

# PARCIAL COMPLETO TEMA 3 RESUELTO

## ENUNCIADO COMPLETO

Considerar el siguiente código en un **MIPS**:

```
i1      or    $25,$14,$23
i2      add   $20,$16,$25
i3      addi  $3,$0,19
i4 lab0: addu  $26,$27,$18
i5      sub   $13,$24,$15
i6      j     lab1
      ----
      ----
i7 lab1: xori  $28,$12,17
i8      lw    $26, 4A04($0)
i9      subi  $26,$26,7
i10     add   $28,$29,$13
i11     slt   $14,$25,$16
i12     mult  $15,$27,$18
i13     j     lab2
      ----
      ----
i14 lab2: or    $21,$11,$29
i15     sw    $0, CA38($0)
i16     addi  $3,$3,-1
i17     bne   $3,$0,lab0
```

El valor de las etiquetas es el siguiente: lab0= 2BCh, lab1= 752A4h, lab2= 79AE0h. De inicio, la memoria de datos tiene el valor constante A en todas las posiciones (cada word de la memoria de datos tiene la constante A).

Supongamos una memoria principal (MM) de  $2^6$  MBytes, y caches de instrucciones (I\$) y datos (D\$). La I\$ tiene 4Kbytes, con organización **(1)** (ver abajo), blocks de  $2^2$  words. La D\$ tiene 2Kbytes, con organización **(0)** y blocks de 64 bytes.

**(0)**: Directa, **(1)**: Asociativa por conjuntos de 2-way

A. (5 puntos) Para la I\$, calcular:

- Formato de una dirección física (con sus campos y tamaños)
- Secuencia de direcciones generadas en la etapa de búsqueda de instrucciones
- Evolución de la zona de control y datos de la cache al ejecutarse completamente este código (escribir sólo los bloques de cache involucrados en la ejecución de este código; la cache está inicialmente vacía). Indicar los valores en la cache que quedan al final de la primera iteración.
- Calcular el número de fallos después de la primera iteración y después de la segunda y el índice de fallos (en %) tras la ejecución completa del código

B. (3 puntos) Para la D\$, calcular

- Formato de una dirección física (con sus campos y tamaños)
- Secuencia de direcciones de la memoria de datos y contenido su contenido (**áreas de control y datos**) tras la ejecución del código (como en el caso anterior, escribir sólo los bloks involucrados y cache de datos está vacía de inicio)
- Calcular el índice de fallos (%)

C. (0.75) Considerar que el procesador trabaja a 1,2 GHz con penalización de 13 ciclos para la I\$ y 15 ciclos para la D\$. Calcular:

- Tiempo medio de acceso a memoria (AMAT) en la ejecución de este código
- CPI efectivo (suponer que  $CPI_{base}$  es 2,1)
- Tiempo de ejecución (en ns.)

D. (0.5) Incorporamos una memoria virtual a nuestro sistema. ¿Cuál será el tamaño mínimo de página virtual (en Kbytes) queremos que el acceso a la TLB pueda realizarse en paralelo con el acceso a la cache en todo momento?

E. (0.75) Pregunta de memoria entrelazada

A) a) direccionamiento de 26 bits ( $2^{26}$  bytes de memoria principal)

It:  $2^{12}$  bytes, bloques de  $2^4$  bytes (en MIPS, word =  $2^2$  bytes)

$$\Rightarrow \# \text{ bloques} = \frac{2^{12}}{2^4} = 2^8 \Rightarrow \# \text{ sets} = \frac{2^8}{\text{asociatividad}} = 2^7.$$

TAG	sets	w	b
15	7	2	2

b) Sumamos de 4 en 4 (palabras de 4 bytes):

2B0, 2B4, 2B8, 2BC, 2C0, 2C4, 752A4,

752A8, 752AC, 752B0, 752B4, 752B8,

752BC, 79AE0, 79AE4, 79AE8, 79AEC.

