segmentado con una latencia de 4 ciclos

Se pide calcular el CPI correspondiente a la ejecución del siguiente código.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
LD F0, 0(R1)	IF	ID	EX	M	WB												
LD F2, 8(R1)		IF	ID	EX	M	WB											
ADDD F4, F0, F2			IF	IDp	ID	Ad d1	Ad d2	Add3	Add4	M	WB						
SUBD F6, F0, F2				IFp	IF	IDp	IDp	IDp	ID	Sub1	Sub2	Sub3	Sub4	M	WB		
ADDI R1, R1, #8									IF	ID	EX	M	WB				
SD F6, 0(R2)										IF	IDp	IDp	IDp	ID	EX	M	WB

$$CPI = \frac{N^{\circ} \text{ ciclos}}{N^{\circ} \text{ instrucciones}} = \frac{17}{6} = 2.83$$

5. Dado el procesador segmentado de la cuestión anterior, si su frecuencia de reloj es 1GHz, ¿cuál es su rendimiento en MFLOPS al ejecutar el código anterior?

$$T = N * CPI * T_c = \frac{N * CPI}{Frecuencia} = \frac{6 * 2.83}{1000000000} = 1,698E - 8s = 1,698E - 5ms$$

$$MFLOPS = \frac{2}{1,698E - 5 * 10^6} = 0,117$$

6. Dado un procesador segmentado en 5 etapas como el visto en clase con las siguientes características:

- Los datos se pueden leer y escribir en el banco de registros en el mismo ciclo de reloj
- Incorpora la técnica de forwarding o cortocircuito para la anticipación de operandos
- Dispone de un sumador/restador en punto flotante segmentado con una latencia de 4 ciclos

¿Cuál es la diferencia en número de ciclos con respecto al mismo código ejecutado en el procesador de la actividad anterior?

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
LD F0, 0(R1)	IF	ID	EX	M	WB									
LD F2, 8(R1)		IF	ID	EX	M	WB								
ADDD F4, F0, F2			IF	IDp	ID	Add1	Add2	Add3	Add4	M	WB			
SUBD F6, F0, F2				IFp	IF	ID	Sub1	Sub2	Sub3	Sub4	M	WB		
ADDI R1, R1, #8						IF	ID	EX	M	WB				
SD F6, 0(R2)							IF	IDp	IDp	IDp	ID	EX	M	WB

Al tener sumadores/restadores en punto flotante segmentado podemos ejecutar varias operaciones de manera simultánea reduciendo así ciclos de reloj.

7. Compara las 2 soluciones estudiadas en clase para los riesgos EDE en función del número de ciclos de ejecución y de la seguridad que ofrecen.

En ambas soluciones existen paradas (IDp) para evitar el riesgo EDE para que los datos puedan almacenarse y llegar a las operaciones de manera adecuada. Todas las instrucciones tienen la

misma duración. Los registros se leen en la segunda etapa y se escriben las instrucciones en el BR en la última etapa.

TEST DISEÑO DEL PROCESADOR: GESTIÓN DE EXCEPCIONES

8. Para la gestión de excepciones, ¿qué modificaciones hacen falta en la ruta de datos para que la CPU pueda ejecutar la rutina de tratamiento de la excepción?

Para poder gestionar las excepciones hace falta añadir los siguientes elementos al procesador:

1. Registros "Cause" y EPC; dirección constante de salto a tratamiento de excepción "4000040"
2. Generación de las señales de control IF.Flush, ID.Flush, EX.Flush para vaciar el pipeline, además de EPCwrite para escribir en EPC, CauseWrite para escribir en "Cause" e IntCause para escribir un 1 sobre el bit apropiado de "Cause".

9. ¿Cómo detecta la rutina de tratamiento de excepción qué rutina de servicio en particular debe ejecutarse para resolver cada excepción?

Se añade una entrada al multiplexor que controla la carga del PC, para poder seleccionar la dirección de la rutina de tratamiento de excepción

10. ¿Qué implicaciones en cuestión de seguridad crees que puede tener el hecho de que no se atiendan las excepciones hasta el final de la ejecución de la instrucción que la produjo?

Si no se atienden las excepciones hasta el final de la ejecución puede provocar fallos en el sistema los cuales permitirían ejecutar instrucciones que no deseadas y también una posible pérdida de datos.

