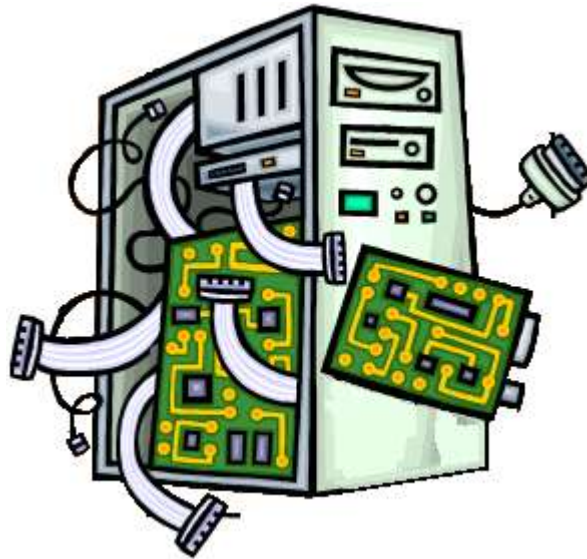
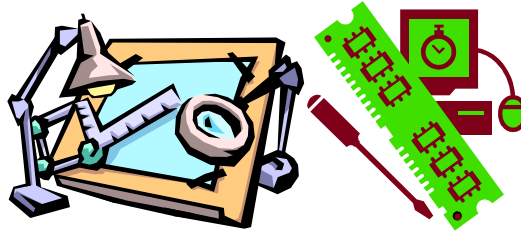


Ejercicios
de
Arquitectura
de
Computadoras



José Garzía

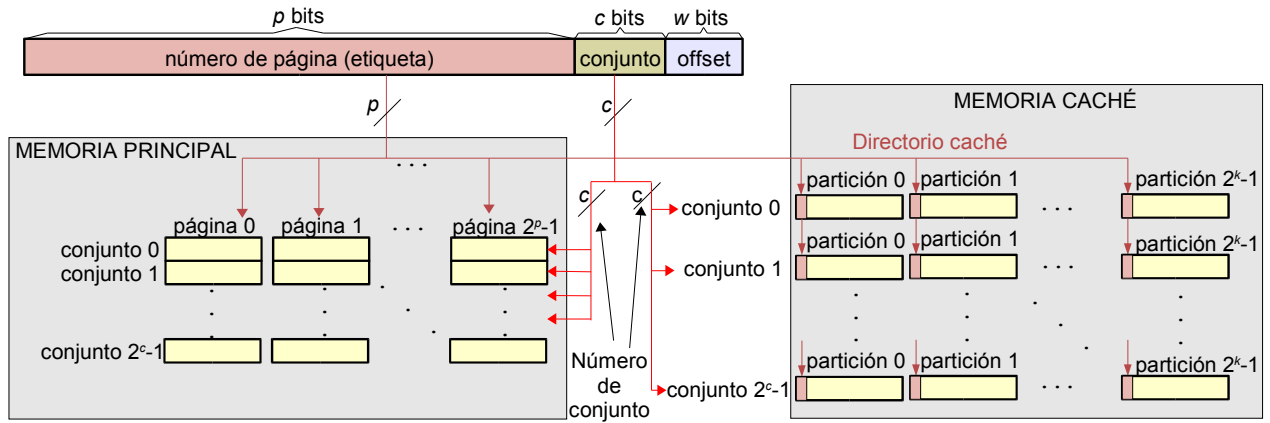
2009

En este tipo de ejercicios debemos tener siempre presentes estas tres ecuaciones:

$$|MP| \equiv (\text{Cantidad de palabras en Memoria Principal}) = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$|\text{caché}| \equiv (\text{Cantidad de palabras en la caché}) = (\text{Cantidad de conjuntos en la caché}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en Memoria Principal}}{\text{Cantidad de conjuntos en la caché}}$$



Una memoria caché asociativa por conjuntos dispone de 16 conjuntos y utiliza particiones de 8 palabras, siendo su capacidad total de 2.048 palabras. La memoria principal tiene una capacidad de 1.024·K palabras. ¿Cuántos bits hay en los diferentes campos de formato de dirección?

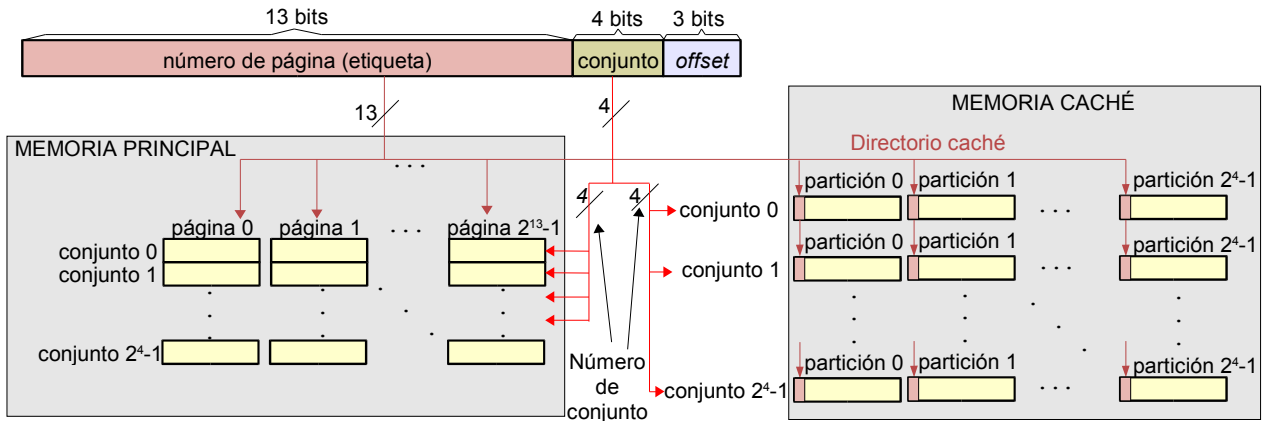
$$|\text{caché}| = (\text{Cantidad de palabras en la caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{11} \text{ palabras} = (2^4 \text{ conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{2^3 \text{ palabras}}{\text{por partición}} \Rightarrow \frac{\text{Cantidad de particiones}}{\text{por conjunto}} = \frac{2^4 \text{ particiones}}{\text{por conjunto}}$$

$$|\text{MP}| = (\text{Cantidad de palabras en la Memoria Principal}) = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{20} = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{2^3 \text{ palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de particiones en Memoria Principal}) = 2^{17} \text{ particiones}$$

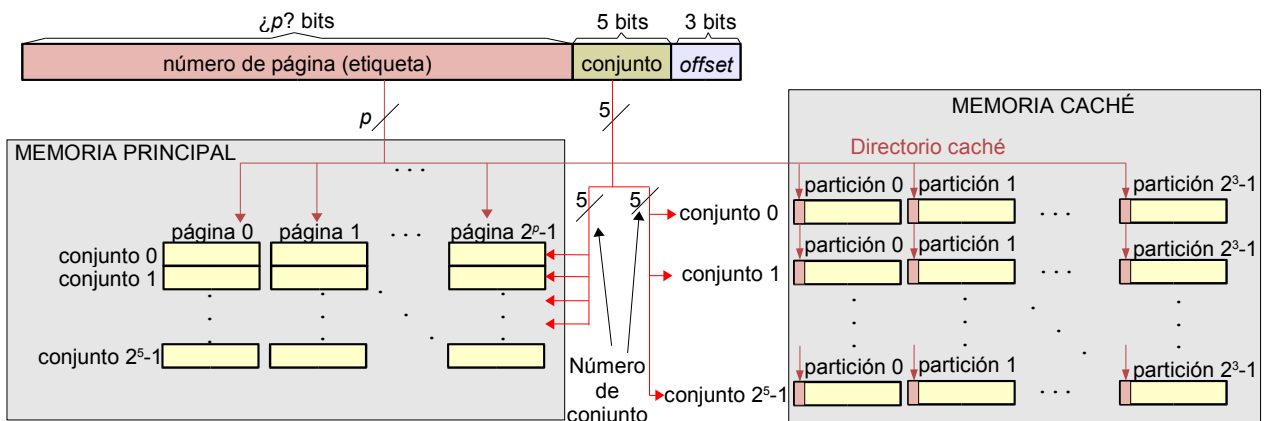
$$(\text{Cantidad de páginas en Memoria Principal}) = \frac{\text{Cantidad de particiones en Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^{17} \text{ particiones}}{2^4 \text{ conjuntos}} = 2^{13} \text{ páginas}$$



Una memoria caché asociativa por conjuntos dispone de 32 conjuntos, utiliza particiones de 8 palabras y su capacidad total es de 2.048 palabras. ¿A cuántas posiciones distintas de la caché (suponiéndola vacía) puede ir cada palabra de MP?

$$|\text{caché}| = (\text{Cantidad de palabras en caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{11} = (2^5 \text{ conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{2^3 \text{ palabras}}{\text{por partición}} \Rightarrow \frac{\text{Cantidad de particiones}}{\text{por conjunto}} = \frac{2^{11}}{2^5 \cdot 2^3} = 2^3 \text{ particiones}$$



No podemos calcular la longitud de todos los campos, pues no nos dieron el tamaño de la Memoria Principal. De todas formas, cada palabra de Memoria Principal puede ir a una cualquiera de las $2^3 = 8$ particiones de cada conjunto.

Una computadora tiene una unidad de memoria con 16 líneas de dirección, y una memoria caché de 1K palabras. La memoria caché utiliza correspondencia asociativa por conjuntos, con un tamaño de partición de 16 palabras y 4 particiones por conjunto. Explique razonadamente en qué campos se divide la dirección de memoria principal 0000110111000011 e indique el valor de cada uno de estos campos en decimal.

