*컴퓨터 및 **VLSI**특론*

Embedded Systems

Eui-Young Chung

연세대학교 전기전자공학부
School of Electrical and Electronic Engineering
Design Technology Laboratory

Outline

- 연구실 소개
- **SoC architecture**
- **Solid-State Disk**

Outline of Design Technology Lab (DTL)

- **Members (as of Mar. 2011)**
  - **PhD candidates: 5**
  - **Master candidates: 9**

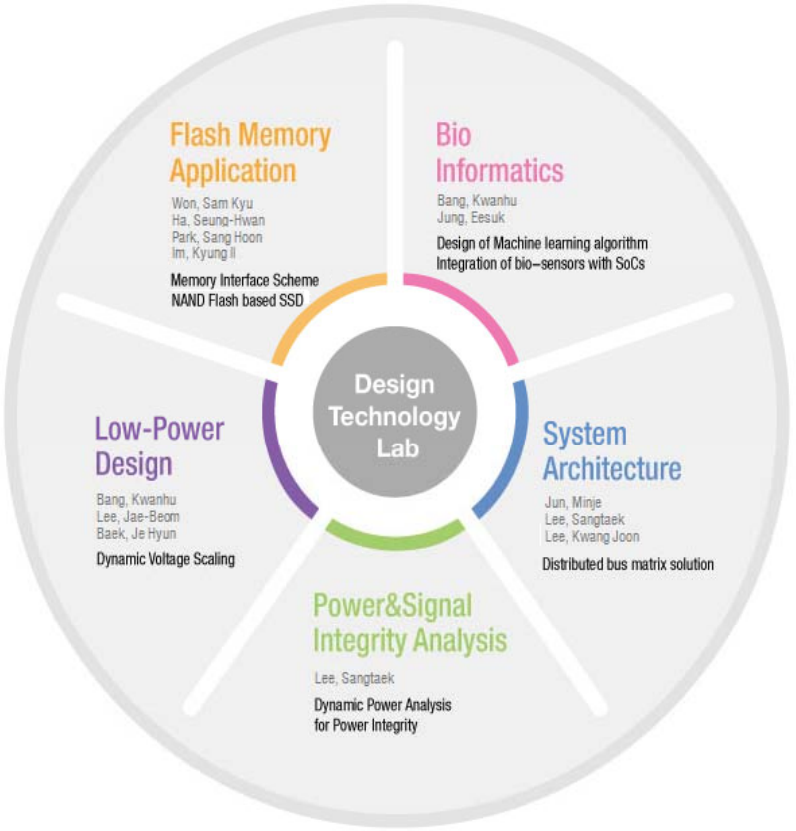
연구실 소개

❑ 구성원

- **5 (박사과정) + 9 (석사과정) (학연 4명: 삼성, LG, 하이닉스)**

❑ 연구 분야

- **SoC / Computer / Bio**
  - **HW + SW**
- **Industry-aligned area**
  - **SoC / Computer**
    - **Solid-State Disk (SSD)**
    - **System architecture**
    - **Low power design**
    - **VLSI CAD**
- **Academia interests**
  - **Bio-informatics**
    - **Parallelization**
    - **Machine learning**



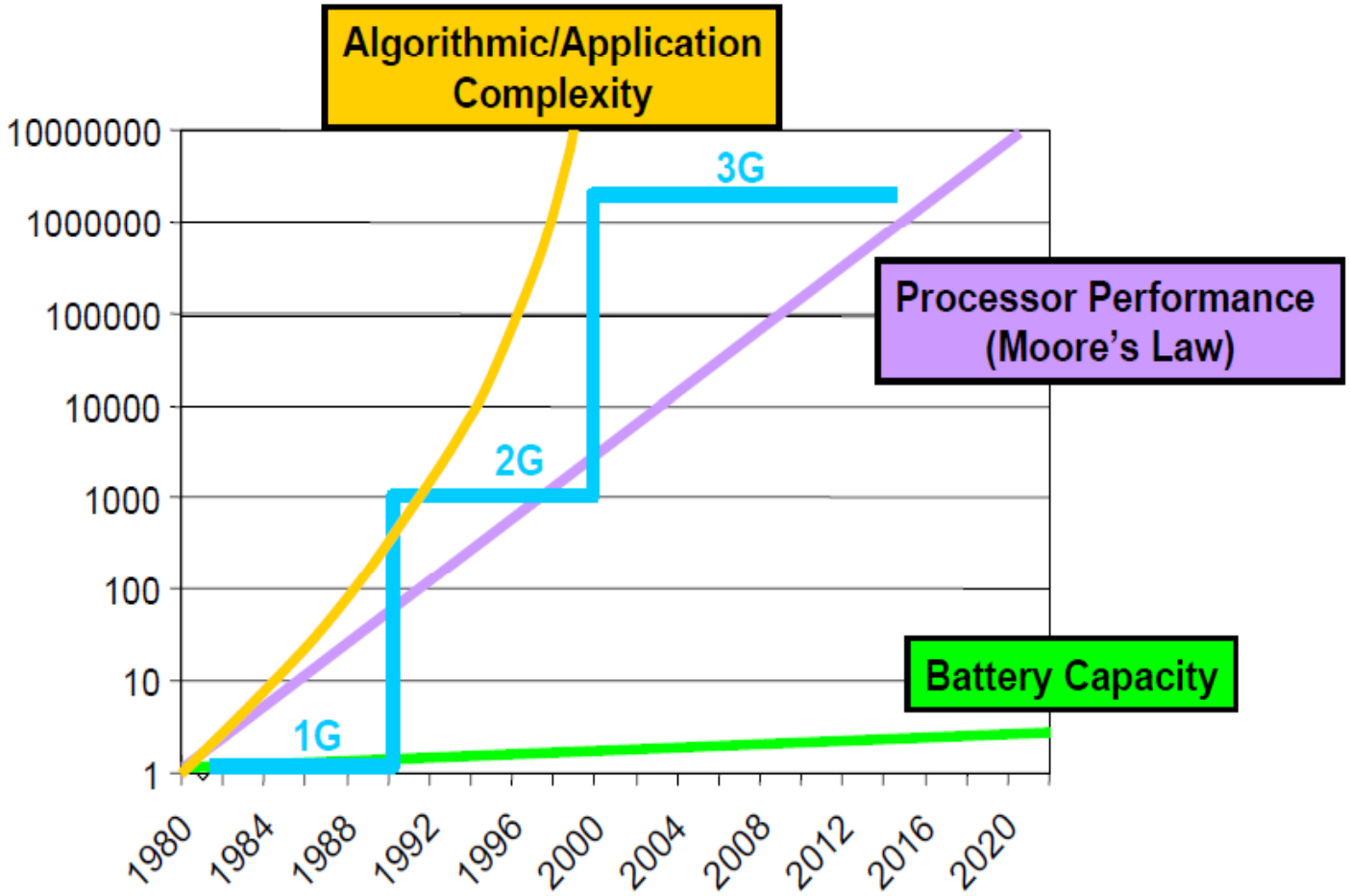
연구 실적

- **On-going projects (연 3억내외)**
  - **SSD** 관련 연구**:** 삼성 **/** 하이닉스 **/** 한국연구재단 기초 연구
  - **Low power design** 관련 연구**:** 한국연구재단**, LG**
  - **System Architecture** 관련 연구**:** 한국연구재단
  - **Bioinformatics** 관련**:** 한국연구재단 신진연구
- 논문
  - **SCI(E)** 급 논문 **22**편 **(**책임 **15**편**) from 2005.09**
  - 유관분야 **grand slam**
    - **IEEE TC, IEEE TCAD, IEEE TVLSI**
  - **IEEE transaction or IF > 3** 이상 논문**: 12**편

