

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 – 11 de febrero de 2010

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

1. (2 puntos) Dada siguiente ristra de 16 bits: 1100 1101 0010 1001, determinar el número decimal que representa, indicando el número decimal concreto, si consideramos las siguientes codificaciones:
- a) (0,2 puntos) Número codificado utilizando la representación sesgada (sesgo normal).
- b) (0,3 puntos) Número codificado utilizando la representación en complemento a la base (B=16).
- c) (0,6 puntos) Número codificado utilizando una representación para números reales en donde se reservan 7 bits para el exponente, B=4, no se permiten números desnormalizados ni otros casos especiales y el redondeo se hace por truncamiento. ¿Cuál es la eficiencia de este código?
- d) (0,9) Número codificado utilizando una representación idéntica al estándar IEEE-754 pero para formatos de 16 bits: 1 bit de signo, 7 de exponente y 8 de mantisa. ¿Cuál es el error de redondeo en esta codificación cuando se representa el número 10^{-3} ?
2. (2 puntos) Utilizando la metodología vista en clase para la inclusión de nuevas instrucciones en el esquema de implementación multiciclo, realice las fases de análisis y diseño de una nueva instrucción, “bltjal \$x, label”, que realiza un salto condicional enlazado (como jal) si el valor del registro x es menor que cero.

| |
|--|
| $\text{si } \$x < 0 \Rightarrow \$ra \leftarrow PC + 4$ $PC \leftarrow \text{label}$ |
|--|

La fase de diseño debe incluir, al menos: las modificaciones a realizar al camino de datos (si las hay), los nuevos estados a añadir al autómata y el valor de las señales de control. La solución se puede dar sobre las figuras que hay en el reverso, pero aun así habrá que justificar en papel las modificaciones realizadas. También será necesario entregar la hoja del examen con el nombre y los apellidos escritos en el lugar correspondiente.

Sugerencia: se puede implementar de forma sencilla la comparación con cero sin necesidad de usar la ALU.

3. (2 puntos) La máquina M posee un sistema de Memoria Virtual con las siguientes características:
- Tabla de páginas de 512 entradas.
 - Memoria física de 256 bytes.
 - Tamaño de página de 32 bytes.
 - TLB asociativo con 4 conjuntos de 2 vías (2 bloques por conjunto). El TLB sigue una estrategia de reemplazo LRU y una política de escritura de postescritura.
- a) (0,5 puntos) Especificar detalladamente el formato de la dirección física y de la dirección virtual. Dibujar un esquema de la tabla de páginas y calcular su tamaño incluyendo los bits de control necesarios (justificando su inclusión). Se parte de una situación inicial en la que las páginas virtuales 1, 3, 5 y 7 se encuentran cargadas en memoria física en las páginas físicas 0, 1, 2 y 3, respectivamente.
- b) (0,3 puntos) Dibujar un esquema detallado del TLB y calcular su tamaño incluyendo todos los bits de control necesarios (justificando su inclusión). El TLB se encuentra inicialmente vacío.
- c) (0,6 puntos) Si durante la ejecución de un programa el procesador genera una lectura a la dirección virtual: 52_{10} , y una escritura a la dirección 190_{10} , indicar cuáles son sus correspondientes NPV (números de página virtual), si habrá acierto o fallo en la Tabla de Páginas, cuáles son sus NPF (números de página física) y la dirección física resultante. Suponer en este apartado que nuestra máquina no dispone de TLB. (En caso necesario, se sabe que la página física 4 se encuentra libre).
- d) (0,6 puntos) Repetir el apartado anterior, pero ahora teniendo en cuenta que nuestra máquina dispone de TLB.

