



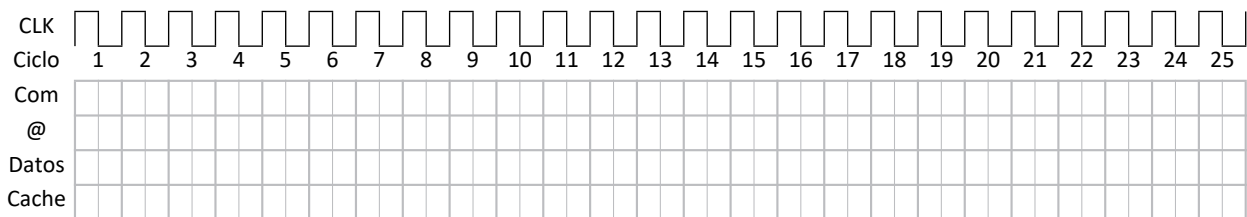
Esta cache está conectada a una memoria principal formada por un único módulo DIMM estándar de 4 Gbytes. Este DIMM tiene 8 chips de memoria **DDR-SDRAM (Double Data Rate Synchronous DRAM)** de un byte de ancho cada uno. El DIMM esta configurado para leer/escribir ráfagas de 64 bytes (justo el tamaño de bloque de las caches). La latencia de fila es de 4 ciclos, la latencia de columna de 3 ciclos y la latencia de precarga de 1 ciclo.

La cache no tiene buffers de escritura, por lo que siempre se escribe en memoria principal primero el bloque reemplazado (en caso que sea pertinente) antes de leer el nuevo bloque, tampoco tiene continuación anticipada ni transferencia en desorden.. El controlador de memoria envía los comandos necesarios a la DDR-SDRAM de forma que ambos bloques sean transferidos lo más rápidamente posible y se maximice el ancho de banda.

En los siguientes cronogramas indica la ocupación de los distintos recursos de la memoria DDR: bus de datos (Datos), bus de direcciones (@), bus de comandos(Com). Indica también la ocupación de la cache (Cache), especifica cuando se accede a la cache si es acierto (A) o fallo (F), y cuando la CPU lee el dato de cache (LC) o escribe el dato en cache (EC). En todos los cronogramas supondremos que no hay ninguna página de DRAM abierta.

Para el caso **fallo y escritura con dirty bit = 0**

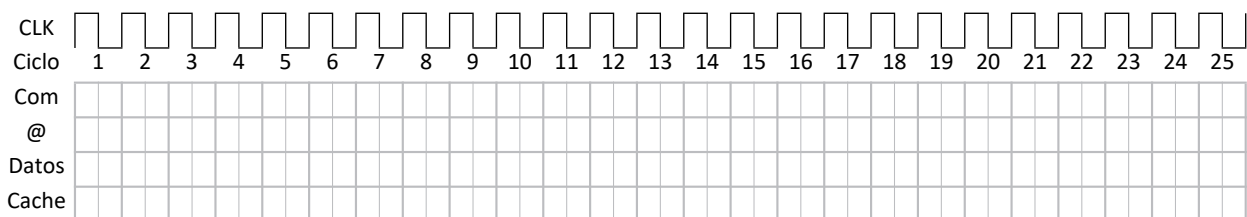
d) **Rellena** el siguiente cronograma.



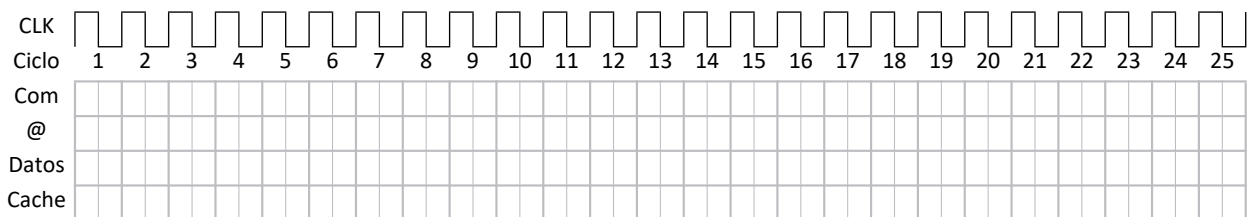
Para el caso **fallo y lectura con dirty bit = 1**

Rellena los siguientes cronogramas en función de la ubicación de los bloques involucrados.

