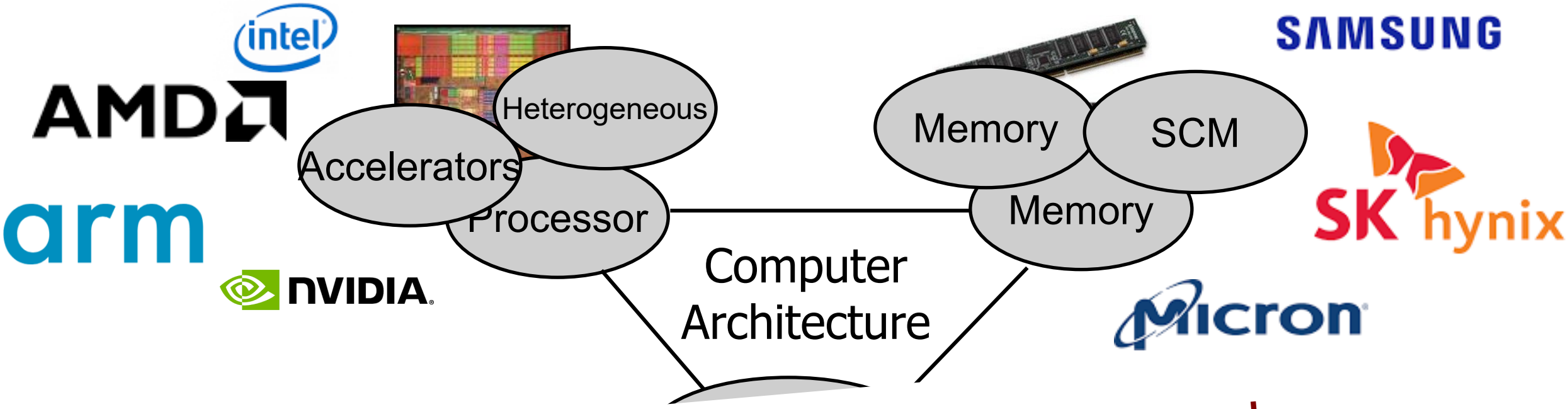
Communication-centric Future SSD using Interconnection Networks

John Kim (김동준)
School of Electrical Engineering
KAIST

Computer Systems & Architecture



Interconnect : Moving bits around

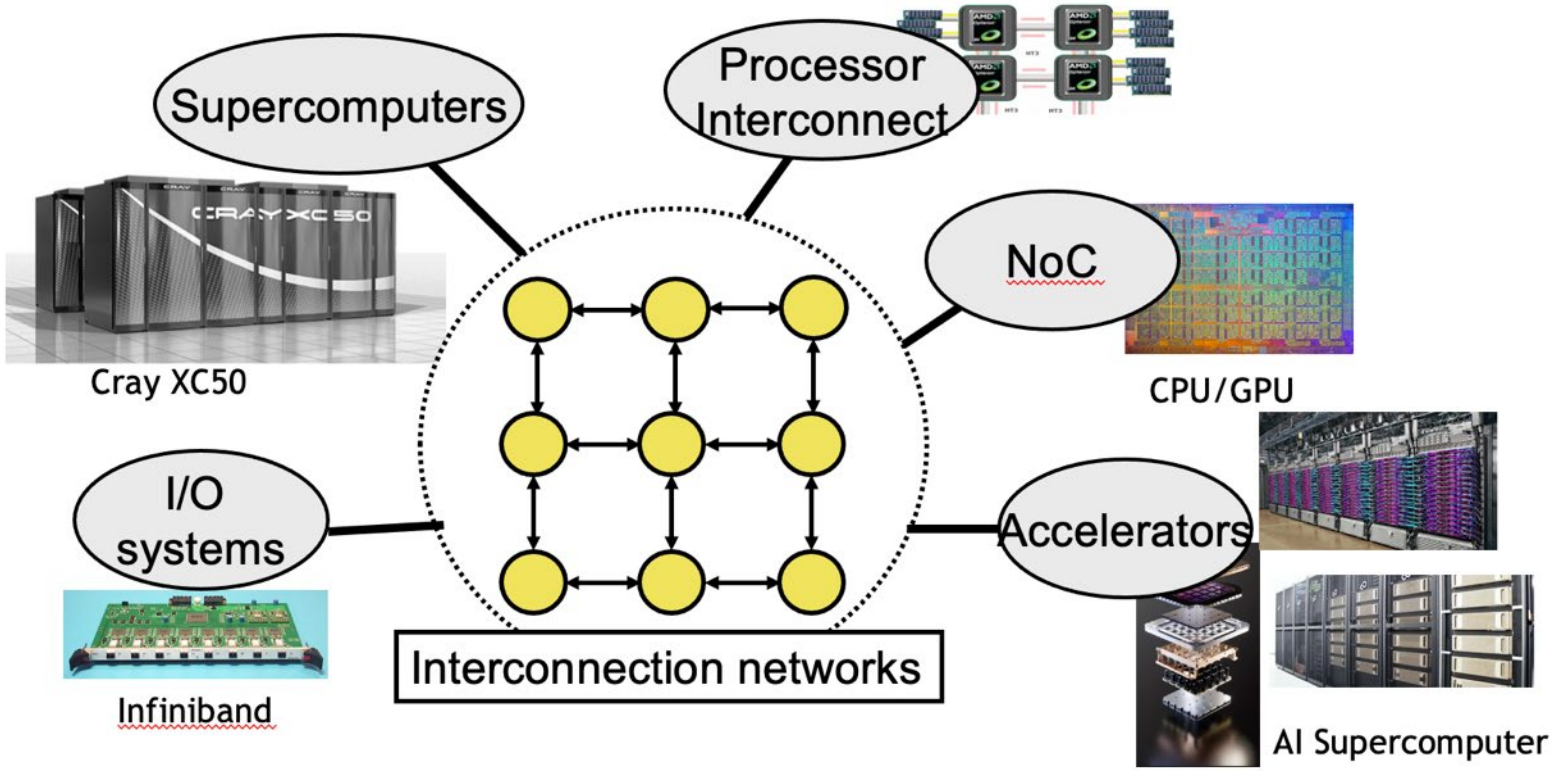
What is an Interconnection Network?

"*A programmable system that enables fast data communication between components of a digital system.*" [Dally & Towles]

- Sharing of expensive communication resources
- Provide a structured way to organize communication

Interconnection Networks

- Microarchitecture - router organization
- Topology - “roadmap” of the network
- Routing - which path a packet takes
- Flow Control - allocation of network resources



Send "*packets*" not signals (wires)

Route Packets, Not Wires: On-Chip Interconnection Networks

William J. Dally and Brian Towles
Computer Systems Laboratory
Stanford University
Stanford, CA 94305
{billd,btowles}@cva.stanford.edu

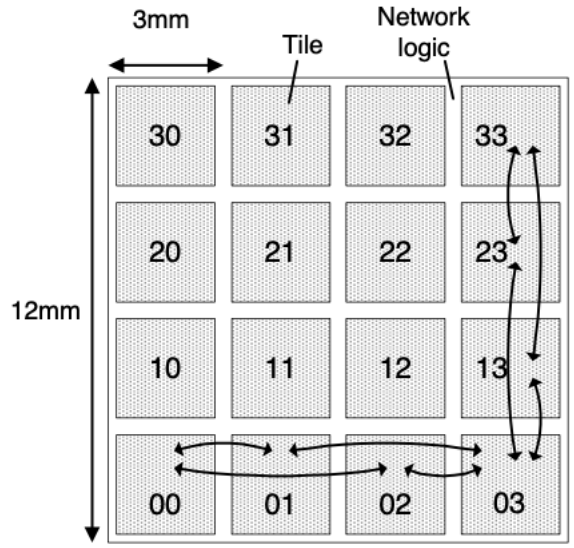


Figure 1 Partitioning the die into module tiles and network logic

Today's Talk

- Interconnection Networks Background

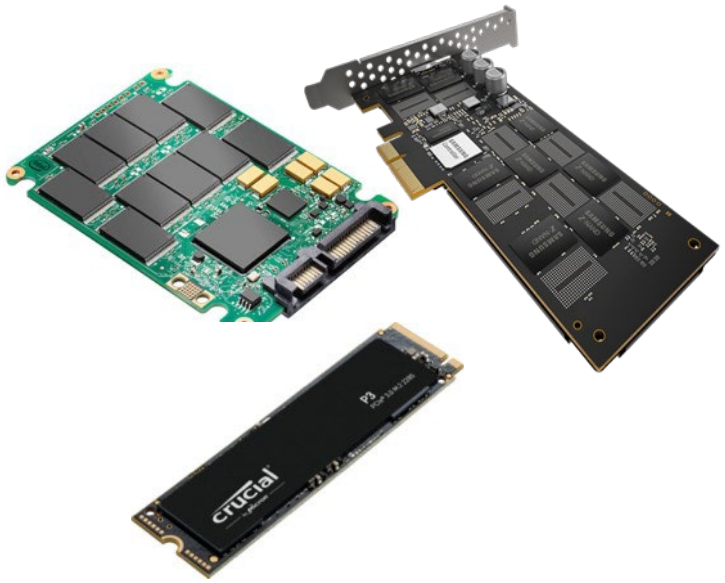
- NetworkSSD : Interconnect of Flash Memory [MICRO'22]
  ➔ Interconnection Networks between the Flash

- DecoupledSSD : Minimizing Data Movement internal to SSD [ISCA'23]
  ➔ Interconnection Networks between the Flash Controller

