Embedded System
Random Performance Issue

하이닉스 반도체㈜

Contents

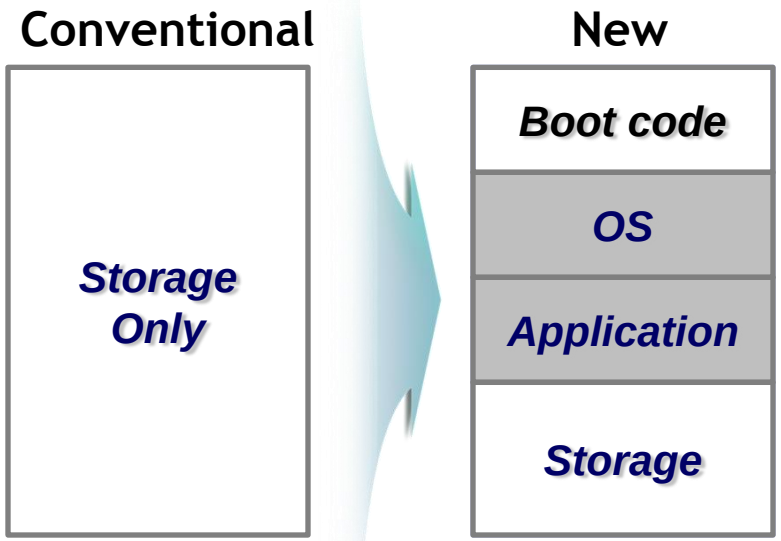
- Trend of storage device market
- Trend of NAND flash tech
- How to increase random performance?
  - Software side
  - Hardware side

- **Trend of storage device market**
- Trend of NAND flash tech
- How to increase random performance?
  - Software side
  - Hardware side

Random IOPS in mobile phone

Smartphone OS의 경우 작은 File size Access가 빈번하게 일어나며
Random IOPS 가 Mobile system의 Key factor 임.

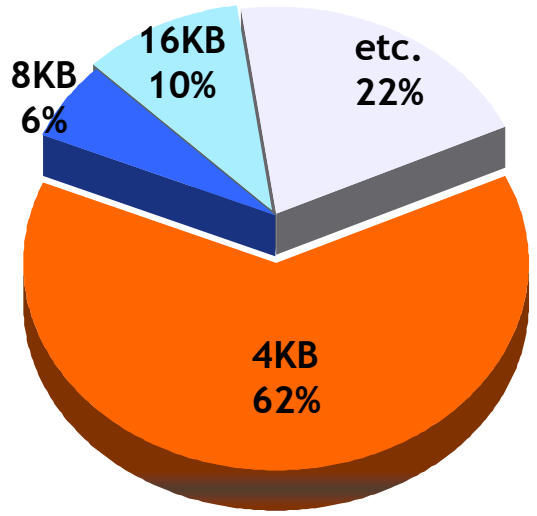
Why Random IOPS



4KB random access가 빈번하게 일어나는 영역

IOPS: 초당 NAND에 Access할 수 있는 횟수

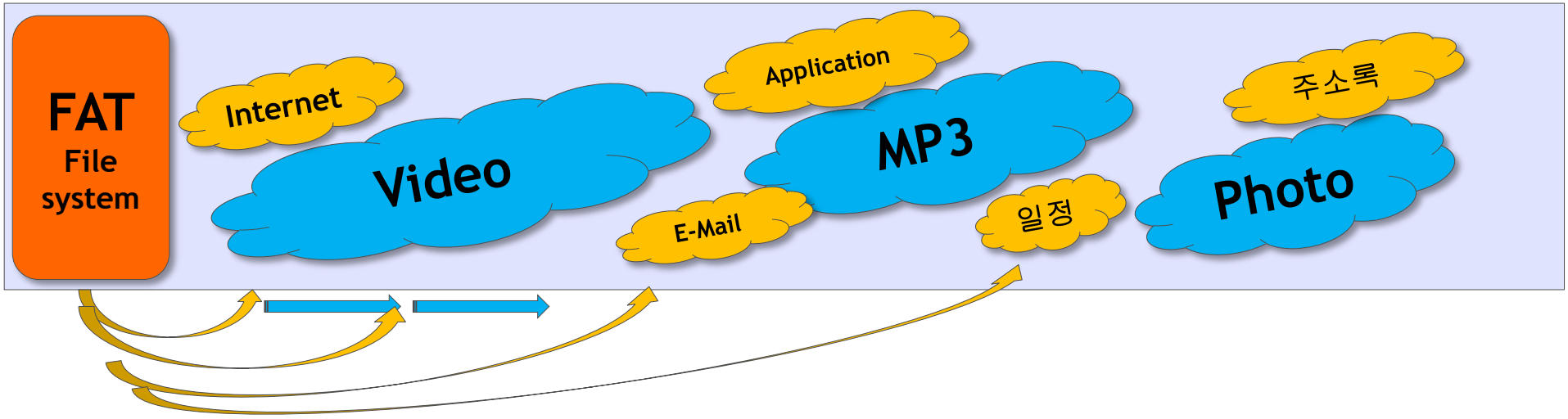
IO Behavior in Smartphone



* 4KB Random IOPS가 Performance 주요 Key Factor임.