Solución al ejercicio 1

- a) $N=1100110100101001)_2$. El sesgo para una representación sesgada utilizando 16 bits es 2^{15} , por lo que $N_{\text{sesg}} = 1100110100101001)_2 - 2^{15} = 0100110100101001)_2 = \mathbf{19753}$.
- b) $N=1100110100101001)_2 = \text{CD}29$ (número negativo, por lo que estará representado en complemento). $N_{C16} = 10000)_{16} - \text{CD}29)_{16} = 32\text{D}7)_{16} = 13015$, de donde: $N_{C16} = \mathbf{-13015}$.
- c) $N=1\ 1001101\ 00101001$, $s=1$, $e=1001101$ y $m=00101001$. Al ser la base 4 y no permitirse números desnormalizados resulta que la ristra que estamos considerando no es válida ya que la mantisa no está normalizada. Solución: **codificación errónea**.

La eficiencia de un código es un valor entre 0 y 1 que se obtiene, en este caso, de dividir los número reales que se pueden representar entre el número de codificaciones disponibles (2^{16}). El número de codificaciones válidas son todas aquellas en donde los 2 primeros bits de la mantisa no son 00 (hay 2^{14} codificaciones de este tipo, número que se obtiene de pensar que esos dos bits tienen que estar a cero y que los 14 restantes pueden tomar cualquier valor), además tendremos en cuenta la doble representación del cero:

$$\tau = (2^{16} - (2^{14} - 1)) / 2^{16} = (65536 - 16384 + 1) / 65536 = 49153 / 65536 = \mathbf{0,75002}$$

- d) $N=1\ 1001101\ 00101001$, $s=1$, $e=1001101$ y $m=00101001$. El número es negativo ($s=1$), el exponente será: $E=e-S=77-(2^6-1)=77-63=+14$. Tendremos por tanto:

$$N_{\text{IEEE-754}} = -1.M \times 2^E = -1.00101001)_2 \times 2^{14} = -(1+2/16+9/256) \times 2^{14} = -(1+2/16+9/256) \times 2^{14} = -1,16015625 \times 2^{14} = -1,16015625 \times 2^{14} = \mathbf{-19008}$$

$$\text{El número } 10^{-3} = 0,001 = 0,00000000010000011000)_2 = 1,0000011000)_2 \times 2^{-10} =$$

$$0,001 \times 16 = \mathbf{0,016}; 0,016 \times 16 = \mathbf{0,256}; 0,256 \times 16 = \mathbf{4,096}; 0,096 \times 16 = \mathbf{1,536}; 0,536 \times 16 = \mathbf{8,576}; \dots$$

$$10^{-3} - 1,00000110)_2 \times 2^{-10} = 10^{-3} - 0,00099945068359375 = 0,00000054931640625 = 5,4931641 \times 10^{-7}$$

$$\mathbf{\text{Error} = 5,4931641 \times 10^{-7}}$$

Solución al ejercicio 2

ANÁLISIS

En primer lugar, hay que tener en cuenta que la comprobación de la condición $\$x < 0$ se puede realizar de varias formas:

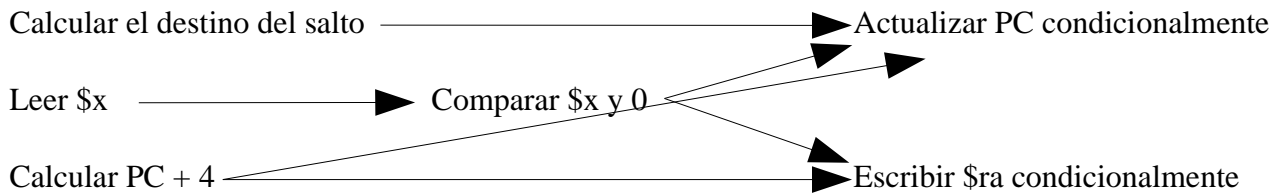
- La forma más general de comparar dos valores es utilizar la ALU para restarlos, tal y como hace la instrucción `Slt`.
- Dado que en este caso se trata solo de saber si un valor es menor que 0, bastaría con mirar su bit de signo.

