

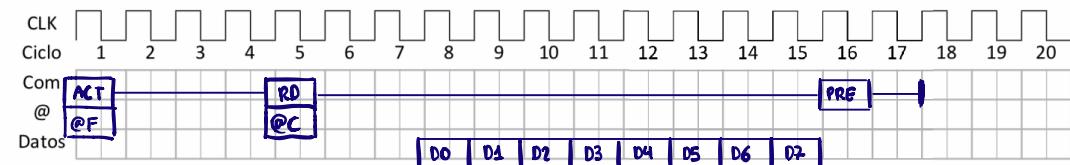
Problema 17. DRAM

Disponemos de un DIMM de memoria DRAM síncrona (SDRAM) con las siguientes características:

- 8 chips de 1 byte cada uno por DIMM
- Latencia de fila: 4 ciclos
- Latencia de columna: 3 ciclos
- Latencia de precarga: 2 ciclos
- Frecuencia de reloj: 200 MHz

A esta memoria realizamos un acceso en lectura en el que leemos un bloque de 64 bytes. Para indicar la ocupación de los distintos recursos utilizaremos la siguiente nomenclatura:

- ACT: comando ACTIVE
 - RD: comando READ
 - PRE: comando PRECHARGE
 - @F: ciclo en que se envía la dirección de fila
 - @C: ciclo en que se envía la dirección de columna
 - Di: ciclo en que se transmite el paquete de datos i (D0, D1, D2, ...)
- a) **Rellenad** el siguiente cronograma indicando la ocupación de los distintos recursos para una operación de lectura de 64 bytes.



- b) **Calculad** el tiempo de ciclo de la memoria en ns.
 c) **Calculad** el ancho de banda teórico máximo suponiendo que el bus de datos está transfiriendo datos continuamente.
 d) **Calculad** el ancho de banda real suponiendo que somos capaces de iniciar un nuevo acceso a un bloque de 64 bytes tan pronto hemos completado el acceso anterior.

$$b) T_c = (200 \cdot 10^{-9})^{-1} = 5 \text{ ns} \quad c) AB_{th} = \frac{64 \text{ B}}{8 \cdot 5 \cdot 10^9} = 1'6 \text{ GB/s} \quad d) AB_{real} = \frac{64}{17 \cdot 5 \cdot 10^9} = 752'94 \text{ MB/s}$$

La tensión de alimentación de esta memoria es de 1.5 voltios, mientras que la corriente consumida depende de la actividad:

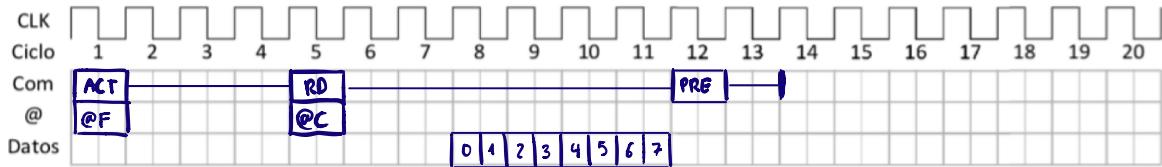
- La memoria está inactiva -> corriente de fugas 200 mA
- Durante toda la operación de lectura (desde que se envía el comando ACTIVE hasta que se completa el PRECHARGE), se consumen 100 mA adicionales debidos al funcionamiento de los componentes internos (además de la corriente de fugas que sigue existiendo)
- Durante la transferencia de datos, además de la corriente de fugas y los componentes internos, hay que alimentar los drivers de entrada/salida, con lo que se consumen otros 500 mA adicionales.
- e) **Calculad** la energía consumida (en julios) y la potencia media consumida (en vatios) en la memoria durante un intervalo de 100 ciclos suponiendo que iniciamos un acceso cada 25 ciclos.

Después de unos años este DIMM de memoria SDRAM es sustituido por un DIMM DRAM DDR (Double Data Rate) manteniendo el resto de características iguales.

