PPRAM Project/Consortium Summary

Kazuaki Murakami

Dept. CS&CE, Kyushu Univ.

http://kasuga.csce.kyushu-u.ac.jp/~ppram

PPRAM Consortium

http://www.k-isit.or.jp/~ppram

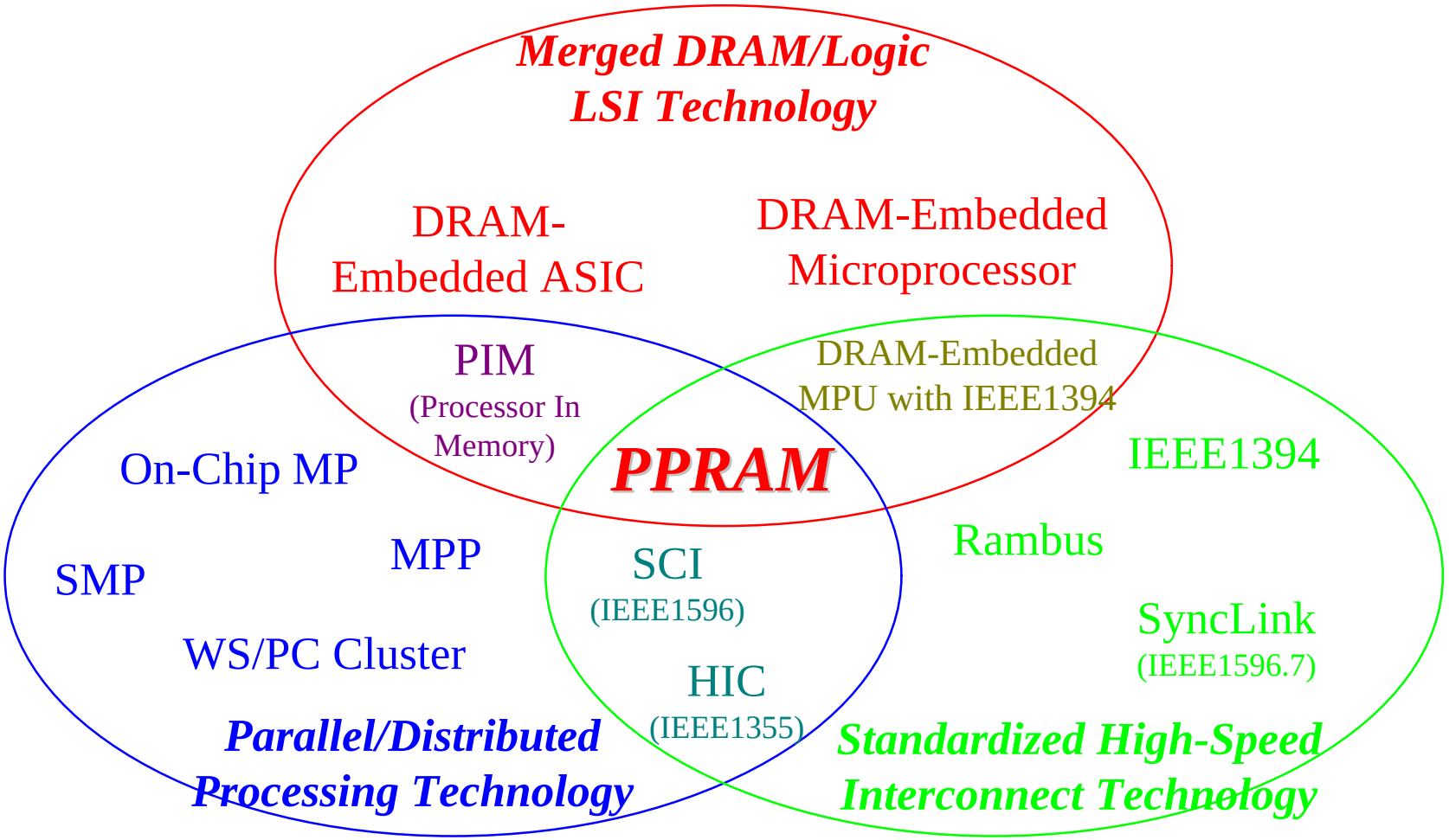
Outline

- What is ***PPRAM*** ?
- Why ***PPRAM*** ?
- Status of ***PPRAM*** Project at Kyushu
- ***PPRAM*** Consortium

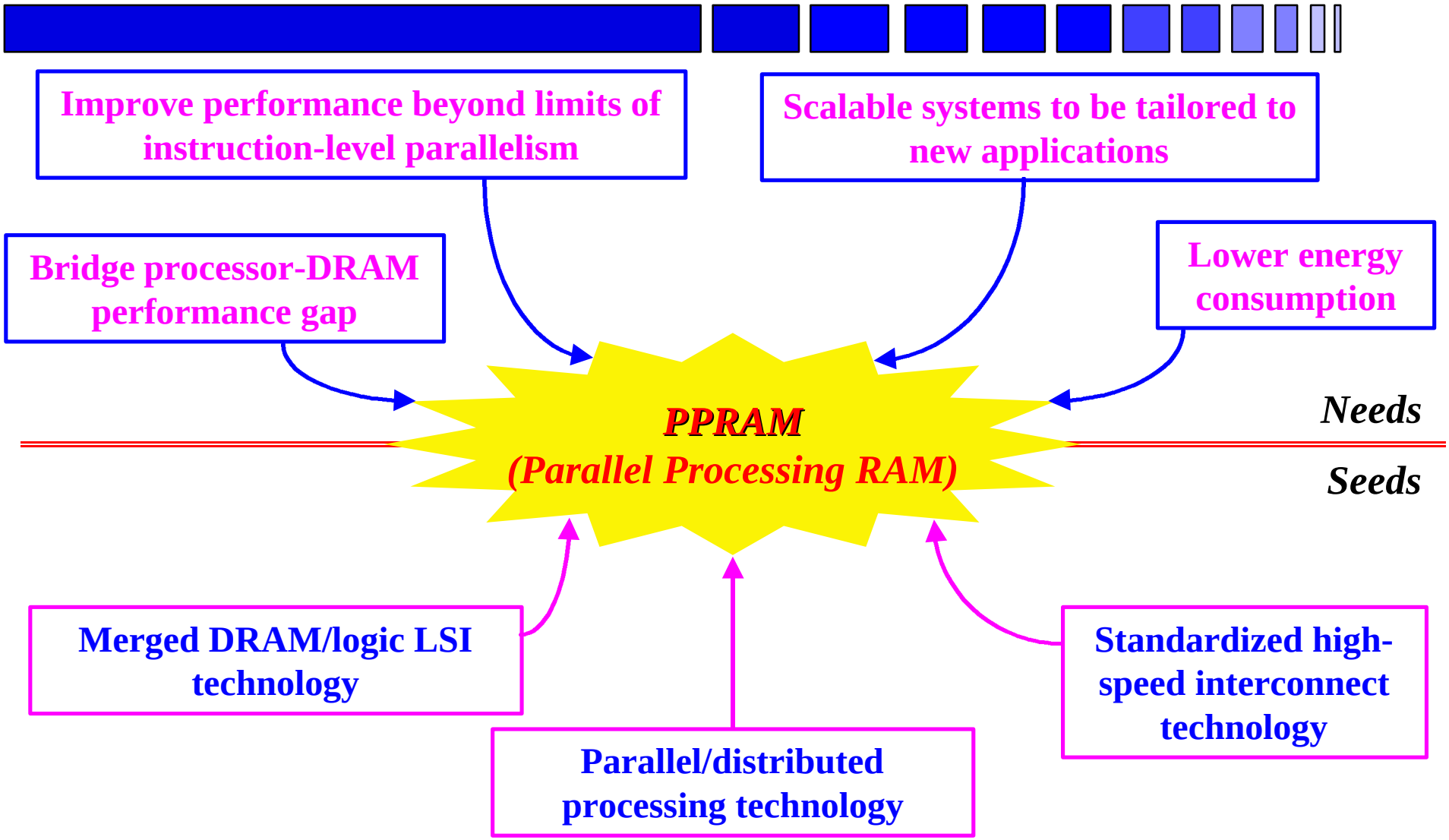
What is ***PPRAM*** ?

- ***PPRAM*** (Parallel Processing Random Access Memory) is an *architectural framework*, or *paradigm*, for microprocessor&memory-based computer systems in the near future.
- Technologies which ***PPRAM*** tries to utilize and exploit include:
  - Merged DRAM/logic LSI technology
  - Parallel/distributed processing technology
  - Standardized high-speed interconnect technology

Technologies ***PPRAM*** Stands on



Multiple Motivations for ***PPRAM***



Goals of ***PPRAM*** (1 of 2)

Improve *cost/energy/performance* over conventional system organizations by means of the following methods

- Bridge *processor-DRAM performance gap* via low on-chip memory latency and ultra-high on-chip memory bandwidth

  ⇐ Merged DRAM/logic LSI technology

- Improve overall performance beyond *the limit of instruction-level parallelism* via multitasking and multiprocessing

  ⇐ Parallel/distributed processing technology

Goals of ***PPRAM*** (2 of 2)

- Have ***PPRAM***-based systems be *scalable with respect to size*, *functionality*, *and performance*, by enabling system designers to make a system of any scale by combining the necessary number and kind of ***PPRAM*** chips

  ⇐ High-speed standard interconnect technology

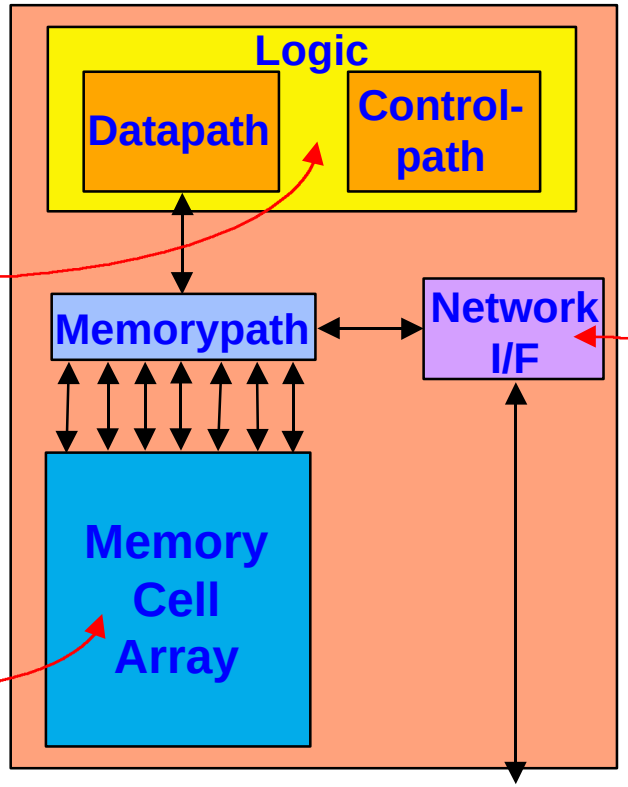
- Improve *energy consumption* of the memory system, by reducing the necessity to drive high-capacitance off-chip buses and by optimizing the number of sense-amplifiers to be activated simultaneously

  ⇐ Merged DRAM/logic LSI technology

What is ***PPRAM*** ?
- Definition of ***PPRAM*** Node -

PPRAM Node = Application-Specific *Logic* +
Application-Specific *Memory* +
Standard *Communication*

Zero or more
- General-purpose processors, and/or
- Special-purpose processors, and/or
- Application-specific logic, and/or
- FPGA, and/or ...



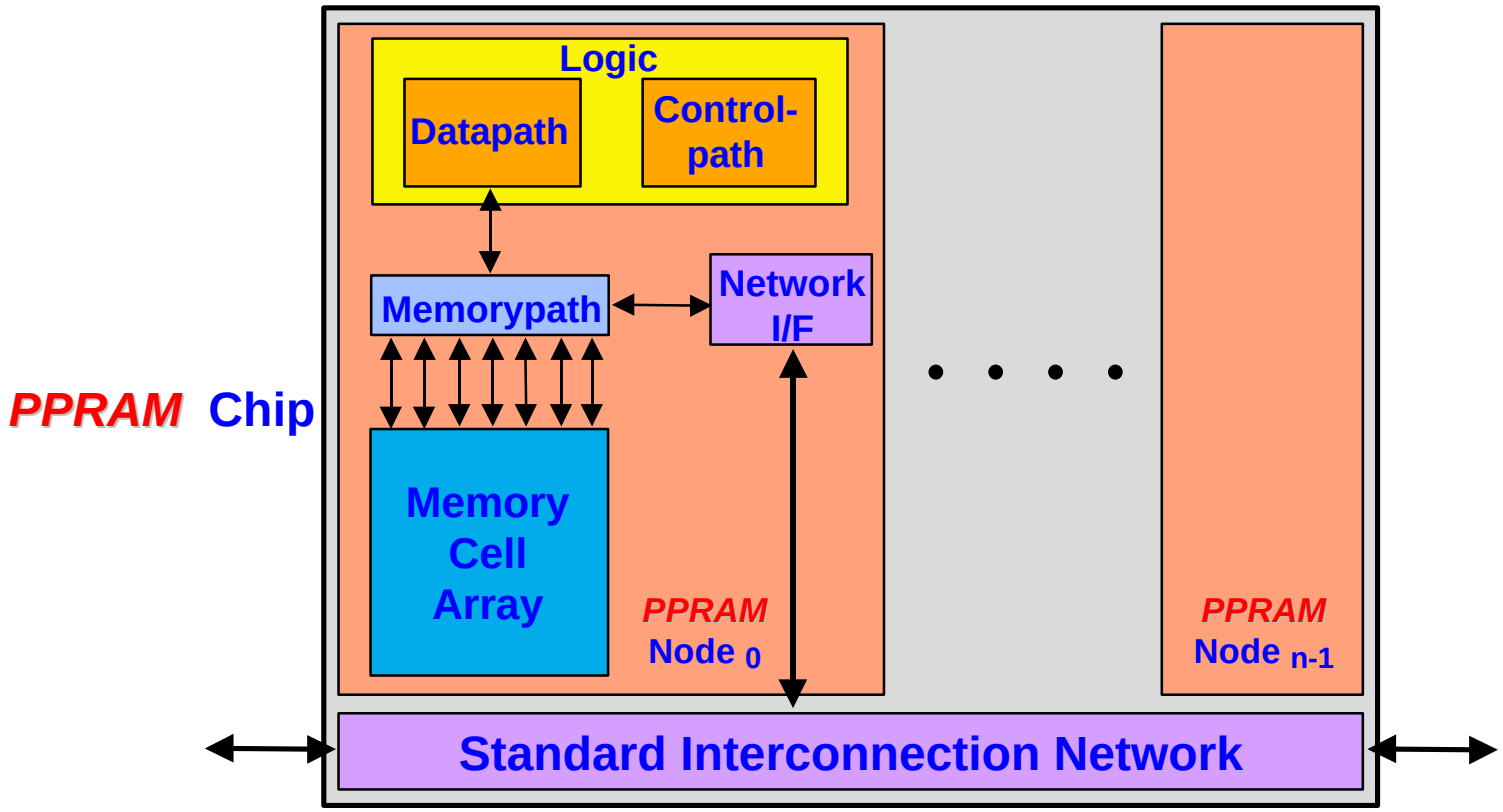
PPRAM **Node**

Zero or more bits of
- DRAM, and/or
- SRAM, and/or
- Flash EEPROM, and/or
- FRAM, and/or ...

One or more network interfaces based on a standardized communication protocol

What is ***PPRAM*** ?
- Definition of ***PPRAM*** Chip -

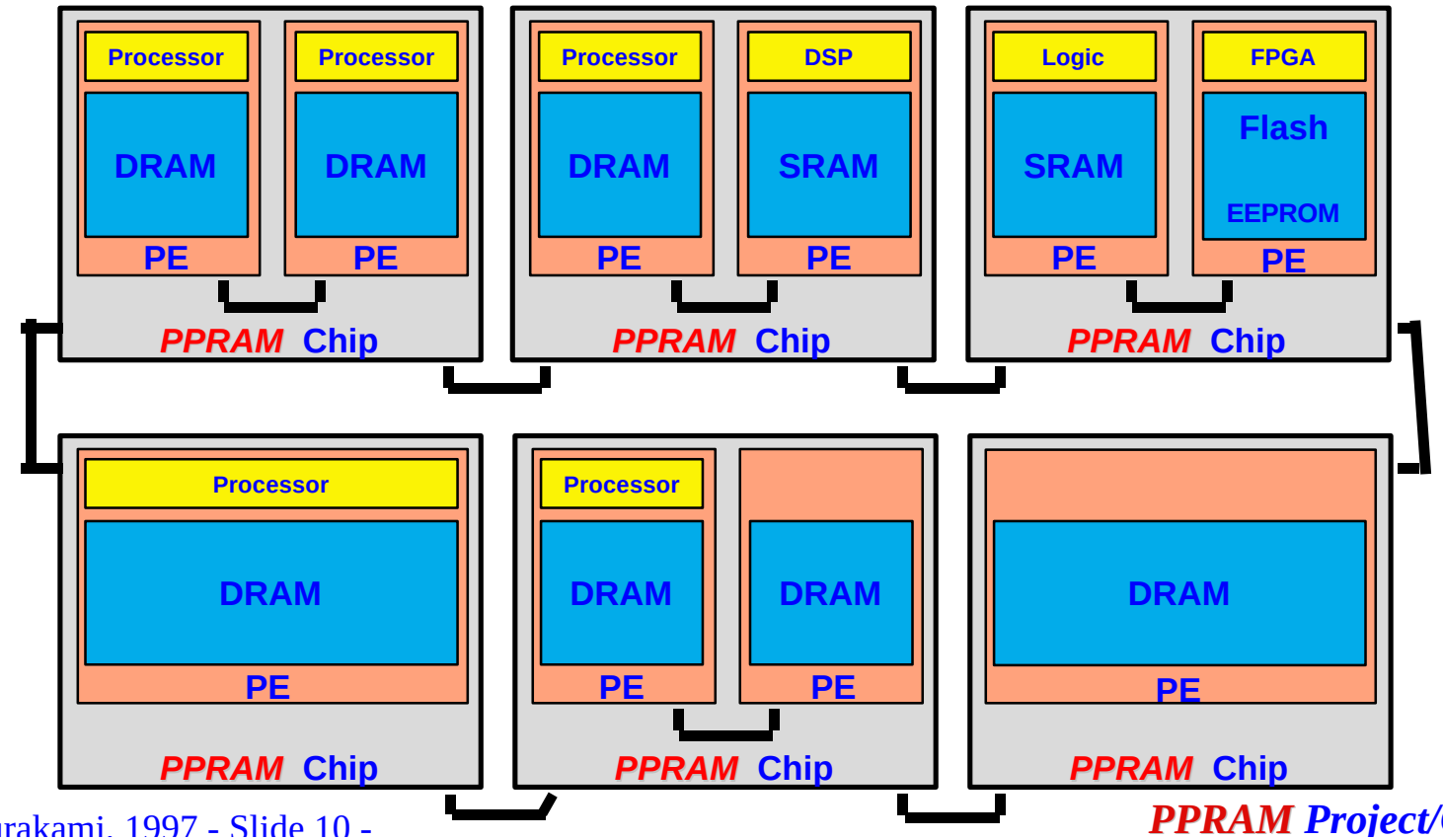
- Network of ***PPRAM*** nodes
  - Integrating one or more ***PPRAM*** nodes on single chip
  - Interconnecting them via the standard network



What is *PPRAM* ?