Source: Intel Mobile platform

User operation



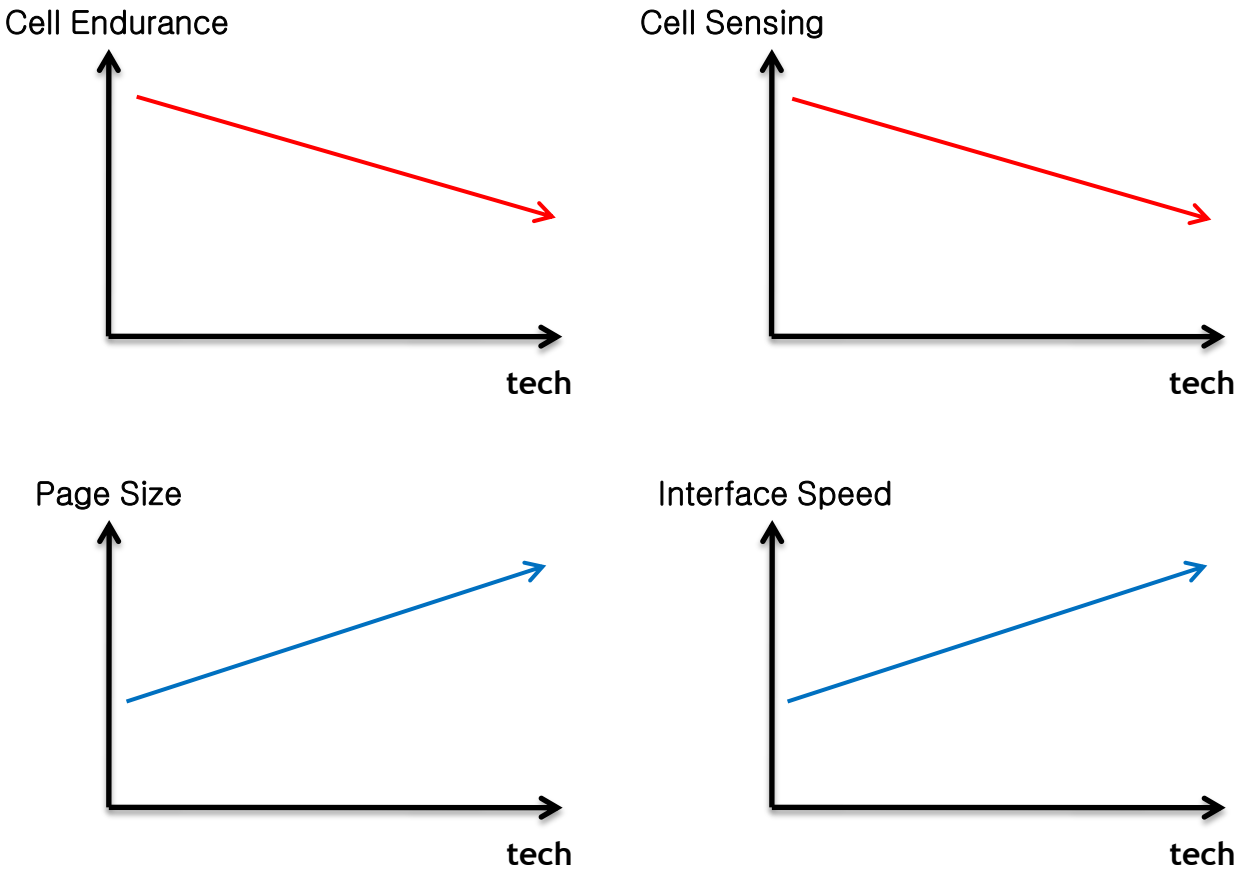
구분	Contents	Data Size	특징
Tiny	FAT File-system data	512 bytes, 1K	• File에 대한 위치 정보등 metadata 기록 • Data와 Data 사이에 빈번한 access 발생 • 동일 위치에 대한 빈번한 read/write
Small	OS System files, Application Internet, e-Mail	4K, 8K, 16K, …	• System, Internet caching에 의한 fragmentation 발생 • 빈번한 read/write 발생
Large	Movies, MP3, Photos	4M, 1.4G, 2.1G, 4.5G, …	• 낮은 update 주기 • 반복된 read 발생

- Trend of storage device market
- **Trend of NAND flash tech**
- How to increase random performance?
  - Software side
  - Hardware side

