Architecture Exploration of High-Performance PCs with a Solid-State Disk

D. Kim, K. Bang, E.-Y. Chung
School of EE, Yonsei University

S. Yoon
School of EE, Korea University

Outline

- Introduction
- Related Works
- Motivation
- The Proposed Techniques
- PC Architecture Exploration
- Experimental Results
- Conclusion and Future Work
- Summary

SSD – The Inevitable Tide

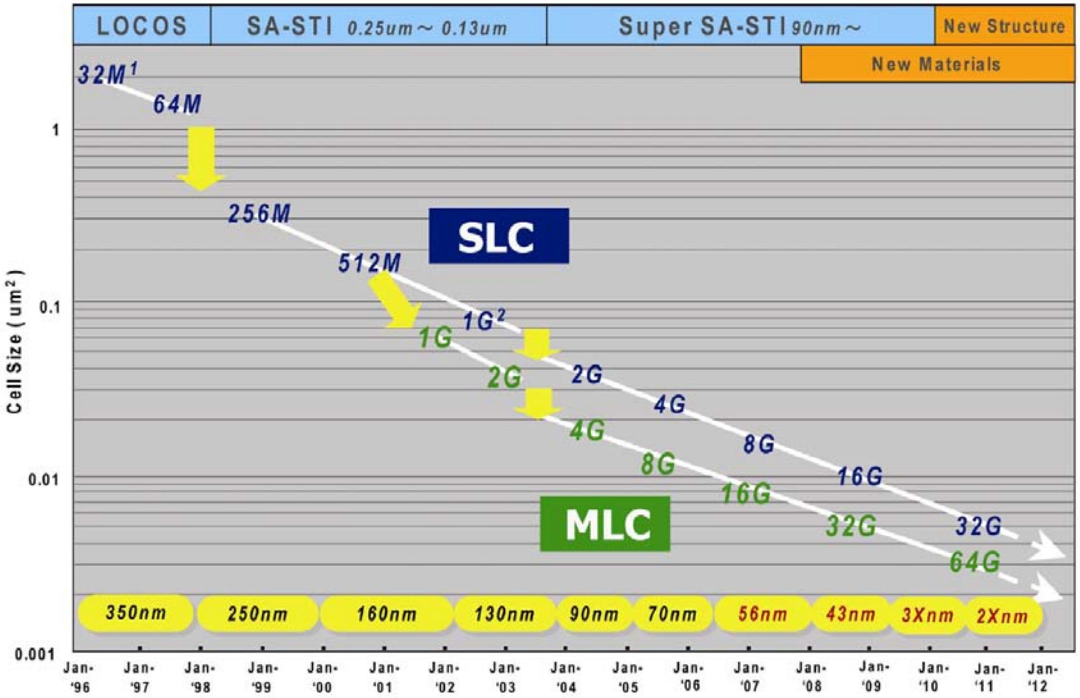
- HDD
  - Mass storage device for the last several decades
- SSD
  - An *electronic* storage device
  - Using non-volatile memory elements
  - High-performance
  - Small form factor
  - Light weight
  - Low power consumption
  - Shock resistance
  - Advantageous for harsh and rugged environment

From Extravagance to Necessity

- The only downside of SSD
  - The higher bit cost than HDD
  - Samsung's 256GB SSD is as much as 860,000 won
  - Seagate's 250GB HDD is just 44,000 won
- The increasing density of NAND flash memory
  - It becomes double every 12 months
  - The price gap keeps narrowing and narrowing…
  - Eventually, it will become negligible in 2012
    - Forecast by IDC (International Data Corporation)

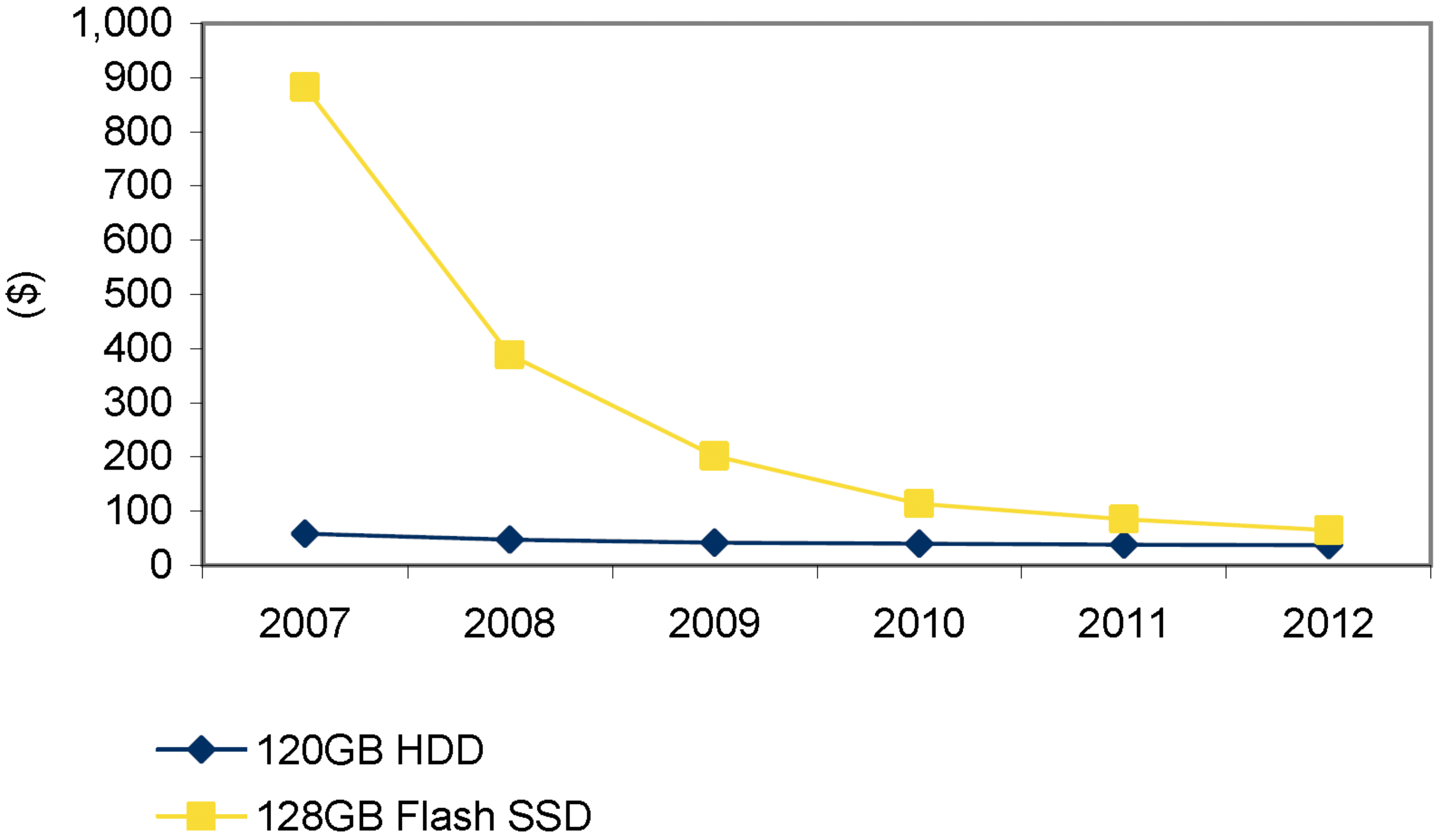
The Increasing Density of NAND

- Multi Level Cell
  - SLC → **DLC** → TLC → QLC ...
- 3D Stacking



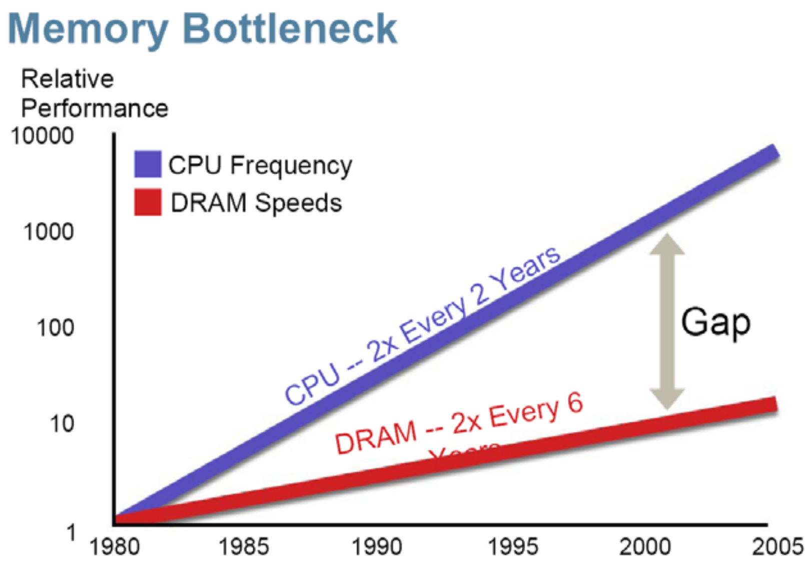
Source: Toshiba 2008

Average Selling Price Comparison

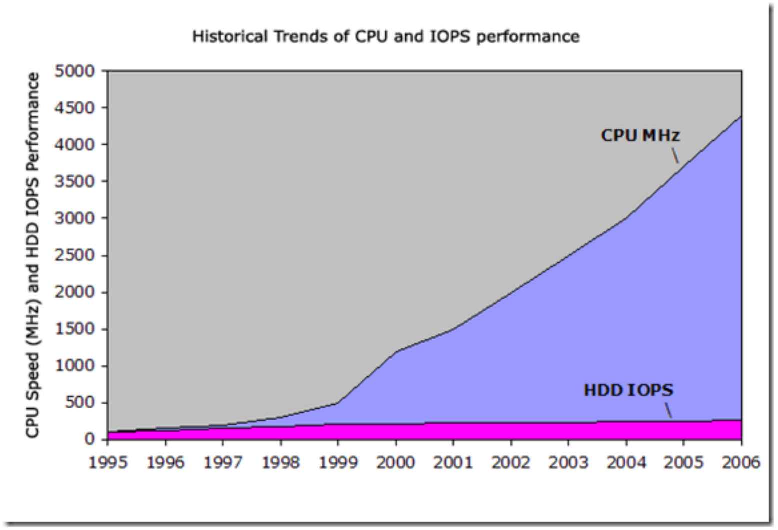


Source : IDC 2008

CPU / Memory Performance Gap



Source: SUN Microsystems 2007



Source: MSDN 2009

- Multi / many-core processors enlarge the gap
  - Intel dual core / quad core ...
  - Nvidia CUDA ...
  - ARM Cortex ...
- High-performance SSD is Strongly Required!

