

Problemas Tema 3

Problema 2. Repaso Cache

Tipo	@	Bloque de memoria	Conjunto de MC	Acierto/Fallo	Lectura MP			Escritura MP		
					Sí/No	@	tamaño	Sí/No	@	tamaño
R byte	8890	889	1	Fallo	Sí	8890	16	No	-	-
W word	EC51	EC5	1	Acierto	No	-	-	Sí	EC51	2
W byte	EC62	EC6	2	Acierto	No	-	-	Sí	EC62	1
W word	23D3	23D	1	Fallo	No	-	-	Sí	23D3	2
W byte	ABA4	ABA	2	Fallo	No	-	-	Sí	ABA4	1
R word	ABA5	ABA	2	Fallo	Sí	ABA5	16	No	-	-
R byte	23D6	23D	1	Fallo	Sí	23D6	16	No	-	-
W word	EC57	EC5	1	Acierto	No	-	-	Sí	EC57	2
R byte	EC68	EC6	2	Acierto	No	-	-	No	-	-
R word	8899	889	1	Fallo	Sí	8890	16	No	-	-

conjunto 0		conjunto 1		conjunto 2		conjunto 3	
EC8	1	EC5	0	EC6	1	EC7	1
AB4	0	889	1	ABA	0	-	0

@ de 16 bits

16 bytes / bloque

Cache 2-asociativa

LRU

8 líneas (bloques)

Write through + Write no allocate

8 bloques

2 bloques / conj.

$\frac{8}{2} = 4$ conjuntos

"

$2^2 \rightarrow \# \text{conj. MC}$

16 bytes / bloque

"

$2^4 \rightarrow \# \text{byte}$

TAG

conj. MC

byte

10

2

4

1) Miss de L

Conjunto 1	
EC5	0
889	1

2) Hit de E

Conjunto 1	
EC5	1
889	0

3) Hit de E

Conjunto 2	
EC6	1
AB2	0

4) Miss de E

5) Miss de E

6) Miss de L

Conjunto 2	
EC6	0
ABA	1

7) Miss de L

Conjunto 1	
EC5	0
23D	1

8) Hit de E

Conjunto 1	
EC5	1
23D	0

9) Hit de L

Conjunto 2	
EC6	1
ABA	0

10) Miss de L

Conjunto 1	
EC5	0
889	1

Problema 4. Repaso Cache

a) Alternativa 1 (escritura inmediata sin asignación)

$$t_{ma} = 0.8 (0.9 * 10 + 0.1(10 + 100 + 10)) + 0.2 * 80 = 32.8 \text{ ns} \xRightarrow[1000 \text{ accesos}]{} 32800 \text{ ns}$$

Alternativa 2 (escritura retardada con asignación)

$$t_{ma} = 0.85 * 10 + 0.15(0.3333(2 * 100 + 2 * 10) + 0.6666(100 + 2 * 10)) = 31.5 \text{ ns}$$

$$\xRightarrow[1000 \text{ accesos}]{} 31500 \text{ ns}$$

b) La alternativa 1, ya que tiene mejor/mayor tasa de aciertos.

c) tamaño de un bloque > tamaño de una palabra.

Problema 5. Repaso Cache

a) $T_{maI} = t_{hit} + t_{axa miss} \times t_{penalització}$
 $= 1 + 0,04 \times 10$
 $= 1,4 \text{ cycles}$

b) $T_{maD} = 1 + 0,1 (0,2 \times 20 + 0,8 \times 15) = 2,6 \text{ cycles}$

c) $T_{ma} = \frac{1,4 \times 1 + 2,6 \times 0,6}{1,6} = 1,85 \text{ cycles}$

d) $CPI = CPI_{ideal} + CPI_{mem}$
 $= CPI_{ideal} + n_{ref} (T_{ma} - t_{hit})$
 $= 1,5 + 1,6 (1,85 - 1)$
 $= 2,86 \text{ cycles}$

$T_{exec} = N \times CPI \times T_c$
 $= 1 \times 2,86 \times 10$
 $= 28,6 \text{ ns}$

Problema 7. Repaso Memoria Virtual

1 página = 8192 bytes

		LRU del TLB	
		+ recently used	- recently used
iteración 0 (i = 0)	(a) M[0], página 0, MISS	0	
	(b) M[8192], página 1, MISS + HIT	1, 0	
	(c) M[16384], página 2, MISS	2, 1, 0	
iteración 1 (i = 512)	(a) M[2048], página 0, HIT	0, 2, 1	
	(b) M[10240], página 1, HIT + HIT	1, 0, 2	
	(c) M[18432], página 2, HIT	2, 1, 0	
iteración 2 (i = 1024)	(a) M[4096], página 0, HIT	0, 2, 1	
	(b) M[12288], página 1, HIT + HIT	1, 0, 2	
	(c) M[20480], página 2, HIT	2, 1, 0	

iteración 3 (i = 1536)	(a) M[6144], página 0, HIT	0, 2, 1
	(b) M[14336], página 1, HIT + HIT	1, 0, 2
	(c) M[22528], página 2, HIT	2, 1, 0
iteración 4 (i = 2048)	(a) M[8192], página 1, HIT	1, 2, 0
	(b) M[16384], página 2, HIT + HIT	2, 1, 0
	(c) M[24576], página 3, MISS	3, 2, 1, 0

• for (int i = 0; i < 512000; i += 512) { ... } \Rightarrow 1000 iteraciones, en cada una hacemos 4 accesos a memoria (1 movl + 2 addl + 1 movl) \Rightarrow 4000 accesos

• iteración 0: 3 MISS y 1 HIT ; de la iteración 1 a la 4: 1 MISS y 15 HIT
(cada 4 iteraciones se repetirá el patrón)

• Fallos de TLB = $3 + \lfloor 999 / 4 \rfloor * 1 = 252 \Rightarrow$ Aciertos de TLB = 3748

d)



• Fallos de TLB = $500 * 3 = 1500$; Aciertos de TLB = $500 * 4 + 500 = 2500$

Problema 7. Repaso Memoria Virtual

Dado el siguiente código escrito en ensamblador x86:

```

movl $0, %ebx
movl $0, %esi
for: cmpl $512*1000, %esi
jge end
(a) movl (%ebx, %esi, 4), %eax
(b) addl %eax, 8*1024(%ebx, %esi, 4)
(c) movl %eax, 16*1024(%ebx, %esi, 4)
    addl $512, %esi
    jmp for
end:
  