- Definition of *PPRAM*-Based System -

- Network of *PPRAM* chips
  - Consisting of one or more *PPRAM* chips
  - Interconnecting them via the standard network



Outline

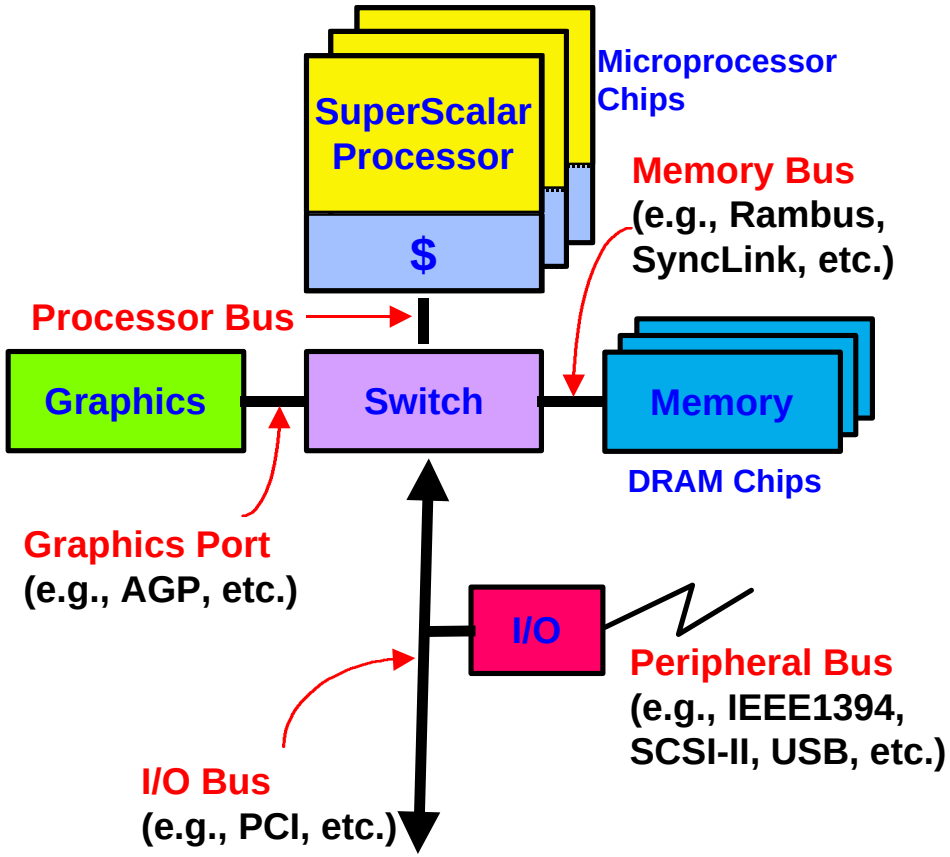
- What is ***PPRAM*** ?
- Why ***PPRAM*** ?
- Status of ***PPRAM*** Project at Kyushu
- ***PPRAM*** Consortium

Why *PPRAM* ?

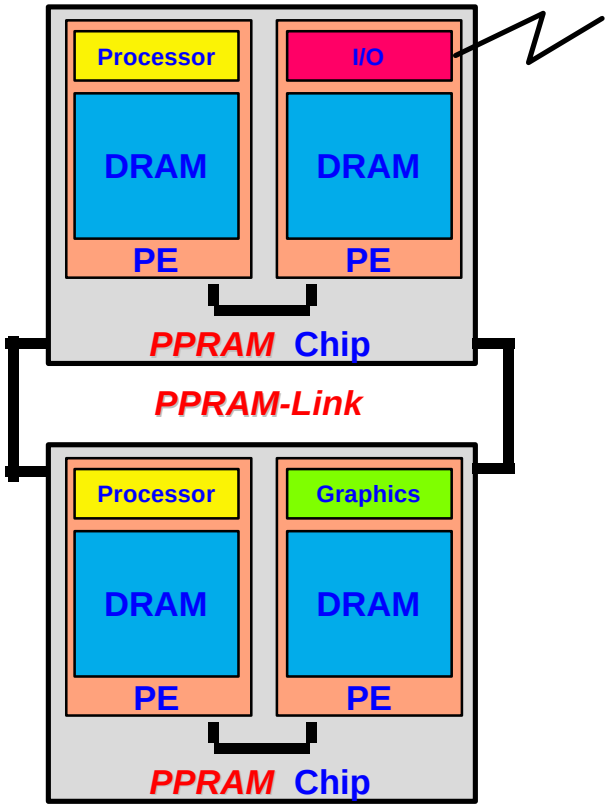
- How shall we partition an entire computer system into multiple chips (if a single chip is not enough) ?
  - We have divided it into microprocessor and DRAM chips for 25 years
- How shall we turn plenty of transistors into performance ?

How to Partition Entire System into Multiple Chips ?

- Traditional partition into separate microprocessor & DRAM chips

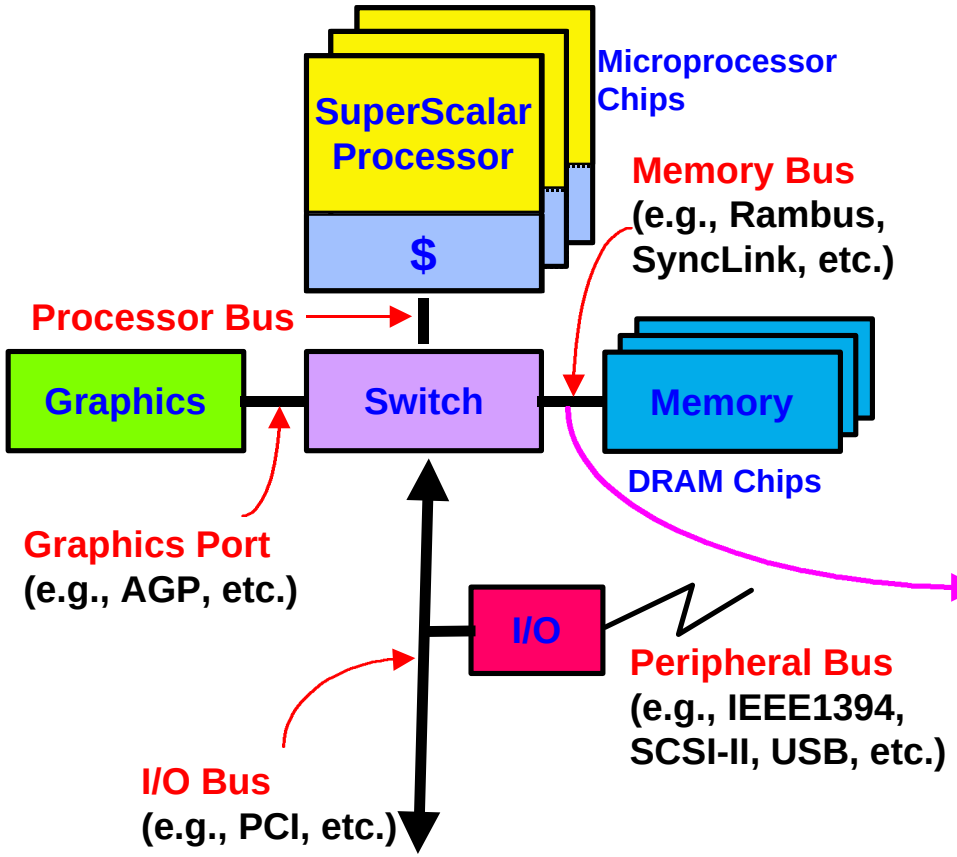


- ***PPRAM*** partition into unified DRAM/processor chips



How Much Memory Bandwidth Is Required ? (1 of 2)

- Traditional partition into separate microprocessor & DRAM chips



Required Memory Bandwidth:

$$RMB = P \times MPC \times CMR \times LS \times CR$$

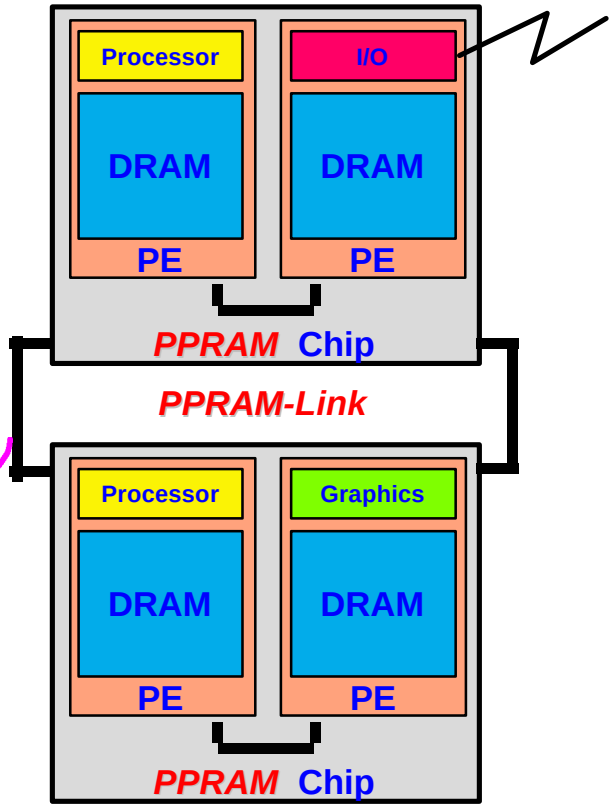
- ***P*** : number of microprocessor chips
- ***MPC*** : memory accesses per clock-cycle
- ***CMR*** : cache-miss rate
- ***LS*** : cache line size
- ***CR*** : clock rate

Ex.)

$$RMB = 2 \times 1.25 \times 5\% \times 64\text{B} \times 500\text{MHz}$$
$$= 4\text{GB/s}$$

How Much Memory Bandwidth Is Required ? (2 of 2)

■ ***PPRAM*** partition into unified DRAM/processor chips



Required Memory Bandwidth:

$$RMB = P \times MPC \times PFR \times PS \times CR$$

- ***P*** : number of ***PPRAM*** chips
- ***MPC*** : memory accesses per clock-cycle
- ***PFR*** : page-fault rate
- ***PS*** : page size
- ***CR*** : clock rate

Ex.)

$$RMB = 2 \times 1.25 \times 0.05\% \times 1024\text{B} \times 500\text{MHz}$$
$$= 640\text{MB/s}$$

How to Turn Plenty of Transistors into Performance ? (1 of 2)

- Enhance performace of single processors
  - from ***Simple*** processor
  - to ***Powerful*** processor
- Increase the number of processors on a single chip
  - from ***Single*** processor
  - to ***Multiple*** processors
- Increase memory size on a chip
  - from the size of SRAM ***Cache-memory***
  - to the size of a whole or part of DRAM ***Main-memory***

How to Turn Plenty of Transistors into Performance ? (2 of 2)

No. of Processors	Single-Processor Performance	Memory Size	Approaches	Examples
Single	Simple	Cache-only	SSC	*RISC*
		Main-memory	SSM	*M32R/D,(Sun)*
	Powerful	Cache-only	SPC	*Superscalar*
		Main-memory	SPM	*IRAM,DataScalar*
Multiple	Simple	Cache-only	MSC	*TI MVP*
		Main-memory	MSM	***PPRAM^R***
	Powerful	Cache-only	MPC	*Micro2000*
		Main-memory	MPM	-

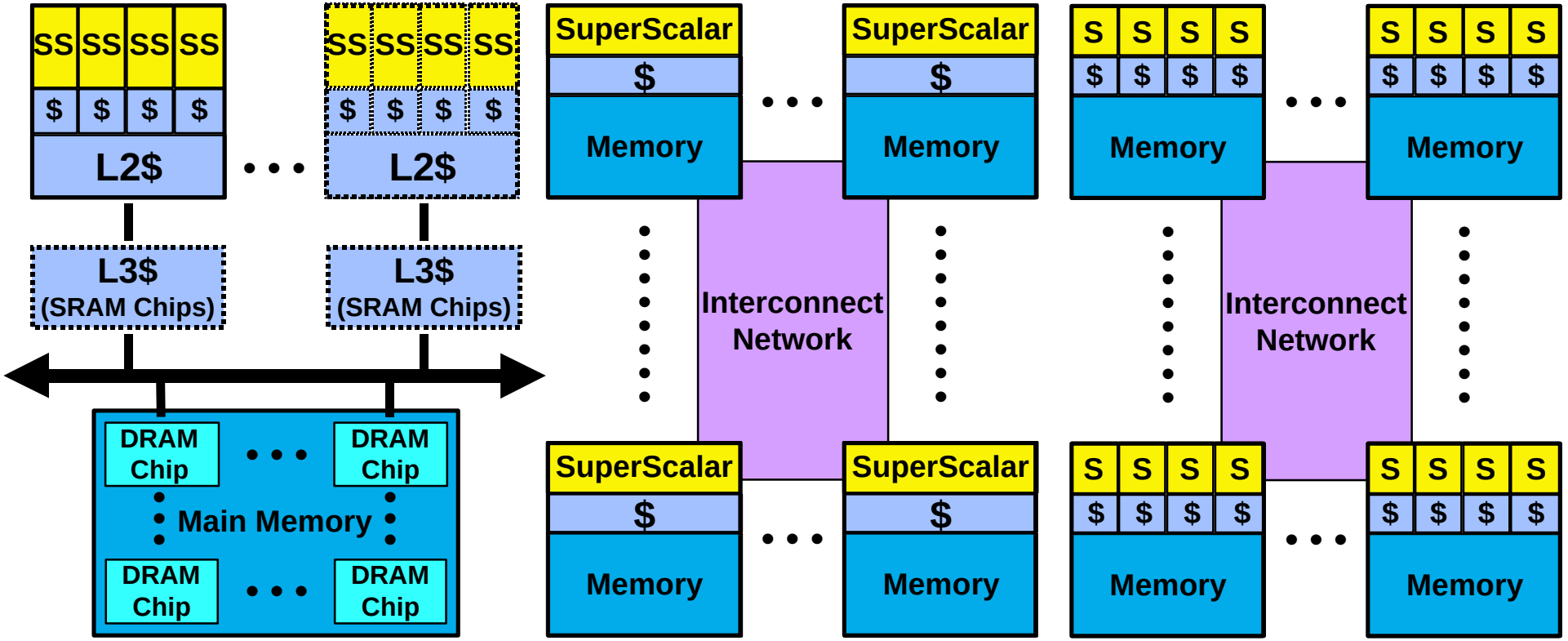
PPRAM approaches

Promising high-end chip architectures

Three Promising Architectures

- MPC vs. SPM vs. MSM -

- **MPC** (**M**ultiple **P**owerful-processors with **C**ache-only)
- **SPM** (**S**ingle **P**owerful-processor with **M**ain-memory)
- **MSM** (**M**ultiple **S**imple-processors with **M**ain-memory)



Performance Modeling

- Program Execution Time:

$$ET = \max_{p}\left(\sum_{t \in T_p} IC_t \times CPI_t \times CCT\right)$$

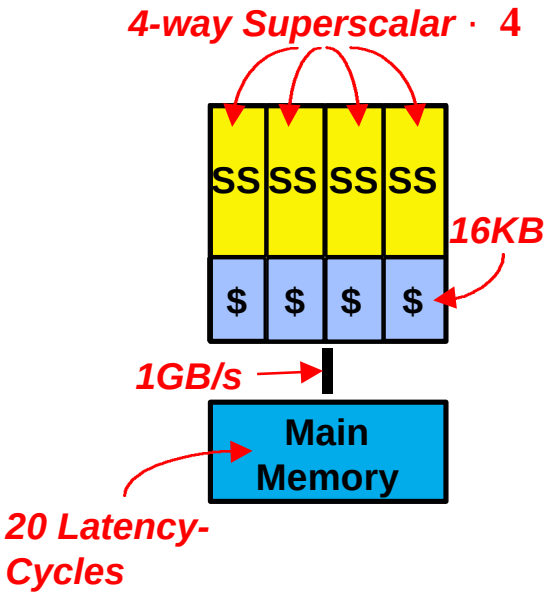
$$\approx \frac{N}{P} \times IC_{mean} \times CPI \times CCT \propto \frac{CPI}{P}$$

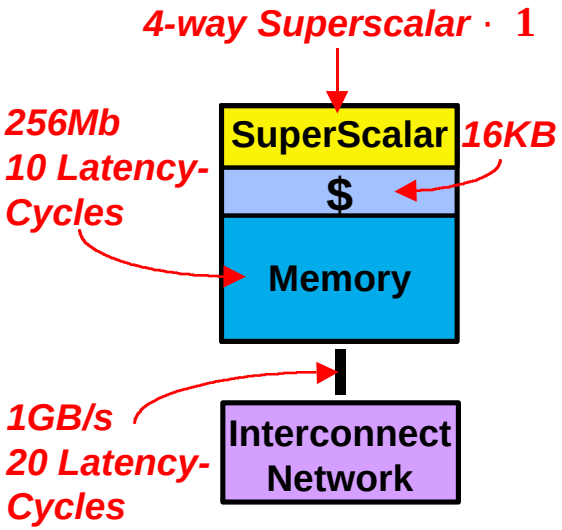
- ***p*** : processor index
- ***t*** : task index
- T_p : set of tasks to be executed by processor ***p***
- IC_t : instruction count on executing task ***t***
- CPI_t : clock-cycles per instruction on executing task ***t***
- ***N*** : total number of tasks to be executed
- ***P*** : number of processors
- IC_{mean} : mean instruction count
- ***CPI*** : clock-cycles per instruction
- ***CCT*** : clock-cycle time

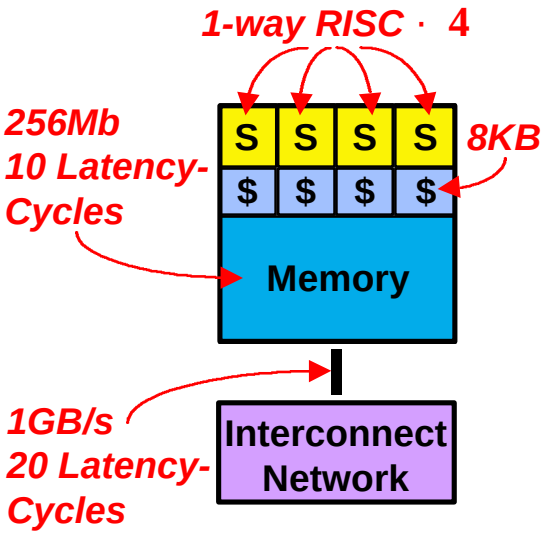
Performance Comparison Conditions

- MPC vs. SPM vs. MSM -

- **MPC** (**M**ultiple **P**owerful-processors with **C**ache-only)
- **SPM** (**S**ingle **P**owerful-processor with **M**ain-memory)
- **MSM** (**M**ultiple **S**imple-processors with **M**ain-memory)