Outline

- Introduction
- Related Works
- Motivation
- The Proposed Techniques
- PC Architecture Exploration
- Experimental Results
- Conclusion and Future Work
- Summary

SSD Internals –

SSD Internals – Memory Hierarchy

- Smart buffer cache [Lee et al. 05]
  - Enhanced exploitation of spatial / temporal locality
  - High performance and low power consumption
- Energy-aware demand paging [Park et al. 04]
  - Minimizes the number of write or erase operations

SSD Internals – Hybrid Systems

- SLC / MLC hybrid SSD [Chang et al. 08]
  - Trade-off performance and cost
  - SLC as a cache block
- FRAM / NAND hybrid SSD [Yoon et al. 07]
  - Meta-data is maintained in a small FRAM
  - Exploiting non-volatility of FRAM
- PRAM / NAND hybrid SSD [Kim et al. 08]
  - PRAM is used for meta-data
  - Firstly under mass production among universal RAMs

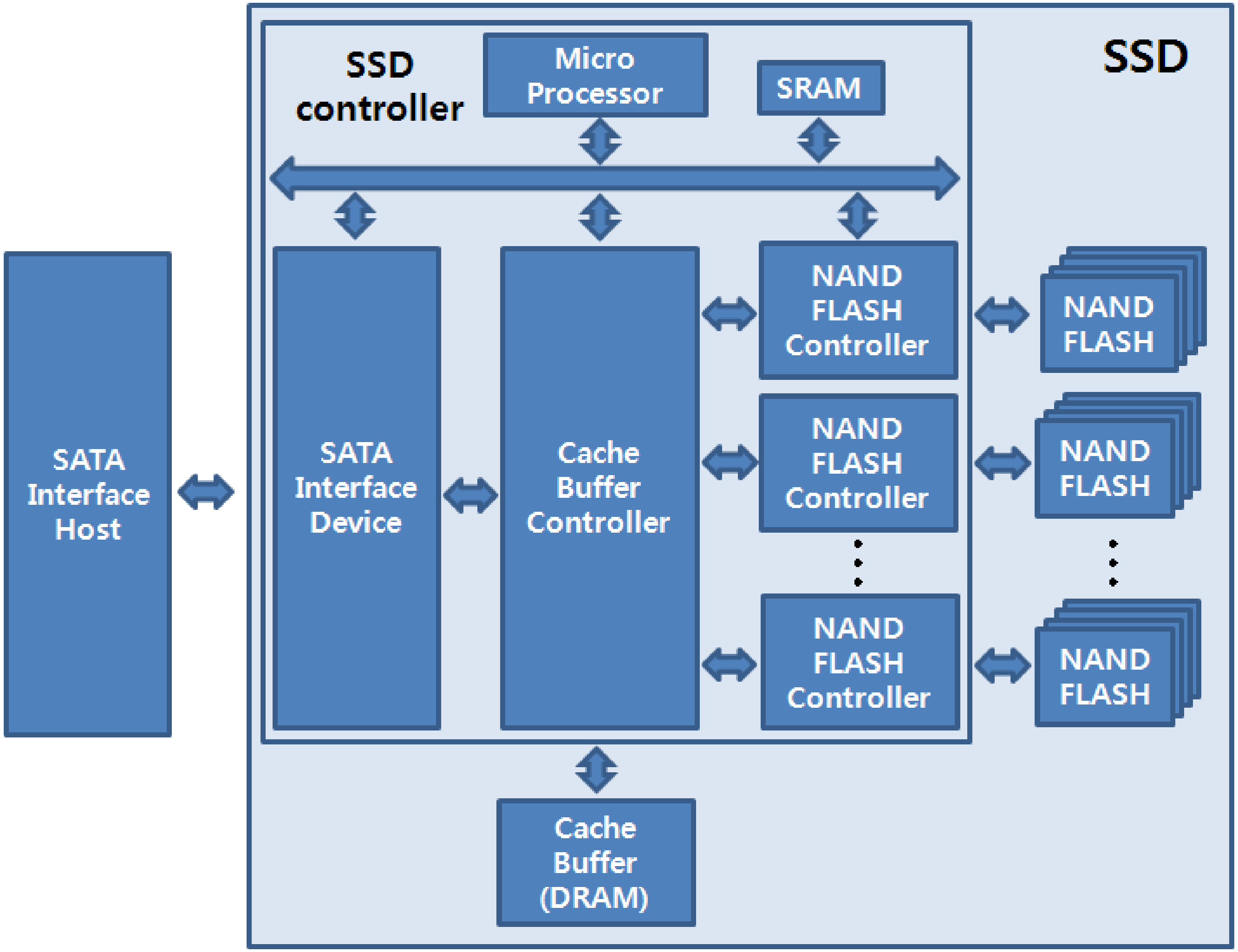
Interaction between Host and SSD

- Robson Architecture by Intel
  - A non-volatile memory layer as a cache for disk
  - Reduces data transfer time and power consumption
- PCIe SSD by FusionIO
  - Resolves bottleneck due to traditional slow I/F
  - Much higher bandwidth (520MB/s)
- NAND-based storage nodes [Lee et al. 08]
  - Several thousands of nodes to build clusters
  - Plugged into ethernet-style backplane

Outline

- Introduction
- Related Works
- Motivation
- The Proposed Techniques
- PC Architecture Exploration
- Experimental Results
- Conclusion and Future Work
- Summary

Traditional Architecture of SSD



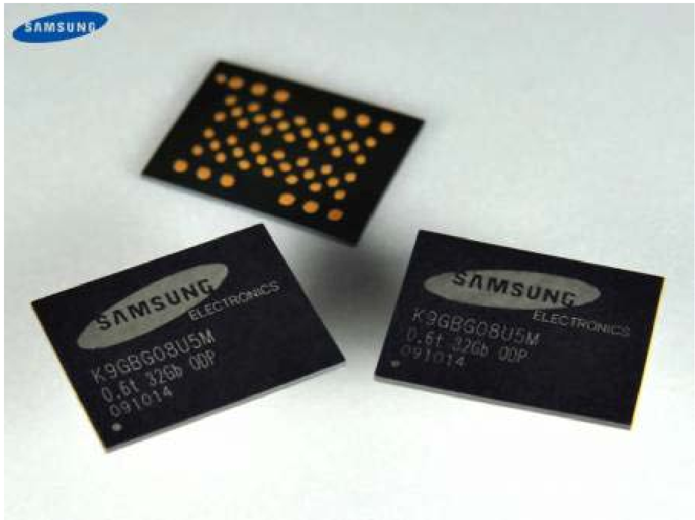
SSD Benchmark Chart

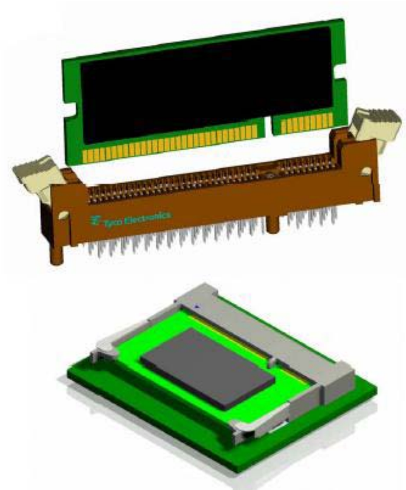
Drive	Read	Write
Intel X-25E	250MB/s	170MB/s
Kingston E	250MB/s	170MB/s
Intel X-25M	250MB/s	70MB/s
Kingston M	250MB/s	70MB/s
OCZ Apex	230MB/s	160MB/s
G.Skill Titan	230MB/s	160MB/s
OCZ Vertex	200MB/s	160MB/s
Patriot Warp V2	175MB/s	100MB/s
OCZ Core V2	170MB/s	100MB/s
G.Skill FM	155MB/s	90MB/s
OCZ Solid	155MB/s	90MB/s
RiData CO4MPN	152MB/s	96MB/s
SuperTalent Masterdrive OX	150MB/s	100MB/s
Transcend TS	145MB/s	92MB/s
RiData CO3M	118MB/s	74MB/s
OCZ SSD	100MB/s	80MB/s
G.Skill FS	100MB/s	80MB/s
Samsung SSD	100MB/s	80MB/s

Source: www.ssdbenchmark.com 2009

Asynchronous DDR NAND Flash

- The most crucial bottleneck of SSD is the performance of NAND flash device
- New DDR type NAND flash devices offer tremendous performance improvements
- Toggle-mode NAND from Samsung & Denali
- ONFi from Hynix, Intel, Micron, etc.