```

Suponiendo que la memoria utiliza páginas de tamaño 8KB y que utilizamos un TLB de 4 entradas (reemplazo LRU), responde a las siguientes preguntas:

a) Para cada uno de los accesos (etiquetas a, b, c), indica a qué página de la memoria virtual se accede en cada una de las 17 primeras iteraciones.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
a	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3	4
b	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4	5
c	2	2	2	2	3	3	3	3	4	4	4	4	5	5	5	5	6

b) Calcula la cantidad de aciertos de TLB, en todo el bucle: 3748

c) Calcula la cantidad de fallos de TLB, en todo el bucle: 252

Suponiendo que la memoria utiliza **páginas de tamaño 4KB** y que utilizamos un **TLB de 4 entradas (reemplazo LRU)**, responde a las siguientes preguntas:

- d) Para cada uno de los accesos (etiquetas a, b, c), indica a qué página de la memoria virtual se accede en cada una de las 17 primeras iteraciones.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
a	0	0	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8
b	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	9	10
c	4	4	5	5	6	6	7	7	8	8	9	9	10	10	11	11	12

- e) **Calcula la cantidad de aciertos de TLB**, en todo el bucle: **2500**
f) **Calcula la cantidad de fallos de TLB**, en todo el bucle: **1500**

Problema 8. Repaso Memoria Virtual

dirección lógica (hexa)	VPN (hexa)	desplazamiento (hexa)	dirección física (hexa)	fallo de página	lectura disco	escritura disco	Página reemplazada	
							VPN	PPN
escritura	F458	7	1458	7458	X	X		
escritura	8666	4	0666	4666				
lectura	1BBF	0	1BBF	1BBF	X	X	2	0
escritura	5C44	2	1C44	3C44	X	X	3	1
lectura	6600	3	0600	6600	X	X	7	3
lectura	4000	2	0000	2000				

Contenido final de la Tabla de Páginas

VPN	P	M	PPN
0	1	0	0
1	0		
2	1	1	1
3	1	0	3
4	1	1	2
5	0		
6	0		
7	0		

Contenido final de Memoria

página física	página lógica
0	0
1	2
2	4
3	3

Problema 9. Caches petites i simples

3.9 a)

	73	55	43	45	73	45	13	43	73	55	45	73	15	43
Directa					X							X		
2 - associativa			X	X					X	X	X		X	
Directa + VC			X	X			X	X		X	X		X	

Cache directa

8 blocs $\Rightarrow \frac{\# \text{ línia MC}}{3 \text{ bits}}$

- 73 (octal) = 3B (hexa) = $0011 \frac{1011}{3}$
- 55 (octal) = 2D (hexa) = $0010 \frac{1101}{5}$
- 43 (octal) = 23 (hexa) = $0010 \frac{0011}{3}$
- 45 (octal) = 0100 $\frac{0101}{5}$
- 13 (octal) = 0000 $\frac{1011}{3}$
- 15 (octal) = 0000 $\frac{1101}{5}$

2 - associativa

4 conjunts $\Rightarrow \frac{\# \text{ conj. MC}}{2 \text{ bits}}$

- 1) 73 \Rightarrow Miss, conjunt 3 (vía 0)
- 2) 55 \Rightarrow Miss, conjunt 1 (vía 0)
- 3) 43 \Rightarrow Miss, conjunt 3 (vía 1)
- 4) 45 \Rightarrow Miss, conjunt 1 (vía 1)
- 5) 73 \Rightarrow Hit
- 6) 45 \Rightarrow Hit
- 7) 13 \Rightarrow Miss, conjunt 3 (vía 1)
- 8) 43 \Rightarrow Miss, conjunt 3 (vía 0)
- 9) 73 \Rightarrow Miss, conjunt 3 (vía 1)
- 10) 55 \Rightarrow Hit
- 11) 45 \Rightarrow Hit
- 12) 73 \Rightarrow Hit
- 13) 15 \Rightarrow Miss, conjunt 1 (vía 0)
- 14) 43 \Rightarrow Hit

Cache directa + VC

- 1) 73 \Rightarrow Miss, 3
- 2) 55 \Rightarrow Miss, 5
- 3) 43 \Rightarrow Miss, 3 \Rightarrow $\frac{VC}{\text{Bloc 0 : 73}}$
- 4) 45 \Rightarrow Miss, 5 \Rightarrow $\frac{VC}{\text{Bloc 1 : 55}}$

$$5) \quad 73 \Rightarrow \begin{array}{l} \text{Miss MC, 3} \\ \text{Hit VC, 0} \end{array} \Rightarrow \begin{array}{l} \text{MC} \quad 3 : 73 \\ \text{VC} \quad 0 : 43 \end{array}$$

$$6) \quad 45 \Rightarrow \text{Hit, 5} \quad 7) \quad 13 \Rightarrow \begin{array}{l} \text{Miss, 3} \\ \text{FIFO} \end{array} \Rightarrow \begin{array}{l} \text{VC} \quad 1 : \cancel{55} \\ 73 \end{array}$$

(Fuera la que lleva + tiempo)

$$8) \quad 43 \Rightarrow \begin{array}{l} \text{Miss MC, 3} \\ \text{Hit VC, 0} \end{array} \Rightarrow \begin{array}{l} \text{MC} \quad 3 : 43 \\ \text{VC} \quad 0 : 13 \end{array}$$

$$9) \quad 73 \Rightarrow \begin{array}{l} \text{Miss MC, 3} \\ \text{Hit VC, 1} \end{array} \Rightarrow \begin{array}{l} \text{MC} \quad 3 : 73 \\ \text{VC} \quad 1 : 43 \end{array}$$

$$10) \quad 55 \Rightarrow \begin{array}{l} \text{Miss, 5} \\ \text{FIFO} \end{array} \Rightarrow \begin{array}{l} \text{VC} \quad 0 : \cancel{13} \quad 45 \end{array}$$

$$11) \quad 45 \Rightarrow \begin{array}{l} \text{Miss MC, 5} \\ \text{Hit VC, 0} \end{array} \Rightarrow \begin{array}{l} \text{MC} \quad 5 : 45 \\ \text{VC} \quad 0 : 55 \end{array}$$

$$12) \quad 73 \Rightarrow \text{Hit, 3} \quad 13) \quad 15 \Rightarrow \begin{array}{l} \text{Miss, 5} \\ \text{FIFO} \end{array} \Rightarrow \begin{array}{l} \text{VC} \quad 1 : \cancel{43} \\ 45 \end{array}$$

$$14) \quad 43 \Rightarrow \begin{array}{l} \text{Miss, 3} \\ \text{FIFO} \end{array} \Rightarrow \begin{array}{l} \text{VC} \quad 0 : \cancel{55} \quad 73 \end{array}$$