Disadvantage of NAND Flash for Performance

- Over-Write 불가능(1->0가능, 0->1 불가능)
  - Write를 위해 Erase동작이 선행되어야 함
- Operation 단위의 차이
  - Page 단위의 write / read
  - Block 단위의 erase
- E/W cycling 제한
  - Wear-leveling 필요
- Bit Flipping 발생
  - ECC 처리 필요

NAND Parameter Trend



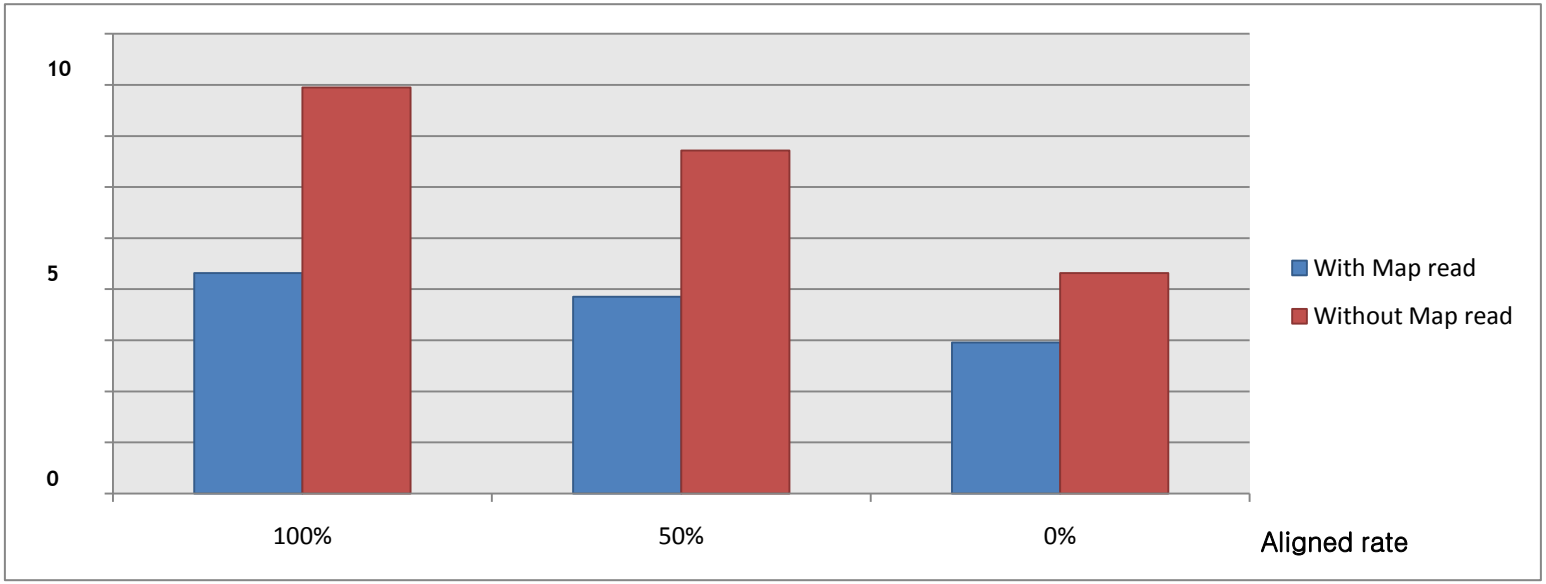
- Trend of storage device market
- Trend of NAND flash tech
- **How to increase random performance?**
  - **Software side**
  - Hardware side

Evolution of FTL Algorithm

- FTL은 Random Performance향상을 위해 진화 중!!
  - BAST(2002): Log Block과 Data Block이 1:1로 연관
  - FAST(2006): Log Block내의 Data Block에 대한 Page Mapping
  - SAST(2007): Log Block에 연관된 Data Block개수를 제한(N)
  - LAST(2009): Random Log Block을 Hot / Cold로 구분
  - KAST(2009): Log Block Association을 줄이기 위한 Block 할당
- 다음의 FTL은?

