

# ESTRUCTURA Y TECNOLOGÍA DE COMPUTADORES

## *1º de I.T.I. Gestión, I.T.I. Sistemas e I. Superior*

### *Examen de teoría – Parte II: cuestiones teórico-prácticas – 8 de septiembre de 2010*

|  |         |
|--|---------|
| Apellidos:   | Nombre: |
| D.N.I.:  |         |
| Grupo:    Sistemas <input type="checkbox"/> Gestión <input type="checkbox"/> Superior <input type="checkbox"/> |         |

#### PARTE II: CUESTIONES TEÓRICO-PRÁCTICAS

1. (2 puntos) Sea un formato para representar número reales utilizando 16 bits, de los cuales 6 están destinados a representar el exponente en notación sesgada. La base utilizada es 2 y se sigue el estándar IEEE-754 visto en clase en todo lo demás: tipo de normalización, casos especiales (incluidos números desnormalizados) y redondeo. Se pide:
  - a) (0,7) El valor máximo, mínimo normalizado y mínimo sin normalizar, en valor absoluto y distinto de 0, que se puede representar en este formato.
  - b) (0,3) El número que en este formato representa la ristra de 16 bits 0x814C.
  - c) (0,7) La representación en este formato del número  $3 \times 10^5$ . ¿Cuál ha sido el error de redondeo (diferencia entre el valor a representar y el finalmente representado) que se ha cometido al transformar el número a este formato?.
  - d) (0,3) Sea X el número obtenido en el apartado b) e  $Y = 3 \times 10^5$ . ¿Cuál es el resultado de  $X+Y$ ? ¿Existe algún problema a la hora de trabajar con números reales en un ordenador
2. (2 puntos) Utilizando la metodología descrita en los apuntes para la inclusión de nuevas instrucciones en el esquema de implementación multiciclo, realice las fases de análisis y diseño de la siguiente instrucción:
  - a) memadd (\$a), (\$b), \$c: Suma el contenido de la posición de memoria contenida en el registro \$b y el contenido del registro \$c, almacenando el resultado en la posición de memoria contenida en el registro \$a. Observe que las dos direcciones de memoria utilizadas en esta instrucción no incluyen ningún desplazamiento.

Para que la solución del ejercicio se considere correcta, es importante que incluya claramente todos los pasos correspondientes de las fases de análisis y diseño. Las modificaciones al camino de datos y la unidad de control deberán estar explicadas por escrito. De forma complementaria, se pueden realizar modificaciones y anotaciones sobre los diagramas originales para facilitar la explicación.
3. (2 puntos) Supongamos una máquina M con un sistema de memoria virtual que tiene una capacidad de hasta 1024 páginas virtuales, 16 páginas físicas y un tamaño de página de 64 bytes. Dicho sistema de memoria virtual posee un TLB asociativo de 2 vías con reemplazo LRU y postescritura capaz de almacenar 16 entradas de la tabla de páginas. Se pide:
  - a) (0,7 puntos) Especificar el formato de la dirección virtual y de la dirección física. Dibujar un esquema detallado de la tabla de páginas y del TLB, incluyendo todos los bits de control necesarios, justificando su inclusión. Hallar el tamaño de ambas estructuras en bits (sin redondear las entradas a múltiplo de byte).

- b) (0,7 puntos) Mostrar la evolución del contenido de la tabla de páginas y del TLB en la realización de las siguientes referencias: L150, E688 y L1207 (todas las referencias están en base 10, L=lectura, E=escritura). Mostrar también las direcciones físicas resultantes. Asumir que en un principio tanto la memoria principal (y, por tanto, la tabla de páginas) como el TLB están vacíos. Indicar si se produce un fallo o un acierto tanto en la tabla de páginas como en el TLB.
- c) (0,3 puntos) La máquina posee también una memoria caché, tanto para instrucciones como para datos, que tiene una tasa de fallos de instrucciones del 3% y una tasa de fallos de datos del 5%. Si el CPI ideal del procesador es 1, calcular la penalización por fallos (suponiendo que es la misma para instrucciones y datos) para que el CPI real sea 2,9 ciclos si el 35% de las instrucciones accede a datos.
- d) (0,3 puntos) Calcular cuánto más rápida sería la máquina si se le añadiera otra caché de segundo nivel que tuviera una tasa de fallos del 5% y un tiempo de acceso de 15 ciclos en el caso de acierto y de 25 ciclos en el caso de fallo, tanto para instrucciones como para datos.