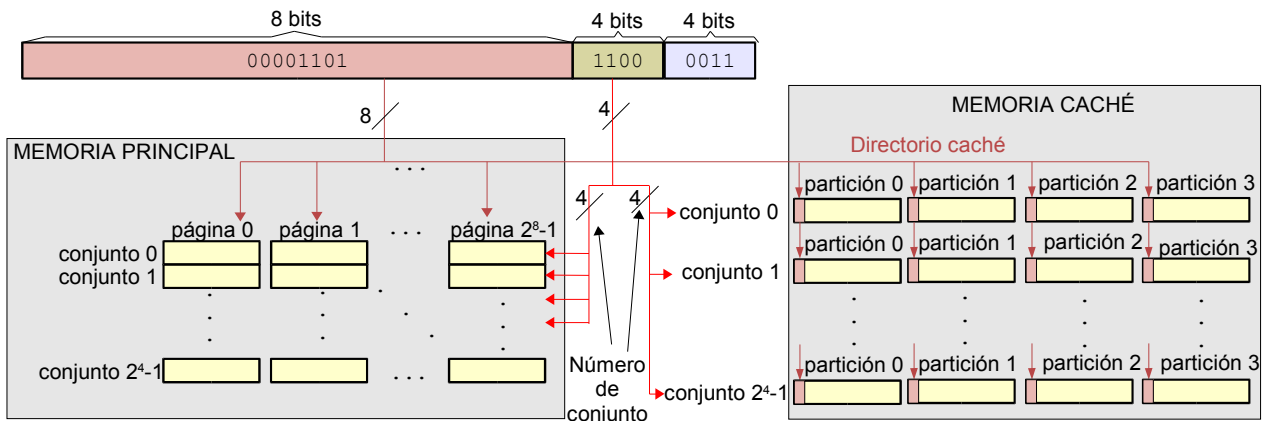
$$|\text{caché}| \equiv (\text{Cantidad de palabras en caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{10} = (\text{Cantidad de conjuntos}) \cdot \frac{2^2 \text{ particiones}}{\text{por conjunto}} \cdot \frac{2^4 \text{ palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de conjuntos}) = \frac{2^{10}}{2^2 \cdot 2^4} = 2^4 \text{ conjuntos}$$

$$|\text{MP}| \equiv (\text{Cantidad de palabras en la Memoria Principal}) = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{16} = (\text{Cantidad de particiones en Memoria Principal}) \cdot \frac{2^4 \text{ palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de particiones en Memoria Principal}) = \frac{2^{16}}{2^4} = 2^{12} \text{ particiones}$$

$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en la Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^{12} \text{ particiones}}{2^4 \text{ conjuntos}} = 2^8 \text{ páginas}$$



Una memoria caché asociativa por conjuntos consta de 16 conjuntos con 4 particiones por conjunto. La memoria principal tiene una capacidad de 1 Mb (2^{20} palabras) dividida en particiones de 2^7 palabras. La palabra almacenada en la dirección de memoria principal, expresada en binario es: 1010 0001 1001 0011 0000.

- a) ¿A qué conjunto debe ir?
- b) ¿Qué etiqueta deberá tener la partición donde vaya?

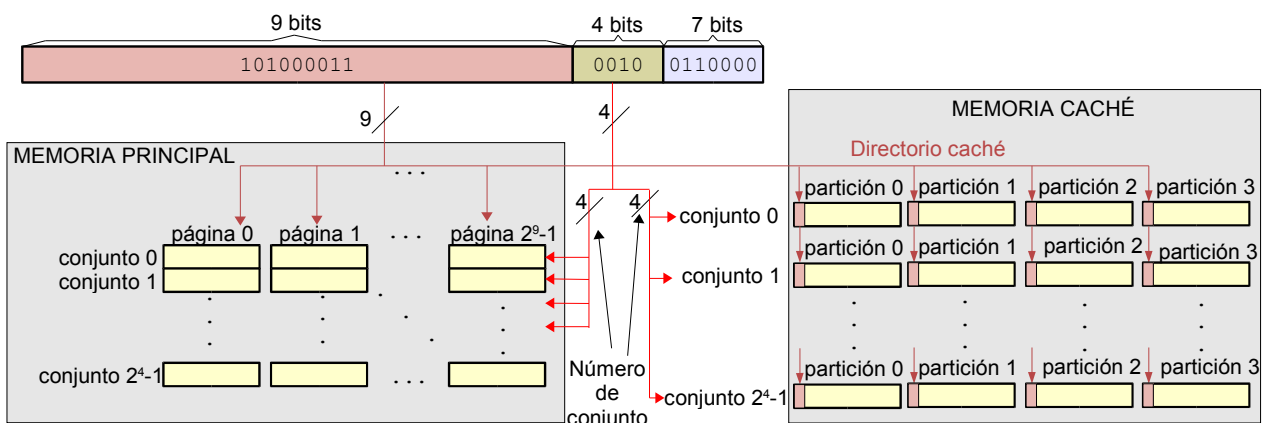
$$|\text{caché}| \equiv (\text{Cantidad de palabras en caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$|\text{caché}| = (2^4 \text{ conjuntos}) \cdot \frac{2^2 \text{ particiones}}{\text{por conjunto}} \cdot \frac{2^7 \text{ palabras}}{\text{por partición}} \Rightarrow |\text{caché}| = 2^{13} \text{ palabras, pero este resultado no se necesitaba en este ejercicio.}$$

$$|\text{MP}| \equiv (\text{Cantidad de palabras en la Memoria Principal}) = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{20} = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{2^7 \text{ palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de particiones en la Memoria Principal}) = 2^{13} \text{ particiones}$$

$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en la Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^{13} \text{ particiones}}{2^4 \text{ conjuntos}} = 2^9 \text{ páginas}$$



Un computador tiene una unidad de memoria de 2^{13} palabras y una memoria caché asociativa por conjuntos de 2^7 palabras, suponiendo que la UCP genera la dirección de memoria principal 00101 1011 0111, indique cuál es el conjunto caché donde irá y qué etiqueta tendrá su partición.

- a) Si la memoria caché tiene 4 conjuntos y 8 particiones por conjunto.
- b) Si la memoria caché tiene 8 conjuntos y 4 particiones por conjunto.

a) Si la memoria caché tiene 4 conjuntos y 8 particiones por conjunto.

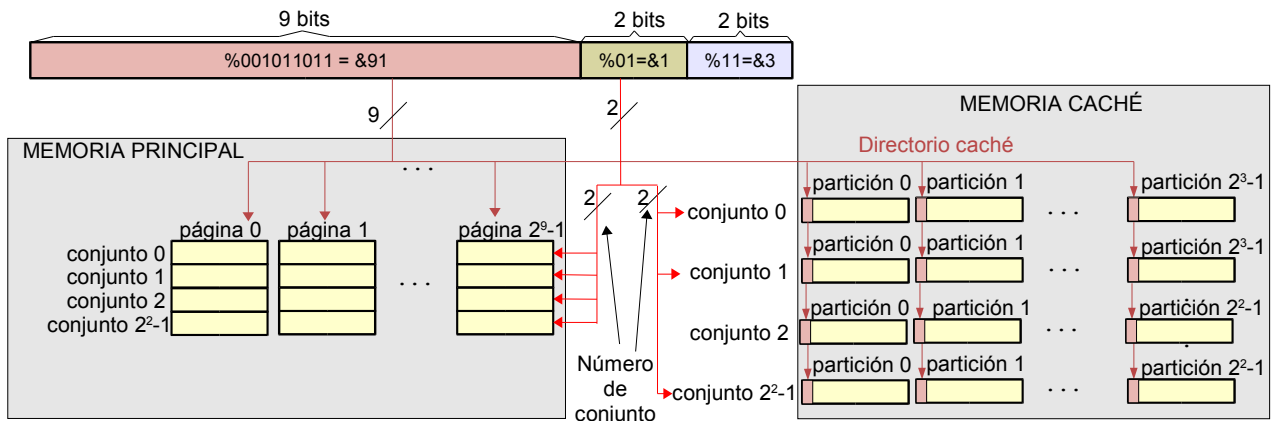
$$|\text{caché}| \equiv (\text{Cantidad de palabras en la caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^7 = \left(2^2 \text{ conjuntos} \right) \cdot \frac{2^3 \text{ particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}} \Rightarrow \frac{\text{Cantidad de palabras}}{\text{por partición}} = \frac{2^7}{2^2 \cdot 2^3} = 2^2 \text{ palabras}$$

$$|\text{MP}| \equiv (\text{Cantidad de palabras en la Memoria Principal}) = (\text{Cantidad de particiones en Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{13} = (\text{Cantidad de particiones en Memoria Principal}) \cdot \frac{2^2 \text{ palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de particiones en Memoria Principal}) = \frac{2^{13}}{2^2} = 2^{11} \text{ particiones}$$

$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en la Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^{11} \text{ particiones}}{2^2 \text{ conjuntos}} = 2^9 \text{ páginas}$$



b) Si la memoria caché tiene 8 conjuntos y 4 particiones por conjunto.

