

Arquitectura e Ingeniería de Computadores. Examen Parcial (Problemas). 10/02/2009

1) En un DLX con segmentación ejecutamos el siguiente fragmento de código:

```

    ADDI r3,r0,#3
L1 : SUBD F2,F6,F8
      SUBD F4,F8,F6
      SD 0(r3),F4
      DIVD F2,F4,F8
      ADDD F2,F8,F8
      SUBI r3,r3,#1
      DIVD F6,F4,F8
      MULD F4,F2,F6
      SUBD F10,F2,F6
      LD F4,0(r3)
      ADDD F0,F4,F2
      BNEZ r3,L1
      MULD F4,F2,F2
end
    
```

Se supone que:

- Un dato se puede escribir en un registro y leer su valor en el mismo ciclo.
- Se dispone de lógica de cortocircuito (*forwarding*).
- Los saltos se resuelven en la etapa de decodificación.
- La detección de todo tipo de riesgos (estructurales y LDE) y generación de paradas se realiza en la etapa de decodificación.
- Los riesgos EDE se resuelven mediante inhibición de escritura.
- Se dispone de un predictor de saltos de 2 bits del tipo BTB, al que se accede en la etapa IF, obteniendo la respuesta al final de dicha fase. El estado inicial del predictor es “salto no tomado fuerte”.
- Dos instrucciones no pueden acceder simultáneamente a la etapa de acceso a memoria ni tampoco a la de escritura en el banco de registros.
- Se dispone de las siguientes unidades funcionales:

UF	Cantidad	Latencia	Segmentación
FP ADDD	1	3	Sí
FP SUBD	1	3	Sí
FP MULD	1	4	Sí
FP DIVD	1	5	No
INT ALU	1	1	No

- A) Representar el diagrama instrucción-tiempo para la primera iteración e indicar los cortocircuitos realizados. Indicar claramente las paradas y sus causas. A la vista del diagrama obtenido, y sin necesidad de desarrollar todas las iteraciones, calcular el número de ciclos que toma la ejecución completa del código. **(3.5 ptos)**
- B) ¿Cómo se vería afectado el número de ciclos resultantes del apartado anterior si el estado inicial del predictor fuese ahora “salto tomado fuerte”? Indicar el comportamiento del salto (fallo/acierto) y la nueva predicción resultante en cada iteración del bucle. **(1.5 ptos)**

2) Se tienen dos configuraciones de memoria cache:

- Una cache asociativa por conjuntos de 4 vías, de 32 KB unificada que usa post escritura sin bit de sucio y asignación en escritura. Esta configuración tiene una tasa de fallos de 0,01.
- Una cache directa unificada de 32 KB que usa post-escritura con bit de sucio y asignación en escritura. Esta configuración tiene una tasa de fallos de 0,03.

Suponiendo que:

- I. El procesador tiene un CPI ideal de 2.
- II. La latencia de memoria son 60 ciclos.
- III. Las transferencias de bloques se realizan a 8 bytes/ciclo.
- IV. El 35% de los bloques se modifican.
- V. El tamaño de bloque es de 64 bytes.
- VI. El 20% de las instrucciones son de acceso a memoria.

A) Calcular los CPI para las dos configuraciones. **(3.5 ptos)**

Si ahora suponemos que es una cache de direcciones virtuales con un TLB de las siguientes características:

- I. 3% de tasa de fallos.
- II. El fallo de TLB produce una penalización de 15 ciclos.

B) Calcular los nuevos CPI teniendo en cuenta el TLB. **(1.5 ptos)**

SOLUCIÓN

1)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	
ADDI, r3,r0,#3	IF	ID	EX	M	WB																										
SUBD F2,F6,F8		IF	ID	S1	S2	S3	M	WB																							
SUBD F4,F8,F6			IF	ID	S1	S2	S3	M	WB																						
SD 0(r3),F4				IF	ID	ID	ID	EX	M	WB																					
DIVD F2,F4,F8					IF	IF	IF	ID	D1	D2	D3	D4	D5	M	IW																
ADDD F2,F8,F8								IF	ID	A1	A2	A3	M	WB																	
SUBI r3, r3,#1									IF	ID	EX	M	WB																		
DIVD F6,F4,F8										IF	ID	ID	ID	D1	D2	D3	D4	D5	M	WB											
MULD F4,F2,F6											IF	IF	IF	ID	ID	ID	ID	ID	M1	M2	M3	M4	M	WB							
SUBD F10,F2,F6														IF	IF	IF	IF	IF	ID	ID	S1	S2	S3	M	WB						
LD F4,0(r3)																			IF	IF	ID	ID	ID	EX	M	WB					
ADDD F0,F4,F2																						IF	IF	IF	ID	ID	A1	A2	A3	M	WB
BNEZ r3,L1																							IF	IF	ID	EX	M	WB			
MULD F4,F2,F2																											IF	X	X	X	...
end																							L1	SUB		IF	ID	S1	...		