## Solución al ejercicio 1

a)

Habrà 1 bit para el signo, 6 para el exponente en notaci3n sesgada y 9 para la mantisa. El valor del sesgo es  $S=2^5-1=31$  y la normalizaci3n tiene que serà la misma que en el estàndar IEEE: 1.m.

$$N_{\text{Max}}=1.11111111_2 \times 2^{E_{\text{max}}}, E_{\text{max}}=62-31=31, N_{\text{Max}}=1+(1-2^{-9}) \times 2^{31}=1.998046875 \times 2^{31}$$

$$N_{\text{Max}}=4.290772992 \times 10^9$$

$$N_{\text{min(norm)}}=1.000000000_2 \times 2^{E_{\text{min}}}, E_{\text{min}}=1-31=-30, N_{\text{min(norm)}}=(1+2^{-9}) \times 2^{-30}=2^{-30}=9.313225746154785 \times 10^{-10}$$

$$N_{\text{min(norm)}}=9.3314156402 \times 10^{-10}$$

$$N_{\text{min(desn)}}=0.000000001_2 \times 2^{E_{\text{min}}}, E_{\text{min}}=0-30=-30, N_{\text{min(desn)}}=2^{-9} \times 2^{-30}=2^{-39}$$

$$N_{\text{min(desn)}}=1.818989403546 \times 10^{-12}$$

b)

$$0x814C = 1\ 000000\ 101001100. s=1, e=000000, m=101001100, e=0, E=0-30=-30$$

$$N = -0.101001100_2 \times 2^{-30} = -(10/16 + 6/256) \times 2^{-30} = -0.6484375 \times 2^{-30} = -6.0390448198 \times 10^{-10}$$

c)

$$3 \times 10^5 = M \times 2^E. \quad 10^5 = 2^E, \quad E = 5 / \log_2 10 = 16.609640474437$$

$$3 \times 10^5 = 1.m \times 2^E. \quad (3 \times 2^{0.609640474437}) \times 2^{16} = 4.577650267166 \times 2^{16}$$

$$4 = 100_2$$

$$0.577650267166 \times 16 = 9.24240427465 \quad (1001)$$

$$0.24240427465 \times 16 = 3.87846839437 \quad (0011)$$

$$0.87846839437 \times 16 = 14.05549431 \quad (1110)$$

$N \approx 100.100100111110_2 \times 2^{16} = 1.001001001111 \times 2^{18} \approx 1.001001010_2 \times 2^{18}$  (en negrita el bits de redondeo y guarda, se redondea al alza)

$$m = 001001010, e = E + S = 18 + 31 = 49 = 110001, s = 0 \text{ (número positivo)}$$

0 110001 001001010 (0x624A). El error cometido en el redondeo es de:  $\epsilon = |3 \times 10^5 - (1 + 2/16 + 5/256) \times 2^{18}| = 3 \times 10^5 - 30032 = 32$

d)

$$X + Y = -0.101001100_2 \times 2^{-30} + 1.001001010_2 \times 2^{18} = 0.000000000_2 \times 2^{18} + 1.001001010_2 \times 2^{18} = Y$$

Como se puede observar, a la hora de trabajar con números reales pueden darse situaciones en donde algunas de las propiedades que esperamos que su cumplan ( $X \neq 0, Y \neq 0 \rightarrow X + Y \neq Y$ ) no lo hacen debido a que trabajamos con una precisión limitada lo que da lugar a problemas de redondeo cuando se operan con valores muy dispares. Así, por ejemplo, tampoco se cumple la propiedad asociativa y la comparación de valores reales se debería de hacer por aproximaci3n.

## Solución al ejercicio 2

Análisis:

### 1. Especificación precisa de la semántica de la instrucción:

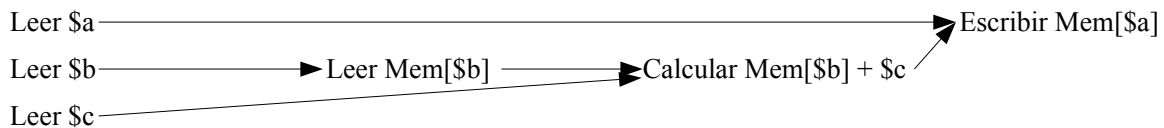
La semántica de “memadd (\$a), (\$b), \$c” es:

$$\text{Mem}[\$a] \leftarrow \text{Mem}[\$b] + \$c$$

### 2. Identificación del trabajo a realizar por cada unidad funcional principal:

- Banco de registros:
  - Leer \$a.
  - Leer \$b.
  - Leer \$c.
- ALU:
  - Sumar  $\text{Mem}[\$b] + \$c$ .
- Memoria:
  - Leer  $\text{Mem}[\$b]$ .
  - Escribir  $\text{Mem}[\$a]$ .