b) No, perquè la línia que porta més temps a la VC és la que porta més temps sense ser utilitzada (LRU). ✓

$$c) \quad \text{CPI ideal} = \frac{12 \cdot 10^9 \text{ cicles}}{10 \cdot 10^9 \text{ instr.}} = 1.2 \text{ c/i} \quad \checkmark$$

$$d) \quad \text{nr} = \frac{3 \cdot 10^9 \text{ accesos}}{10 \cdot 10^9 \text{ instr.}} = 0.3 \text{ accesos/instr.} \quad \checkmark$$

$$e) \quad \begin{aligned} \text{Cicles} &= N \times \text{CPI} = N \times (\text{CPI ideal} + \text{CPI mem}) \\ &= 10 \cdot 10^9 (1.2 + 0.3 \times 0.1 \times 10) \quad \checkmark \\ &= 1.5 \cdot 10^{10} \text{ cicles} \end{aligned}$$

$$f) \quad \text{Texe} = \text{cicles} \times T_c = 1.5 \cdot 10^{10} \times 10 \cdot 10^{-9} = 150 \text{ s} \quad \checkmark$$

g) Pel temps de cicle / temps d'accés ~

- 3.9
- h) $Cicles = 10 \cdot 10^9 (1.2 + 0.3 \times 0.05 \times 9) = 1.335 \cdot 10^{10} \text{ cicles}$ ✓
- i) $T_{exe} = 1.335 \cdot 10^{10} \times 12 \cdot 10^{-9} = 160.2 \text{ s}$ ✓
- j) $Cicles = 10 \cdot 10^9 (1.2 + 0.3 \times 0.06 \times 10) = 1.38 \cdot 10^{10} \text{ cicles}$ ✓
- k) $T_{exe} = 1.38 \cdot 10^{10} \times 11 \cdot 10^{-9} = 151.8 \text{ s}$ ✓
- l) Perquè els accessos que s'han de fer a la victim cache tenen una penalització addicional d'un cicle
- m) $P(\text{Fallar a MC}) = 0.1$
 $P(\text{Fallar a MC} \wedge \text{Fallar a VC}) = 0.06$
- Diagrama de probabilitats:
- ```

graph LR
 FMC -- 0.1 --> FVC
 FMC -- 0.9 --> FVNC
 FVC -- 0.6 --> FVC
 FVC -- 0.4 --> FVNC
 FVNC -- 0.9 --> FVNC
 FVNC -- 0.1 --> FVC

```
- $P(FVC | FMC) = 0.6 \Rightarrow P(FVNC | FMC) = 0.4$
- $P(FMC \wedge FVC) = P(FMC) \times P(FVC | FMC)$   
 $0.06 = 0.1 \times P(FVC | FMC)$
- $P(FVNC | FMC) = 0.4 \Rightarrow P(FVC | FMC) = 0.6$
- $P(\text{Fallar a MC} \wedge \text{Encertar a VC}) = 0.1 \times 0.4 = 0.04$  ✓
- n)  $Cicles = 10 \cdot 10^9 (1.2 + 0.3 (0.06 \times 11 + 0.04 \times 1))$   
 $= 1.41 \cdot 10^{10} \text{ cicles}$  ✓
- o)  $T_{exe} = 1.41 \cdot 10^{10} \times 10 \cdot 10^{-9} = 141 \text{ s}$  ✓

## Problema 10. Predicción de vía

### Problema 10. Predicción de vía

- a) **Calculad la potencia media dinámica** (debida a conmutación), la potencia media estática (debida a fugas) y la **potencia media total**.

$$P_{conm} = CV^2F = 5 \times 10^{-9} \text{ F} \cdot (1.2 \text{ V})^2 \cdot 2 \times 10^9 \text{ Hz} = 14.4 \text{ W}$$

$$P_{fuga} = IV = 3 \text{ A} \cdot 1.2 \text{ V} = 3.6 \text{ W}$$

$$P_{total} = 14.4 \text{ W} + 3.6 \text{ W} = 18 \text{ W}$$

- b) **Calculad el numero de conjuntos, el de bloques de cache, el de vías y el de bloques por vía.**

$$2^{17} \text{ bytes} / 2^6 \text{ bytes/bloque} = 2^{11} \text{ bloques} = 2048 \text{ bloques}$$

$$2048 \text{ bloques} / 2 \text{ bloques/cjto} = 1024 \text{ cjtos}$$

2-asociativa  $\rightarrow$  2 vías

1024 cjtos  $\rightarrow$  1024 bloques/vía



- c) **Dibujad** una dirección indicando claramente los campos usados para seleccionar el byte dentro del bloque, seleccionar el conjunto de la cache y los bits usados como etiqueta.

| TAG (32) | Cjto (10) | byte (6) |

- d) **Calculad** el tamaño en bits de la memoria de datos y el de la memoria de etiquetas de una vía.

1 vía -> 1024 bloques

M datos = 1024 bloq \* 64 bytes/bloq \* 8bits/byte = 524288 bits

M etiq = 1024 bloq \* 32 bits/bloq = 32768 bits

- e) **Calculad** la potencia media estática (debida a fugas) de la cache. bits totales = 557056 bits

I fuga = 557056 bits \*  $3 \times 10^{-6}$  A/bit = 1,671 A

P fuga = I \* V \* #vías = 1,671 A \* 1,2 V \* 2 vías = 4 W

- f) **Calculad** los MFLOPS de la implementación paralela.

Mflops =  $2 \times 10^9$  flop / 5 s \*  $10^{-6}$  MFLOPS/flop = 400 MFLOPS

- g) **Calculad** el CPI de la implementación paralela y el CPI que obtendríamos con una memoria ideal (CPI<sub>ideal</sub>) en donde todos los accesos tardan 1 ciclo.

Ciclos totales = 5 s \*  $2 \times 10^9$  Hz =  $10 \times 10^9$  ciclos CPI

paralela =  $10 \times 10^9$  ciclos /  $4 \times 10^9$  instr = 2,5 c/i

Cilos perdidos mem = 0,1 fallos/acceso \*  $10^9$  accesos \* 20 ciclos/fallo =  $2 \times 10^9$  ciclos

Ciclos ideal =  $10 \times 10^9$  ciclos -  $2 \times 10^9$  ciclos =  $8 \times 10^9$  ciclos

CPI ideal =  $8 \times 10^9$  ciclos /  $4 \times 10^9$  instr = 2 c/i

- h) **Calculad** la energía dinámica consumida por un acceso a la cache. Para simplificar asumiremos que todos los accesos consumen lo mismo sean acierto o fallo, la energía extra consumida en acceder a memoria principal en caso de fallo está fuera de los objetivos de este problema.