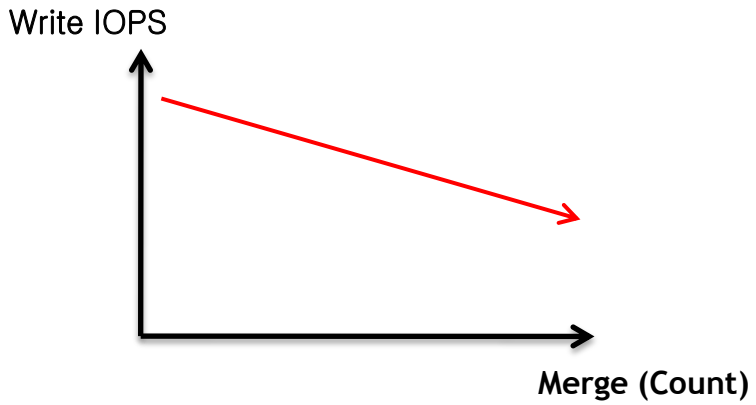
Random Read Issues

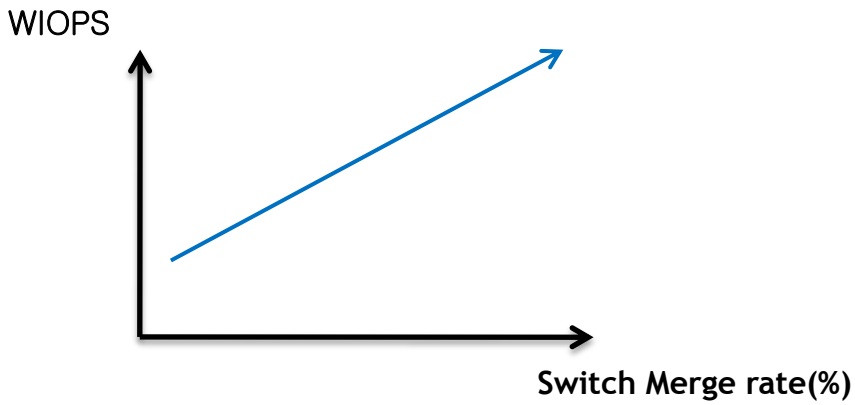
- Key Issues
  - Mis-aligned 상황 회피
  - Map Hit 확률 증대



Random Write Issues

- Key Issues
  - Merge 횟수 감소
  - Switch/Partial Merge 유도



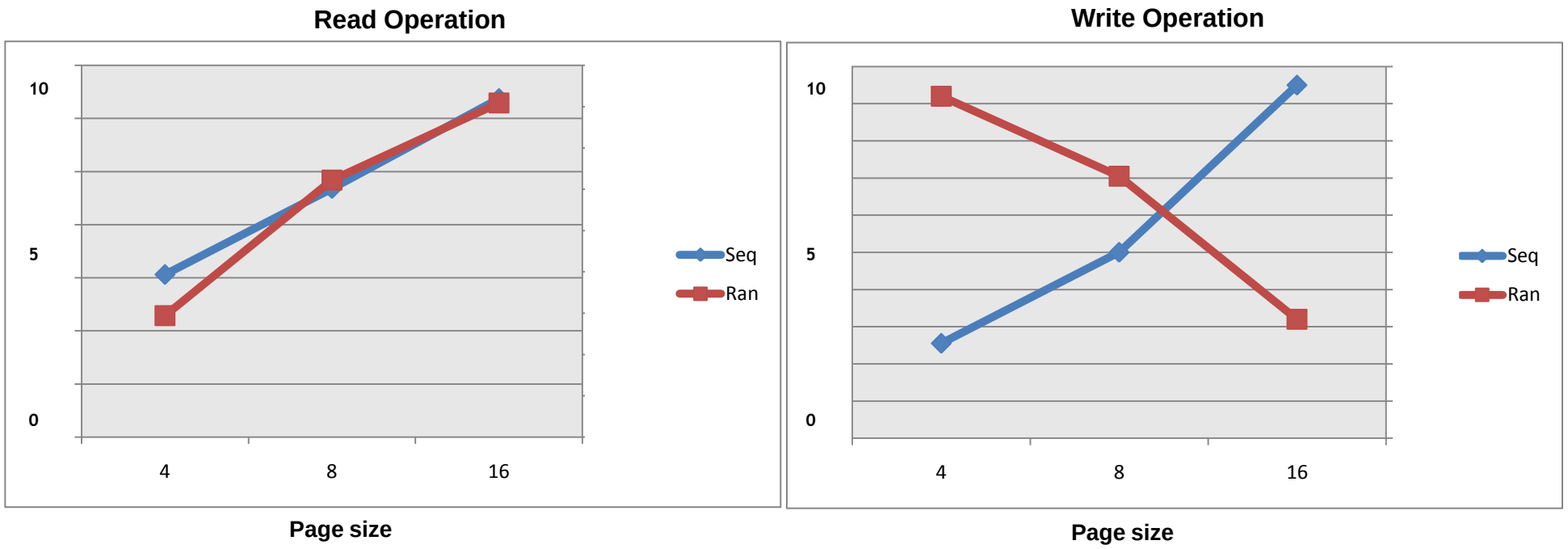


- Trend of storage device market
- Trend of NAND flash tech
- **How to increase random performance?**
  - Software side
  - **Hardware side**

Random 성능과 HW 구조

- **NAND parameter**
  - ✓ Page size
  - ✓ Pages per Block
- **Architecture**
  - ✓ Multi-Channel / Multi-Way
  - ✓ Cache Operation
  - ✓ Over Provisioning