Outline

- 연구실 소개
- **SoC architecture**
- **Solid-State Disk**

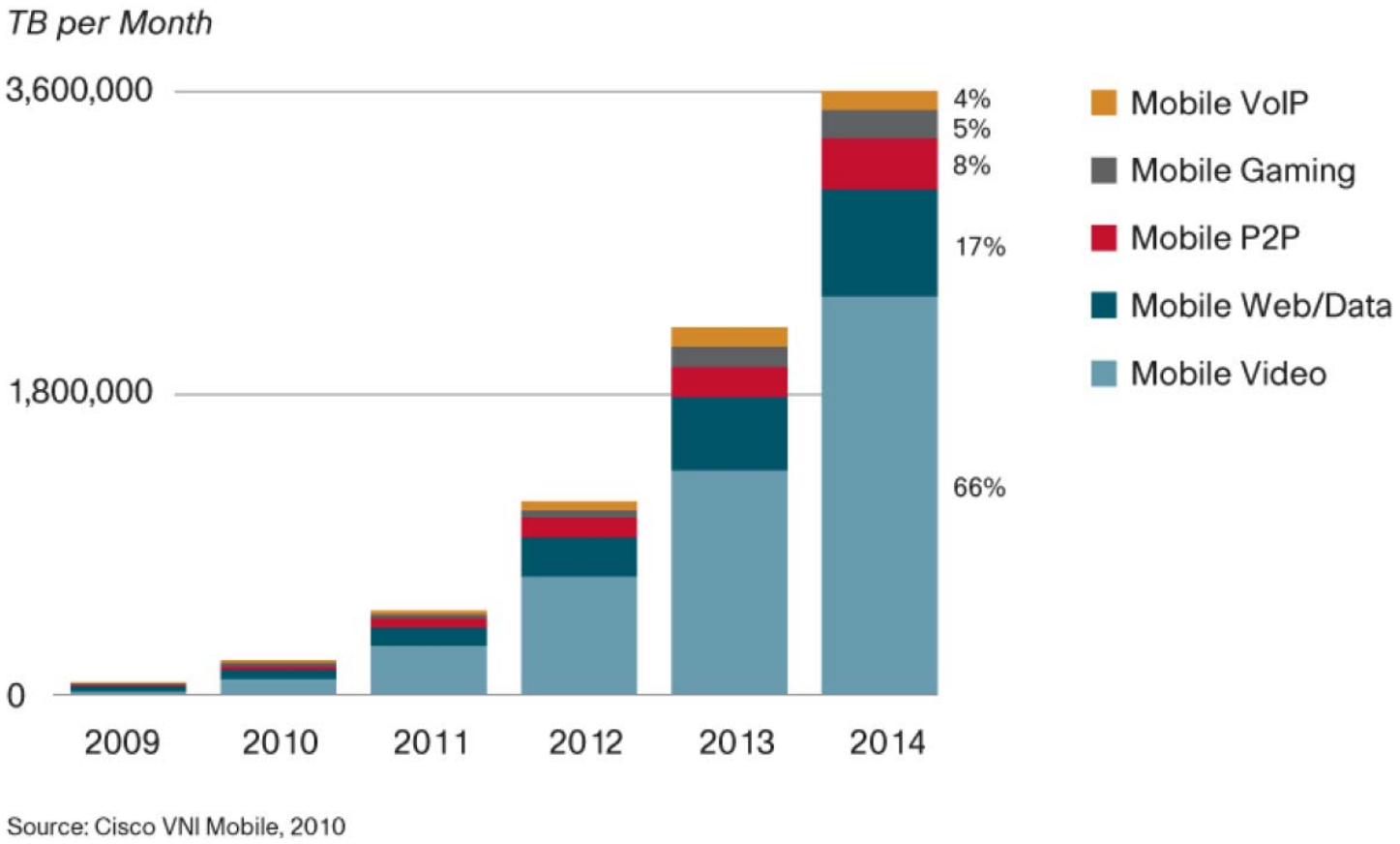
System Functional Requirements



Source : Processor design System-on-Chip computing for ASICs and FPGAs

Data Increase and classification

- **Mobile video will account for 66% of global mobile data traffic by 2014**



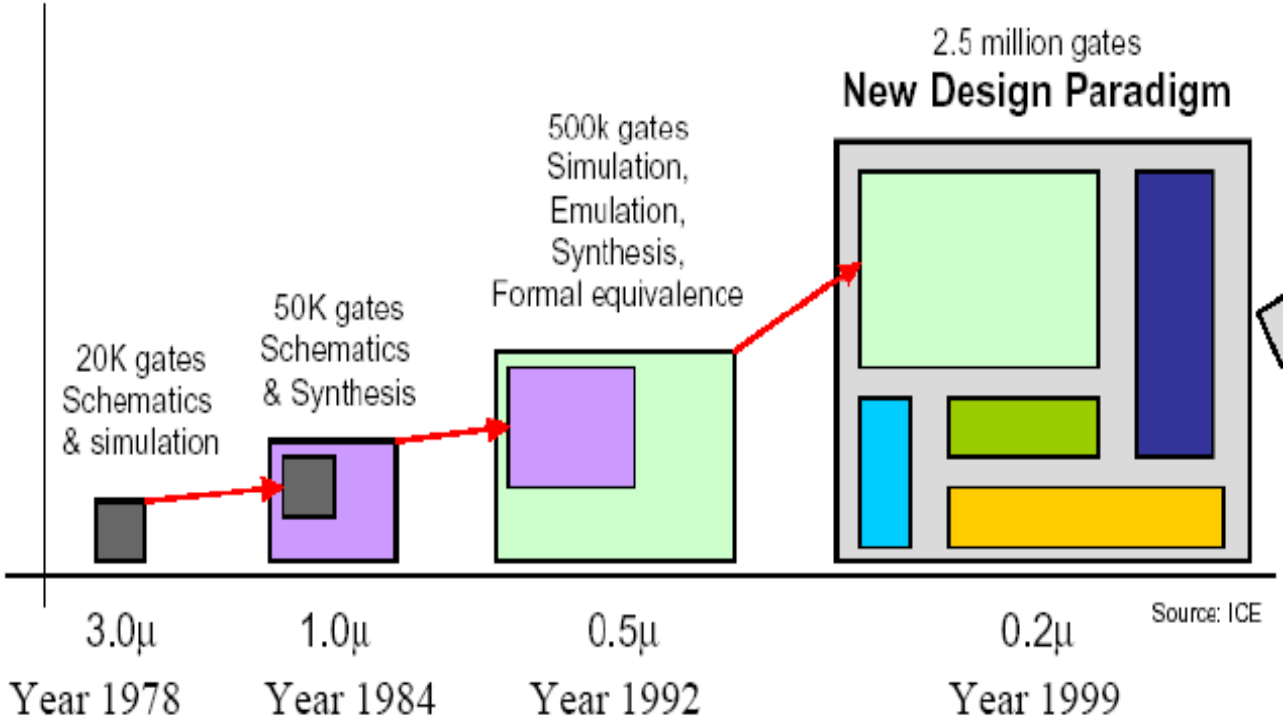
Source: Cisco VNI Mobile, 2010

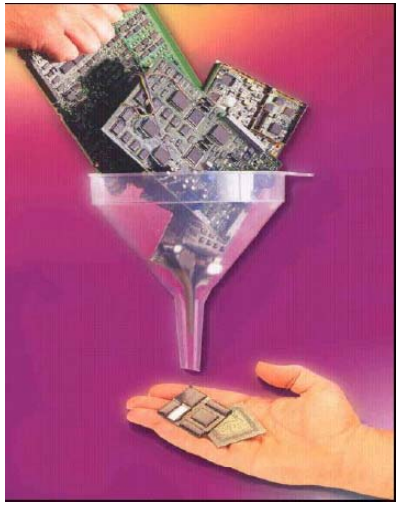
Data-intensive applications

- More computing power
  - Multi- / Many- core processing
- Higher data communication bandwidth
  - High-speed on-chip interconnects
  - Fast storage devices
    - On-chip memories as well as external storages
      - DRAMs
      - SSDs
- More power demanding
  - Reduce the power of frequently used resources
    - On-chip interconnects
    - Storages

Evolution of Microelectronics

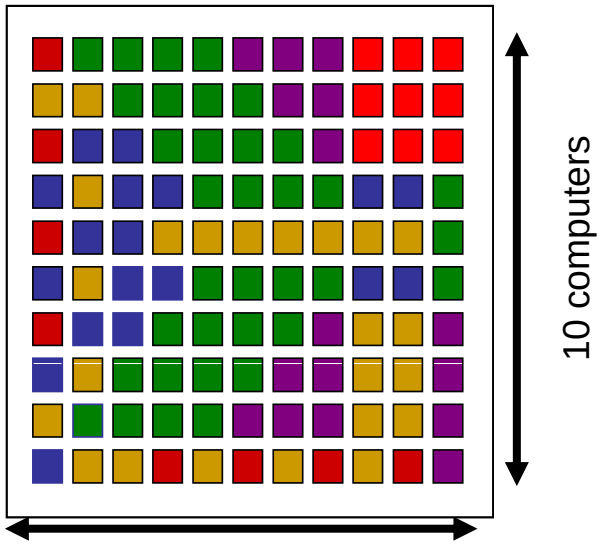
- **Yesterday's chip is today's function**





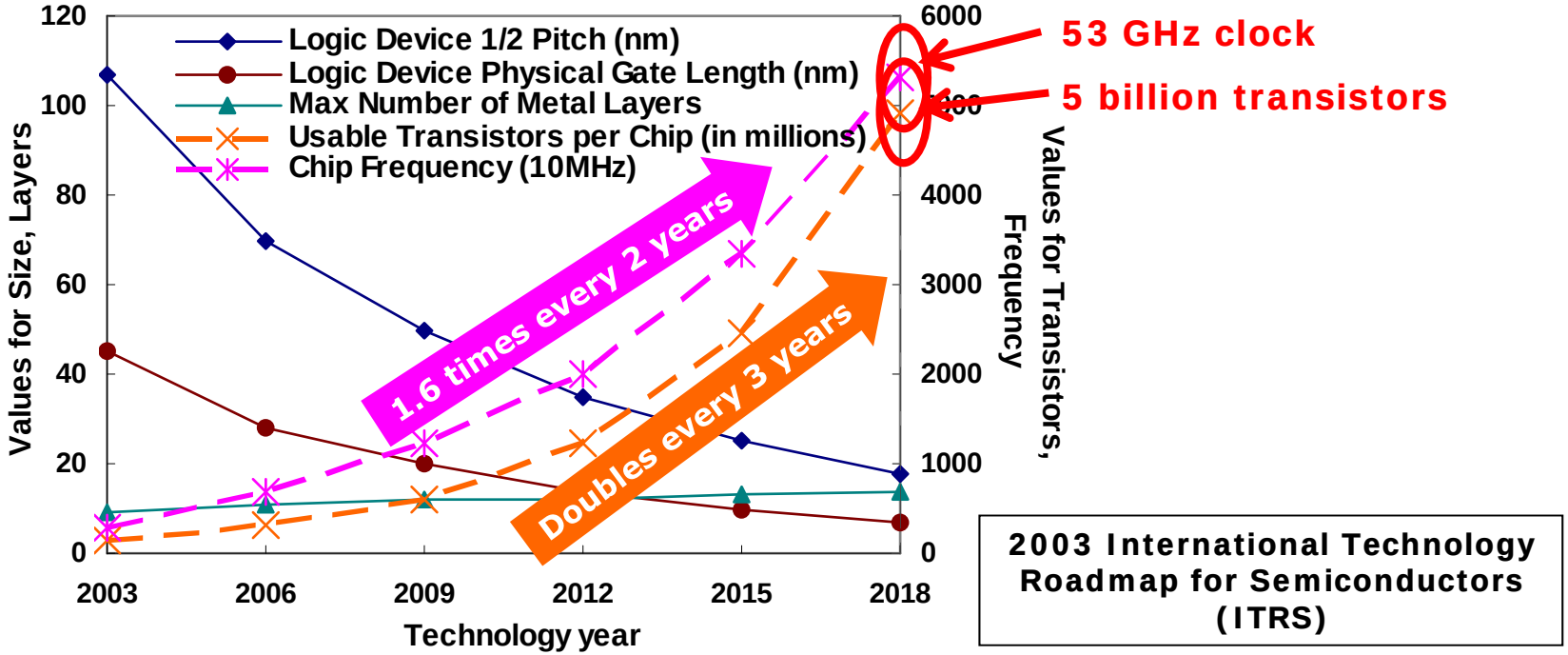
Billion Transistor Era

- 1 billion transistor SoCs are expected to be used in products by 2008.
  - Tens or even hundreds of computer-like resources in a single chip.
- According to ITRS, SoCs at 50nm will have 4 billion transistors and operate at 10Ghz in the next decade.



Silicon Technology Advance

- **High-volume, high-frequency chips**



2003 International Technology Roadmap for Semiconductors (ITRS)

  - **High integration density**
    - **Macrosystems ⇨ Microsystems**
    - **Complex on-chip communication requirements**

System-on-Chips (SoCs)

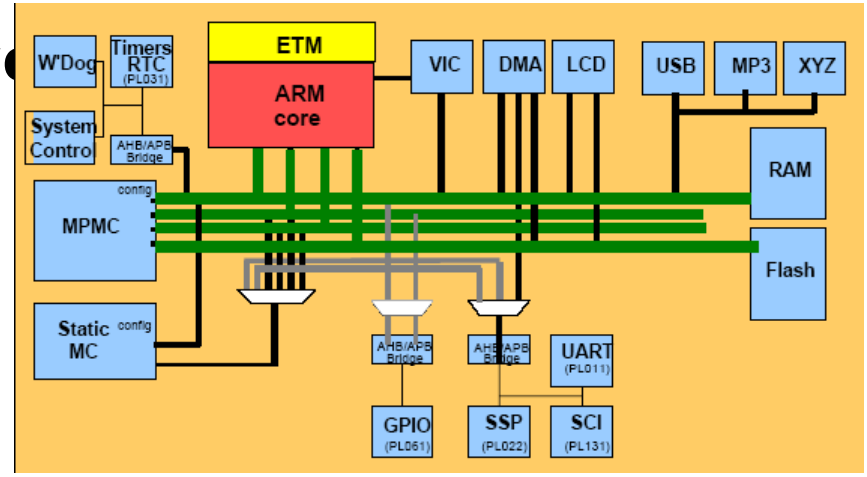
- Solution to cope with increasing circuit complexity.
  - System in terms of subsystems
  - Different Levels of Concepts and Abstraction
- Efficient reuse of designs and design experience
  - Pre-designed Intellectual property (IP) cores
    - Processors, Cache and Memory cores
    - DSP cores
    - Buses (?)
  - Meeting the TTM (time-to-market) constraint
- Facilitated by new design methodologies
  - Interface-based design
  - Platform-based design

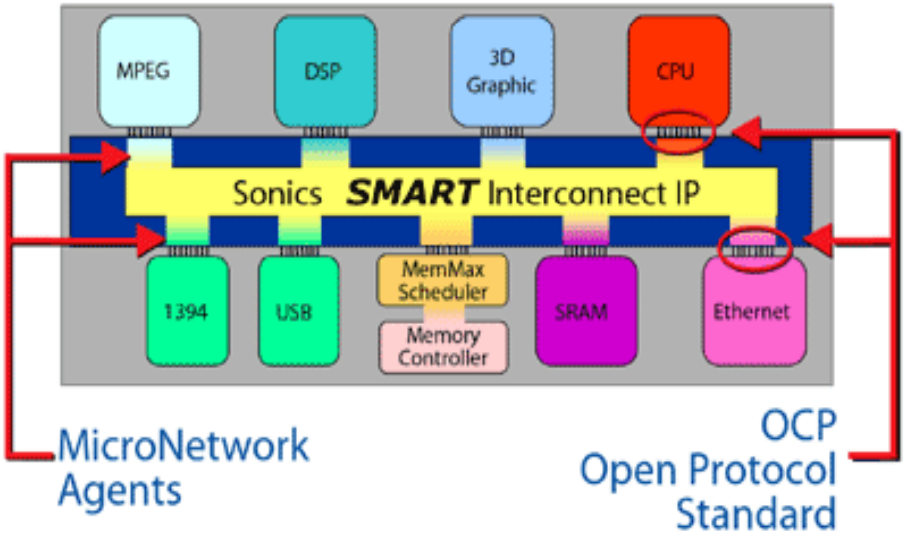
On-Chip Interconnects

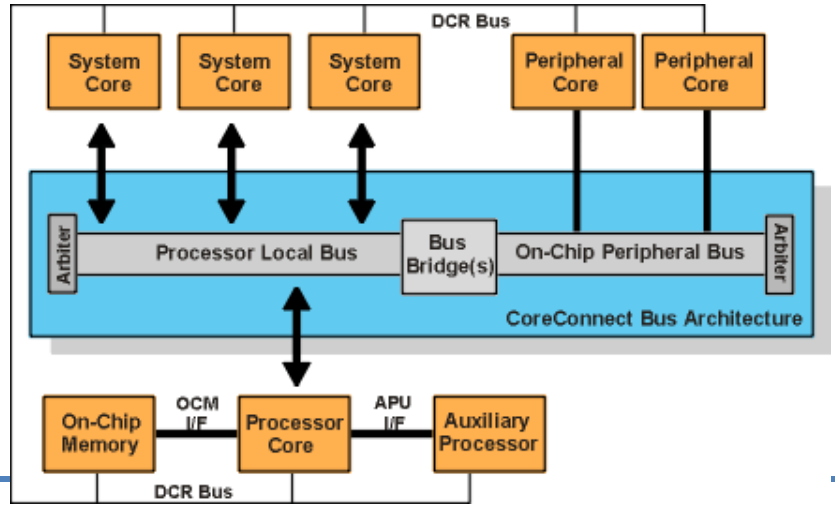
- Communication channels for functional modules (or IP blocks) integrated in a single chip.
  - Shared media like a bus
  - Dedicated point-to-point links
- So far, on-chip interconnects provide limited bandwidth for lower-performance, lower-power cores.
- Standardized bus systems with the incorporation of pre-designed Intellectual Property (IP) cores.
  - AMBA (Advanced Microcontroller Bus Architecture) by ARM
  - SiliconBackplane uNetwork by Sonics
  - CoreConnect by IBM

Popular Industry Solutions

- AMBA (Advanced Microcontroller Bus Architecture)
  - ARM
- SiliconBackplane MicroNetwo
  - Sonics
- CoreConnect
  - IBM

