El sistema de entrada/salida

- Gestión de bloques disponibles
- Caché de disco
- Scheduler de disco
- Driver
- SSDs vs discos duros

Gestión de bloques disponibles

- Lo intuitivo es usar un lista enlazada de bloques disponibles
- Pero con el tiempo la partición se fragmenta: los archivos quedan dispersos en el disco
- Recuerde: el sistema de archivos debe hacer todo lo posible porque los archivos queden contiguos en el disco
- Es mejor usar un vector de bits *disp* en donde *disp[i]* indica si el bloque *i* está disponible
- La variable *next* indica cuál es el próximo bloque por revisar
- Cuando se necesita un nuevo bloque se revisa *disp[next]*, *disp[next+1]*, *disp[next+2]*, etc., hasta encontrar un bloque disponible
- *next* se deja apuntando al bloque que no se alcanzó a revisar
- Así, los bloques asignados quedan suficientemente contiguos
- El vector de bits ocupa un espacio insignificante en la partición: un bit por cada bloque

Capa: caché de disco

- Almacena los bloques recientemente usados de una partición
- Ocupa típicamente el 30% de la RAM del computador: es enorme
- Cuando se lee un bloque, se revisa primero si está en el caché y si se encuentra ahí el acceso es muy rápido
- Si no se encuentra en el caché, el acceso es mucho más lento: se le pide al scheduler de disco que lo lea de disco y se almacena en el caché reemplazando algún otro bloque que lleve un tiempo significativo sin ser usado
- Lee más bloques que los solicitados porque es probable que se soliciten pronto (*read-ahead*)
- Cuando se escribe, se escribe primero en el caché y se lleva a disco más tarde (*write-after*)
- Un proceso *daemon* llamado *update* lleva a disco cada 30 segundos todas las escrituras pendientes invocando la llamada a sistema *sync*
- Se puede forzar la escritura invocando explícitamente *sync* o ejecutando el comando *sync* que llama a *sync*

Capa: scheduler de disco

- Pueden ser varios los procesos que solicitan acceder al disco concurrentemente
- El *scheduler de disco* decide en qué orden se atienden las solicitudes de los procesos
- Por ejemplo en un instante dado se acaba de acceder al sector 600 pedido por el proceso P0 y están encoladas solicitudes de acceso de los procesos P1, P2, P3, P4 y P5 a los bloques 400, 100, 800, 500 y 900 respectivamente
- Atender a los procesos por orden de llegada es lo más justo: 400, 100, 800, 500, 900, 601*, 401*, 901*, etc.
- 601*, 401*, etc., son nuevas solicitudes de los procesos P0 y P1.
- Pero hay estrategias de atención más eficientes que *reducen el movimiento del cabezal*
- *Shortest Seek Time First* (*SSTF*): Lo más eficiente es acceder siempre al sector que signifique el mínimo moviento del cabezal: 500, 400, 100, 401*, 601*, 800, 900, ...
- Desventaja: hambruna
- Ejemplo: P1 accedió a 100, P2 solicita 200, P3 solicita 900
- Orden de atención: 100, 200, 101*, 201*, 102*, 202*, 103*, 203*, etc.

Estrategias LOOK y C-LOOK

- LOOK o Método del ascensor: se atiende barriendo primero en una dirección y luego barriendo en la dirección opuesta: 800, 900, 500, 400, 100, 401*, 601*, etc.
- Problema: los bloques céntricos se atienden más frecuentemente que los bloques en los extremos
- C-LOOK: como LOOK pero se atiende siempre en la misma dirección, por ejemplo de subida: 800, 900, 100, 400, …
- Ventaja: no hay bloques previlegiados

Capa: driver del disco

- Implementa una API estándar para acceder a todos los tipos de disco: M.2, SATA, ATA, SCSI, etc.
- Ve el disco como un arreglo de sectores de 512 bytes (o 4096 bytes en discos de más de 2 TB)
- Accede directamente a los puertos de entrada/salida de la interfaz del disco para ejecutar comandos de lectura o escritura de *n* sectores a partir del *k*-ésimo sector

Capa: el SSD o M.2

- Un SSD se comporta exactamente como un disco SATA de 2.5'' y por lo tanto lo puede reemplazar