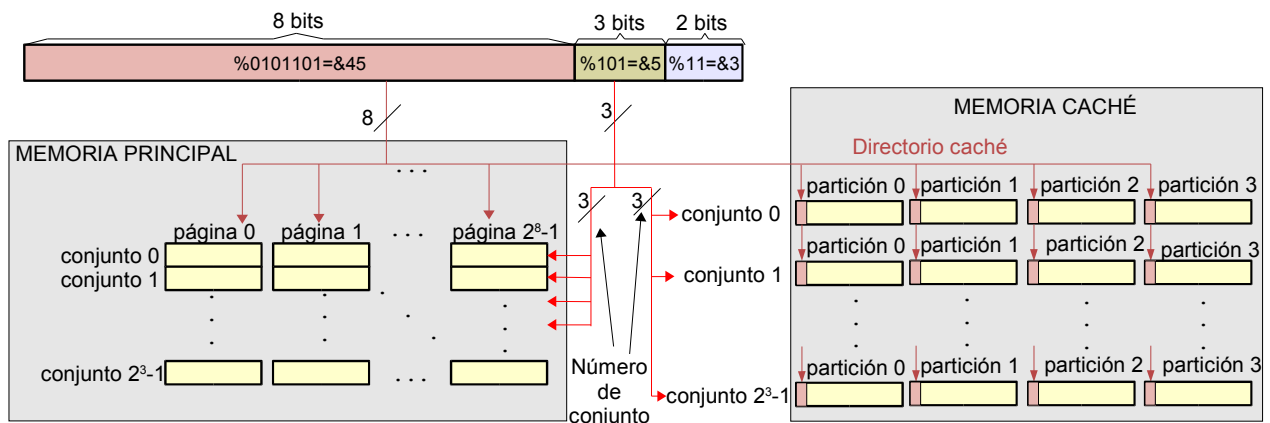
$$|\text{caché}| \equiv (\text{Cantidad de palabras en la caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^7 = \left(2^3 \text{ conjuntos} \right) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}} \Rightarrow \frac{\text{Cantidad de palabras}}{\text{por partición}} = \frac{2^7}{2^3 \cdot 2^2} = 2^2 \text{ palabras}$$

$$|\text{MP}| \equiv (\text{Cantidad de palabras en la Memoria Principal}) = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{13} (\text{Cantidad de particiones en Memoria Principal}) \cdot \frac{2^2 \text{ de palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de particiones en Memoria Principal}) = \frac{2^{13}}{2^2} = 2^{11} \text{ particiones}$$

$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en la Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^{11} \text{ particiones}}{2^3 \text{ conjuntos}} = 2^8 \text{ páginas}$$



✎ Un computador tiene una unidad de memoria de 2^{12} palabras y una memoria caché de 2^8 palabras. La memoria caché es asociativa por conjuntos, con 16 conjuntos y 2 particiones por conjunto. Cuando se necesita un reemplazo en la caché, en el conjunto correspondiente se reemplaza la partición más antigua.

Suponiendo que inicialmente la memoria caché está vacía. ¿Cuántos fallos se producirían en la caché si se leyeran consecutivamente las siguientes direcciones de la MP?

```

0000 0000 1111
0000 0100 1111
0001 0000 1111
0001 0000 1100
0010 1000 1111
0000 0000 1101
    
```

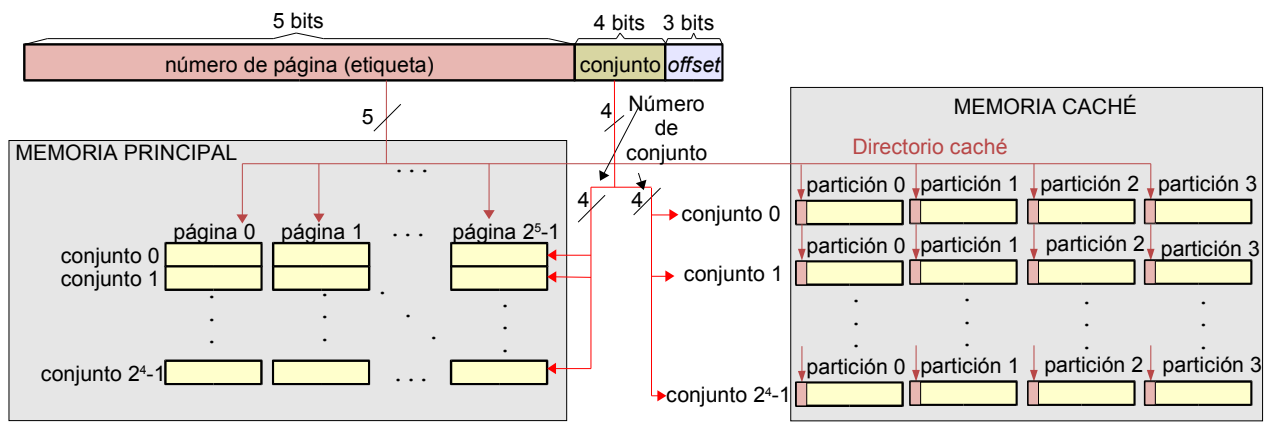
$$|caché| = (\text{Cantidad de palabras en la caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^8 = (2^4 \text{ conjuntos}) \cdot \frac{2 \text{ particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}} \Rightarrow \frac{\text{Cantidad de palabras}}{\text{por partición}} = 2^3 \text{ palabras}$$

$$|MP| = (\text{Cantidad de palabras en la Memoria Principal}) = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{12} = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{2^3 \text{ de palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de particiones en la Memoria Principal}) = 2^9 \text{ particiones}$$

$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en la Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^9 \text{ particiones}}{2^4 \text{ conjuntos}} = 2^5 \text{ páginas}$$



- 00000 0001 111 → Va al conjunto 1. **Provoca fallo**, pues la caché está inicialmente vacía.
- 00000 1001 111 → Va al conjunto 9. **Provoca fallo**.
- 00010 0001 111 → Va al conjunto 1. **Provoca fallo**, la etiqueta que había en el conjunto 1 era 00000. Ésta tiene como etiqueta 00010. Puede ir a la partición 1 de este mismo conjunto, por lo que **no provoca reemplazo**.
- 00010 0001 100 → Va al conjunto 1. **No provoca fallo**, pues la etiqueta 00010 ya estaba en la partición 1 del conjunto 1.
- 00101 0001 111 → Va al conjunto 1. **Provoca fallo**, pues la etiqueta 00101 no estaba presente. También **provoca reemplazo**, pues el conjunto 1 ya estaba lleno. Debe ser reemplazada la que estaba en la partición 0 del conjunto 1.
- 00000 0001 101 → Va al conjunto 1. **Provoca fallo**, pues la etiqueta 00000 acaba de ser reemplazada. También **provoca reemplazo**. Debe ser reemplazada la partición 1 del conjunto 1.

✍ Sea una memoria caché de 128 bytes con estrategia de ubicación asociativa por conjuntos, de forma que hay 4 conjuntos y cada uno puede almacenar 4 particiones. Las direcciones físicas están compuestas por 32 bits y la unidad más pequeña direccionable es el byte.

- Trace un esquema de la sección de control y la zona de almacenamiento de la caché, indicando cómo se relacionan las direcciones físicas con las direcciones caché.
- ¿A qué particiones de la caché puede asignarse la dirección física 000010AF (hexadecimal)?
- Si las direcciones físicas (hexadecimales) 00001AF y FFFF7Axy pueden ser asignadas simultáneamente al mismo conjunto caché, ¿qué valores pueden tener x e y?

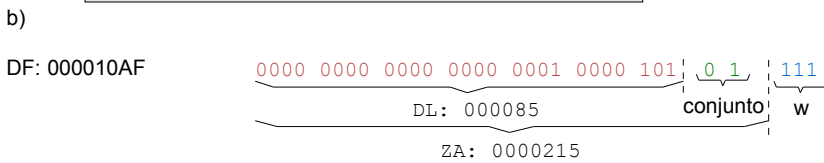
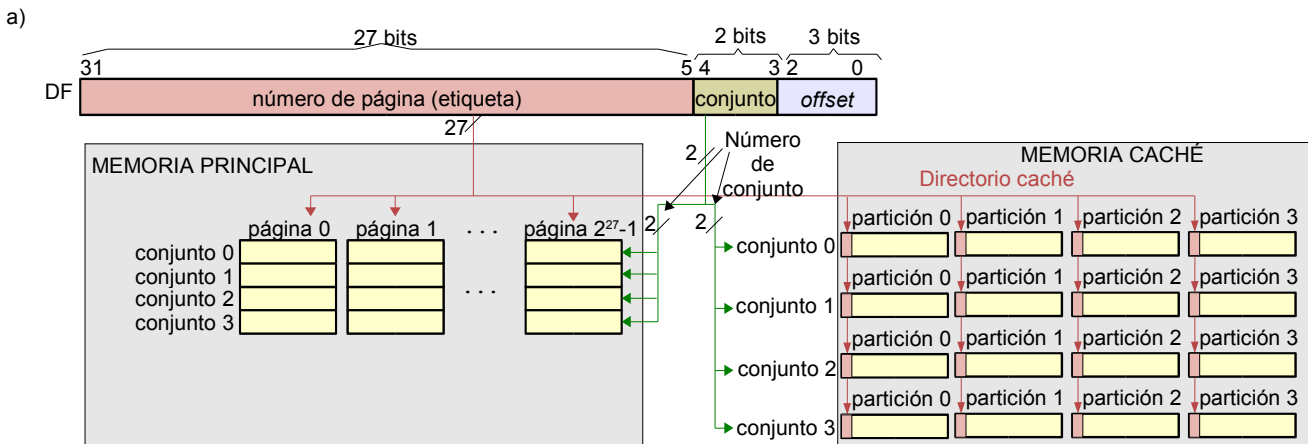
$$|\text{caché}| = (\text{Cantidad de palabras en la caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^7 \text{ palabras} = (2^2 \text{ conjuntos}) \cdot \frac{2^2 \text{ particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}} \Rightarrow \frac{\text{Cantidad de palabras}}{\text{por partición}} = 2^3 \text{ palabras}$$

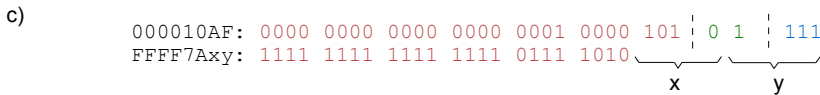
$$|\text{MP}| = (\text{Cantidad de palabras en la Memoria Principal}) = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{32} = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{2^3 \text{ de palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de particiones en la Memoria Principal}) = 2^{29} \text{ particiones}$$

$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en la Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^{29} \text{ particiones}}{2^2 \text{ conjuntos}} = 2^{27} \text{ páginas}$$



A las cuatro líneas del conjunto 01



Pueden ser asignadas al mismo conjunto: 01

x → Todas las combinaciones de x con el bit menos significativo a 0.

y → Todas las combinaciones con el bit más significativo a 1.