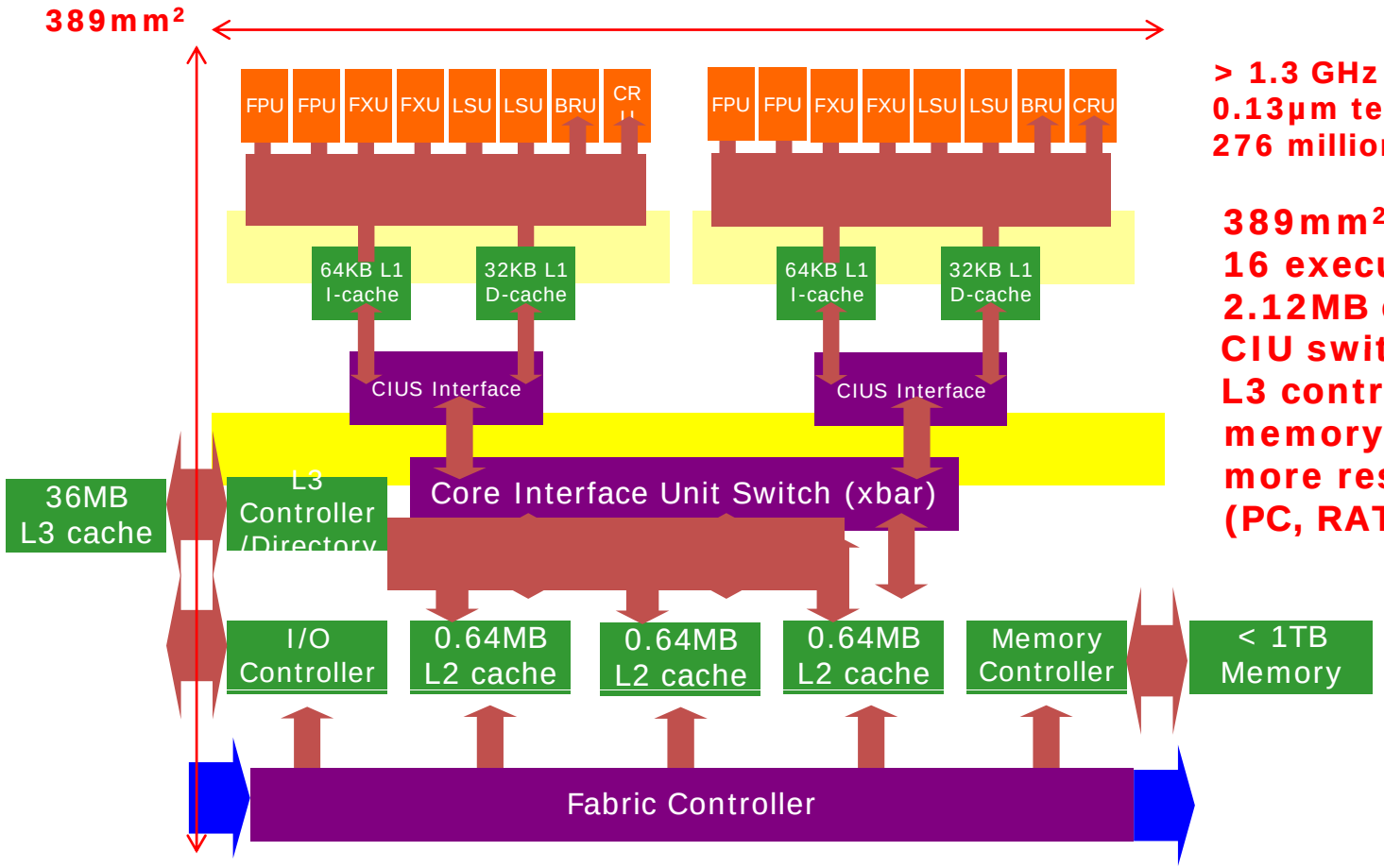






High Density SoC Based on Buses

IBM POWER5 (Year 2004)



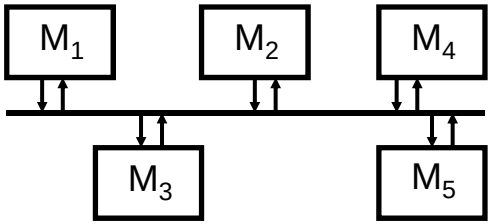
> 1.3 GHz
> 0.13µm tech
> 276 million Tr.

389mm² chip has
16 execution units,
2.12MB cache,
CIU switch, I/O controller,
L3 controller/directory,
memory controller, and
more resources for SMT
(PC, RAT, RF, CL, etc.)

*Refer to http://www-03.ibm.com/servers/eserver/pseries/hardware/whitepapers/power4.html

Characteristics of Bus-based Interconnects

- Communication based on shared-medium (e.g. bus)
  - Multiplexer-oriented topologies
- Pros
  - Simple topology, low area cost and extensibility
- Cons
  - Performance bottleneck
  - Scalability problem
  - Power consumption inefficient
  - Unpredictable performance



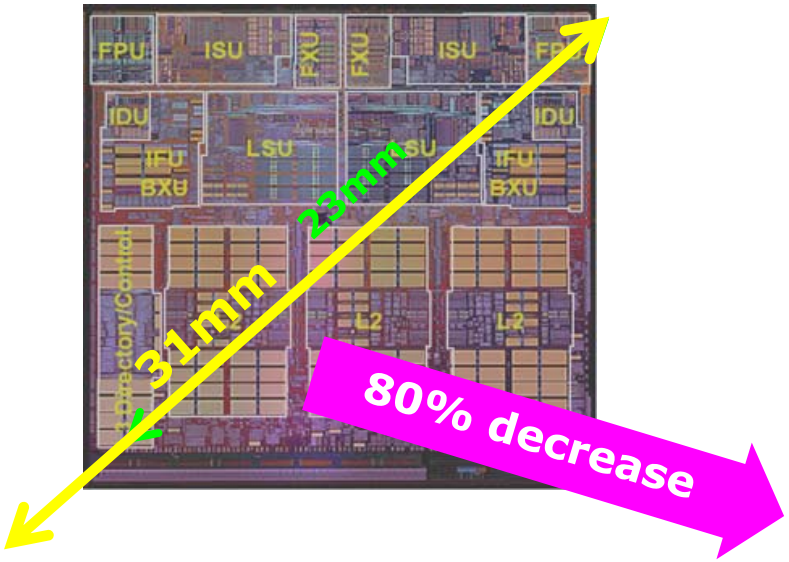
Properties Limiting the Use of Bus

- **Wire delay**
  - Wires become “longer”, and wire delay becomes a performance bottleneck
  - Partition a long wire in segments with repeaters
  - Synchronization problem
- **Power**
  - More energy consumption due to longer wires
  - To reduce delay, bigger drivers are used, which increase energy consumption
  - Typical solutions
    - Reduce voltage swing
      - Good for performance and power
      - But reduces noise margins => more errors!
    - Differential signaling
- **Signal integrity**
  - Growing capacitive and inductive coupling between wires
  - IR drop, Cross-talk, Electro-migration, …

Reachable Physical Distance Per Clock

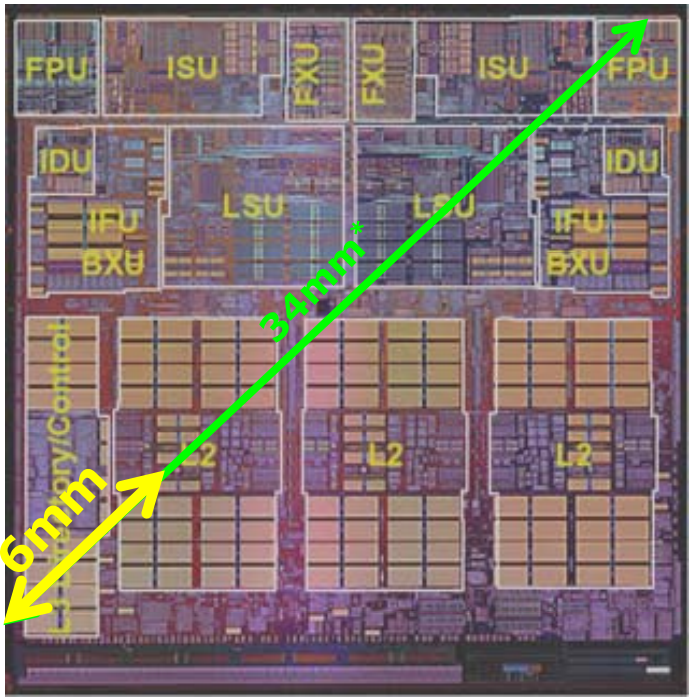
Reachable physical distance within one clock

1GHz POWER4 (.18µm)



100% reachable in one clock

8GHz* POWER6 (.065µm)

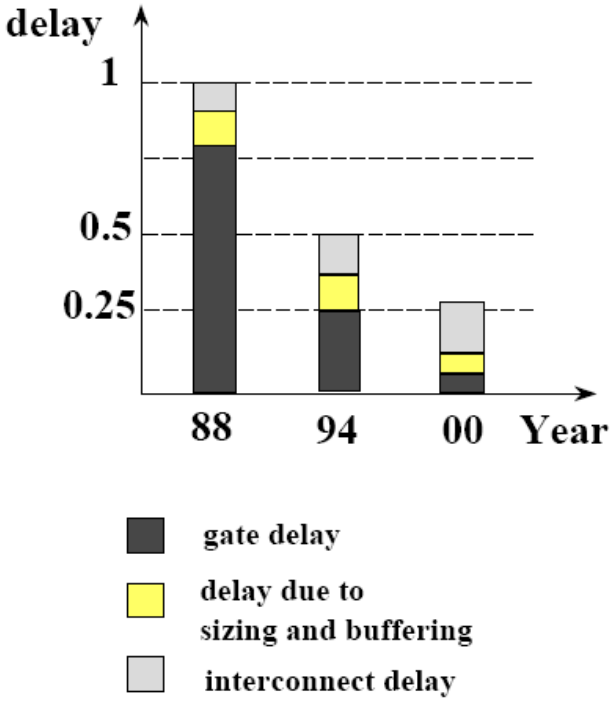


18% reachable in one clock

* Projected values (Year 2006)

Performance Impact of On-Chip Interconnect

Operation	Delay (0.13um)	(0.05um)
32b ALU Operation	650ps	250ps
32b Register Read	325ps	125ps
Read 32b from 8KB RAM	780ps	300ps
Transfer 32b across chip (10mm)	1400ps	2300ps
Transfer 32b across chip (20mm)	2800ps	4600ps

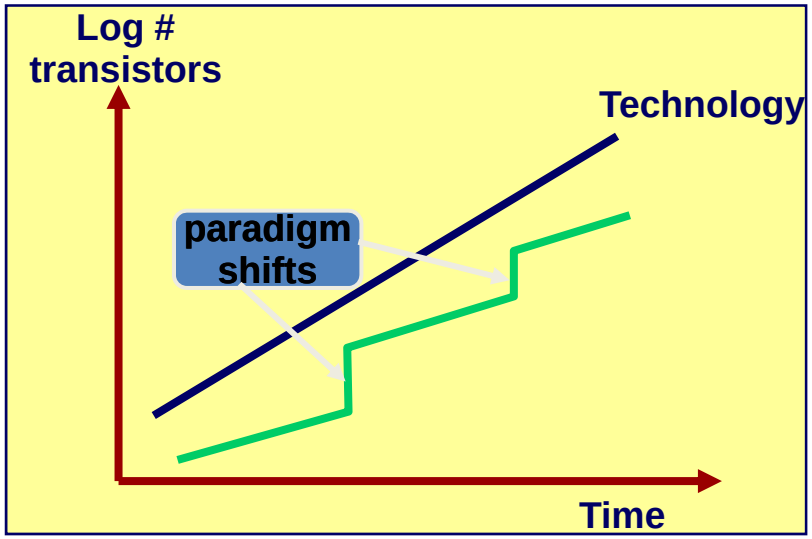


2:1 global on-chip comm to operation delay
9:1 in 2010

Taken from W.J. Dally presentation: Computer architecture is all about interconnect (it is now and it will be more so in 2010) HPCA Panel February 4, 2002

Challenges in SoC Design

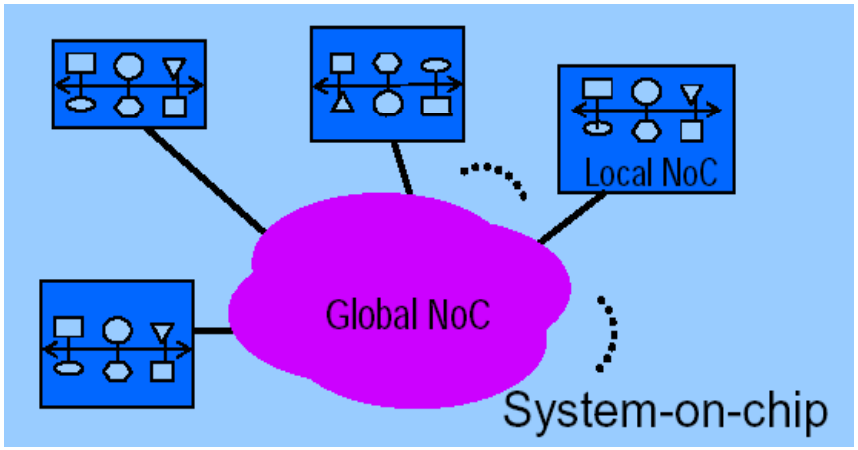
- **Time-to-market pressure**
  - **Design productivity gap**
- **Complexity**
  - **Heterogeneous**
  - **Deep submicron effects**
  - **Performance / Energy / Cost tradeoff**
  - **Scalable architecture**
- **New Design Paradigm**
  - **IP / Platform-based design**
  - **Error tolerant design strategy**
  - **Interconnect oriented design**