- Evaluations

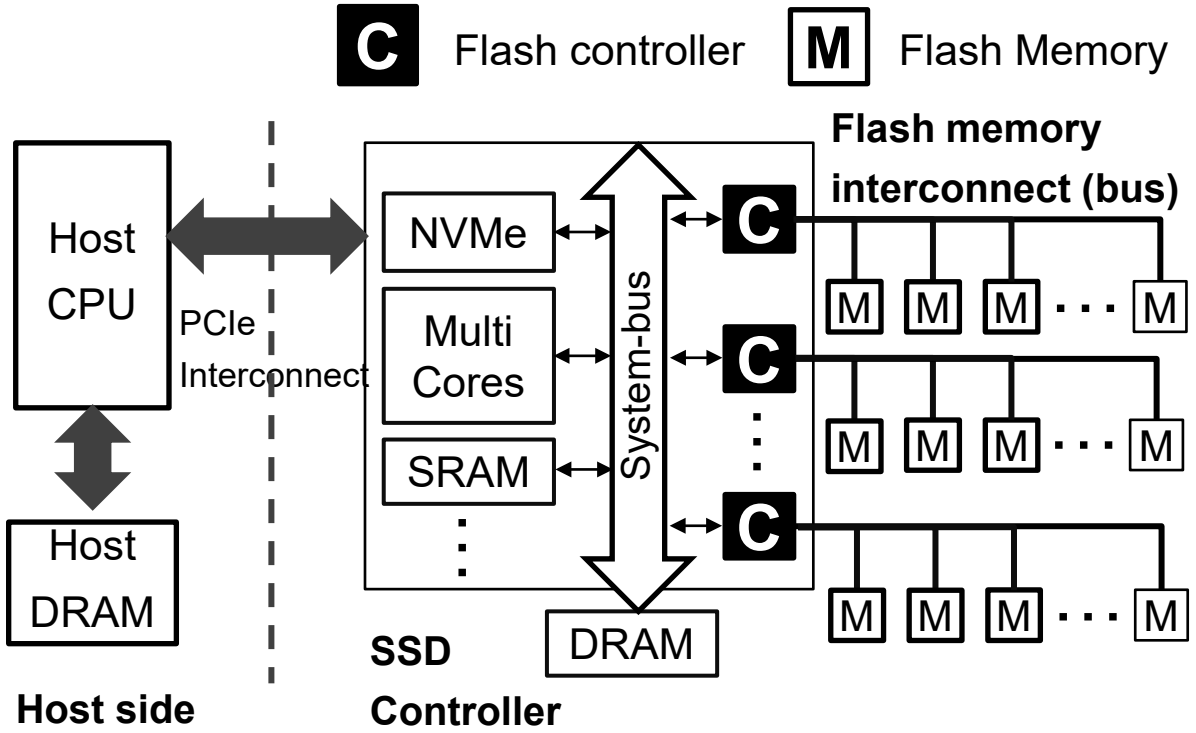
- Summary

SSD Architecture Overview



Solid State Drives

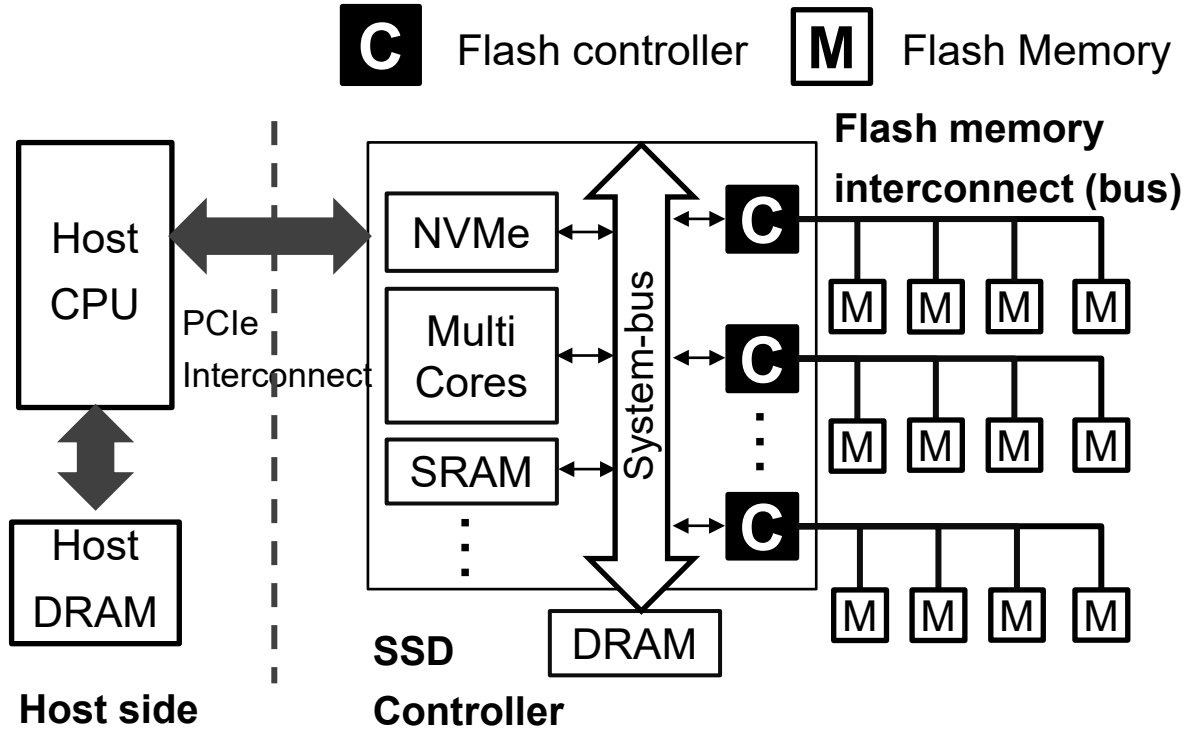
=



Networked SSD: Flash Memory Interconnection Network for High-Bandwidth SSD

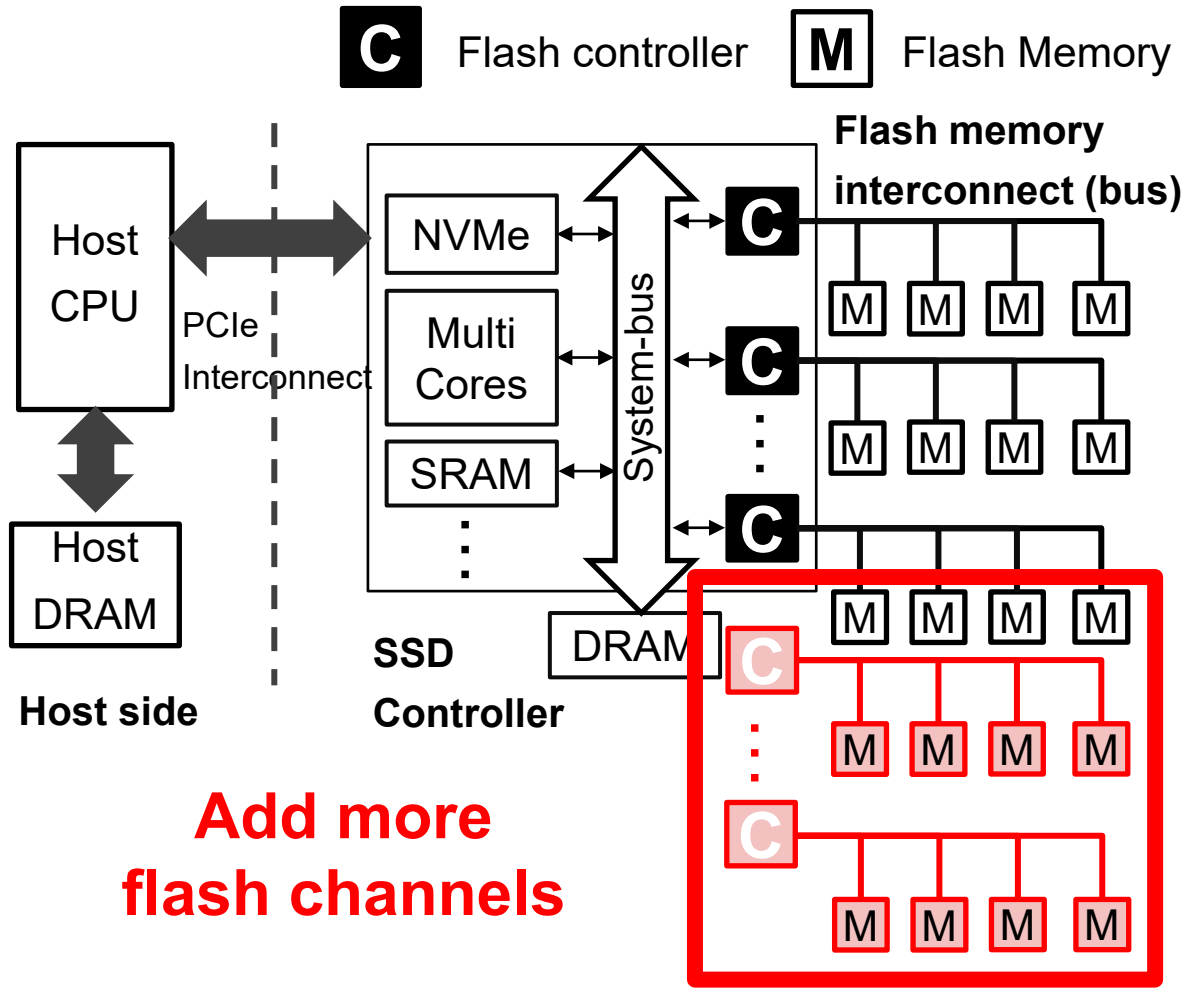
[MICRO'22]

Scalable, High Bandwidth SSD



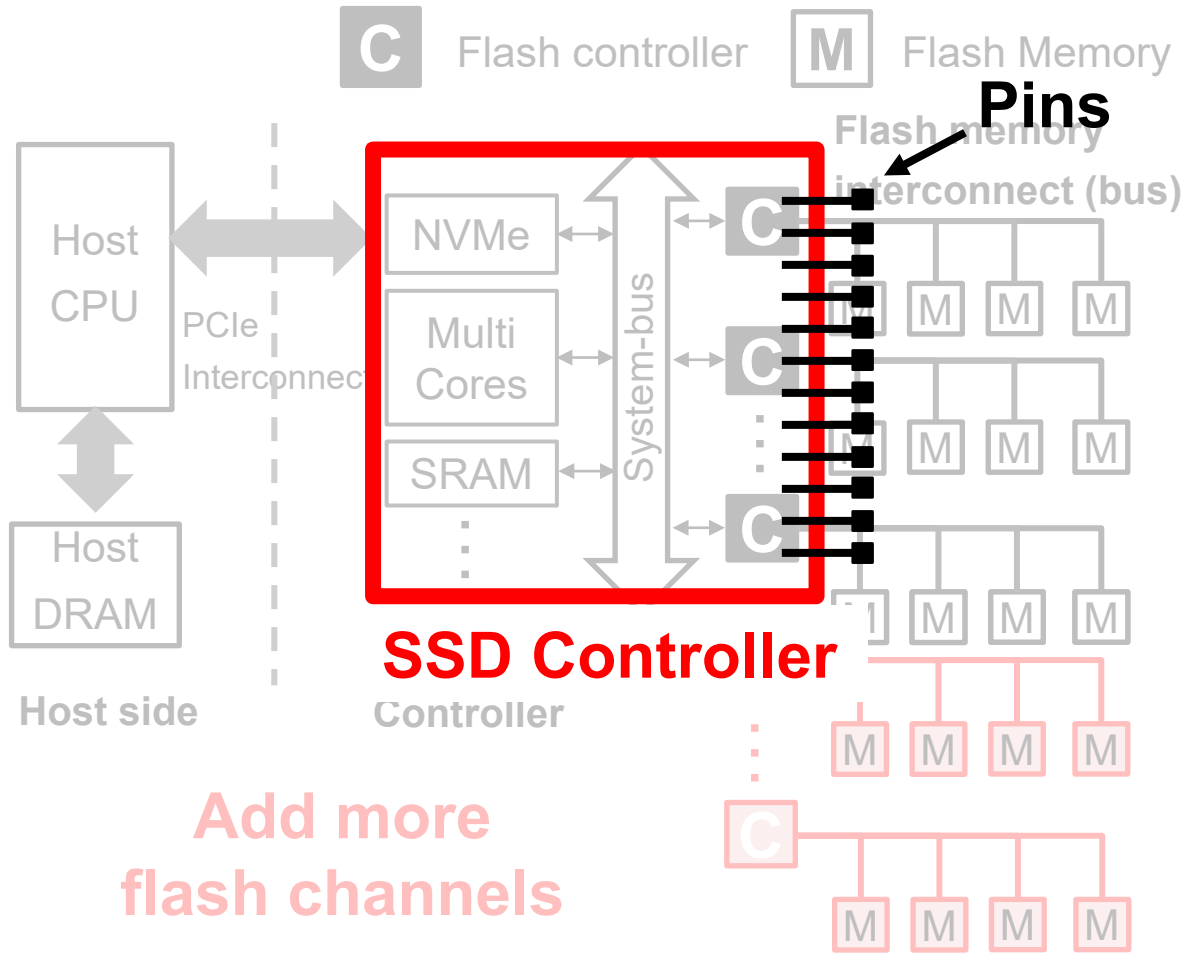
[Kim et al MICRO'22]

Scalable, High Bandwidth SSD



[Kim et al MICRO'22]

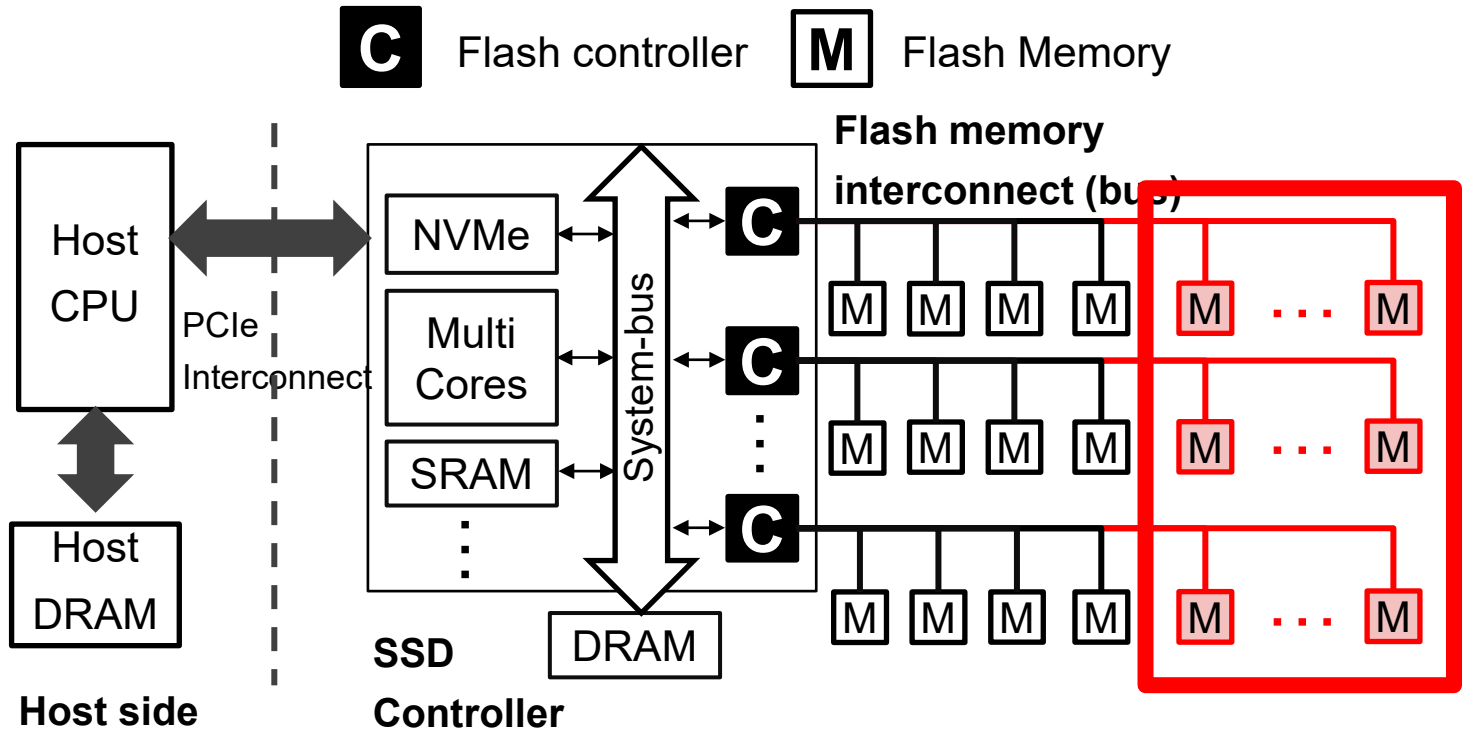
Scalable, High Bandwidth SSD



Limited I/O bandwidth

[Kim et al MICRO'22]

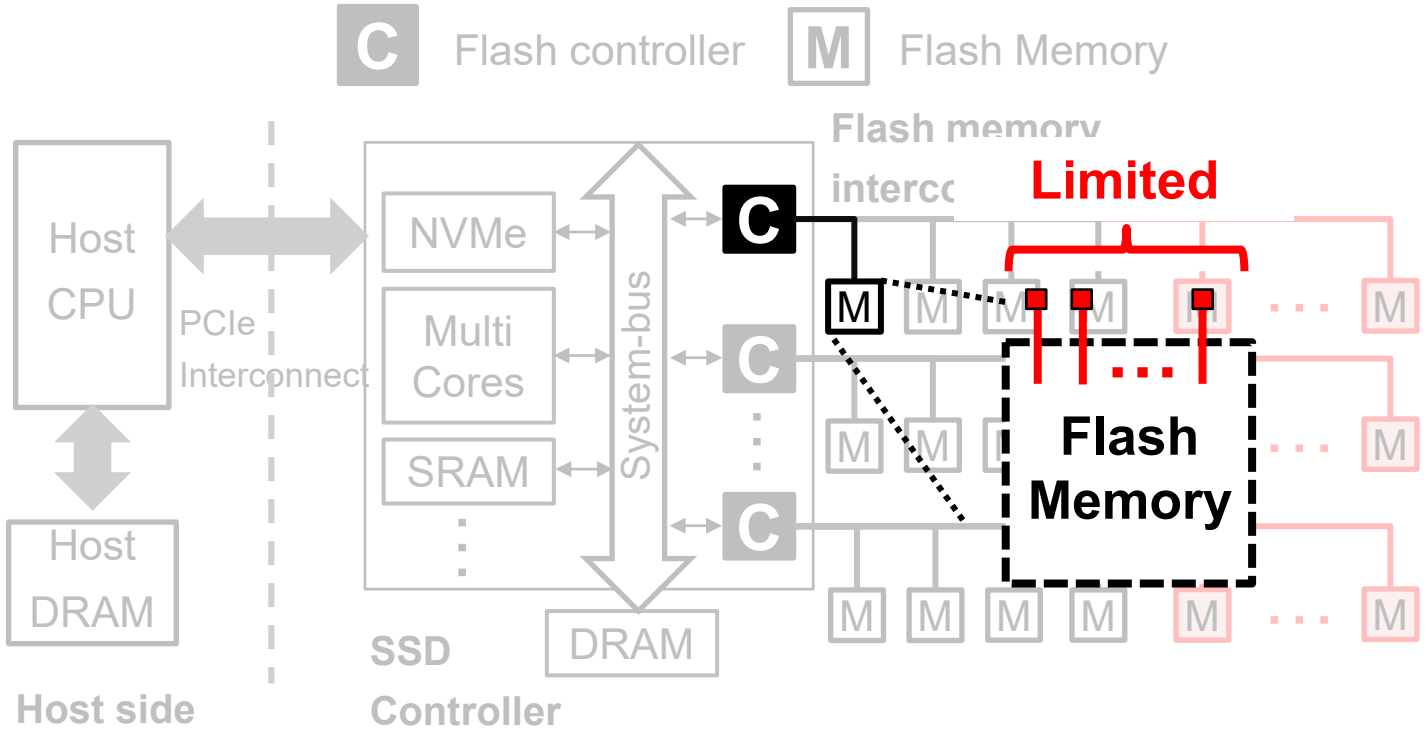
Scalable, High Bandwidth SSD



Add more flash memory

[Kim et al MICRO'22]

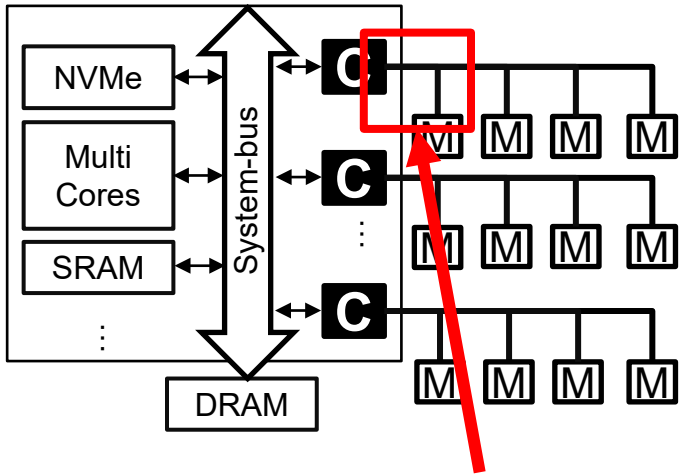
Scalable, High Bandwidth SSD



Flash memory I/O bandwidth is also limited

[Kim et al MICRO'22]