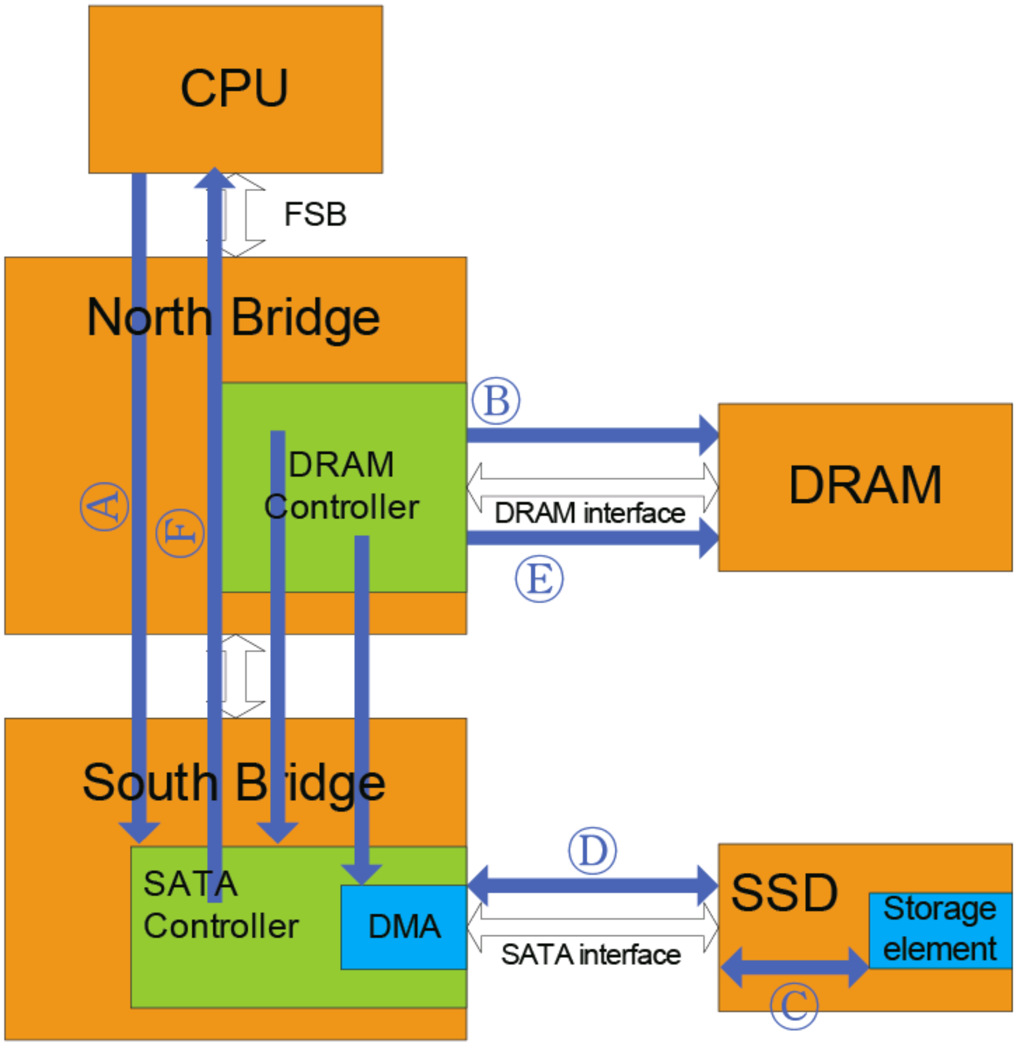




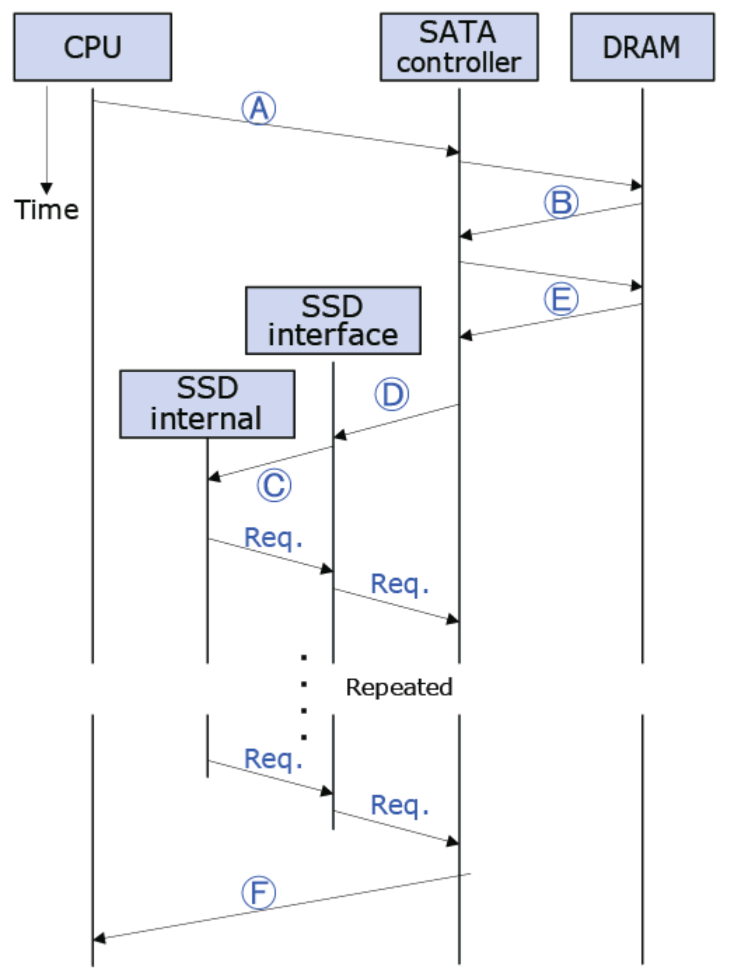
The 3rd Generation High-speed I/F

- SSDs are close to saturating the SATA 2.0
  - 3 Gbit/s (300 MB/s) limit
- SATA 3 / USB 3 / PCIe 3
  - SATA 3.0 will offer 6 Gbit/s (600MB/s)
  - USB 3.0 SuperSpeed will provides 4.8 Gbit/s (572MB/s)
  - PCIe 3.0 will add a Gen3-signalling mode, at 1 GB/s
- DDR 3
  - DDR3-1600 shows 12.5GB/s by 64-bit data width

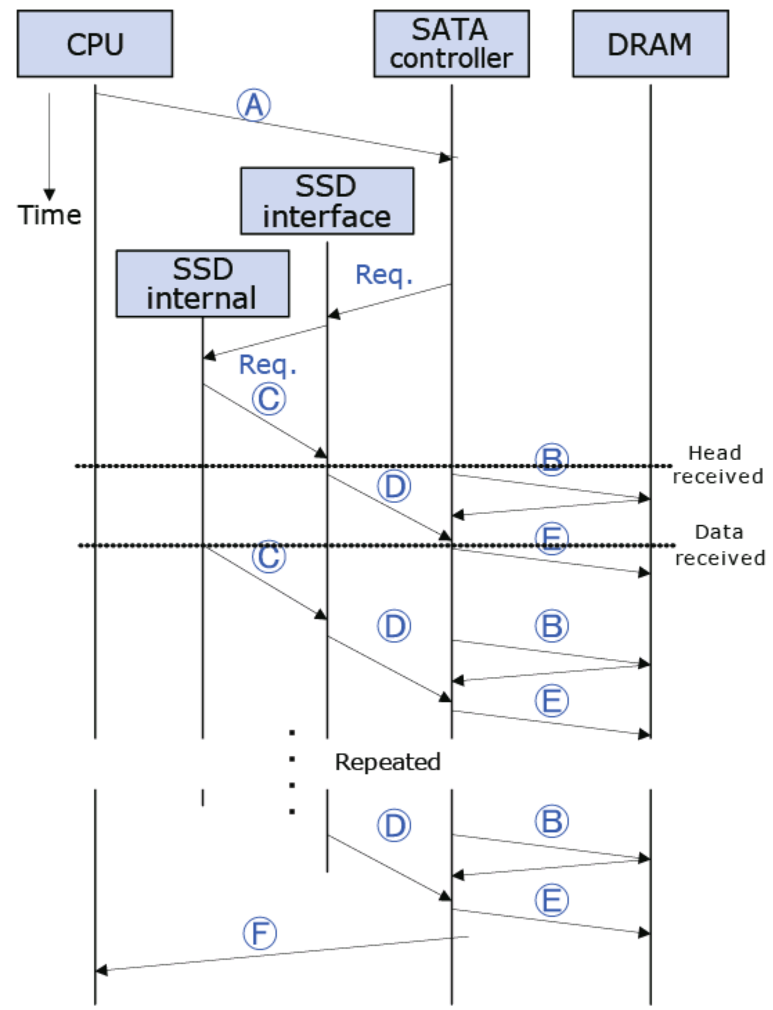
Conventional PC Architecture



DMA in Conventional PC



DMA write (consecutive)



DMA read (pipelined)

Limitations of Conventional PC / SSD

- Simply replaces a HDD
  - The same interface protocol for compatibility
  - Maximum bandwidth of ATA is only 133 ~ 300 MB/s
  - May become a bottleneck in the near future
- Data go through both North & South bridges
  - A single data request must be arbitrated twice
  - Both bridges are not designated for SSD
  - SSD is connected together with slow peripherals
- Page fault transfer must be serialized
  - DMA read for a new page must wait until completion of DMA write for a victim page

Architecture Exploration Aspects

- Host interface scheme
  - Location of SSD in PC
    - From the conventional south bridge to north bridge
  - Interface Protocol
    - Using huge bandwidth of DDR offered by north bridge
    - DDR 2 is widely used in PC at the time
- Data transfer concurrency
  - Minimize the conflicts between CPU-to-Mem and Mem-to-SSD
  - A dual port DRAM and a dual port SSD

Outline

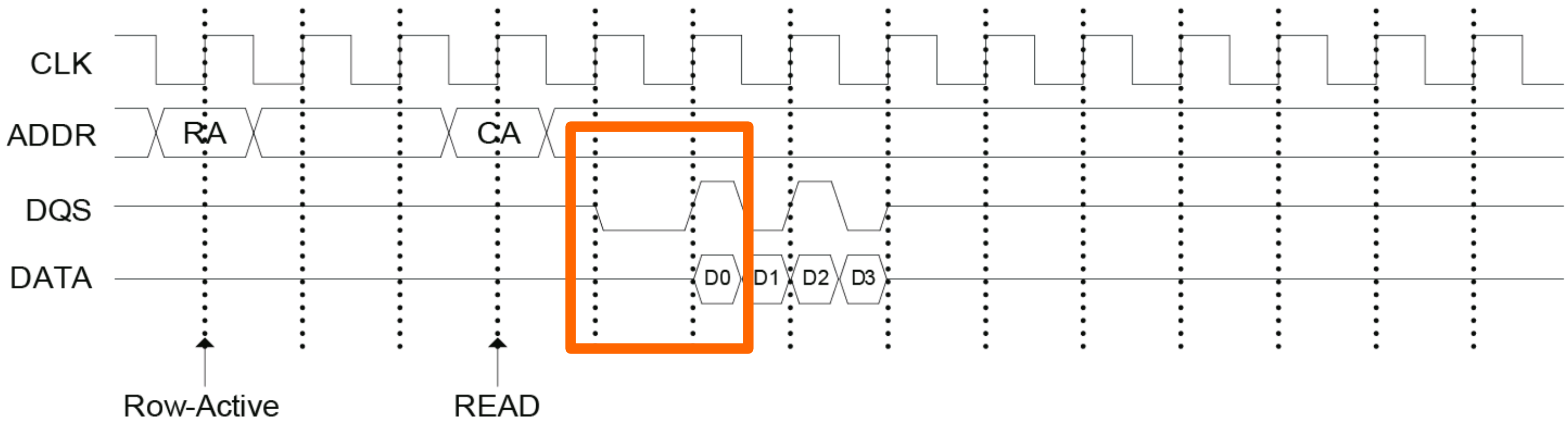
- Introduction
- Related Works
- Motivation
- The Proposed Techniques
- PC Architecture Exploration
- Experimental Results
- Conclusion and Future Work
- Summary

SSD with DDR DRAM Interface

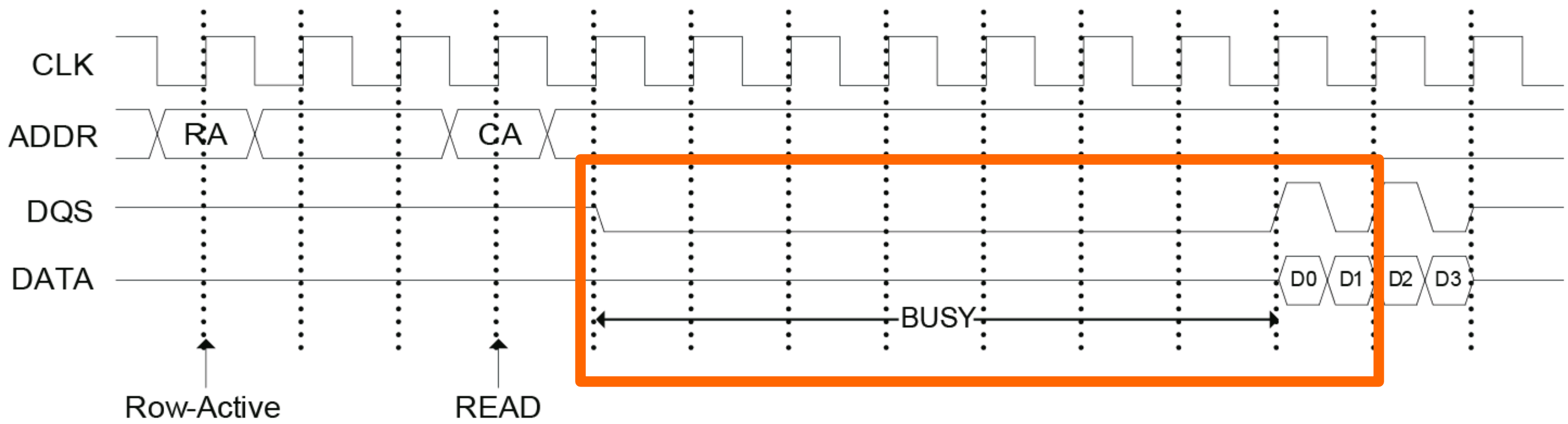
- The fastest DDR DRAM interface
  - 64-bit bus width is widely used
  - Using both rising and falling edge for data transfer
  - High frequency for communication with processors operating with several GHz

- Fixed access latency
  - Latencies such as Column Address Strobe are fixed
  - SSD cannot guarantee internal maximum latency
    - Cache buffer hit vs. cache buffer miss
  - Needs a signal for data readiness
  - DQS pin is used to support arbitrary latencies