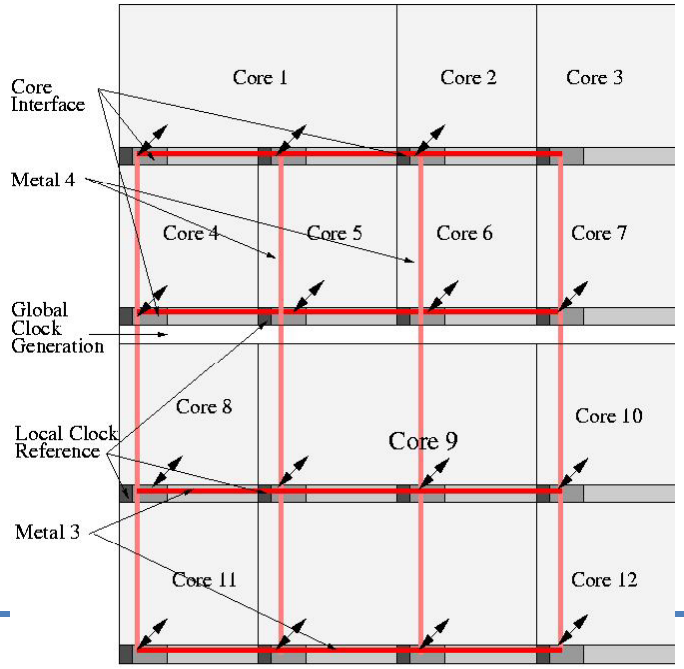


⇨ Paradigms shifts in design methodology is the only escape

Network on Chip

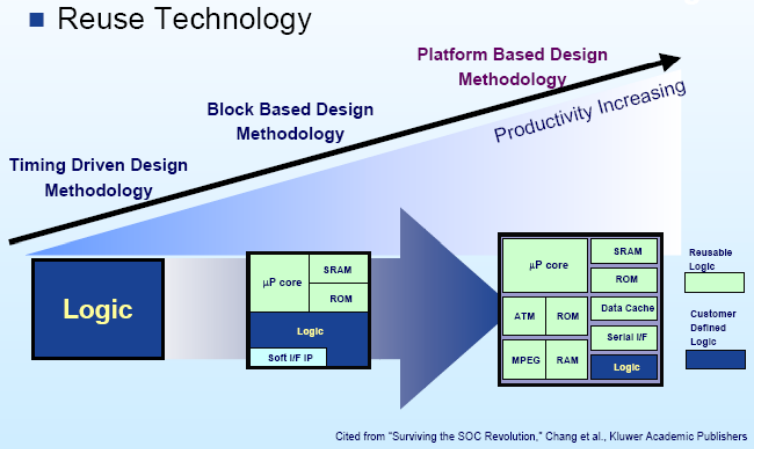
- **Communication channels for computer systems or modules integrated within a single chip.**
  - **Resources are interconnected by a network of switches.**
- **Large-scale integration of SoCs with the scalable interconnects**



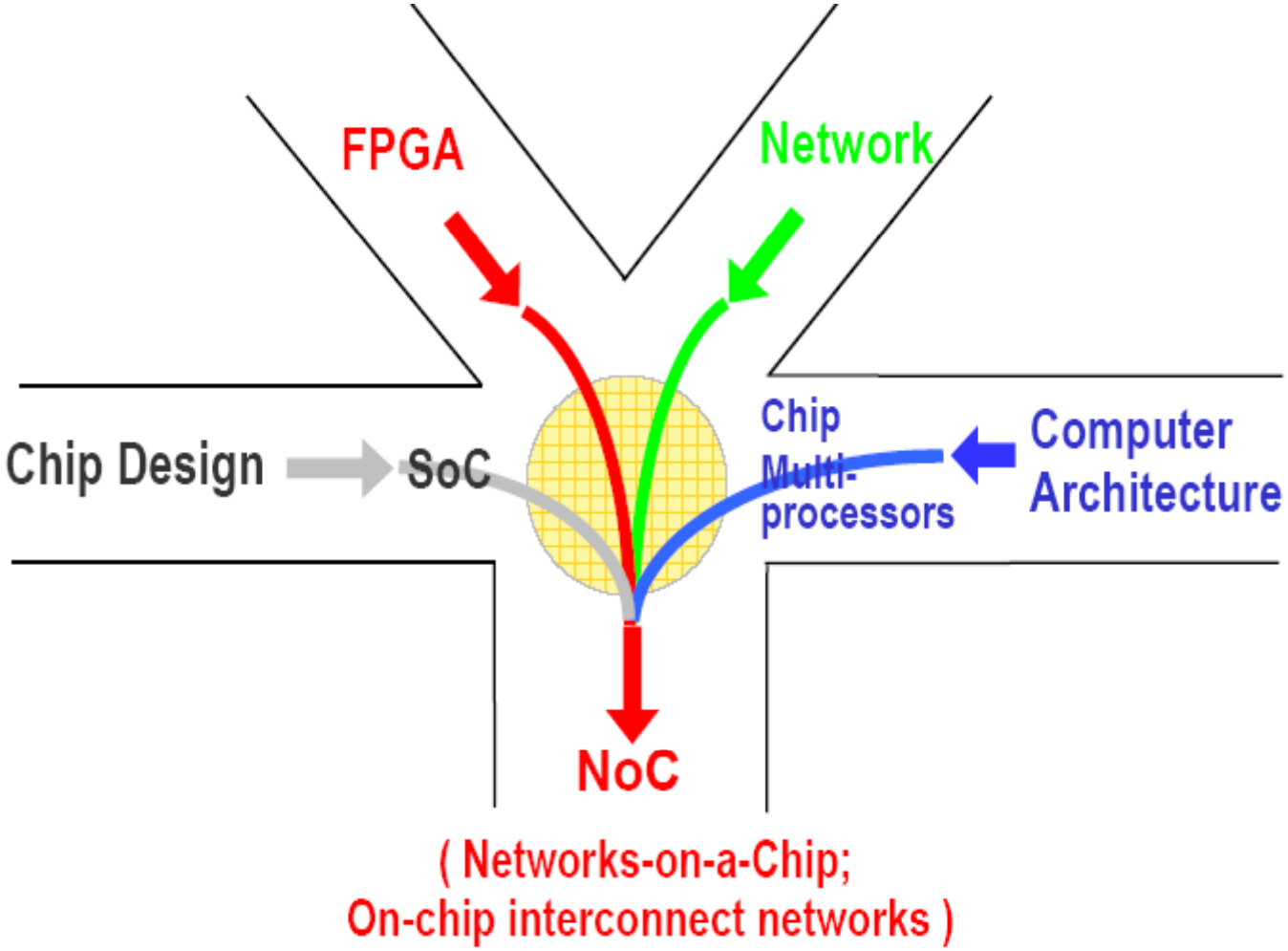


Advantages of NoC

- **Reuse**
  - Components and resources
  - Communication platform
  - Design and verification time
- **Predictability**
  - Communication performance
  - Electrical properties
- **Scalability**
  - Computing, Memory and Interconnection Resources
- **Modular, compositional**
  - Decoupling computation and communication



Yet Another Interconnection Network



Outline

- 연구실 소개
- **SoC architecture**
- **Solid-State Disk**

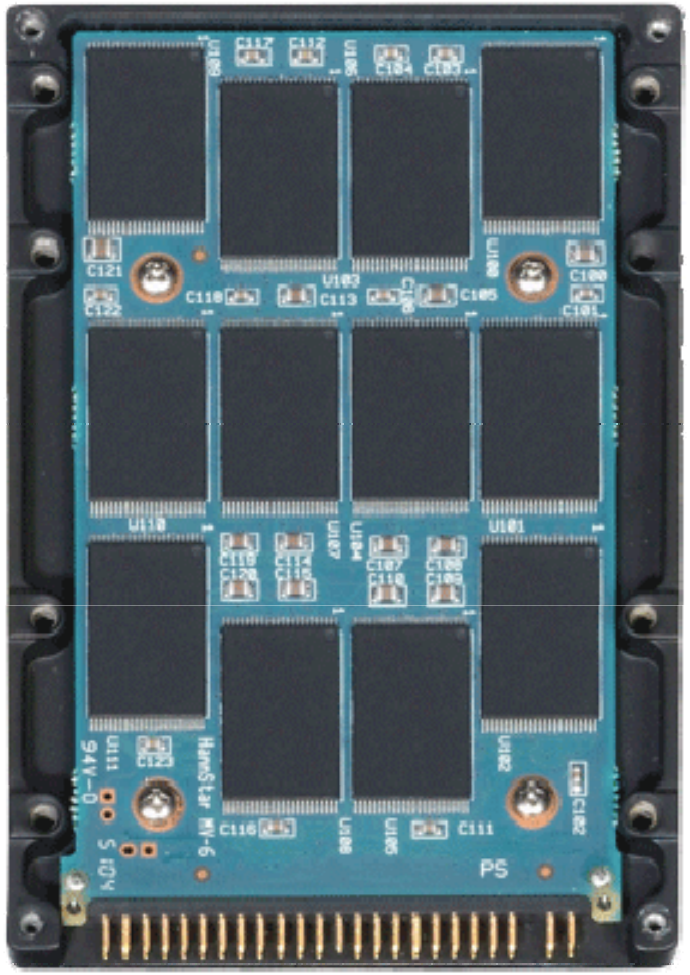
What is an SSD?

- **Abbreviation for Solid-State Disk**
  - A data storage device that uses solid-state memory
- **“Solid-state”**
  - built entirely from solid materials
  - contrast with the earlier technologies of vacuum and gas-discharge tube devices
  - electro-mechanical devices (relays, switches and other devices with moving parts) are excluded
  - Ex) Transistor, microprocessor, DRAM, flash memory…
- **A drive built from non-volatile memory chips**
  - not from magnetic disks as in HDD

Characteristics of SSD

- **Performance**
  - I/O performance superior to HDD
  - Outstanding performance with RAID configuration
  - Fully compatible SATA interface
- **Durability and Stability**
  - Simple combination of semiconductor chips
  - High shock and vibration tolerance
  - MTBF superior to HDD
- **Power consumption**
  - Power consumed at data access only (under 3W)
  - HDD always consumes power in order to operate mechanical parts (10~20W)

