

PROBLEMA 19

FIT (failures in time) \rightarrow por cada 10^9 horas por Mbit

25000 fallos por 10^9 horas por Mbit

a) MTTF por Mbit

$$\frac{10^9}{25000} = 40000 \text{ horas para 1 fallo cada Mbit}$$

b) MTTF de 1 bit individual (en horas y en millones de años)

$$\text{MTTF} = 40000 \frac{\text{horas}}{\text{Mbit}} \cdot \frac{2^{20} \text{ bits}}{\text{Mbit}} = 41,9 \cdot 10^9 \text{ horas}$$

$$41,9 \cdot 10^9 \text{ horas} \times \frac{1 \text{ día}}{24 \text{ horas}} \times \frac{1 \text{ año}}{365 \text{ días}} = \frac{1 \text{ millón de años}}{10^6 \text{ años}} = 4,18 \text{ millones de años}$$

c) Calculad el MTTF de la memoria de un computador con 16 GB de memoria instalados

$$16 \text{ GB} \approx \frac{3}{2^{10}} = 131072 \text{ Mbits}$$

$$\text{MTTF} = \frac{40000 \text{ horas}}{131072 \text{ Mbits}} = 0,305 \text{ horas}$$

1 de cada 20000 fallos no eran recuperables

d) Calculad el MTTF de la memoria de un computador con 16 GB de memoria ECC instalados.

$$\text{fallos} = 1/\text{MTTF} = 1/0,305 = 3,28 \text{ fallos/hora} \times 1/20000 = 164 \cdot 10^6 \text{ fallos/hora}$$

$$\text{MTTF} = 1/\text{fallos} = 1/164 \cdot 10^6 = 6098 \text{ horas}$$

500000 servidores de 16 GB de memoria

e) Calculad cuántos DIMM cambia google cada día.

$$24 \frac{\text{horas}}{\text{día}} \cdot \frac{1}{6098 \text{ horas}} \cdot 500000 \text{ servidores} = 1968 \text{ DIMMs al día}$$

1 chip de memoria \rightarrow 70 MJoules de energía

DIMM \rightarrow 18 chips de memoria (16 datos + 2 paridad)

1 MJoule \rightarrow 50g de CO_2

f) Calculad la energía equivalente consumida en fabricar los DIMMs que se reemplazan cada año y el CO_2 equivalente emitido (en toneladas)

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

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

PROBLEMA 20

```

movl $0, %esi
movl $0, %eax
L: movl a(, %esi, 8), %ecx; load a[2 * i]
a: addl %ecx, %eax; suma += a[2 * i]
i: incl %esi; ++i
c: cmpl $N, %esi; i < N
j: jl L

```

dirección de inicio de a 0×20000000
 $N = 64\,000\,000$ ($64 \cdot 10^6$)

$f = 2\text{ GHz}$
 $D1 = 32\text{ bytes}$

DIMM (8 chips de 1 byte cada uno): latencia fila = 9 ciclos
 latencia columna = 9 ciclos
 precharge = 8 ciclos

a) Calcular la tasa de fallos de la cache de datos (D1) al ejecutar el bucle

En movl se mueven 4 bytes, y como los bloques son de 32 bytes, falla 1 de cada 4 veces que accede a la cache. ($1/4 = 0.25$; tasa de fallos = 25%)

b) Completar el Cronograma 5: propia página

c) Calcular el CPI al ejecutar el bucle y el Tex del bucle

$$CPI = (0.25 \cdot 28 + 0.25 \cdot 5) \cdot 64 \cdot 10^6 = 638 \cdot 10^6 \text{ ciclos}$$

$$5 \text{ instrucciones} \cdot 64 \cdot 10^6 = 320 \cdot 10^6 \text{ instrucciones}$$

$$CPI = 638 \cdot 10^6 / 320 \cdot 10^6 = 2.15 \text{ c/i}$$

$$Texe = N \cdot CPI \cdot 1/f = 320 \cdot 10^6 \cdot 2.15 \cdot 1/2 \cdot 10^9 = 0.344 \text{ segundos}$$

Prefetch

d) Completar el Cronograma 6: propia página

e) Calcular el nº de fallos completos y el nº de fallos parciales. ¿crees que, para calcular el CPI, vale la pena tener en cuenta los fallos completos, dado el nº de iteraciones que ejecutamos?

Solo el fallo de la 1ª iteración es completo, todos los otros son parciales. No vale la pena tenerlo en cuenta, ya que es 1 dentro de los $16 \cdot 10^6$ fallos parciales

f) Calcular los ciclos perdidos en caso de tener un fallo parcial

10 ciclos perdidos (del HISS a cuando obtenemos el dato A)

g) Calcular el CPI, el Texe y el speed-up respecto al procesador original

$$\left. \begin{array}{l} \text{instrucciones: } 320 \cdot 10^6 \text{ instrucciones} \\ \text{ciclos: } (0.25 \cdot 45 + 0.25 \cdot 5) \cdot 64 \cdot 10^6 = 480 \cdot 10^6 \text{ ciclos} \end{array} \right\} CPI = \frac{480}{320} = 1.5 \text{ c/i}$$

$$T_{exe} = N \times CPI \times 1/F = 320 \cdot 10^6 \cdot 1.15 \cdot 1/2 \cdot 10^9 = 0.124 \text{ segundos}$$

$$\text{Speed-up} = \frac{0.344}{0.124} = 1.433 \rightarrow 43.33 \% \text{ más rápido}$$