Page size에 따른 성능변화

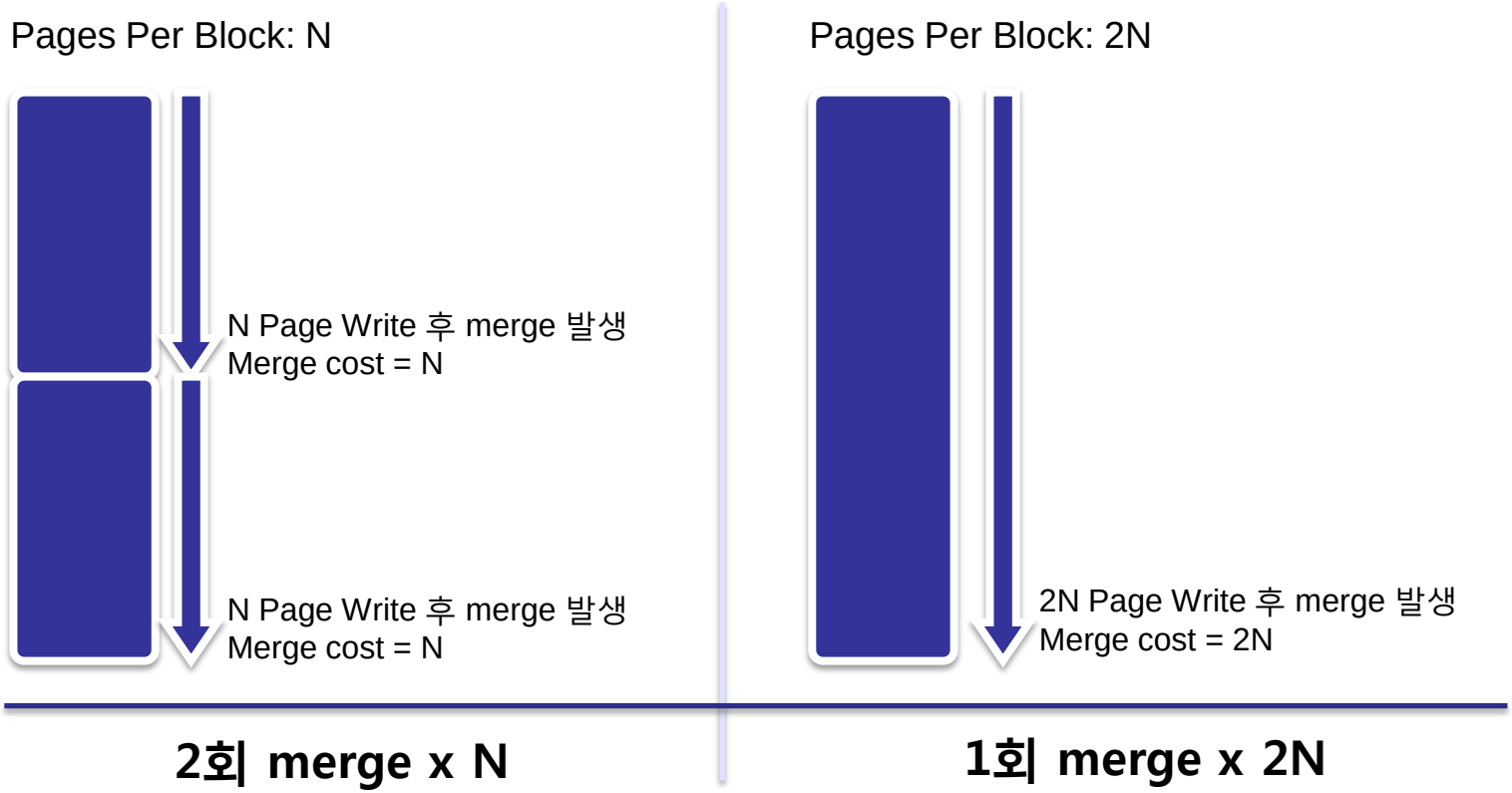


- Page size 증가
  - ✓ Sequential Throughput 증가
  - ✓ Random Read의 경우 Mis-align 감소로 성능향상
  - ✓ Random Write의 경우 Log block utilization문제로 성능저하 발생