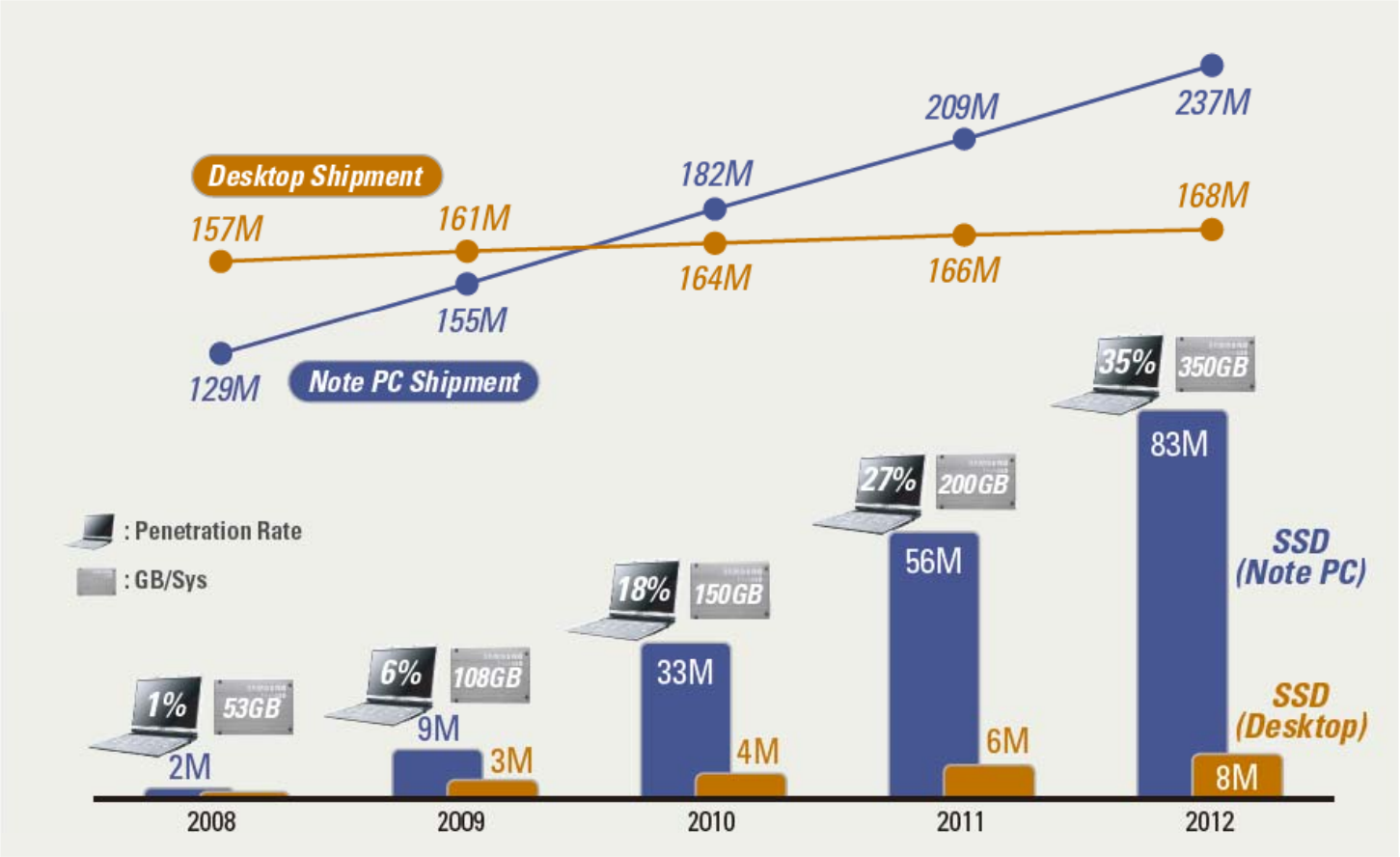
The Internals of SSD and HDD



Comparative Analysis of SSD and HDD

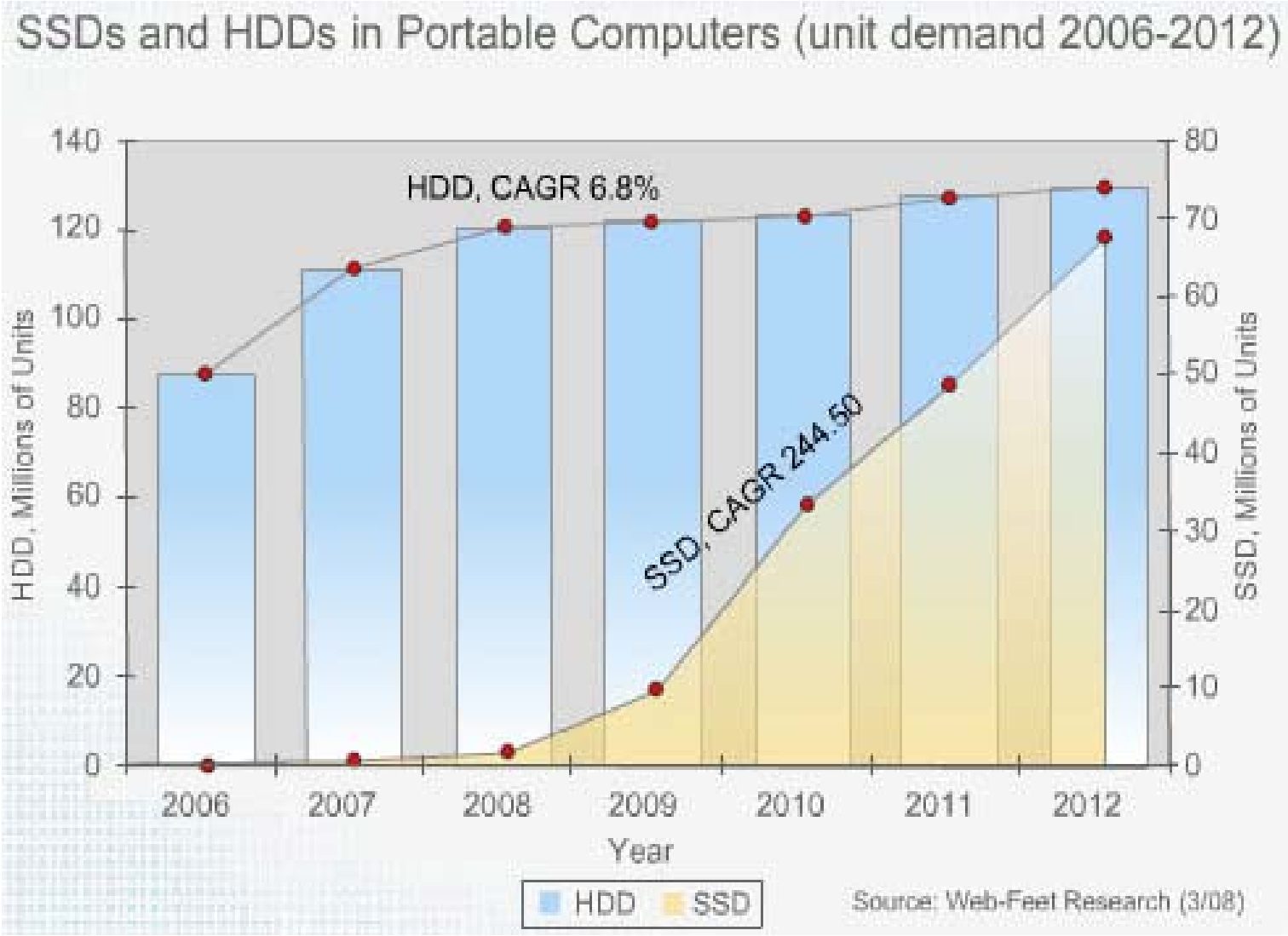
	SSD	HDD
Industry Standard Dimensions	✔	✔
Industry Standard Interface	✔	✔
Rugged / No Moving Parts	✔	
Ultra Low Power Consumption	✔	
Silent Operation	✔	
Fast Access Time	✔	
Fast Enter/Exit Hibernate	✔	
Fast Sustained Read/Write Speed	✔	✔
Light Weight	✔	
Low Cost per GB		✔
Very High Capacity		✔

SSD Market Forecast



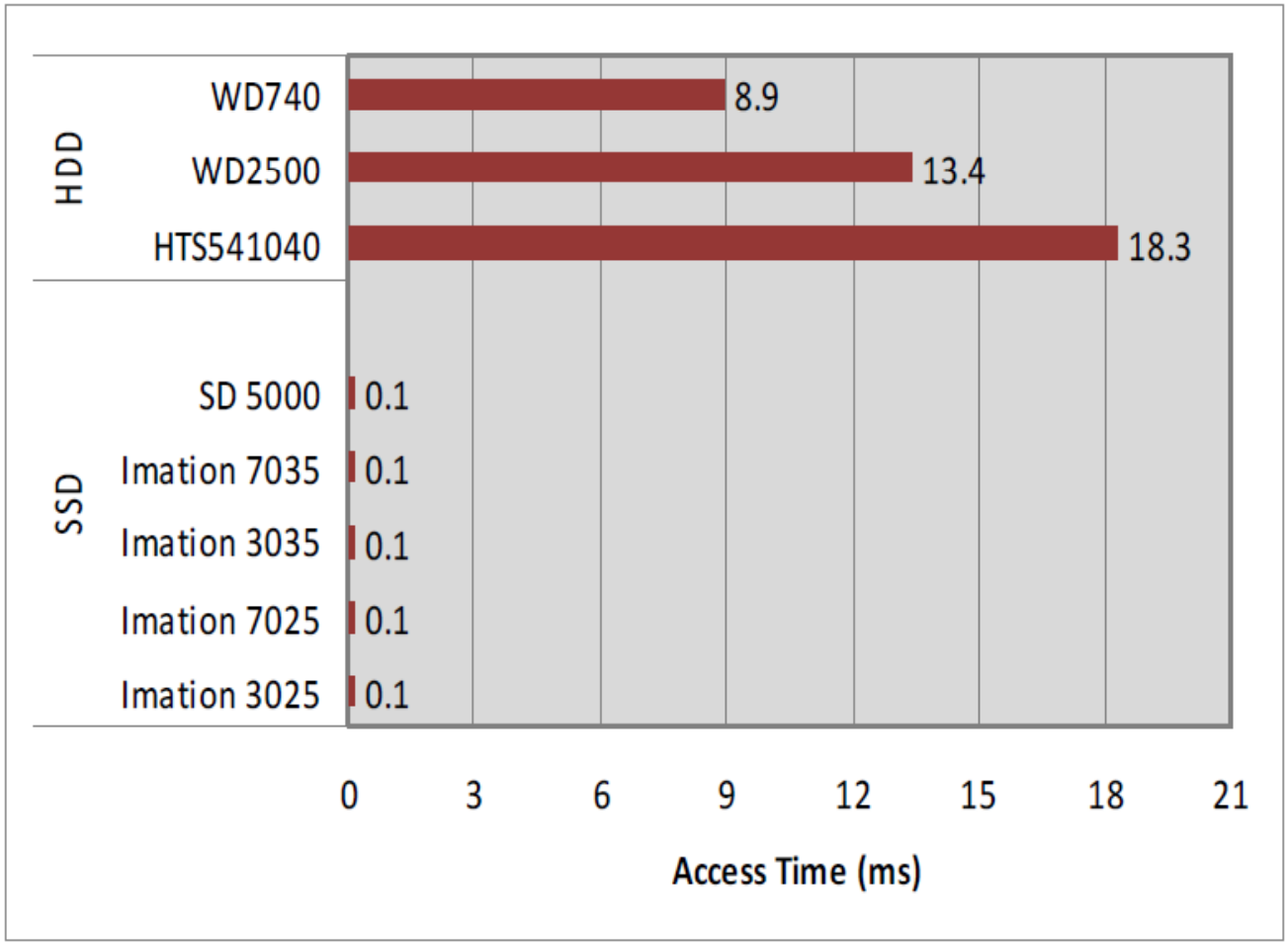
- **And smart devices…**

Change in Storage market



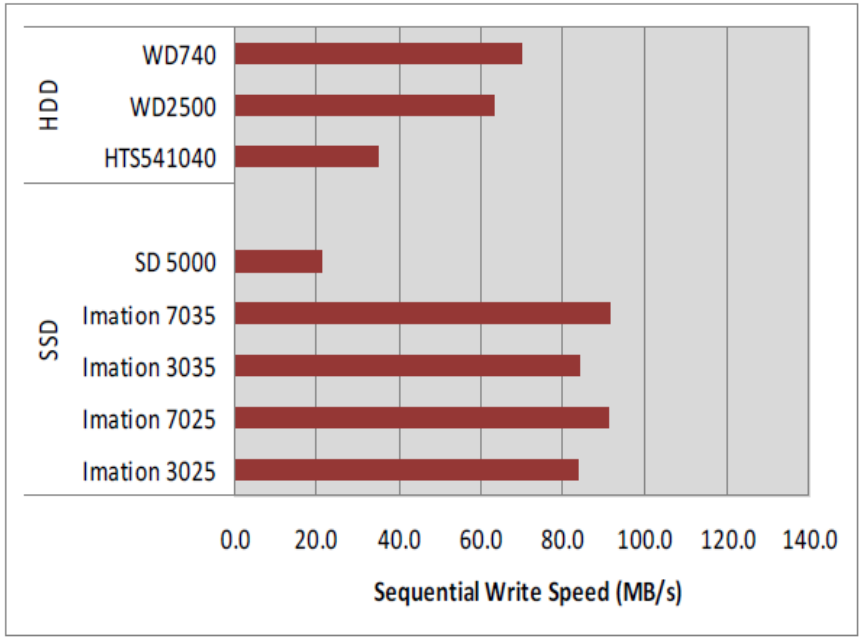
Performance Comparison (1)

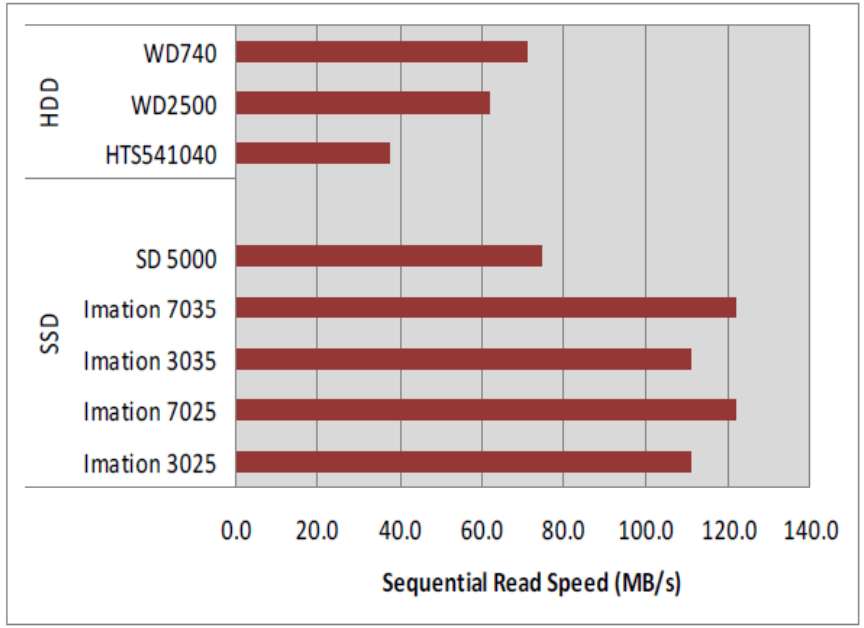
❑ Access time



Performance Comparison (2)

