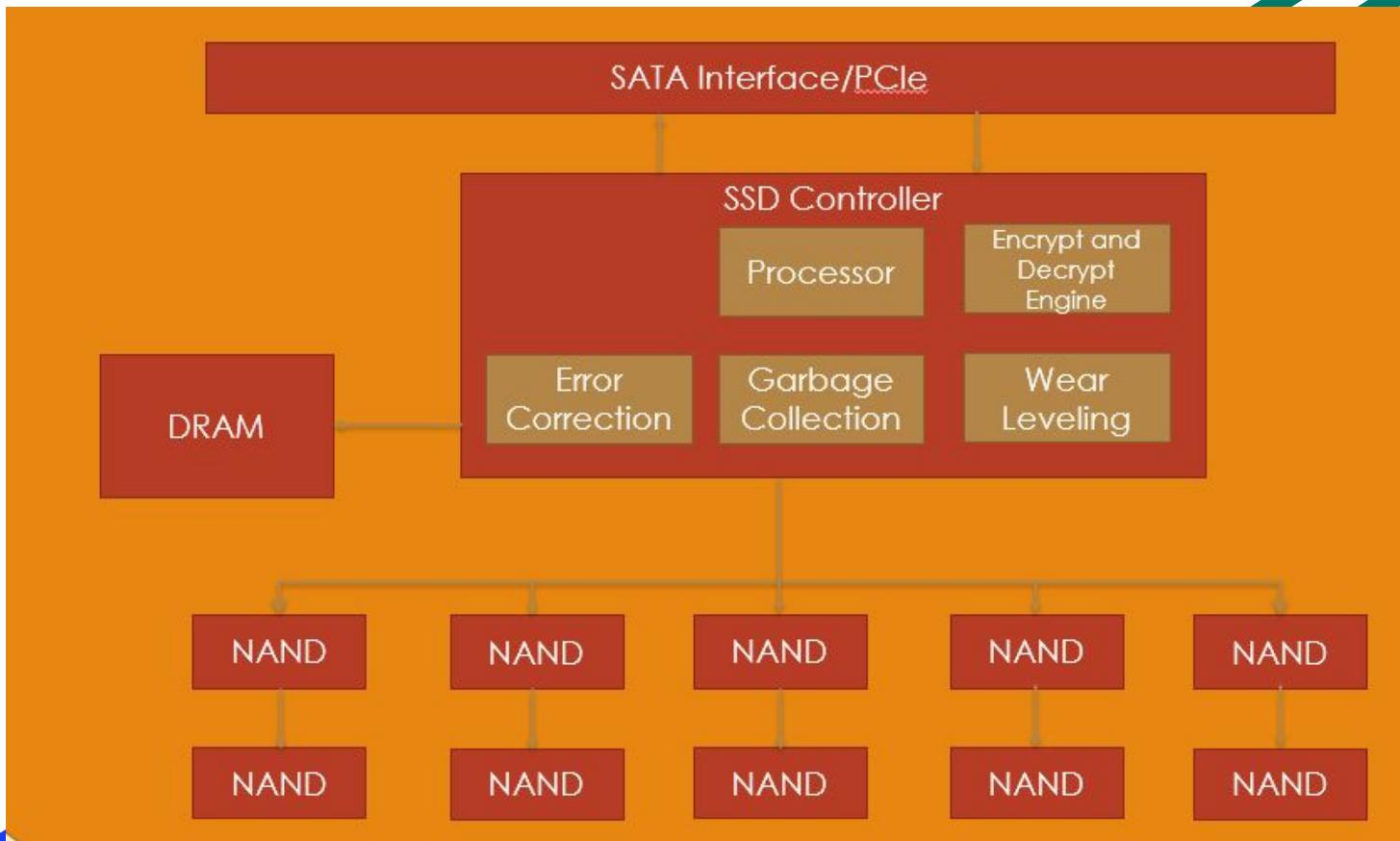


Solid State Drives

***DRISHTI GOEL
AESHA ALTHNYAN
CATHALINA FONTELLE
MENG XU***

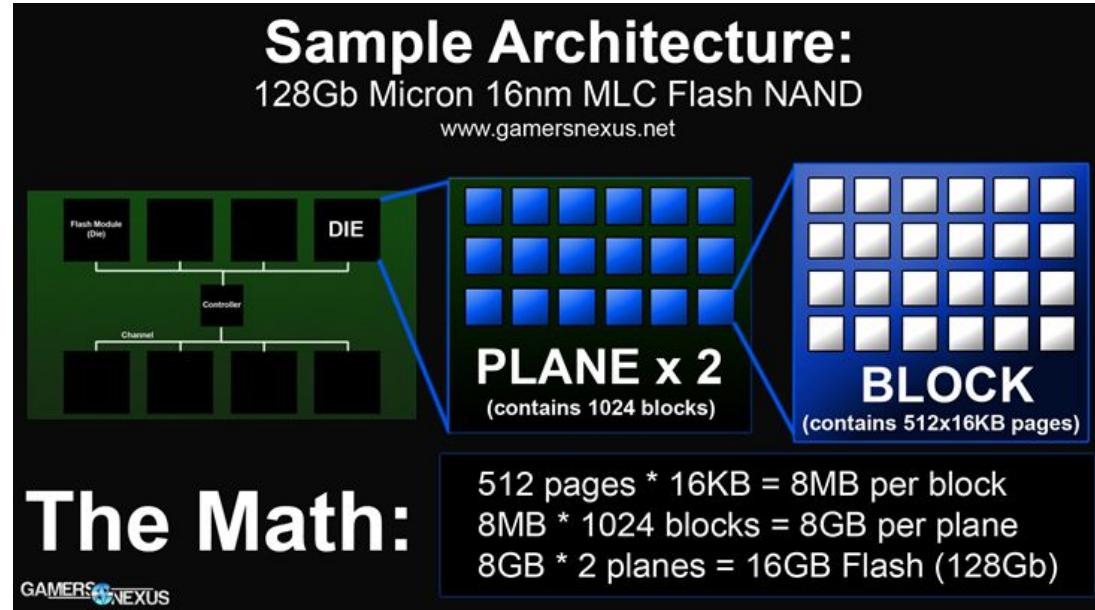
Contents

- **SSD Performance >>> HDD Performance (up to 10-20 times)**
- **What are the techniques used?**
- **Mechanism of Read/Write on SSD**
- **What factors affects the performance of SSD?**
- **Project Approach**



NAND ARCHITECTURE

- Die
- Planes
- Block
- Pages
- Cell

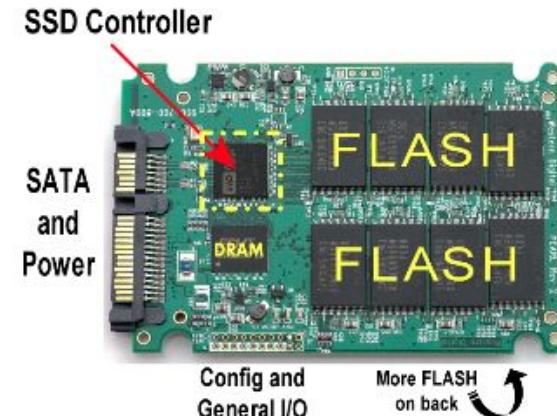


DIFFERENCE BETWEEN SLC, MLC and TLC

Type of Nand	SLC	MLC	TLC
Speed	Fastest	Slower	More Slower
Price	Expensive	Cheaper	More Cheaper 30% lower than MLC
Degrade	Slowest	Slower than LC	Degrade Faster
Endurance	90-100,000 program/erase cycles per cell	10,000 program/erase cycles per cell	3-5,000 program/erase cycles per cell
Bits	1 bit/cell	2 or more bits per cell	3bit/cell

SSD Controller

- Called processor
- Has electronics that bridge the Flash memory components to the SSD input/output interfaces.
- Embedded processor
- Executes firmware-level software.
- Has a DDR3 or DDR4 memory pool to help with managing the NAND
- Not much detail on SSD controllers !
 - because companies lock down their secret.