Cualquiera de las dos formas anteriores valdría. Utilizaremos la segunda por ser más sencilla y para evitar tener que usar la ALU para la comparación. De esta forma, el uso que se hará de las unidades funcionales para esta instrucción será:

- Banco de registros:
 - Leer $\$x$.
 - Escribir $\text{PC} + 4$ en $\$ra$ (\$31) condicionalmente.
- ALU:
 - Calcular $\text{PC} + 4$ (se calcula siempre en el ciclo 1).
 - Calcular la dirección de destino del salto (se calcula siempre en el ciclo 2).
- Memoria: no se accede a memoria.

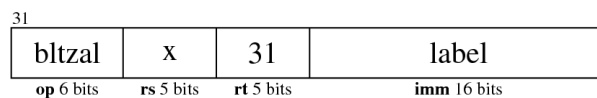
Además, se escribe en el PC si el resultado de la comparación es que $\$x$ es menor que 0.

Las dependencias entre las operaciones determinan el orden en el que se han de realizar:



DISEÑO

Debido a que la instrucción tiene dos operandos en registros (se lee \$x y se escribe \$ra) y otro operando inmediato (la etiqueta de destino), debemos usar el formato I. Usaremos la siguiente distribución de campos:



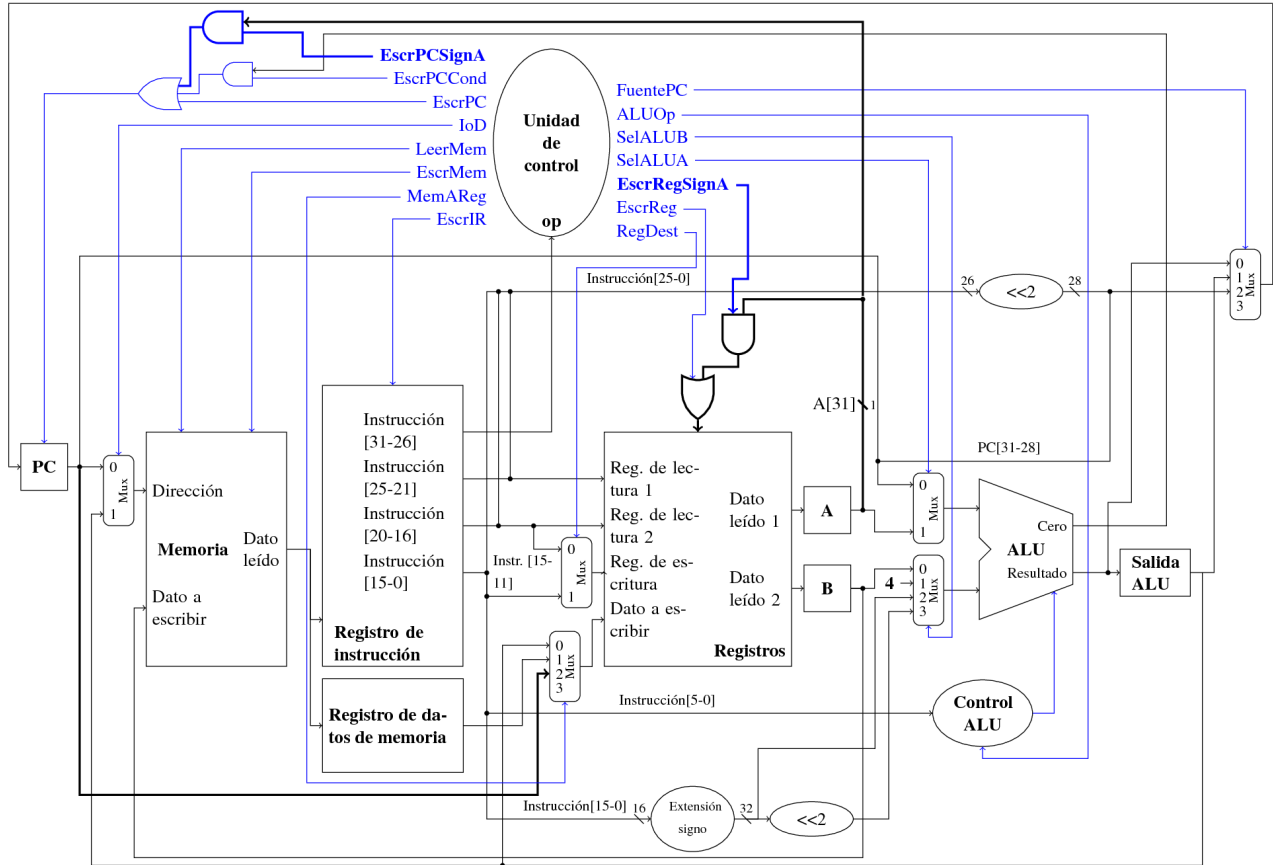
El trabajo se distribuirá en ciclos de la manera siguiente:

| Ciclo 1 | Ciclo 2 | Ciclo 3 |
|------------------|--------------------------|----------------------------|
| IR ← Memoria[PC] | A ← \$x | si A[31] = 1 ⇒ PC ← ALUOut |
| PC ← PC + 4 | ALUOut ← PC + (imm << 2) | si A[31] = 1 ⇒ \$31 ← PC |

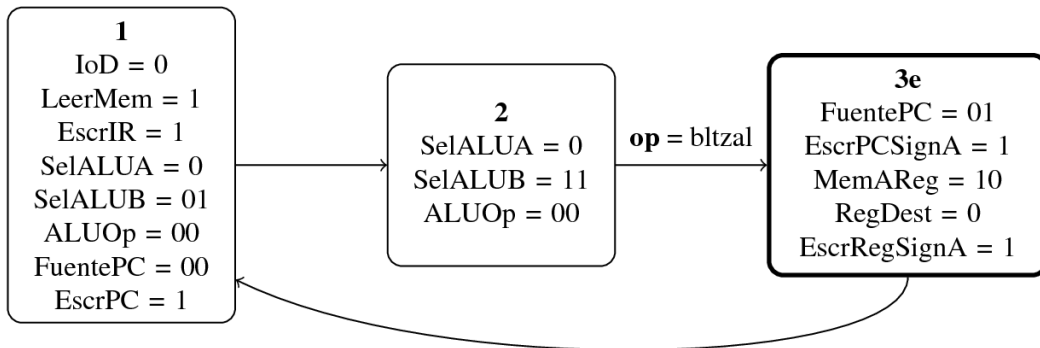
De las acciones especificadas en la tabla anterior, no podemos realizar las del tercer ciclo sin modificar el camino de datos original. Las modificaciones son las siguientes:

- Para “si A[31] = 1 ⇒ PC ← ALUOut”:
 La salida del registro ALUOut ya está conectada con el PC, pero tenemos que lograr que la señal de habilitación de escritura se active sólo cuando A[31] sea 1. Para ello, seguiremos un esquema análogo al de la señal EscrPCCond:
 Añadiremos una puerta AND conectada a la puerta OR que controla la señal de habilitación de escritura del PC. Esta puerta AND tendrá 2 entradas: una conectada al bit 31 del registro A y la otra conectada a una nueva señal de control que llamaremos EscrPCSignA. Cuando la señal EscrPCSignA esté activa, se escribirá en el PC si el bit de signo de A está activo.
- Para “si A[31] = 1 ⇒ \$31 ← PC”:
 Tenemos que conectar la salida del registro PC a la entrada de datos del banco de registros. Para ello, ampliaremos el multiplexor controlado por la señal de control MemAReg, la cual pasa a tener dos bits en lugar de uno. Cuando MemAReg valga 10, se escribirá en el banco de registros el valor del registro PC (si se escribe algo). El significado del resto de valores posibles de MemAReg que se indican en la tabla 2 no cambia (simplemente se añade un cero a la izquierda a los valores).
 Además, necesitamos que la escritura en el banco de registros sea condicional. Actualmente, la escritura está controlada por la señal EscrReg, que causa una escritura incondicional.
 Añadiremos una puerta OR conectada a la señal de habilitación de escritura del banco de registros. A esta puerta OR se le conectará, por un lado, la señal de control EscrReg (para que el significado de ésta no cambie) y, por otro lado, una puerta AND adicional. A esta puerta AND se le conectará el bit 31 del registro A y una nueva señal de control que llamaremos EscrRegSignA. Cuando la señal EscrRegSignA esté activa, se escribirá en el banco de registros si el bit de signo de A está activo.