### 3. Establecimiento del orden de precedencia entre las distintas tareas a realizar:



Diseño:

### 1. Definición de la codificación de la instrucción:

Debido a que la instrucción hace referencia a tres registros, debemos usar el formato R. Usaremos la siguiente distribución de campos:

memadd (\$a), (\$b), \$c → 

|        |     |     |     |   |   |
|--------|-----|-----|-----|---|---|
| memadd | \$b | \$c | \$a | 0 | 0 |
|--------|-----|-----|-----|---|---|

### 2. División del trabajo en ciclos:

| Ciclo 1                                      | Ciclo 2            | Ciclo 3                               | Ciclo 4                                   | Ciclo 5                                  |
|--|--------------------|---------------------------------------|---|--|
| $\text{IR} \leftarrow \text{Mem}[\text{PC}]$ | $A \leftarrow \$b$ | $\text{MDR} \leftarrow \text{Mem}[A]$ | $\text{ALUOut} \leftarrow \text{MDR} + B$ | $\text{Mem}[A] \leftarrow \text{ALUOut}$ |
| $\text{PC} \leftarrow \text{PC} + 4$         | $B \leftarrow \$c$ |                                       | $A \leftarrow \$a$                        |  |

### 3. Extensión del camino de datos:

No podemos realizar algunas de las acciones especificadas en la tabla anterior con el camino de datos visto en clase. Por tanto, tendremos que extenderlo de la siguiente manera:

- Para  $\text{MDR} \leftarrow \text{Mem}[A]$ : Ampliar el multiplexor controlado por la señal IoD con una nueva entrada conectada a la salida del registro A. La señal IoD pasa a tener 2 bits de ancho y su significado será:
  - IoD = 00: El PC suministra la dirección para acceder a memoria.
  - IoD = 01: ALUOut proporciona la dirección para acceder a memoria.
  - IoD = 10: El registro A proporciona la dirección para acceder a memoria.
- Para  $\text{ALUOut} \leftarrow \text{MDR} + B$ : Ampliar el multiplexor controlado por SelALUA, conectando una nueva entrada desde la salida del registro MDR. La señal SelALUA para a tener 2 bits de ancho y su significado será:

- SelALUA = 00: El primer operando de la ALU es el PC.
  - SelALUA = 01: El primer operando de la ALU es el registro A.
  - SelALUA = 10: El primer operando de la ALU es el registro MDR.
  - Para  $A \leftarrow \$a$ : Añadir un multiplexor a la entrada del puerto "Reg. de lectura 1" que tenga dos entradas: una proveniente de los bits de la instrucción 25 a 21 (campo rs) y otra proveniente de los bits 20 a 16 (campo rd). Este multiplexor estará controlado por una nueva señal de control que llamaremos FuenteRegA con el siguiente significado:
    - FuenteRegA = 0: El registro a leer está codificado en el campo rs. Esta sería la opción por defecto para los estados del autómata de control original.
    - FuenteRegA = 1: El registro a leer está codificado en el campo rd.
  - Para  $Mem[A] \leftarrow ALUOut$ : Además de la ampliación del multiplexor controlado por IoD mencionada en el primer punto, también es necesario añadir un multiplexor a la entrada del puerto "Dato a escribir". Este multiplexor tendrá dos entradas: una proveniente de la salida del registro B y otra proveniente de la salida del registro ALUOut. Se controlará por una nueva señal que llamaremos FuenteMem y que tendrá el siguiente significado:
    - FuenteMem = 0: El dato a escribir se obtiene del registro B. Esta sería la opción por defecto para los estados del autómata de control original.
    - FuenteMem = 1: El dato a escribir se obtiene del registro ALUOut.
4. **Extensión del control:** Habrá que añadir 3 nuevos estados:

### Solución al ejercicio 3

a)

1024 páginas virtuales =  $2^{10} \rightarrow$  10 bits para el número de página virtual

16 páginas físicas =  $2^4 \rightarrow$  4 bits para el número de página física

Tamaño de página de 64 bytes =  $2^6 \rightarrow$  6 bits para el desplazamiento

Así, los formatos para la dirección virtual y la dirección física son:

#### **Dirección virtual: 16 bits**

|                                |                        |
|--------------------------------|------------------------|
| Número página virtual: 10 bits | Desplazamiento: 6 bits |
|--------------------------------|------------------------|

#### **Dirección física: 10 bits**

|                              |                        |
|------------------------------|------------------------|
| Número página física: 4 bits | Desplazamiento: 6 bits |
|------------------------------|------------------------|

### **Tabla de páginas**

|        |          |          |          |                                |
|--------|----------|----------|----------|--------------------------------|
| Tamaño | 1        | 1        | 1        | 4                              |
|        | <b>V</b> | <b>M</b> | <b>U</b> | <b>Número de página física</b> |
| 0      | 0        | 0        | 0        |                                |
| 1      | 0        | 0        | 0        |                                |
| 2      | 0        | 0        | 0        |                                |
| 3      | 0        | 0        | 0        |                                |
| 4      | 0        | 0        | 0        |                                |
| ...    | ...      | ...      | ...      | ...                            |
| 1023   | 0        | 0        | 0        |                                |

$$\text{Tamaño} = 2^{10} * (3 + 4) \text{ bits} = 7168 \text{ bits}$$

### TLB

|     | <b>V</b> | <b>U</b> | <b>M</b> | <b>ETI</b> | <b>M'</b> | <b>U'</b> | <b>NPF</b> |
|-----|----------|----------|----------|------------|-----------|-----------|------------|
| 0   | 1        | 1        | 1        | 7          | 1         | 1         | 4          |
| 1   |          |          |          |            |           |           |            |
| 2   |          |          |          |            |           |           |            |
| ... | ...      | ...      | ...      | ...        | ...       | ...       | ...        |
| 7   |          |          |          |            |           |           |            |

|     | <b>V</b> | <b>U</b> | <b>M</b> | <b>ETI</b> | <b>M'</b> | <b>U'</b> | <b>NPF</b> |
|-----|----------|----------|----------|------------|-----------|-----------|------------|
| 0   | 1        | 1        | 1        | 7          | 1         | 1         | 4          |
| 1   |          |          |          |            |           |           |            |
| 2   |          |          |          |            |           |           |            |
| ... | ...      | ...      | ...      | ...        | ...       | ...       | ...        |
| 7   |          |          |          |            |           |           |            |

Tenemos 8 conjuntos y 2 vías, lo que nos permite almacenar en el TLB hasta 16 entradas distintas de la tabla de páginas. Por lo tanto, de los 10 bits que conforman el NPV, necesitamos 3 bits para el índice y nos quedan 7 bits para la etiqueta.

$$\text{Tamaño} = 8 * 2 * (3+7+2+4) \text{ bits} = 256 \text{ bits}$$

El bit V (validez) siempre se debe incluir para saber si el contenido de una entrada es válido o no. El bit M (modificación) se debe incluir tanto en la tabla de páginas como en el TLB porque ambos utilizan una estrategia de postescritura. Finalmente, el bit U (uso) se puede incluir o no ya que por sí sólo no sirve para implementar un reemplazo LRU y además, en el caso de la tabla de páginas, no nos dicen qué algoritmo de reemplazo se va a utilizar. No obstante, puesto que no nos dicen que el reemplazo sea aleatorio, lo vamos a incluir para así poder implementar políticas de reemplazo pseudo-LRU.

b)

**Lectura 150 = 0000 0000 1001 0110**

Número de Página Virtual (NPV)= 00 0000 0010 = 2

Desplazamiento= 01 0110 = 22

Para realizar el proceso de traducción de dirección virtual a dirección física, en primer lugar se utiliza el TLB, para ello el NPV se descompone en índice + etiqueta:

$$(NPV) = 00\ 0000\ 0010 \rightarrow \text{índice} = 010 = 2, \text{etiqueta} = 0000\ 000 = 0$$

|     | V   | U   | M   | ETI | M'  | U'  | NPF |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   |     |     |     |     |     |     |     |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

|     | V   | U   | M   | ETI | M'  | U'  | NPF |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   |     |     |     |     |     |     |     |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

Como en la posición 2 del TLB no hay nada en ninguno de los 2 bloques que la conforman, se produce un FALLO DE TLB, siendo necesario acceder a la Tabla de Páginas (TP).