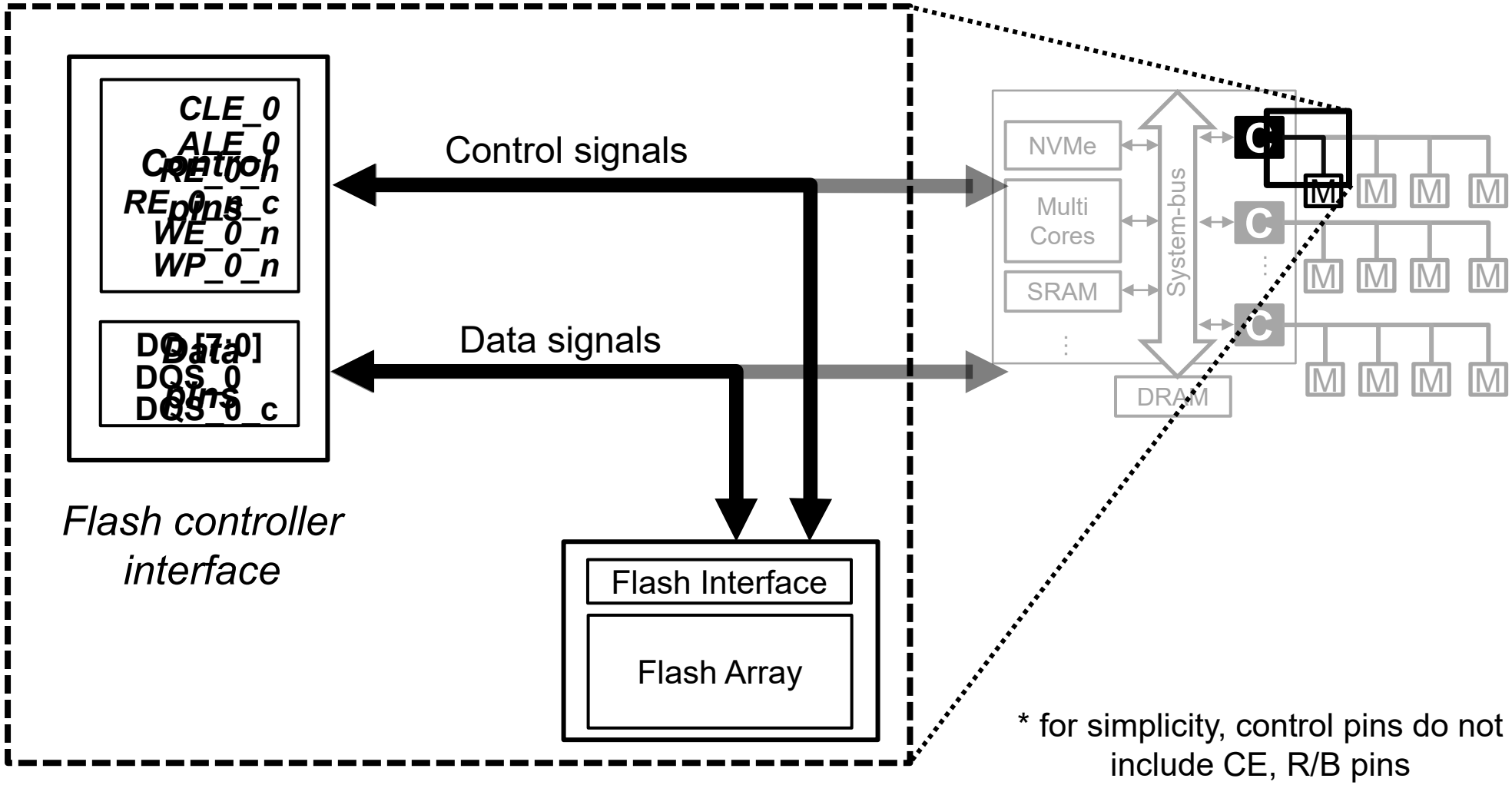
Dedicated Signal Flash Memory Interface



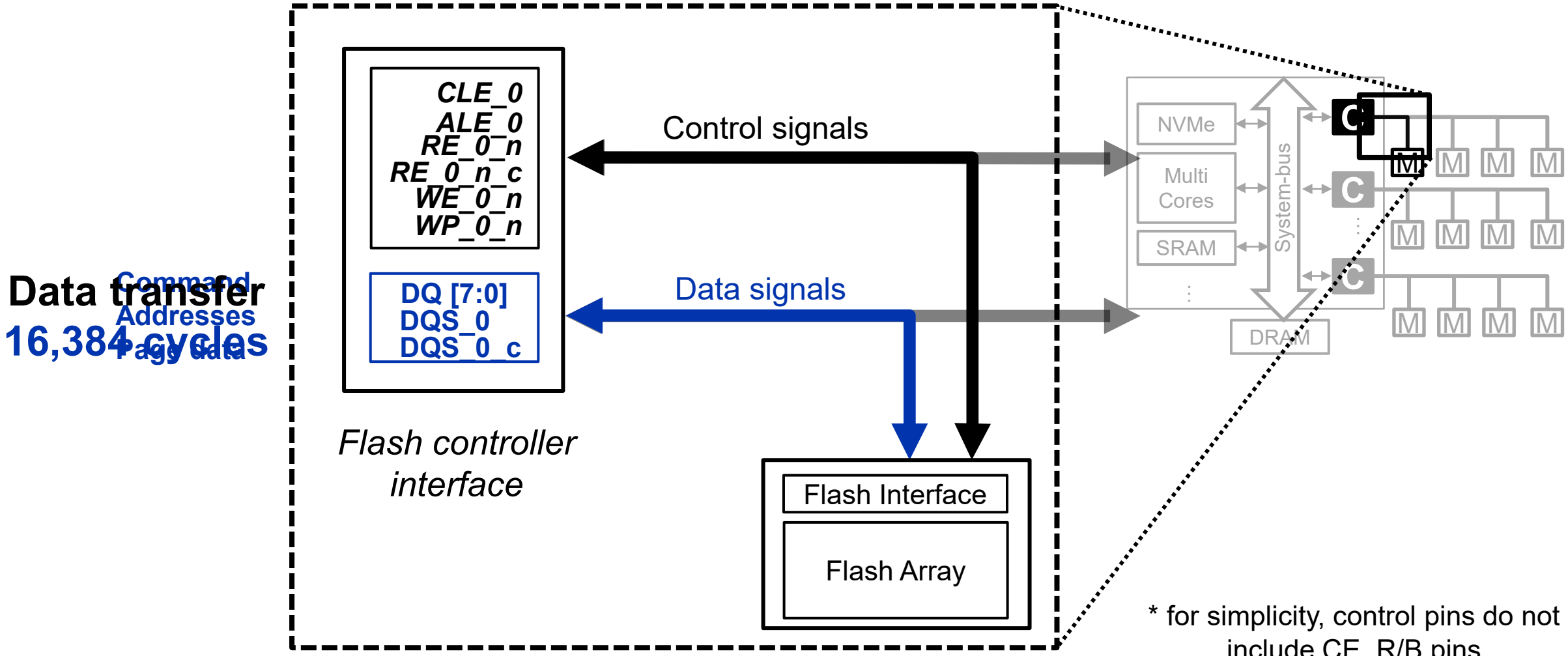
Signal-based flash memory interface pins

Symbols	Type	Description
CLE	Control	Command Latch Enable
ALE	Control	Address Latch Enable
RE	Control	Read Enable
RE_c	Control	Read Enable Complement
WE	Control	Write Enable
WP	Control	Write Protection
CE	Control	Chip Enable
R/B_n	Control	Ready/Busy
DQ[7:0]	Data I/O	Data Input/Outputs
DQS	Data I/O	Data Strobe
DQS_c	Data I/O	Data Strobe Complement