Las modificaciones realizadas se pueden ver a continuación:



Habrá que añadir un nuevo estado al autómata de control:



Solución al ejercicio 3

a) (0'5 puntos)

512 páginas virtuales = $2^9 \rightarrow$ 9 bits para el número de página virtual
 Memoria física de 256 Bytes = $2^8 \rightarrow$ 8 bits para la dirección física
 Tamaño de página de 32 bytes = $2^5 \rightarrow$ 5 bits para el desplazamiento

Con todo ello tenemos:

Dirección virtual: 14 bits

| | |
|-------------------------------|------------------------|
| Número página virtual: 9 bits | Desplazamiento: 5 bits |
|-------------------------------|------------------------|

Dirección física: 8 bits

| | |
|------------------------------|------------------------|
| Número página física: 3 bits | Desplazamiento: 5 bits |
|------------------------------|------------------------|

Tabla de páginas:

| Tamaño | 1 | 1 | 1 | 3 bits |
|---------|---|---|---|----------------------|
| | V | M | U | Número Página Física |
| 0 | 0 | 0 | 0 | - |
| 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | - |
| 3 | 1 | 0 | 0 | 1 |
| 4 | 0 | 0 | 0 | - |
| 5 | 1 | 0 | 0 | 2 |
| 6 | 0 | 0 | 0 | - |
| 7 | 1 | 0 | 0 | 3 |
| 8 | 0 | 0 | 0 | - |
| ... | 0 | 0 | 0 | - |
| 2^9-2 | 0 | 0 | 0 | - |
| 2^9-1 | 0 | 0 | 0 | - |

Tamaño = $2^9 * (3 + 3)$ bits = **3072 bits**

Tamaño (ajustando el tamaño a 1 byte por cada entrada) = $2^9 * 1$ byte = **512 bytes**

b) (0'3 puntos)

| | V | U | M | Etí | M' | U' | NPF |
|---|---|---|---|-----|----|----|-----|
| 0 | 1 | 1 | 1 | 7 | 1 | 1 | 3 |
| 1 | | | | | | | |
| 2 | | | | | | | |
| 3 | | | | | | | |

| | V | U | M | Etí | M' | U' | NPF |
|---|---|---|---|-----|----|----|-----|
| 0 | 1 | 1 | 1 | 7 | 1 | 1 | 3 |
| 1 | | | | | | | |
| 2 | | | | | | | |
| 3 | | | | | | | |

Tenemos 4 conjuntos. Por lo tanto, de los 9 bits que conforman el NPV, necesitamos para el índice 2 bits y nos quedan 7 bits para la etiqueta.

Tamaño = $4 * 2 * (3 + 7 + 2 + 3)$ bits = **120 bits**

c) (0'6 puntos)

LECTURA 52)₁₀ = 00 0000 0011 0100 (recordar que la dirección virtual tiene 14 bits)

Descomponemos la dirección virtual en: NPV (los 9 bits superiores) + desplazamiento (los 5 bits inferiores)

NPV = $00\ 0000\ 001 = 1$)₁₀

desplazamiento = $1\ 0100 = 20$)₁₀

Por lo tanto el NPV es 1.

¿Hay acierto en la tabla de páginas?

| | V | M | U | Número Página Física |
|----------|----------|----------|----------|----------------------|
| 0 | 0 | 0 | 0 | - |
| 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | - |
| 3 | 1 | 0 | 0 | 1 |
| 4 | 0 | 0 | 0 | - |
| 5 | 1 | 0 | 0 | 2 |
| 6 | 0 | 0 | 0 | - |

| | | | | |
|---------|---|---|---|---|
| 7 | 1 | 0 | 0 | 3 |
| 8 | 0 | 0 | 0 | - |
| ... | 0 | 0 | 0 | - |
| 2^9-2 | 0 | 0 | 0 | - |
| 2^9-1 | 0 | 0 | 0 | - |

Como en la posición 1 de la TP el bit de validez está a 1 eso significa que la página virtual 1 sí está cargada en memoria física, con lo que se produce un **ACIERTO DE PÁGINA**.
Número de Página Física, **NPF = 0**.