Como en la posición 2 de la TP el bit de validez está a 0 eso significa que la página virtual 2 no está cargada en memoria física, con lo que se produce un FALLO DE PÁGINA. Entonces el sistema operativo accede a disco y trae a memoria física la página correspondiente. Ésta se almacenará en la primera página física libre, es decir, en la página física 0 (ya que la memoria principal está vacía inicialmente). La tabla de páginas queda como sigue:

|      | V   | M   | U   | Número de página física |
|------|-----|-----|-----|-------------------------|
| 0    | 0   | 0   | 0   |                         |
| 1    | 0   | 0   | 0   |                         |
| 2    | 1   | 0   | 0   | 0                       |
| 3    | 0   | 0   | 0   |                         |
| 4    | 0   | 0   | 0   |                         |
| ...  | ... | ... | ... | ...                     |
| 1023 | 0   | 0   | 0   |                         |

|     | V   | U   | M   | ETI | M'  | U'  | NPF |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   | 1   | 1   | 1   | 0   | 0   | 1   | 0   |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

|     | V   | U   | M   | ETI | M'  | U'  | NPF |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   |     |     |     |     |     |     |     |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

Ya se puede realizar la traducción de la dirección virtual a dirección física utilizando la información del TLB:

**Dirección virtual:** 0000 0000 1001 0110 = 150 → **Dirección física:** 0000 01 0110 = 22

Como la operación a realizar es una lectura se pone a 1 el bit U' del TLB. El bit U se actualiza al estar haciendo uso de esta entrada, así como el bit M, pues hemos modificado el bit U' que nos trajimos de la TP.

**Escritura 688 = 0000 0010 1011 0000**

Número de Página Virtual (NPV)= 0000 0010 10 = 10

Desplazamiento= 110000 = 48

Para realizar el proceso de traducción de dirección virtual a dirección física, en primer lugar se utiliza el TLB, para ello el NPV se descompone en índice + etiqueta:

(NPV)= 0000 0010 10 → índice = 010 = 2, etiqueta = 0000 001 = 1

|     | V   | U   | M   | ETI | M'  | U'  | NPF |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   | 1   | 1   | 1   | 0   | 0   | 1   | 0   |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

|     | V   | U   | M   | ETI | M'  | U'  | NPF |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   |     |     |     |     |     |     |     |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

Como en la posición 2 del TLB en uno de los bloques no hay nada y en el otro, al comparar las etiquetas, éstas no coinciden, se produce un FALLO DE TLB, siendo necesario acceder a la Tabla de Páginas (TP).

Como en la posición 10 de la TP el bit de validez está a 0, eso significa que la página virtual 10 no está cargada en memoria física, con lo que se produce un FALLO DE PÁGINA. Entonces el sistema operativo accede a disco y trae a memoria física la página correspondiente. Ésta se almacenará en la siguiente página física libre, es decir, en la página física 1. La tabla de páginas queda como sigue:

|           | V        | M        | U        | Número de página física |
|-----------|----------|----------|----------|-------------------------|
| 0         | 0        | 0        | 0        |                         |
| 1         | 0        | 0        | 0        |                         |
| 2         | 1        | 0        | 0        | 0                       |
| ...       | ...      | ...      | ...      | ...                     |
| <b>10</b> | <b>1</b> | <b>0</b> | <b>0</b> | <b>1</b>                |
| ...       | ...      | ...      | ...      | ...                     |
| 1023      | 0        | 0        | 0        |                         |

|     | V   | U   | M   | ETI | M'  | U'  | NPF |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   | 1   | 0   | 1   | 0   | 0   | 1   | 0   |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

|     | V   | U   | M   | ETI | M'  | U'  | NPF |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   | 1   | 1   | 1   | 1   | 1   | 1   | 1   |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

Ya se puede realizar la traducción de la dirección virtual a dirección física utilizando la información del TLB:

**Dirección virtual: 0000 0010 1011 0000 = 688 → Dirección física: 00 0111 0000 = 112**

Como la operación a realizar es una escritura se pone a 1 tanto el bit M' como el bit U' del TLB. El bit U se actualiza al estar haciendo uso de esta entrada, así como el bit M, pues hemos modificado los bits M' y U' que nos trajimos de la TP. También ponemos a 0 el bit U del otro bloque de ese conjunto para indicar que de los dos es el que lleva más tiempo sin ser referenciado.