SSD Controller

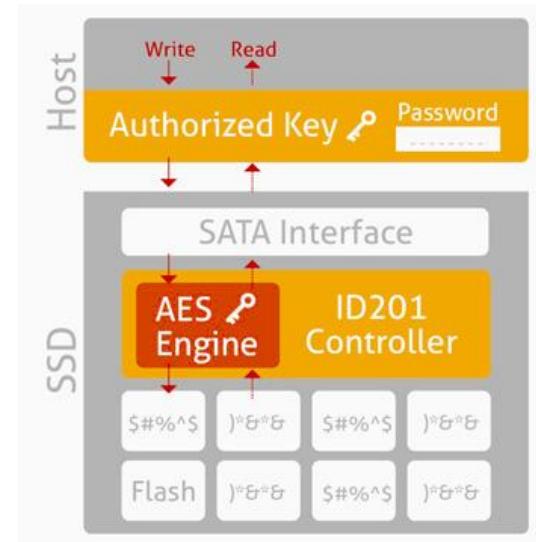
1. Processor:

- 16 or 32 bit embedded microcontroller
- programmed in C
- compiled to run on the target CPU engine
- Execute firmware for the main Flash application
- diagnostics
- caching
- security functions

SSD Controller

2. Encrypt and decrypt engine:

- Encryption in HDs: straightforward .. locate the data and overwrite the bits.
- Because of the wear-leveling algorithms in SSDs:
 - Remaining data are spread through a drive
 - Difficult to securely destroy sensitive information without erasing the whole drive.
- Now: SSDs have Hardware-based disk encryption on board.
 - Advanced Encryption Standard (AES) encryption
 - Encrypted SSDs operate at full speed, no system performance impact.
 - Handled by a dedicated crypto processor on the drive.



SSD Controller

3. Error correction:

- No mechanical parts: less unexpected drive failure.
- **SSDs & HDDs** use error correction: to provide virtually error free storage.
- **NAND** flash: controller maps around bad memory areas and use ECC
- **ECC**: Hardware function that adds redundancy bits with each byte stored.
 - for data error detection and correction upon readout.
- Performed automatically in the ECC hardware

SSD Controller

4. Wear leveling:

- Technique to increase the lifetime of the memory.
- principle : evenly distribute writing on all blocks of a SSD.
 - All cells receive the same number of writes, to avoid writing too often on the same blocks.
 - EX: write 100 GB of data daily on a SSD with 400 GB of space
 - wear leveling: ensures that the 100 GB of data is not always at the same location in the physical flash blocks.