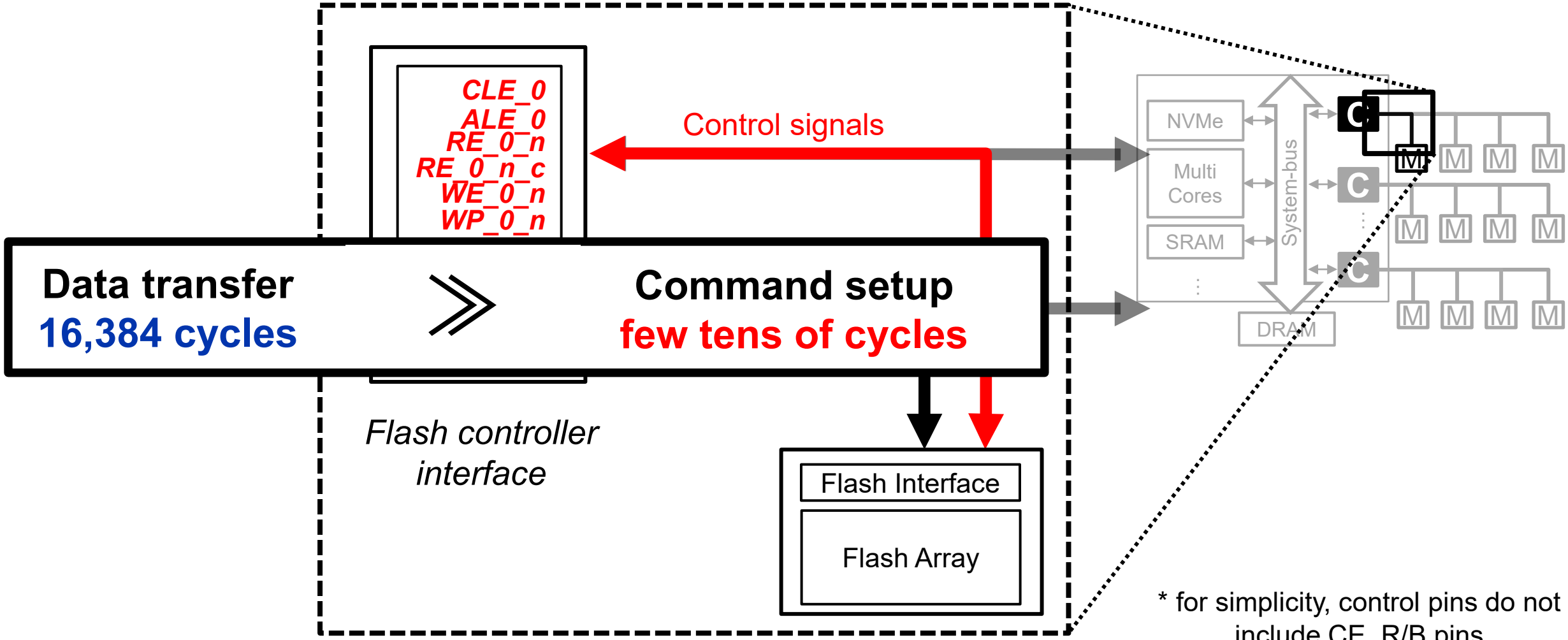
Dedicated Signal Flash Memory Interface



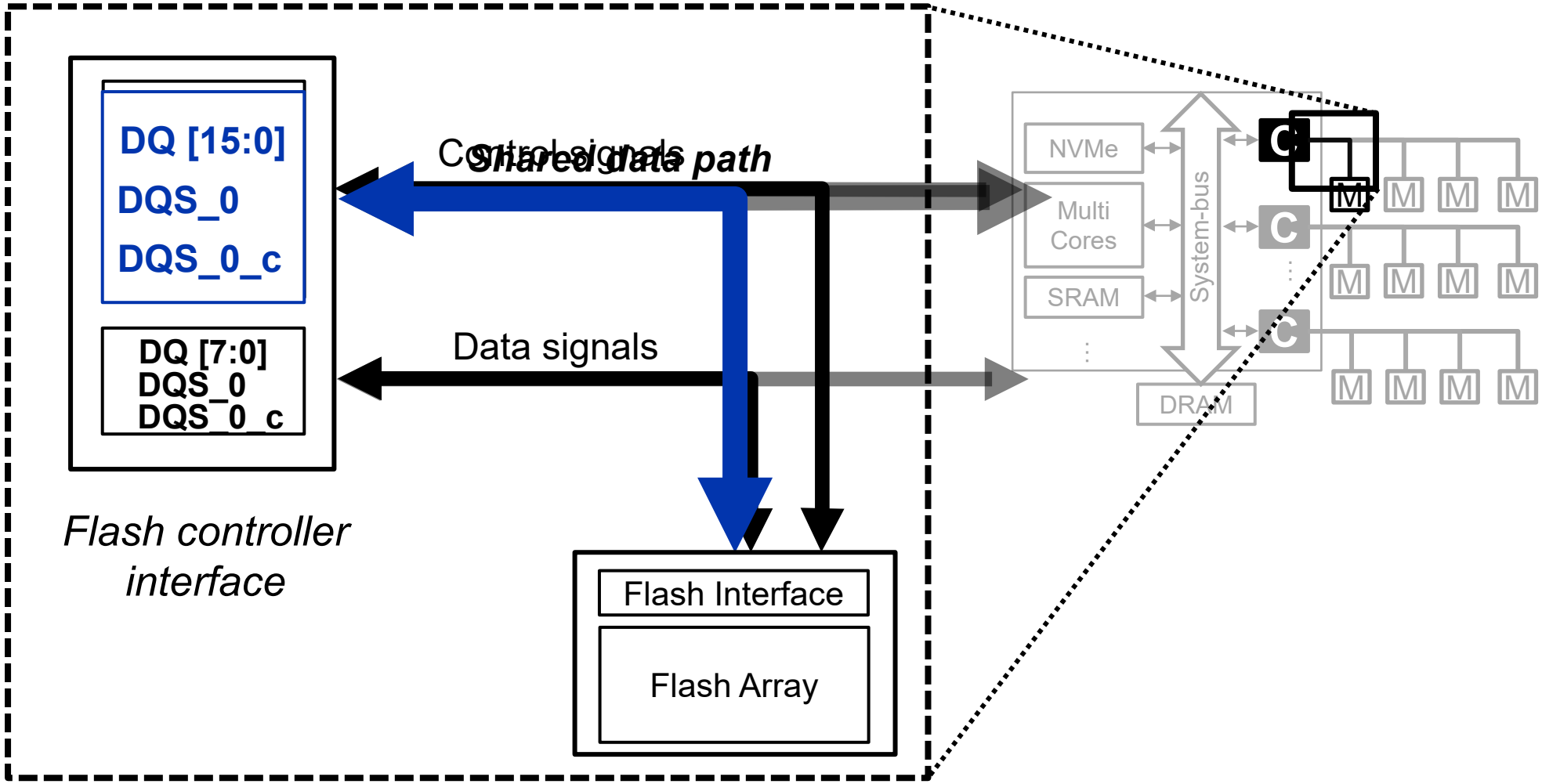
Dedicated Signal Flash Memory Interface



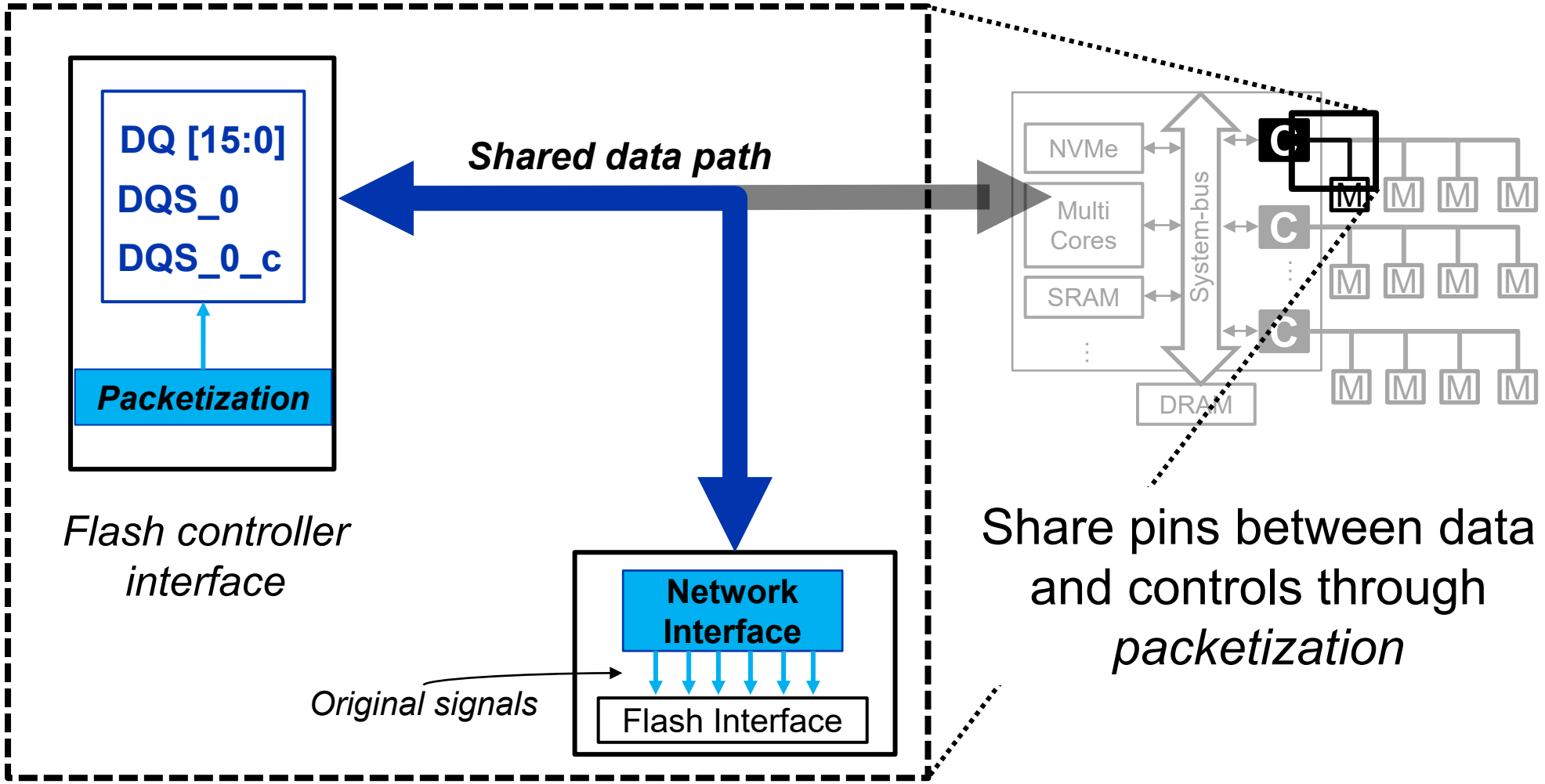
Dedicated Signal Flash Memory Interface



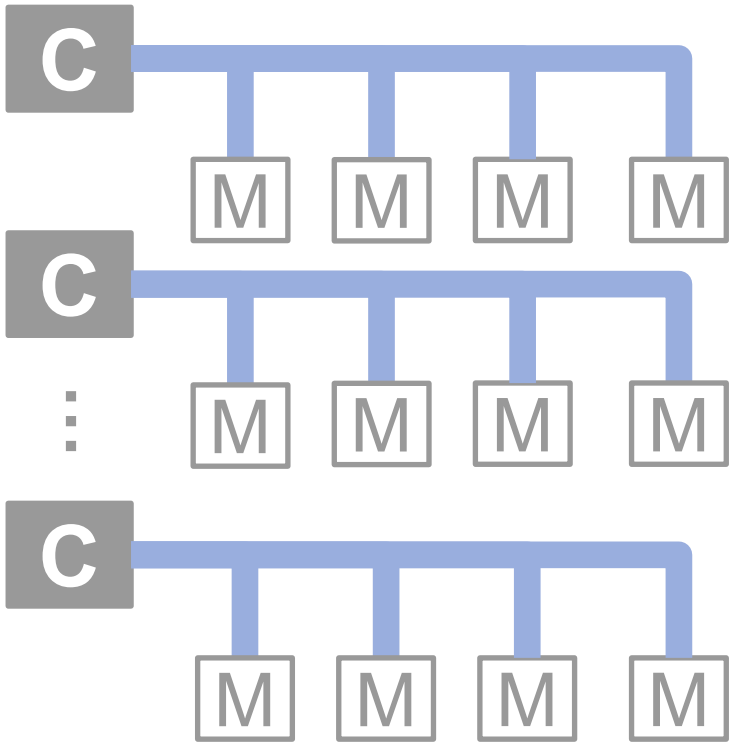
Packetized Flash Memory Interface



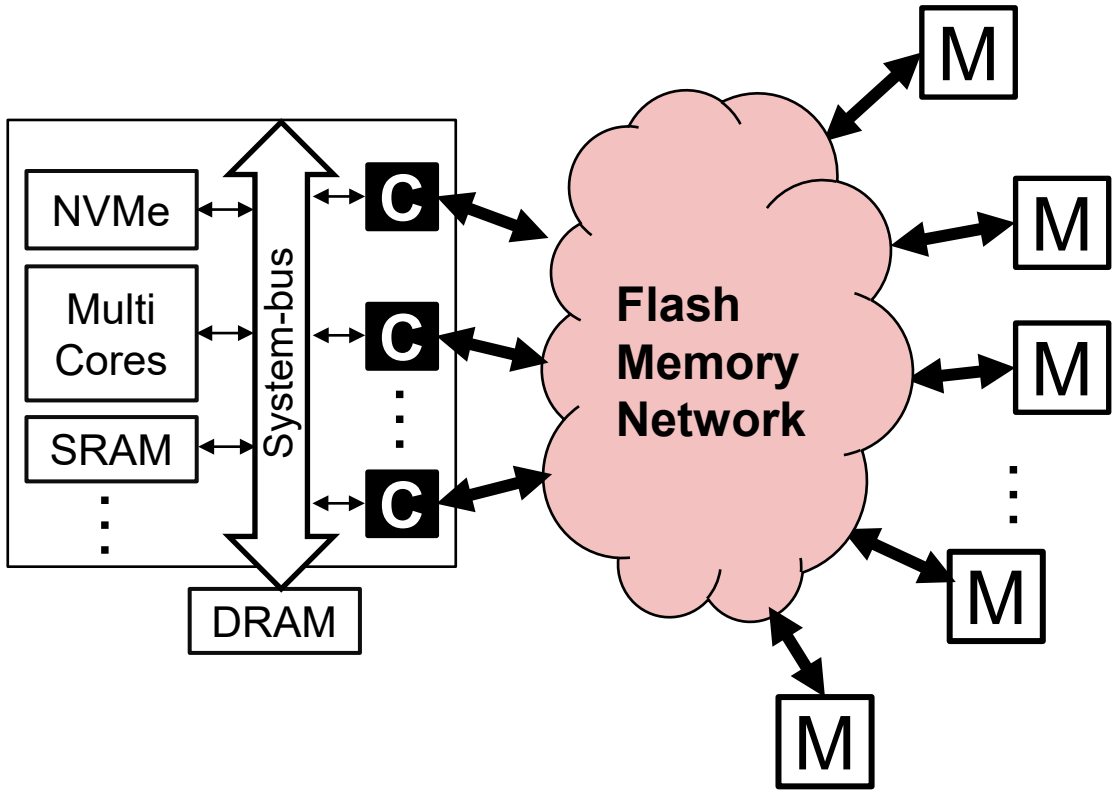
Packetized Flash Memory Interface



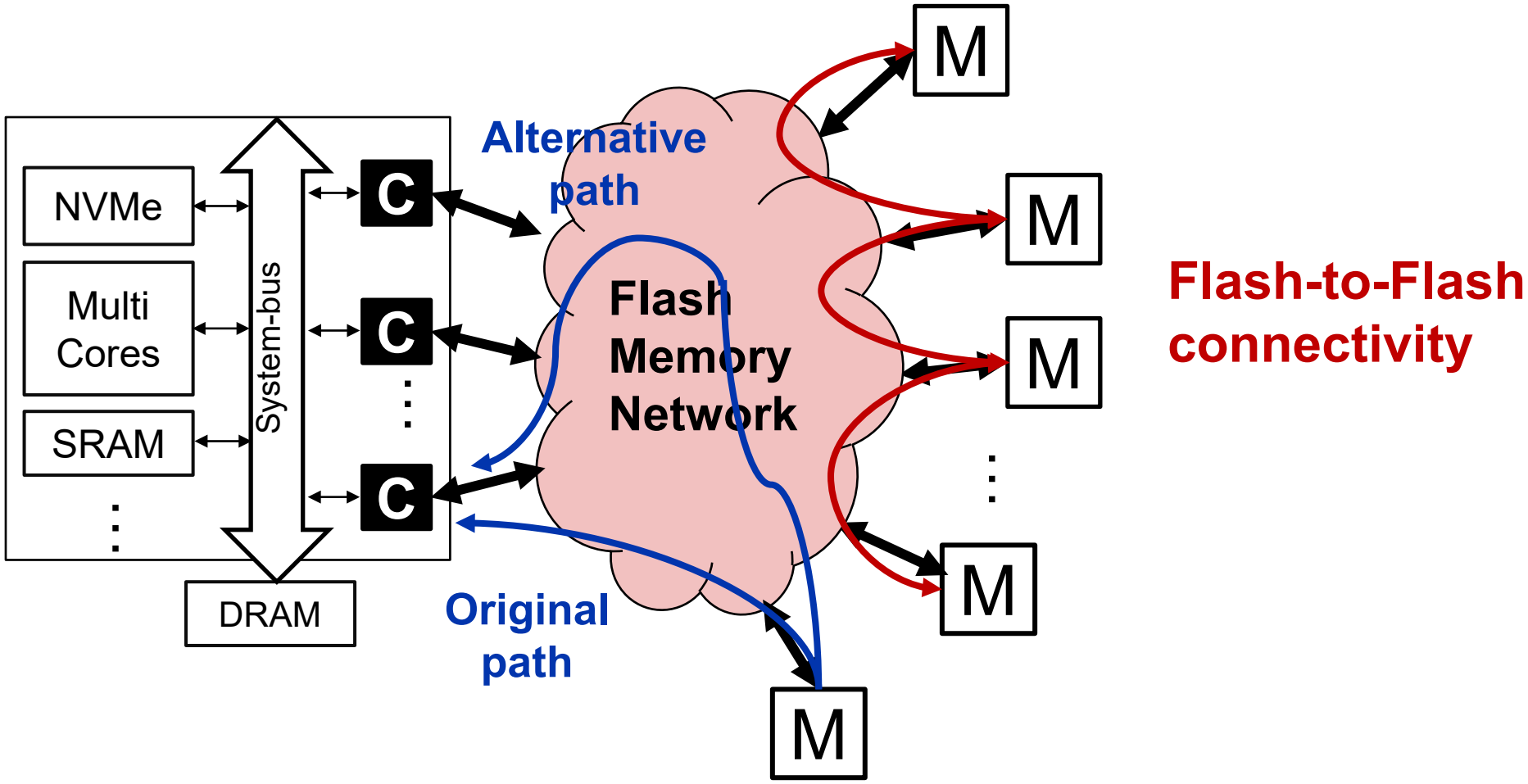