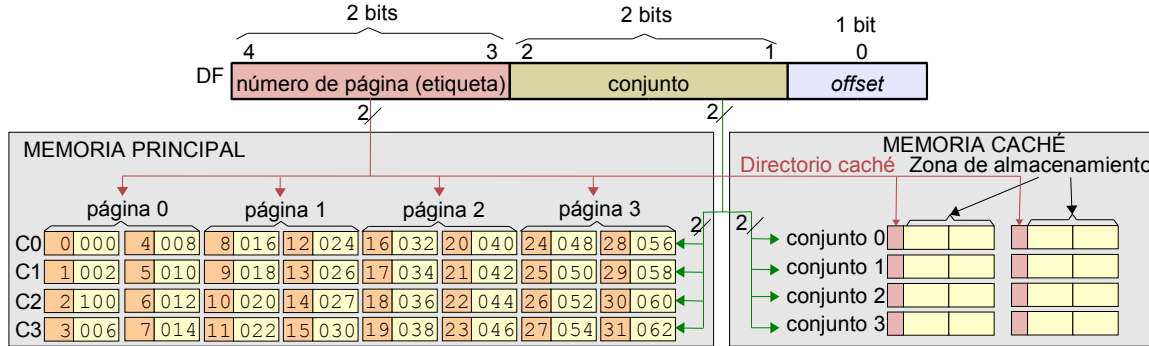
x	y		
0000	0	1000	8
0010	2	1001	9
0100	4	1010	A
0110	6	1011	B
1000	8	1100	C
1010	A	1101	D
1100	C	1110	E
1100	E	1111	F

✍ Sea un computador con una memoria caché de una capacidad de 8 particiones de 2 bytes cada una, incluida en un sistema sin memoria virtual. La caché está organizada asociativamente en 4 conjuntos. La operación de búsqueda en la caché es por demanda. El reemplazo es FIFO (el primero en entrar es el primero en salir). La actualización de la memoria principal es mediante post-escritura bajo fallo. La memoria principal está organizada en bytes y tiene una capacidad de 32 bytes, de forma que el contenido de la posición δ es $2 \cdot \delta$. Si definimos $W(x, \delta)$ como la escritura del contenido de la variable x sobre la posición de memoria de dirección δ ; y la función $R(x, \delta)$ como la lectura sobre la variable x del contenido de la posición de memoria de dirección δ , analice cómo evoluciona el contenido de la zona de almacenamiento y del directorio de la caché y el contenido de la memoria principal cuando el procesador emite la siguiente secuencia de operaciones:

- R(x_0 , 17)
- R(x_1 , 6)
- W(100, 8)
- W(101, 16)
- R(x_2 , 9)
- R(x_3 , 25)

Analice también el contenido de las variables x_0, x_1, x_2 y x_3 .

Organización y contenidos (en base decimal) iniciales:



Instrucción 1ª: R(x_0 , 17)

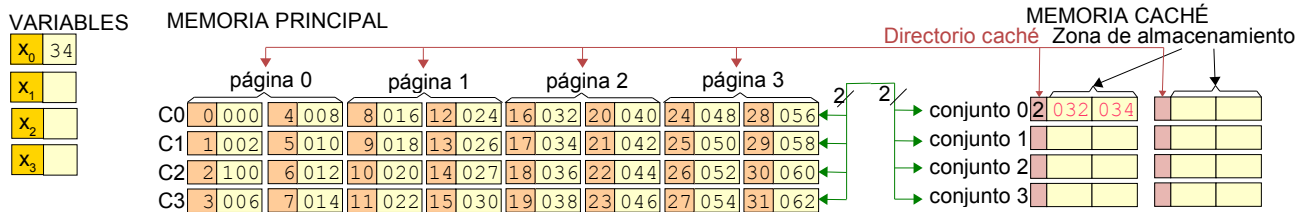
Se debe acceder a la posición 10001 (17) de la MP. Por supuesto, ocurre un fallo de página caché, pues inicialmente está vacía.

10001 determina:

- ✓ Conjunto 0.
- ✓ 2ª palabra dentro de la línea.
- ✓ Almacenar $10_b = 2_d$ en el directorio.

Como es almacenada una línea entera, también se almacena el contenido de la dirección anterior (&16)=832.

Estado actual:

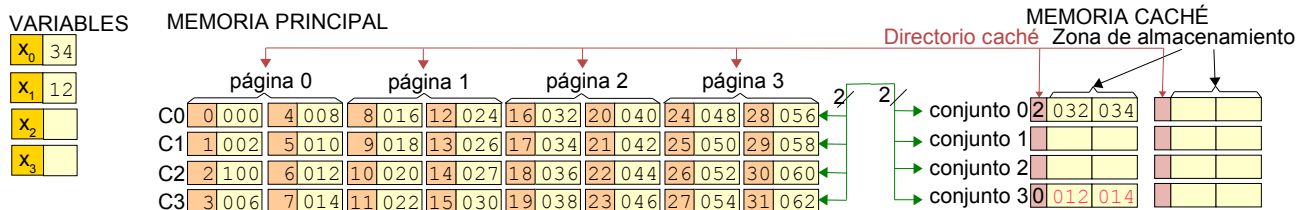


Instrucción 2ª: R(x_1 , 6)

00110 determina:

- ✓ Conjunto 3.
- ✓ Almacenar $0_b = 0_d$ en el directorio.

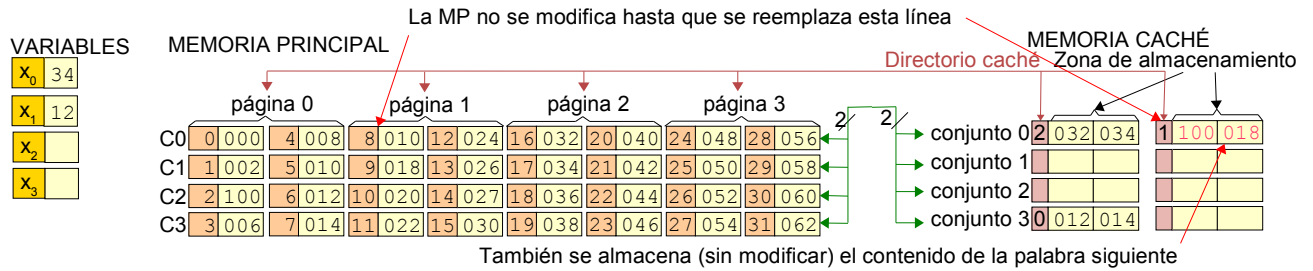
Estado actual:



Instrucción 3ª: w(100, 8)

Se debe acceder a la posición 01000 de la MP.
Se anota 100 en su correspondiente palabra caché.

Estado actual:

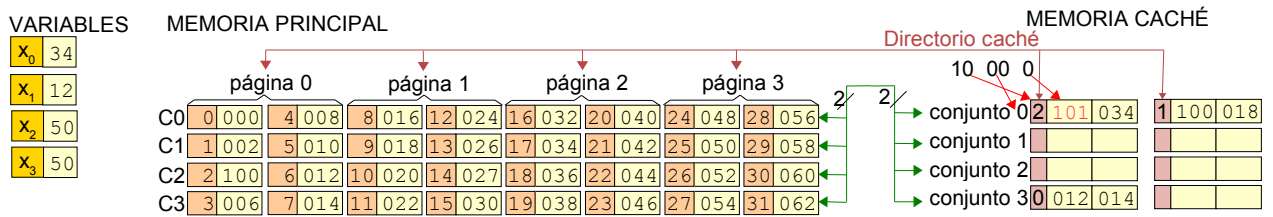


Instrucción 4ª: w(101, 16)

Se debe acceder a la posición 10000, la cual ya está en la caché.

Estado actual:

Hasta ahora no ha habido ningún reemplazo, por lo cual, la MP aún no se ha modificado:



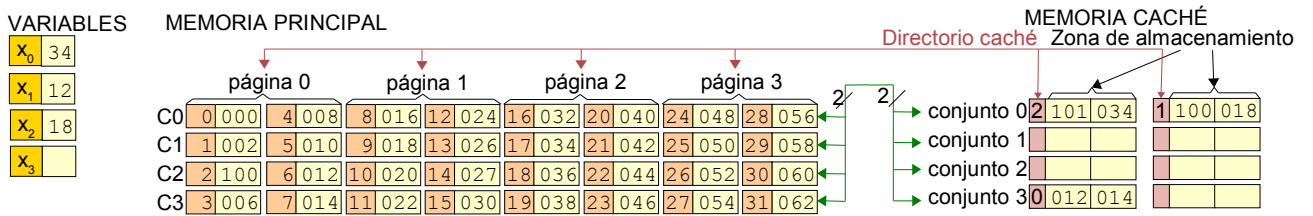
Instrucción 5ª: R(x₂, 9)

Se debe acceder a la posición 10000.