- **Algorithms:**

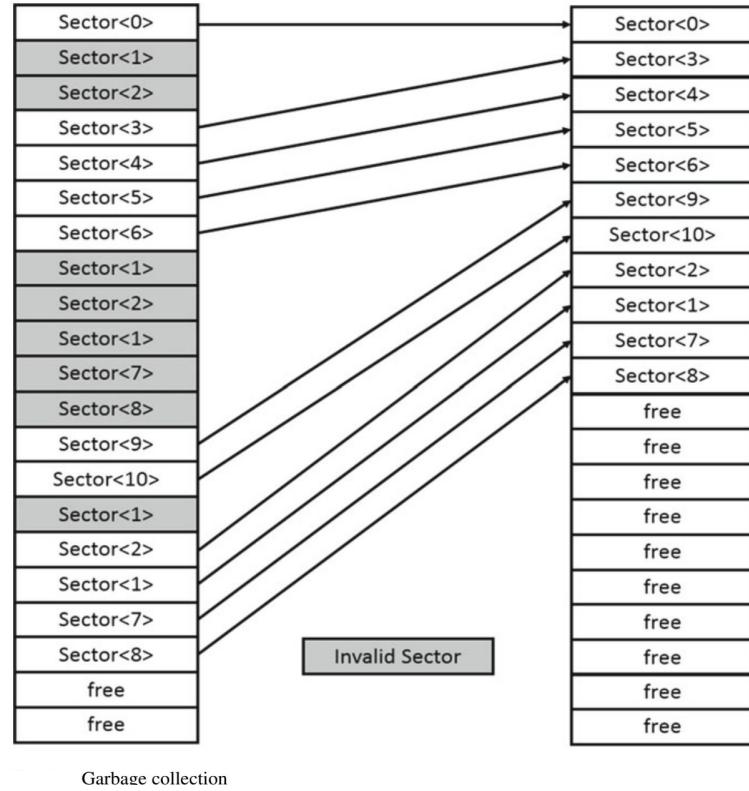
- Dynamic
 - Static



SSD Controller

5. Garbage Collection:

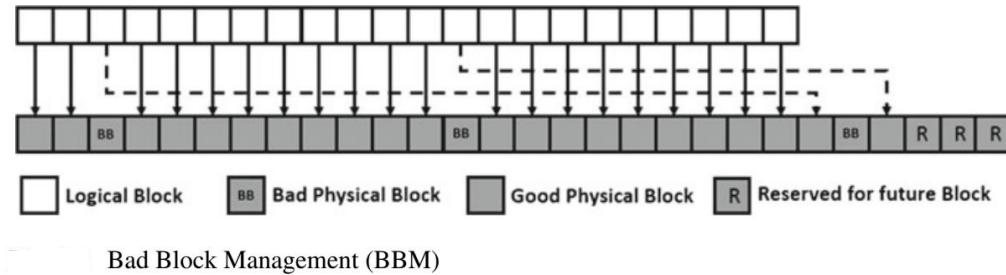
- Collects blocks containing invalid sectors where number of free sectors fall below a certain level.
- Copy all valid sectors to new block from over-provision space
- erase block with invalid sectors and mark it available for use



SSD Controller

6. Bad Block Management:

- Creates and maintain a map of bad blocks

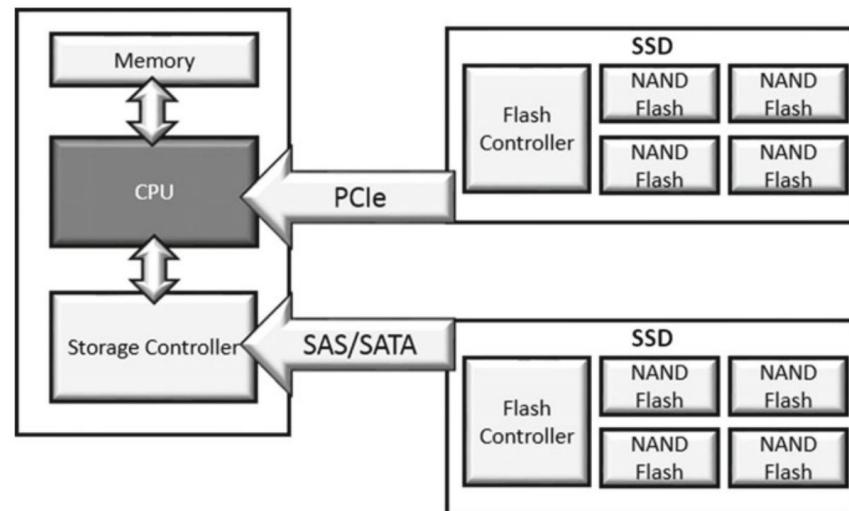


SSD Interfaces

- **Serial ATA (SATA)** - based on a point-to-point physical connection with speeds of 3-6Gbps
- **Serial Attached SCSI (SAS)** - based on a point-to-point physical connection with speeds of 6-12Gbps
- **PCIe** - point-to-point topology, with separate serial links connecting every device to the host. Speeds can reach up to 32Gbps.