1 acceso cache -> 2 vías etiquetas + 2 vías datos

Energía = 2 \* 5 nJ + 2 \* 25 nJ = 60 nJ

- i) **Calculad** la potencia (dinámica) media consumida en acceder a la cache

$10^9$  accesos / 5 segundos =  $0,2 \times 10^9$  accesos/s

Potencia = energía / t =  $0,2 \times 10^9$  accesos/s \*  $60 \times 10^{-9}$  J = 12 J/s = 12 W

- j) **Calculad** la potencia media total (estática+dinámica) consumida por el sistema CPU-cache.

P<sub>total</sub> = P<sub>CPU</sub> + P<sub>cache\_fugas</sub> + P<sub>cache\_conmut</sub> = 18 W + 4 W + 12 W = 34 W

- k) **Calculad** la energía total consumida para ejecutar el benchmark y la eficiencia en MFLOPS/Watt.

E = P \* t = 34 W \* 5 s = 170 J

Eficiencia = 400 MFLOPS / 34 W = 11,76 MFLOPS/W

- l) **Calculad** el tiempo de ejecución y los MFLOPS de la implementación serie.

Ciclos = ciclos ideal + ciclos perdidos fallos + ciclos perdidos aciertos

=  $10 \times 10^9$  ciclos + 0,9 aciertos/acceso \*  $10^9$  accesos \* 1 ciclos/acierto =  $10,9 \times 10^9$  ciclos

Texe =  $10,9 \times 10^9$  ciclos /  $2 \times 10^9$  Hz = 5,45 s

Mflops =  $2 \times 10^9$  flop / 5,45 s \*  $10^{-6}$  MFLOPS/flop = 367 MFLOPS

- m) **Calculad** la energía consumida por un acceso a la cache.

1 acceso -> 2 vías etiquetas + 1 vía datos

E = 2 \* 5 nJ + 25 nJ = 35 nJ (en los fallos nos podríamos ahorrar el acceso a datos, pero lo damos por bueno)

- n) **Calculad** la potencia (dinámica) media consumida en acceder a la cache

$10^9$  accesos / 5,45 segundos =  $0,183 \times 10^9$  accesos/s

Potencia = energía / t =  $0,183 \times 10^9$  accesos/s \*  $35 \times 10^{-9}$  J = 6,42 J/s = 6,42 W

- o) **Calculad** la potencia media total consumida por el sistema CPU-cache.

P<sub>total</sub> = P<sub>CPU</sub> + P<sub>cache\_fugas</sub> + P<sub>cache\_conmut</sub> = 18 W + 4 W + 6,42 W = 28,42 W

- p) **Calculad** la energía total consumida para ejecutar el benchmark y la eficiencia en MFLOPS/Watt.

E = P \* t = 28,42 W \* 5,45 s = 155 J

Eficiencia = 367 MFLOPS / 28,42 W = 12,91 MFLOPS/W

- q) ¿Puede darse al caso de que un acierto del predictor de vía sea fallo de cache? ¿porqué?

NO

- r) Calculad la potencia media estática (debida a fugas) del predictor y compárala con la de la cache (se calcula de la misma forma ya que se ha empleado el mismo tipo de memoria estática).

$I_{\text{fuga}} = 8192 \text{ bits} * 3 \times 10^{-6} \text{ A/bit} = 24,6 \text{ mA}$   
 $fuga = I * V = 24,6 \times 10^{-3} \text{ A} * 1,2 \text{ V} = 29,5 \text{ mW}$   
 (es mucho menor, miliwatios vs watios)

- s) Calculad el tiempo de ejecución y los MFLOPS de la implementación con **predictor de vía**.

Ciclos = ciclos ideal + ciclos perdidos fallos cache + ciclos perdidos fallo predictor  
 $= 10 \times 10^9 \text{ ciclos} + 0,2 \text{ aciertos/acceso} * 10^9 \text{ accesos} * 1 \text{ ciclos/acierto} = 10,2 * 10^9 \text{ ciclos}$   
 $T_{\text{exe}} = 10,2 * 10^9 \text{ ciclos} / 2 \times 10^9 \text{ Hz} = 5,1 \text{ s}$   
 $Mflops = 2 * 10^9 \text{ flop} / 5,1 \text{ s} * 10^{-6} \text{ MFLOPS/flop} = 392 \text{ MFLOPS}$

- t) Calculad la energía consumida por un acceso en que el predictor acierta y uno en que el predictor falla (tener en cuenta la energía consumida por el acceso al predictor). Calcular también la energía media consumida por acceso.

acierto predictor  $\rightarrow$  predictor + 1 vía etiquetas + 1 vía datos  
 fallo predictor  $\rightarrow$  predictor + 2 vías etiquetas + 2 vías datos  
 $E_{\text{acierto}} = 1 \text{ nJ} + 5 \text{ nJ} + 25 \text{ nJ} = 31 \text{ nJ}$   $E_{\text{fallo}} = 1 \text{ nJ} + 10 \text{ nJ} + 50 \text{ nJ} = 61 \text{ nJ}$   
 $E_{\text{media}} = 0,8 * 31 \text{ nJ} + 0,2 * 61 \text{ nJ} = 37 \text{ nJ}$

- u) Calculad la potencia (dinámica) media consumida en acceder a la cache  $10^9 \text{ accesos} / 5,1 \text{ segundos} = 0,196 \times 10^9 \text{ accesos/s}$

Potencia = energía / t =  $0,196 \times 10^9 \text{ accesos/s} * 37 \times 10^{-9} \text{ J} = 7,25 \text{ J/s} = 7,25 \text{ W}$

- v) Calculad la potencia media total consumida por el sistema CPU-cache (acuérdate de las fugas del predictor).  $P_{\text{total}} = P_{\text{CPU}} + P_{\text{cache\_fugas}} + P_{\text{pred\_fugas}} + P_{\text{cache\&pred\_conmut}}$

$= 18 \text{ W} + 4 \text{ W} + 0,03 \text{ W} + 7,25 \text{ W} = 29,28 \text{ W}$

