



IIC2343 - Arquitectura de Computadores (II/2025)

Ayudantía 13

Ayudantes: Daniela Ríos (danielaarp@uc.cl), Alberto Maturana (alberto.maturana@uc.cl), Ignacio Gajardo (gajardo.ignacio@uc.cl)

Pregunta 1: Preguntas Conceptuales

- (a) [P1.B | I3-2024-2] ¿Cómo incide el parámetro “ N ” en el *hit-time* de una memoria caché *N-way associative*?

Solución: El parámetro N corresponde a la cantidad de líneas por conjunto en una caché *N-way associative*. Luego, el *hit-time* en este tipo de caché corresponde al tiempo de búsqueda de un bloque en las líneas del conjunto al que se mapea. Por ende, a mayor N , mayor será el *hit-time* al tener que revisar una mayor cantidad de líneas en el conjunto.

- (b) [P1.C | I2-2023-2] Suponga que posee un computador que hace uso de una caché *directly mapped* y una persona, externa al curso, le dice que su tiempo promedio de acceso a memoria sería **definitivamente** menor si es que su caché fuera de tipo *fully associative*. ¿Es esto cierto? Justifique su respuesta en base a los criterios que inciden sobre el tiempo promedio de acceso a memoria.

Solución: La oración anterior **no es cierta**. Si bien las funciones de correspondencia *fully associative* hacen un uso completo de la caché y pueden tener un *hit rate* mayor, su *hit time* es también mayor al tener que realizar una búsqueda en todas sus líneas para verificar la existencia de un bloque de memoria. Entonces, si la función de correspondencia no incidiera en el *hit rate* para una secuencia de accesos determinada, el tiempo promedio para una caché *directly mapped* sería menor.

- (c) [P1.B | I3-2024-1] Suponga que posee dos computadores con arquitecturas similares, salvo por sus memorias caché: poseen las mismas dimensiones, pero una es *16-way associative* y la otra es *8-way associative*. ¿Cuál presenta un *hit-time* promedio menor? Justifique su respuesta.

Solución: Considerando que ambas memorias caché poseen las mismas dimensiones, la que presenta un menor *hit-time* promedio corresponde a la *8-way associative*. Esto se

debe a que la búsqueda en memorias caché *N-way associative* se realiza solo sobre las líneas del conjunto al que se mapea el bloque de la palabra buscada. Como N equivale a la cantidad de líneas por conjunto, en una caché *8-way associative* la búsqueda se realiza sobre la mitad de las líneas de una *16-way associative*, lo que implica un menor tiempo para determinar si hay *hit* o no.

Pregunta 2: Caché - N-Way Associative y LRU (P1.C | I3-2024-1)

Suponga que, para una memoria principal de 256 palabras de 1 byte, posee una caché de 16 líneas y 4 palabras por línea con el siguiente estado intermedio:

Línea	Validez	Tag	Timestamp
0	1	20	1
1	1	5	11
2	1	10	4
3	1	25	9
4	1	31	8
5	1	16	3
6	1	1	6
7	1	8	2

Línea	Validez	Tag	Timestamp
8	0	10	-
9	0	20	-
10	0	30	-
11	0	5	-
12	0	15	-
13	0	2	-
14	0	1	-
15	0	0	-

La columna “Línea” representa el índice de línea; “Validez” representa el *valid bit*; “Tag” representa el valor decimal del *tag* que indica el bloque de memoria almacenado en la línea; y “Timestamp” representa el tiempo **desde el último acceso a la línea**. A partir de este estado, se realiza el acceso a memoria de las siguientes direcciones: 0x54, 0xE4, 0xE3, 0xCA, 0xCC, 0xA1. Indique, para cada acceso, si existe un *hit* o *miss*; si corresponde, la línea de la caché que es modificada; y el *hit-rate* para la función de correspondencia *8-way associative*. Asuma una política de reemplazo LRU, si corresponde (*i.e.* se reemplaza la línea con mayor *timestamp*).

Para responder, complete las tablas adjuntas al enunciado. El *tag* de cada línea lo puede escribir en base binaria o decimal. No necesita indicar el *timestamp* final por línea, pero sí debe considerar su valor en caso de reemplazos.

Nº Acceso	Dirección	Tag	Hit/Miss	ID Línea modificada	Comentarios
0	0x54				
1	0xE4				
2	0xE3				
3	0xCA				
4	0xCC				
5	0xA1				

Hit-rate:

Solución: Al tener una memoria de 256 palabras, se necesitan $\log_2(256) = 8$ bits para cada dirección. Por otra parte, al ser las líneas de 4 palabras, se requiere $\log_2(4) = 2$ bits de *offset* para ubicar una palabra dentro de una línea. Por otra parte, en una caché 8-way *associative* se tienen conjuntos de 8 líneas, lo que deriva en un total de $\frac{16}{8} = 2$ conjuntos y, de esta forma, $\log_2(2) = 1$ bit de índice de conjunto. Esto, finalmente, resulta en $8 - 1 - 2 = 5$ bits de *tag*. En este caso, el conjunto 0 corresponde a las líneas 0-7 y el conjunto 1 a las líneas 8-15.