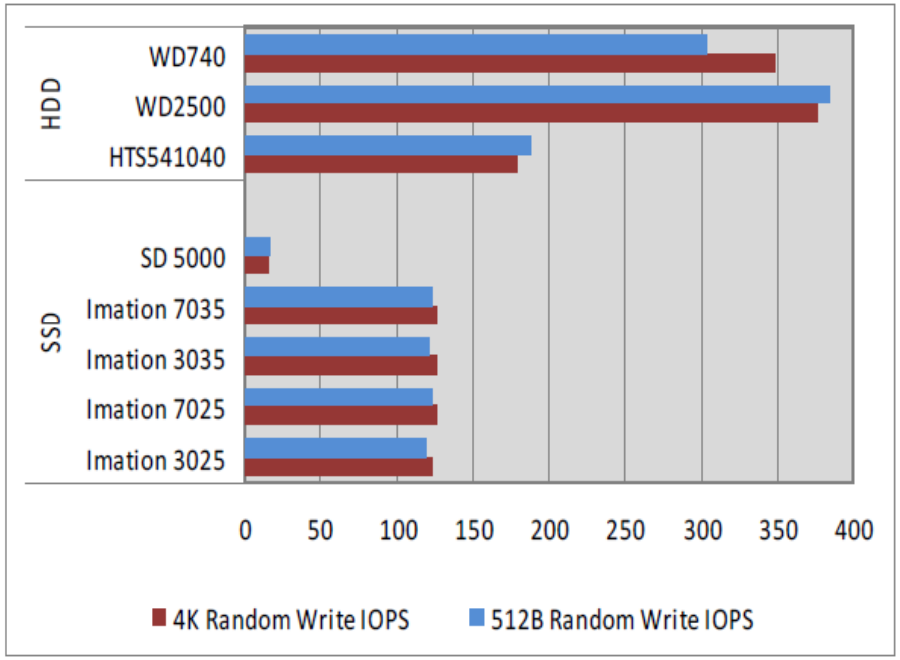
❑ Sequential write and read

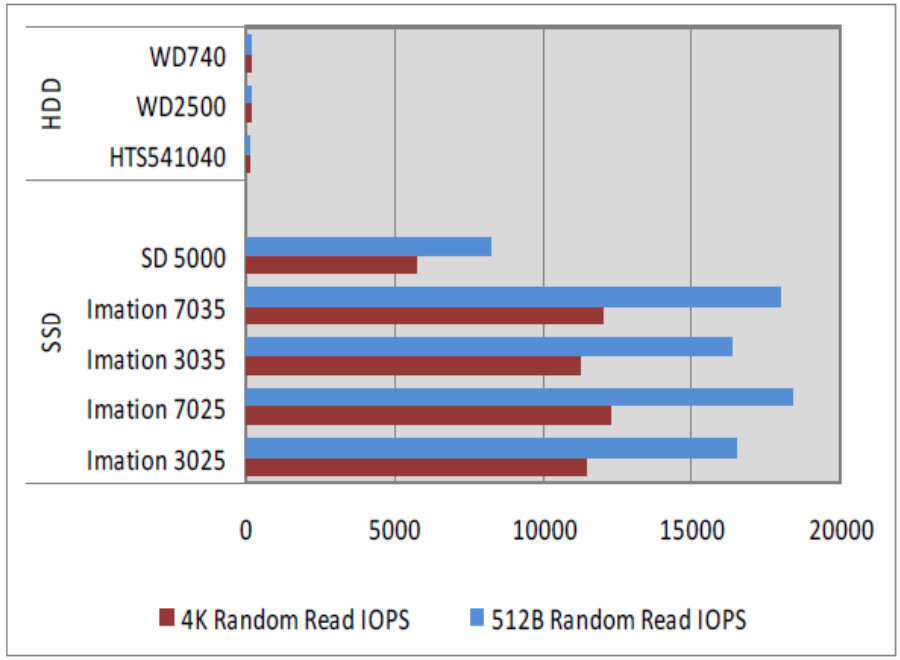




Performance Comparison (3)

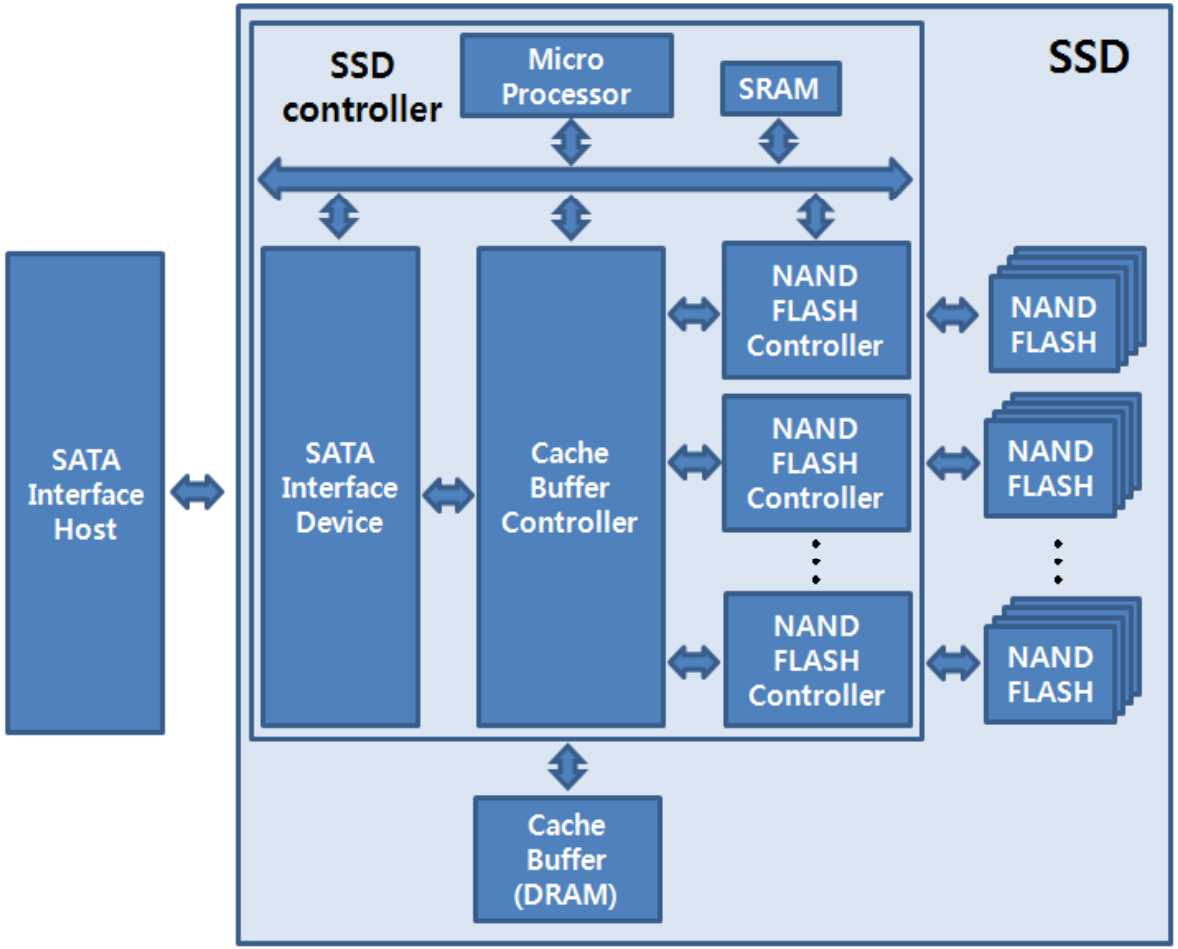
Random write and read





The Architecture of SSD

❑ Block diagram



The Architecture of SSD (cont)

- **Overall architecture**
  - Consists of an SSD controller and bunch of NAND flash memory chips
  - The external interface of SSD is engaged into the host interface counterpart
- **Microprocessor and internal bus**
  - Control of other components of the SSD controller
  - Execute SSD management firmware named Flash Translation Layer (FTL)
- **On-chip SRAM and SRAM controller**
  - Fast SRAM is used for temporal storage of microprocessor
  - SRAM is typically embedded on the chip

The Architecture of SSD (cont)

- **External interface**
  - The system (OS) accesses the SSD by this interface
  - SATA or other high-speed serial interfaces are used
- **Cache Buffer and cache buffer controller**
  - A disk usually have a cache buffer of several tens of MB
  - External DRAM is used for read cache and write buffer
- **NAND flash and NAND flash controller**
  - Single Level Cell (SLC) or Multi Level Cell (MLC)
  - Multi-channel / multi-way

Microcontroller

- **The brain of the SSD subsystem**
  - Execute the SSD firmware called flash translation layer
  - Handle interrupts from other components such as external interface or cache buffer
- **Flash Translation Layer (FTL)**
  - An intermediate software layer between the host application and flash memory
  - Flash memory has limitation of “erase-before-write”
  - The FTL redirects each write request to an empty location that has been erased in advance
  - Extra storage to maintain the address translation information and extra flash memory operations to prepare empty locations are needed

SSD 평가 지표

- **Throughput**
  - **MB/s**
  - **Sequential data**와 **random data**에 따라 다름
- **Cost per Capacity**
  - **$/GB**
  - **SSD**의 주요 극복 과제 중 하나
- **Power consumption**
  - **mW**
  - **Active** 및 **idle** 상태에 따라 다름
  - 다채널 구조에서 특히 중요

사용된 Benchmark Tool

- **Crystal Disk Mark 3.0**
  - 기본적인 **sequential / random read / write**의 **throughput**
  - **Seq : sequential (1024KB block size)**
  - **512K : random (512KB block size)**
  - **4K : random (4K block size)**
  - **4K QD32 : random w/ NCQ depth 32 (4K block size)**
    - **32 consecutive 4K data transactions**
- **PCMark Vantage**
  - 특정 시나리오 별 **throughput**
  - **Streaming game, importing photo, starting OS** 등

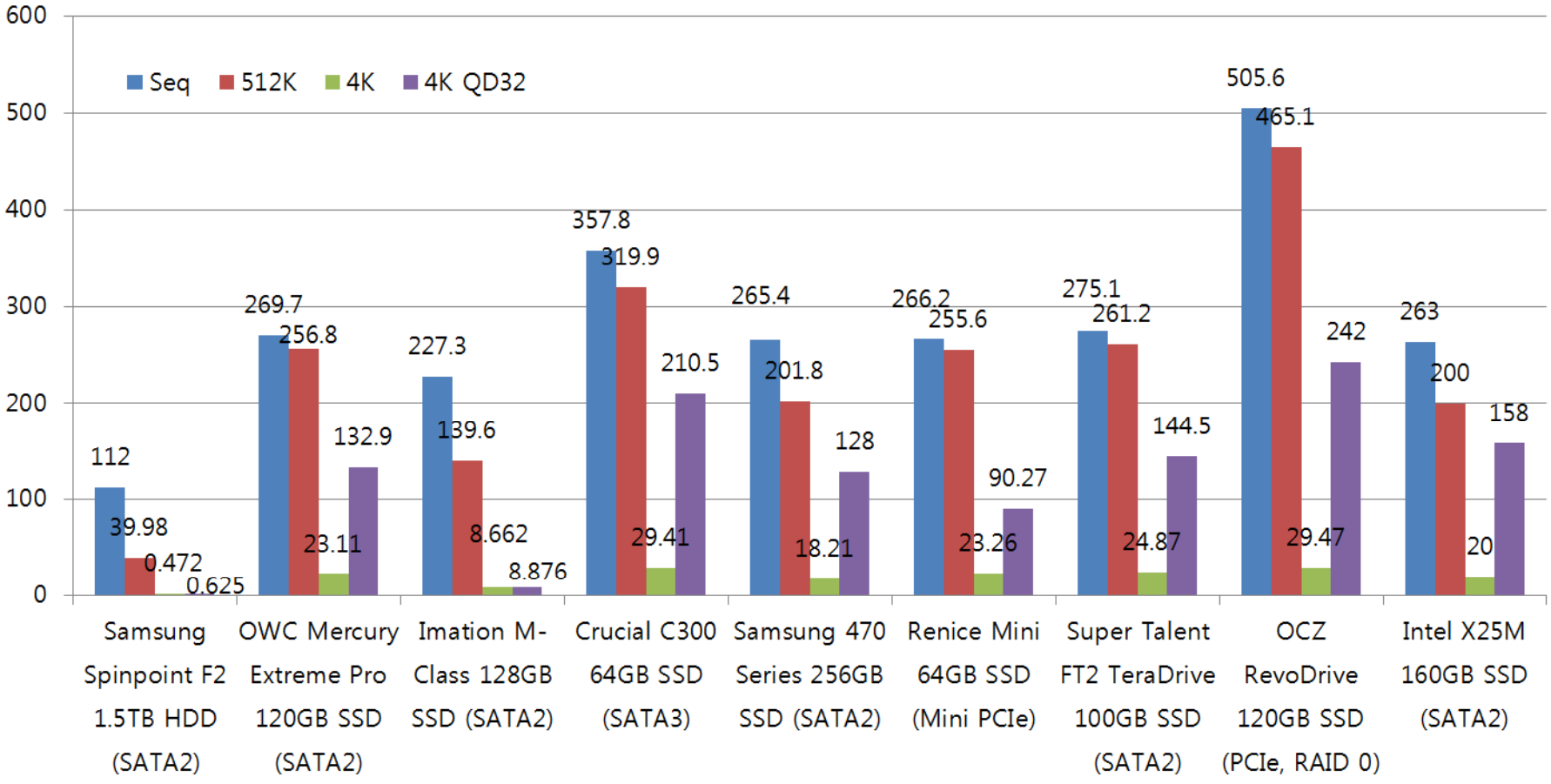
Benchmark 대상 디스크

❑ Benchmark 대상 디스크

Company	Product	Capa.	I/F	Ch.	Controller
Samsung	Spinpoint F2 HDD	1.5TB	SATA 2	-	Samsung
OWC	Mercury Extreme Pro	120GB	SATA 2	16	Sandforce
Imation	M-Class	128GB	SATA 2	16	Phison
Crucial	C300	64GB	SATA 3	16	Marvell
Samsung	470 Series	256GB	SATA 2	16	Samsung
Renice	Mini	64GB	PCIe	8	Sandforce
SuperTalent	FT2 TeraDrive	100GB	SATA 2	16	Sandforce
OCZ	RevoDrive	120GB	PCIe RAID 0	32	Sandforce
Intel	X25-M	160GB	SATA2	10	Intel