Packetization Increases *Effective* Bandwidth



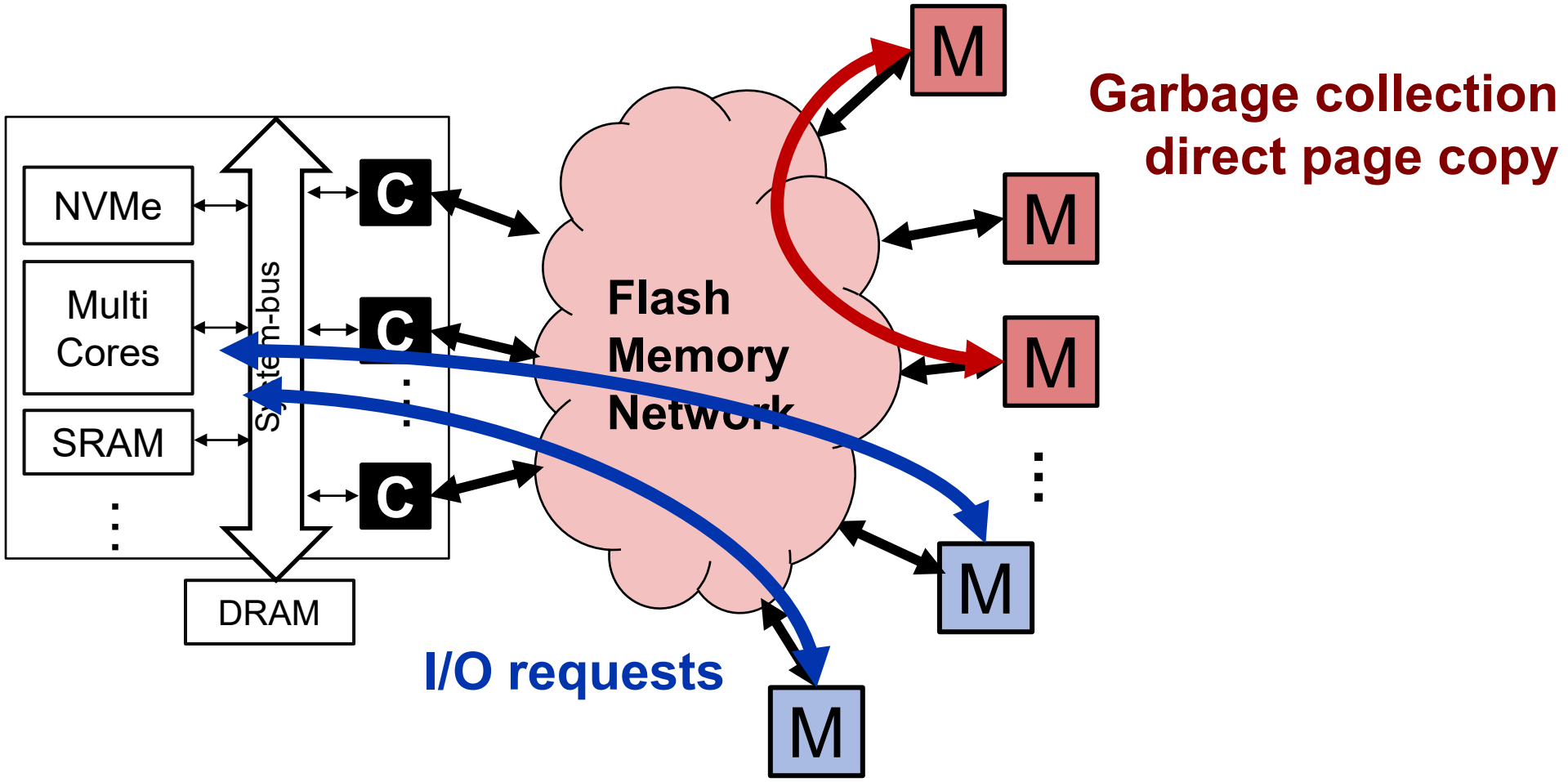
***Packet-based* communication *enables* Interconnection Network**



Benefits of Flash Memory Network

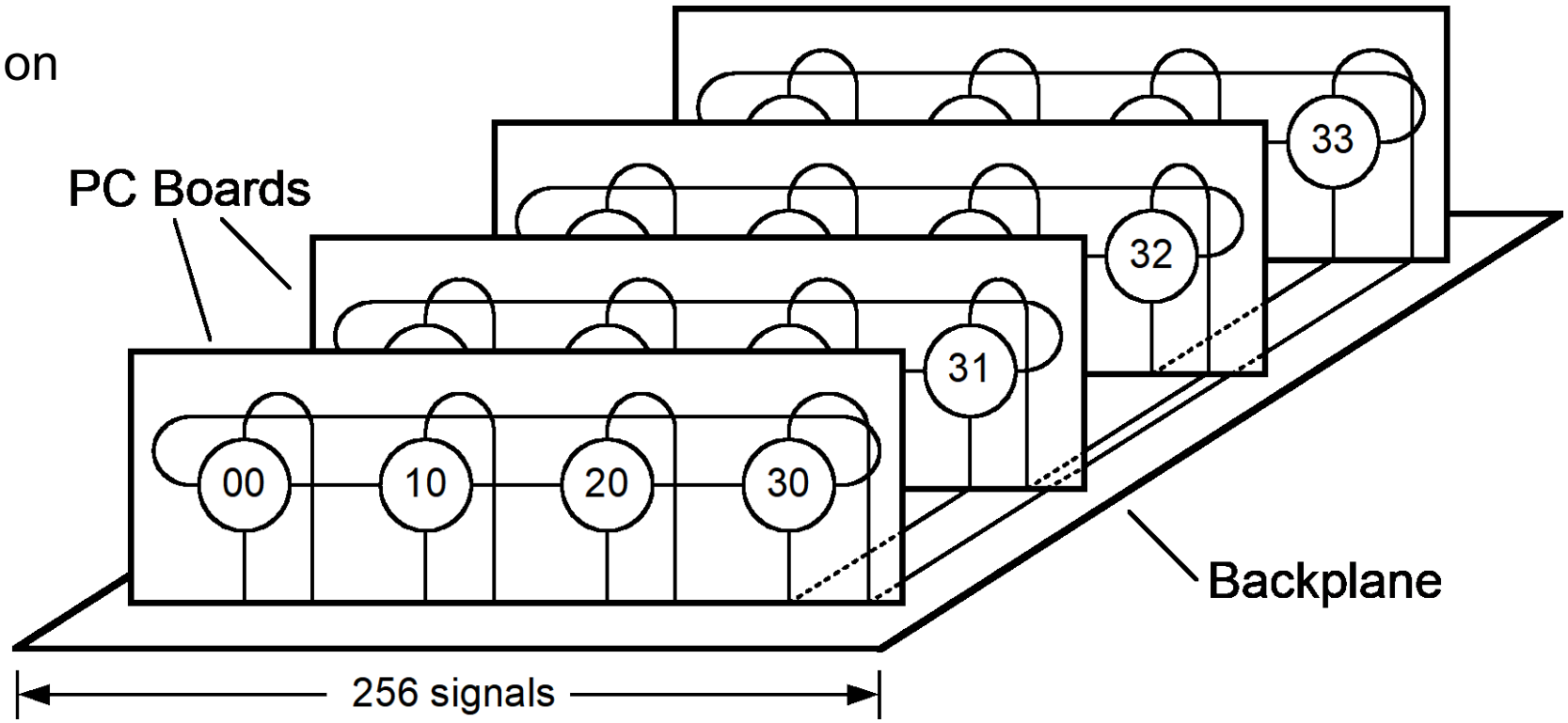


Enabling Spatial Garbage Collection

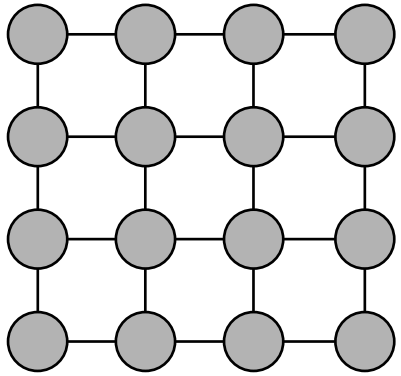


Topology is Constrained by Packaging & Technology

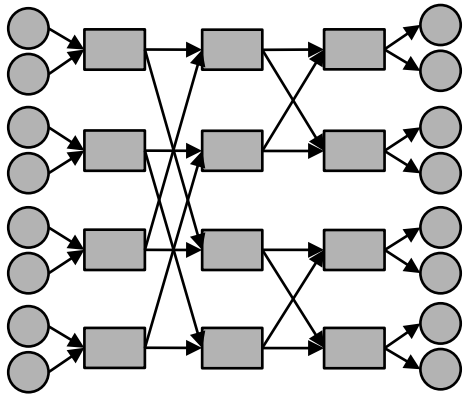
Chip pin count
Board pin count
Cable/backplane bisection