Ya se puede realizar la traducción de dirección virtual a dirección física utilizando el NPF+desplazamiento:

| Dirección física: 8 bits | |
|--------------------------|------------------------|
| NPF: 3 bits | Desplazamiento: 5 bits |

Dirección virtual: 00 0000 0011 0100 = 52_{10} → **Dirección física:** 0001 0100 = 20_{10}

Hay que tener en cuenta que tenemos que actualizar el bit de uso (U) de la posición 1 de la tabla de páginas ya que acabamos de hacer uso de la página física que en esta se referencia. El bit de modificación (M) no hay que actualizarlo pues ha sido una lectura.

ESCRITURA 190₁₀ = **00 0000 1011 1110** (recordar que la dirección virtual tiene 14 bits)

Descomponemos la dirección virtual en: NPV (los 9 bits superiores) + desplazamiento (los 5 bits inferiores)

NPV = 00 0000 101 = 5_{10}

desplazamiento = 1 1110 = 30_{10}

Por lo tanto el NPV es 5.

¿Hay acierto en la Tabla de Páginas?

| | V | M | U | Número Página Física |
|---------|----------|----------|----------|----------------------|
| 0 | 0 | 0 | 0 | - |
| 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | - |
| 3 | 1 | 0 | 1 | 1 |
| 4 | 0 | 0 | 0 | - |
| ⇒ 5 | 1 | 0 | 0 | 2 |
| 6 | 0 | 0 | 0 | - |
| 7 | 1 | 0 | 0 | 3 |
| 8 | 0 | 0 | 0 | - |
| ... | 0 | 0 | 0 | - |
| 11 | 0 | 0 | 0 | - |
| ... | 0 | 0 | 0 | - |
| 2^9-2 | 0 | 0 | 0 | - |
| 2^9-1 | 0 | 0 | 0 | - |

Como en la posición 5 de la TP el bit de validez está a 1 eso significa que la página virtual 5 sí está cargada en memoria física, con lo que se produce un **ACIERTO DE PÁGINA**.
Número de Página Física, **NPF = 2**.

Ya se puede realizar la traducción de dirección virtual a dirección física utilizando el NPF+desplazamiento:

Dirección física: 8 bits

| | |
|-------------|------------------------|
| NPF: 3 bits | Desplazamiento: 5 bits |
|-------------|------------------------|

Dirección virtual: $00\ 0000\ 1011\ 1110 = 190)_{10} \rightarrow$ **Dirección física:** $0101\ 1110 = 94)_{10}$

Al igual que antes, tenemos que actualizar el bit de uso de la posición 5 de la tabla de páginas, pues hemos hecho uso de la página a la que referencia, pero, además, tenemos que actualizar el bit de modificación ya que se trata de una escritura. La tabla de páginas queda finalmente de la siguiente forma:

| | V | M | U | Número Página Física |
|-----------------|----------|----------|----------|----------------------|
| 0 | 0 | 0 | 0 | - |
| 1 | 1 | 0 | 1 | 0 |
| 2 | 0 | 0 | 0 | - |
| 3 | 1 | 0 | 0 | 1 |
| 4 | 0 | 0 | 0 | - |
| \Rightarrow 5 | 1 | 1 | 1 | 2 |
| 6 | 0 | 0 | 0 | - |
| 7 | 1 | 0 | 0 | 3 |
| 8 | 0 | 0 | 0 | - |
| ... | 0 | 0 | 0 | - |
| 11 | 0 | 0 | 0 | - |
| ... | 0 | 0 | 0 | - |
| 2^9-2 | 0 | 0 | 0 | - |
| 2^9-1 | 0 | 0 | 0 | - |

d) (0'6 puntos)

LECTURA 52) $_{10} = 00\ 0000\ 0011\ 0100$ (recordar que la dirección virtual tiene 14 bits)

Descomponemos la dirección virtual en: NPV (los 9 bits superiores) + desplazamiento (los 5 bits inferiores)

NPV = $00\ 0000\ 001 = 1)_{10}$

desplazamiento = $1\ 0100 = 20)_{10}$

Por lo tanto el NPV es 1.

