

Ejercicios Tema 3: Gestión de Memoria.

1. **Calcular el tiempo de acceso medio a memoria de una máquina con una frecuencia de reloj de 400 Mhz, una penalización de fallo de 20 ciclos de reloj, una tasa de fallos de 0.05 fallos por instrucción y un tiempo de acceso a la cache de 1 ciclo de reloj (incluido el tiempo para la detección de acierto). Suponer que las penalizaciones de fallos en lectura y escritura son iguales.**

$$400 \text{ Mhz} = 1/4 \cdot 10^8 = 2.5\text{ns por ciclo}$$

$$T_a = T_{\text{acierto}} \cdot \text{Frec}_{\text{acierto}} + T_{\text{fallo}} \cdot \text{Frec}_{\text{fallo}} = 1 \cdot 2.5\text{ns} \cdot 0.95 + 20 \cdot 2.5\text{ns} \cdot 0.05 = 4,875 \text{ ns.}$$

2. **Suponer que en la caché anterior mejoramos la tasa de fallos de 0.05 a 0.03 fallos por instrucción duplicando el tamaño de la memoria caché. Esto causa que el tiempo de acceso a la caché pase a ser de 1.2 ciclos de reloj. En base al tiempo de acceso medio a la memoria, ¿crees que es una buena solución?**

Sí lo es porque el tiempo de acceso medio pasa a ser 4.41 ns que es menor que en el caso anterior. $T_a = 1.2 \cdot 2.5 \cdot 0.97 + 20 \cdot 2.5 \cdot 0.03 = 4,41 \text{ ns.}$

3. **¿Cuántos bits se necesitan para realizar una cache de correspondencia directa con 64KiB de datos? Supóngase bloques de una palabra, direccionamiento a nivel de bytes y ancho del bus de datos y direcciones de 32 bits.**

Se sabe que 64KiB son 16 K palabras ($2^9 / 2^5 = 2^4$), que son 2^{14} palabras y, con puesto que el tamaño de bloque es equivalente a una palabra, dispondremos por tanto de 2^{14} bloques.

Cada bloque tiene 32 bits de datos más la etiqueta, que tiene, en función del tipo de caché determinado por el enunciado (correspondencia directa) $32 - 14 - 2$ bits (bits dirección – bits bloque – bytes por palabra), más un bit de validez y otro de suciedad. Por tanto el tamaño de la caché es:

$$2^{14} \cdot (32 + (32 - 14 - 2) + 1 + 1) = 2^{14} \cdot 50 = 800 \cdot 2^{10} = 800 \text{ Kibits}$$

Ó 100 KiB para una caché de 64 KiB.

Por tanto, para esta cache, el número total de bits en la cache es más de 1.5 veces más de lo necesario para almacenar los datos.

4. **Dada la siguiente secuencia de referencias a palabras de una memoria de 256 MiB y suponiendo una memoria caché de correspondencia directa de 16 bloques y 4 bytes por bloque que inicialmente está vacía, etiquetar cada referencia de la lista como un acierto o un fallo y mostrar el contenido final de la cache.**

1, 4, 8, 5, 20, 17, 19, 56, 9, 11, 4, 43, 5, 6, 9, 17

Nota: Estos valores marcan indican la *dirección de memoria en formato decimal*, **obviando los bits de desplazamiento.**

En primer lugar obtenemos los valores necesarios para definir el contenido de la caché:

16 bloques => **4 bits** indican el bloque

4 bytes por bloque => **2 bits** indican el byte dentro del bloque (en este caso se omitirá)

256MiB de memoria => 2^{28} bytes / 2^2 bytes/bloque = 2^{26} bloques

26 (MP) – 4 (Cache) = **22 bits** para definir el bloque de memoria dentro de cada línea de caché.