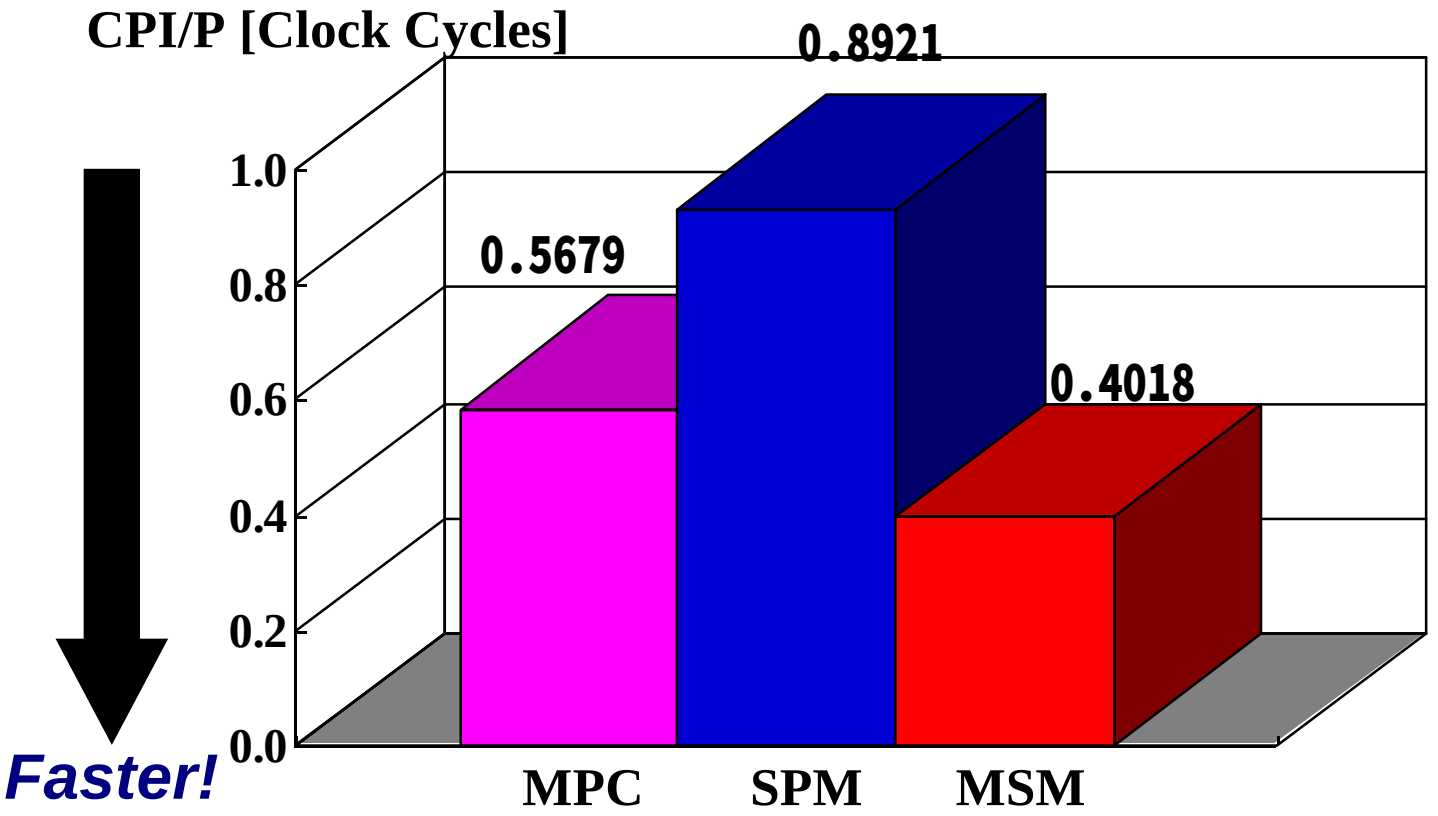




Die Size: ***389mm²*** @0.25µm	Die Size: ***445mm²*** @0.25µm	Die Size: ***427mm²*** @0.25µm

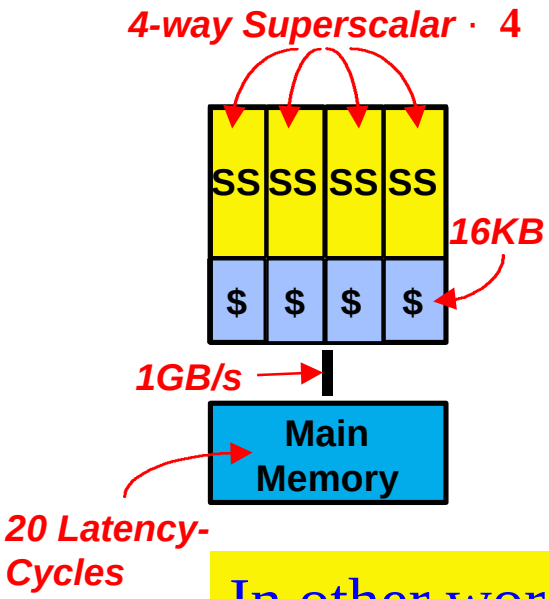
Performance Comparison Results

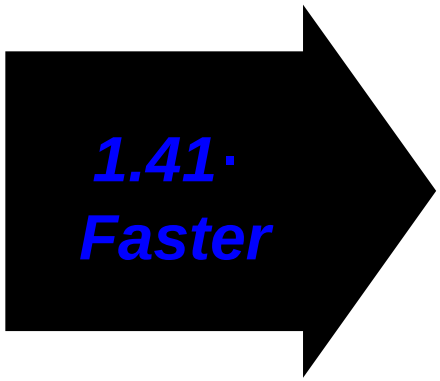
- MPC vs. SPM vs. MSM -



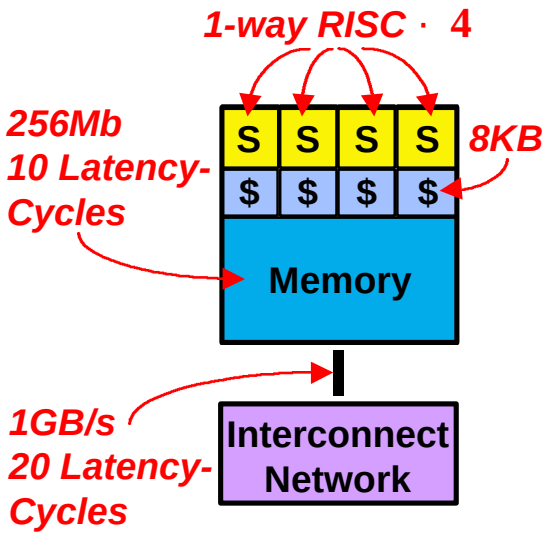
Performance Advantage of MSM (*PPRAM*^{R}) over MPC

- **MPC (Multiple Powerful-processors with Cache-only)**



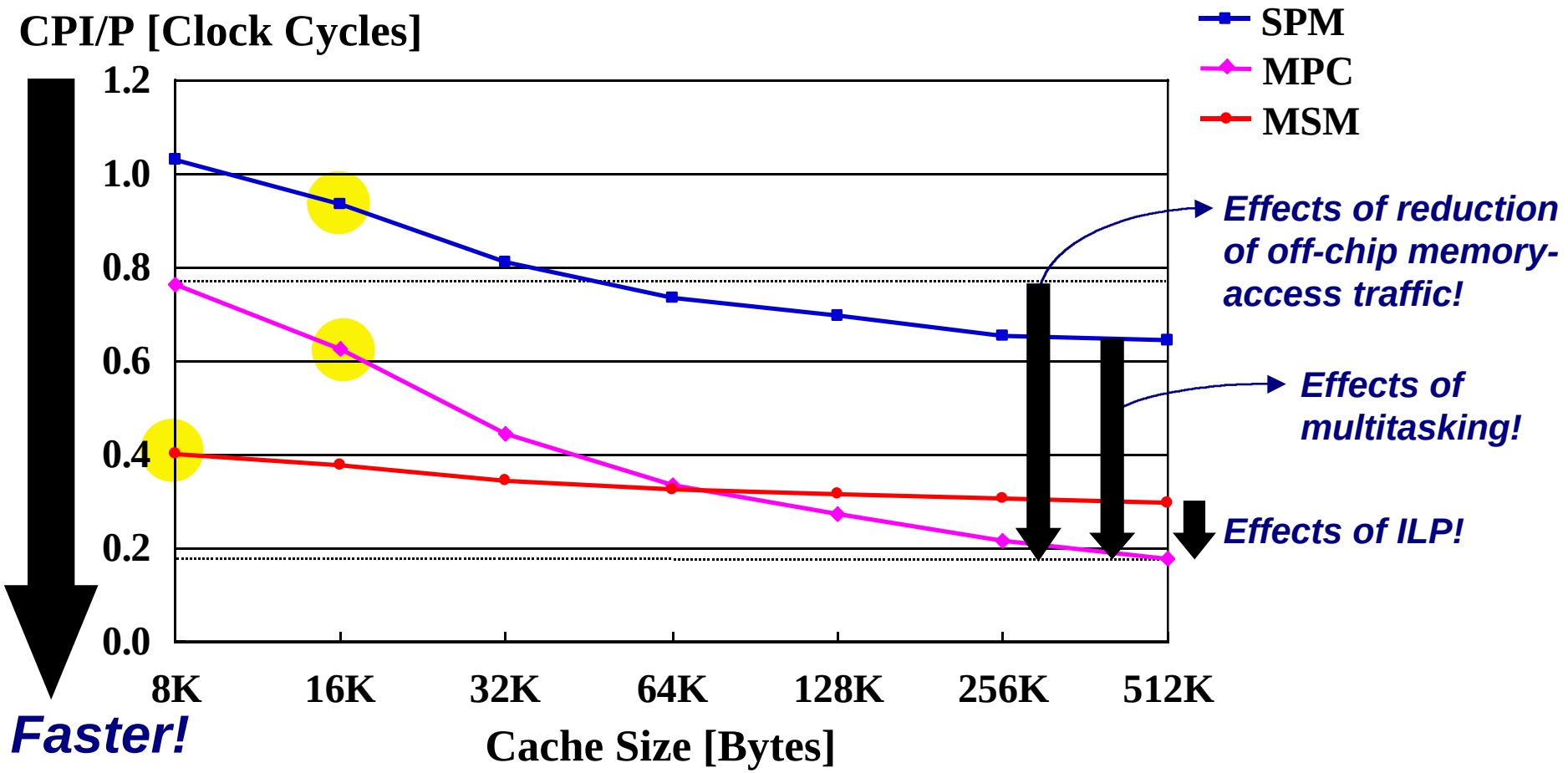


- **MSM (Multiple Simple-processors with Main-memory)**

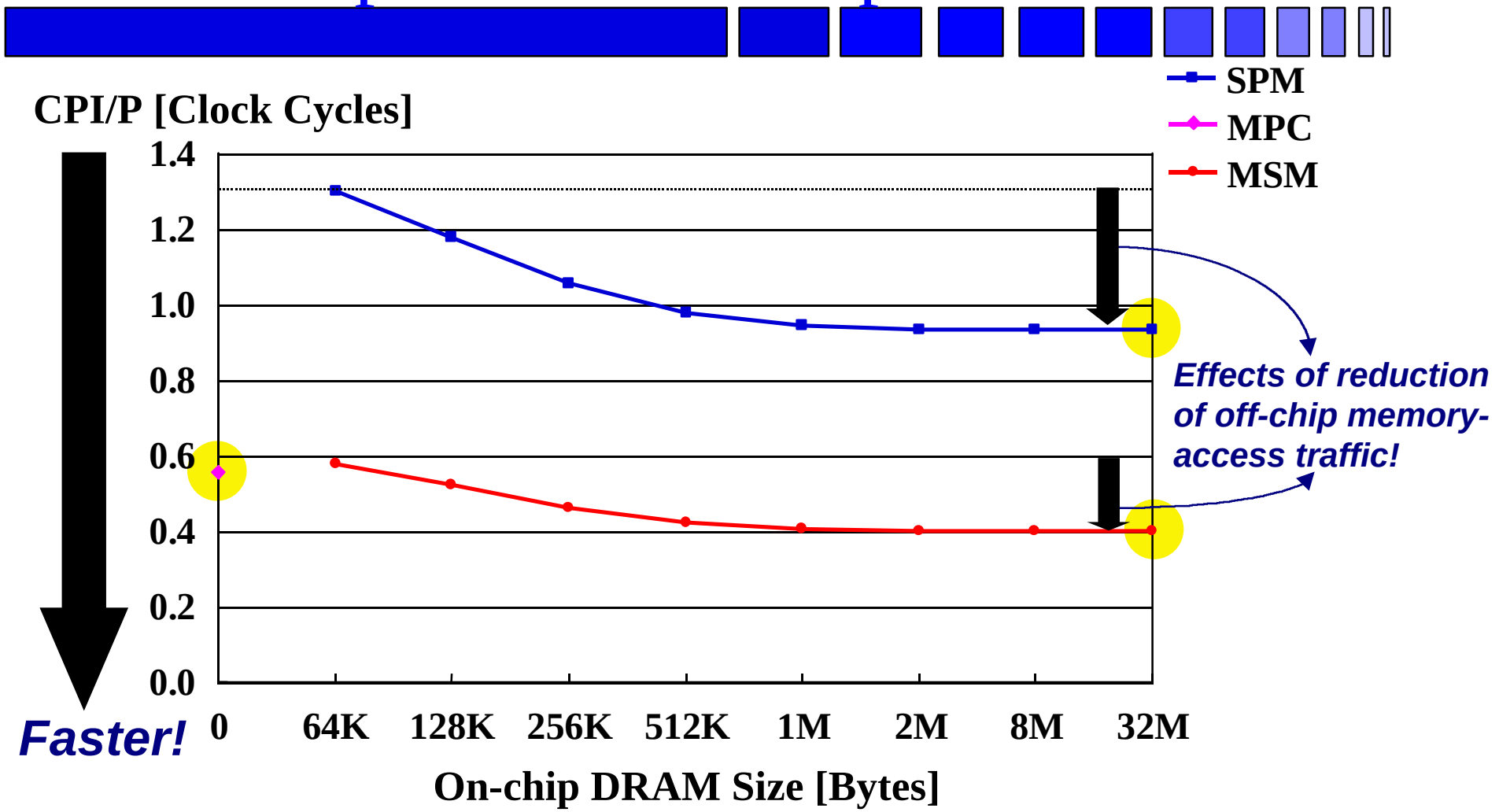


In other words, logic in a hybrid DRAM/logic process has the *performance margin* of ***30% (=1-1/1.41)*** against logic in a logic process

Performance Comparison Results
- Impact of Cache Size -



Performance Comparison Results
- Impact of On-Chip DRAM Size -

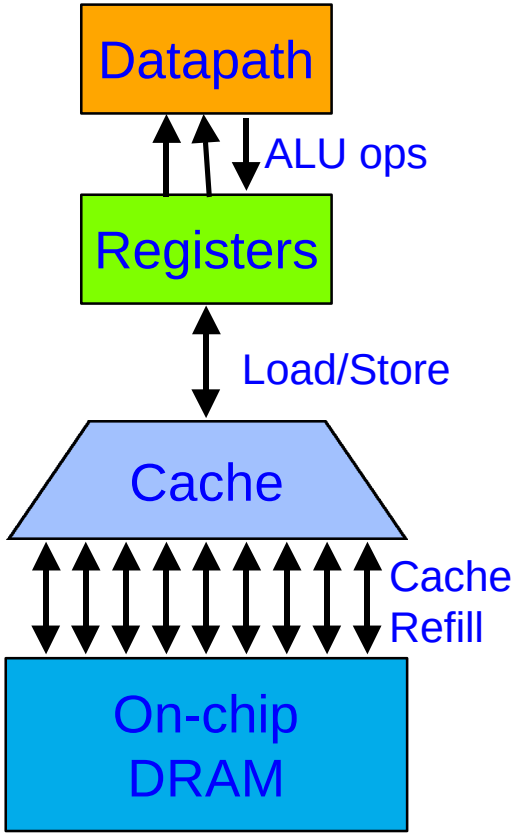


Why ***PPRAM*** ?
- Answers -

- How shall we partition an entire computer system into multiple chips (if a single chip is not enough) ?
  - Partition into unified DRAM/processor (i.e., ***PPRAM***) chips
    - ⇒ *Relieve inter-chip memory bandwidth requirement*
- How shall we turn plenty of transistors into performance ?
  - Put DRAM main memory as much as possible
    - ⇒ *Reduce off-chip memory access traffic*
    - ⇒ *Exploit ultra-high on-chip memory bandwidth*
  - Put multiple simple processors rather than powerful ones
    - ⇒ *Exploit parallelism at higher-level than instruction-level*

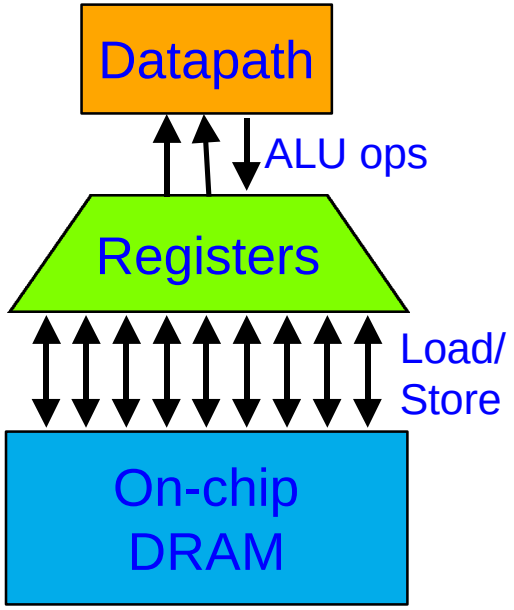
How to Turn High On-Chip Memory Bandwidth into Performance

■ Cache refill



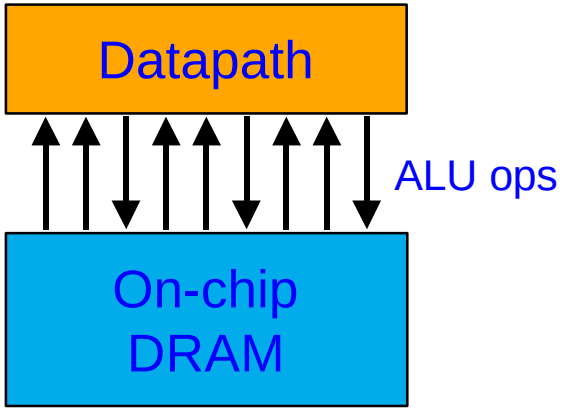
Ex.) ***PPRAM^{R}***

■ Load/Store operations



Ex.) Vector IRAM

■ ALU operations



Ex.) ***PPRAM$_{moe}$***

How to Turn High On-Chip Memory Bandwidth into Performance - Cache Refill -

- Mitsubishi M32R/D

Shimizu, T. et al., A Multimedia 32b RISC Microprocessor with 16Mb DRAM, *ISSCC'96 Digest of Technical Papers*, pp.216-217, Feb. 1996.