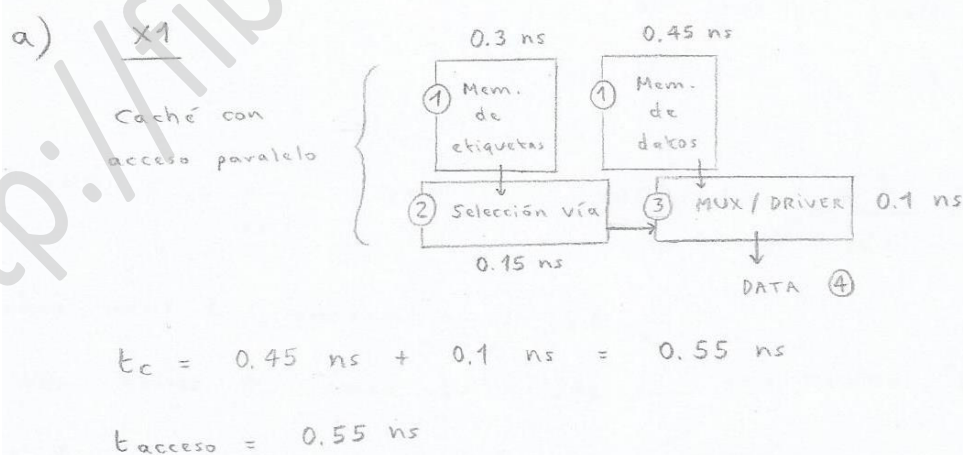
- w) Calculad la energía total consumida para ejecutar el **benchmark** y la eficiencia en MFLOPS/Watt.

$E = P * t = 29,28 \text{ W} * 5,1 \text{ s} = 149 \text{ J}$   
 Eficiencia =  $392 \text{ MFLOPS} / 29,28 \text{ W} = 13,39 \text{ MFLOPS/W}$

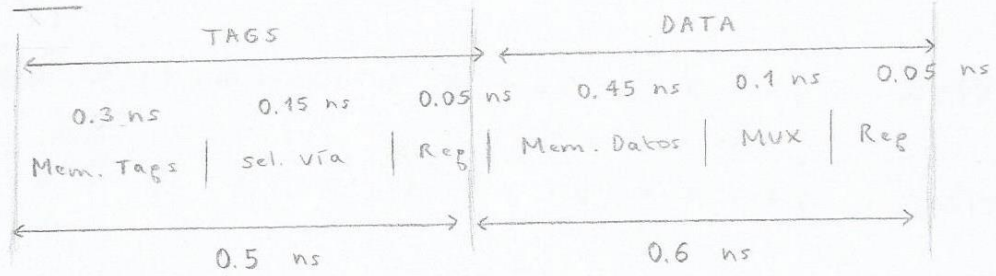
- x) Calculad la ganancia en eficiencia energética de la implementación serie sobre la paralela y la de predicción de vía sobre la serie.

serie/paralelo =  $12,91 / 11,76 = 1,098 = 9,8\%$   
 $P_{\text{via}}/\text{serie} = 13,39 / 12,91 = 1,037 = 3,7\%$

## Problema 11. Caches segmentadas



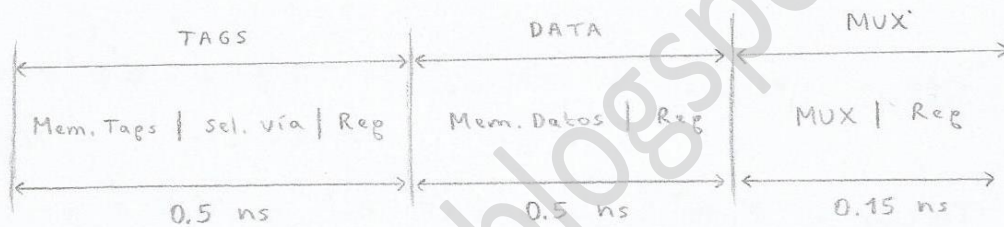
X2



$$t_c = 0.6 \text{ ns}$$

$$t_{\text{accesso}} = 0.6 \times 2 = 1.2 \text{ ns}$$

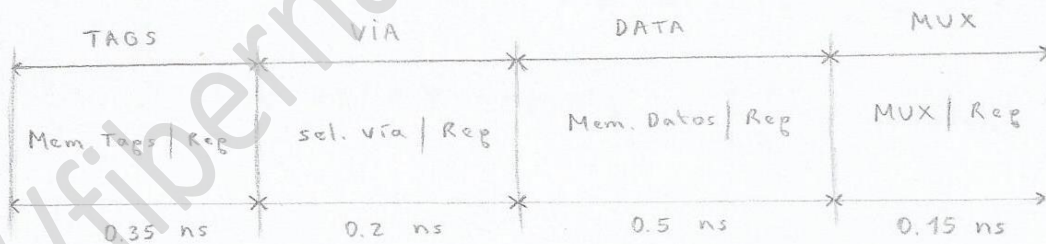
X3



$$t_c = 0.5 \text{ ns}$$

$$t_{\text{accesso}} = 0.5 \times 3 = 1.5 \text{ ns}$$

X4



$$t_c = 0.5 \text{ ns}$$

$$t_{\text{accesso}} = 0.5 \times 4 = 2 \text{ ns}$$



- b)
- X2 tiene el peor  $t_c$
  - X4 tiene el peor  $t_{acceso}$

c)

$$f_{x1} = \frac{1}{t_{cx1}} = 1.82 \text{ GHz} \quad f_{x3} = \frac{1}{t_{cx3}} = 2 \text{ GHz}$$

d)

$2 \cdot 10^9$  instrucciones

- $\rightarrow 1.2 \cdot 10^9$  aritméticas, 5 ciclos cada una (60%)
- $\rightarrow 400 \cdot 10^6$  de salto, 4 ciclos cada una (20%)
- $\rightarrow 400 \cdot 10^6$  de acceso a memoria, 4 ciclos + (20%)  
ciclos del acceso a la cache