SSD Interfaces

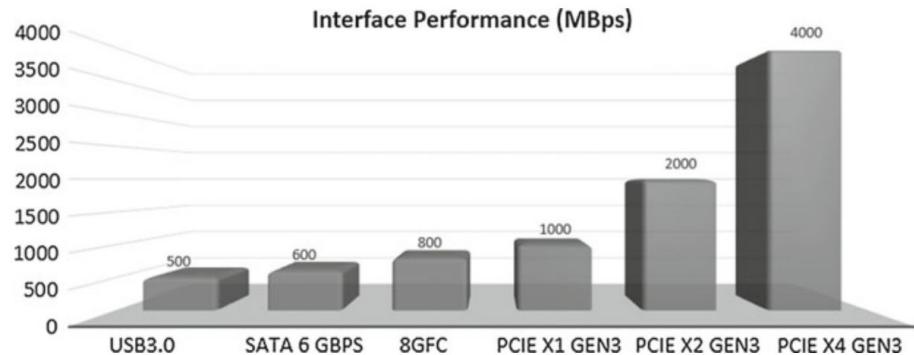
- **PCIe** - direct connection to the CPU
- **SAS/SATA** - connect to the CPU via Storage Controller



PCIe SSD versus SAS/SATA SSD

SSD Interfaces

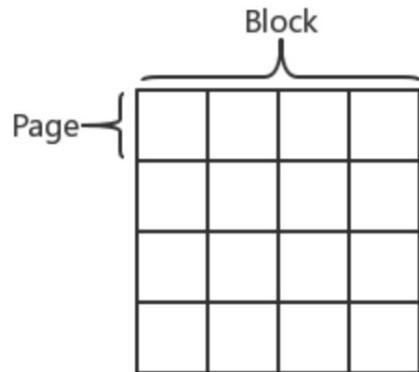
Performance comparisons



Interface performance. PCIe improves overall system performance by reducing latency and increasing throughput

SSD's Read

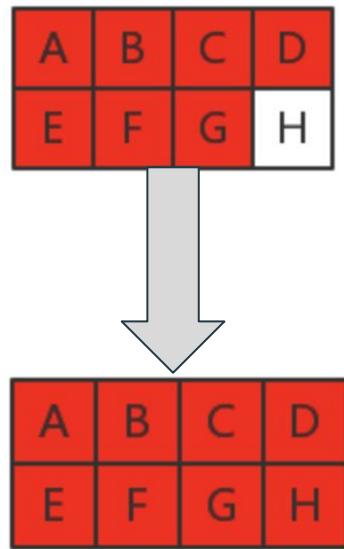
- In SSD, there is a mapping table which be used to maintained a mapping of logical addresses to physical addresses. For each time doing the read, can directly calculate the physical address by logically looking up the table.



SSD's Write

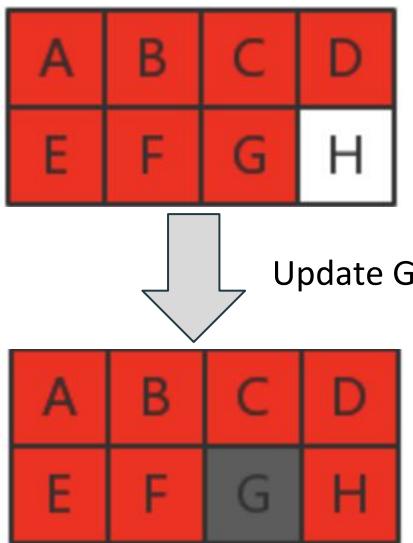
- NAND Flash must be written in units of page each time, and can only write to idle pages, and cannot overwrite pages that have original content.
- When erasing data, it can only be erased in blocks due to high voltage.