Timing Diagram of DDR I/F SSD

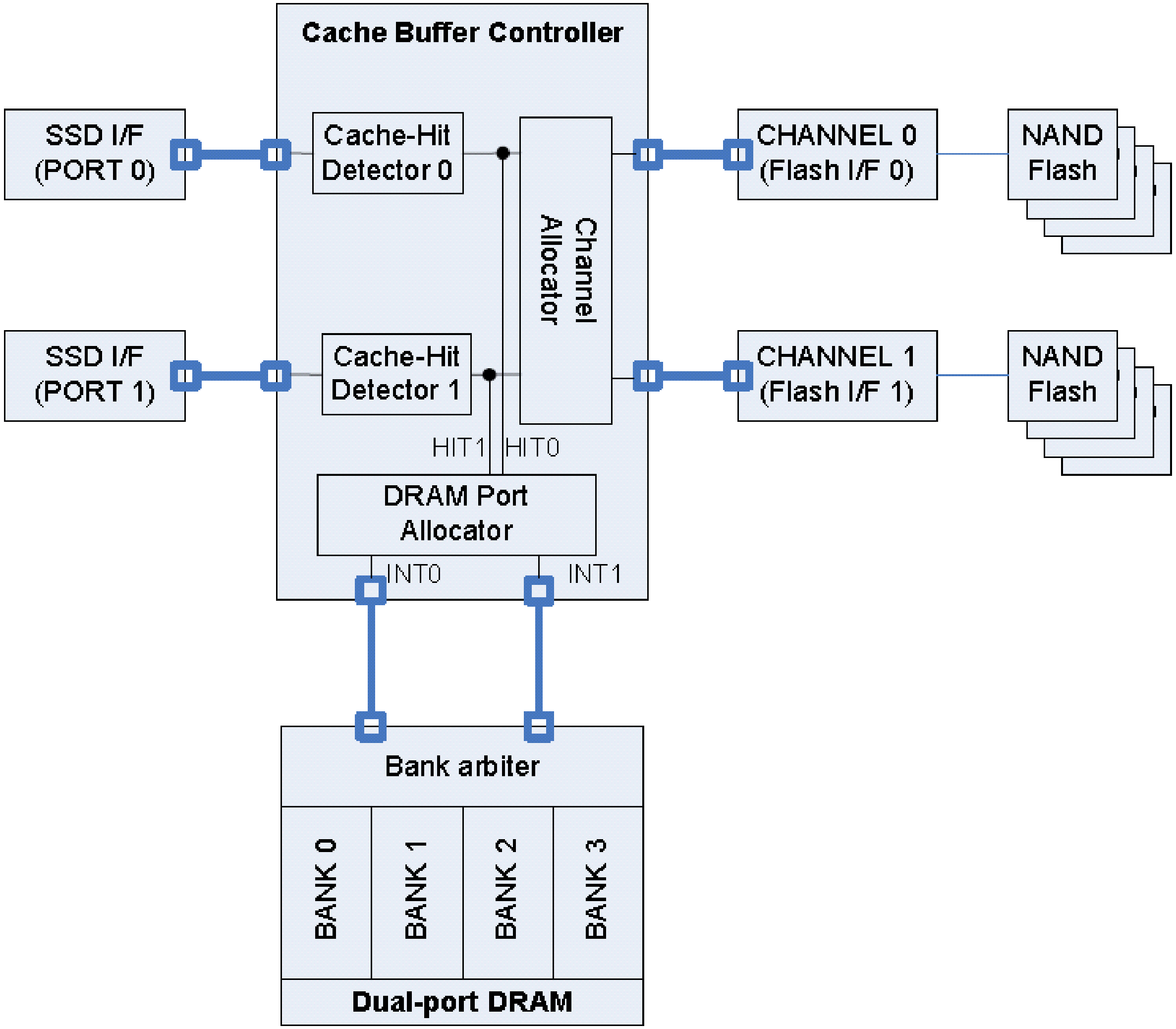


Cache buffer hit

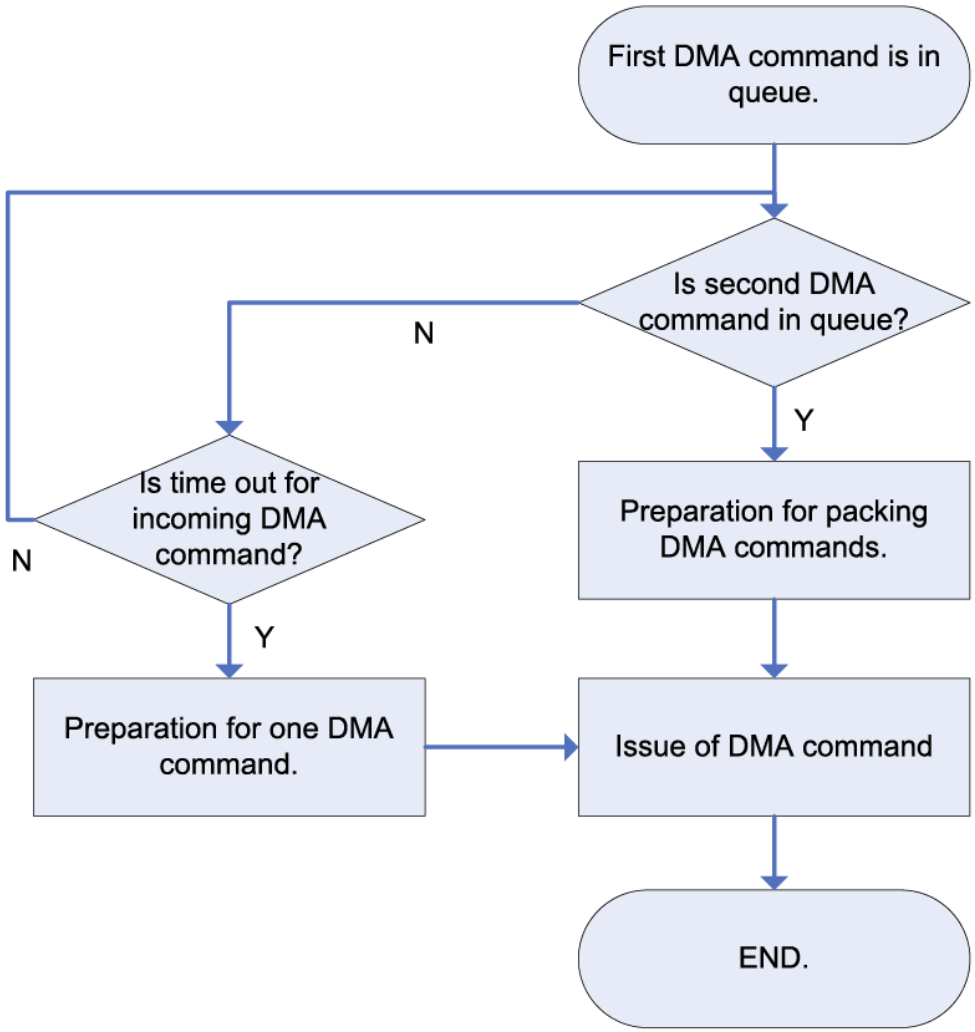


Cache buffer miss

Dual Port SSD Internals



DMA Command Pack for Dual DMA



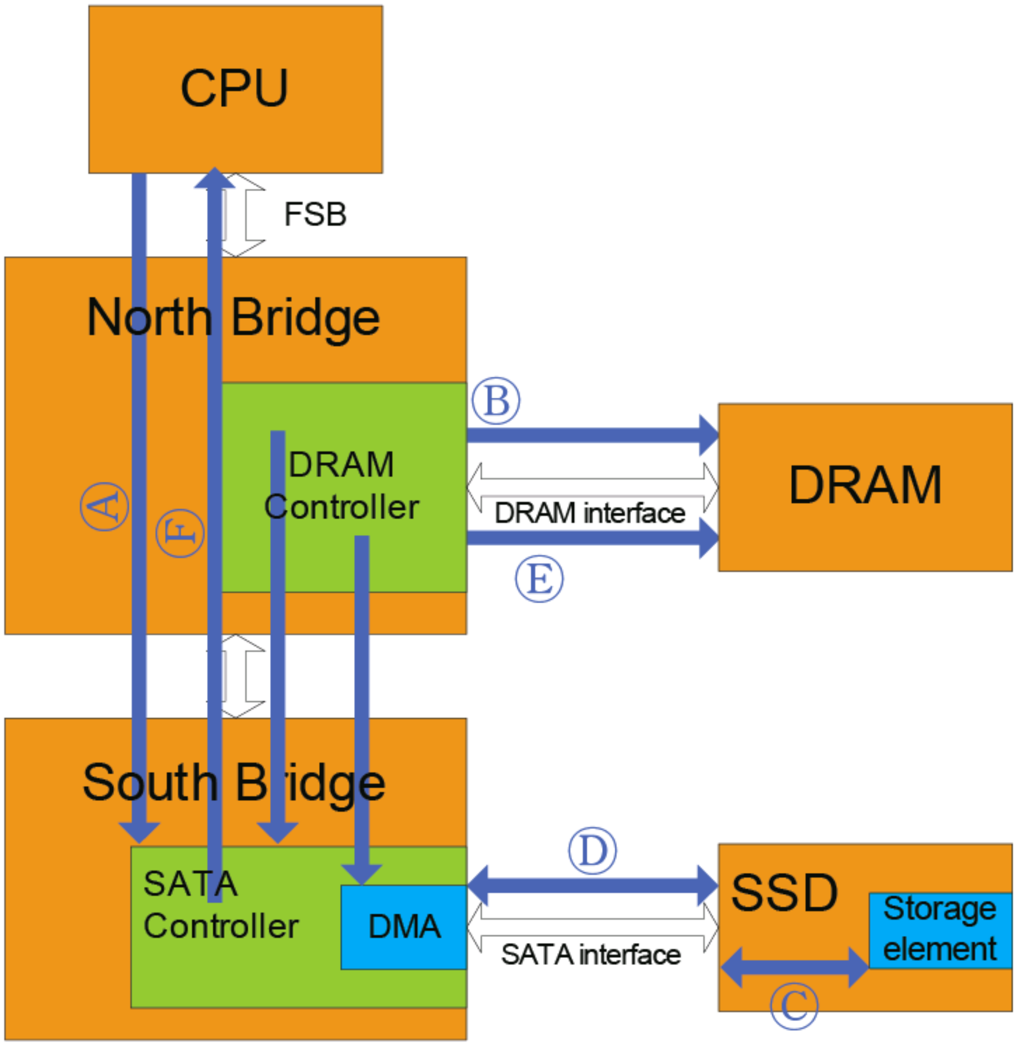
Outline

- Introduction
- Related Works
- Motivation
- The Proposed Techniques
- PC Architecture Exploration
- Experimental Results
- Conclusion and Future Work
- Summary