PROBLEMAS DISEÑO DEL PROCESADOR MULTICICLO

Problema 1: Dado un procesador segmentado en 5 etapas con las siguientes características:

- Los datos se pueden leer y escribir en el banco de registros en el mismo ciclo de reloj
- No incorpora la técnica de forwarding o cortocircuito para la anticipación de operandos
- Los saltos se resuelven en la etapa de decodificación y la siguiente instrucción empieza a ejecutarse y se descarta en caso de tomarse el salto

Se ejecuta en él una aplicación con las siguientes características:

- El 20% de las instrucciones son saltos condicionales, y de ellos se toman el 40%. No existen dependencias de datos con las instrucciones de salto.
- El 18% de las veces, la instrucción $li+1$ tiene una dependencia LDE con la instrucción li (30% son instrucciones load)
- El 6% de las veces, la instrucción $li+2$ tiene una dependencia LDE con la instrucción li (30% de ellas son instrucciones load). En estos casos no existe dependencias de datos entre las instrucciones $li+1$ y $li+2$

Se pide calcular:

a) CPI

Salto: Si se toma se pierde un ciclo. 20% de instrucciones son saltos, se toman el 40%

$\rightarrow 0.2 * 0.4 = 0.08$ instrucciones tardan 1 ciclo más

Dependencias LDE en la inst. $li+1$ penalizan 2 ciclos $\rightarrow 0.18$ instrucciones

Dependencias LDE en la inst. $li+2$ penalizan 1 ciclo $\rightarrow 0.06$ instrucciones

$CPI_{ideal} = 1$

$$CPI = 1 + \sum_{i=1}^n (frec_i * penalizacion_i) = 1 + 0.08 * 1 + 0.18 * 2 + 0.06 * 1 = 1.5$$

b) CPI tras añadir la técnica de forwarding o cortocircuito

Salto: igual que en el apartado a)

Dependencias LDE en la inst. $li+1$: Solo afectan al load $\rightarrow 1$ ciclo de penalización

30% de estas instrucciones son load $\rightarrow 0.18 * 0.3 = 0.054$ instrucciones paran 1 ciclo

$$CPI = 1 + 0.08 * 1 + 0.054 * 1 = 1.134$$

c) Speedup del procesador a) frente al b)

$$a) Speedup = \frac{1.5}{1.134} = 1.32 \rightarrow CPI_a = 1.32$$

$$b) Speedup = \frac{1.134}{1.5} = 0.756 \rightarrow CPI_b = 0.756$$

Problema 2. Dado un procesador segmentado en 5 etapas con las siguientes características:

- Los datos se pueden leer y escribir en el banco de registros en el mismo ciclo de reloj
- La detección de riesgos y paradas del pipeline se realizan en la etapa de decodificación
- Incorpora la técnica de forwarding o cortocircuito para la anticipación de operandos
- Los saltos se resuelven en la etapa de decodificación y se resuelven mediante paradas
- Los riesgos estructurales referentes a la etapa de acceso a memoria se resuelven mediante paradas en el último ciclo de la etapa de ejecución
- Los riesgos EDE se resuelven mediante la inhibición de la escritura de la primera instrucción
- 1 sumador en punto flotante segmentado de latencia 2 ciclos
- 1 multiplicador en punto flotante segmentado de latencia 5 ciclos.

Se ejecuta el siguiente programa:

```
LD      F10, 0(R1)
MULD    F4, F0, F10
LD      F12, 0(R2)
ADDD    F2, F12, F4
LD      F4, 8(R1)
MULD    F12, F4, F12
LD      F12, 16(R1)
```

- a) Representa el diagrama de instrucción/tiempo señalando los cortocircuitos
b) Calcula el CPI

$$CPI = \frac{N^{\circ} \text{ ciclos}}{N^{\circ} \text{ instrucciones}} = \frac{20}{7} = 2.86$$

<u>Ejercicio 2.a</u>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
LD F10, 0(R1)	IF	ID	EX	M	WB															
MULD F4, F0, F10		IF	IDp	ID	MUL1	MUL2	MUL3	MUL4	MUL5	M	WB									
LD F12, 0(R2)			IFp	IF	ID	EX	M	WB												
ADDD F2, F12, F4					IF	IDp	IDp	IDp	ID	ADD1	ADD2	M	WB							
LD F4, 8(R1)						IFp	IFp	IFp	IF	ID	EXp	EX	M	WB						
MULD F12, F4, F12										IF	IDp	IDp	ID	MUL1	MUL2	MUL3	MUL4	MUL5	M	WB
LD F12, 16(R1)											IFp	IFp	IF	ID	EX	M	WB			