$\rightarrow$  X1: 1 ciclo  
 $\rightarrow$  X3: 3 ciclos

$$CPI_{x1} = 0.6 \times 5 + 0.2 \times 4 + 0.2(4+1) = 4.8 \text{ c/i}$$

$$CPI_{x3} = 0.6 \times 5 + 0.2 \times 4 + 0.2(4+3) = 5.2 \text{ c/i}$$

e)

$$T_{exe x1} = 2 \cdot 10^9 \times 4.8 \times 0.55 \cdot 10^{-9} = 5.28 \text{ s}$$

$$T_{exe x3} = 2 \cdot 10^9 \times 5.2 \times 0.5 \cdot 10^{-9} = 5.2 \text{ s}$$

$$Speedup = \frac{5.28}{5.2} = 1.015 \text{ (1.5 \%)}$$

f)

$$CPI = CPI_{ideal} + CPI_{mem} = 4.8 + 0.2 \times 0.1 \times 60 = 6 \text{ c/i}$$

$$T_{exe x1} = 2 \cdot 10^9 \times 6 \times 0.55 \cdot 10^{-9} = 6.6 \text{ s}$$

$$CPI = 5.2 + 0.2 \times 0.1 \times 60 = 6.4 \text{ c/i}$$

$$T_{exe x3} = 2 \cdot 10^9 \times 6.4 \times 0.5 \cdot 10^{-9} = 6.4 \text{ s}$$

$$Speedup = \frac{6.6}{6.4} = 1.03125 \text{ (3.125 \%)}$$

## Problema 12. Cache no bloqueante.

### Problema 12. Cache no bloqueante.

- a) **Calculad el CPI de P en el procesador IDEAL ( $CPI_{IDEAL}$ )**

$$CPI_{IDEAL} = 5 \times 10^9 \text{ ciclos} / 2 \times 10^9 \text{ instrucciones} = 2,5 \text{ ciclos / instrucción}$$

- b) **Calculad el número medio de ciclos transcurridos entre 2 fallos.**

$$5 \times 10^9 \text{ ciclos} / 50 \times 10^6 \text{ fallos} = 100 \text{ ciclos entre fallos}$$

- c) **Calculad el CPI de P en el procesador B ( $CPI_B$ )**

$$CPI_B = 4 \text{ s} * 2 \times 10^9 \text{ Hz} / 2 \times 10^9 \text{ instrucciones} = 4 \text{ ciclos / instrucción}$$

- d) **Calculad el tiempo de penalización por fallo de cache ( $T_{pf}$ ) en ciclos.**

$$\text{ciclos}_B = \text{ciclos}_{IDEAL} + T_{pf} * \text{numero de fallos}$$

$$T_{pf} = (8 \times 10^9 - 5 \times 10^9) \text{ ciclos} / 50 \times 10^6 \text{ fallos} = 60 \text{ ciclos / fallo}$$

- e) **Calculad la probabilidad de que se produzca un segundo fallo durante el servicio de un fallo anterior**

probabilidad de fallar en un ciclo es  $p = 1/100$  (inversa del tiempo medio entre fallos)

probabilidad de tener un fallo en 60 ciclos (intervalo de servicio de F1) es  $1 - \text{probabilidad de no fallar en ningún ciclo}$  (repetimos un proceso independiente 60 veces con probabilidad  $p$ )

$$P(\text{fallo en el intervalo}) = 1 - (1 - p)^{60} = 1 - (1 - 1/100)^{60} = 0,453$$

- f) **¿Puede producirse un tercer fallo?**

No, a partir del segundo deja de ejecutar instrucciones.

Si se produce un segundo fallo durante el intervalo de servicio de un fallo anterior, este se puede producir en cualquiera de los ciclos que dura el servicio, con la misma probabilidad. Es decir, se trata de una distribución de probabilidad **uniforme discreta** (dado de 60 caras).

- g) **Calculad cuantos ciclos se pierden como máximo y como mínimo en función de en que ciclo del intervalo se produce el segundo fallo.**

en el 1º 59 ciclos, en el último 0 ciclos

- h) **Calculad el número medio de ciclos perdidos debido al segundo fallo (repasa cual es el valor medio esperado en una distribución de probabilidad uniforme discreta, o sea un dado numerado de 0 a 59)**

$$\text{Número medio de ciclos perdidos} = (0 + 59) / 2 = 29,5 \text{ ciclos/fallo}$$

- i) **Calculad el número de ciclos necesario para ejecutar P en el procesador N (con cache no bloqueante)**

$$\text{ciclos}_N = \text{ciclos}_{IDEAL} + \text{ciclos}_{MEM} \text{ (ciclos esperando a memoria)}$$

$$\text{ciclos}_{MEM} = \# \text{fallos} * \text{probabilidad de 2º fallo} * \text{ciclos perdidos 2º fallo}$$

$$\text{ciclos}_N = 5 \times 10^9 \text{ ciclos} + 50 \times 10^6 \text{ fallos} * 0,453 * 29,5 \text{ ciclos/fallo} = 5,67 \times 10^9 \text{ ciclos}$$

- j) **Calculad la ganancia (speedup) del procesador N sobre el B**  
 $\text{Guany} = 4 \text{ s} / (5,67 \times 10^9 \text{ ciclos} / 1,9 \times 10^9 \text{ Hz})$   
 $= 1,34$

El procesador con cache no bloqueante es un 34% más rápido que el que tiene cache bloqueante a pesar de funcionar a menor frecuencia



### Problema 13. Continuación anticipada, Transferencia en desorden

$$a) \quad t_c = \frac{1}{f} = \frac{1}{2 \cdot 10^9} = 5 \cdot 10^{-10} \text{ s}$$

$$T_{exe} = \frac{\text{ciclos}}{f} \Rightarrow \text{ciclos} = 2 \times 2 \cdot 10^9 = 4 \cdot 10^9 \text{ ciclos}$$

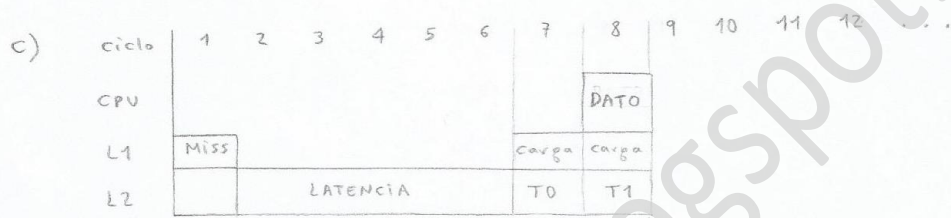
$$b) \quad \text{ciclos penalización} = 5 + 4 + 1 = 10 \text{ ciclos}$$