How to Turn High On-Chip Memory Bandwidth into Performance - Cache Refill -

- Sun Microsystems

Saulsbury, A. et al., Missing the Memory Wall: The Case for Processor/Memory Integration, *Proc. ISCA'96*, pp.90-101, May 1996.

Performance Evaluation
- Sun Microsystems -

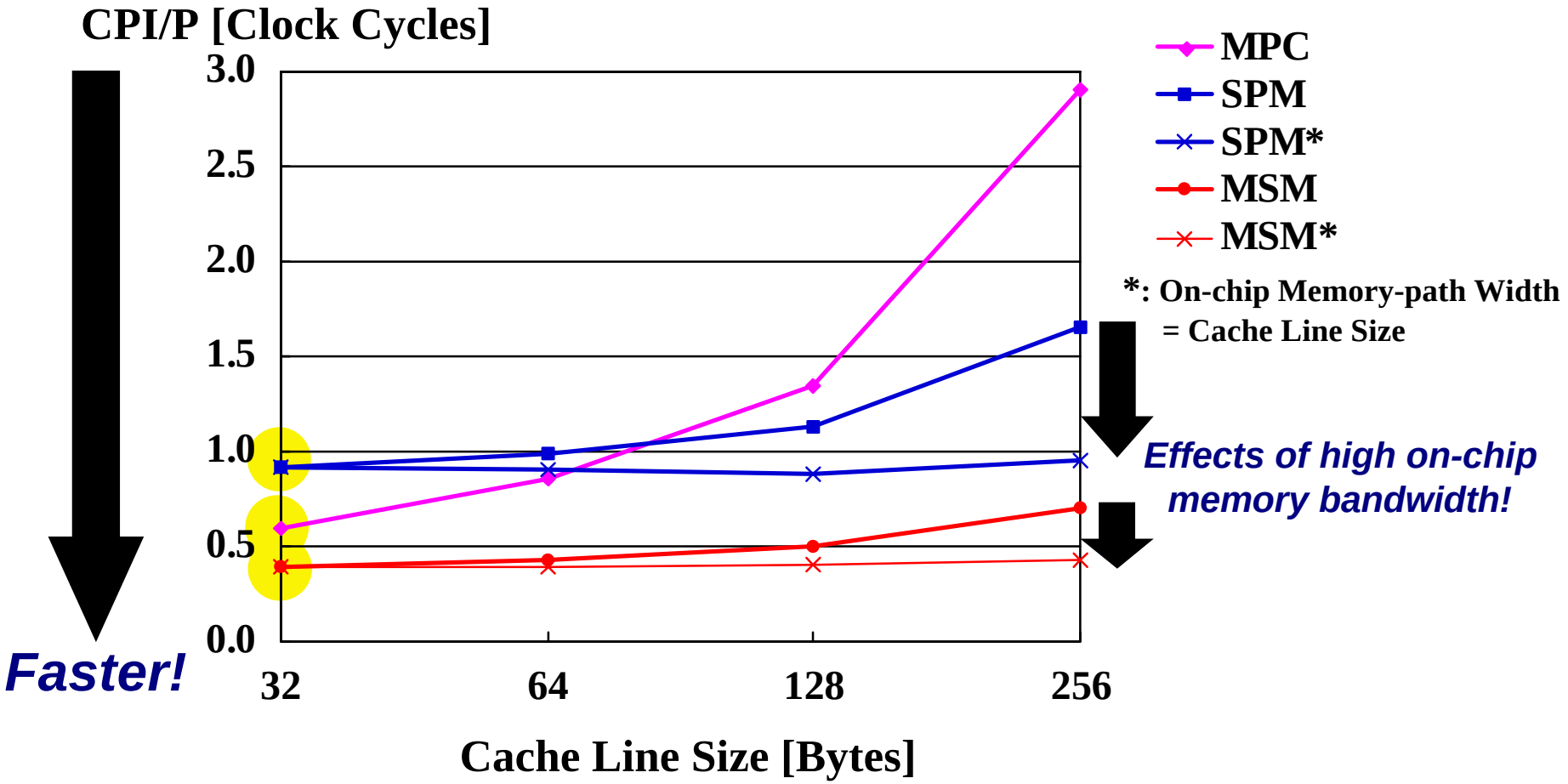
Saulsbury, A. et al., Missing the Memory Wall: The Case for Processor/Memory Integration, *Proc. ISCA'96*, pp.90-101, May 1996.

How to Turn High On-Chip Memory Bandwidth into Performance - Cache Refill -

- Kyushu ***PPRAM^R***

Murakami, K. et al., Parallel Processing RAM Chip with 256Mb DRAM and Quad Processors, *ISSCC'97 Digest of Technical Papers*, pp.228-229, Feb. 1997.

Performance Comparison Results
- Impact of Cache Line Size -



How to Turn High On-Chip Memory Bandwidth into Performance - Load/Store Ops -

- UC-Berkeley Vector IRAM

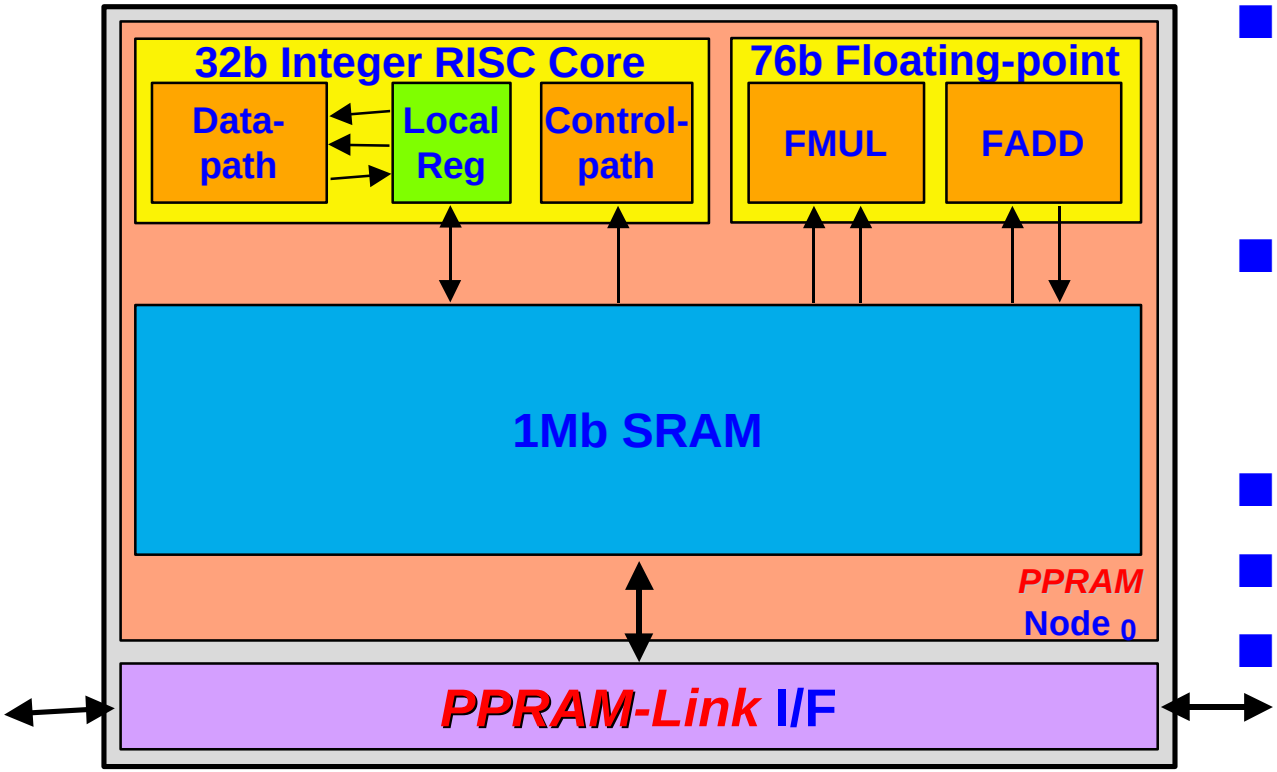
Patterson, D. et al., A Case for Intelligent DRAM: IRAM, *Hot Chips VIII*, Aug. 1996.

Performance Evaluation
- UC-Berkeley Vector IRAM -

Patterson, D. et al., Intelligent RAM (IRAM) Chips that Remember and Compute, *ISSCC'97 Digest of Technical Papers*, pp.224-225, Feb. 1997.

How to Turn High On-Chip Memory Bandwidth into Performance - ALU Ops -

- ***PPRAM$_{MOE}$***



- 0.35µm CMOS logic process with 3+ metal layers
- 32b Integer RISC Core + 76b Floating-point Mul&Add
- 1Mb SRAM
- Die size: *225mm^2*
- 200MFlop/s@100MHz

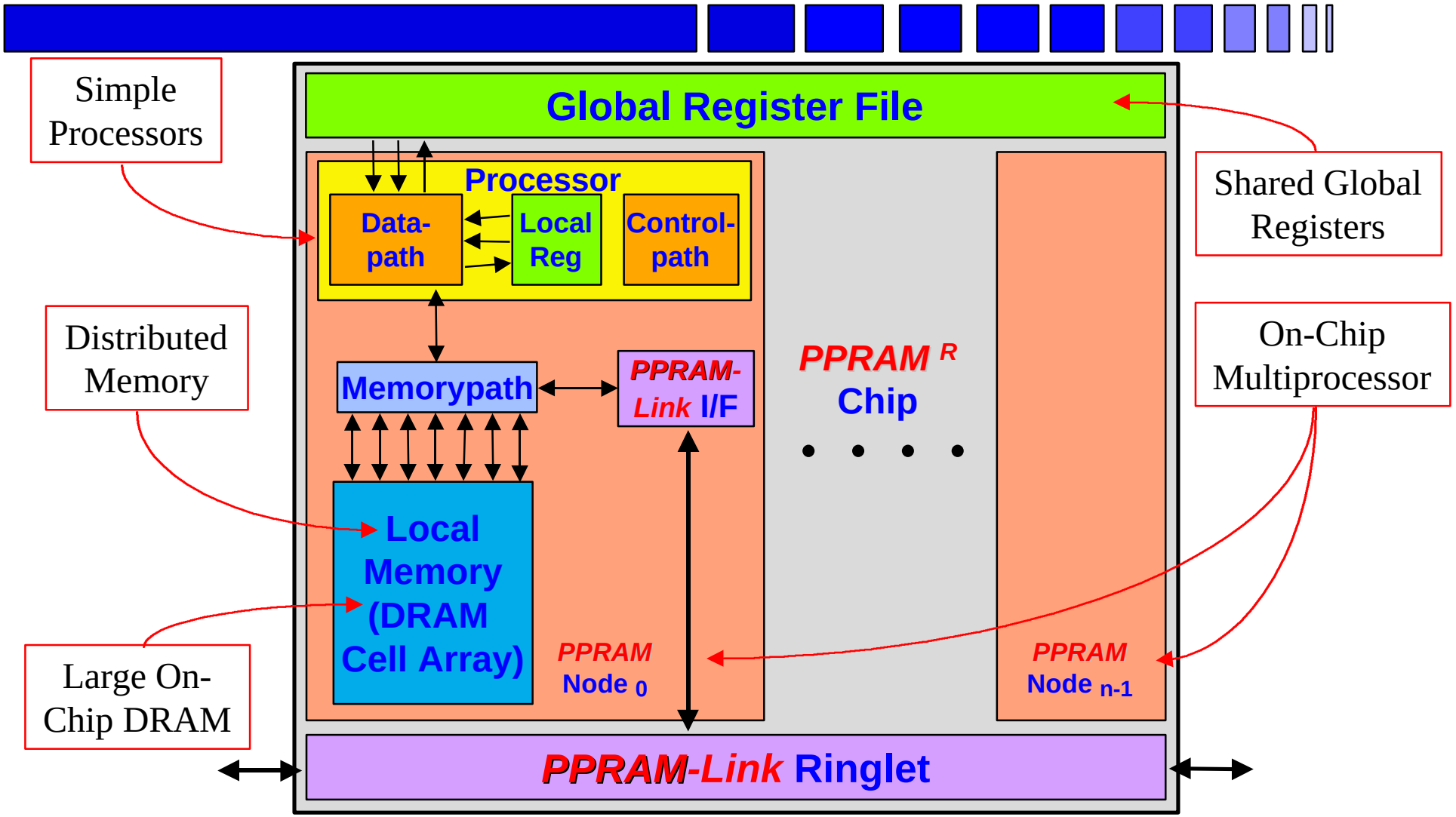
Outline

- What is ***PPRAM*** ?
- Why ***PPRAM*** ?
- Status of ***PPRAM*** Project at Kyushu
- ***PPRAM*** Consortium

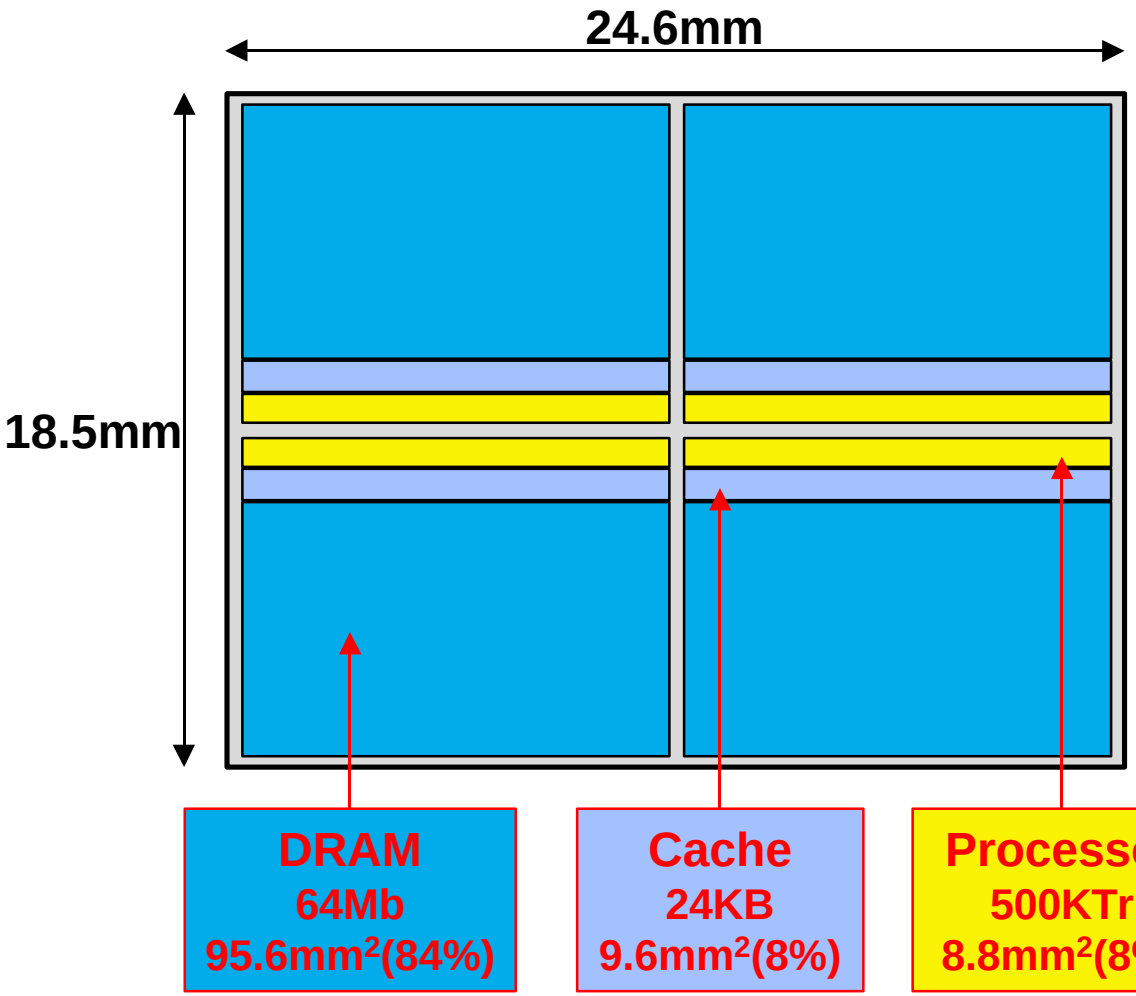
Status of ***PPRAM*** Project at Kyushu

- ***PPRAM^R 256-4***
  - Based on ***reference PPRAM*** (***PPRAM^R***) architecture and ***PPRAM-Link*** standard
  - Fabricate prototype chip with 256Mb DRAM and 4 media processors using 0.25µm-CMOS hybrid-DRAM/logic process by March 1999
- ***PPRAM$_{moe}$***
  - Processing node for *MOE* (*Molecular Orbital calculation Engine*), each with 1Mb SRAM, one 32b RISC integer processor, one 64b floating-point multiply&add unit, and ***PPRAM-Link*** interface
  - Collaborate with Fuji Xerox, Taisho Pharmaceutical, other three universities
  - Fabricate 0.35µm ASIC chip by March 1998
  - Also develop *IEEE1394* - ***PPRAM-Link*** bridge chip

Reference PPRAM (PPRAM^R)