Lo subrayado se repite 18 veces más (en total, son 19).

c) Evolución de la zona de control y datos:



Dirección	TAG	Set	1 <sup>ra</sup> it. A/F	2da it. A/F
280	0h	2Bh	F	Añetón TODOS
284	"	"	A	
288	"	"	A	
28C	"	"	A	
2C0	0h	2Ch	F	
2C4	"	"	A	
752A4	EAh	2Ah	F	
752A8	"	"	A	
752AC	"	"	A	
752B0	EAh	2Bh	F	
752B4	"	"	A	
752B8	"	"	A	
752BC	"	"	A	
79AE0	F3h	2Eh	F	
79AE4	"	"	A	
79AE8	"	"	A	
79AEC	"	"	A	

Todos fallos compulsory.

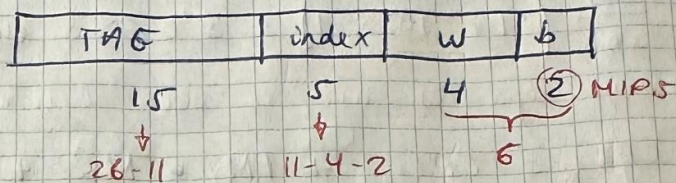
$$IC = 3 + 14 \cdot 19 = 269.$$

$$\text{Miss Rate} \pm = \frac{5}{269} = 0.01858 = 1.858\%.$$

SET	V	TAG	word 3	word 2	word 1	word 0
2Bh	1	0h	i4	i3	i2	i1
	1	EAh	i13	i12	i11	i10
2Ch	1	0h			i6	i5
	0					
2Ah	1	EAh	i9	i8	i7	—
	0					
2Eh	1	F3h	i17	i16	i15	i14
	0					



B) a) Tamaño:  $2^4$  bytes, directa, bloques de  $2^6$  bytes.



b) Secuencia de direcciones:

4A04, CA38, y esto se hace en total 19 veces. (Hay que mirar los loads y stores).

Dirección	TAG	index	A/F 1 <sup>ra</sup> it.	A/F 2 <sup>da</sup> it.
4A04	0009h	08h	F	F
CA38	0019h	08h	F	F

Siempre se solapan, van al mismo conjunto y es asignación directa.

Words							
TAG	v	15	14	13	12	...	0
08h 0019h	1	A	0	A	A	...	A

Nota: block offset de CA38 es 11 1000, luego va a la palabra 1110b = 14d

$$c) f = 1.2 \text{ GHz} \Rightarrow T_c = \frac{1}{1.2 \cdot 10^9} = 0.83 \text{ ns.}$$

$$PM_I = 13, PM_D = 15.$$

$MR_I = 0.0186$ ,  $MR_D = 1$  (estos dos se sacan de otros apartados).

$$\text{Referencias totales} = 269 + 38 = 307.$$

PM = penalty miss.

MR = miss rate.



$$ANAT = \frac{269}{307} \left( 1 + 13 \cdot 0'0186 \right) +$$

$$+ \frac{38}{307} \left( 1 + 15 \cdot 1 \right) = 3'0685 \text{ cc} = 2'557 \text{ ns.}$$

Que yo sepa es así, si veis algún fallo avisadme :)

$$b) CPI_y = CPI_{base} + PM_I \cdot MR_I + \frac{38}{269} PM_D \cdot I_D =$$

$$= 2'1 + 13 \cdot 0'0186 + \frac{38}{269} \cdot 15 = 4'46.$$

$$c) T_g = \frac{IC \cdot CPI_y}{f} = \frac{269 \cdot 4'46}{1'2 \cdot 10^9} = 999'97 \text{ ns}$$

D) No lo hemos dado, pero yo entiendo que al menos debe ocupar los bits que cubre la caché, que son 11, luego al menos debe ser de  $2^{11}$  bytes = 2Kbytes.

En el direccionamiento, esto es, al menos 26 - TAG.

E) Bloque de  $2^3$  palabras, como es MIPS  
 $\Rightarrow$  bloques de  $2^5$  bytes.

Bus de  $2^2$  bytes = 1 palabra.

Como hay  $2^3$  módulos, todos se pueden ir calculando a la vez, pero las palabras se devuelven de una en una, pues no cabe más de 1 palabra en el bus.

Esto es: 1 de dar la dirección,  
 20 de buscar el dato y 1\*8 de devolver las palabras = 29.