- Pero los datos se almacenan en memoria flash y por lo tanto los accesos directos son mucho más rápidos: ~ 40 mil acceso/seg
- Un SSD no tiene cabezal
- Se pueden leer/escribir secuencialemente a la máxima tasa de transferencia de SATA2: 500 MB/seg

- Un dispositivo M.2 NVMe también almacena datos en memoria flash pero usa un formato más compacto

- No es compatible con un disco duro
- Usa hasta 4 líneas PCI-express para transferir datos y por lo tanto llega a ser hasta 4 veces más rápido que un SSD en acceso secuencial

SSD vs disco duro

- Un SSD de 1 TB cuesta ~ 2 veces el costo de un disco duro de 1 TB
- Pero hoy en día un drive de 256 GB es más barato que un disco duro de 1 TB
- No hay discos duros de 256 GB
- Es más barato vender un notebook con un SSD de 256 GB que con un disco de 1 TB y el producto es de mejor calidad
- Ya no se deberían vender notebooks con discos duros
- El tiempo de vida de un disco duro es de 5 años

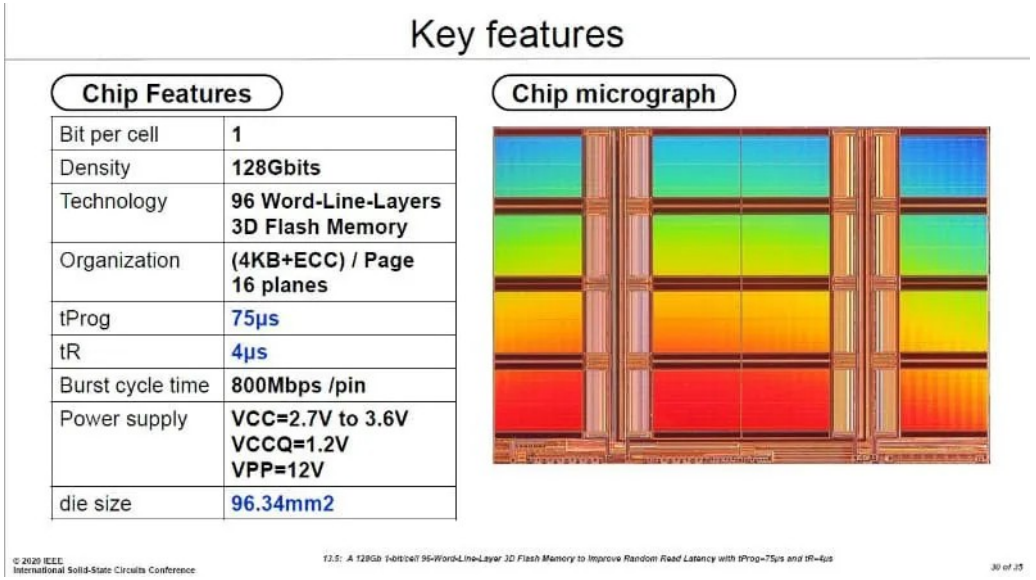
- El tiempo de vida de un SSD se mide en *drive writes per day* (*DWPD*): típicamente medio drive por día durante 5 años
- Pero un SSD puede durar más años si se escribe menos, un disco duro no
- En un notebook con poca memoria, el paginamiento puede escribir más que medio drive por día
- Las celdas de memoria flash se desgastan con las escrituras, pero no con las lecturas
- Un disco duro no se desgasta con el uso

Las memorias flash

- En un SSD los datos se almacenan en celdas de memoria flash
- Hoy en día las celdas de un SSD son del tipo TLC (*triple level cell*)
- Cada celda memoriza 8 niveles de voltaje, lo que se traduce en que almacena 3 bits
- Una celda SLC (*single level cell*) almacena un solo bit por celda
- Las escrituras son mucho más rápidas en una celda SLC que en una celda TLC
- Una celda TLC se puede reescribir unas 300 a 1000 veces, una SLC unas 100 mil veces
- Un drive moderno usa celdas TLC con un caché SLC de unos 16 GB
- Los datos se escriben primero en el caché SLC, y se graban más tarde en la TLC
- Si se escribe en el caché demasiado rápido, se llena y el SSD se pone muy lento porque se escribe a la velocidad de las celdas TLC
- Esto no sucede en el uso normal de un SSD