New Written



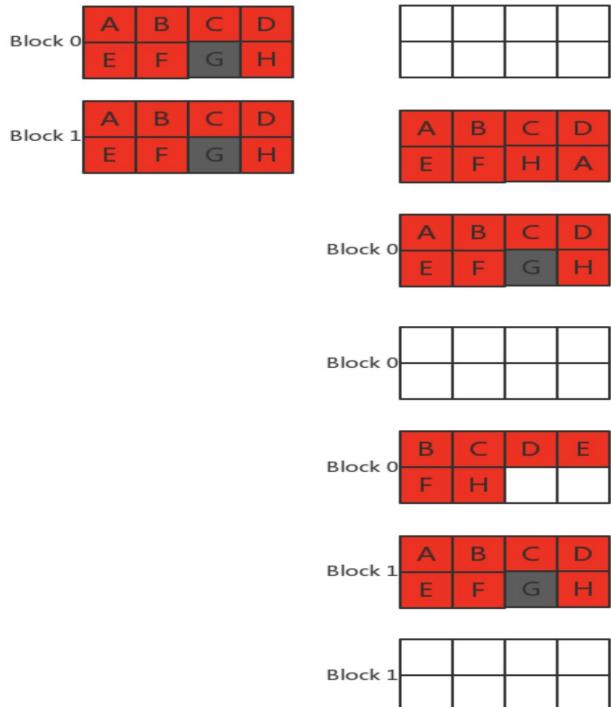
- Find a free page
- Write data to the idle page
- Update mapping table

Update



- Since the SSD cannot do the overwrite, first find a free page H
- Read the data in page G to the buffer inside the SSD, and update the updated bytes to the buffer.
- Write the data in the buffer to H
- Update G page in mapping table, set to invalid page
- Update the H page in the mapping table and add the mapping

Over-provisioning and garbage collection of SSD



Over-Provisioning

- First, find a free block from the over-provisioning space.
- Copy the ABCDEFH of Block0 and the A of Block1 to the free block
- Erase Block 0
- Copy Block1's BCDEFH to Block0, then Block0 has two free pages.
- Erase Block1

Factors contemplated for performance

- Input/Output request
- Write History
- Data placement
- Paralellism
- Write Ordering
- Workload management

Performance Metrics

IOPS

- Inputs/Outputs per second

Throughput

- Rate of data transfer or the bandwidth

Response Time

- The time between the request of data from storage (SSD) and when the data is provided to the processor

Project Approach

- Why do we need simulators:

Since we want to discuss that SSD performance in over time and usage, we have to use simulators to gather accurate results.

Simulators

- Performance-Model Simulators:
 - Mainly used for hardware architecture design analysis
 - Provide data under specific Hardware environment
 - Example: DiskSIM, Eagle Tree
- Behavior Model Simulators:
 - Mainly used for firmware development and verification
 - Be used under specific hardware platform
 - Example: VISSM, eVSSIM

Project Apporach

- Use the simulator: Eagle Tree
- Change the configuration of the SSD to see how they going to affect the performance of write and read

ANY
QUESTIONS?
•

References

- Understanding Intrinsic Characteristics and System Implications of Flash Memory based Solid State Drives(Key Paper)
- http://www.samsung.com/hu/business-images/resource/white-paper/2014/01/Whitepaper-Samsung_SSD-0.pdf
- <https://www.microsoft.com/en-us/research/wp-content/uploads/2008/06/USENIX-08-SSD.pdf>
- <http://bit.csc.lsu.edu/~fchen/publications/papers/hpca11.pdf>
- <https://www.cactus-tech.com/resources/blog/details/solid-state-drive-primer-11-controller-functions-garbage-collection>
- http://www-users.cs.umn.edu/~park/docs/thesis_PhD.pdf
- https://www.usenix.org/legacy/event/fast11/posters_files/Park_D.pdf
- <https://www.snia.org/sites/default/files/SNIASSSI.SSDPerformance-APrimer2013.pdf>
- <http://www.kingston.com/us/community/articleDetail?ArticleId=10>
- https://www.usenix.org/legacy/events/fast11/tech/full_papers/Wei.pdf