10000 determina:

- ✓ Conjunto 0. No hay fallo de línea, pues ya estaba allí.
- ✓ Almacenar 0_b=0_d en el directorio.

Estado actual:



Instrucción 6ª: R(x₃, 25)

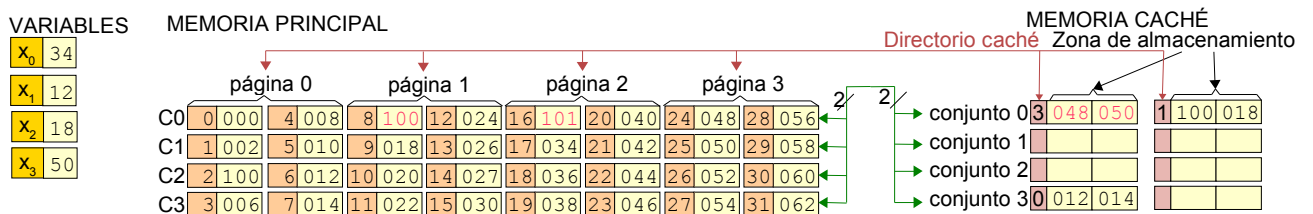
Se debe acceder a la posición 11001.

11001 determina:

- ✓ Nueva Dir=3. No coincide con ninguna de las que había.
- ✓ Conjunto 0. Como este conjunto estaba lleno, se debe reemplazar la línea más antiguamente referenciada.

Con el reemplazo, es el momento de actualizar la memoria principal.

Estado actual:



Los contenidos de la MP son los iniciales, excepto en las posiciones 8 y 16; cuyos contenidos son 100 y 101, respectivamente.

✏ Sea un sistema caché capaz de almacenar 4 particiones, cada una de ellas de 8 bytes. La memoria principal es de 64 Kbytes. La caché inicialmente está vacía. Si es necesario reemplazar alguna partición, ésta se elige mediante el algoritmo LRU. El procesador emite la siguiente secuencia de direcciones físicas (en notación hexadecimal):

32B5, 32B8, 4A91, 4220, A727 y C0AA

Analice cómo evoluciona el contenido de la zona de almacenamiento y del directorio de la caché, haciendo un recuento de los fallos de línea con cada algoritmo de ubicación:

- a) Totalmente asociativa.
- b) Directa

$$|MP| = (\text{Cantidad de palabras en la Memoria Principal}) = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{16} = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{2^3 \text{ de palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de particiones en la Memoria Principal}) = 2^{13} \text{ particiones}$$

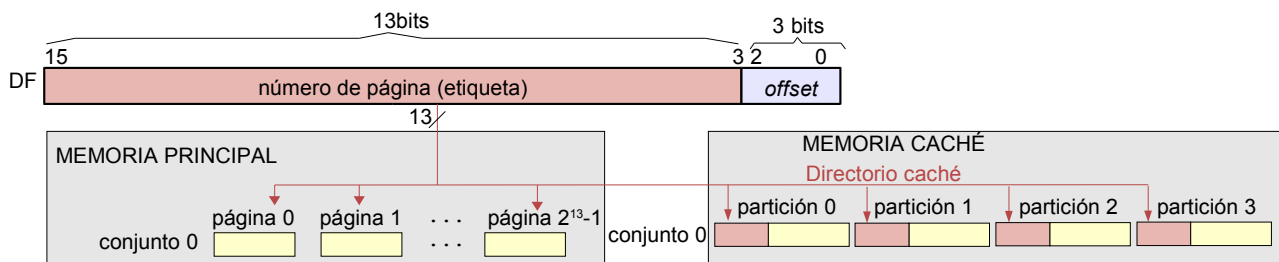
$$|MP| = 64 \text{ Kbytes} = 2^{16} \text{ bytes} \Rightarrow ||DF|| = 16 \text{ bits}$$

$$|\text{caché}| = (\text{Cantidad de palabras en la caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

a) Con algoritmo de ubicación totalmente asociativa.

$$|\text{caché}| = (2^0 \text{ conjuntos}) \cdot \frac{2^2 \text{ particiones}}{\text{por conjunto}} \cdot \frac{2^3 \text{ palabras}}{\text{por partición}} = 2^5 \text{ palabras}$$

$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en la Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^{13} \text{ particiones}}{2^0 \text{ conjuntos}} = 2^{13} \text{ páginas}$$

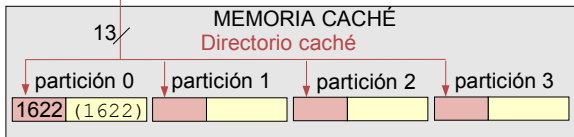


Evolución de la caché:

Instrucción 1ª: 32B5

Fallo de caché (inicialmente está vacía).

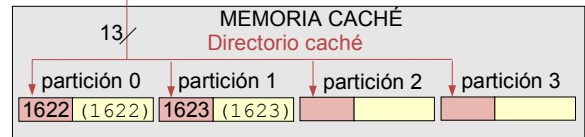
0011 0010 1011 0 | 101
 etiqueta | offset
 (directorio caché)
 DC = 1622



Instrucción 2ª: 32B8

Fallo de caché.

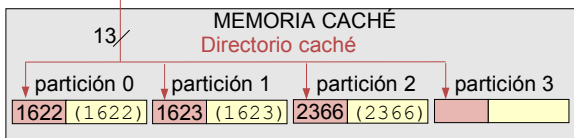
0011 0010 1011 1 | 000
 etiqueta | offset
 (directorio caché)
 DC = 1623



Instrucción 3ª: 4A91

Fallo de caché.

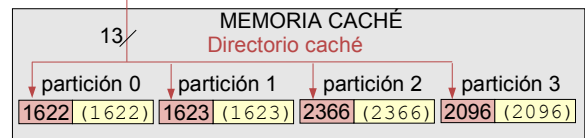
0100 1010 1001 0 | 001
 etiqueta | offset
 (directorio caché)
 DC = 2366



Instrucción 4ª: 4220

Fallo de caché.

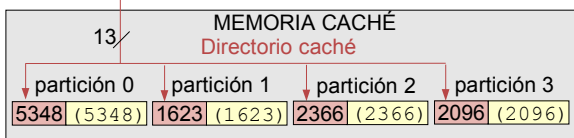
0100 0010 0010 0 | 000
 etiqueta | offset
 (directorio caché)
 DC = 2096



Instrucción 5ª: A727

Fallo de caché y reemplazo.

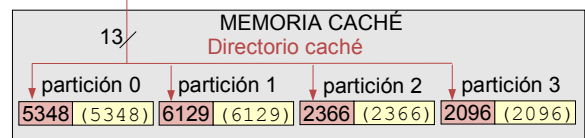
1010 0111 0010 0 | 111
 etiqueta | offset
 (directorio caché)
 DC = 5348



Instrucción 6ª: C0AA

Fallo de caché y reemplazo.

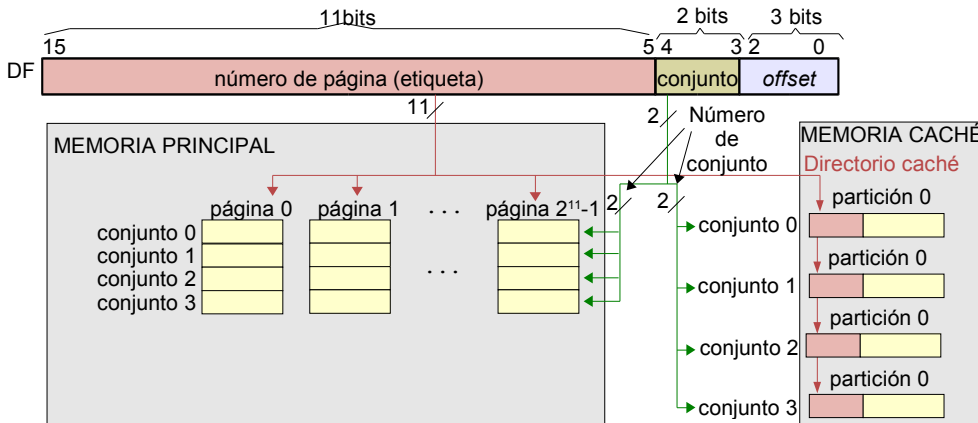
1100 0000 1010 1 | 010
 etiqueta | offset
 (directorio caché)
 DC = 6129



b) Con algoritmo de ubicación directa.

$$|\text{caché}| = \left(2^2 \text{ conjuntos} \right) \cdot \frac{2^0 \text{ particiones}}{\text{por conjunto}} \cdot \frac{2^3 \text{ palabras}}{\text{por partición}} = 2^5 \text{ palabras}$$

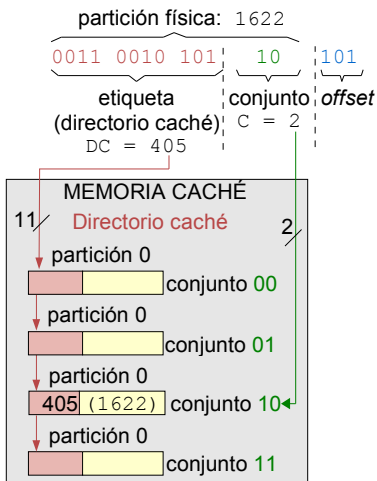
$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en la Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^{13} \text{ particiones}}{2^2 \text{ conjuntos}} = 2^{11} \text{ páginas}$$



Evolución de la caché:

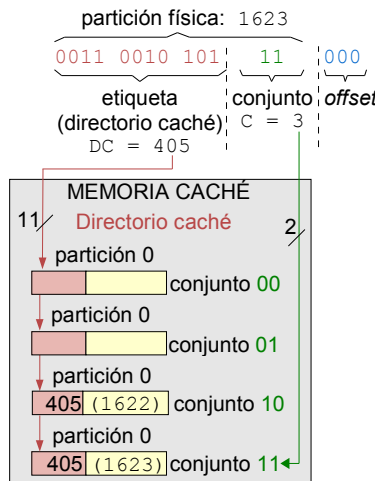
Instrucción 1ª: **32B5**

Fallo de caché (inicialmente está vacía).