# PARCIAL COMPLETO TEMA 3 RESUELTO

## ENUNCIADO COMPLETO

Considerar el siguiente código en un MIPS:

```
i1      or    $25,$14,$23
i2      add   $20,$16,$25
i3      addi  $3,$0,19
i4 lab0: addu  $26,$27,$18
i5      sub   $13,$24,$15
i6      j     lab1
      ----
      ----
i7 lab1: xori  $28,$12,17
i8      lw    $26, 4A04($0)
i9      subi  $26,$26,7
i10     add   $28,$29,$13
i11     slt   $14,$25,$16
i12     mult  $15,$27,$18
i13     j     lab2
      ----
      ----
i14 lab2: or    $21,$11,$29
i15     sw    $0, CA38($0)
i16     addi  $3,$3,-1
i17     bne   $3,$0,lab0
```

El valor de las etiquetas es el siguiente: lab0= 2BCh, lab1= 752A4h, lab2= 79AE0h. De inicio, la memoria de datos tiene el valor constante A en todas las posiciones (cada word de la memoria de datos tiene la constante A).

Supongamos una memoria principal (MM) de  $2^6$  MBytes, y caches de instrucciones (I\$) y datos (D\$). La I\$ tiene 4Kbytes, con organización **(1)** (ver abajo), blocks de  $2^2$  words. La D\$ tiene 2Kbytes, con organización **(0)** y blocks de 64 bytes.

**(0)**: Directa, **(1)**: Asociativa por conjuntos de 2-way

A. (5 puntos) Para la I\$, calcular:

- Formato de una dirección física (con sus campos y tamaños)
- Secuencia de direcciones generadas en la etapa de búsqueda de instrucciones
- Evolución de la zona de control y datos de la cache al ejecutarse completamente este código (escribir sólo los bloques de cache involucrados en la ejecución de este código; la cache está inicialmente vacía). Indicar los valores en la cache que quedan al final de la primera iteración.
- Calcular el número de fallos después de la primera iteración y después de la segunda y el índice de fallos (en %) tras la ejecución completa del código

B. (3 puntos) Para la D\$, calcular

- Formato de una dirección física (con sus campos y tamaños)
- Secuencia de direcciones de la memoria de datos y contenido su contenido (**áreas de control y datos**) tras la ejecución del código (como en el caso anterior, escribir sólo los bloks involucrados y cache de datos está vacía de inicio)
- Calcular el índice de fallos (%)

C. (0.75) Considerar que el procesador trabaja a 1,2 GHz con penalización de 13 ciclos para la I\$ y 15 ciclos para la D\$. Calcular:

- Tiempo medio de acceso a memoria (AMAT) en la ejecución de este código
- CPI efectivo (suponer que  $CPI_{base}$  es 2,1)
- Tiempo de ejecución (en ns.)

D. (0.5) Incorporamos una memoria virtual a nuestro sistema. ¿Cuál será el tamaño mínimo de página virtual (en Kbytes) queremos que el acceso a la TLB pueda realizarse en paralelo con el acceso a la cache en todo momento?

E. (0.75) Pregunta de memoria entrelazada



A) a) Direcccionamiento de 26 bits ( $2^{26}$  bytes de memoria principal)

It:  $2^{12}$  bytes, bloques de  $2^4$  bytes (en MIPS, word =  $2^2$  bytes)

$$\Rightarrow \# \text{ bloques} = \frac{2^{12}}{2^4} = 2^8 \Rightarrow \# \text{ sets} = \frac{2^8}{\text{asociatividad}} = 2^7.$$

TAG	sets	w	b
15	7	2	2

b) Sumamos de 4 en 4 (palabras de 4 bytes):

2B0, 2B4, 2B8, 2BC, 2C0, 2C4, 752A4,

752A8, 752AC, 752B0, 752B4, 752B8,

752BC, 79AE0, 79AE4, 79AE8, 79AEC.

