

(a) Direcciones virtuales

Dado que la memoria es de 128Kbytes (2^{17} bytes), direccionable sólo por palabras, con palabras de 16 bits cada una, tenemos que la dirección de cada palabra es de $128\text{Kbytes}/2(\text{bytes/palabra})=64\text{K}$ palabras \rightarrow 16 bits de dirección física de la memoria principal. Con esto podemos armar las direcciones virtuales:

1. Correspondencia directa:

Como es una caché de 8 bloques (líneas) de correspondencia directa, se usan 3 bits para direccionar cada bloque. En este caso, la dirección virtual queda de la siguiente forma:

Etiqueta	Bloque
13 bits	3 bits

2. Correspondencia Completamente Asociativa:

Como es tradicionalmente, la dirección física de la memoria se corresponde con una dirección virtual compuesta de sólo etiqueta.

Etiqueta
16 bits

3. Correspondencia Asociativa por Conjuntos, de 2 conjuntos

Se requiere de un bit para seleccionar el conjunto, en ese caso la dirección virtual queda:

Etiqueta	Conjunto
15 bits	1 bit

(b) Tablas

Para la corrida con la caché de mapeo directo, la caché cambia como se muestra en la figura 1. El reemplazo se realiza de acuerdo al número de línea que indiquen los 3 últimos bits de la dirección especificada. Por ejemplo, para $j=1$, se está escribiendo en j (posición $0x7A2B$), en suma (posición $0x7A28$) y se está leyendo $A[0][1]$. Cada uno es grabado en la caché de acuerdo a los 3 últimos bits de la dirección física a la cual corresponden, y sustituyen al valor anterior que se encuentre en la casilla que vaya a ocupar.

Contenido del caché tras el paso:									
Posición del Bloque (línea)	j=1	j=3	j=5	j=7	j=9	i=6	i=4	i=2	i=0
0	suma	suma	suma	suma	suma	A[0][6]	A[0][4]	A[0][2]	A[0][0]
1					ave	ave	Ave	ave	ave
2						i	i	i	i
3	j	j	j	j	j	j	J	j	j
4	A[0][1]	A[0][3]	A[0][5]	A[0][7]	A[0][9]	A[0][7]	A[0][5]	A[0][3]	A[0][1]
5									
6									
7									

Figura 1. Correspondencia Directa

Con la correspondencia completamente asociativa hay que ir pendiente de cual fue la posición menos recientemente usada (LRU) para hacer las sustituciones. En este caso, la caché se va llenando de forma ascendente (bloques del 0 al 7). Para el caso $j=7$, ya se han almacenado previamente en caché las variables $A[0][0]$ a $A[0][6]$. Nótese que $A[0][6]$ de la corrida anterior ($j=6$) ocupa la posición donde estaba $A[0][0]$ (LRU en esa iteración) y $A[0][7]$ ocupa la posición donde estaba $A[0][1]$ (LRU en la iteración 7). Ver figura 2.

Contenido del caché tras el paso:					
Posición del Bloque (línea)	j=7	j=8	j=9	i=1	i=0
0	suma	suma	suma	A[0][4]	A[0][4]
1	j	j	j	A[0][5]	A[0][5]
2	A[0][6]	A[0][6]	A[0][6]	A[0][6]	A[0][0]
3	A[0][7]	A[0][7]	A[0][7]	A[0][1]	A[0][1]
4	A[0][2]	A[0][8]	A[0][8]	A[0][2]	A[0][2]
5	A[0][3]	A[0][3]	A[0][9]	A[0][3]	A[0][3]
6	A[0][4]	A[0][4]	Ave	ave	ave
7	A[0][5]	A[0][5]	A[0][5]	i	i

Figura 2. Correspondencia Completamente Asociativa

La caché asociativa por conjuntos de 2 vías almacena en uno de los 2 conjuntos de acuerdo al último bit de la dirección física. Si es par almacena en el conjunto 0, si es impar, almacena en el conjunto 1. Las únicas variables con direcciones de memoria impar utilizadas son j (0x7A2B) y ave (0x7A29), por tanto están ubicadas en el conjunto 1. Para los casos de caché completamente asociativa y asociativa por conjunto, hay que notar que el orden de los for (ascendente y descendente) rige quien es el LRU en cada caso.

		Contenido del caché tras el paso:					
		j=3	j=7	j=9	i=4	i=2	i=0
Posición del Bloque (línea)							
Conjunto 1	0	suma	suma	suma	A[0][4]	A[0][4]	A[0][1]
	1	A[0][3]	A[0][6]	A[0][9]	A[0][6]	A[0][3]	A[0][0]
	2	A[0][1]	A[0][7]	A[0][7]	i	i	i
	3	A[0][2]	A[0][5]	A[0][8]	A[0][5]	A[0][2]	A[0][2]
Conjunto 2	4	j	j	j	j	j	J
	5			ave	ave	ave	ave
	6						
	7						

Correspondencia Asociativa por conjunto de 2 vías