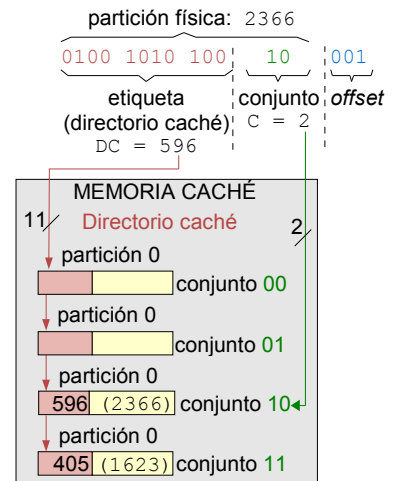
Instrucción 2ª: **32B8**

Fallo de caché.



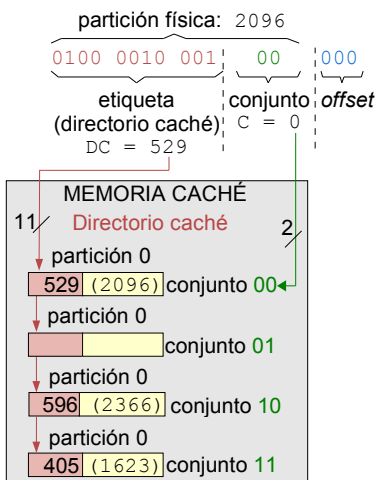
Instrucción 3ª: **4A91**

Fallo de caché y reemplazo.



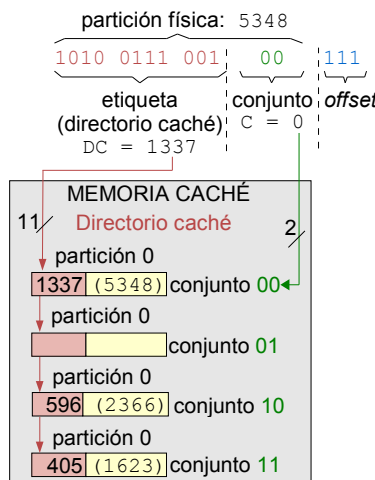
Instrucción 4ª: **4220**

Fallo de caché.



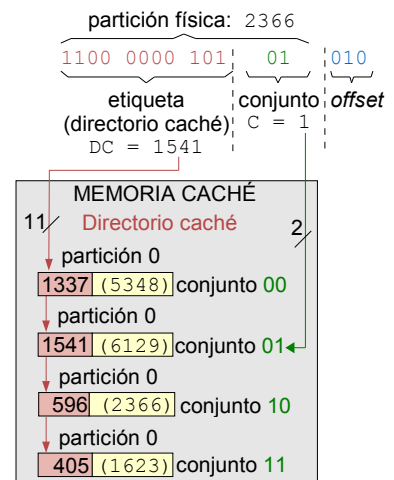
Instrucción 5ª: **A727**

Fallo de caché y reemplazo.

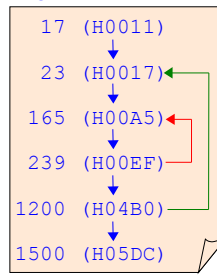


Instrucción 6ª: **COAA**

Fallo de caché.



Sea un computador cuya memoria principal es de 64K palabras y su memoria caché es de de 1K palabras con correspondencia directa y tamaño de bloque de 2^7 palabras. El tiempo de acceso a la memoria principal es de $10 \cdot \tau$ y el tiempo de acceso a la memoria caché es de $1 \cdot \tau$. En dicho computador se ejecuta un programa cuya estructura general se muestra en la siguiente figura:



Tiene dos bucles anidados, un bucle interior menor que se ejecuta 20 veces y un bucle exterior que se ejecuta 10 veces. Las direcciones mostradas se corresponden con el principio y final de cada uno de estos bucles, y el principio y final del programa. Las direcciones se muestran en formato *decimal (hexadecimal)*, por ejemplo $165_d (00A5_h)$.

Todas las direcciones de memoria de las diferentes secciones, 17-22, 23-164, 165-238, etc., contienen instrucciones que deben ejecutarse en secuencia de línea recta.

- Especifique el número de bits de los campos en que se descompone una dirección de memoria principal de este sistema.
- Calcule cuánto tiempo se tardaría en ejecutar este programa si este computador no tuviera memoria caché.
- Analice cuántos fallos se producen en la caché durante la ejecución de este programa. (Sugerencia: diferencie entre la primera ejecución del bucle exterior y las nueve restantes.)
- Si cada vez que se produce un fallo primero se mueve el bloque completo de memoria principal a memoria caché y después se lee el dato de la caché, calcule cuánto tiempo se tardaría en ejecutar este programa en este computador.

a)

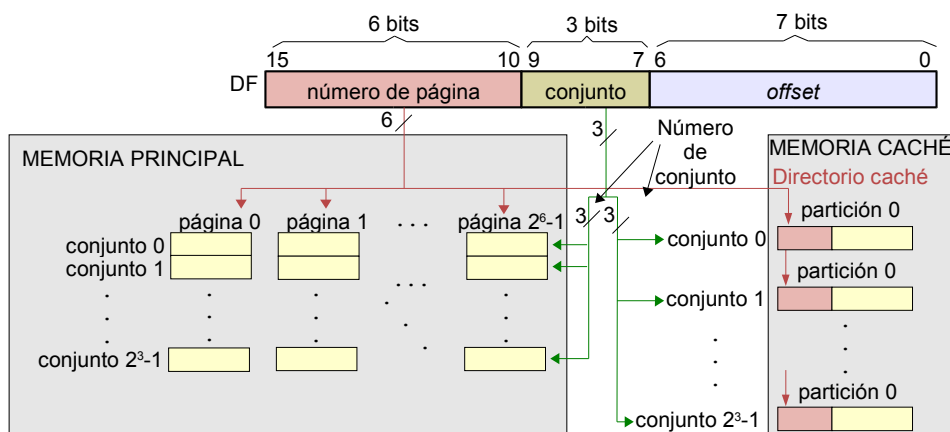
$$|\text{caché}| \equiv (\text{Cantidad de palabras en la caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{10} = (\text{Cantidad de conjuntos}) \cdot \frac{1 \text{ partición}}{\text{por conjunto (por ser correspondencia directa)}} \cdot \frac{2^7 \text{ palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de conjuntos}) = 2^3 \text{ conjuntos}$$

$$|\text{MP}| \equiv (\text{Cantidad de palabras en la Memoria Principal}) = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{16} = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{2^7 \text{ palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de particiones en Memoria Principal}) = 2^9 \text{ particiones}$$

$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en la Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^9 \text{ particiones}}{2^3 \text{ conjuntos}} = 2^6 \text{ páginas}$$



b)

Nº de lecturas fuera del bucle externo: $(23-17) + (1501-1201) = 306$
 Nº de lecturas en el bucle interno: $((240-165) : 20) = 1500$
 Nº de lecturas en el bucle externo: $((165-23) + (\text{Cantidad de lecturas en bucle interno}) + (1201-240)) \cdot 10 = (142 + 1500 + 961) \cdot 10 = 26030$
 Nº total de lecturas: $306 + 26030 = 26336$
 Tiempo total debido a las lecturas sin memoria caché $26336 \cdot 10\tau = 263360 \cdot \tau$

c)

✓ Instrucciones previas a la entrada en el bucle externo

0011 _h	0000	00	00	0	001	0001.	Fallo, pues en el conjunto 0 de la caché no había nada.
....	
0016 _h	0000	00	00	0	001	0110.	

Total de fallos en este segmento del programa: 1

✓ Primera ejecución del bucle externo.

0017 _h	0000	00	00	0	001	0111.	Entrada en el bucle externo.
....	
007F _h	0000	00	00	0	111	1111.	
0080 _h	0000	00	00	1	000	0000.	Fallo, pues en el conjunto 1 de la caché no había nada.
0081 _h	0000	00	00	1	000	0001.	
....	
00A5 _h	0000	00	00	1	010	0101.	Entrada en el bucle interno.
....	} Bucle interno
00EF _h	0000	00	00	1	110	1111.	

Ahora (si este bucle interno aún no se ha repetido 20 veces) se vuelve a la dirección 00A5_h. Fijémonos que todas las direcciones del bucle interno están en el conjunto 1 de la página 0. Por este motivo, durante la ejecución del bucle interno no se producen fallos ni se provocan reemplazos.