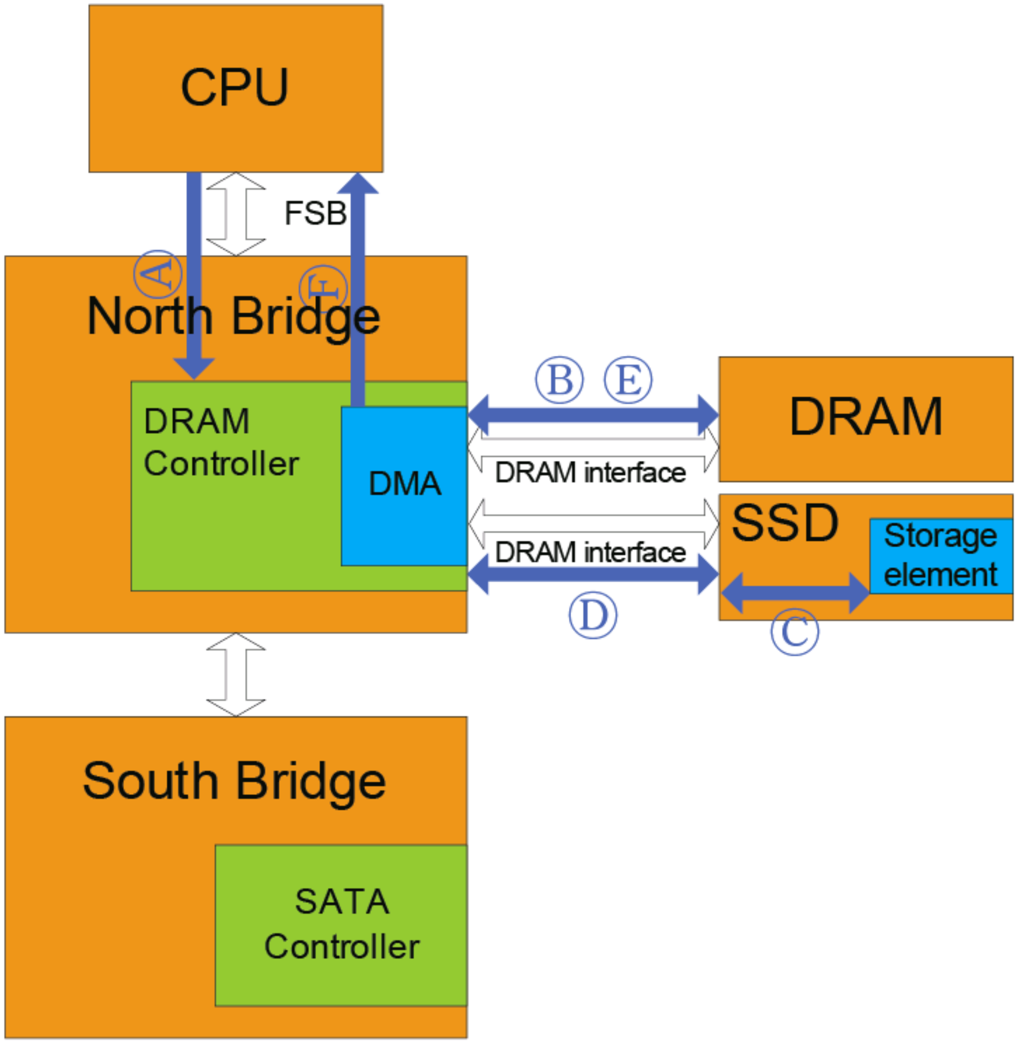
4 Architectural Choices

Direct Path	Location of the SSD	
	South Bridge	North Bridge
No	SBSP Architecture Conventional Architecture	NBSP Architecture 1. DMA in DRAM controller in NB 2. DQS scheme supported DRAM controller in NB
Yes	SBDP Architecture 1. Dual-port DRAM supported DRAM controller in NB 2. DMA command packing supported OS	NBDP Architecture 1. DMA in DRAM controller in NB 2. DQS scheme supported DRAM controller in NB 3. Dual-port DRAM supported DRAM controller in NB 4. DMA command packing supported OS