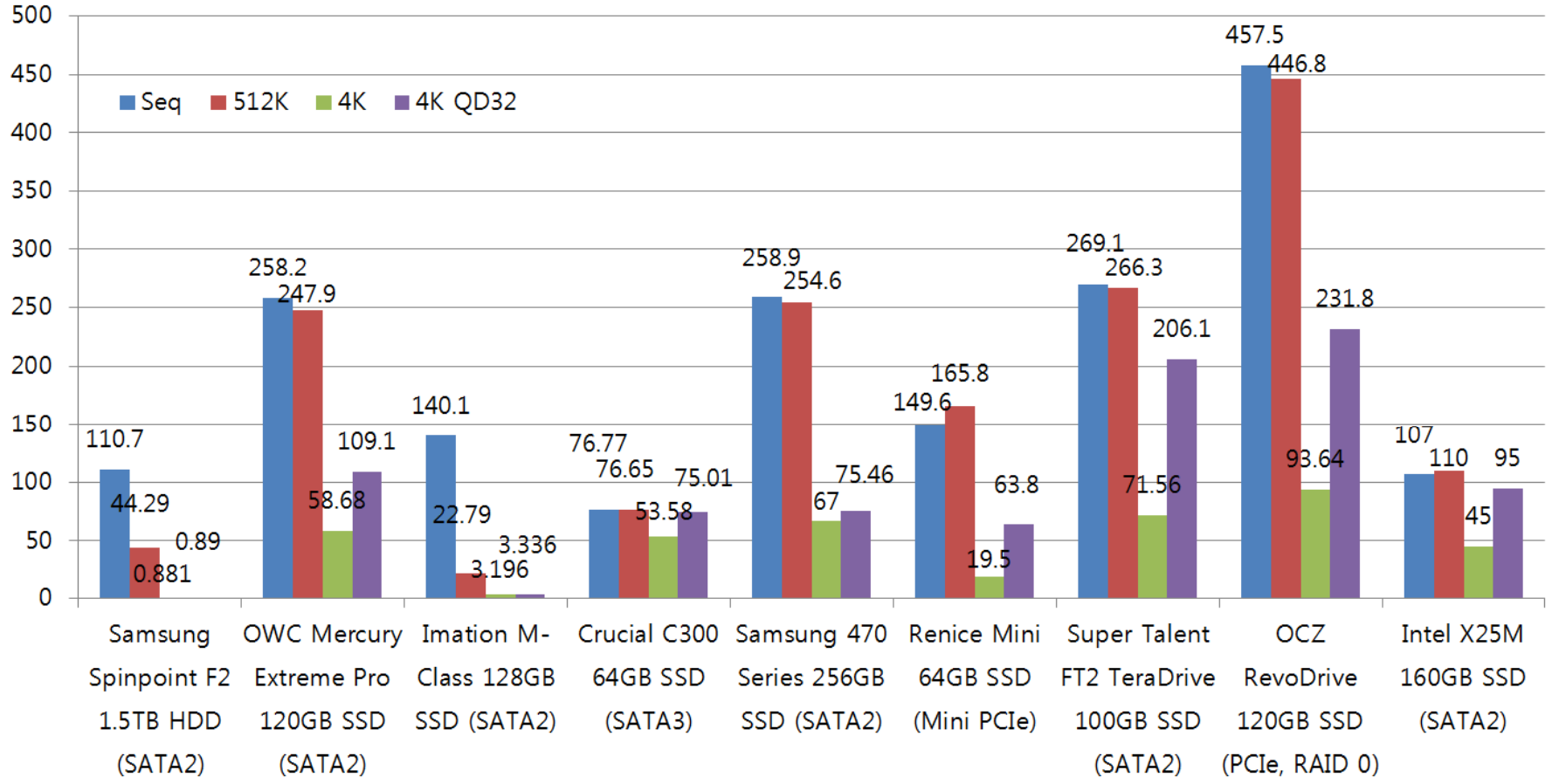
SSD Benchmark (1)

❑ Crystal Disk Mark 3.0 – Read (MB/s)



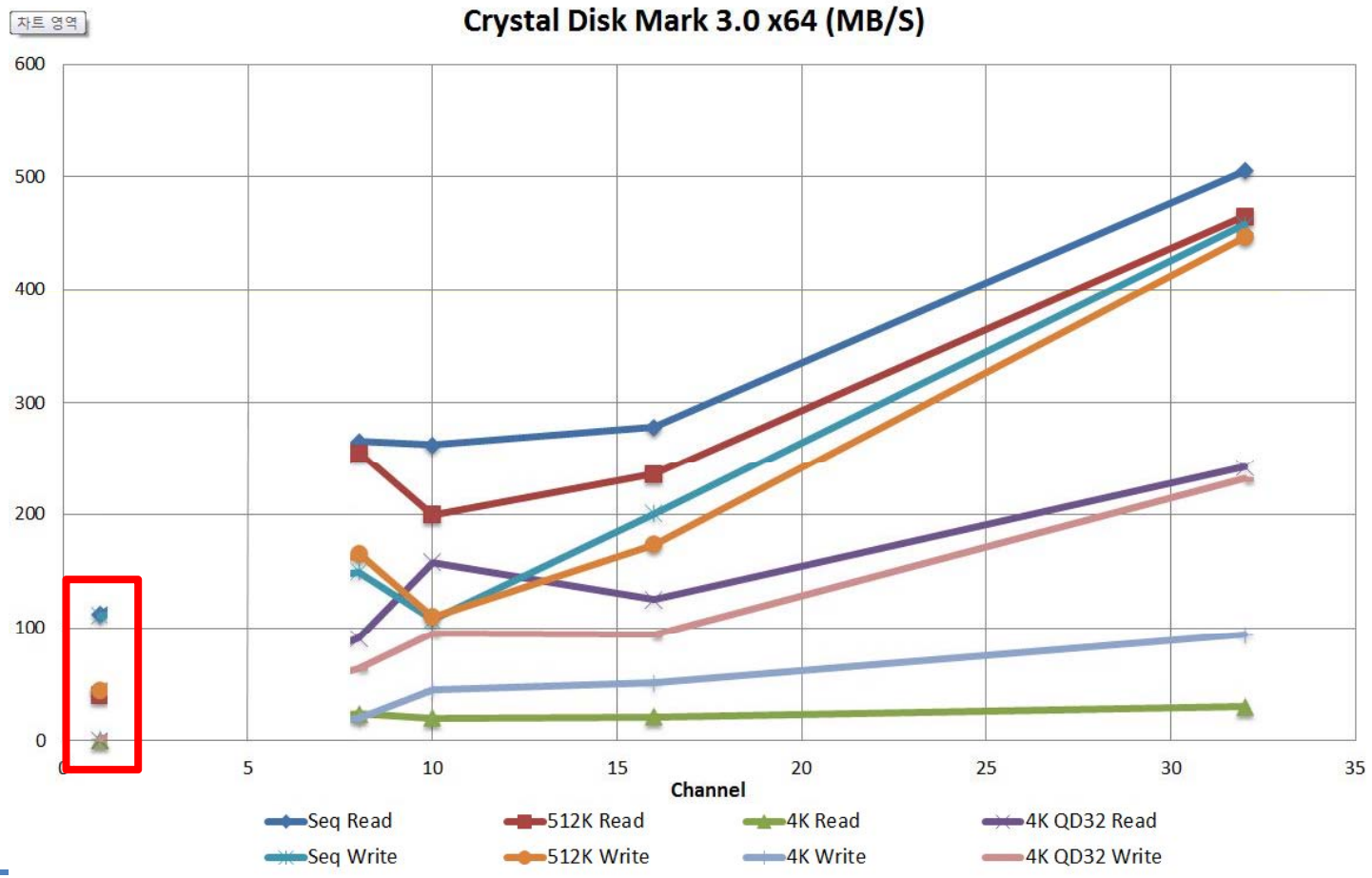
SSD Benchmark (2)

❑ Crystal Disk Mark 3.0 – Write (MB/s)



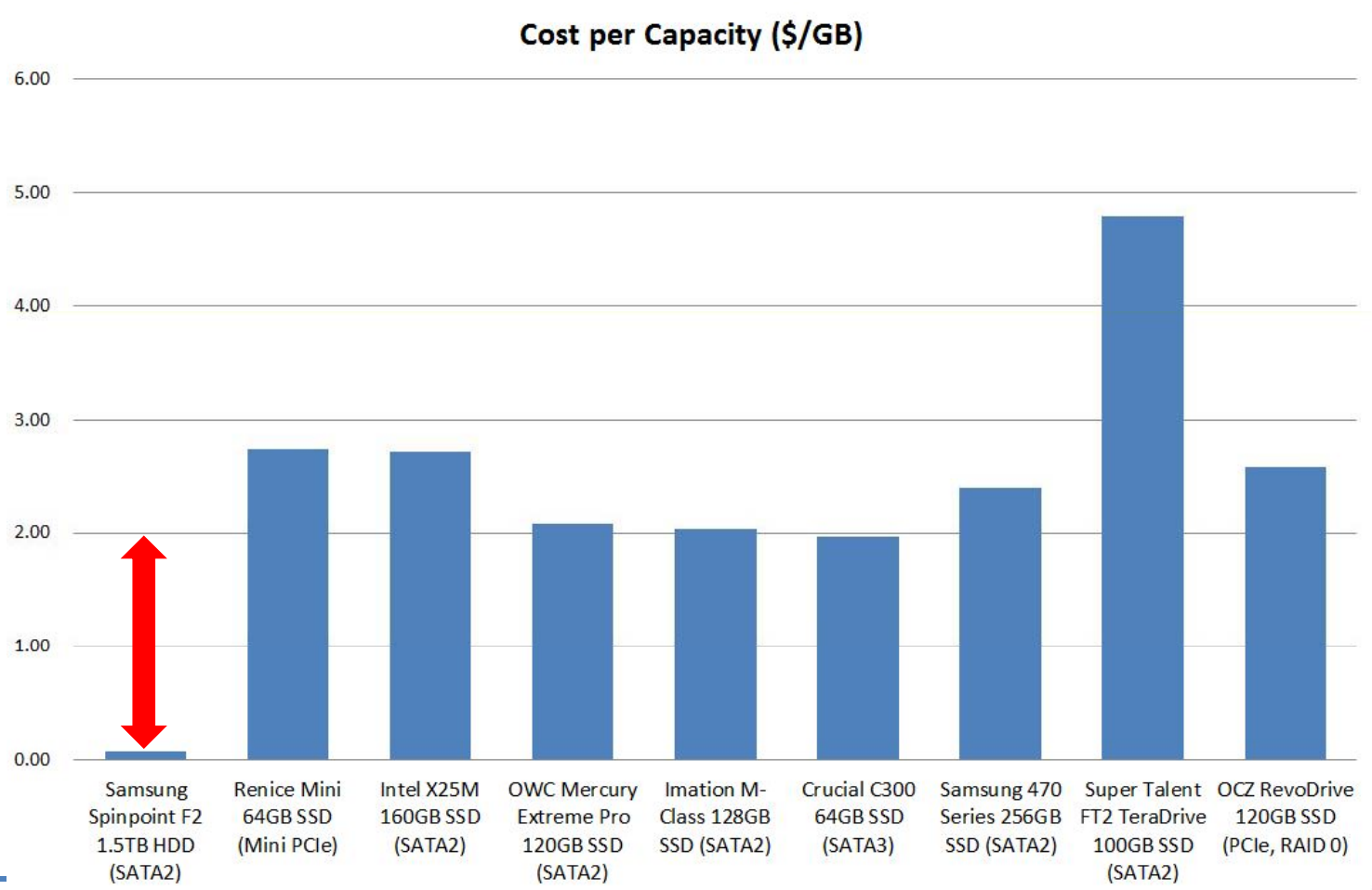
SSD Benchmark (3)