XX: Parada por riesgo LDE

XX: Parada por riesgo estructural (siguiente etapa ocupada)

XX: Riesgo estructural: unidad no segmentada

XX: Riesgo EDE (inhibición de escritura)

XX: Riesgo estructural: dos instrucciones intentan acceder simultáneamente a memoria

A) La primera vez que se llega al salto (r3=2), la predicción es NTF, pero el salto se toma, por lo que se falla y se debe cancelar la instrucción iniciada (MULD F4,F2,F2). Predictor pasa a NTD

La segunda vez que se llega al salto (r3=1), la predicción es NTD, pero el salto nuevamente se toma, por lo que falla la predicción y por tanto se cancela la misma instrucción que en el caso anterior. El predictor pasa a TD

La tercera vez que se llega al salto (r3=0), la predicción es TD, pero ahora el salto no se toma, por lo que se vuelve a fallar y la instrucción cancelada en este caso es la correspondiente a la etiqueta L1. Posteriormente se ejecuta la última instrucción del código y el programa finaliza.

Ciclos: la ejecución del salto en su primera iteración comienza en el ciclo 25 y como el predictor falla hay un ciclo de penalización de manera que el fetch de SUBD se hace en el ciclo 27.

El fetch del salto en su segunda iteración se hará en el ciclo 50, por lo que debido al ciclo de penalización que introduce el fallo del predictor el fetch de SUBD se hace en el ciclo 52.

Finalmente, el fetch del salto (3ª iteración) se hace en el ciclo 75, por lo que la última instrucción del código se inicia en el ciclo 77, finalizando ésta en el ciclo 84.

TOTAL CICLOS = 84 (26(1ªit.)+25(2ªit.)+25(3ªit.)+8(última instrucción))

B) Si partimos del estado TF, en las dos primeras iteraciones del salto acertaríamos, por lo que no habría 2 ciclos de penalización, y no se modificaría el estado del predictor. En la última iteración fallaríamos (el salto no se toma y el predictor pasa a TD) por lo que sí tendríamos el ciclo de penalización al igual que en el apartado anterior. Conclusión, nos ahorraríamos 2 ciclos y por tanto

TOTAL CICLOS = 82

2)

A)

El ciclo por instrucción medio de acceso a la jerarquía de memoria viene dado por la siguiente expresión

$$CPI = CPI\ ideal + tasa\ de\ acceso \times tasa\ de\ fallos \times Penalización\ por\ fallo.$$

Como dan la tasa de fallo de las dos organizaciones no es necesario tener en cuenta ni el tamaño ni la política de emplazamiento, puesto que en la tasa de fallos ya van implícitos estos datos.

Para las dos organizaciones los tiempos de lectura y de escritura de un bloque vienen dados por la siguiente expresión:

$$Tiempo\ lectura\ o\ escritura\ de\ bloque = latencia + (tamaño\ bloque / velocidad\ de\ transf.)$$

La latencia hace referencia al tiempo que se tarda en recuperar la primera palabra y el ancho de banda determina el tiempo que tarda en traerse el resto del bloque.

$$Tiempo\ lectura\ o\ escritura\ de\ bloque = 60 + (64 / 8) = 68\ ciclos$$

Sin bit dirty.

Si no existe bit dirty hay que salvar siempre el bloque que se va a reemplazar antes de traer el nuevo bloque. Esto ocurre tanto para los fallos de lectura como para los fallos de escritura.

$$Penalización\ por\ fallo\ de\ lectura: \textit{Escritura\ bloque} + \textit{lectura\ bloque}$$

$$Penalización\ fallos\ de\ escritura: \textit{escritura\ bloque} + \textit{lectura\ bloque}$$

$$Penalización\ media\ de\ memoria = tasa\ de\ accesos\ a\ memoria \times tasa\ fallos \times penalización = 1,2 \times 0,01 \times (2 \times 68) = 1,632$$

$$CPI = 2 + 1,632 = 3,632$$

Con bit dirty

Igual que antes la penalizaciones por lectura y escritura son idénticas. Pero ahora hay que tener en cuenta el bit dirty

$$Penalización\ con\ dirty\ activado = \textit{escritura\ bloque} + \textit{lectura\ bloque}$$

$$Penalización\ con\ dirty\ desactivado = \textit{lectura\ bloque}$$

Entonces:

$$Penalización\ media\ de\ memoria = 1,2 \times 0,03 \times (0,35 \times 2 \times 68 + 0,65 \times 1 \times 68) = 3,3048$$

$$CPI = 2 + 3,3048 = 5,3048$$

B)

En estos casos solo se accede al TLB en caso de fallo de cache. La tasa de fallos a los que se aplica la penalización de la TLB es tasa fallos cache x tasa fallos TLB. Luego la penalización por fallo en TLB es

• **Para la primera organización**

$$Penalización\ media\ adicional = 1,2 \times 0,01 \times 0,03 \times 15$$

$$CPI = 3,632 + 1,2 \times 0,01 \times 0,03 \times 15 = 3,6374$$

- **Para la segunda organización**

Penalización media adicional = $1,2 \times 0,03 \times 0,03 \times 15$

CPI = $5,3048 + 1,2 \times 0,03 \times 0,03 \times 15 = 5,321$