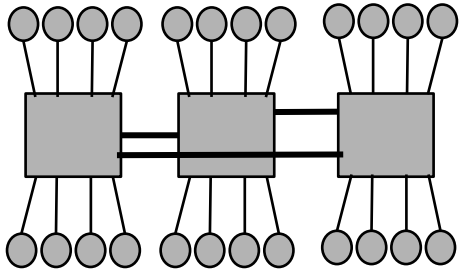
Topology for Flash Memory Interconnection Network



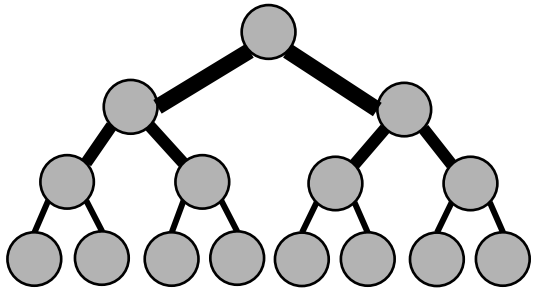
Mesh



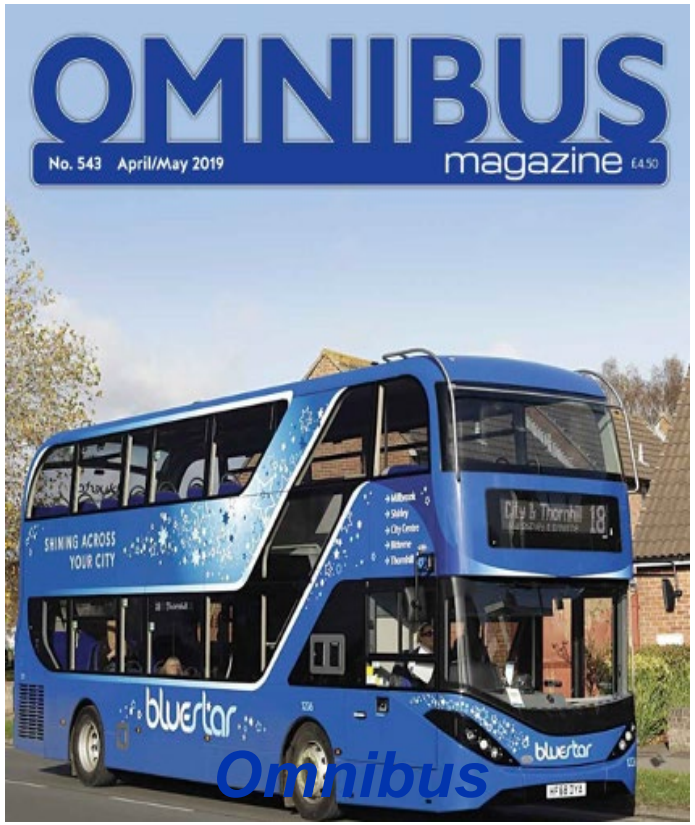
Fly



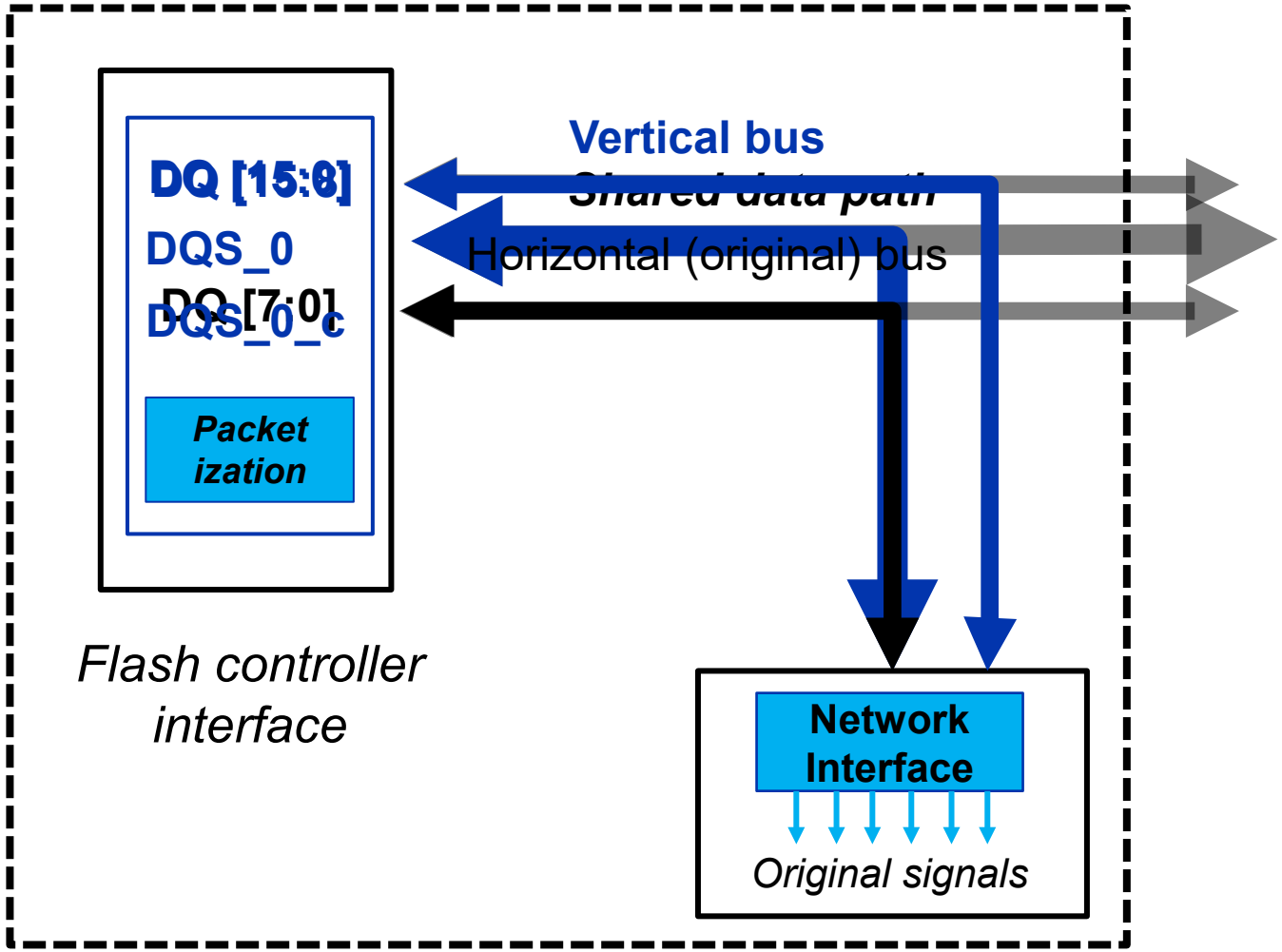
Fully-connected



Fat-tree

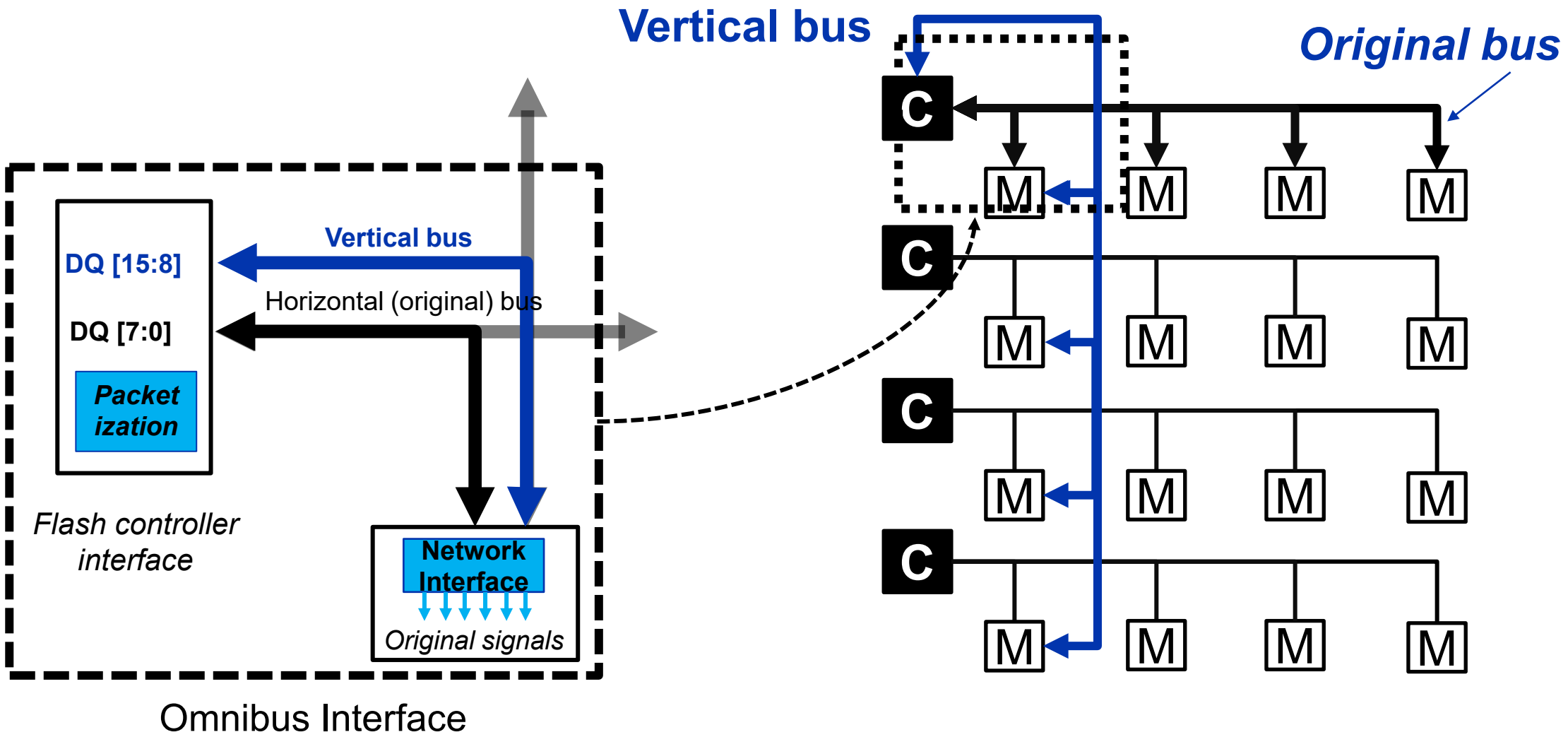


Omnibus topology

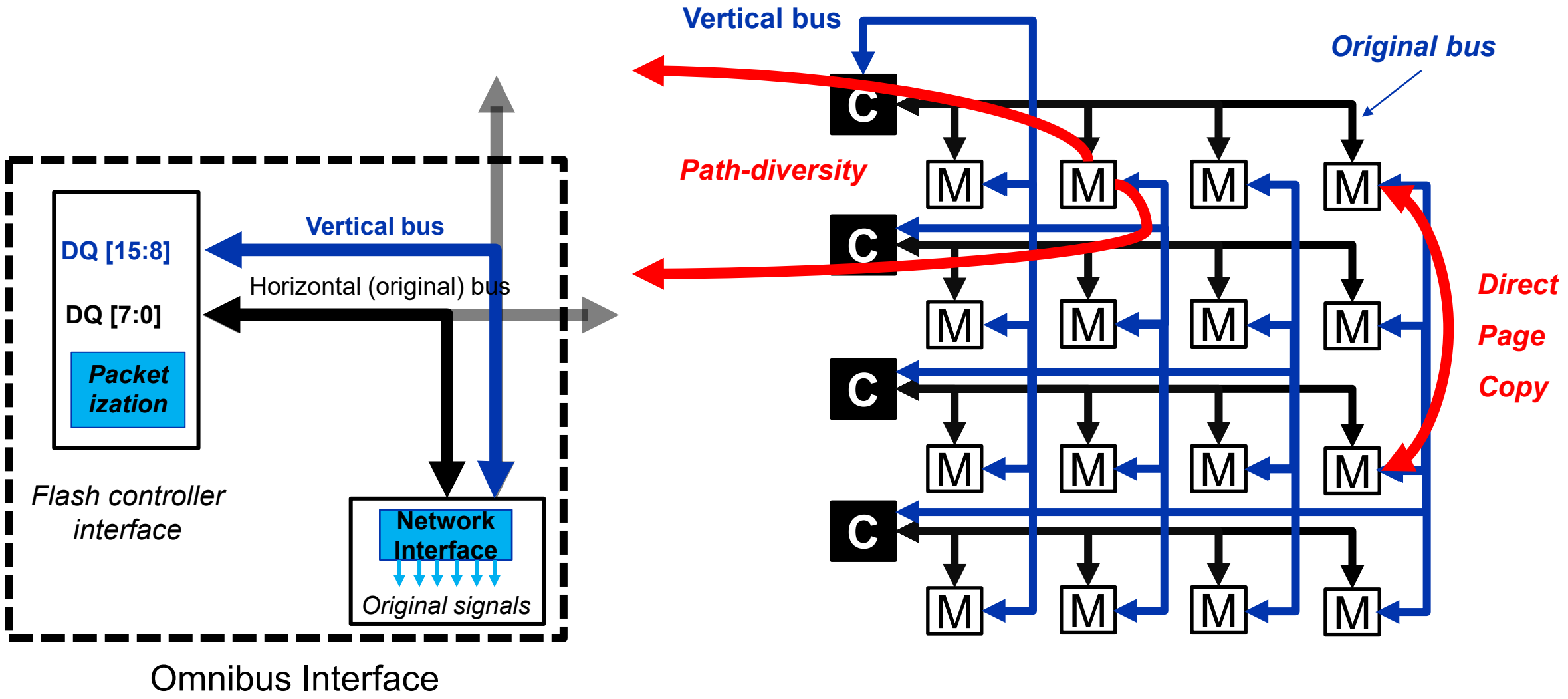


***Insight:* partitioning bandwidth**

Omnibus topology



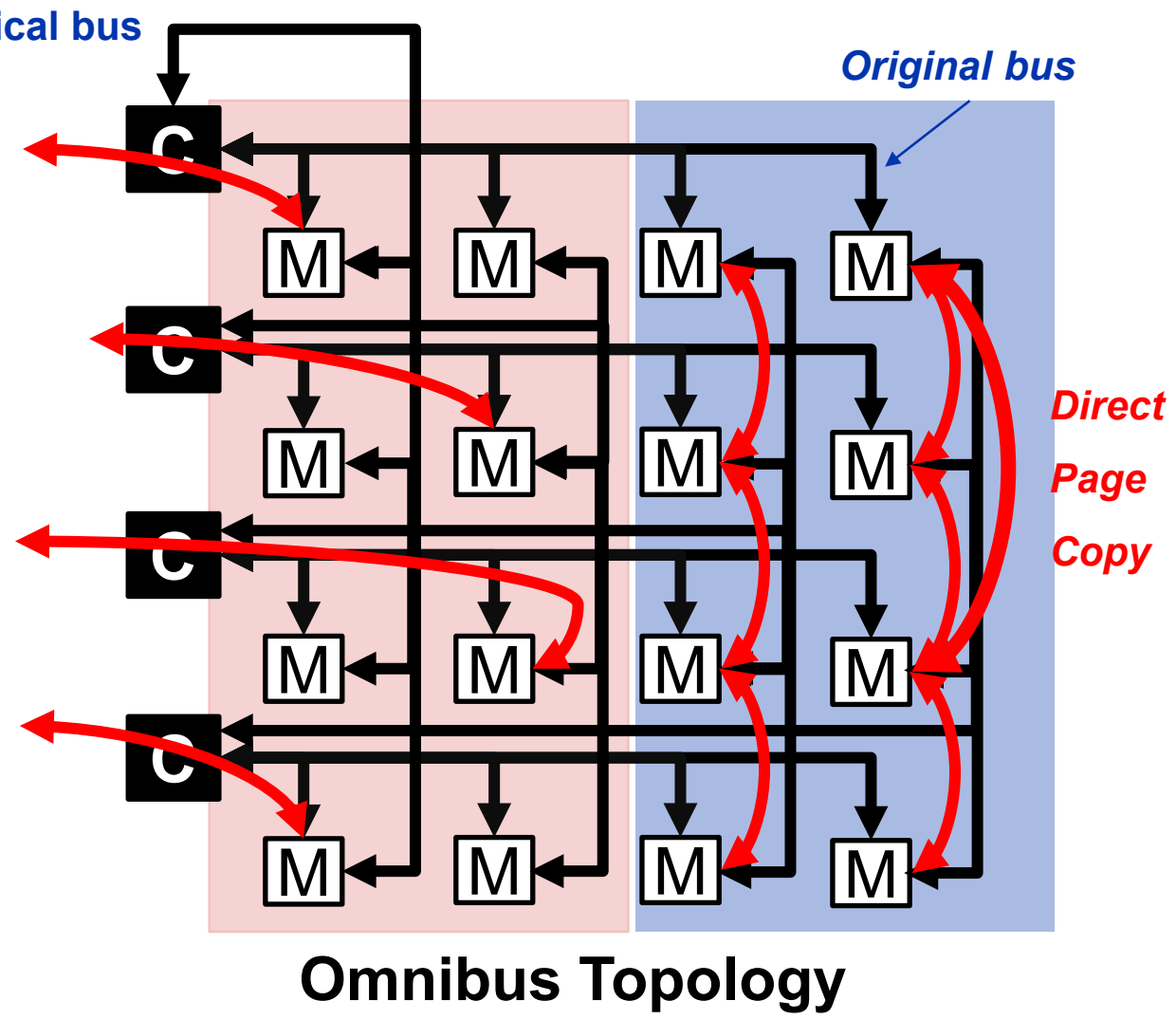
Omnibus topology



Spatial Garbage Collection

I/O request handling through both channels

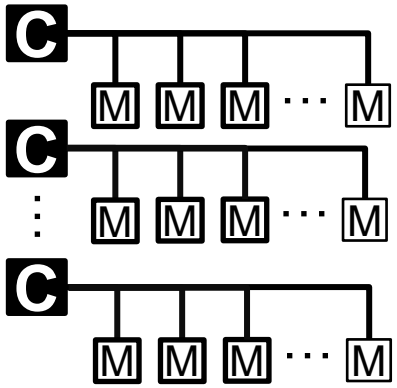
Garbage collection page copies through vertical channels



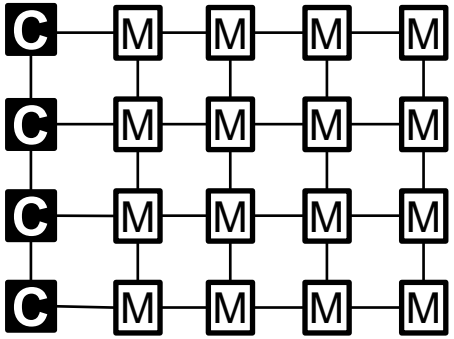
Omnibus Topology

Evaluation Setup

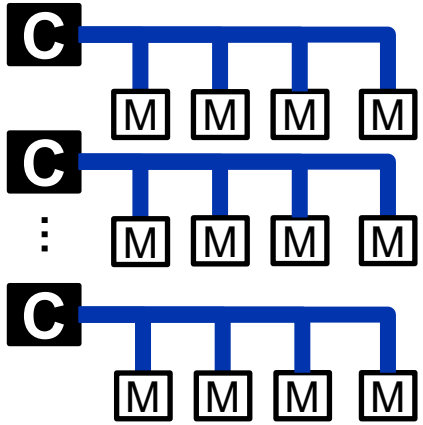
- **SimpleSSD-standalone 2.0 version**
  - NVMe interface (PCIe 4.0 4 lanes), Organization: 8 ch, 8 way, 1 die, 4 planes
  - Flash memory: Ultra-Low Latency parameters –
  - Flash bus: 2D structure modification
  - Garbage collection: victim (greedy) / free (global) selection – parallel GC
- **Workloads**:
  - Synthetic workloads and trace-driven evaluation



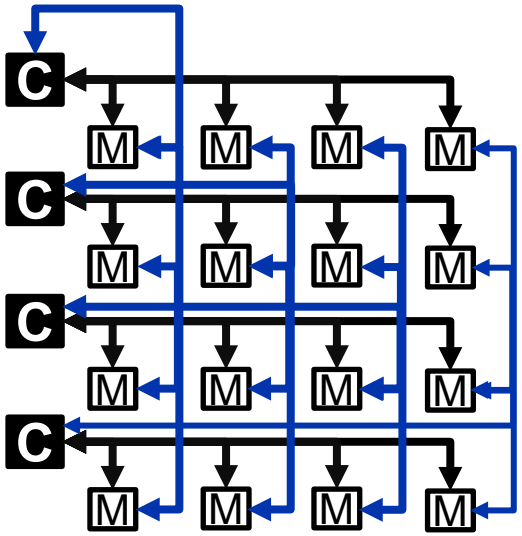
baselineSSD



NoSSD (mesh)