Pages per Block에 따른 Random 성능

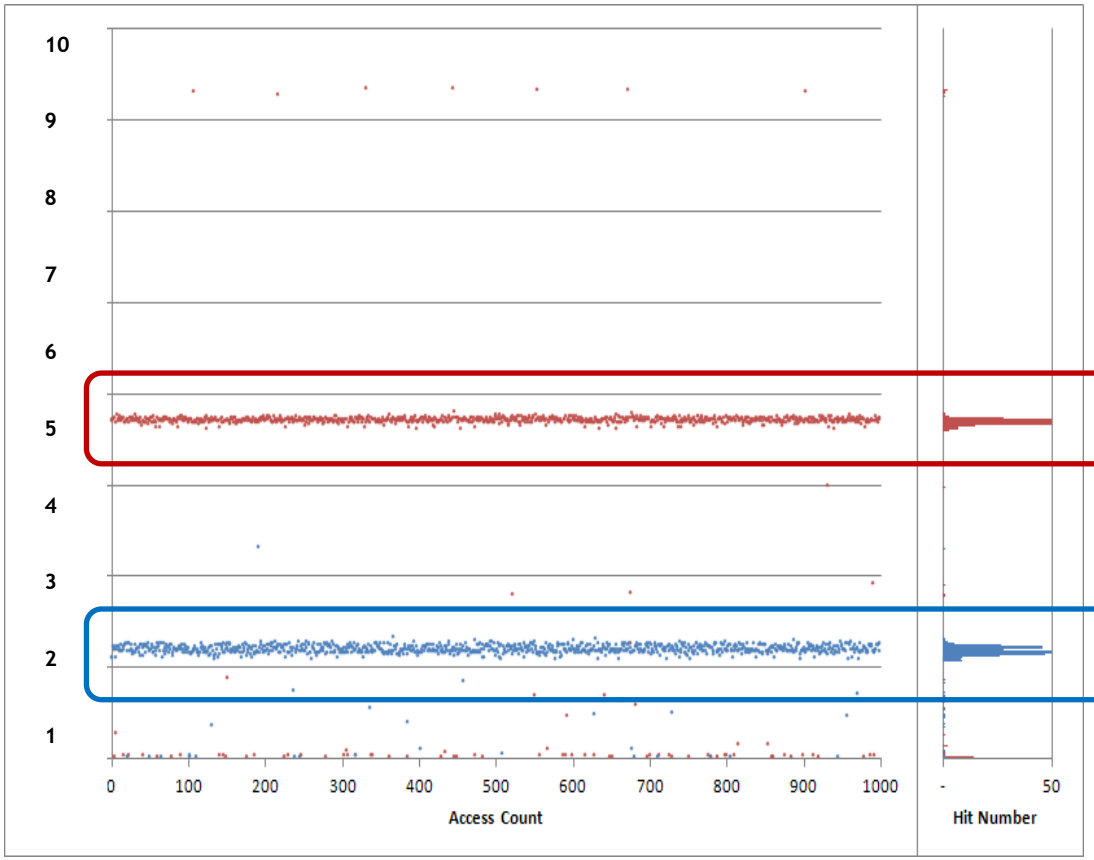
▪ **FTL개선으로 Random Performance 저하 최소화 가능**

Mapping 방식	성능 저하	비 고
Block mapping	50% 정도 성능 저하	저용량 메모리 사용
Hybrid mapping	0 ~ 5% 저하	eMMC controller
Paged mapping	성능 저하 요소 없음	SSD, Hybrid eMMC



Pages per Block에 따른 Write Latency Time

- Block copy에 소요되는 시간이 page개수 만큼 증가
- 이로 인한 응답시간은 상대적으로 증가!!
- 해결방안은?