Procesador sin PREFETCH con un controlador de memoria avanzado que no cierra la página (PRECHARGE) después de cada acceso.

h) Calcular el nº de bloques que almacena una página, cuántos accesos a SRAM tienen que abrir página y cuántos pueden reusar la página que está abierta. ¿crees que vale la pena tener en cuenta el hecho que el 1º acceso no necesita cerrar página?

cada página almacena 256 Bytes / 32 bytes = 8 bloques
(1 de cada 8 fallos)

solo 1 de cada 32 accesos tienen que abrir la página y cerrar la anterior

No es necesario tenerlo en cuenta (9 ciclos sobre más de $1 \cdot 10^6$ ciclos)

i) Dibujar en el cronograma 7: página siguiente

j) Dibujar en el cronograma 8: página siguiente

k) Calcular el $T_{transición}$ de los fallos que abren página y el de los que reusan una página abierta.

T_{pf} abren página = 31 ciclos

T_{pf} no abren página = 14 ciclos

l) Calcular el CPI, el T_{exe} y el speed-up respecto al procesador original.

$$CPI = 1 + 0.25(1/8 \cdot 31 + 7/8 \cdot 14)/5 = 1.806 \text{ cli}$$

$$T_{exe} = 320 \cdot 10^6 \cdot 1.806 \cdot 1/2 \cdot 10^9 = 0.288 \text{ segundos}$$

$$\text{Speed-up} = 0.344 / 0.288 = 1.1944 = 19.44 \% \text{ más rápido}$$

COMBINAR PREFETCH CON CONTROLADOR DE MEMORIA AVANZADO

m) Dibujar en el cronograma 9: página siguiente

n) Dibujar en el cronograma 10: página siguiente

o) Calcular los ciclos perdidos por fallo parcial en ambos casos. ¿Se produce fallo parcial con los dos tipos de prefetch?

Si no abre página \rightarrow No tiene ciclos perdidos Si abre página \rightarrow 11 ciclos perdidos

p) Calcular el CPI, el T_{exe} y el speed-up respecto al procesador original

$$CPI = 1 + 0.25(1/8 \cdot 11)/5 = 1.068$$

$$T_{exe} = 320 \cdot 10^6 \cdot 1.068 \cdot 1/2 \cdot 10^9 = 0.175$$

$$\text{Speed-up} = 0.344 / 0.17 = 2.02 \rightarrow 102 \% \text{ más rápido}$$

6DRAM con 2 bancos

2K \rightarrow banco 0

2K \rightarrow banco 1

2K \rightarrow banco 0

...

SIN PREFETCH

q) Dibujar en el cronograma 11: página siguiente

r) Calcular el T_{pf} de los fallos que abren página y el de los que reusan una página abierta

Los que abren página \rightarrow 31 ciclos

Los que reusan página \rightarrow 14 ciclos

Los que abren página cuando cambiam de banco \rightarrow 23 ciclos

s) Calcular el CPI, el T_{exe} y el speed-up respecto al procesador original

$$CPI = 1 + 0.25 [1/64 \cdot 23 + 7/64 \cdot 31 + 56/64 \cdot 14] / 5 = 1.8 \text{ c/i}$$

$$T_{exe} = 320 \cdot 10^6 \cdot 1.8 \cdot 1/2 \cdot 10^9 = 0.288 \text{ segundos}$$

$$\text{Speed-up} = 0.344 / 0.288 = 1.1944 \rightarrow 19.44 \% \text{ más rápido}$$

CON PREFETCH

t) Calcular el CPI, el T_{exe} y el speed-up respecto al procesador original.

Con prefetch \rightarrow se reducen 20 ciclos los ciclos de T_{pf}

$$CPI = 1 + 0.25 [1/64 \cdot 3 + 7/64 \cdot 11] / 5 = 1.0625 \text{ c/i}$$

$$T_{exe} = 320 \cdot 10^6 \cdot 1.0625 \cdot 1/2 \cdot 10^9 = 0.17 \text{ s}$$

$$\text{Speed-up} = 0.344 / 0.17 = 2.0235 \rightarrow 102.35 \% \text{ más rápido}$$

Cronograma 5: SIN prefetch

[illegible]

Cronograma 7: Fallo que NO abre página.

Ciclo	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
Cache	M														D																														
Comando SDRAM		Rd																																											
Datos SDRAM											d0	d1	d2	d3																															

Cronograma 8: Fallo que SI abre página.

Ciclo	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
Cache	M																																											
Comando SDRAM		R							A																																			
Datos SDRAM																																												

Cronograma 9: Prefetch que NO abre página.

Iteración	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
Ciclo	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	
movl a(%esi,8), %ecx	L					L					L					L					L						L																		
addl %ecx, %eax		a					a					a					a					a					a																		
incl %esi			i					i					i					i					i					i																	
cmpl \$N, %esi				c					c					c					c										c																
jll L					j					j					j					j										j															
Cache	H					H					H					H						H																							
Comando SDRAM		Rd																																											
Datos SDRAM																																													

[illegible][illegible]