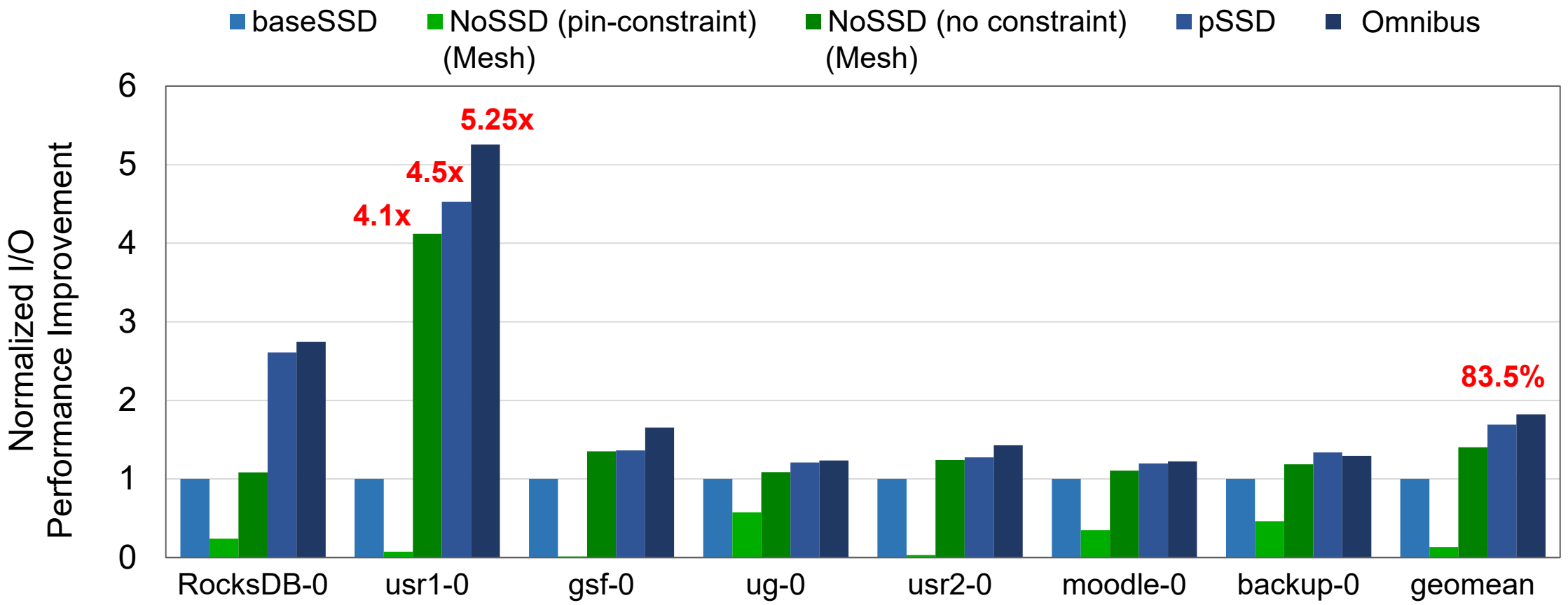
pSSD



Omnibus

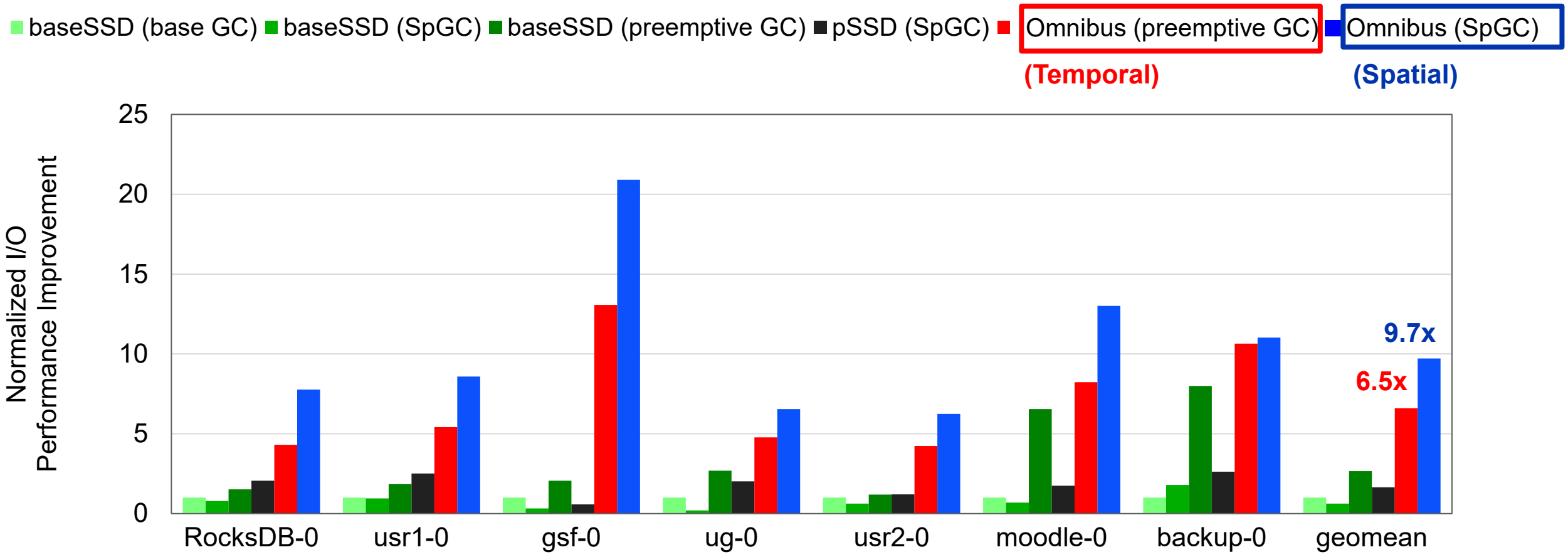
I/O performance

83.5% of average I/O latency has improved



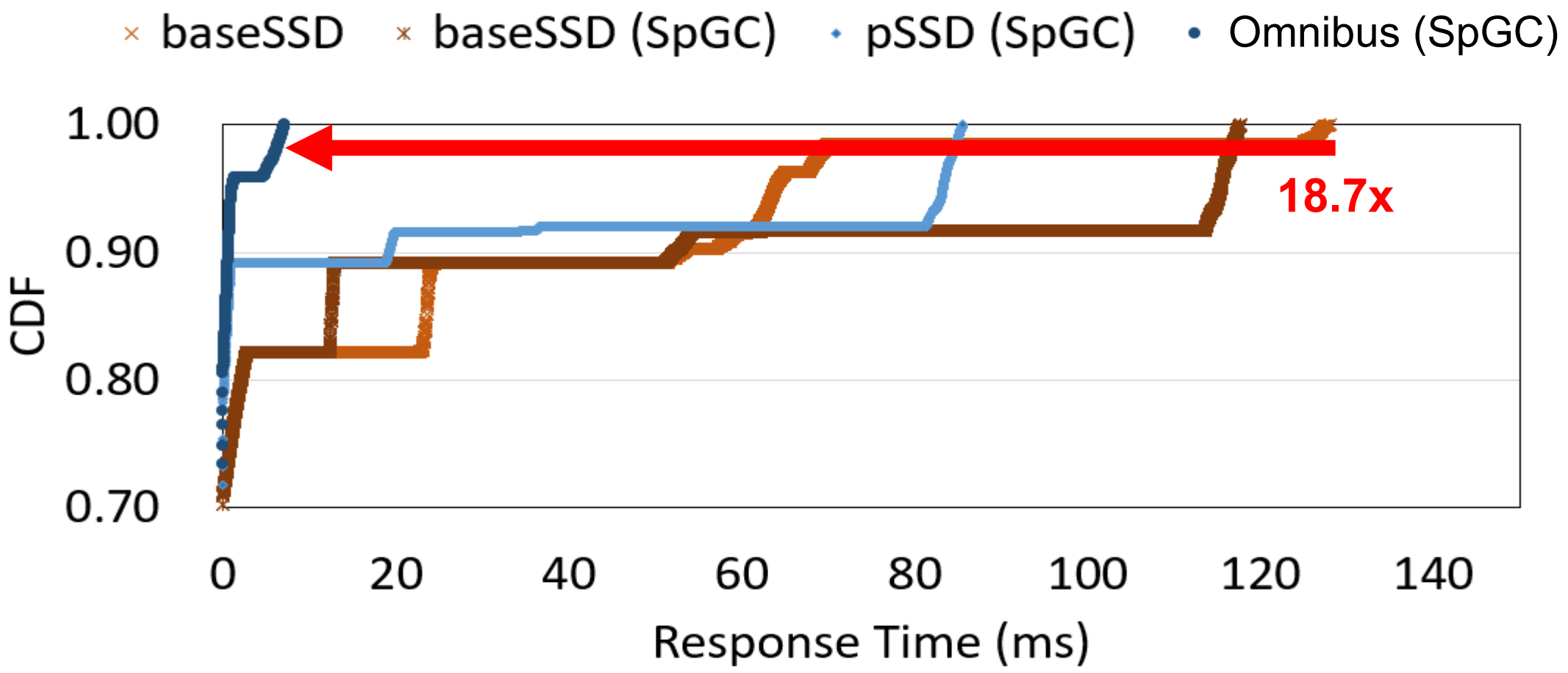
I/O and GC interference

9.7x average I/O latency improvement



Tail-latency

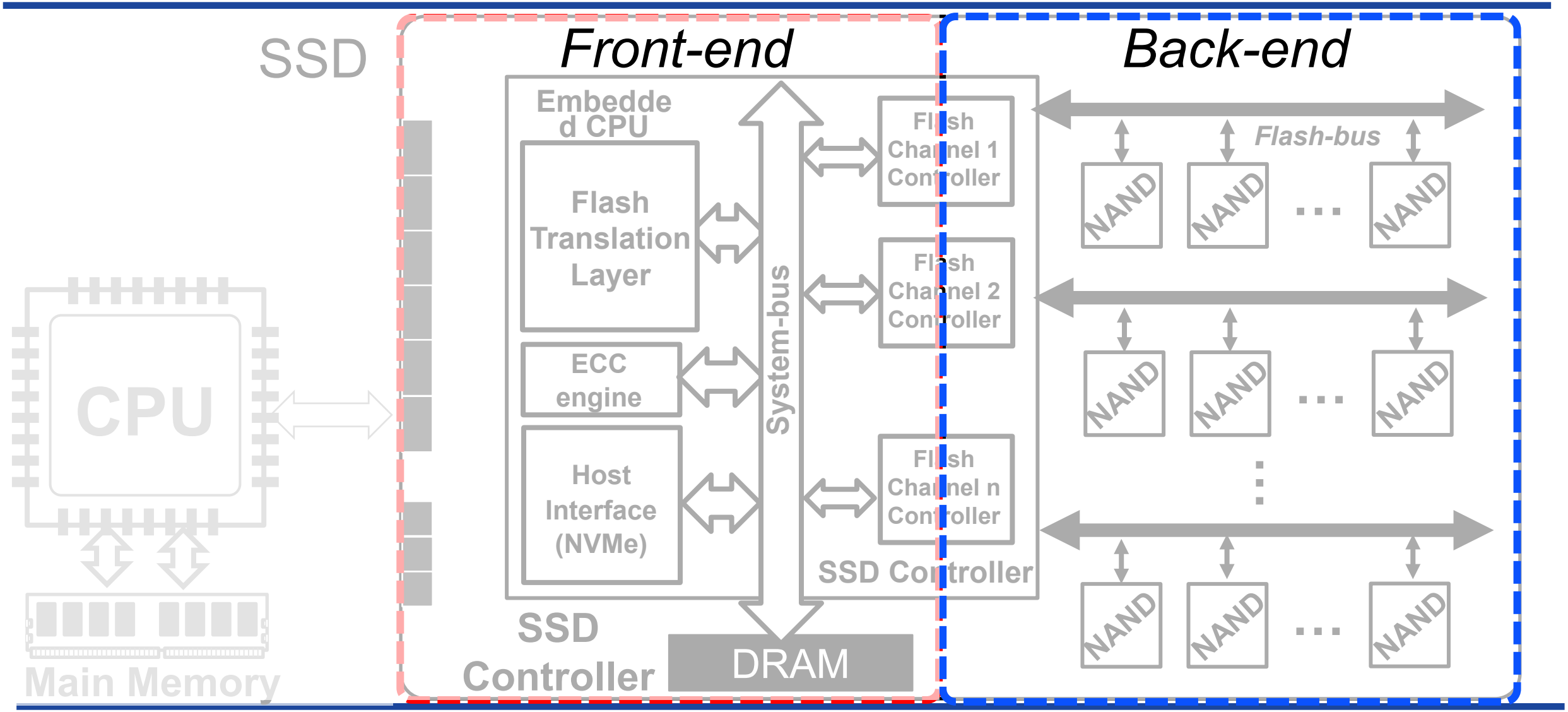
Over 18x reduction in tail-latency



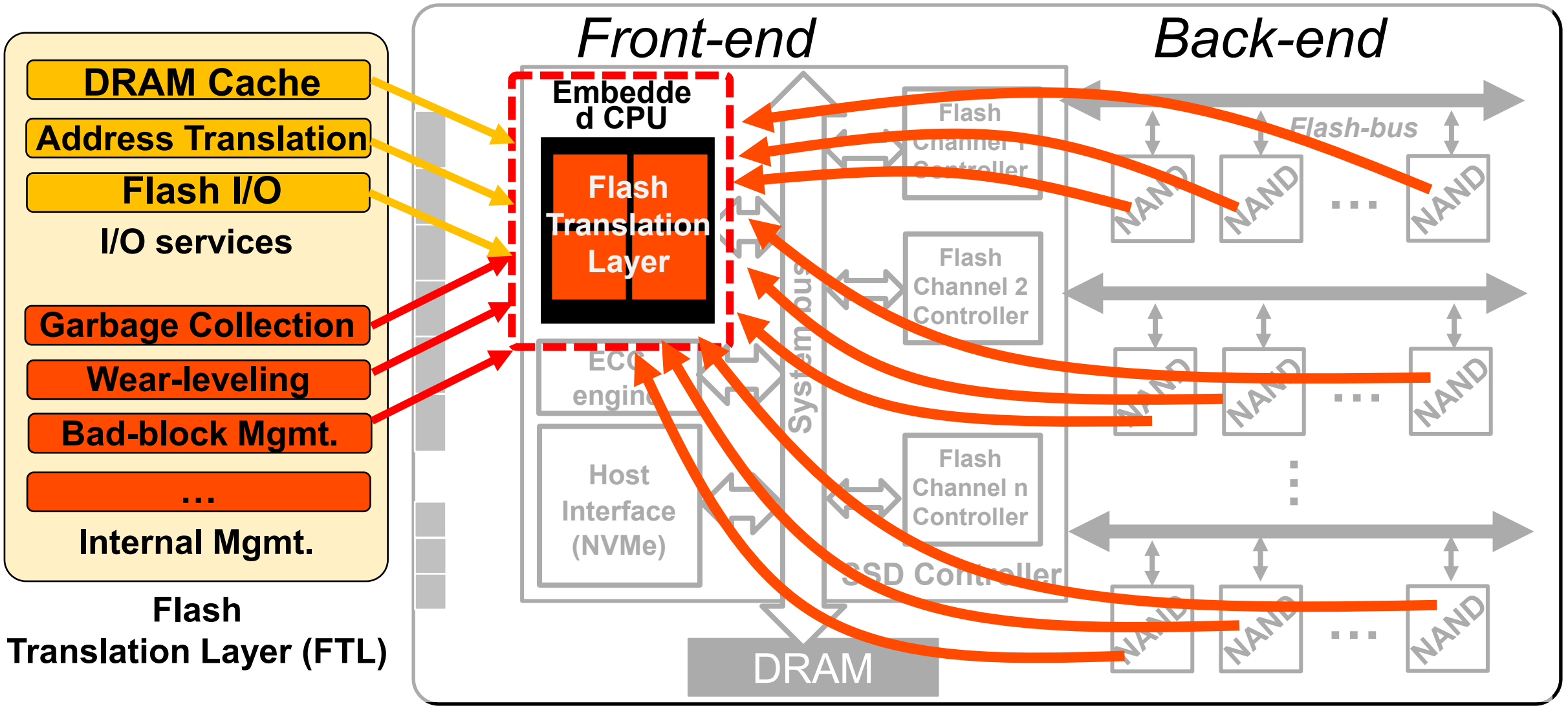
Decoupled SSD: Rethinking SSD Architecture through Network-based Flash Controllers

[ISCA'23]

Modern SSD architecture



Modern SSD architecture



Modern SSDs are *tightly*-coupled

DRAM Cache

Address Translation

Flash I/O

I/O services

Garbage Collection

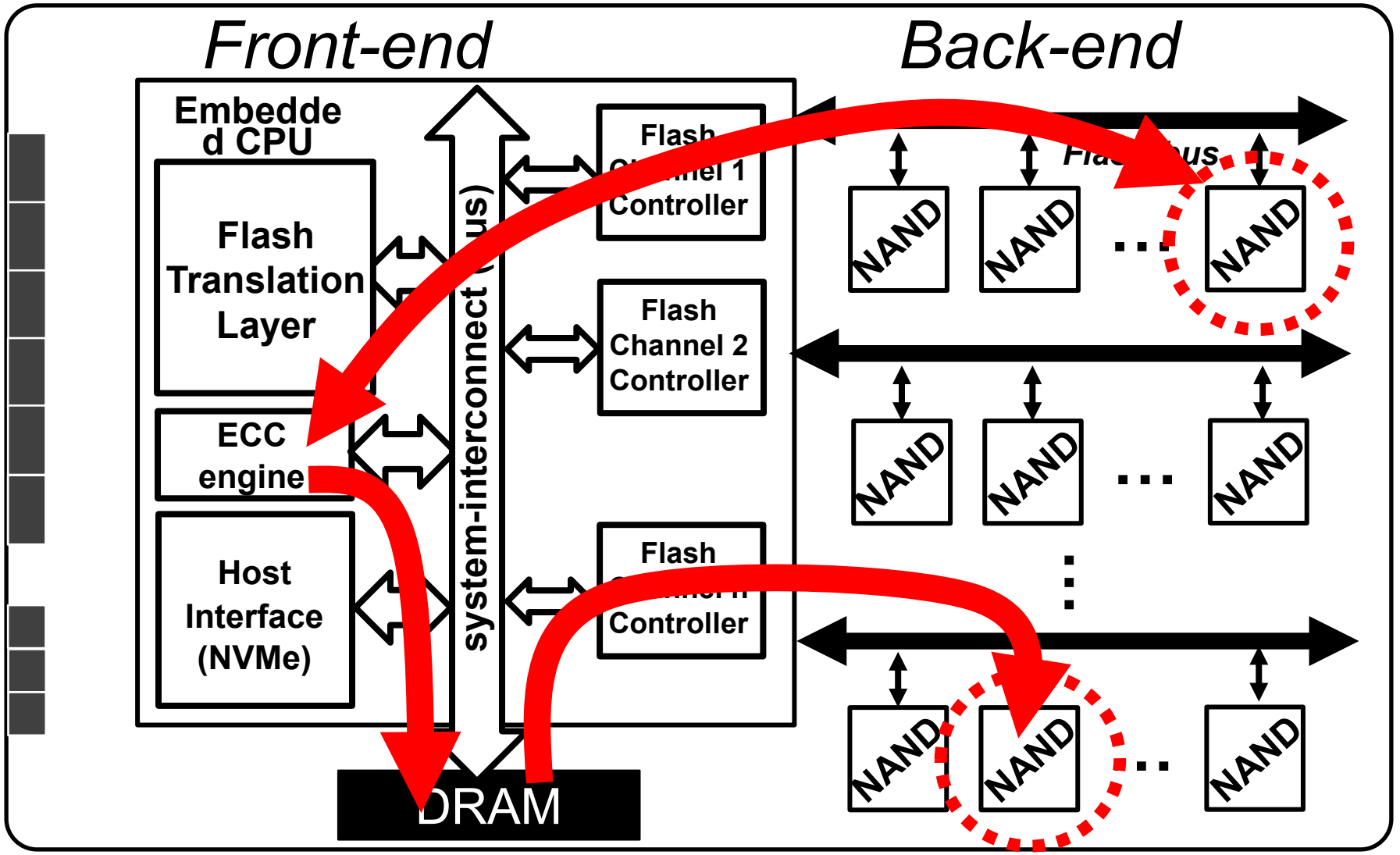
Wear-leveling

Bad-block Mgmt.

…

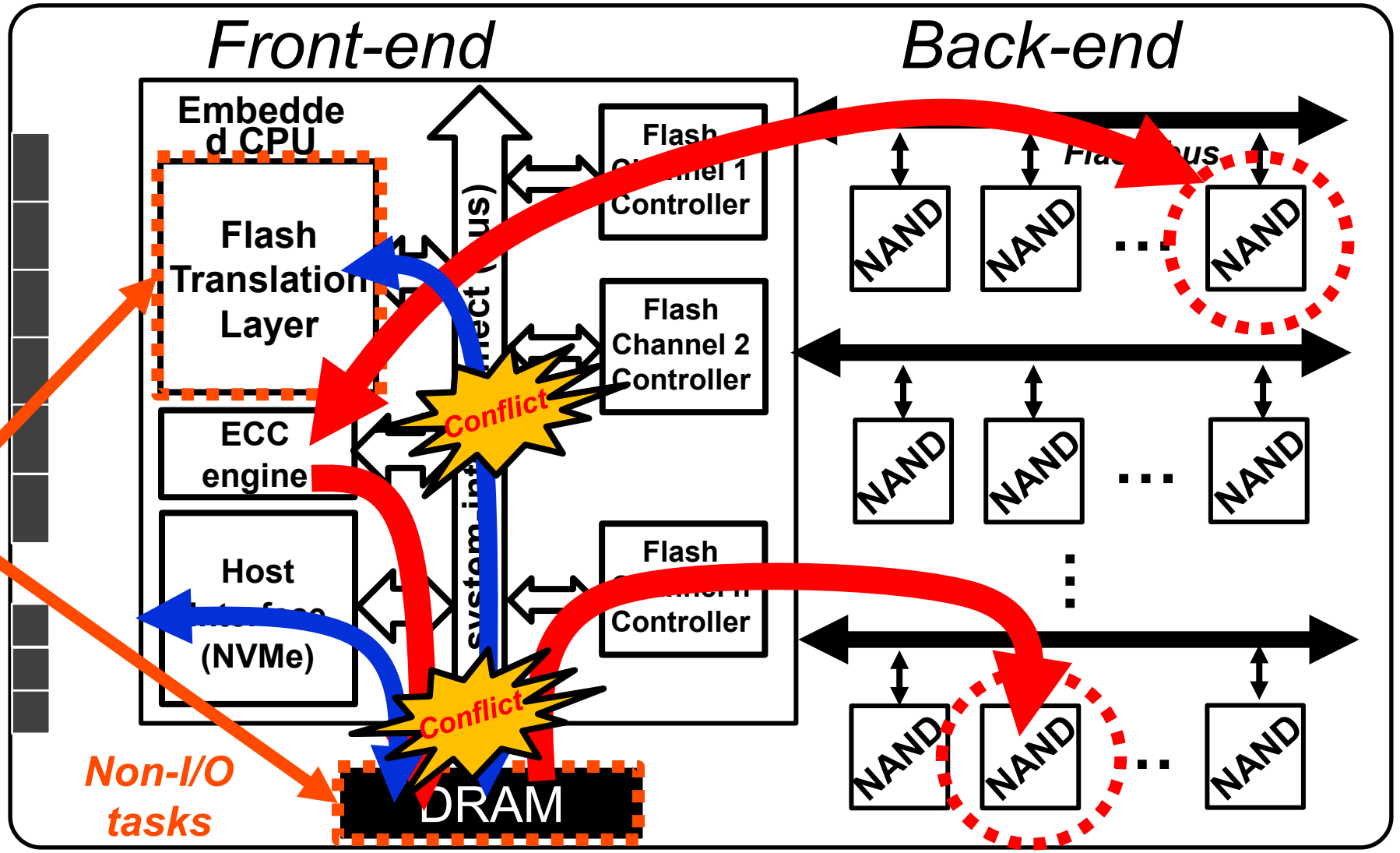
Internal Mgmt.