- **Channel 개수 효과 – Synthetic trace**



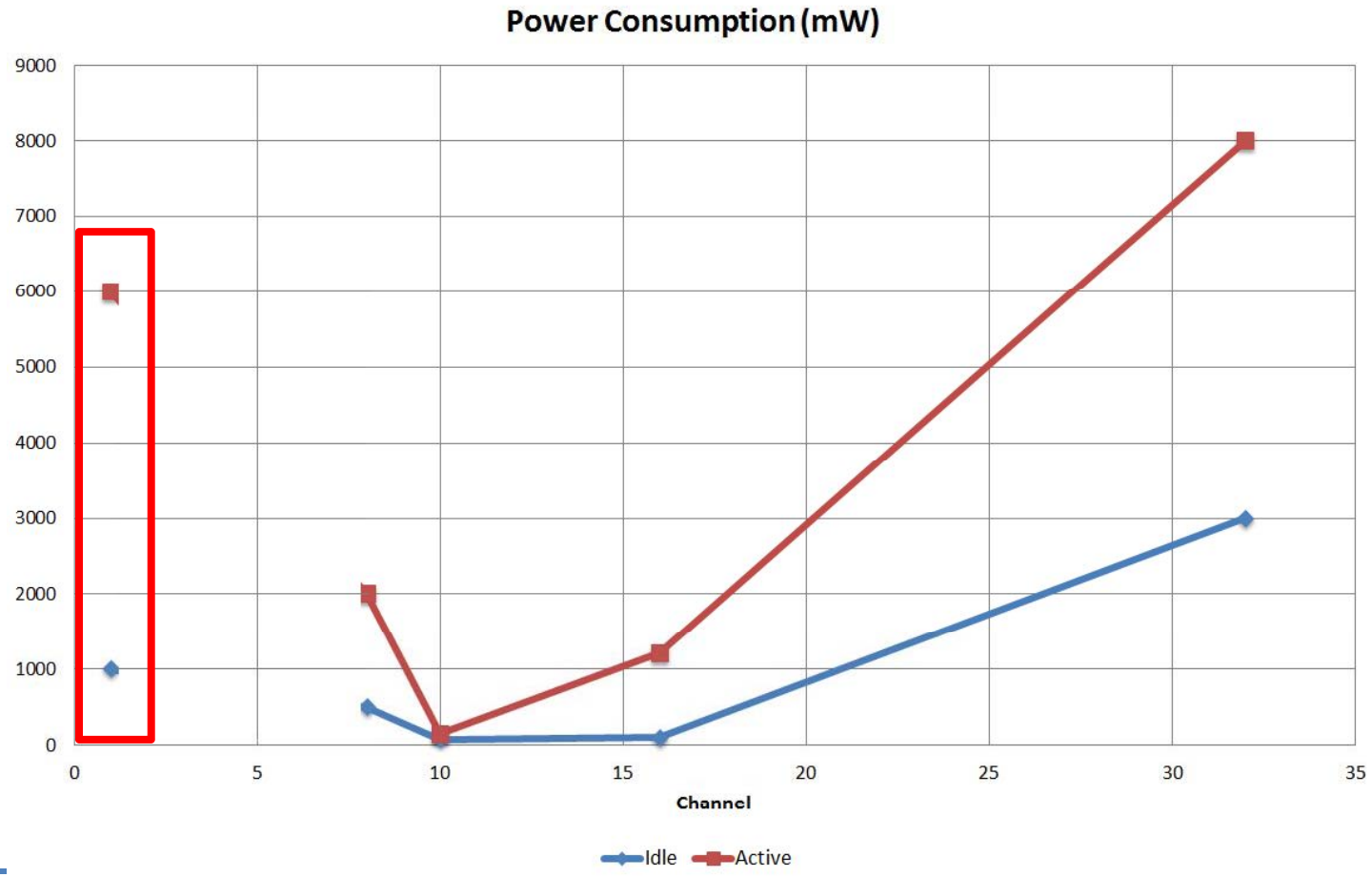
SSD Cost

❑ Cost per Capacity



SSD Power Consumption

❑ Active vs Idle

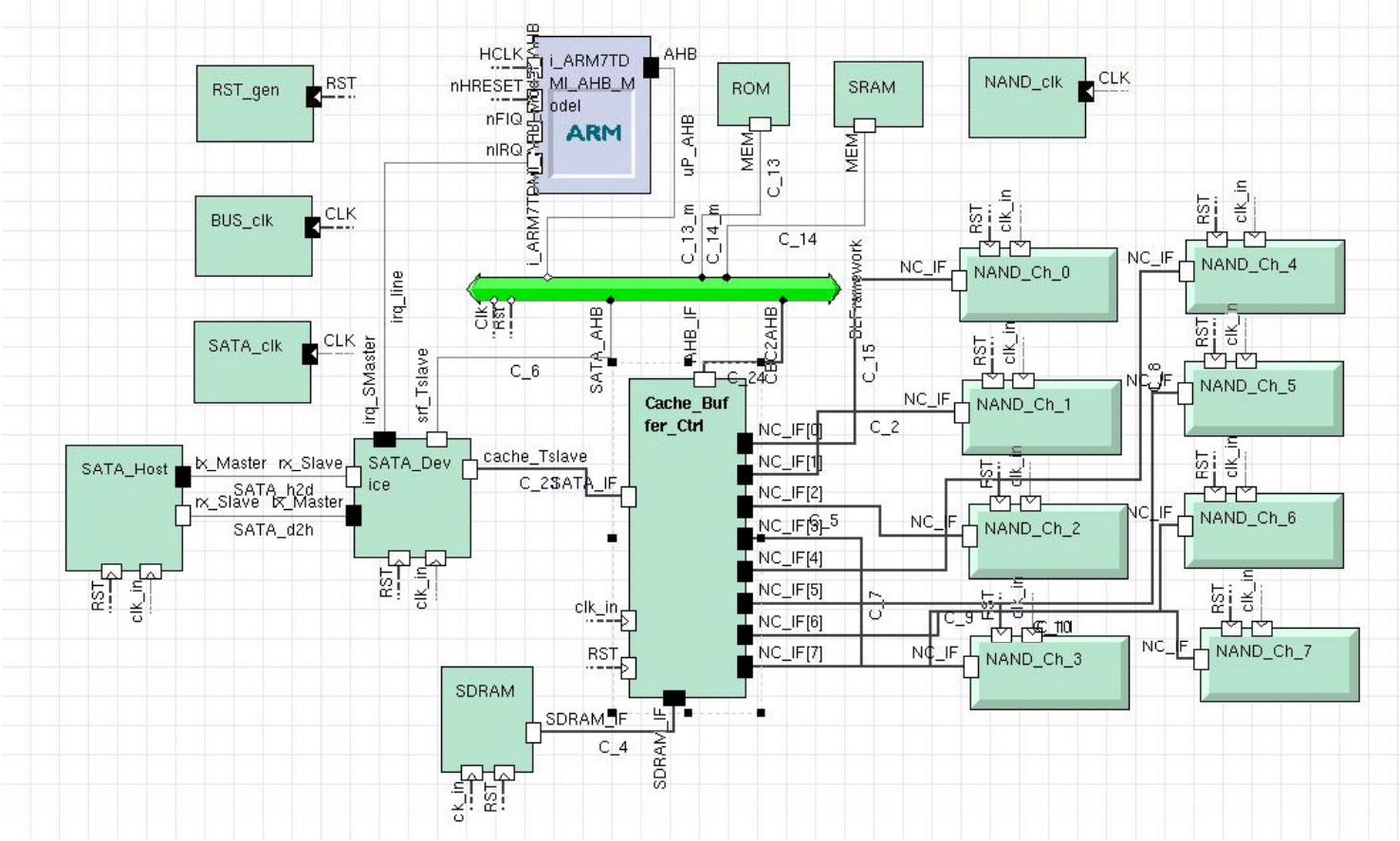


DTL의 SSD 연구 Infra 기술

- **SSD simulator**
  - **C/C++, SystemC**와 같은 상위 수준 언어를 이용
  - **read()** 또는 **write()**와 같은 **transaction** 단위 모델링
  - 빠른 구현 및 **RTL**보다 **10~1000**배 빠른 시뮬레이션 가능
- **FTL**
  - 알고리즘 구현 후 이를 **SSD** 플랫폼에 **porting** 완료 수준
  - **BAST**를 중심으로 여러 **mapping** 알고리즘 가능
  - **Garbage collection** 및 **wear-leveling** 구현
- **RTL IP** 개발
  - **DDR DRAM / NAND memory controller**
  - **Cache buffer controller**
- **New architecture**
  - **Multi-core** 기반 고속 구조 연구

SSD 구조 연구 (1)

- 개발한 **SSD TLM platform**의 구조



SSD 구조 연구 (2)

- 개발한 **TLM platform**의 특징
  - **FTL**을 수행하는 프로세서 **path**와 **disk data path** 분리 구조
    - **Data path**는 **bus**가 아닌 **point-to-point** 연결로 전송 속도 향상
    - 전체 **latency** 향상에 기여
  - **Multi-channel / multi-way** 지원
    - **Sequential access**에 효과적
  - **DRAM** 캐시 버퍼
    - **NAND**의 성능이 전체 **SSD** 성능의 **dominant part**
    - 상대적으로 느린 **NAND** 영향 최소화를 위해 **DRAM** 캐시 버퍼 채용
    - **DRAM**에 **meta-data** 및 **data**를 버퍼링하여 캐시로 활용
  - 캐시버퍼 동작의 중요성
    - **DRAM**과 **NAND**를 동시에 사용하여 중첩효과를 극대화
    - 보조 전원 및 **write-back** 정책을 사용하여 **early write confirm**

FTL 연구 수준 (1)

- **BAST/FAST** 알고리즘 기반의 **FTL** 개발
  - **Data block / log block**을 사용하는 **hybrid mapping**
  - **Block-level**보다 고성능, **page-level**보다 저용량
- **Channel / way interleaving**
  - 단일 **NAND** 대상의 **BAST**를 복수 **channel/way** 구조에 적용
  - 각 **channel/way**에 **interleaved** 형태로 저장되도록 주소 매핑

- **Free block queue**
  - **Wear-leveling**을 위해 **free block**을 관리하는 **queue**
  - 최근 **erase**된 **block**을 나중에 사용함으로써 **wear-leveling**

FTL 연구 수준 (2)

- **On-demand mapping table loading**
  - **FTL**이 사용하는 **SRAM**의 크기는 현실적으로 제한적
  - 대용량 **SSD**의 **mapping table** 전체를 **SRAM**에 **load** 불가
  - **SRAM ↔ DRAM ↔ NAND** 계층구조에서 필요한 부분만 **load**
  - **Mapping table** 전체는 **NAND**의 여러 **page**에 걸쳐 저장
  - 필요한 **mapping table**의 **fragment**를 **page** 단위 **load**
  - **SRAM**에 있는 **fragment**는 다양한 **replacement** 적용 가능
    - **Ex) Round-Robin, Least Recently Used, Least Frequently Used**

SSD 구조 연구 향후 지향점

- **SSD simulator** 구조 탐색 기능 확장
  - **SATA NCQ**를 최대한 활용하는 **SSD scheduling**
  - **PCIe, USB 3.0** 등 차세대 **I/F** 적용
  - **Data** 캐시버퍼와 **meta-data** 캐시버퍼 분리 및 정책 차별화
  - **SLC/MLC** 및 **block/page** 크기 등 **NAND** 구조 다변화
  - **DDR synch NAND, PRAM** 등 다양한 **NVRAM** 지원

- **SSD simulator** 속도 향상
  - 다양한 **SSD** 구조에 대한 **timing model** 도출
  - 개발된 **model**은 **Fast and Accurate simulator**에 적용
  - **Power model** 장착

FTL 연구 향후 지향점

- 다양한 **FTL** 알고리즘 구현
  - **Super-block FTL, LAST, KAST** 등의 발표된 알고리즘
  - 새로운 **FTL** 알고리즘 개발
  - **Meta-data loading** 관련 분석

- **Idle-time**을 이용한 성능 향상
  - **Idle-time**을 수학적으로 예측
  - **Idle-time**에 수행하는 **task**의 **scheduling**
    - **Ex) garbage collection**

- **Power-aware SSD simulator**
  - **Peak / average power management**
  - **Power failure** 시 보조 전원을 통한 **backup/recovery** 기법

Summary

- Data-intensive applications are getting popular
  - Requires high-performance data storage and communication
  - Power is also a critical factor
- On-chip interconnection network
  - Cascaded crossbar network is dominant
  - Automated method is essential to cope with the increased complexity
- Mass-storage devices
  - SSD will replace HDD soon.
  - Still a lot of issues to be addressed
    - Random access speed / peak power issue / power failure protection / ECC / meta data management …