¿Hay acierto en el TLB?

NPV = $00\ 0000\ 001 = 1)_{10} \rightarrow$ Etiqueta $\rightarrow 00\ 0000\ 0 = 0)_{10}$

Posición en TLB $\rightarrow 01 = 1)_{10}$

| | V | U | M | Etí | M' | U' | NPF |
|---|----------|----------|----------|------------|-----------|-----------|------------|
| 0 | 0 | | | | | | |
| 1 | 0 | | | | | | |
| 2 | 0 | | | | | | |
| 3 | 0 | | | | | | |

| | V | U | M | Etí | M' | U' | NPF |
|---|----------|----------|----------|------------|-----------|-----------|------------|
| 0 | 0 | | | | | | |
| 1 | 0 | | | | | | |
| 2 | 0 | | | | | | |
| 3 | 0 | | | | | | |

Inicialmente el TLB está vacío; concretamente, el conjunto 1 tiene los dos bits de validez a 0, por lo que se produce un fallo y tenemos que acudir a la tabla de páginas.

¿Hay acierto en la Tabla de Páginas?

| | V | M | U | Número Página Física |
|-----------------|----------|----------|----------|----------------------|
| 0 | 0 | 0 | 0 | - |
| \Rightarrow 1 | 1 | 0 | 0 | 0 |

| | | | | |
|---------|---|---|---|---|
| 2 | 0 | 0 | 0 | - |
| 3 | 1 | 0 | 0 | 1 |
| 4 | 0 | 0 | 0 | - |
| 5 | 1 | 0 | 0 | 2 |
| 6 | 0 | 0 | 0 | - |
| 7 | 1 | 0 | 0 | 3 |
| 8 | 0 | 0 | 0 | - |
| ... | 0 | 0 | 0 | - |
| 2^9-2 | 0 | 0 | 0 | - |
| 2^9-1 | 0 | 0 | 0 | - |

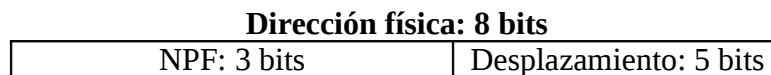
Como en la posición 1 de la TP el bit de validez está a 1 eso significa que la página virtual 1 sí está cargada en memoria física, con lo que se produce un **ACIERTO DE PÁGINA**.
Número de Página Física, **NPF = 0**.

Ahora lo que hacemos es actualizar el TLB, que quedará de la siguiente manera:

| | V | U | M | Et _i | M' | U' | NPF |
|---|---|---|---|-----------------|----|----|-----|
| 0 | 0 | | | | | | |
| 1 | 1 | 1 | 1 | 0 ₁₀ | 0 | 1 | 0 |
| 2 | 0 | | | | | | |
| 3 | 0 | | | | | | |

| | V | U | M | Et _i | M' | U' | NPF |
|---|---|---|---|-----------------|----|----|-----|
| 0 | 0 | | | | | | |
| 1 | 0 | | | | | | |
| 2 | 0 | | | | | | |
| 3 | 0 | | | | | | |

Ya se puede realizar la traducción de dirección virtual a dirección física utilizando el NPF+desplazamiento:



Dirección virtual: 00 0000 0011 0100 = 52₁₀ → **Dirección física:** 0001 0100 = 20₁₀

En este caso tenemos que actualizar el bit de uso (U') de la página en el TLB pero no en la tabla de páginas, pues el TLB sigue una política de postescritura.