**Lectura 1207 = 0000 0100 1011 0111**

Número de Página Virtual (NPV)= 0000 0100 10 = 18

Desplazamiento= 11 0111 = 55

Para realizar el proceso de traducción de dirección virtual a dirección física, en primer lugar se utiliza el TLB, para ello el NPV se descompone en índice + etiqueta:

(NPV)= 0000 0100 10 → índice = 010 = 2, etiqueta = 0000 010 = 2

|     | V   | U   | M   | ETI | M'  | U'  | NPF |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   | 1   | 0   | 1   | 0   | 0   | 1   | 0   |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

|     | V   | U   | M   | ETI | M'  | U'  | NPF |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   | 1   | 1   | 1   | 1   | 1   | 1   | 1   |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

En la posición 2 del TLB están los dos bloques en uso, por lo que tenemos que comprobar las etiquetas para ver si hay acierto. Esto no ocurre, pues la etiqueta buscada es 2 y ésta no está en ninguno de los dos bloques, por lo que se produce un FALLO DE TLB, por lo que habrá que ir a la tabla de páginas.

Como en la posición 18 de la TP el bit de validez está a 0 eso significa que la página virtual 18 no está cargada en memoria física, con lo que se produce un FALLO DE PÁGINA. Entonces el sistema operativo accede a disco y trae a memoria física la página correspondiente. Ésta se almacenará en la siguiente página física libre, es decir, en la página física 2. La tabla de páginas queda como sigue:

|      | V   | M   | U   | Número de página física |
|------|-----|-----|-----|-------------------------|
| 0    | 0   | 0   | 0   |                         |
| 1    | 0   | 0   | 0   |                         |
| 2    | 1   | 0   | 1   | 0                       |
| ...  | ... | ... | ... | ...                     |
| 10   | 1   | 0   | 0   | 1                       |
| ...  | ... | ... | ... | ...                     |
| 18   | 1   | 0   | 0   | 2                       |
| ...  | ... | ... | ... | ...                     |
| 1023 |     |     |     |                         |

Como se ha producido un fallo de TLB y el conjunto está completo, es necesario reemplazar el bloque de esta entrada con bit U = 0 para alojar la nueva traducción. Antes de hacer esto tenemos que comprobar el valor del bit M, es decir, si este tiene valor 1 tendremos que acceder a la tabla de páginas y modificar los valores de los bits M y U con los nuevos valores de los bits U' y M' del TLB, lo cual está reflejado en la tabla de páginas de arriba.

| V | U | M | ETI | M' | U' | NPF |
|---|---|---|-----|----|----|-----|
|---|---|---|-----|----|----|-----|

| V | U | M | ETI | M' | U' | NPF |
|---|---|---|-----|----|----|-----|
|---|---|---|-----|----|----|-----|

|     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   | 1   | 1   | 1   | 2   | 0   | 1   | 2   |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

|     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   |     |     |     |     |     |     |     |
| 1   |     |     |     |     |     |     |     |
| 2   | 1   | 0   | 1   | 1   | 1   | 1   | 1   |
| ... | ... | ... | ... | ... | ... | ... | ... |
| 7   |     |     |     |     |     |     |     |

Ya se puede realizar la traducción de la dirección virtual a dirección física utilizando la información del TLB:

**Dirección virtual:** 0000 0100 1011 0111 = 1207 → **Dirección física:** 00 1011 0111 = 183

Como la operación a realizar es una lectura se pone a 1 el bit U' del TLB. El bit U se actualiza al estar haciendo uso de esta entrada, así como el bit M, pues hemos modificado el bit U'. También ponemos a 0 el bit U del otro bloque de ese conjunto para indicar que de los dos es el que lleva más tiempo sin ser referenciado.

c)

$$CPI_{real} = CPI_{ideal} + TF_{inst} * PF + DpI * TF_{datos} * PF$$

$$2,9 = 1 + 0,03 * PF + 0,35 * 0,05 * PF$$

Despejando en la fórmula anterior, **PF=40 ciclos.**

d)

En este caso, para la caché de primer nivel la penalización por fallo viene dada por la caché de segundo nivel de la siguiente forma:

$$PF = 15 + 0,05 * 25 = 16,25 \text{ ciclos}$$

(también se puede dar por válida la opción  $PF = 0,95 * 15 + 0,05 * 25 = 15,5$  ciclos)

$$CPI_{real} = CPI_{ideal} + TF_{inst} * PF + DpI * TF_{datos} * PF$$

$$CPI_{real} = 1 + 0,03 * 16,25 + 0,35 * 0,05 * 16,25 = 1,772 \text{ ciclos}$$

Por lo tanto, con la nueva caché la máquina es  $2,9 / 1,772 = 1,64$  veces más rápida.