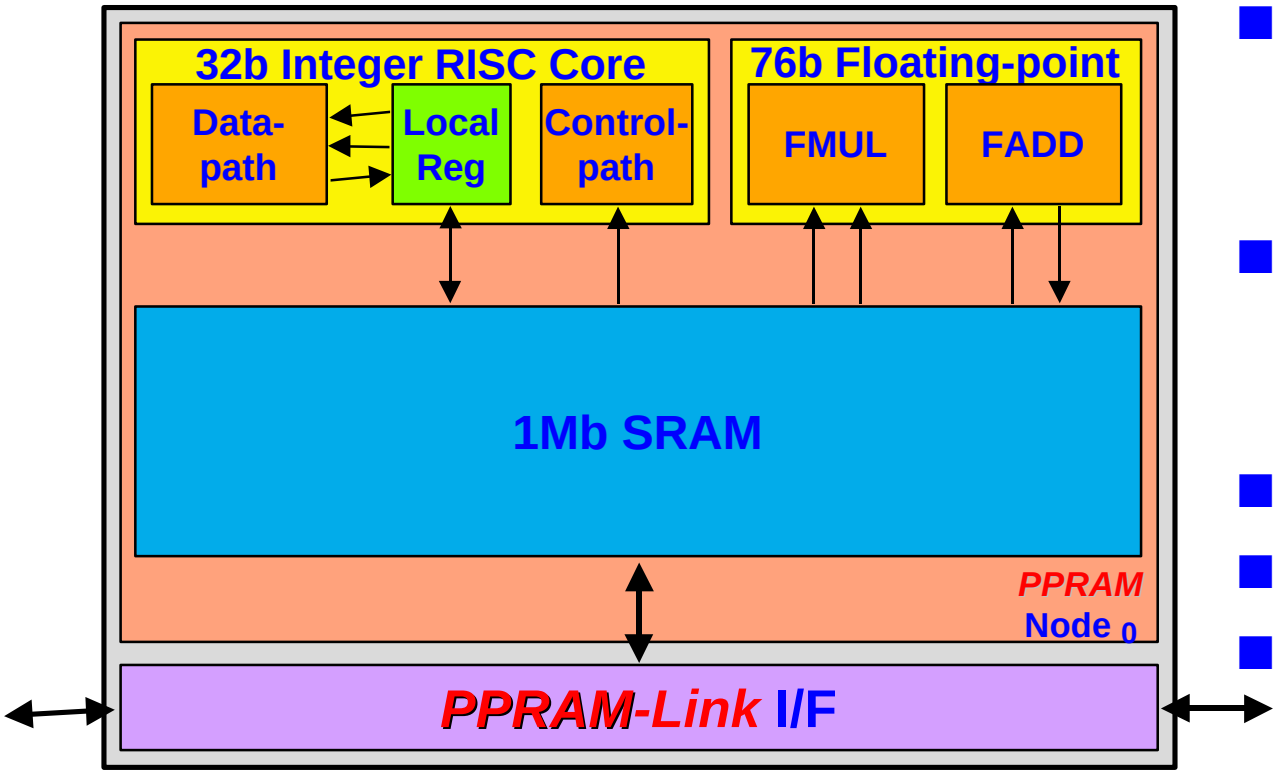


PPRAM^R 256-4 @ 0.25µm



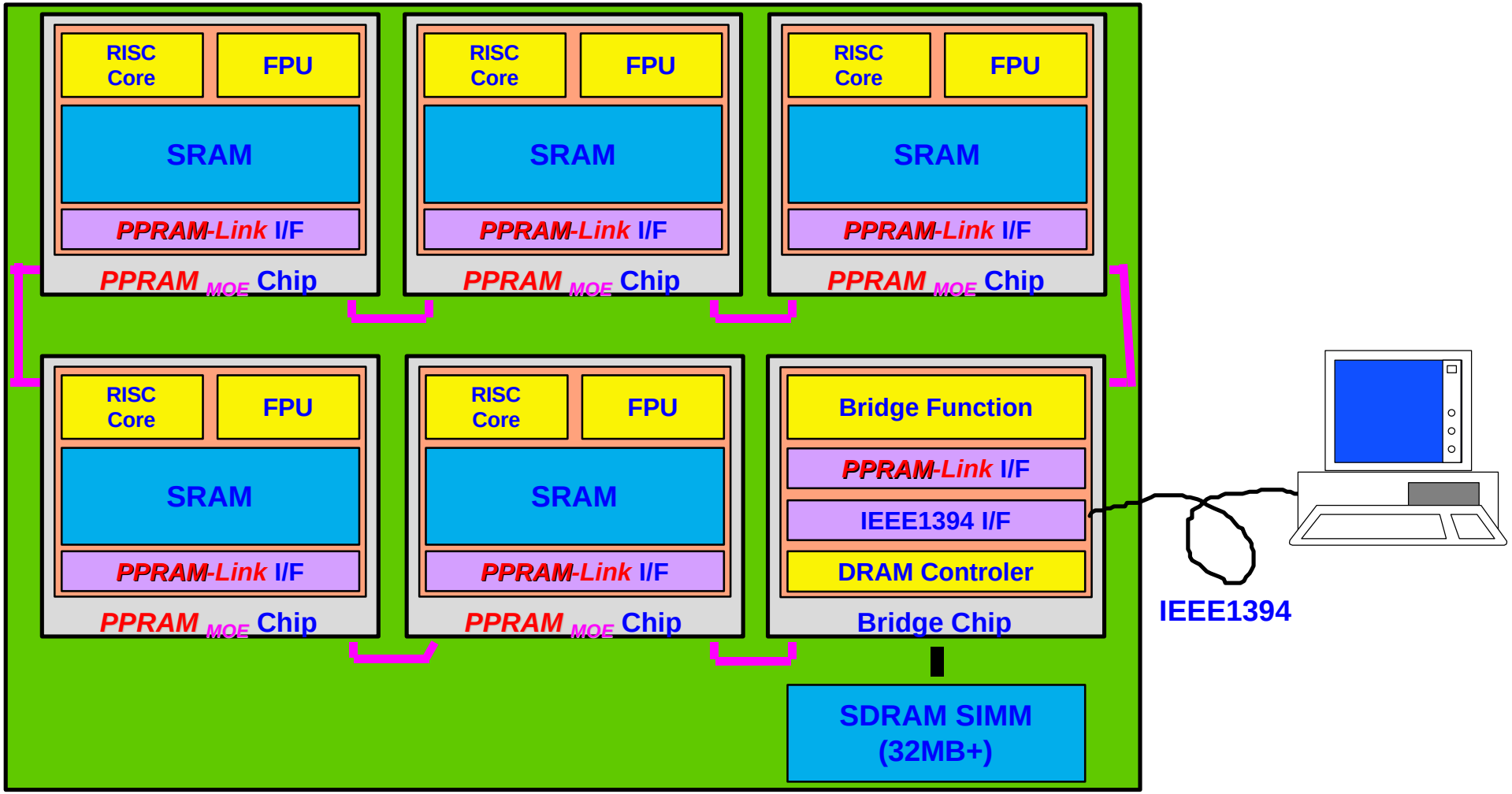
- 0.25µm CMOS, hybrid DRAM/logic process with 3+ metal layers
- 4PEs
- 256Mb DRAM
- PE area: *114.0mm²*
- Die size: *456mm²*

PPRAM MOE @ 0.35µm



- 0.35µm CMOS logic process with 3+ metal layers
- 32b Integer RISC Core + 76b Floating-point Mul&Add
- 1Mb SRAM
- Die size: *225mm²*
- 200MFlop/s@100MHz

1GFlop/s *MOE* Board

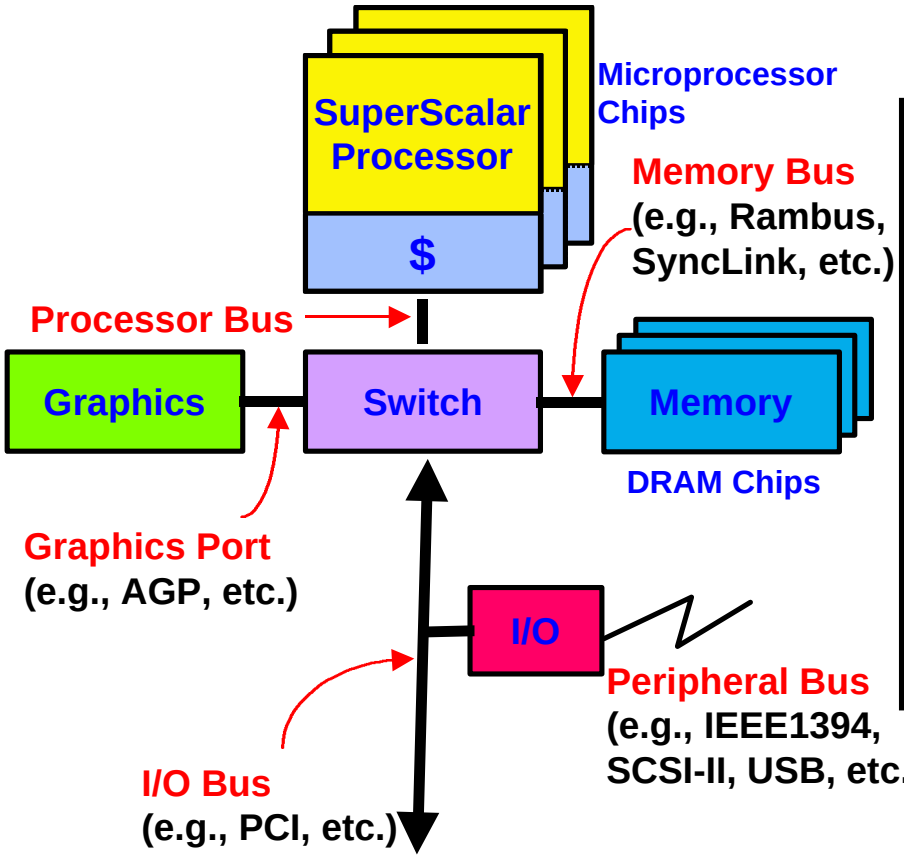


Outline

- What is ***PPRAM*** ?
- Why ***PPRAM*** ?
- Status of ***PPRAM*** Project at Kyushu
- ***PPRAM*** Consortium

PPRAM Paradigm Shift

■ Paradigm of today

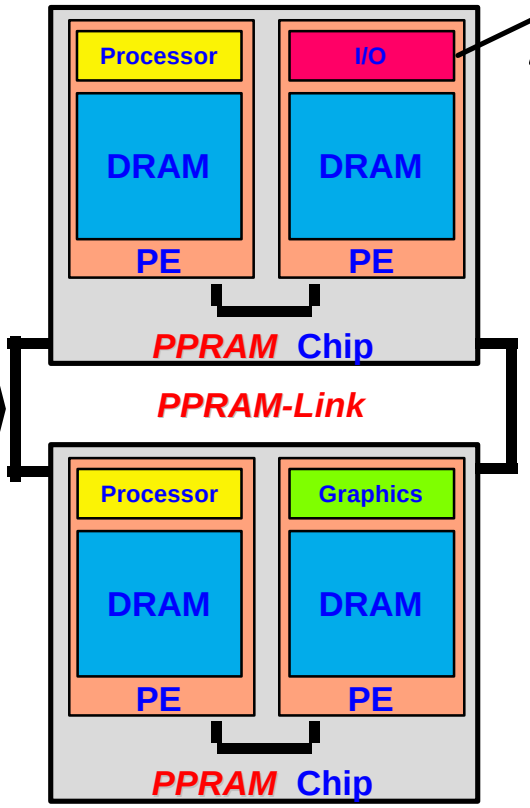


■ *PPRAM* Paradigm

Merged DRAM/ logic LSI

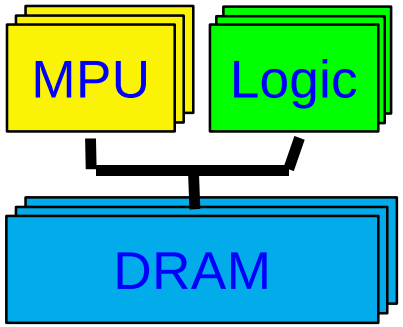
Parallel/distributed processing

Standard high-speed interconnect

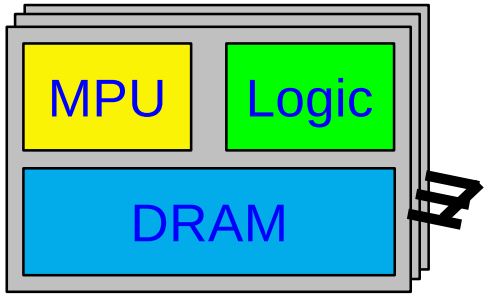


Issues on Realization of ***PPRAM*** Paradigm

■Paradigm of Today



■ ***PPRAM*** Paradigm



- What ***PPRAM*** nodes/chips shall be made ?
- How shall we design/fabricate/test ***PPRAM*** chips ?
- How shall we interconnect ***PPRAM*** chips and make system?
- What ***PPRAM***-based systems shall be made ?

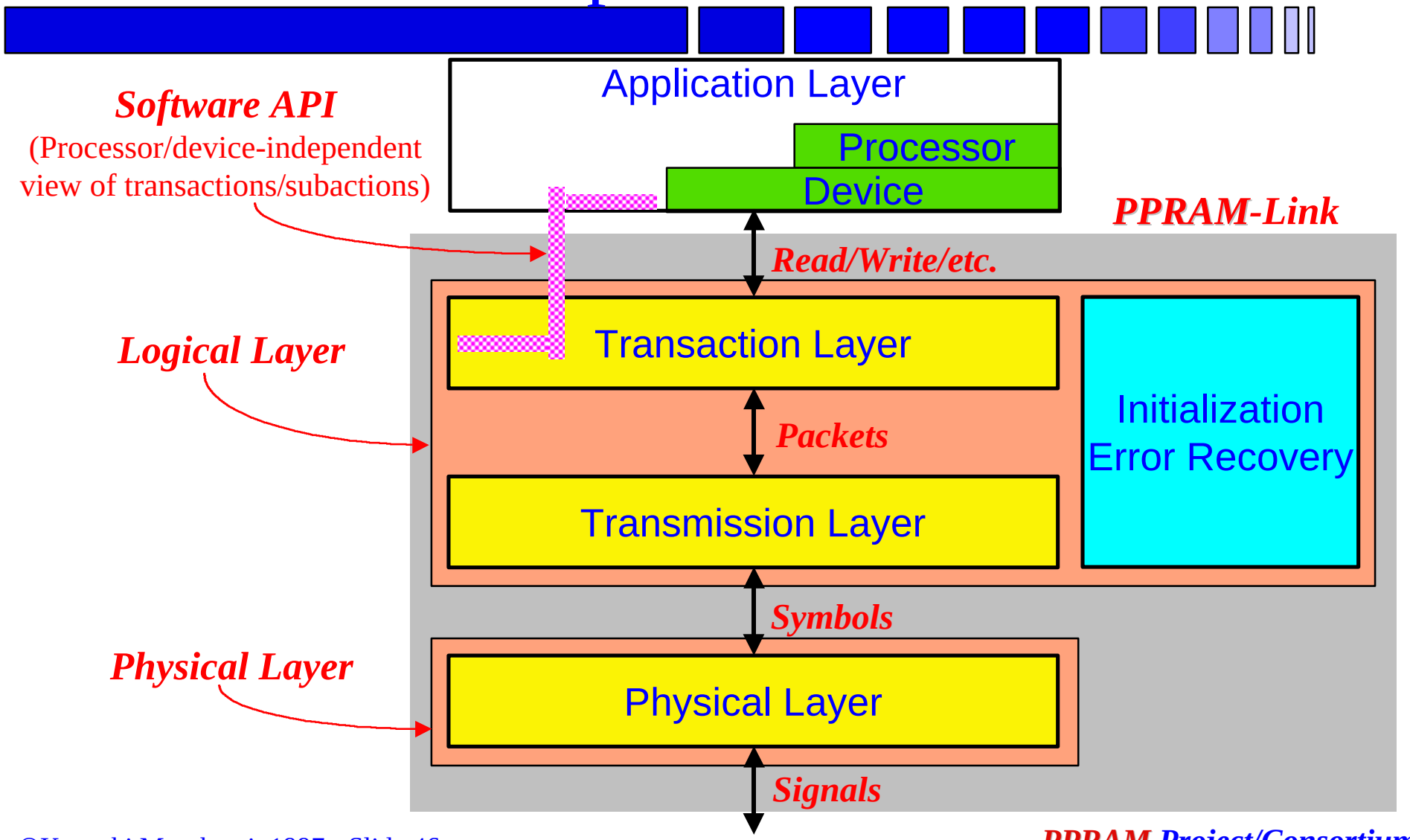
PPRAM Consortium
- Objectives and Activities -

- Objectives
  - Solve the following issues on realization of ***PPRAM*** paradigm:
    - Primary: How shall we interconnect ***PPRAM*** chips and make system ?
    - Secondary: How shall we design/fabricate/test ***PPRAM*** chips ?
  - Promote the proliferation of ***PPRAM*** paradigm
- Activities
  - Define a communication protocol and interface (***PPRAM-Link***) optimized for interconnecting ***PPRAM*** nodes/chips
  - Provide an open forum for discussing the secondary issues (design/fabrication/test) to be solved
  - Promote industry awareness and acceptance of ***PPRAM-Link*** standard
  - Submit, as appropriate, proposals to national/international standards bodies

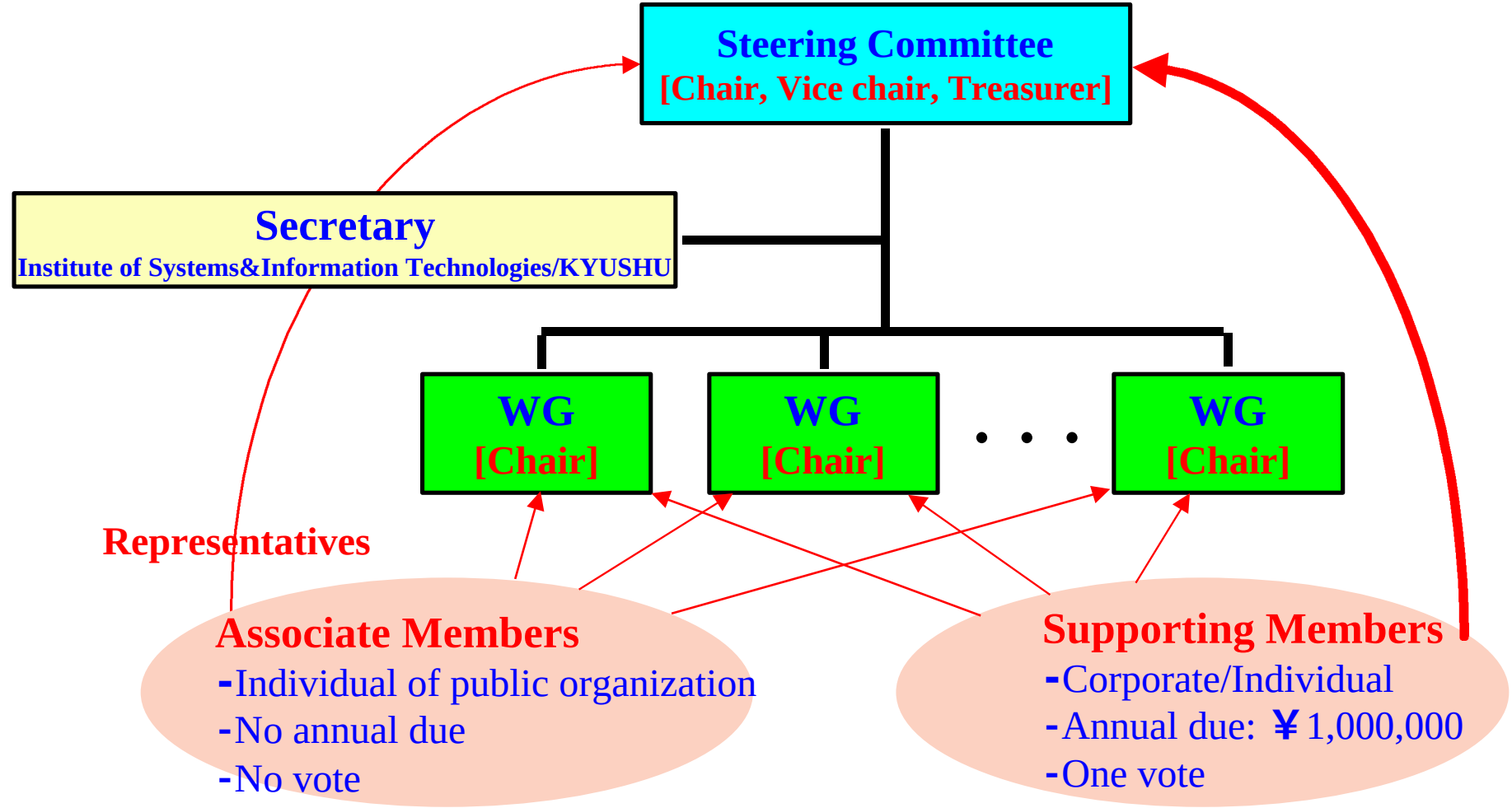
Goals of ***PPRAM-Link***

- High-performance and low-cost
  - Performance: ***>1GB/s per link*** @2000
  - Cost: ***<10% of die size*** @2000
- Communicate via *memory read/write* rather than I/O
  - NUMA (NonUniform Memory Access)
- Support *real-time transfer*
  - Correspond to *isochronous transfer* (e.g., IEEE1394)
- Scalable moderately
  - ***Not*** an interface for MPP (Massively Parallel Processor)
  - NCC (Non Cache Coherent) NUMA
- Provide *hot plug/plug&play*
  - Facilitate the usage of ***PPRAM*** cards