ESCRITURA 190₁₀ = 00 0000 1011 1110 (recordar que la dirección virtual tiene 14 bits)
Descomponemos la dirección virtual en: NPV (los 9 bits superiores) + desplazamiento (los 5 bits inferiores)

NPV = 00 0000 101 = 5₁₀
desplazamiento = 1 1110 = 30₁₀

Por lo tanto el NPV es 5.

¿Hay acierto en el TLB?

NPV = 00 0000 101 = 5₁₀ → Etiqueta → 00 0000 1 = 1₁₀
Posición en TLB → 01 = 1₁₀

| | V | U | M | Et _i | M' | U' | NPF |
|---|---|---|---|-----------------|----|----|-----|
| 0 | 0 | | | | | | |
| 1 | 1 | 1 | 1 | 0 ₁₀ | 0 | 1 | 0 |
| 2 | 0 | | | | | | |
| 3 | 0 | | | | | | |

| | V | U | M | Et _i | M' | U' | NPF |
|---|---|---|---|-----------------|----|----|-----|
| 0 | 0 | | | | | | |
| 1 | 0 | | | | | | |
| 2 | 0 | | | | | | |
| 3 | 0 | | | | | | |

Como podemos ver, en una de las entradas del conjunto 1 el bit de validez está puesto a 1, por lo que tenemos que proceder a comprobar si las etiquetas coinciden y, por lo tanto, se produce un acierto. Esto no es así, ya que la etiqueta en el TLB vale 0 y la etiqueta de la referencia es 1, por

lo que tenemos que comprobar la otra entrada. En la otra entrada el bit de validez está puesto a 0, lo que significa que tenemos un fallo de TLB y tenemos que acceder a la tabla de páginas.

¿Hay acierto en la Tabla de Páginas?

| | V | M | U | Número Página Física |
|---------|---|---|---|----------------------|
| 0 | 0 | 0 | 0 | - |
| 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | - |
| 3 | 1 | 0 | 0 | 1 |
| 4 | 0 | 0 | 0 | - |
| ⇒ 5 | 1 | 0 | 0 | 2 |
| 6 | 0 | 0 | 0 | - |
| 7 | 1 | 0 | 0 | 3 |
| 8 | 0 | 0 | 0 | - |
| ... | 0 | 0 | 0 | - |
| 11 | 0 | 0 | 0 | - |
| ... | 0 | 0 | 0 | - |
| 2^9-2 | 0 | 0 | 0 | - |
| 2^9-1 | 0 | 0 | 0 | - |

Como en la posición 5 de la TP el bit de validez está a 1 eso significa que la página virtual 5 sí está cargada en memoria física, con lo que se produce un **ACIERTO DE PÁGINA**.

Número de Página Física, **NPF = 2**.

Ahora actualizamos el TLB, que quedará de la siguiente manera:

| | V | U | M | Et _i | M' | U' | NPF |
|---|---|---|---|------------------|----|----|-----|
| 0 | 0 | | | | | | |
| 1 | 1 | 0 | 1 | 0) ₁₀ | 0 | 1 | 0 |
| 2 | 0 | | | | | | |
| 3 | 0 | | | | | | |

| | V | U | M | Et _i | M' | U' | NPF |
|---|---|---|---|------------------|----|----|-----|
| 0 | 0 | | | | | | |
| 1 | 1 | 1 | 1 | 1) ₁₀ | 1 | 1 | 2 |
| 2 | 0 | | | | | | |
| 3 | 0 | | | | | | |

Como vemos, el bit U de la primera vía del conjunto 1 se ha puesto a 0 para poder aplicar correctamente una política de reemplazo LRU en caso necesario. También hemos actualizado el bit de uso (U') y el bit de modificación (M') de la página 5 en el TLB pero no en la tabla de páginas, pues el TLB sigue una política de postescritura.

Ya se puede realizar la traducción de dirección virtual a dirección física utilizando el NPF+desplazamiento:

Dirección física: 8 bits

| | |
|-------------|------------------------|
| NPF: 3 bits | Desplazamiento: 5 bits |
|-------------|------------------------|

Dirección virtual: 00 0000 1011 1110 = 190)₁₀ → Dirección física: 0101 1110 = 94)₁₀