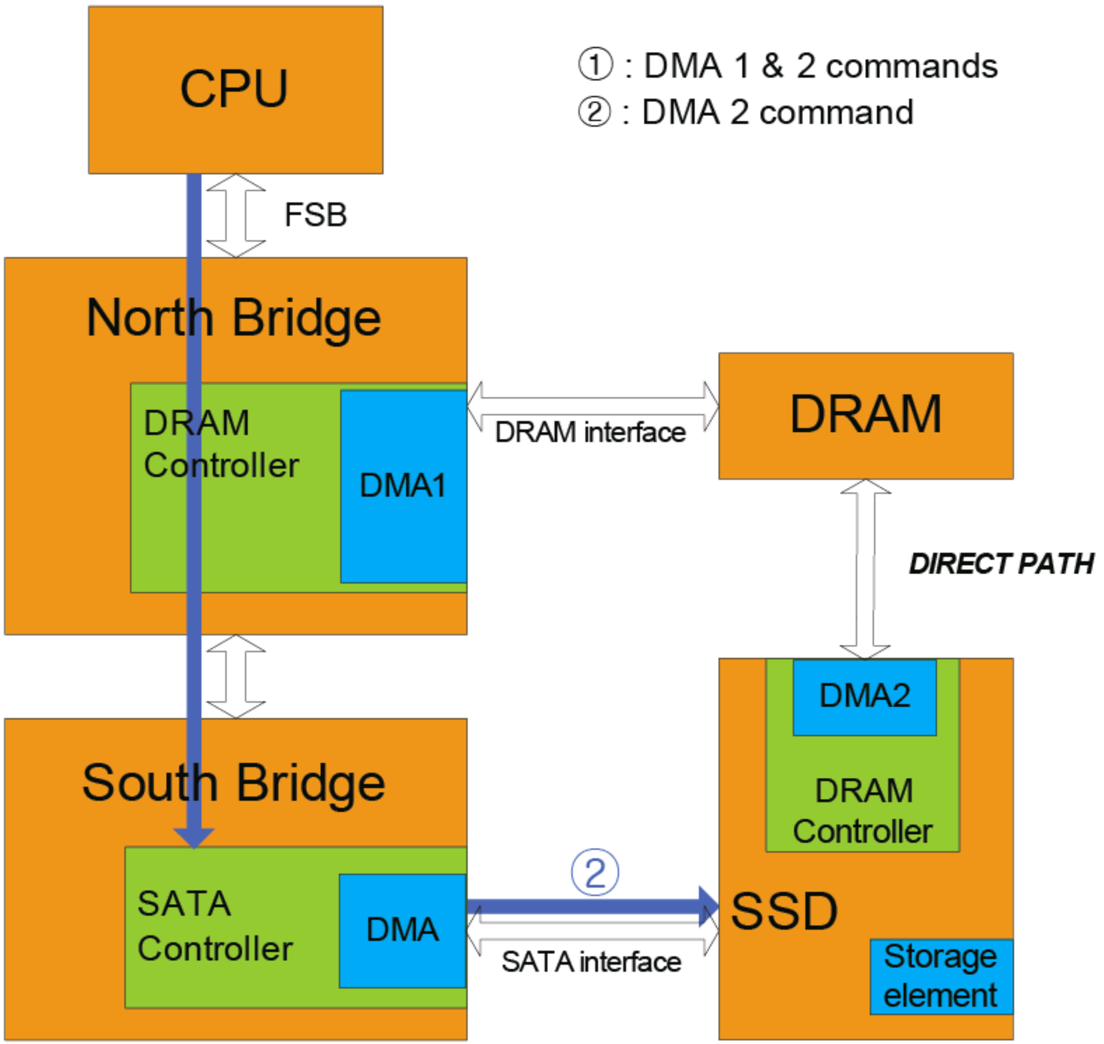
Conventional SBSP Architecture



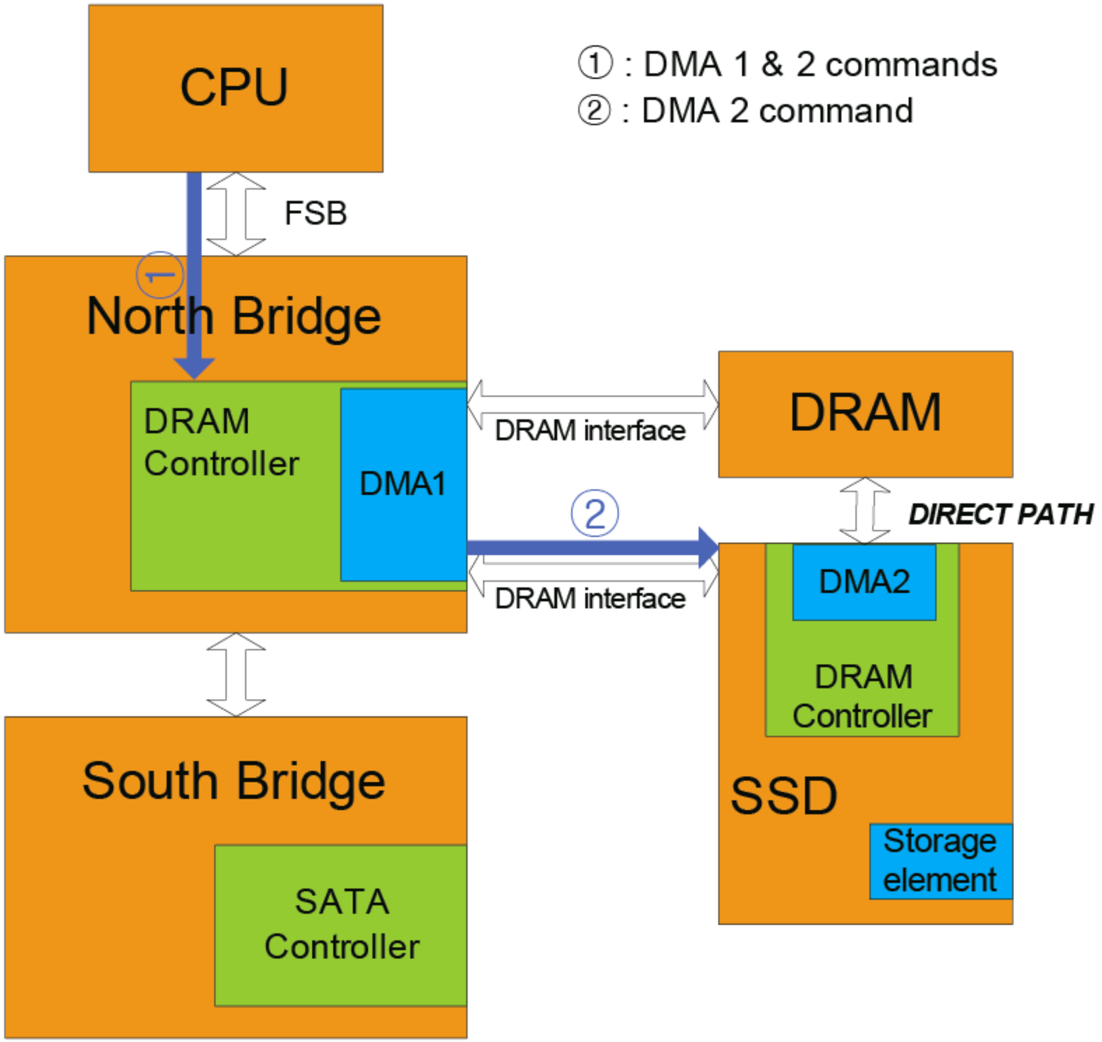
NBSP Architecture



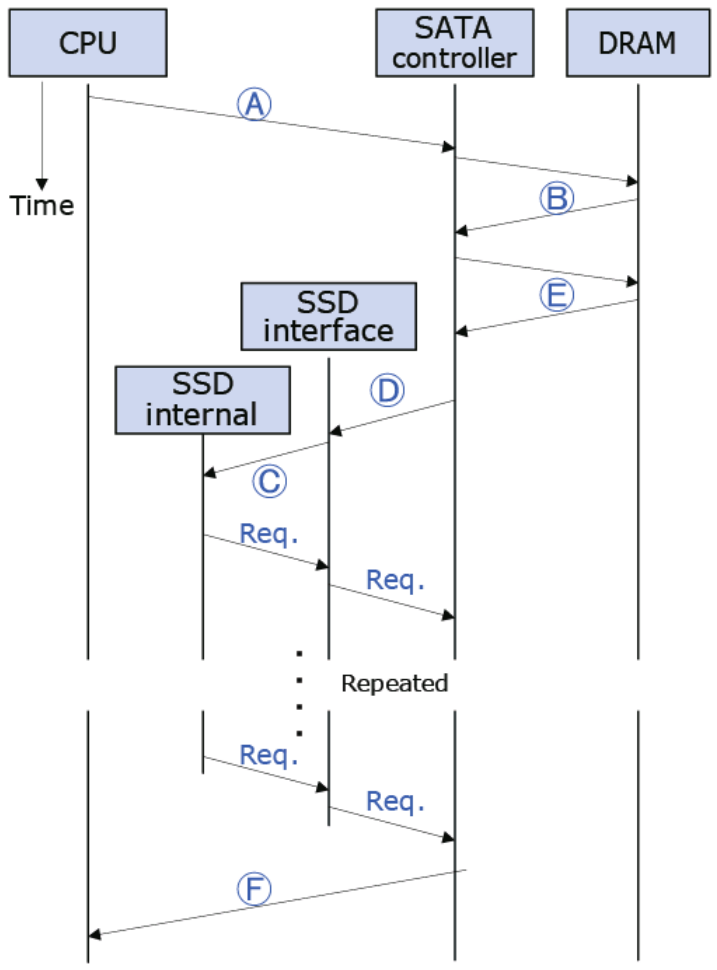
SBDP Architecture



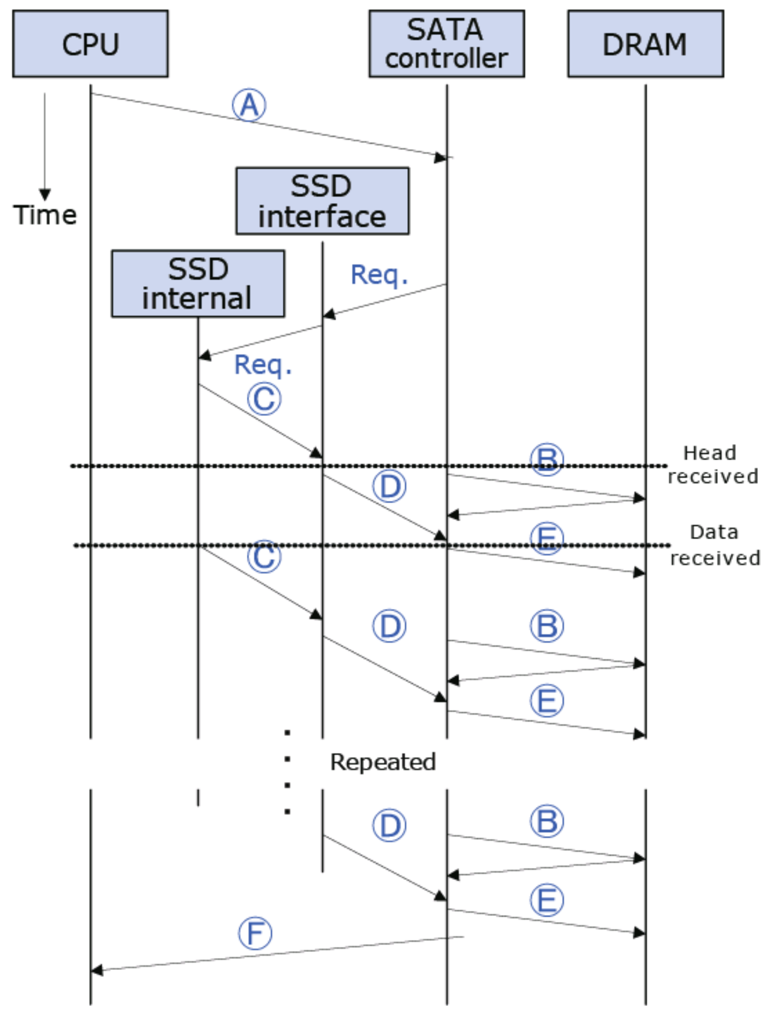
NBDP Architecture



DMA in Conventional SSD (Revisit)

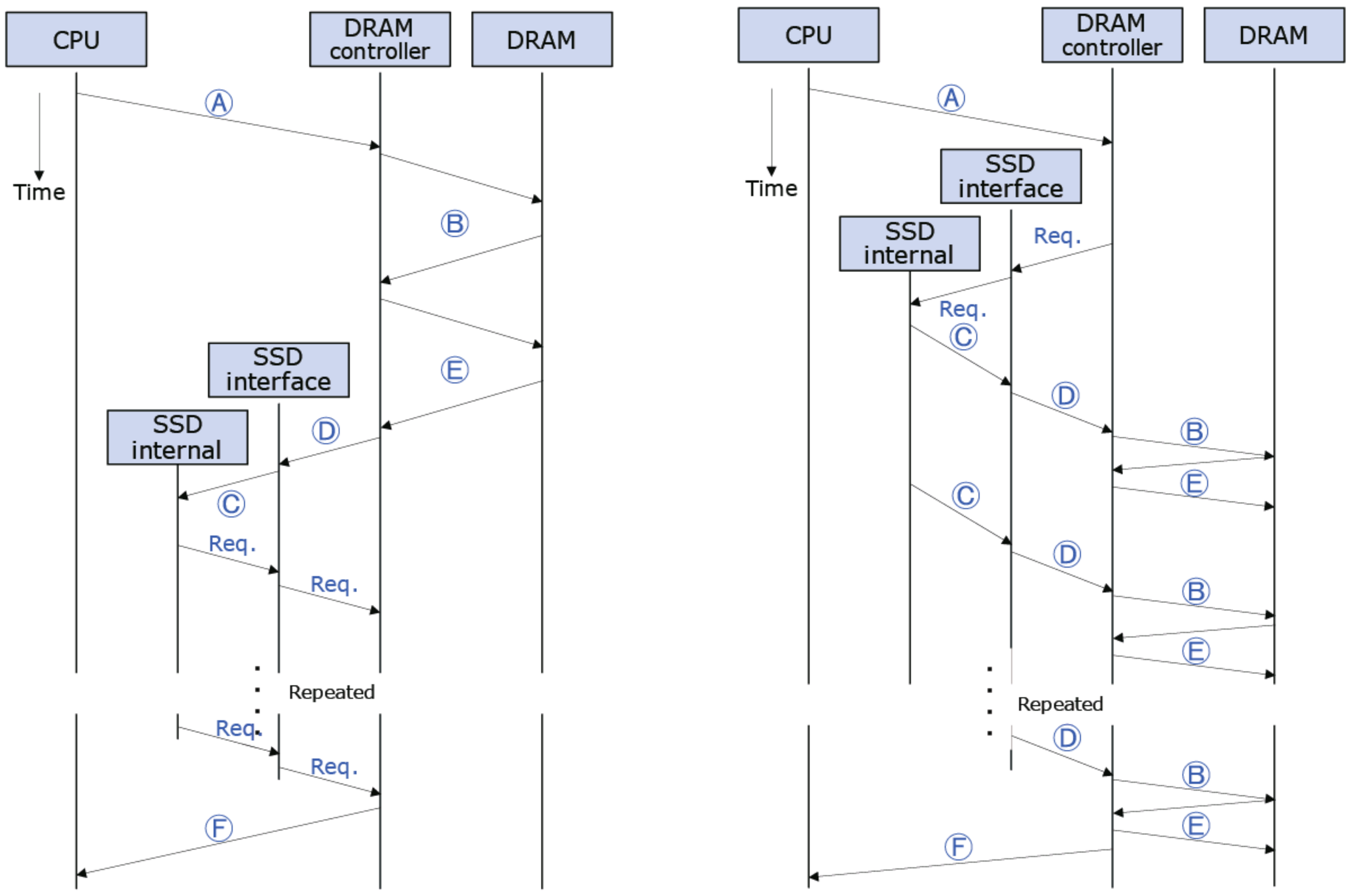


DMA write (consecutive)



DMA read (pipelined)

DMA in Dual Port SSD



DMA write (consecutive) **DMA read (pipelined)**

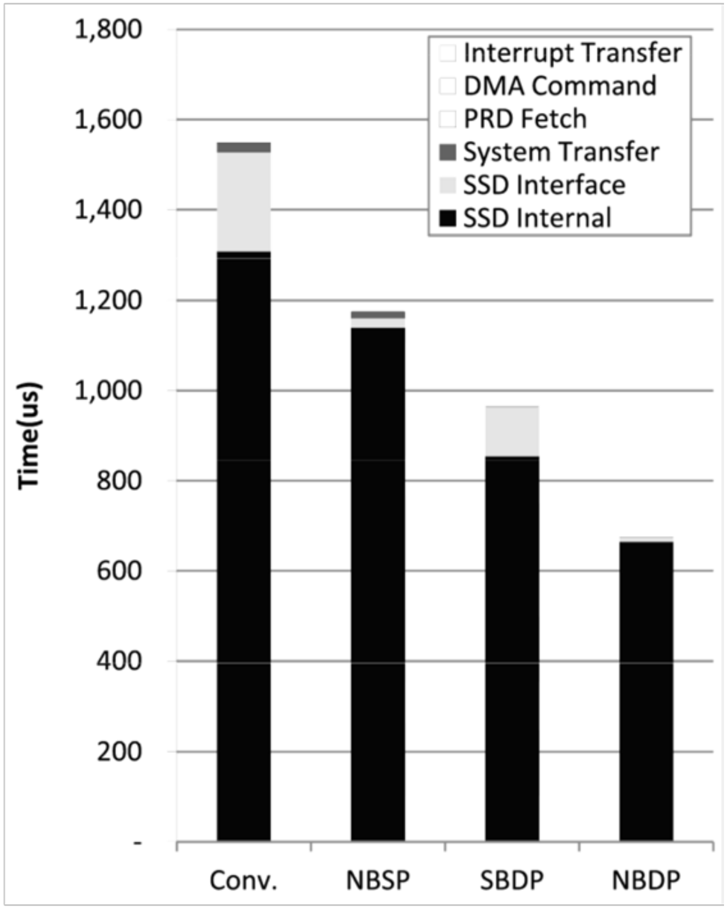
Outline

- Introduction
- Related Works
- Motivation
- The Proposed Techniques
- PC Architecture Exploration
- Experimental Results
- Conclusion and Future Work
- Summary

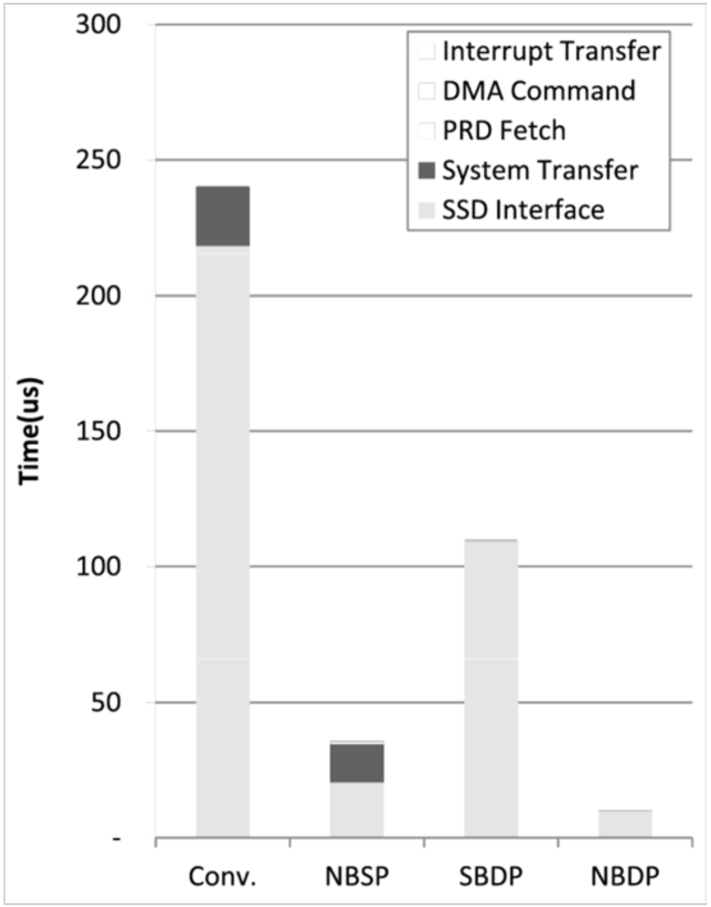
Experimental Setup

- Modeling transaction-level PC architecture
  - Using high-level SystemC language
  - Cycle accurate modeling, especially SATA, DDR, and PCIe protocols
- The main specification
  - Based on Intel's 965 chipset for North Bridge and ICH8 for South Bridge
  - Bridge internals and externals are linked by PCIe
  - DDR2-800 (6.4GB/s) is selected for NB
  - All other peripherals are not considered except CPU, NB, SB, DRAM, and SSD
    - E.g. Graphics, Sound, Mouse, Keyboard, etc.