(latencia) (bus) (lectura L1 + envío dato)

$$T_{exe} = t_c \times \text{ciclos} = t_c (\text{ciclos ideal} + n_r \times \text{taxa miss} \times \text{ciclos penal.})$$

$$= 5 \cdot 10^{-10} (4 \cdot 10^9 + 10^9 \times 0.2 \times 10)$$

$$= 3 \text{ s}$$

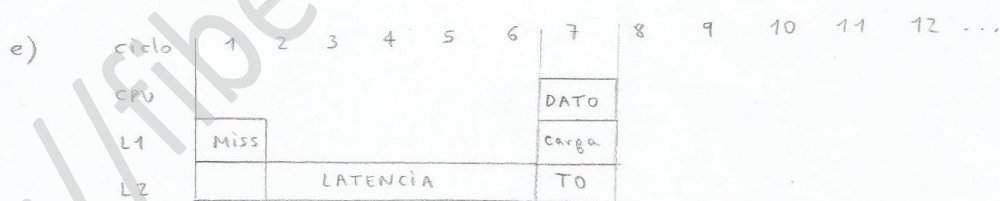


2 ciclos porque el bus es de 8 bytes y queremos el byte 12 del bloque

$$d) \quad \text{Ciclos penalización} = 5 + 0.7 \times 1 + 0.1 \times 2 + 0.1 \times 3 + 0.1 \times 4 = 6.6 \text{ ciclos} \approx 7 \text{ ciclos}$$

$$T_{exe} = t_c \times \text{ciclos} = 5 \cdot 10^{-10} (4 \cdot 10^9 + 10^9 \times 0.2 \times 7)$$

$$= 2.7 \text{ s}$$



$$f) \quad \text{Ciclos penalización} = 5 + 1 = 6 \text{ ciclos}$$

$$T_{exe} = 5 \cdot 10^{-10} (4 \cdot 10^9 + 10^9 \times 0.2 \times 6) = 2.6 \text{ s}$$

$$g) \quad \text{Speedup} = \frac{3}{2.7} = 1.1 \quad (11.11 \%)$$

(early restart)

$$\text{Speedup} = \frac{3}{2.6} = 1.1538 \quad (15.38 \%)$$

(transferencia en desorden)

# Problema 14. Prefetch

3.14

@ de 16 bits

3 - asociativa

LRU

12 bloques

64 bytes / bloque

CB + WA

12 bloques = 2<sup>2</sup> # conj. MC conjuntos

3 bloques / conj.

2<sup>6</sup> bytes / bloque  
# byte

TAG # conj. MC # byte  
8 2 6

| a) | tipo | @    | bloque de memoria | TAG | conjunto MC | A/F | bloque reemplazado | bytes escritura MP | bytes lectura MP |
|----|------|------|-------------------|-----|-------------|-----|--------------------|--------------------|------------------|
|    | L    | B128 | 2C4               | B1  | 0           | F   | 2B0 (AC)           | 0                  | 64               |
|    | L    | B145 | 2C5               | B1  | 1           | F   | 2B1 (AC)           | 0                  | 64               |
|    | L    | B1AF | 2C6               | B1  | 2           | F   | 2B2 (AC)           | 64                 | 64               |
|    | L    | B1C4 | 2C7               | B1  | 3           | F   | 2B3 (AC)           | 64                 | 64               |
|    | E    | 4387 | 10E               | 43  | 2           | A   | -                  | 0                  | 0                |
|    | L    | 1108 | 044               | 11  | 0           | F   | 10C (43)           | 64                 | 64               |
|    | E    | 1199 | 046               | 11  | 2           | F   | 10E (13)           | 0                  | 64               |
|    | L    | 11AA | 046               | 11  | 2           | A   | -                  | 0                  | 0                |

1) LRU conjunto 0

|    |   |
|----|---|
| B1 | 0 |
| 13 | 1 |
| 43 | 1 |

2) LRU conjunto 1

|    |   |
|----|---|
| B1 | 0 |
| 13 | 1 |
| 43 | 1 |

3) LRU conjunto 2

|    |   |
|----|---|
| B1 | 0 |
| 13 | 0 |
| 43 | 0 |

4) LRU conjunto 3

|    |   |
|----|---|
| B1 | 0 |
| 13 | 0 |
| 43 | 0 |

5) Hit E ⇒ DB = 1

LRU conjunto 2

|    |   |
|----|---|
| 43 | 1 |
| B1 | 0 |
| 13 | 0 |

6) LRU conjunto 0

|    |   |
|----|---|
| 11 | 0 |
| B1 | 0 |
| 13 | 1 |

7) LRU conjunto 2

|    |            |
|----|------------|
| 11 | ① → Miss E |
| B1 | 0          |
| 13 | 0          |

A la cache anterior le añadimos un **buffer de prefetch** de una entrada. En este **buffer** se hace prebúsqueda hardware del bloque **i+1** cuando se accede (tanto en acierto como en fallo) al bloque **i**, siempre que el **i+1** no esté ya en la cache o en el **buffer**. En este último caso, no se realiza **prefetch**.

- b) **Rellenad** la siguiente tabla (mismas referencias que la anterior) indicando, para cada referencia, el número de bloque de memoria que le corresponde, la etiqueta (TAG), a qué conjunto de MC va a parar, si se produce acierto o fallo en la cache (A/F), el número de bytes leídos de MP (si se lee de MP), el número de bytes escritos en MP (si se escribe en MP), el bloque de MP que se encuentra en el **buffer** (si procede), si se produce acierto o fallo (A/F) en el **buffer** y el bloque que se prebusca de MP (si procede).

| tipo | dirección (hex) | bloque de memoria (hex) | TAG (hex) | conjunto MC | Cache ¿acierto o fallo? | bytes escritura MP | bytes lectura MP | bloque actual buffer | Buffer ¿acierto o fallo? | bloque prefetch buffer |
|------|-----------------|-------------------------|-----------|-------------|-------------------------|--------------------|------------------|----------------------|--------------------------|------------------------|
| LECT | B12B            | 2C4                     | B1        | 0           | F                       | 0                  | 128              | -                    | F                        | 2C5                    |
| LECT | B145            | 2C5                     | B1        | 1           | F                       | 0                  | 64               | 2C5                  | A                        | 2C6                    |
| LECT | B1AF            | 2C6                     | B1        | 2           | F                       | 64                 | 64               | 2C6                  | A                        | 2C7                    |
| LECT | B1C4            | 2C7                     | B1        | 3           | F                       | 64                 | 64               | 2C7                  | A                        | 2C8                    |
| ESCR | 4387            | 10E                     | 43        | 2           | A                       | 0                  | 0                | 2C8                  | -                        | -                      |
| LECT | 1108            | 044                     | 11        | 0           | F                       | 64                 | 128              | 2C8                  | F                        | 045                    |
| ESCR | 1199            | 046                     | 11        | 2           | F                       | 0                  | 128              | 045                  | F                        | 047                    |
| LECT | 11AA            | 046                     | 11        | 2           | A                       | 0                  | 0                | 047                  | -                        | -                      |

→ 10E en cache ya  
→ 047 en el buffer ya

### Problema 17. DRAM

3.17

a)