Continuamos tras salir del bucle interno:

00F0 _h	0000	00	00	1	111	0000.	
....	
00FF _h	0000	00	00	1	111	1111.	
0100 _h	0000	00	01	0	000	0000.	Fallo, pues en el conjunto 2 de la caché no había nada.
0101 _h	0000	00	01	0	000	0001.	
....	
017F _h	0000	00	01	0	111	1111.	
0180 _h	0000	00	01	1	000	0000.	Fallo, pues en el conjunto 3 de la caché no había nada.
0181 _h	0000	00	01	1	000	0001.	
....	
01FF _h	0000	00	01	1	111	1111.	
0200 _h	0000	00	10	0	000	0000.	Fallo, pues en el conjunto 4 de la caché no había nada.
0201 _h	0000	00	10	0	000	0001.	
....	
027F _h	0000	00	10	0	111	1111.	
0280 _h	0000	00	10	1	000	0000.	Fallo, pues en el conjunto 5 de la caché no había nada.
0281 _h	0000	00	10	1	000	0001.	
....	
02FF _h	0000	00	10	1	111	1111.	
0300 _h	0000	00	11	0	000	0000.	Fallo, pues en el conjunto 6 de la caché no había nada.
0301 _h	0000	00	11	0	000	0001.	
....	
037F _h	0000	00	11	0	111	1111.	
0380 _h	0000	00	11	1	000	0000.	Fallo, pues en el conjunto 7 de la caché no había nada.
0381 _h	0000	00	11	1	000	0001.	
....	
03FF _h	0000	00	11	1	111	1111.	
0400 _h	0000	01	00	0	000	0000.	Fallo y reemplazo, pues el conjunto 0 de la caché procedía de la página 0.
0401 _h	0000	01	00	0	000	0001.	
....	
047F _h	0000	01	00	0	111	1111.	
0480 _h	0000	01	00	1	000	0000.	Fallo y reemplazo, pues el conjunto 1 de la caché procedía de la página 0.
0481 _h	0000	01	00	1	000	0001.	
....	
04B0 _h	0000	01	00	1	001	0000.	Fin del bucle externo.

Total de fallos en este segmento del programa: 9

✓ Sigüientes ejecuciones del bucle externo.

Se produce fallo y reemplazo al pedir los conjuntos 0 y 1 de la página 0, pues son reemplazados al final del bucle, al pedir sus conjuntos tocayos de la página 1.

No se produce fallo al pedir los conjuntos 2, 3, 4, 5, 6 ni 7 de la página 0, pues ya estaban en la caché desde la primera pasada por el bucle; y nunca son reemplazados.

Se produce fallo y reemplazo al pedir los conjuntos 0 y 1 de la página 1, pues son reemplazados al comienzo del bucle, al pedir sus conjuntos tocayos de la página 0.

Total de fallos en este segmento del programa: 4

✓ Instrucciones posteriores a la salida del bucle externo.

04B1 _h 0000	01	00	1	001	0001.
.....
04FF _h 0000	01	00	1	111	1111.
0500 _h 0000	01	01	0	000	0000. <i>Fallo y reemplazo</i> , pues el conjunto 2 de la caché procedía de la página 0.
0501 _h 0000	01	01	0	000	0001.
.....
057F _h 0000	01	01	0	111	1111.
0580 _h 0000	01	01	1	000	0000. <i>Fallo y reemplazo</i> , pues el conjunto 3 de la caché procedía de la página 0.
0581 _h 0000	01	01	1	000	0001.
.....
05DC _h 0000	01	01	1	101	1100. Fin del programa.

Total de fallos en este fragmento del programa: 2

Número total de fallos: $1 + 9 \cdot 1 + 4 \cdot 9 + 2 = 48$

d) Según nos especifican, cuando se produce un fallo, el contenido de la dirección fallida no se lleva desde la memoria principal simultáneamente a la caché y a la CPU. Tal como está diseñado este sistema de caché, cuando se produce el fallo, primero se lleva el contenido de las 128 direcciones cuya partición contiene a la dirección pedida desde la memoria principal a la caché y después se lee en la caché la dirección de marras. Por este motivo, siempre se lee en la caché, tanto si no hay fallo, como si lo hay. La conclusión es que:

Coste total = (Número total de direcciones buscadas) · (Tiempo de caché) + (Número total de fallos) · (penalización por fallo)

Siendo:

(penalización por fallo) = $128 \cdot 10 \cdot \tau = 1280 \cdot \tau$

Coste total = $26336 \cdot \tau + 48 \cdot 1280 \cdot \tau = 87776 \cdot \tau$

✍ Un computador tiene una memoria principal de 64K palabras de 16 bits por palabra. También tiene una memoria caché totalmente asociativa de 2K palabras y 256 palabras por bloque. La memoria caché es 7 veces más veloz que la memoria principal. Considérese que la memoria caché está inicialmente vacía y que cuando está llena, se reemplaza el bloque que se haya utilizado más recientemente. Supóngase que la UCP ejecuta un bucle 5 veces, en el que accede a 2301 palabras, desde la dirección 0 hasta la dirección 2300. Se pide:

- Especifique el número de bits de los en que se descompone una dirección de memoria principal de este sistema.
- Calcule cuánto tiempo se tardaría en realizar estos accesos a memoria principal si este computador no tuviera memoria caché.
- Analice la evolución del directorio caché.
- Calcule cuántos fallos se producen en la caché a la hora de realizar estos accesos a memoria principal.
- Si cada vez que se produce un fallo primero se mueve el bloque completo de memoria principal a la memoria caché y después se lee el dato de la caché, calcule cuánto tiempo tardaría la UCP en realizar estos accesos a memoria.

Solución:

a)

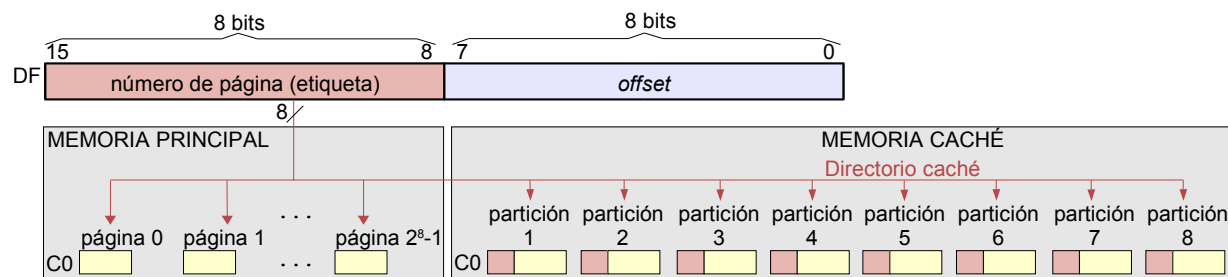
$$|\text{caché}| \equiv (\text{Cantidad de palabras en la caché}) = (\text{Cantidad de conjuntos}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{11} = (1 \text{ conjunto (por ser completamente asociativa)}) \cdot \frac{\text{Cantidad de particiones}}{\text{por conjunto}} \cdot \frac{2^8 \text{ palabras}}{\text{por partición}} \Rightarrow \frac{\text{Cantidad de particiones}}{\text{por conjunto}} = \frac{2^{11}}{2^8} = 2^3 \text{ particiones}$$

$$|\text{MP}| \equiv (\text{Cantidad de palabras en la Memoria Principal}) = (\text{Cantidad de particiones en la Memoria Principal}) \cdot \frac{\text{Cantidad de palabras}}{\text{por partición}}$$

$$2^{16} = (\text{Cantidad de particiones en Memoria Principal}) \cdot \frac{2^8 \text{ palabras}}{\text{por partición}} \Rightarrow (\text{Cantidad de particiones en Memoria Principal}) = \frac{2^{16}}{2^8} = 2^8 \text{ particiones}$$

$$(\text{Cantidad de páginas en la Memoria Principal}) = \frac{\text{Cantidad de particiones en la Memoria Principal}}{\text{Cantidad de conjuntos en la caché}} = \frac{2^8 \text{ particiones}}{2^0 \text{ conjuntos}} = 2^8 \text{ páginas}$$



- b) Si definimos τ como el tiempo de lectura en la caché, el tiempo de lectura en la memoria principal es $7 \cdot \tau$. El tiempo total sin memoria caché es:

$$T_{\text{Sin caché}} \equiv \frac{\text{Tiempo de lectura}}{\text{por palabra}} \cdot \frac{\text{Cantidad de palabras}}{\text{por iteración de bucle}} \cdot \text{Cantidad de iteraciones de bucle} = \frac{7 \cdot \tau}{\text{por palabra}} \cdot \frac{2301 \text{ palabras}}{\text{por iteración de bucle}} \cdot 5 \text{ iteración de bucle} = 80535 \cdot \tau$$

c)

Iteración 1ª:

00 | 00 → **Provoca fallo**, pues la caché está inicialmente vacía. Va a la partición 0 con la etiqueta 00.

00 | 01 }
.. | .. } Aciertos
00 | FF }

01 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 01. Va a la partición 1 con la etiqueta 01.

01 | 01 }
.. | .. } Aciertos
01 | FF }

02 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 02. Va a la partición 2 con la etiqueta 02.

02 | 01 }
.. | .. } Aciertos
02 | FF }

03 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 03. Va a la partición 3 con la etiqueta 03.

03 | 01 }
.. | .. } Aciertos
03 | FF }

04 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 04. Va a la partición 4 con la etiqueta 04.

04 | 01 }
.. | .. } Aciertos
04 | FF }

05 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 05. Va a la partición 5 con la etiqueta 05.

05 | 01 }
.. | .. } Aciertos
05 | FF }

06 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 06. Va a la partición 6 con la etiqueta 06.

06 | 01 }
.. | .. } Aciertos
06 | FF }

07 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 07. Va a la partición 7 con la etiqueta 07.

07 | 01 }
.. | .. } Aciertos
07 | FF }

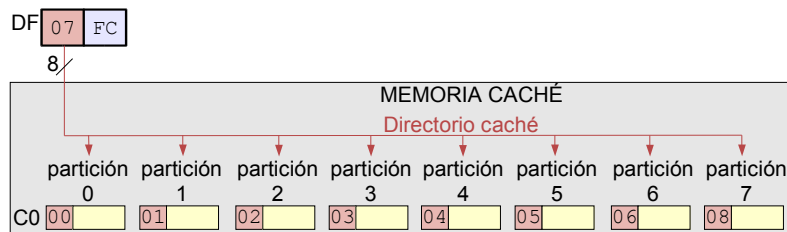
Hasta aquí ha habido 8 fallos y ningún reemplazo. Pero ahora acaba de llenarse la caché. A partir de ahora, en cada fallo habrá reemplazo.