Referencia de palabra	Referencia de byte (en binario)	Acierto/Fallo	Nº bloque
1	(0...00000000) (0001) (00)	Fallo	1
4	(0...00000000) (0100) (00)	Fallo	4
8	(0...00000000) (1000) (00)	Fallo	8
5	(0...00000000) (0101) (00)	Fallo	5
20	(0...00000001) (0100) (00)	Fallo	4
17	(0...00000001) (0001) (00)	Fallo	1
19	(0...00000001) (0011) (00)	Fallo	3
56	(0...00000011) (1000) (00)	Fallo	8
9	(0...00000000) (1001) (00)	Fallo	9
11	(0...00000000) (1011) (00)	Fallo	11
4	(0...00000000) (0100) (00)	Fallo	4
43	(0...00000010) (1011) (00)	Fallo	11
5	(0...00000000) (0101) (00)	Acierto	5
6	(0...00000000) (0110) (00)	Fallo	6
9	(0...00000000) (1001) (00)	Acierto	9
17	(0...00000001) (0001) (00)	Acierto	1

Contenido final de la caché:

Nº Bloque	Validez	Tag	Datos (en su lugar referencia de palabra)
0	0	-	-
1	1	0...0000 0001 = 1	17
2	0	-	-
3	1	0...0000 0001 = 1	19
4	1	0...0000 0000 = 0	4
5	1	0...0000 0000 = 0	5
6	1	0...0000 0000 = 0	6
7	0	-	-
8	1	0...0000 0011 = 3	56
9	1	0...0000 0000 = 0	9
10	0	-	-
11	1	0...0000 0010 = 2	43
12	0	-	-
13	0	-	-
14	0	-	-
15	0	-	-

5. Sea un sistema con una memoria principal de 1Mpalabras de tamaño, con unos tiempos de acceso de 100ns y dividida en bloques de 2 palabras. Sabiendo que el sistema tiene una memoria caché asociativa por conjuntos de 4 vías con 256 palabras y un tiempo de acceso de 10ns (búsqueda + acceso), calcular:

- a) Número de bits para direccionar memoria principal (Mp)

Si la memoria tiene un tamaño de 1Mpalabras, necesitaremos acceder a 2^{20} posiciones y por tanto el ancho de las direcciones físicas es de 20 bits

- b) Número de conjuntos de la memoria caché

Como el grado de asociatividad es 4, disponemos de 4 bloques por conjunto. Dado que la memoria caché dispone de $2^8/2=2^7$ líneas de caché, el número de conjuntos será:

$$v = 2^7/4 = 32 \text{ conjuntos}$$

- c) Formato de las direcciones de memoria caché y tamaño de etiqueta

Para acceder a 32 conjuntos de caché necesitamos 5 bits. Para direccionar r entre 2 palabras/bloque necesitamos 1 bit. Por tanto, la etiqueta tiene una longitud de $20 - 5 - 1 = 14$ bits.

- d) Mejora del rendimiento del sistema si la tasa de fallos es del 25%

El rendimiento sin memoria caché será de 100 ns para cada acceso. Si incorporamos la memoria caché será:

$$T_{\text{acc}} = 0.75 \cdot 100 \text{ ns} + (1-0.75) \cdot (10 \text{ ns} + 100 \text{ ns}) = 35 \text{ ns}.$$

6. Sea un computador con una memoria caché con 8 líneas de 2 bytes cada una con correspondencia asociativa por conjuntos de 2 vías. La operación de búsqueda de caché es por demanda, mientras que el reemplazo será tipo FIFO. La actualización de la memoria principal es mediante post-escritura bajo fallo. La memoria principal tiene una capacidad de 32B, siendo el tamaño de palabra igual a 8 bits.

Si definimos $R(x,d)$ como la lectura sobre la variable x del contenido de la posición de memoria de dirección d, y $W(x,d)$ la escritura de la variable x sobre la posición de memoria d, analizar cómo evoluciona el contenido de la memoria principal y de la caché cuando el procesador emite la siguiente secuencia de instrucciones:

$$R(x0, 17), R(x1, 6), W(100, 8), W(101, 8), R(x2, 9), R(x3, 25)$$

SOLUCION: Longitud dirección memoria física = 2^5 palabras \rightarrow 5 bits; (1 offset) + (2 conjunto) + (2 etiq.)

Accesos: Todos dan fallo salvo $W(101, 8)$ y $R(x2, 9)$;

$R(x3, 25)$ reemplaza la línea 0 del conjunto 0