PPRAM-Link Protocol Stack
- Scope of Standardization -



PPRAM Consortium
- Organization -



PPRAM Consortium
- Statement of Openness -

- Supporting Members committed to open competition
- All Steering Committee meetings open to all Supporting Members and representatives of Associate Members
- Implementation or use of anything proposed by ***PPRAM*** Consortium is voluntary
- No discrimination: an individual or legal entity interested in promoting ***PPRAM*** may be a Supporting Member provided they accept the Articles

PPRAM Consortium
- Intellectual Property Rights -

- All material presented to ***PPRAM*** Consortium is considered nonconfidential
- Intellectual property generated through the activities of ***PPRAM*** Consortium will be licensed on open, reasonable and nondiscriminatory terms (but more favorable pricing to Members)

PPRAM Consortium

- Current Supporting Members and Officers -

- Supporting members (as of July 1997)
  - Fuji Xerox, Matsushita, Mitsubishi, NEC, Oki, Samsung, SONY, Taito, TI, Toshiba
  - *Pending*: Fujitsu, Hitachi, NTT, and so on
- Officers
  - Chair: Kazuaki Murakami, Kyushu Univ.
  - Vice chair: Eiji Masuda, Toshiba
  - Treasurer: Akira Matsuzawa, Matsushita

PPRAM Consortium
- Plan for 1997 -

- Found three Working Groups for ***PPRAM-Link***:
  - ***PPRAM-Link*** Physical-layer WG
  - ***PPRAM-Link*** Logical-layer WG
  - ***PPRAM-Link*** API WG
- Meet once every 4-6 weeks
- Complete ***PPRAM-Link*** ***Standard Draft 1.0*** by end of 1997 and distribute it in public

Backup Slides

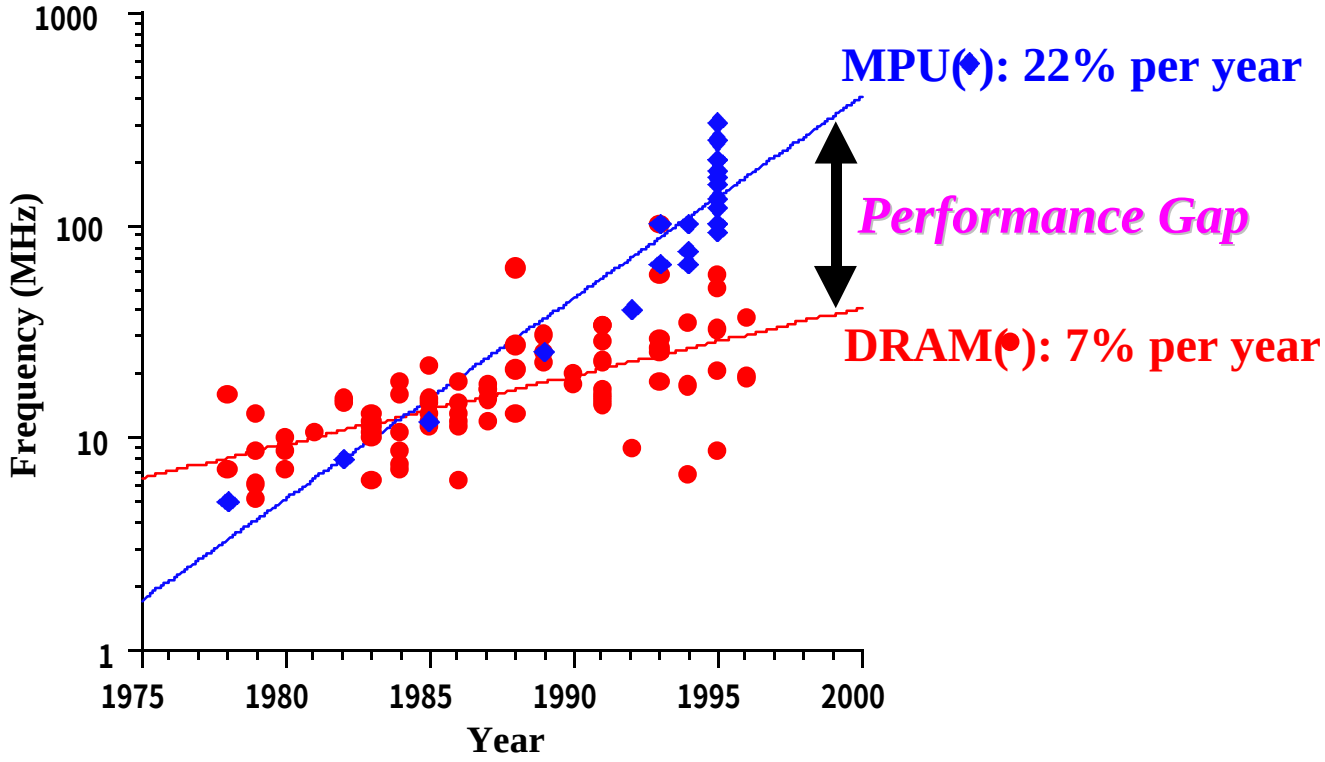
(The following slides are used to help answer questions)

Why Mixing DRAM and Logic on a Single Chip Now ?

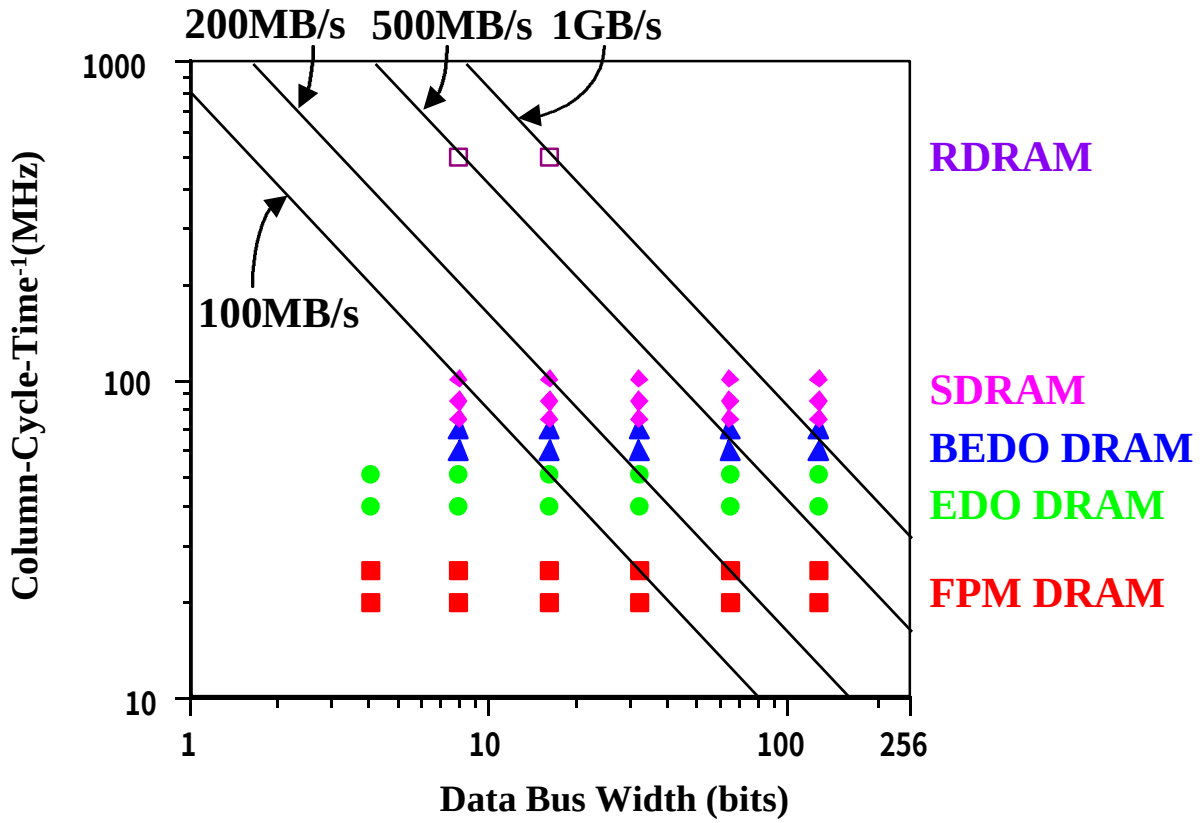
- Advance of semiconductor technology now makes “Fast Logic + Large DRAM” available at a reasonable cost.
- Gap between processor and DRAM speed, the “Memory Wall”, is now the biggest performance bottleneck.
- DRAM manufacturers are now facing challenges and therefore interested in merged DRAM/logic LSI.

Memory Wall

-Growing Processor-DRAM Performance Gap-



Memory Wall
-Limits of DRAM Bandwidth-



Source: Nikkei Electronics, no.641, July 31, 1995

Merged DRAM/Logic LSIs - Pros & Cons -

- Pros
  - High memory bandwidth
  - Low memory latency
  - Low energy consumption
  - Optimal memory size & organization
  - Low EMI
  - Low PCB cost, ...
- Cons
  - “Fast Logic + Large DRAM” ???
  - High processing cost
  - High testing cost
  - CAD ???
  - Killer applications ???

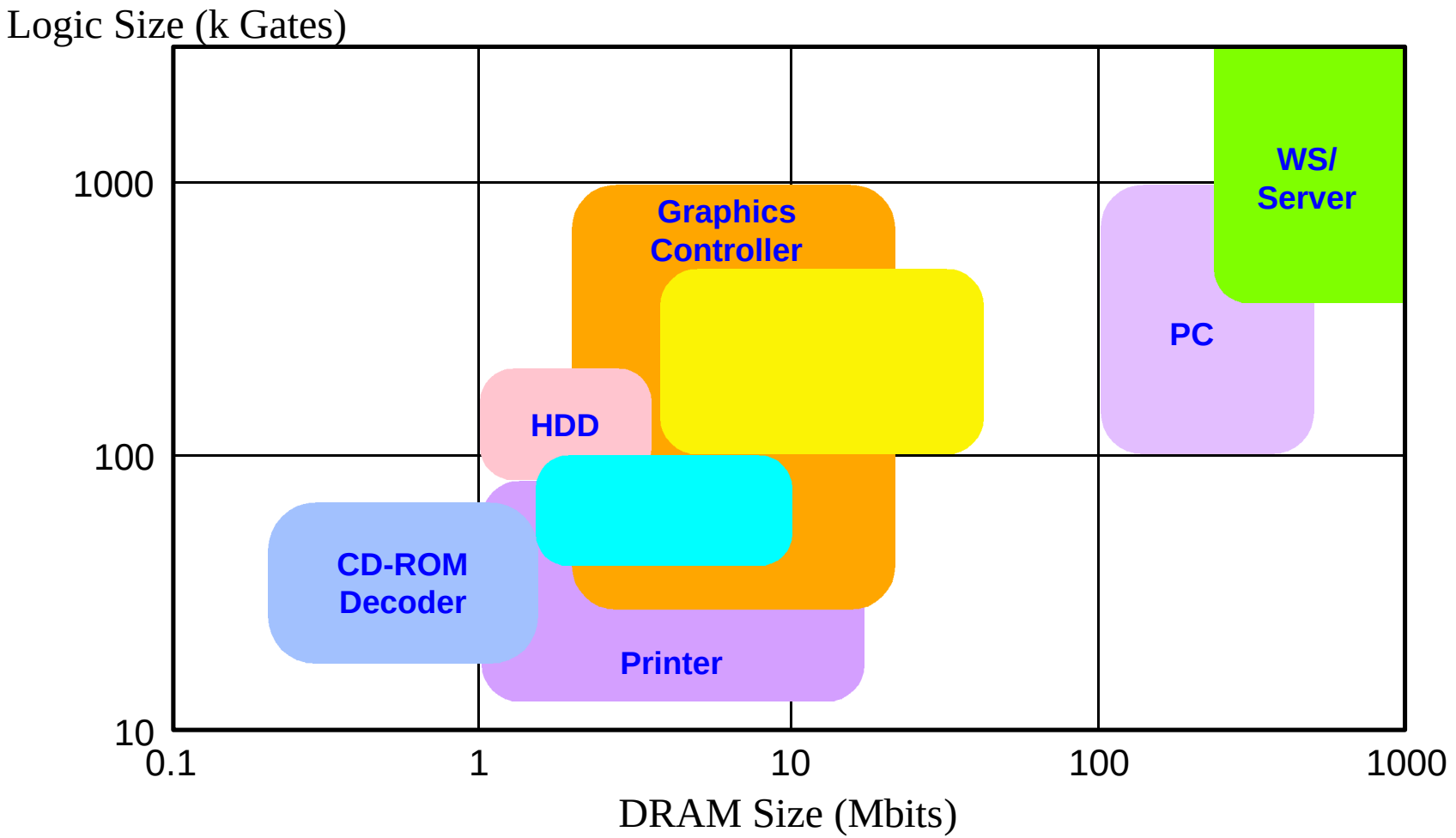
Low Energy Consumption
- Ex.) Mitsubishi M32R/D -

Source: Nikkei Microdevices, no.129, p.62, March 1996

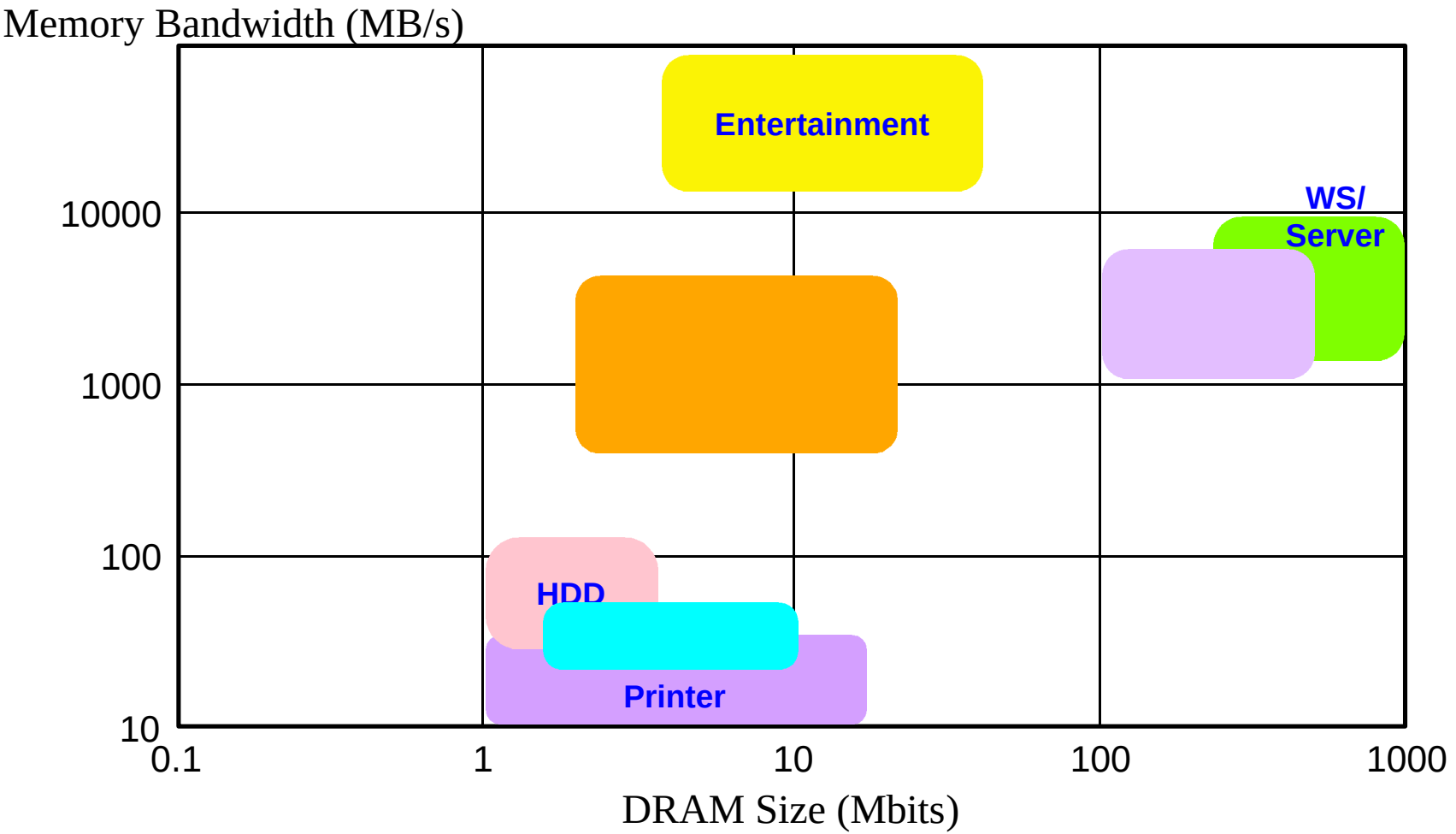
How Much is Processing Cost ?