- f) **Rellenad** el siguiente cronograma indicando la ocupación de los distintos recursos para una operación de lectura de 64 bytes en la nueva memoria DDR.

$$\left. \begin{aligned}
 e) \quad P_{(\text{activa})} &= \frac{7+2}{25} \cdot 1'5 \cdot (200 + 100) \cdot 10^{-3} = 0'162 \text{ W} \\
 P_{(\text{transferencia})} &= \frac{8}{25} \cdot 1'5 \cdot (200 + 100 + 500) \cdot 10^{-3} = 0'384 \text{ W} \\
 P_{(\text{inactiva})} &= \frac{25-17}{25} \cdot 1'5 \cdot 200 \cdot 10^{-3} = 0'096 \text{ W}
 \end{aligned} \right\} \quad P = 0'642 \text{ W} \\
 E_{\text{coms}} = P \cdot t = P \cdot \#c \cdot t_c = 0'642 \cdot 100 \cdot 5 \cdot 10^{-9} = \boxed{321 \text{ mJ}}$$

f)



Problema 18. Cache Multinivell, DRAM

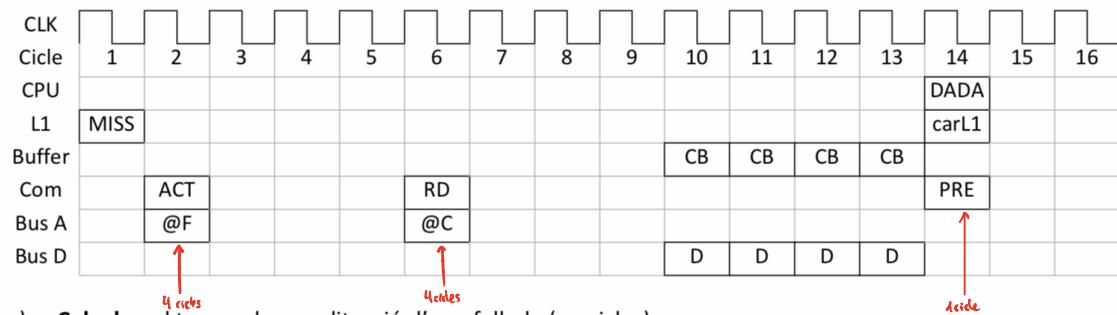
Tenim el disseny de una CPU que tindrà un temps de cicle (T_c) de 10 ns. A l'executar un programa P (que executa 5×10^9 instruccions) en un simulador on tots els accessos a memòria tarden 1 cicle s'ha mesurat un CPI de 1,8 cicles/instrucció (que anomenarem CPI_{ideal}).

- a) **Calculeu** el temps d'execució (T_{exec}) del programa P en aquest sistema de memòria ideal (en segons).

Per mesurar l'impacte de la cerca d'instruccions en el rendiment, hem modificat el simulador per analitzar un sistema amb una cache d'instruccions (que anomenarem L1) i una memòria principal SDRAM (els accessos a dades segueixen tardant 1 cicle). La mida de bloc de L1 es de 32 bytes i el temps d'accés en cas d'encert a L1 (T_h) es de 1 cicle. Pel programa P la taxa de fallades (m_1) de L1 es del 10%. La memòria principal està formada per un DIMM SDRAM de 8 bytes d'amplada amb una latència de fila de 4 cicles, una latència de columna també de 4 cicles i un temps per la comanda PRECHARGE de 1 cicle.

- b) **Calculeu** quants accessos a L1 fa el programa P.

El següent cronograma mostra els passos en cas de fallada a L1. En el cicle 1 s'accedeix a L1 i es detecta que es una fallada de cache. En el cicle 2 s'envia la comanda ACTIVE i l'adreça de fila (Bus A) per activar la pàgina corresponent de memòria i 4 cicles després (cicle 6) s'envia la comanda READ i l'adreça de columna. Al cicle 10 (4 cicles després de RD) apareixen les dades (4 cicles 8 bytes/cicle) al bus de dades (Bus D). Les dades es van carregant a un *buffer* (cycles CB) a mesura que van apareixent pel bus (cycles etiquetats D). Finalment al cicle 14, un cop s'ha transmès tot el bloc al *buffer*, es passa la instrucció a la CPU (DADA) i en paral·lel s'escriu el bloc a L1 (carL1) i s'activa la comanda PRECHARGE per tancar la pàgina (PRE).