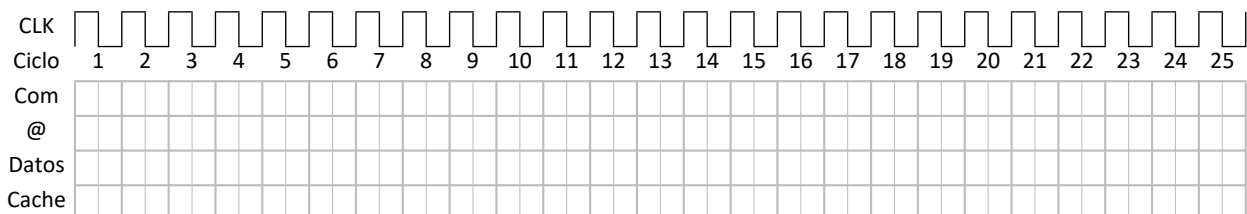
e) Ambos bloques están ubicados en el mismo banco pero en páginas distintas.



f) Ambos bloques están ubicados en la misma página.



g) Ambos bloques están ubicados en bancos distintos.





El procesador Pci es un procesador CISC en donde, en media, cada instrucción CISC se descompone durante la decodificación en 1,5 micro-operaciones (uops). Sabemos que un acceso a MP consume 40 nanojoules, un acceso a la cache de instrucciones consume 5 nano joules, decodificar una instrucción CISC consume 3 nanojoules, y ejecutar una micro-operación (uop) consume 4 nanojoules.

d) **Calcula** la energía consumida por el procesador Pci durante la ejecución de la aplicación A

Algunos procesadores incluyen, además de una cache de instrucciones de nivel 1, una cache de uops (que denominan de nivel 0) donde se guardan las uops que ya han sido traducidas para que en caso de hit en la uop-cache no sea necesario decodificar y traducir las instrucciones CISC. La introducción de una uop-cache reduce tanto el número de instrucciones CISC decodificadas como el número de accesos a la cache de instrucciones, el número de accesos a MP se mantiene.

En nuestro procesador (que ahora llamaremos Puop) hemos introducido una pequeña uop-cache. Para la aplicación A, gracias a la uop-cache, solo el 20% de las instrucciones CISC deben ser leídas de la cache de instrucciones y decodificadas. Sabemos que el tamaño de bloque de la cache de instrucciones es de 32 bytes, que cada instrucción CISC tiene un tamaño de 4 bytes y cada uop tiene un tamaño de 6 bytes.

e) **Calcula** el número de bytes leídos de la uop-cache, el número de bytes leídos de la cache de instrucciones y el número de bytes que se leen de la memoria principal durante la ejecución de A.

Una de las ventajas de la cache de micro-uops es que se produce un ahorro de energía ya que no es necesario decodificar todas las instrucciones dinámicas. Un acceso a la uop-cache para leer una uop consume 2 nanojoules.

f) **Calcula** la energía consumida por el procesador Puop durante la ejecución de la aplicación A.



- d) **Describe** las principales características de cada uno de estos sistemas RAID, **dibujando** un esquema de cómo se distribuyen los datos y especificando el tipo de entrelazado, Terabytes de información redundante, número mínimo de discos que han de fallar para que el sistema deje de ser operativo, ancho de banda **efectivo** de las lecturas en acceso secuencial y ancho de banda **efectivo** de las escrituras en acceso secuencial.

Decidimos configurar el sistema de discos como RAID 6. Al sistema multiprocesador con 32 procesadores y RAID 6 de 6 discos le llamaremos PC3

- e) **Calcula** el speed-up máximo, **sobre el PC2**, que podemos esperar al ejecutar el programa P en el **PC3**, asumiendo que todos los accesos a disco son **lecturas secuenciales**.

Sabemos que la Fase 3 tarda 15000 segundos con un solo disco D (PC1 y PC2) y que el 60% del tiempo se realizan **lecturas secuenciales** y el 40% del tiempo se realizan **escrituras aleatorias**.

- f) **Calcula** el tiempo de la Fase 3 en el **PC3** y el ancho de banda efectivo del RAID 6 durante la Fase 3