Lo subrayado se repite 18 veces más (en total, son 19).

c) Evolución de la zona de control y datos:



Dirección	TAG	Set	1 <sup>ra</sup> it. A/F	2da it. A/F
280	0h	2Bh	F	Añetón TODOS
284	"	"	A	
288	"	"	A	
28C	"	"	A	
2C0	0h	2Ch	F	
2C4	"	"	A	
752A4	EAh	2Ah	F	
752A8	"	"	A	
752AC	"	"	A	
752B0	EAh	2Bh	F	
752B4	"	"	A	
752B8	"	"	A	
752BC	"	"	A	
79AE0	F3h	2Eh	F	
79AE4	"	"	A	
79AE8	"	"	A	
79AEC	"	"	A	

Todos fallan compulsory.

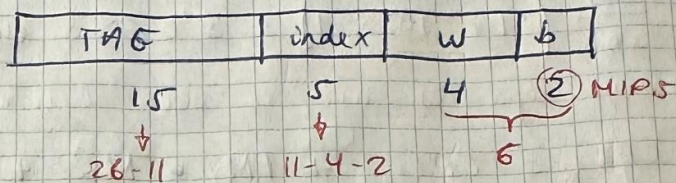
$$IC = 3 + 14 \cdot 19 = 269.$$

$$\text{Miss Rate} \pm = \frac{5}{269} = 0.01858 = 1.858\%.$$

SET	V	TAG	word 3	word 2	word 1	word 0
2Bh	1	0h	i4	i3	i2	i1
	1	EAh	i13	i12	i11	i10
2Ch	1	0h			i6	i5
	0					
2Ah	1	EAh	i9	i8	i7	—
	0					
2Eh	1	F3h	i17	i16	i15	i14
	0					



B) a) Tamaño:  $2^4$  bytes, directa, bloques de  $2^6$  bytes.



b) Secuencia de direcciones:

4A04, CA38, y esto se hace en total 19 veces. (hay que mirar los loads y stores).

Dirección	TAG	index	A/F 1 <sup>ra</sup> it.	A/F 2 <sup>da</sup> it.
4A04	0009h	08h	F	F
CA38	0019h	08h	F	F

Siempre se solapan, van al mismo conjunto y es asignación directa.

words							
TAG	v	15	14	13	12	...	0
08h 0019h	1	A	0	A	A	...	A

Nota: block offset de CA38 es 11 1000, luego va a la palabra 1110b = 14d

c)  $f = 1.2 \text{ GHz} \Rightarrow T_c = \frac{1}{1.2 \cdot 10^9} = 0.83 \text{ ns}$ .

$PM_I = 13, PM_D = 15$ .

$MR_I = 0.0186, MR_D = 1$  (estos dos se sacan de otros apartados).

Referencias totales =  $269 + 38 = 307$ .

PM = penalty miss.

MR = miss rate.



$$ANAT = \frac{269}{307} \left( 1 + 13 \cdot 0'0186 \right) +$$

$$+ \frac{38}{307} \left( 1 + 15 \cdot 1 \right) = 3'0685 \text{ cc} = 2'557 \text{ ns.}$$

Que yo sepa es así, si veis algún fallo avisadme :)

$$b) CPI_y = CPI_{base} + PM_I \cdot MR_I + \frac{38}{269} PM_D \cdot I_D =$$

$$= 2'1 + 13 \cdot 0'0186 + \frac{38}{269} \cdot 15 = 4'46.$$

$$c) T_g = \frac{IC \cdot CPI_y}{f} = \frac{269 \cdot 4'46}{1'2 \cdot 10^9} = 999'97 \text{ ns}$$

D) No lo hemos dado, pero yo entiendo que al menos debe ocupar los bits que cubre la caché, que son 11, luego al menos debe ser de  $2^{11}$  bytes = 2Kbytes.

En el direccionamiento, esto es, al menos 26 - TAG.

E) Bloque de  $2^3$  palabras, como es MIPS

⇒ bloques de  $2^5$  bytes.

Bus de  $2^2$  bytes = 1 palabra.

Como hay  $2^3$  módulos, todos se pueden ir calculando a la vez, pero las palabras se devuelven de una en una, pues no cabe más de 1 palabra en el bus.

Esto es: 1 de dar la dirección, 20 de buscar el dato y 1+8 de devolver las palabras = 29.