DMA Read with Cache Miss



Total Time



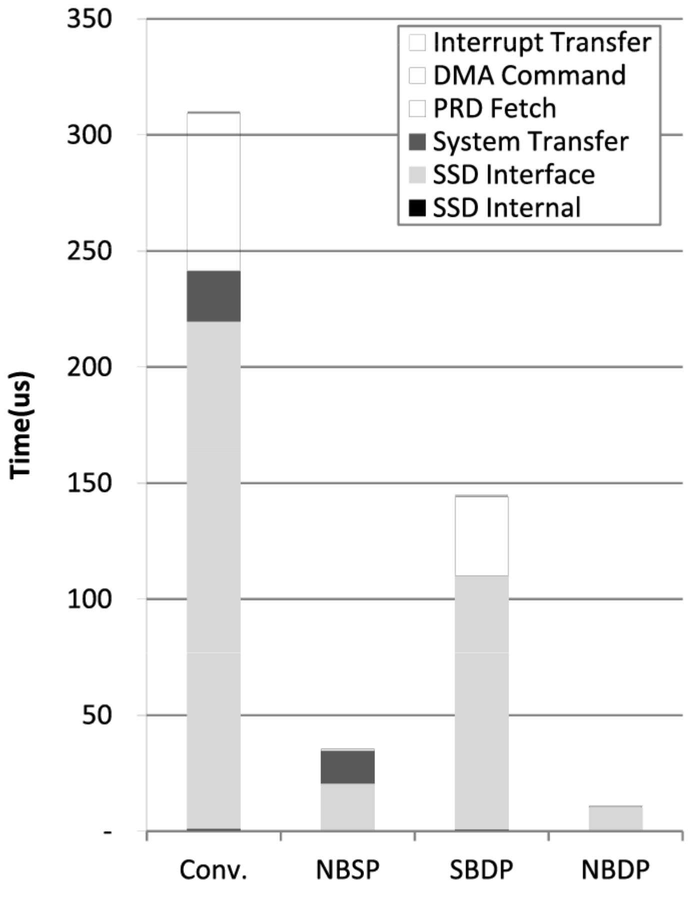
Excluding SSD internal

Contribution Breakdown in $ Miss

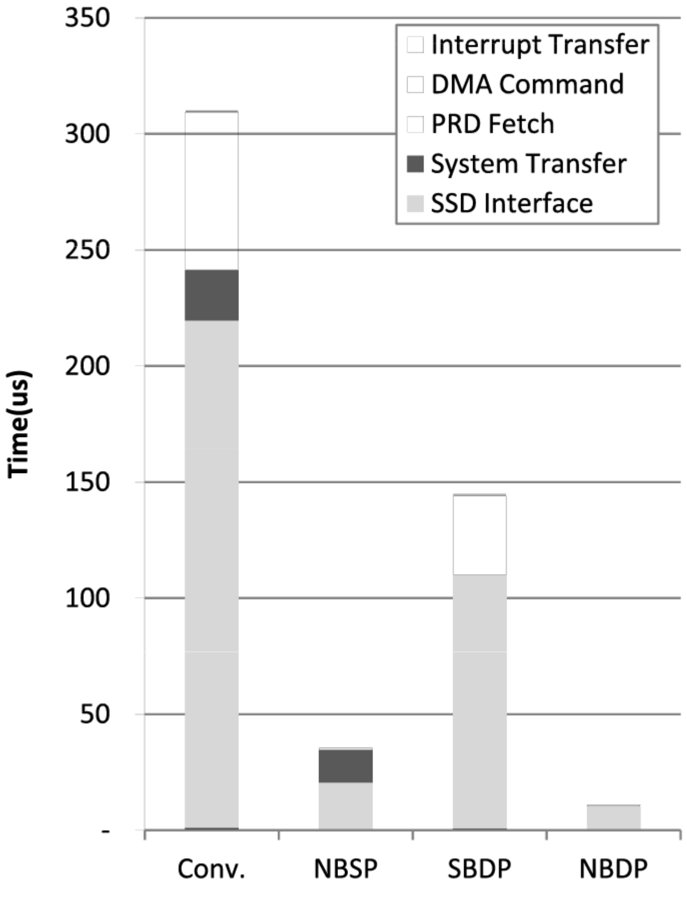
	SB elimination	DRAM I/F	Direct path
NBSP	1.89%	98.11%	–
NBDP	2.52%	11.85%	85.85%

- Improvement by SB elimination is marginal
  - The reduction of transfer time is hidden by long SSD internal latency
- DRAM interface is dominant for NBSP
- Direct path is dominant for NBDP

DMA Read with Cache Hit



Total Time



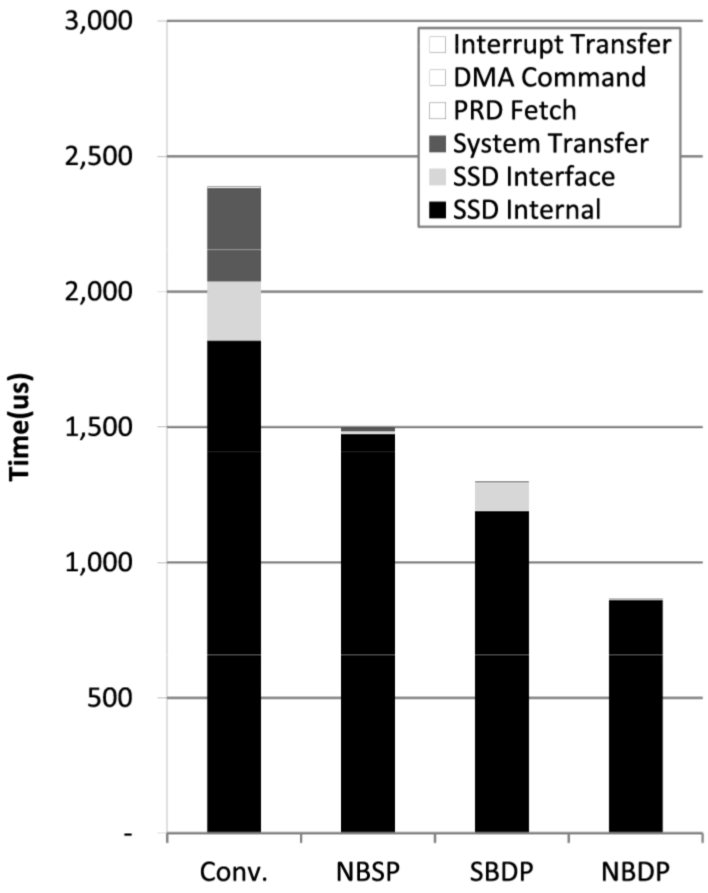
Excluding SSD internal

Contribution Breakdown in $ Hit

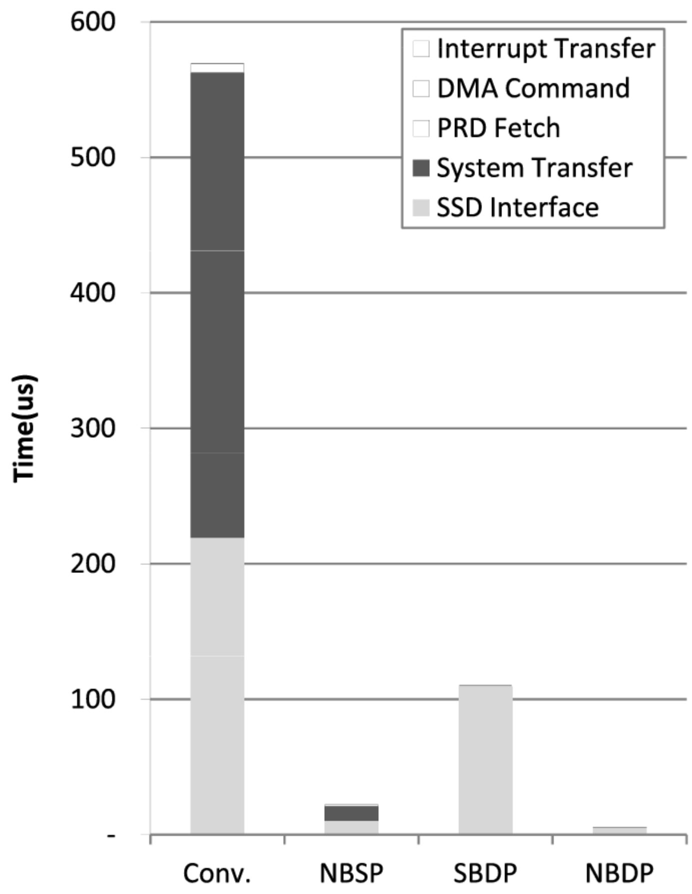
	SB elimination	DRAM I/F	Direct path
NBSP	27.48%	72.52%	–
NBDP	30.04%	34.79%	35.16%

- Contribution of SB elimination is not marginal
  - No hidden effect by SSD internal latency
- DRAM interface is still stronger for NBSP
- Overall even contribution for NBDP

DMA Write



Total Time



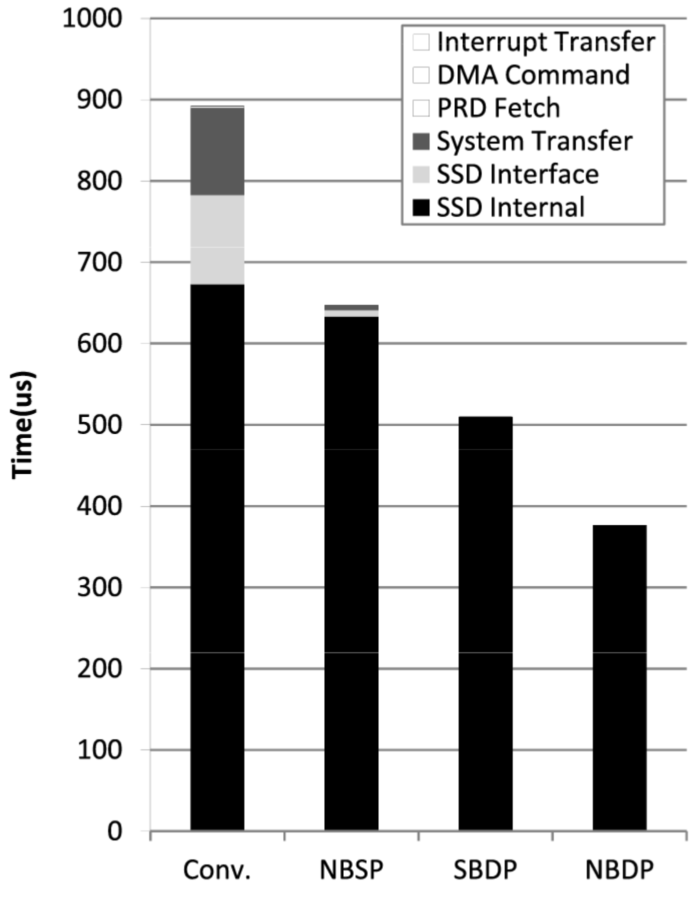
Excluding SSD internal

Contribution Breakdown in Write

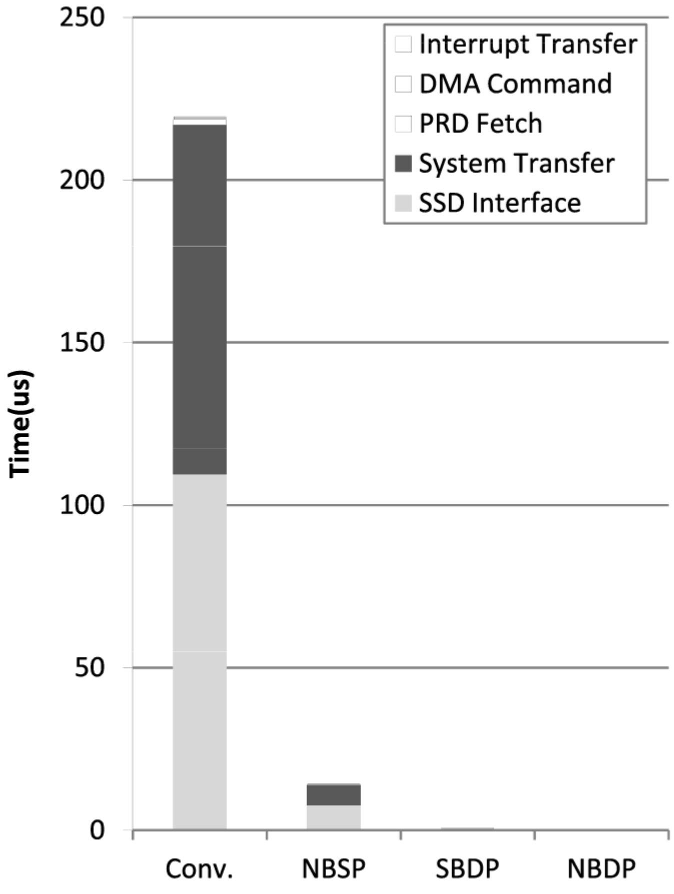
	SB elimination	DRAM I/F	Direct path
NBSP	37.92%	62.08%	–
NBDP	23.00%	7.03%	69.97%