La memoria flash 3D

- Hoy en día las celdas de memoria flash se disponen de manera vertical: usan las 3 dimensiones
- Típicamente 96 capas
- Este avance ha permitido aumentar considerablemente la capacidad

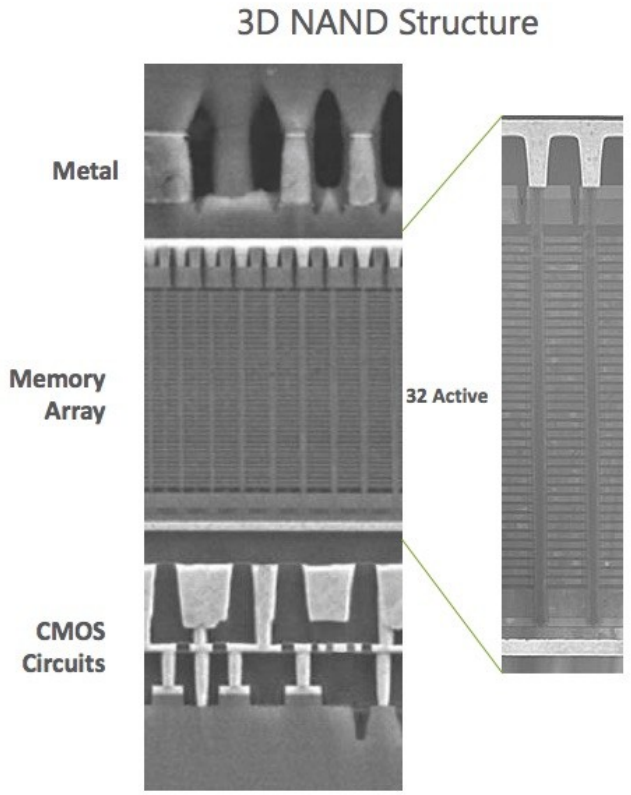


Key features

Chip Features

Bit per cell	1
Density	128Gbits
Technology	96 Word-Line-Layers 3D Flash Memory
Organization	(4KB+ECC) / Page 16 planes
tProg	75µs
tR	4µs
Burst cycle time	800Mbps /pin
Power supply	VCC=2.7V to 3.6V VCCQ=1.2V VPP=12V
die size	96.34mm2

Chip micrograph



- Usan fábricas de unos 30 nm
- Usar 7 nm disminuiría el límite de reescrituras

Nivelación del desgaste en un SSD

- En un SSD es crítico que las celdas se desgasten por parejo
- En una partición hay *hot spots*: sectores que se escriben mucho más frecuentemente
- Se desgastarían prematuramente
- En un SSD un mismo sector se escribe en distintas partes de la memoria flash para nivelar el desgaste
- De esto se encarga el controlador del SSD
- El controlador mantiene un diccionario con la ubicación de cada sector en la memoria flash
- En los SSDs caros el diccionario se almacena en una DRAM, obteniendo un máximo desempeño
- Los SSDs baratos almacenan el diccionario en la misma memoria flash, reduciendo significativamente el desempeño
- Los SSDs de gama media poseen un caché para almacenar las entradas del diccionario recientemente utilizadas
- La velocidad de acceso secuencial en un SSD se obtiene gracias al uso en paralelo de múltiples canales de memoria flash

Conclusiones

- El sistema de archivos fue pensado para discos: se esfuerza en que los archivos queden consecutivos en el disco, si se reescribe el disco, se graba en la misma ubicación
- Funciona razonablemente en acceso secuencial al disco, pero es muy lento en acceso directo
- También funciona con un SSD, y mucho más rápido, pero se podría ganar más con sistemas de archivos diseñados para SSDs, evitando por ejemplo el controlador que encarece el SSD
- El acceso pasa por diferentes capas en el sistema operativo hasta llegar al comando que lee o escribe en el disco o SSD
- Si el archivo no es realmente un archivo, el subsistema o módulo respectivo se encarga del acceso, sin pasar por el sistema de archivos
- Prefiera computadores con SSDs pero grabe sus películas en discos
- Tenga presente que los SSDs se desgastan con las escrituras
- Tenga presente que los discos duran 5 años
- ¡Respalde su información!