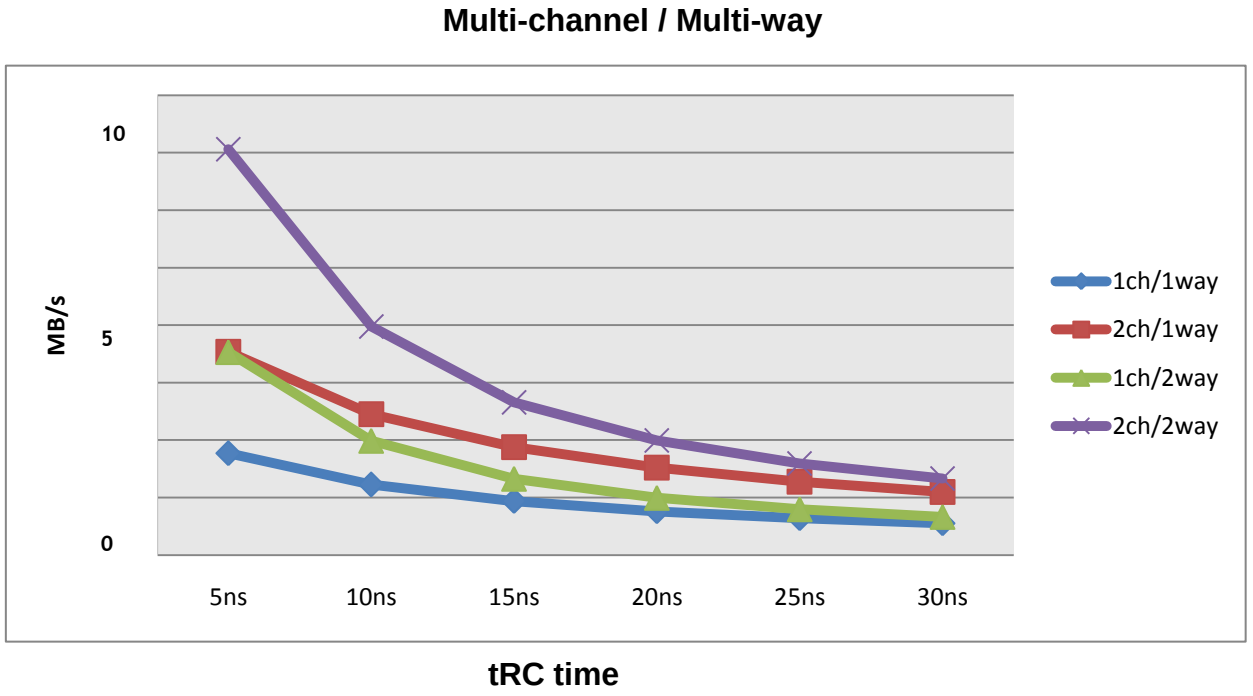
2N Pages Latency

N Pages Latency

Multi-channel / Multi-Way

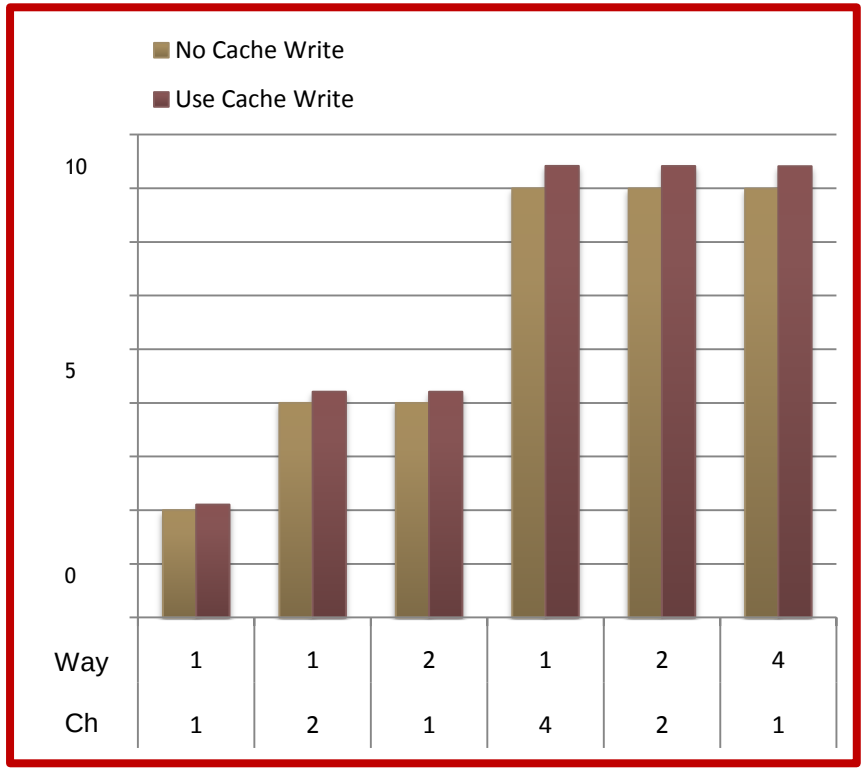
- Channel 이 증가하면 성능 향상
- Way가 증가해도 향상(Channel 이 우세)

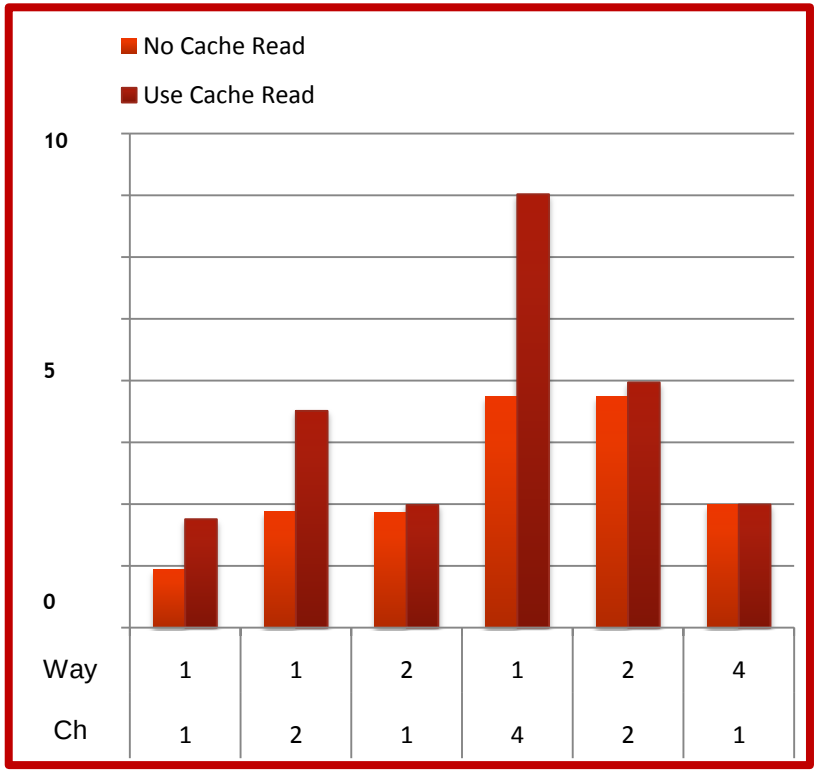
✓단, Embedded system에서 Current Consumption 문제로 병렬처리의 한계가 존재함