- DRAM interface is stronger for NBSP
  - Similar to DMA read cache hit
- Direct path is dominant for NBDP
  - Similar to DMA read cache miss

Page Fault Scenario



Total Time



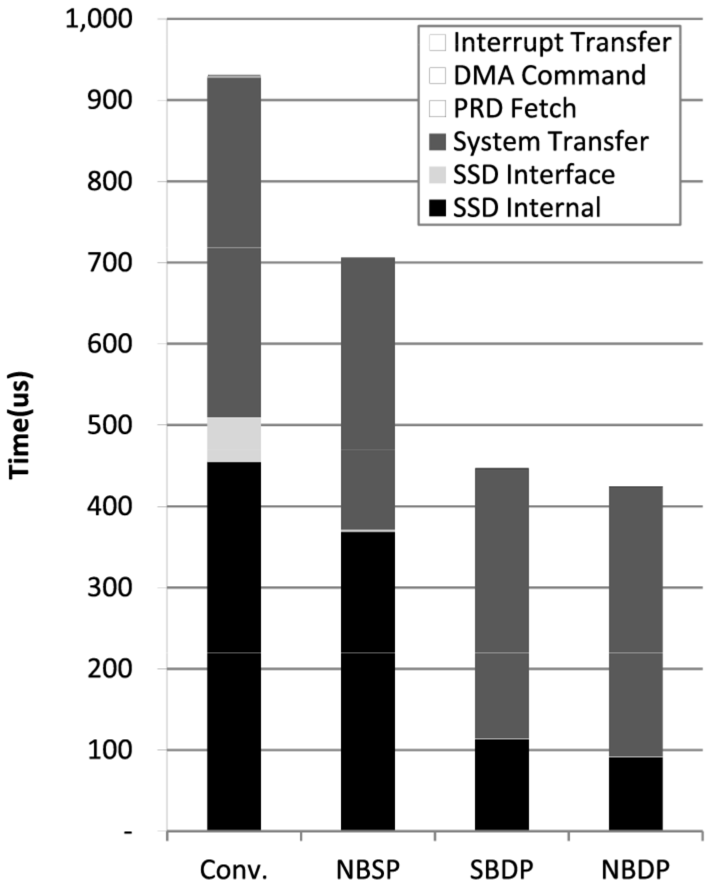
Excluding SSD internal

Contribution Breakdown in P.F.

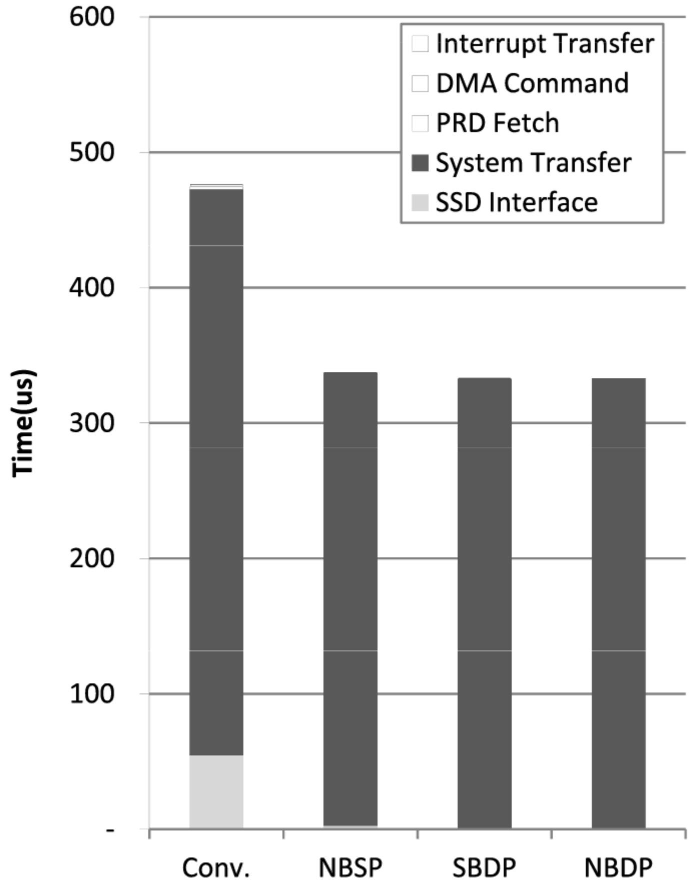
	SB elimination	DRAM I/F	Direct path
NBSP	42.23%	57.77%	–
NBDP	21.31%	10.60%	68.09%

- Similar to DMA write for both NBSP and NBDP
- Victim page write is dominant and new page read is marginal for page fault

Network Download Scenario



Total Time



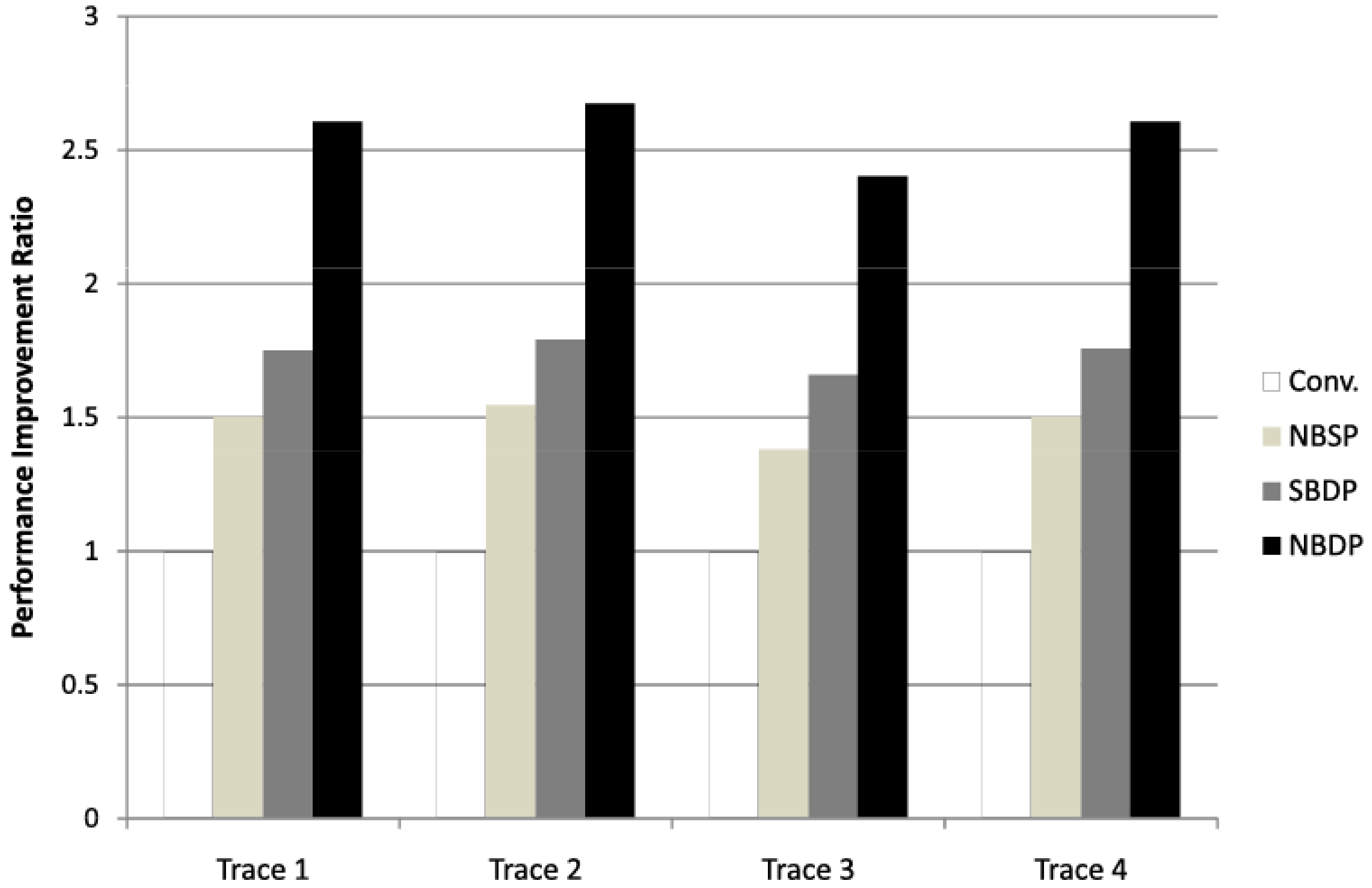
Excluding SSD internal

Contribution Breakdown in N.D.

	SB elimination	DRAM I/F	Direct path
NBSP	38.52%	61.48%	–
NBDP	–	–	100%

- DRAM interface is stronger for NBSP
  - Similar to DMA read cache hit or DMA write
- The communication between main memory and SSD occurs solely via the direct path

Real Traces – About 20% Hit Ratio



Max. Performance Improve Ratio

Sub-Op.	SBSP	NBSP	SBDP	NBDP
DMA Read miss	1	1.31	1.60	2.29
DMA Read hit	1	8.70	2.13	28.56
DMA Write	1	1.59	1.83	2.75
Page Fault	1	1.37	1.74	2.37
Network Download	1	1.31	2.08	2.19
Real Trace 1	1	1.50	1.74	2.60
Real Trace 2	1	1.54	1.79	2.67
Real Trace 3	1	1.38	1.65	2.40
Real Trace 4	1	1.50	1.75	2.60

- Cache hit ratio is the most important
  - Ideally, all the data are read from high-speed DRAM
- Cache buffer should be carefully designed!

Outline

- Introduction
- Related Works
- Motivation
- The Proposed Techniques
- PC Architecture Exploration
- Experimental Results
- Conclusion and Future Work
- Summary

Conclusion and Future Work

- How to make an extreme-performance SSD
  - SSD should be located quite close to main memory
  - The number of path(or bandwidth) is as important as high-speed interface
  - Make 100% cache hit by all means
- We DREAM a solution to satisfy the conditions
- A single general memory device having DRAM and NAND at the same memory hierarchy
  - Replace traditional main memory and disks
  - Not only high-performance SSD internal arch.
  - But also paradigm shift on PC architecture and OS

Outline

- Introduction
- Related Works
- Motivation
- The Proposed Techniques
- PC Architecture Exploration
- Experimental Results
- Conclusion and Future Work
- Summary

Summary

- SSD is the solution to overcome CPU/IO gap
- Technical directions to next generation SSD
  - SSD internal architecture
  - SSD interface scheme
  - NAND flash interface scheme
  - System-level architecture exploration
- Architectural improvement will be driven by IO device in this Exa-byte era

Thank you!

Q & A