- c) **Calculeu** el temps de penalització d'una fallada (en cicles).
d) **Calculeu** el temps mig d'accés a memòria (T_{mam}) pels accessos a instruccions (en nanosegons).
e) **Calculeu** el CPI amb aquesta jerarquia de memòria.
f) **Calculeu** el temps d'execució (T_{exec}) del programa P (en segons).

$$a) T_{exec} = 10 \cdot 10^9 \cdot 5 \cdot 10^8 \cdot 1.8 = 90s$$

$$c) 13 \text{ cicles}$$

$$e) CPI = CPI_{ideal} + CPI_{mem} = 1.8 + \frac{1}{10} \cdot m_1 \cdot t_{PF} = 1.8 + 1.01 \cdot 13 = 34 \text{ c/c}$$

b) Totes les instruccions \rightarrow accés cache \rightarrow $5 \cdot 10^9$ accessos

$$d) T_{mam} = (1 + 0.1 \cdot 13) \cdot 10 = 23ms$$

$$f) T_{exec} = 5 \cdot 10^9 \cdot 34 \cdot 10^{-9} = 155s$$

A aquest sistema afegim un segon nivell de cache (L2) entre la cache d'instruccions (L1) i la memòria principal (SDRAM) de forma que, si es falla a L1 s'accedeix a L2 i només en cas de fallar al segon nivell s'ha d'accedir a memòria principal. La taxa local de fallades (m_2) de L2 es del 30%. La mida de bloc de L2 es també de 32 bytes.

- g) **Calculeu** el percentatge d'accisos que fallen a L1 i encerten a L2.
h) **Calculeu** el percentatge d'accisos que fallen a L1 i a L2.

$$g) 0.1 \cdot 0.7 = 0.07 \rightarrow 7\%$$

$$h) 0.1 \cdot 0.3 = 0.03 \rightarrow 3\%$$

El següent cronograma mostra els passos en cas de fallada a L1 i encert a L2. Al cicle 1 s'accedeix a L1 i es detecta que es una fallada a L1. Un accés a L2 tarda 4 cicles (del 2 al 5). En el cicle 2 (TAG) es llegeix la memòria d'etiquetes, en el cicle 3 (CMP) es comparen les etiquetes i es comprova que es *hit* a L2, en el cicles 4 i 5 es llegeix la memòria de dades de la L2 (RD1 i RD2). Finalment al cicle 6 s'escriu el bloc a L1 (carL1) i, en paral·lel, es passa la instrucció a la CPU (DADA).

CLK	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Cicle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
CPU						DADA										
L1	MISS					carL1										
L2		TAG	CMP	RD1	RD2											

- i) Calculeu el temps de penalització en cas de fallar a L1 i encertar a L2 (en cicles). *i) 5 cicles*

El següent cronograma mostra els passos en cas de fallada a L1 i a L2. Al cicle 1 s'accedeix a L1 i es detecta que es una fallada a L1. En el cicle 2 (TAG) es llegeix la memòria d'etiquetes, en el cicle 3 es comparen les etiquetes i es comprova que es *miss* a L2. Dels cicles 4 al 15 es llegeix el bloc de la SDRAM tal com ja s'ha explicat per la configuració amb un sol nivell de cache. Un cop tenim el bloc al *buffer*, aquest s'escriu simultàniament a L1 (carL1), L2 (2 cicles WR1 i WR2) i es passa la instrucció a la CPU (DADA).

CLK	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
Cicle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
CPU																DADA		
L1	MISS															carL1		
L2		TAG	CMP														WR1	WR2
Buffer															CB	CB	CB	CB
Com																		PRE
@																		
Datos															D	D	D	D

- j) Calculeu el temps de penalització en cas de fallar a L1 i a L2 (en cicles).
k) Calculeu el temps mig d'accés a memòria (T_{mam}) pels accessos a instruccions (en nanosegons).
l) Calculeu el CPI amb aquesta jerarquia de memòria.
m) Calculeu el temps d'execució (T_{exec}) del programa P (en segons).
n) Calculeu el guany (speed-up) del sistema amb L1 i L2 respecte el sistema que només té L1.

j) 14 cicles

$$k) T_{mam} = (1 + 0.07 \cdot 5 + 0.03 \cdot 14) \cdot 10 = 17.7 \text{ ms}$$

$$l) CPI = CPI_{ideal} + CPI_{mem} = 1.8 + \frac{1}{1000} \cdot m \cdot tpf = 1.8 + 1 \cdot (0.07 \cdot 5 + 0.03 \cdot 14) = 2.57 \text{ CPI}$$

$$m) T_{exec} = 5 \cdot 10^9 \cdot 10 \cdot 10^{-9} \cdot 2.57 = 128.5 \text{ s}$$

$$m) Speedup = \frac{155}{128.5} = 1.1673$$