Source: Nikkei Microdevices, no.135, p.77, Sept. 1996

Applications of Merged DRAM/Logic LSIs



Applications of Merged DRAM/Logic LSIs



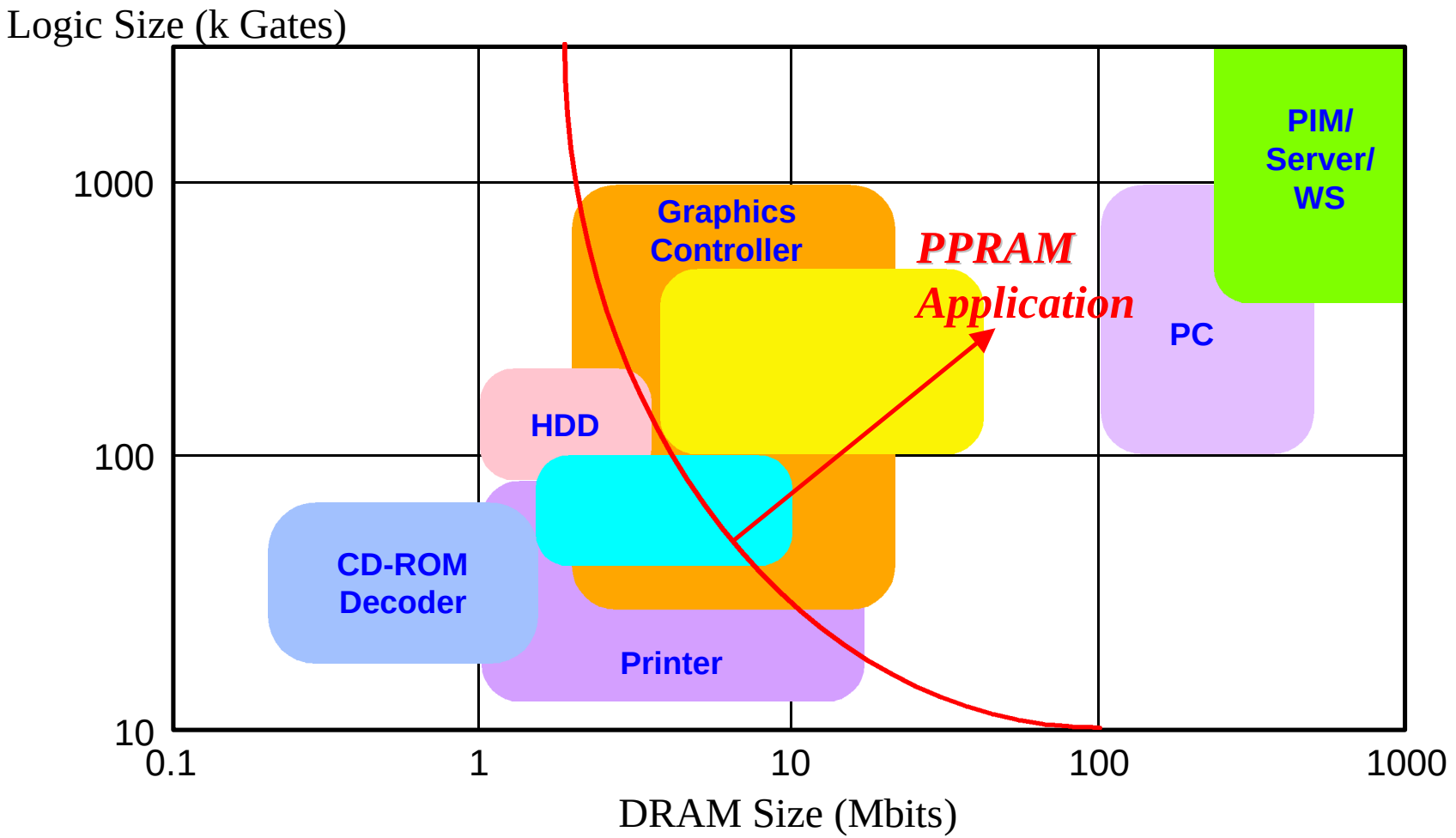
Merged DRAM/Logic LSI Architectures

- Application-specific logic-embedded DRAM
  - Mitsubishi 3D-RAM, ...
- Application-specific DRAM-embedded controller/accelerator/processor
  - Hitachi Media Chip (8Mb)
  - Oki Multimedia Accelerator MSM7680 (1.3MB)
  - NEC 2D Graphics Controller mPD76230 (2MB), ...

Merged DRAM/Logic LSI Architectures

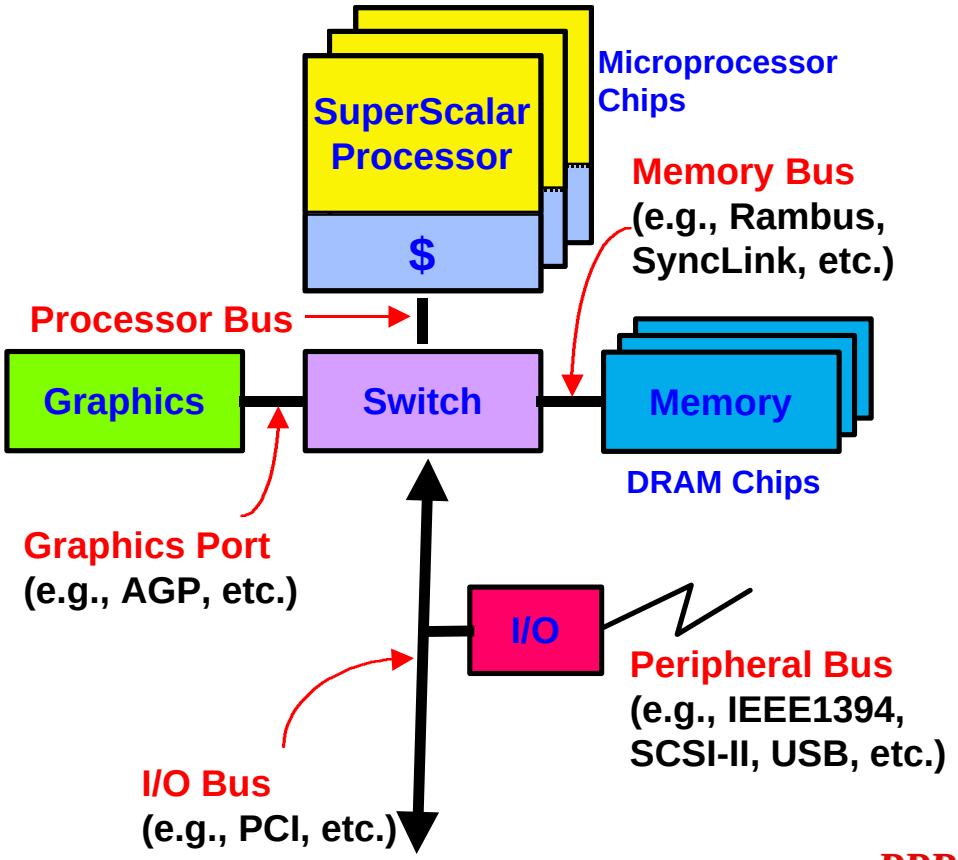
- DRAM-embedded microcontroller/microprocessor
  - Uniprocessor
    - » Mitsubishi M32R/D
    - » Sun Microsystems
    - » UC-Berkeley Vector IRAM
    - » UW-Madison DataScalar, ...
  - Multiprocessor
    - » IBM Execube
    - » Kyushu ***PPRAM^{R}***, ...

Applications of ***PPRAM***



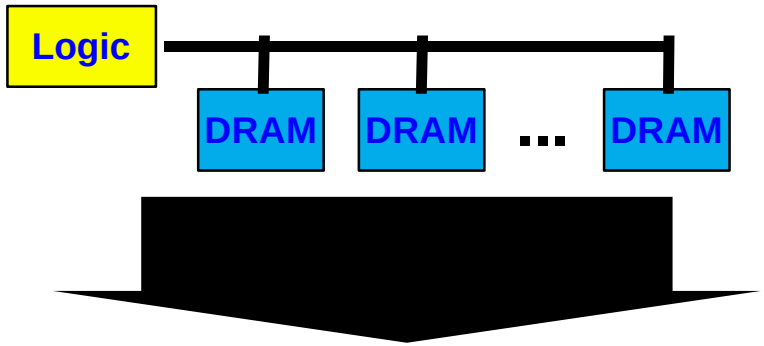
Inter-LSI Interconnection - Today -

- Traditional PC-style Inter-LSI Interconnection

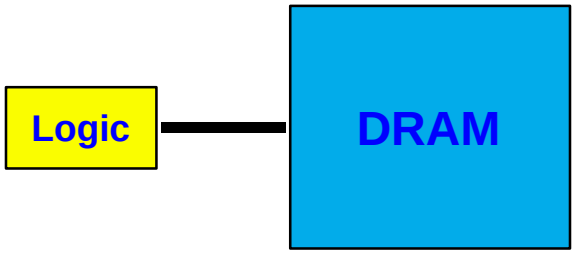


Inter-LSI Interconnection
- Future Possibilities -

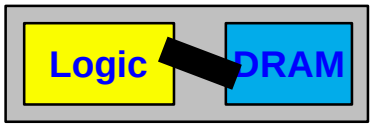
- DRAM-Logic Bus Interconnect



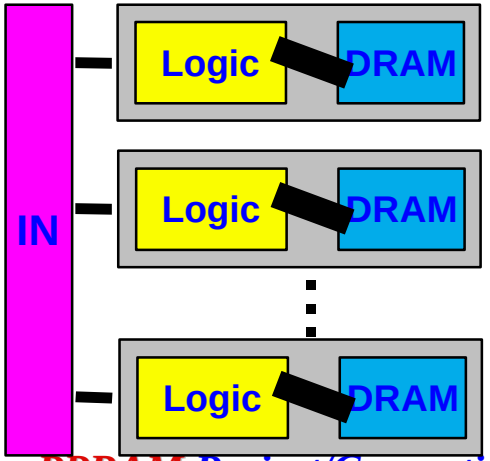
- DRAM-Logic Point-to-Point Interconnect



- Connectionless Merged DRAM/Logic LSI



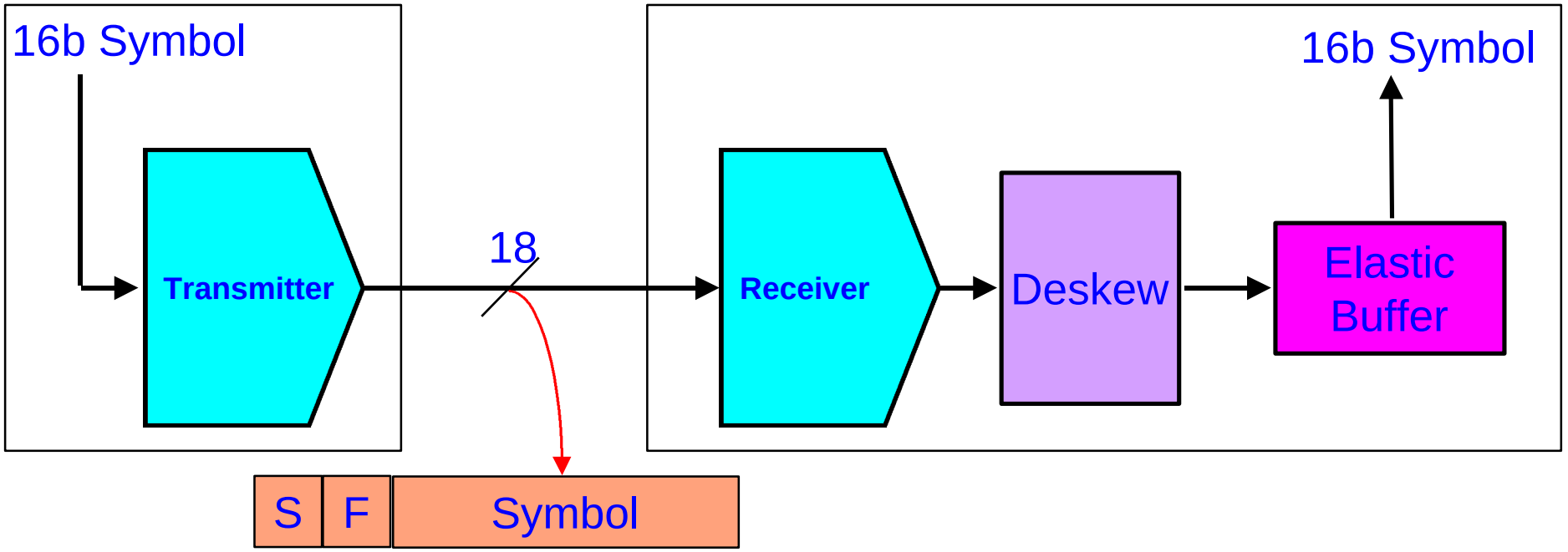
- Merged-DRAM/Logic LSI Interconnect



Physical Layer
- Issues and Alternatives -

- Bus vs. Point-to-Point
- Unidirectional vs. Bidirectional
- Parallel vs. Serial
- Clocking scheme
- Data encoding scheme
- Electrical vs. Optical
- Hot plug/Plug&Play

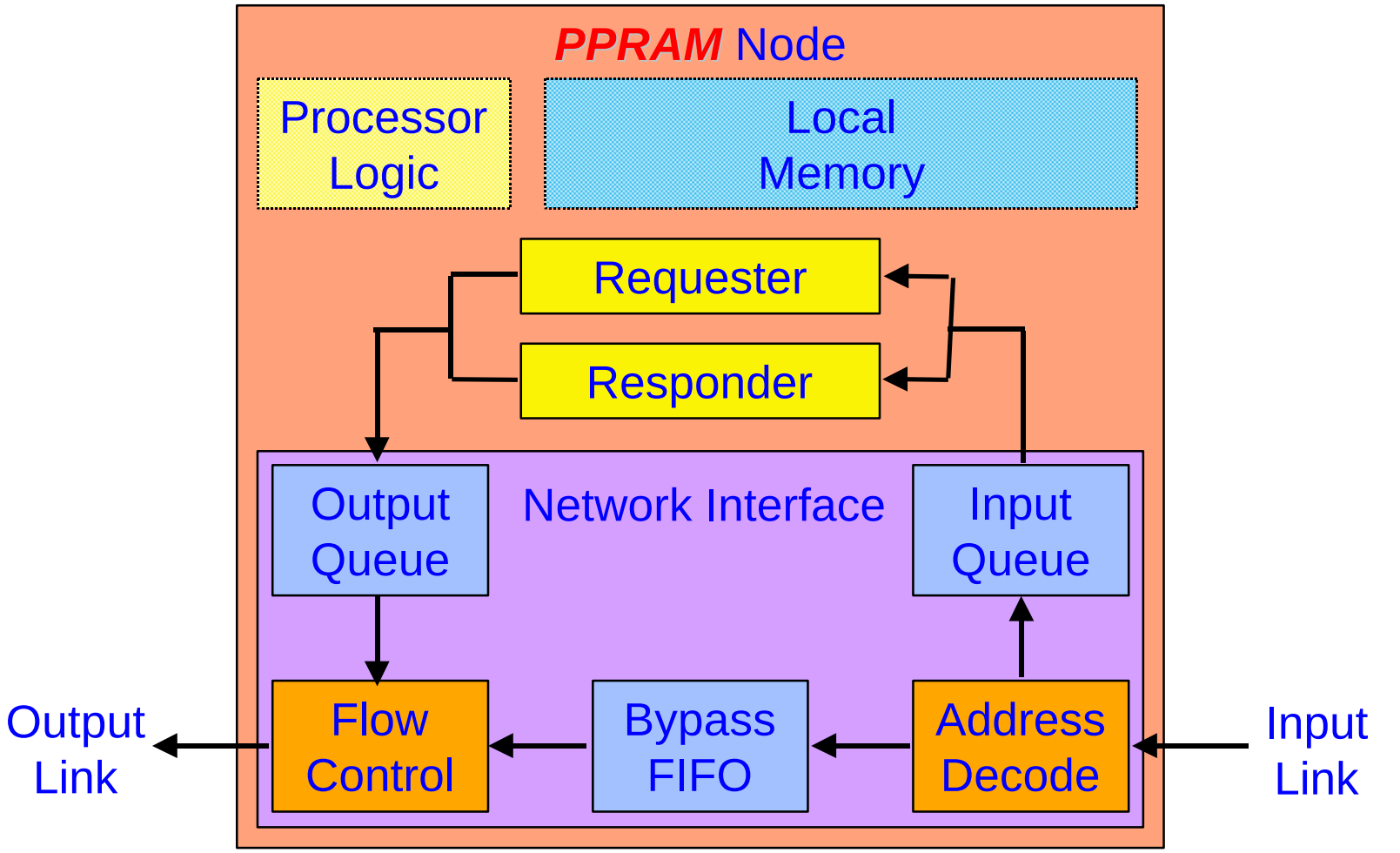
Physical Layer
- Case Study: SCI Type 18-DE -



Transmission Layer
- Requirements and Issues -

- Requirements
  - Deadlock-free
  - Starvation-free
    - » Fairness
    - » Guaranteed delivery
  - Real-time transport
    - » Unfairness
    - » Guaranteed timing
- Issues
  - Routing
  - Forwarding
    - » Store&Forward vs. Wormhole routing vs. Virtual cut-through
  - Flow control
    - » Bandwidth allocation
    - » Queue allocation