Con esto en consideración, se obtiene el siguiente resultado de la secuencia de accesos a memoria:

1. Acceso a dirección $0x54 = 84 = 01010|1|00b$. Resulta en *miss* al no existir línea válida con *tag* $01010b = 10$ en el conjunto 1. Se copia el bloque sobre la línea 8.
2. Acceso a dirección $0xE4 = 228 = 11100|1|00b$. Resulta en *miss* al no existir línea válida con *tag* $11100b = 28$ en el conjunto 1. Se copia el bloque sobre la línea 9.
3. Acceso a dirección $0xE3 = 227 = 11100|0|11b$. Resulta en *miss* al no existir línea válida con *tag* $11100b = 28$ en el conjunto 0. Se copia el bloque sobre la línea 1, al ser esta la de mayor *timestamp*. Ahora, la línea 3 es la que posee mayor *timestamp*, en caso de requerir un reemplazo.
4. Acceso a dirección $0xCA = 202 = 11001|0|10b$. Resulta en *hit*, ya que la línea 3 posee *tag* igual a $11001b = 25$ en el conjunto 0. Ahora, la línea 4 es la que posee mayor *timestamp*, en caso de requerir un reemplazo.
5. Acceso a dirección $0xCC = 204 = 11001|1|00b$. Resulta en *miss* al no existir línea válida con *tag* $11001b = 25$ en el conjunto 1. Se copia el bloque sobre la línea 10.
6. Acceso a dirección $0xA1 = 161 = 10100|0|01b$. Resulta en *hit*, ya que la línea 0 posee *tag* igual a $10100b = 20$ en el conjunto 0.

A continuación, se muestra el estado final de la caché:

Línea	Validez	Tag
0	1	20
1	1	28
2	1	10
3	1	25
4	1	31
5	1	16
6	1	1
7	1	8

Línea	Validez	Tag
8	1	10
9	1	28
10	1	25
11	0	-
12	0	-
13	0	-
14	0	-
15	0	-

Hit-rate: $\frac{2}{6} = 0.\bar{3} = 33, \bar{3} \%$

Pregunta 3: Caché: Fully Associative y LRU

Suponga que, para una memoria principal de 256 palabras de 1 byte, posee el siguiente estado en una caché de 8 líneas y 2 palabras por línea, cada una de 1 byte:

Línea	Validez	Timestamp	Tag
0	1	2	5
1	1	7	15
2	0	5	9
3	1	0	1
4	1	3	3
5	1	6	9
6	1	4	5
7	1	1	8

La columna “Línea” representa el índice de línea; “Validez” representa el *valid bit*; “Timestamp” representa el tiempo **desde el último acceso a la línea**; y “Tag” representa el valor, en base decimal, del *tag* que indica el bloque de memoria almacenado en la línea. Con este estado de caché, se realiza el acceso a memoria a las siguientes direcciones: 0x8F, 0x17, 0x11, 0x3D, 0xF5, 0x16. Indique, para cada acceso, si existe un *hit* o *miss*; si corresponde, la línea de la caché que es modificada; y el *hit-rate* para la función de correspondencia *Fully Associative*. Asuma una política de reemplazo LRU, si corresponde (*i.e.* se reemplaza la línea con mayor *timestamp*).

Para responder, complete las tablas adjuntas al enunciado. El *tag* de cada línea lo puede escribir en base binaria o decimal. No necesita indicar la actualización de *timestamp*, pero sí debe considerar su valor en caso de reemplazos.

N° Acceso	Dirección	Tag	Hit/Miss	ID Línea modificada	Comentarios
0	0x8F				
1	0x17				
2	0x11				
3	0x3D				
4	0xF5				
5	0x16				

Hit-rate:

Solución: Al tener una memoria de 256 palabras se necesitan $\log_2(256) = 8$ bits para cada dirección. Por otra parte, al ser las líneas de 2 palabras, se requiere $\log_2(2) = 1$ bit de *offset* para ubicar una palabra dentro de una línea. Con esto en consideración, se tiene el siguiente resultado:

Para *Fully associative*, solo se es necesario el bit de *offset* y los demás serán los bits del *tag*, resultando en $8 - 1 = 7$ bits para el *tag*. A continuación, la tabla de accesos resultante:

N° Acceso	Dirección	Tag	Hit/Miss	ID Línea modificada	Comentarios
0	0x8F	1000111b = 71 = 0x47	Miss	10b = 2	No existe coincidencia de <i>tag</i> . El contenido se escribe en la única línea disponible, que corresponde a la 2. La línea de mayor <i>timestamp</i> es la 1.
1	0x17	1011b = 11 = 0xB	Miss	1b = 1	No existe coincidencia de <i>tag</i> . Como no hay espacio libre se reemplaza la línea de mayor <i>timestamp</i> , en este caso, la línea 1. La nueva línea de mayor <i>timestamp</i> es la 5
2	0x11	1000b = 8 = 0x8	Hit	-	El <i>tag</i> coincide con el de la línea 7 y su bit de validez es 1, por lo que hay hit. La nueva línea de mayor <i>timestamp</i> sigue siendo la 5.
3	0x3D	11110b = 30 = 0x1E	Miss	101b = 5	No existe coincidencia de <i>tag</i> . Se reemplaza la línea de mayor <i>timestamp</i> , en este caso, la línea 5. La nueva línea de mayor <i>timestamp</i> es la 6.
4	0xF5	1111010b = 122 = 0x7A	Miss	110b = 6	No existe coincidencia de <i>tag</i> . Se reemplaza la línea de mayor <i>timestamp</i> , en este caso, la línea 6. La nueva línea de mayor <i>timestamp</i> es la 4.
5	0x16	1011b = 11 = 0xB	Hit	-	El <i>tag</i> coincide con el de la línea 1 y su bit de validez es 1, por lo que hay <i>hit</i> . La nueva línea de mayor <i>timestamp</i> sigue siendo la 4.

Hit-rate: $\frac{2}{6} = 0.\bar{3} = 33.\bar{3} \%$

Feedback ayudantía

Escanee el QR para entregar *feedback* sobre la ayudantía.