Se transmiten los 8 paquetes de datos (8 porque queremos leer 64 bytes y tenemos 8 chips de 1 byte)

→ 8 bytes/ciclo × 8 ciclos = 64 bytes

b)

$$T_c = \frac{1}{200 \cdot 10^6} = 5 \text{ ns}$$

× 17 ciclos = 85 ns

c)

Ancho de banda teórico máximo =  $\frac{64 \text{ bytes}}{8 \text{ ciclos}} \times 200 \cdot 10^6 \text{ Hz} = 1.6 \cdot 10^9 \text{ bytes/s} = 1.6 \text{ GB/s}$

(de bytes a GB)

d)

Ancho de banda real =  $\frac{64 \text{ bytes}}{17 \text{ ciclos}} \times 200 \cdot 10^6 \text{ ciclos/s} = 752.94 \cdot 10^6 \text{ bytes/s} = 752.94 \text{ MB/s}$

e)  $V = 1.5 \text{ V}$

Memoria inactiva  $\Rightarrow I_{\text{defuga}} = 200 \text{ mA}$

Desde ACT a PRE,  $+ 100 \text{ mA}$

Transferencia de datos,  $+ 500 \text{ mA}$

$$\frac{7+2}{25} \times 300 \cdot 10^{-3} \text{ A} \times 1.5 \text{ V} = 0.162 \text{ W}$$

$$\frac{8}{25} \times 800 \cdot 10^{-3} \text{ A} \times 1.5 \text{ V} = 0.384 \text{ W}$$

$$\frac{8}{25} \times 200 \cdot 10^{-3} \text{ A} \times 1.5 \text{ V} = 0.096 \text{ W}$$

del ciclo  
18 al 25

$$P = \frac{E}{t} \Rightarrow E = P \times t = 0.642 \text{ W} \times (100 \text{ ciclos} \times 5 \cdot 10^{-9} \frac{\text{s}}{\text{ciclo}}) = 3.21 \cdot 10^{-7} \text{ J} \quad (321 \text{ nJ})$$

f) Double data rate

$\hookrightarrow$  en vez de 8 bytes/ciclo,

16 bytes/ciclo,

la transmisión durará

4 ciclos y la lectura

entera 13 ciclos

$$P = 0.642 \text{ W}$$

### Problema 18. Cache Multinivell, DRAM

(3.18)

$$T_c = 10 \text{ ns}$$

$$N = 5 \cdot 10^9 \text{ instr.}$$

$$\text{CPI ideal} = 1.8 \text{ c/i}$$

a)  $T_{\text{exe}} = N \times \text{CPI} \times T_c = 90 \text{ s}$

b) Accesos a la caché d'instruccions?

Tants com instruccions:  $5 \cdot 10^9$  accesos

c)  $t_{\text{penalització Fallada}} = 13 \text{ cycles}$

d)  $T_{\text{ma}} = t_{\text{hit}} + t_{\text{taxmiss}} \times t_{\text{pf}}$   
 $= 1 + 0.1 \times 13$   
 $= 2.3 \text{ cycles} \xrightarrow{T_c = 10 \text{ ns}} 23 \text{ ns}$

e)

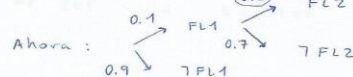
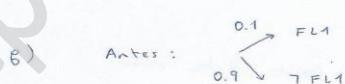
$$\text{CPI} = \text{CPI ideal} + \underbrace{\text{CPI mem}}_{\substack{\text{nr} \times \text{taxmiss} \times t_{\text{pf}} \\ \text{acc/i}}} = 1.8 + 1 \times 0.1 \times 13 = 3.1 \text{ c/i}$$

$$\text{Ciclos} = \underbrace{\text{ciclos ideal}}_{\text{ciclos} = T_{\text{exe}} \cdot f} + \underbrace{\text{ciclos mem}}_{\substack{\text{Fallos} \times t_{\text{pf}} \\ \text{accesos} \times \text{taxmiss}}} = \frac{90 \text{ s}}{10 \cdot 10^{-9} \text{ s/ciclo}} + (5 \cdot 10^9 \times 0.1) \times 13 = 1.55 \cdot 10^{10} \text{ ciclos}$$

$$\downarrow \div \text{instr.}$$

$$3.1 \text{ c/i}$$

f)  $T_{\text{exe}} = N \times \text{CPI} \times T_c = 155 \text{ s}$



taxa local de fallados de L2 es del 30 %

$$P(\text{FL1} \wedge 7\text{FL2}) = 0.1 \times 0.7 = 0.07 \quad (7 \%)$$

h)  $P(\text{FL1} \wedge \text{FL2}) = 0.1 \times 0.3 = 0.03 \quad (3 \%)$

i) 5 cycles

j) 15 cycles (del 2 al 16)



### Problema 19. Detección y corrección de errores

a)  $MTTF = \frac{10^9}{25000} = 40000$  horas de media para 1 fallo cada Mbit

b)  $10^6$  bits, 40000 horas  $\Rightarrow$  1 bit,  $4 * 10^{10}$  horas



$MTTF_{\text{bit}} > MTTF_{\text{Mbit}}$  : tienen que pasar  
más horas (más improbable) para que  
haya un fallo en un bit que para que haya  
uno en  $10^6$  bits

$$4 * 10^{10} \text{ horas} * \frac{1 \text{ día}}{24 \text{ horas}} * \frac{1 \text{ año}}{365 \text{ días}} * \frac{1 \text{ millón de años}}{10^6 \text{ años}} = 4.57 \text{ millones de años}$$

c) 1 Mbit, 40000 horas  $\Rightarrow$  131072 Mbits (16 GB), 0.305 horas

d)  $MTTF = 0.305 * 20000 = 6100$  horas

e)  $\frac{1}{6100} \text{ fallos/hora} * \frac{24 \text{ horas}}{1 \text{ día}} * 500000 \text{ servidores} = 1968 \text{ DIMMs}$

f)  $\frac{70 \text{ MJ}}{1 \text{ chip}} * \frac{18 \text{ chips}}{1 \text{ DIMM}} * \frac{1968 \text{ DIMMs}}{1 \text{ día}} * \frac{365 \text{ días}}{1 \text{ año}} = 905 * 10^6 \text{ MJ/año}$

$$\frac{50 \text{ g de CO}_2}{1 \text{ MJ}} * \frac{905 * 10^6 \text{ MJ}}{1 \text{ año}} * \frac{1 \text{ tonelada de CO}_2}{10^6 \text{ gramos de CO}_2} = 45250 \text{ toneladas de CO}_2/\text{año}$$