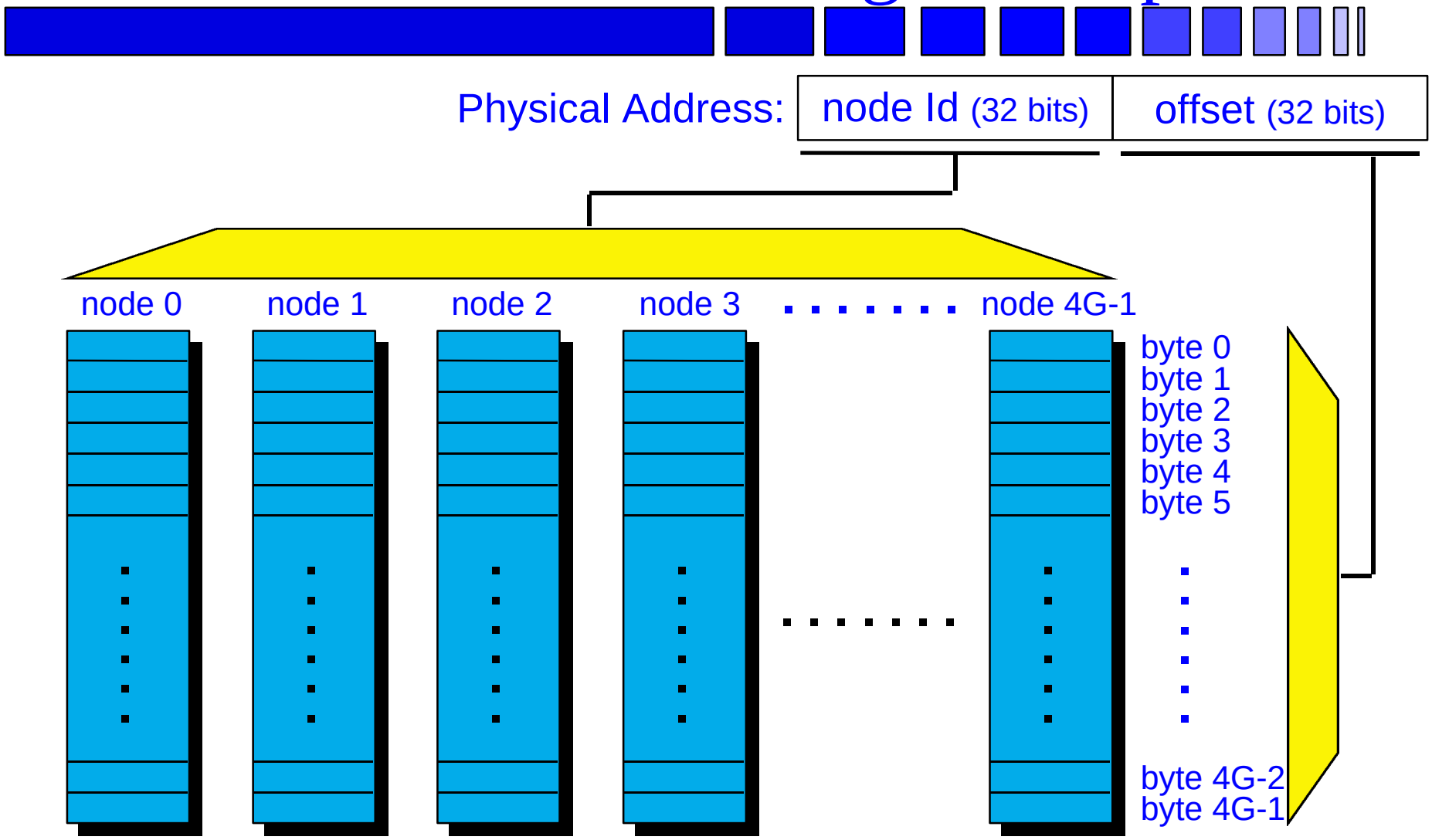
Transmission Layer
- *PPRAM* Node Model: A Proposal -



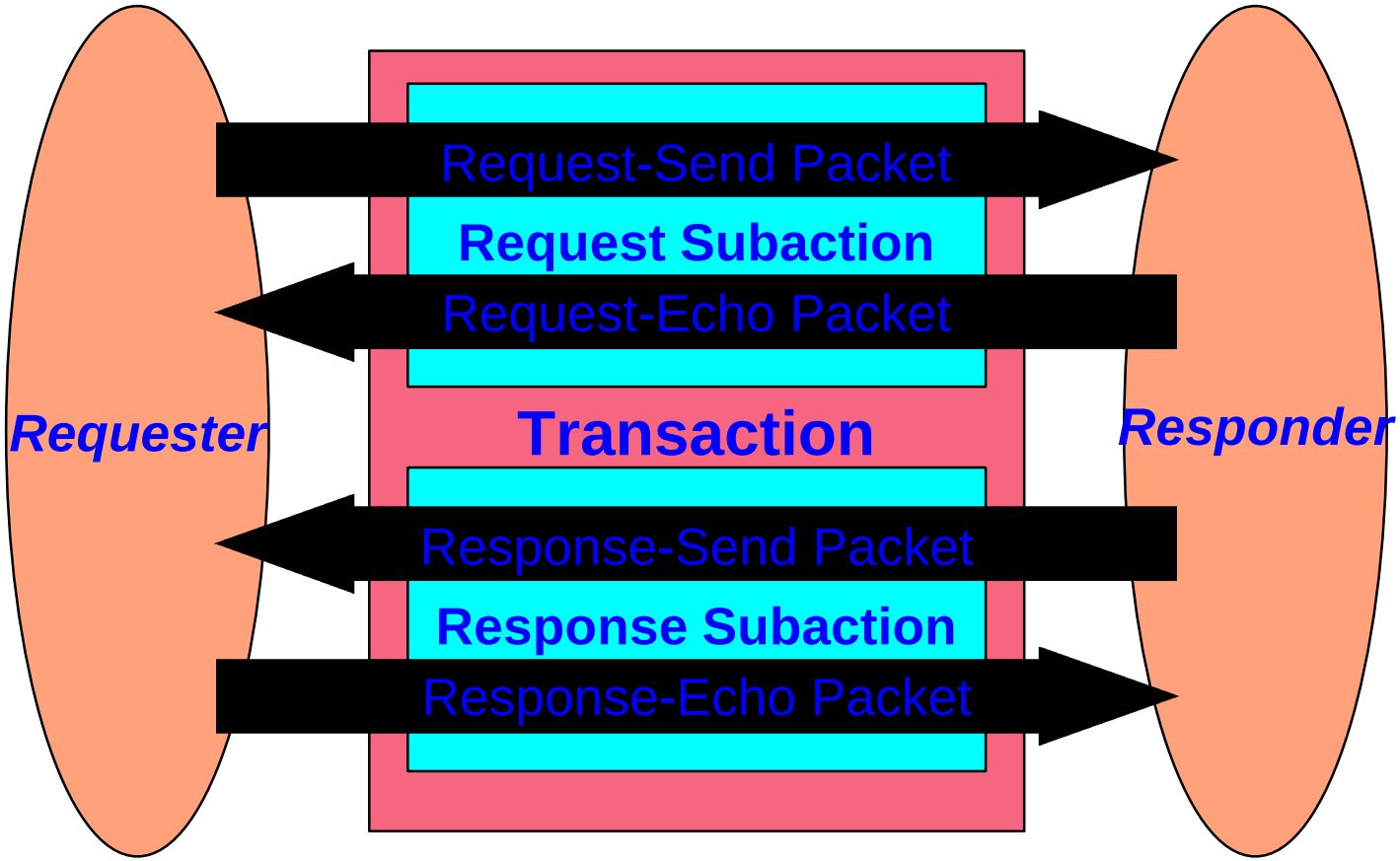
Transaction Layer
- Issues and Alternatives -

- Transaction architecture
  - Memory vs. I/O channel (message passing)
- Transaction protocol
  - Unified vs. Split transactions
- Cache coherence
  - Supported (CC-NUMA) vs. Not supported (NCC-NUMA)
- Special transport (e.g.; real-time, broadcast, etc.)
  - Supported vs. Not supported

Transaction Architecture
- Fixed 64-bit Addressing: A Proposal -

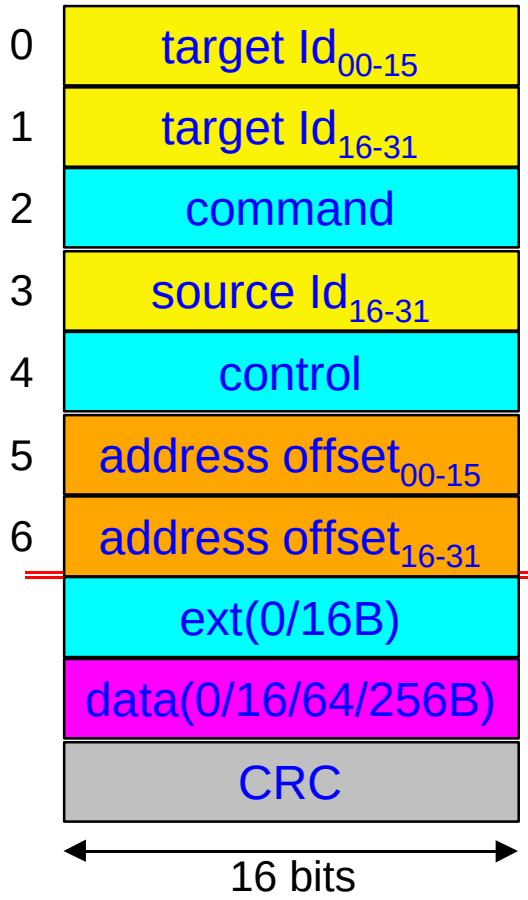


Transaction Protocol: A Proposal
- Transactions, Subactions, Packets -

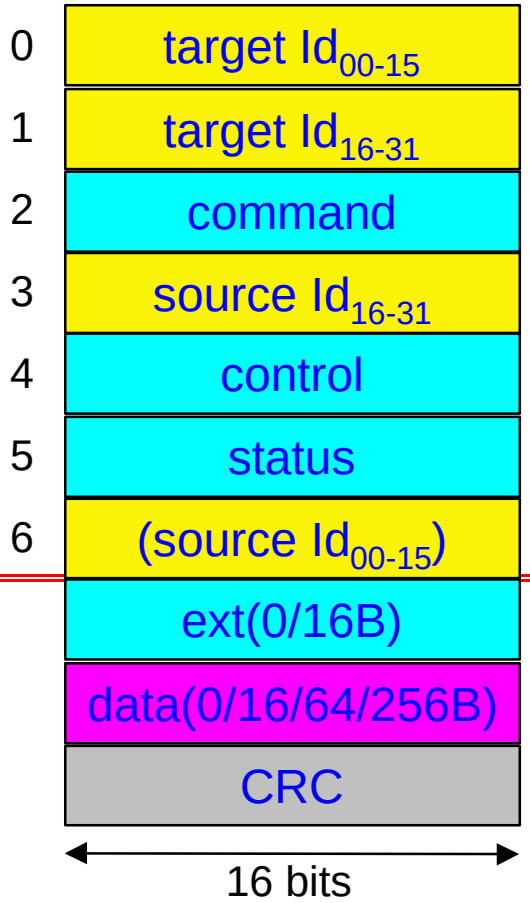


Transaction Protocol
- Packet Formats: A Proposal -

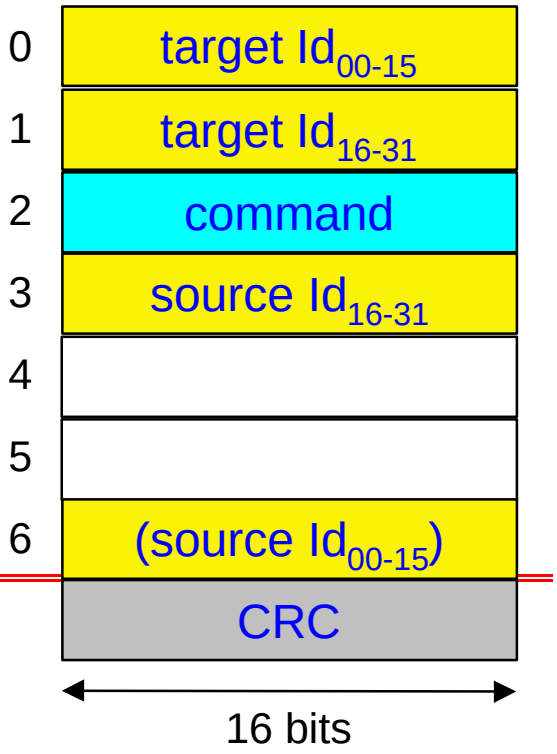
■Request-send



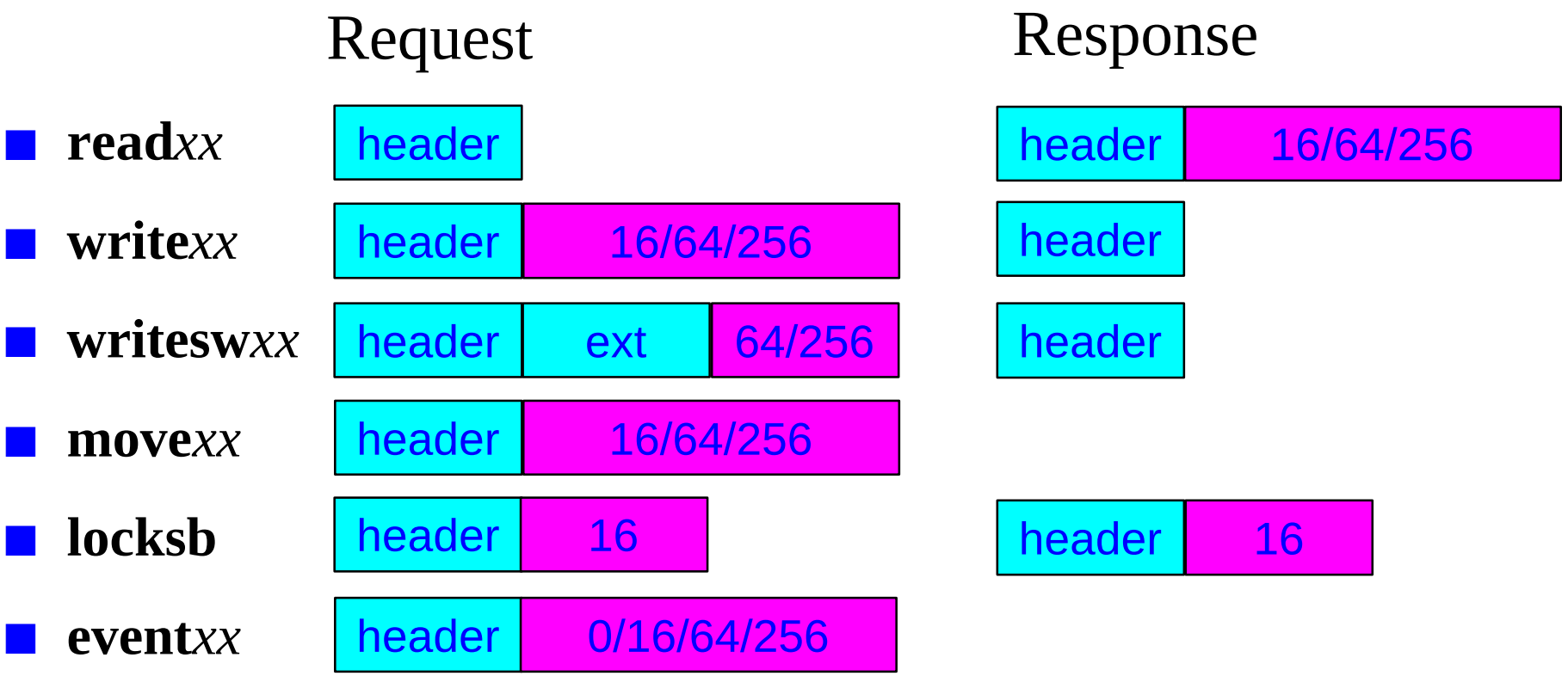
■Response-send



■Echo

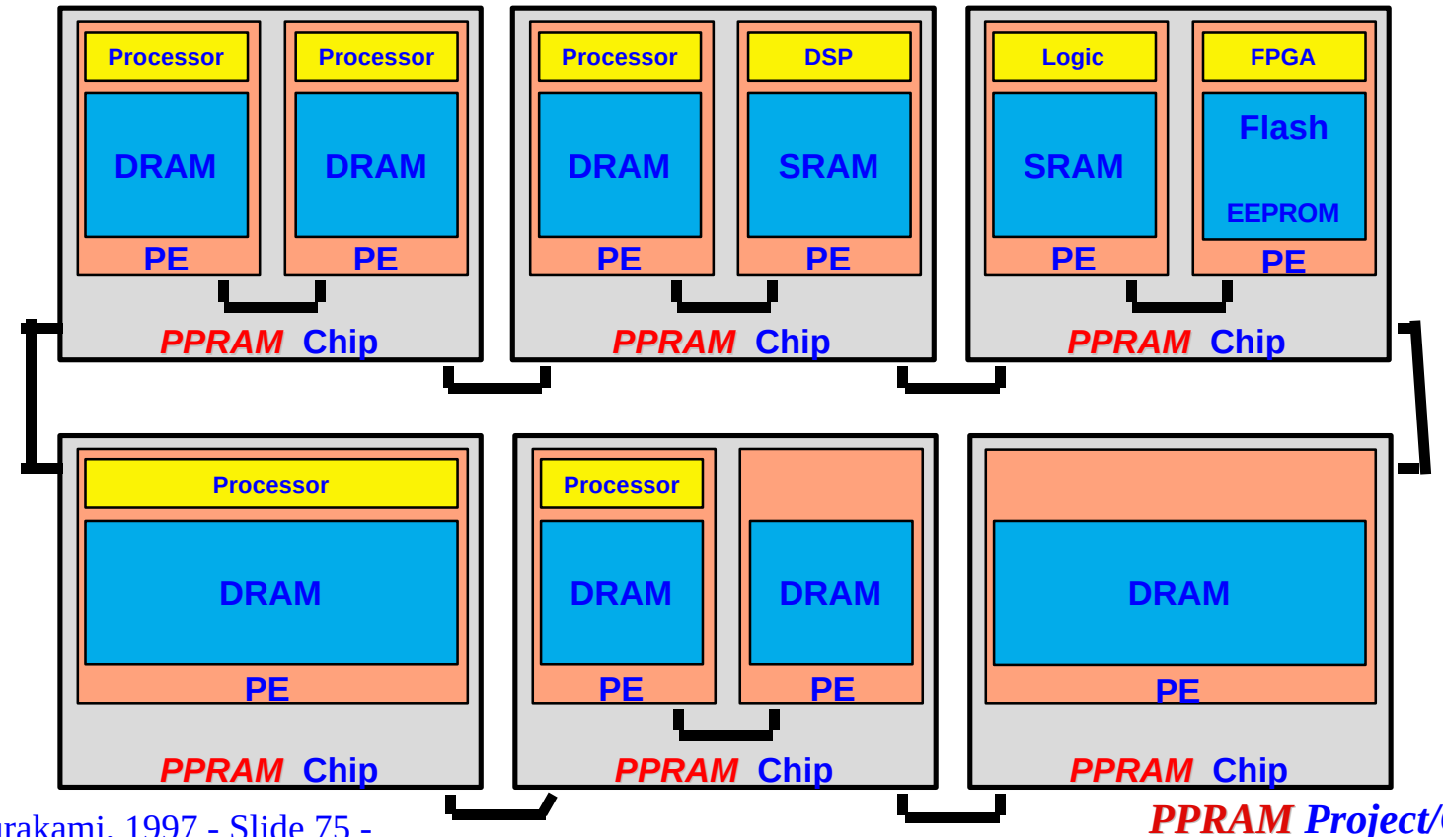


Transaction Protocol
- Transaction Types: A Proposal -



Topology-Independent Network

- Ring Interconnect
  - Cheapest, but slowest



Topology-Independent Network

- Switch Interconnect
  - Fastest, but expensive