08 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 08.

Reemplaza a la más recientemente usada (la que había en la partición 7 con la etiqueta 07), actualizando la etiqueta a 08.

08 | 01 }
.. | .. } Aciertos
07 | FC }

Estado actual del directorio caché:



Iteración 2ª:

00 00 }
.. | .. } Aciertos, pues la etiqueta 00 está presente en la partición 0 de la caché.
00 FF }

01 00 }
.. | .. } Aciertos, pues la etiqueta 01 está presente en la partición 1 de la caché.
01 FF }

02 00 }
.. | .. } Aciertos, pues la etiqueta 02 está presente en la partición 2 de la caché.
02 FF }

03 00 }
.. | .. } Aciertos, pues la etiqueta 03 está presente en la partición 3 de la caché.
03 FF }

04 00 }
.. | .. } Aciertos, pues la etiqueta 04 está presente en la partición 4 de la caché.
04 FF }

05 00 }
.. | .. } Aciertos, pues la etiqueta 05 está presente en la partición 5 de la caché.
05 FF }

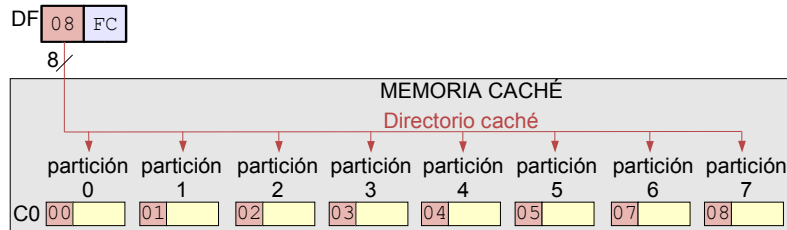
06 00 }
.. | .. } Aciertos, pues la etiqueta 06 está presente en la partición 6 de la caché.
06 FF }

07 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 07.
Reemplaza a la más recientemente usada (la que había en la partición 6 con la etiqueta 06), actualizando la etiqueta a 07.

07 01 }
.. | .. } Aciertos
07 FF }

08 00 }
.. | .. } Aciertos, pues la etiqueta 08 está presente en la partición 7 de la caché.
08 FC }

Estado actual del directorio caché:



Iteración 3ª:

00 00 }
.. | .. } Aciertos, pues la etiqueta 00 está presente en la partición 0 de la caché.
00 FF }

01 00 }
.. | .. } Aciertos, pues la etiqueta 01 está presente en la partición 1 de la caché.
01 FF }

02 00 }
.. | .. } Aciertos, pues la etiqueta 02 está presente en la partición 2 de la caché.
02 FF }

03 00 }
.. | .. } Aciertos, pues la etiqueta 03 está presente en la partición 3 de la caché.
03 FF }

04 00 }
.. | .. } Aciertos, pues la etiqueta 04 está presente en la partición 4 de la caché.
04 FF }

05 00 }
.. | .. } Aciertos, pues la etiqueta 05 está presente en la partición 5 de la caché.
05 FF }

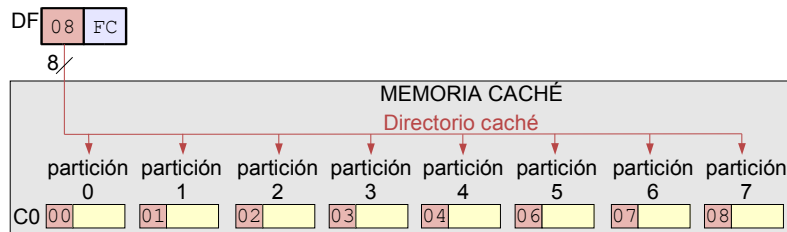
06 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 06.
Reemplaza a la más recientemente usada (la que había en la partición 5 con la etiqueta 05), actualizando la etiqueta a 06.

06 01 }
.. | .. } Aciertos
06 FF }

07 01 }
.. | .. } Aciertos, pues la etiqueta 07 está presente en la partición 6 de la caché.
07 FF }

08 00 }
.. | .. } Aciertos, pues la etiqueta 08 está presente en la partición 7 de la caché.
08 FC }

Estado actual del directorio caché:



Iteración 4ª:

00 00 }
.. | .. } Aciertos, pues la etiqueta 00 está presente en la partición 0 de la caché.
00 FF }

01 00 }
.. | .. } Aciertos, pues la etiqueta 01 está presente en la partición 1 de la caché.
01 FF }

02 00 }
.. | .. } Aciertos, pues la etiqueta 02 está presente en la partición 2 de la caché.
02 FF }

03 00 }
.. | .. } Aciertos, pues la etiqueta 03 está presente en la partición 3 de la caché.
03 FF }

04 00 }
.. | .. } Aciertos, pues la etiqueta 04 está presente en la partición 4 de la caché.
04 FF }

05 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 05.
Reemplaza a la más recientemente usada (la que había en la partición 4 con la etiqueta 04), actualizando la etiqueta a 05.

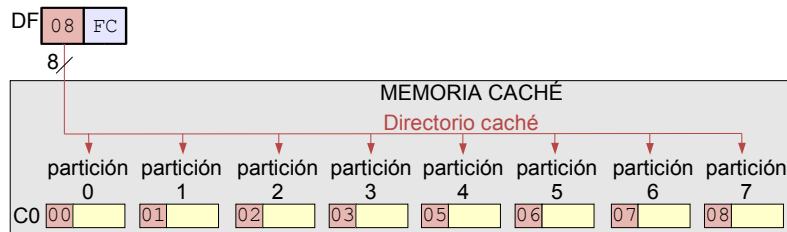
05 01 }
.. | .. } Aciertos
05 FF }

06 00 }
.. | .. } Aciertos, pues la etiqueta 06 está presente en la partición 5 de la caché.
06 FF }

07 01 }
.. | .. } Aciertos, pues la etiqueta 07 está presente en la partición 6 de la caché.
07 FF }

08 00 }
.. | .. } Aciertos, pues la etiqueta 08 está presente en la partición 7 de la caché.
08 FC }

Estado actual del directorio caché:



Iteración 5ª:

00 | 00 }
 .. | .. } Aciertos, pues la etiqueta 00 está presente en la partición 0 de la caché.
 00 | FF }

01 | 00 }
 .. | .. } Aciertos, pues la etiqueta 01 está presente en la partición 1 de la caché.
 01 | FF }

02 | 00 }
 .. | .. } Aciertos, pues la etiqueta 02 está presente en la partición 2 de la caché.
 02 | FF }

03 | 00 }
 .. | .. } Aciertos, pues la etiqueta 03 está presente en la partición 3 de la caché.
 03 | FF }

04 | 00 → **Provoca fallo**, pues en la caché no hay ninguna partición con la etiqueta 04.
 Reemplaza a la más recientemente usada (la que había en la partición 3 con la etiqueta 03, actualizando la etiqueta a 04).

04 | 01 }
 .. | .. } Aciertos
 04 | FF }

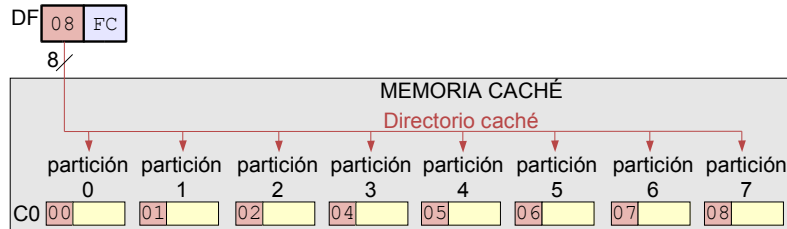
05 | 00 }
 .. | .. } Aciertos, pues la etiqueta 04 está presente en la partición 4 de la caché.
 05 | FF }

06 | 00 }
 .. | .. } Aciertos, pues la etiqueta 06 está presente en la partición 5 de la caché.
 06 | FF }

07 | 01 }
 .. | .. } Aciertos, pues la etiqueta 07 está presente en la partición 6 de la caché.
 07 | FF }

08 | 00 }
 .. | .. } Aciertos, pues la etiqueta 08 está presente en la partición 7 de la caché.
 08 | FC }

Estado actual del directorio caché:



d) Haciendo el recuento de fallos:

- 9 fallos en la primera pasada
- 1 fallo en la segunda pasada
- 1 fallo en la tercera pasada
- 1 fallo en la cuarta pasada
- 1 fallo en la quinta pasada

En total 13 fallos.

e) Según nos especifican, cuando se produce un fallo, el contenido de la dirección fallida no se lleva desde la memoria principal simultáneamente a la caché y a la CPU. Tal como está diseñado este sistema de caché, cuando se produce el fallo, primero se lleva el contenido de las 256 direcciones cuya partición contiene a la dirección pedida desde la memoria principal a la caché y después se lee en la caché la dirección de marras. Por este motivo, siempre se lee en la caché, tanto si no hay fallo, como si lo hay. La conclusión es que:

$$\text{Coste total} = (\text{Número total de direcciones buscadas}) \cdot (\text{Tiempo de caché}) + (\text{Número total de fallos}) \cdot (\text{penalización por fallo})$$

Siendo:

$$(\text{penalización por fallo}) = 256 \cdot 7 \cdot \tau = 1792 \cdot \tau$$

$$\text{Coste total} = 2301 \cdot 5 \cdot \tau + 13 \cdot 1792 \cdot \tau = 34801 \cdot \tau$$