Cache Operation

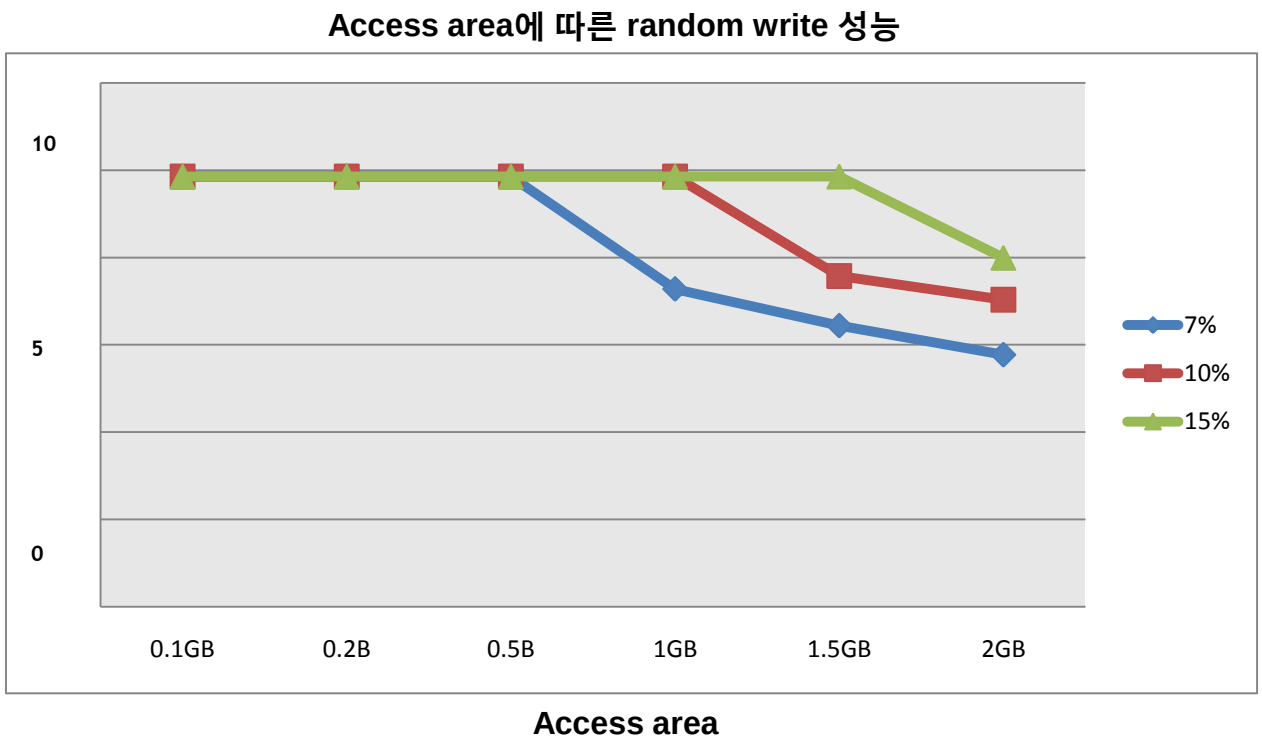
- Sequential Performance는 Cache Operation이 유용함
- Transfer Time의 비중이 큰 Read Operation에 두드러짐





Over provisioning과 access area 관계

Buffer 영역이 클수록 random performance 향상



Over provisioning 이 R.W 성능을 좌우한다!

Performance 구성 요소

구분		Items	비고
Random performance	HW	• tRC/tWC, eMMC clock • tRead • tProgram • tErase • DRAM size	Cost 증가
	SW	• Over provisioning • FTL Algorithm ➢ L2P map structure ➢ Buffer block management	Software algorithm의 복잡도 증가로 Over-head발생
Sequential performance	HW	• Plane, Channel, Way • SDRAM	Performance와 Power 간의 trade-off
	SW	• Parallelism ➢ Multi plane operation ➢ Multi channel operation ➢ interleaving operation	Software 복잡도 증가로 Over-head 발생

Q & A