Flash Translation Layer (FTL)



Limitations of modern SSD

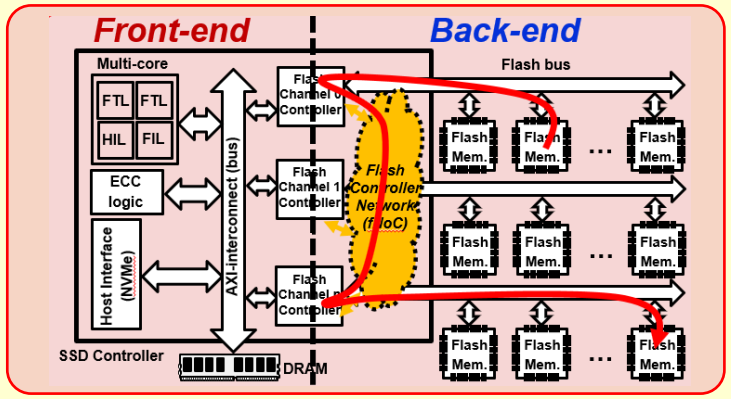
1. Indirect Data Movement
2. Unnecessary Interference
3. Waste of Compute and Memory

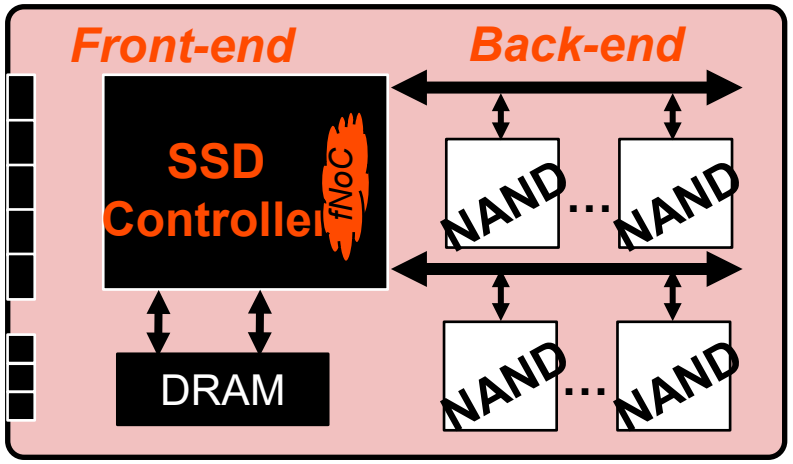


Decoupled SSD (dSSD) architecture

Garbage Collection

- Efficient Data movement
- Advanced commands such as global copy-back





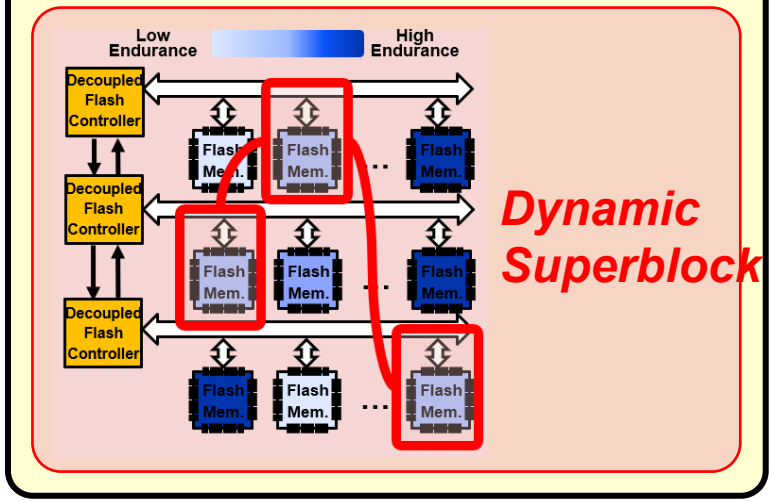
Decoupled SSD Architecture

Maximize I/O performance

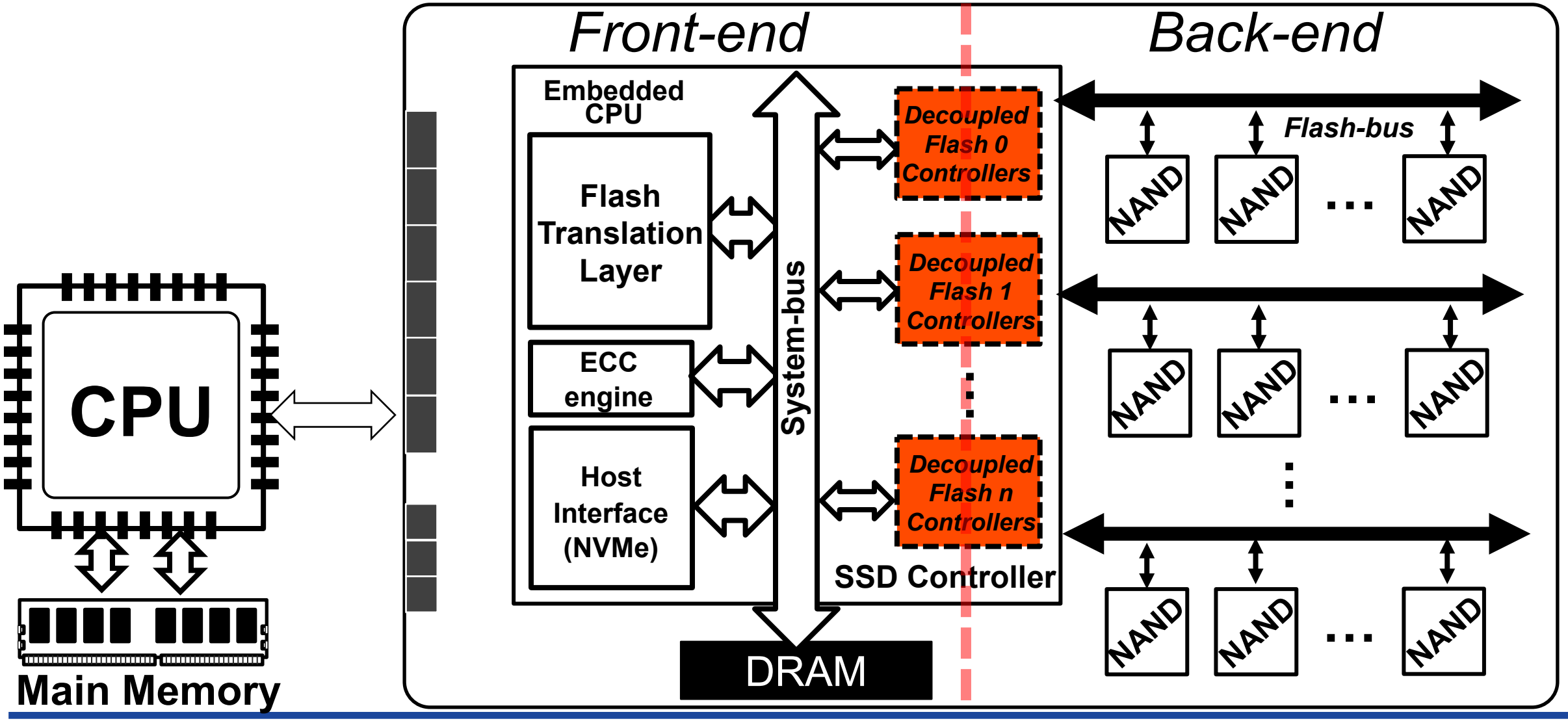
- *Minimize interference between I/O and GC*
- *Minimized tail-latency*

Bad-block Management

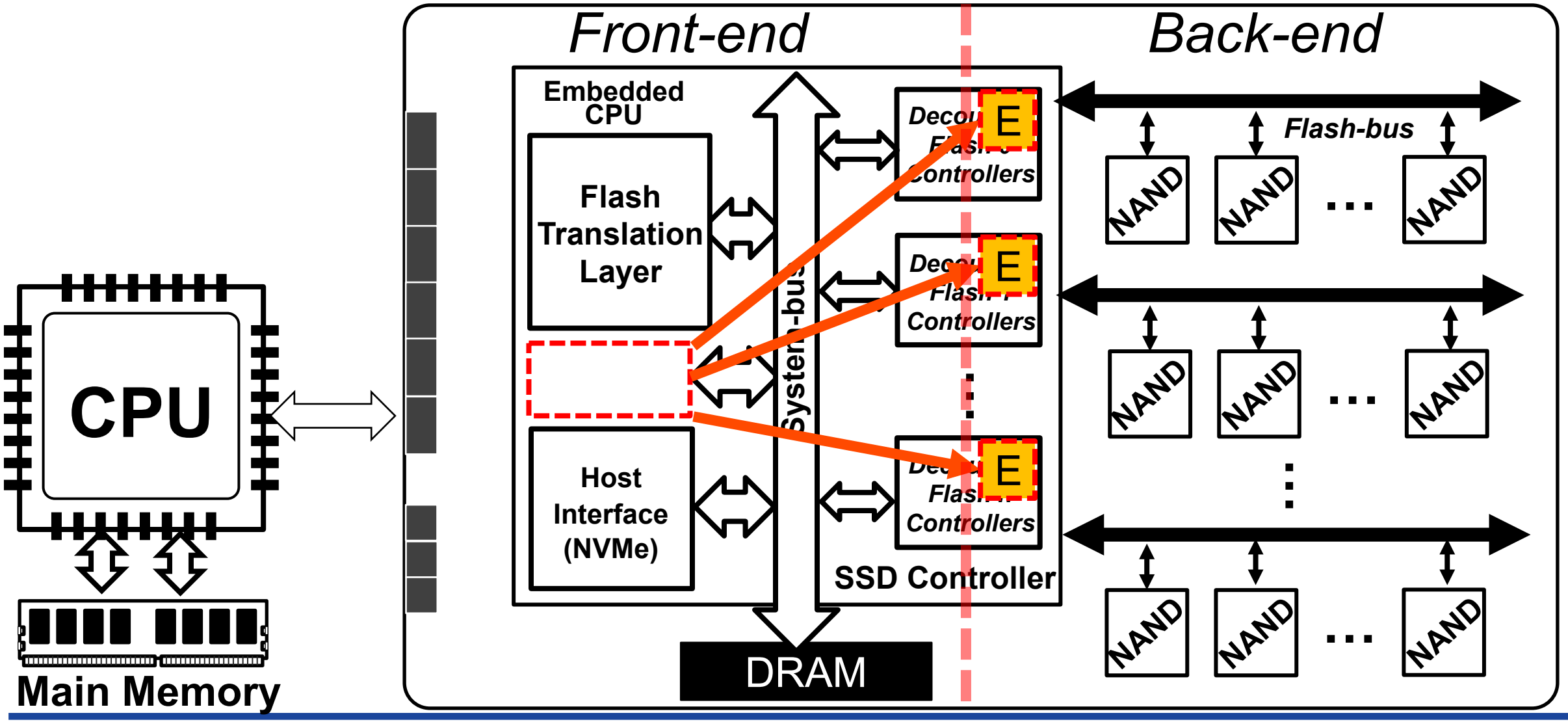
- Support (hardware) dynamic superblock formation
- Offload remapping



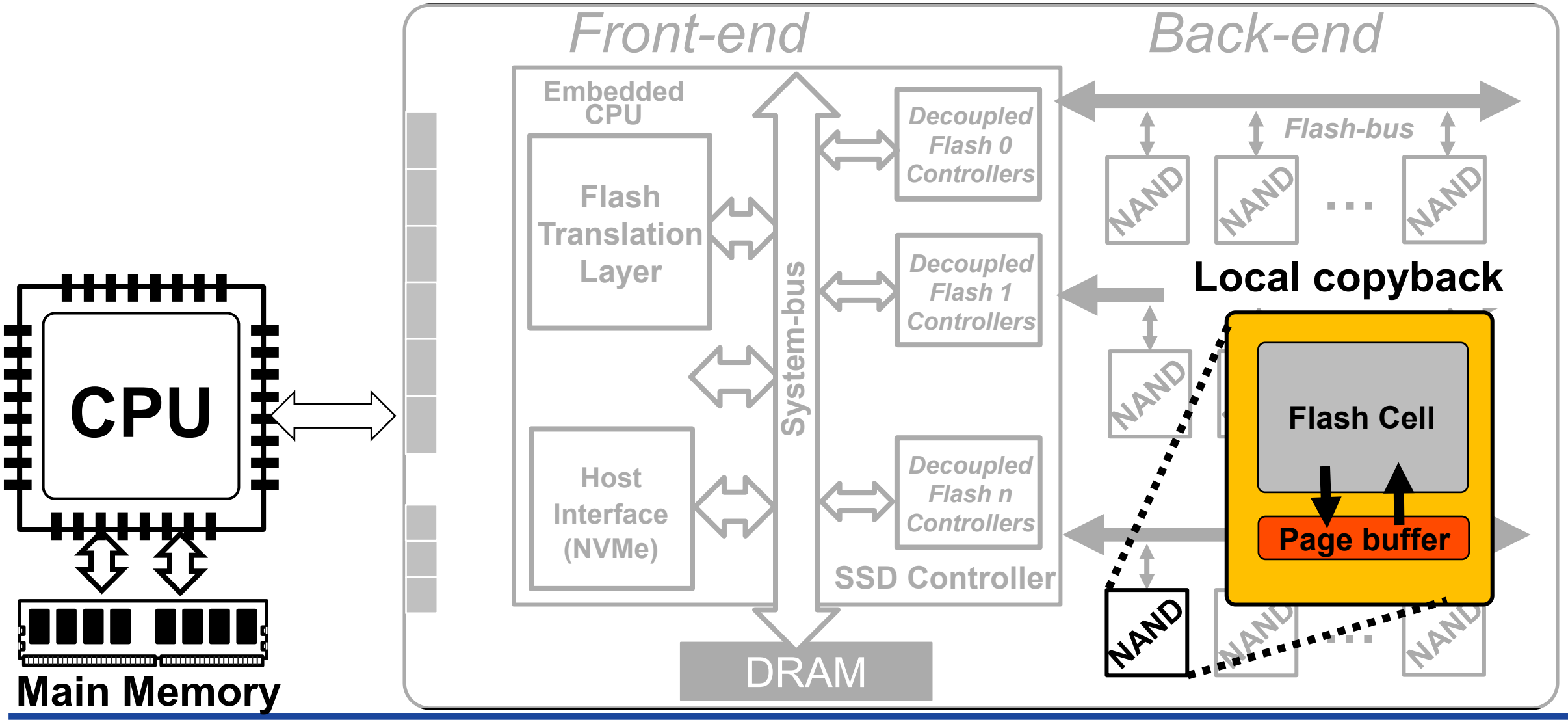
Decoupled flash controller



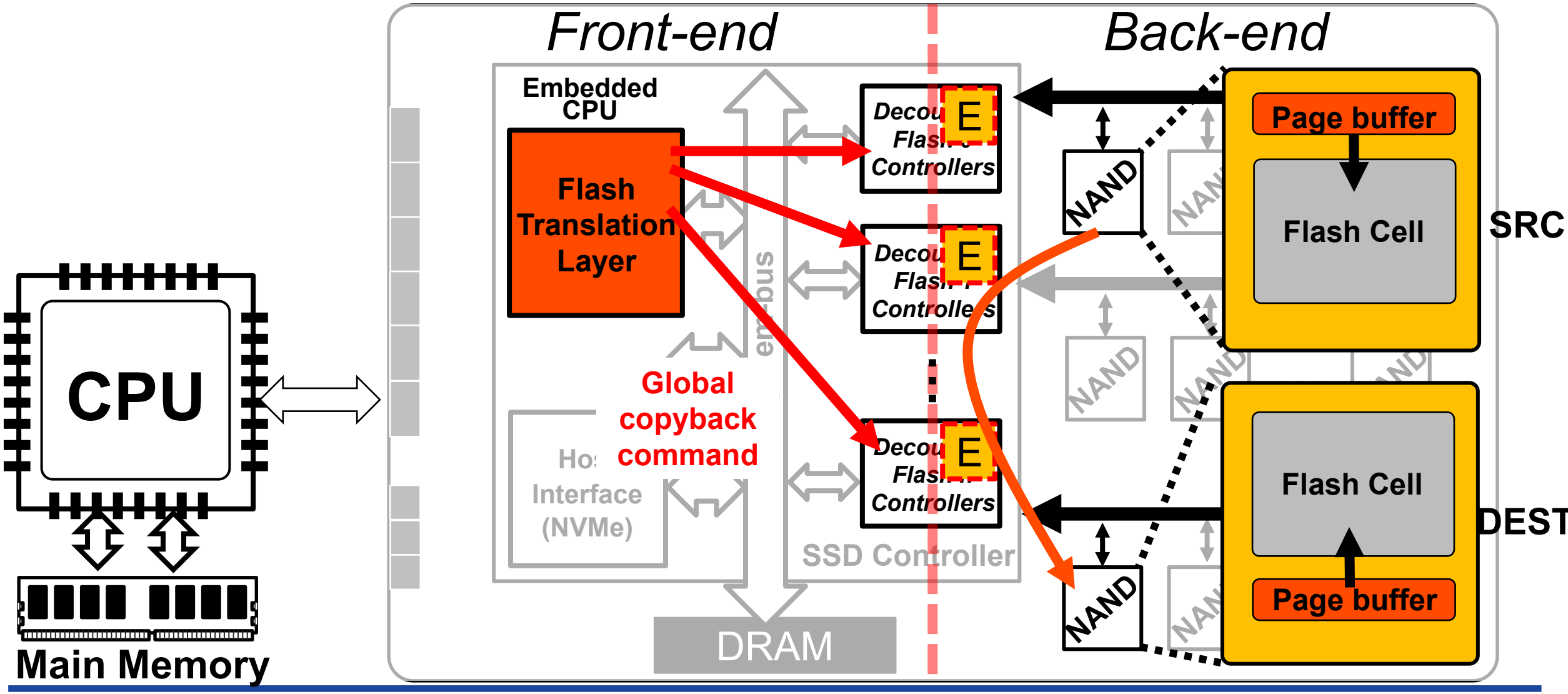
Integrated ECC engine into flash controllers



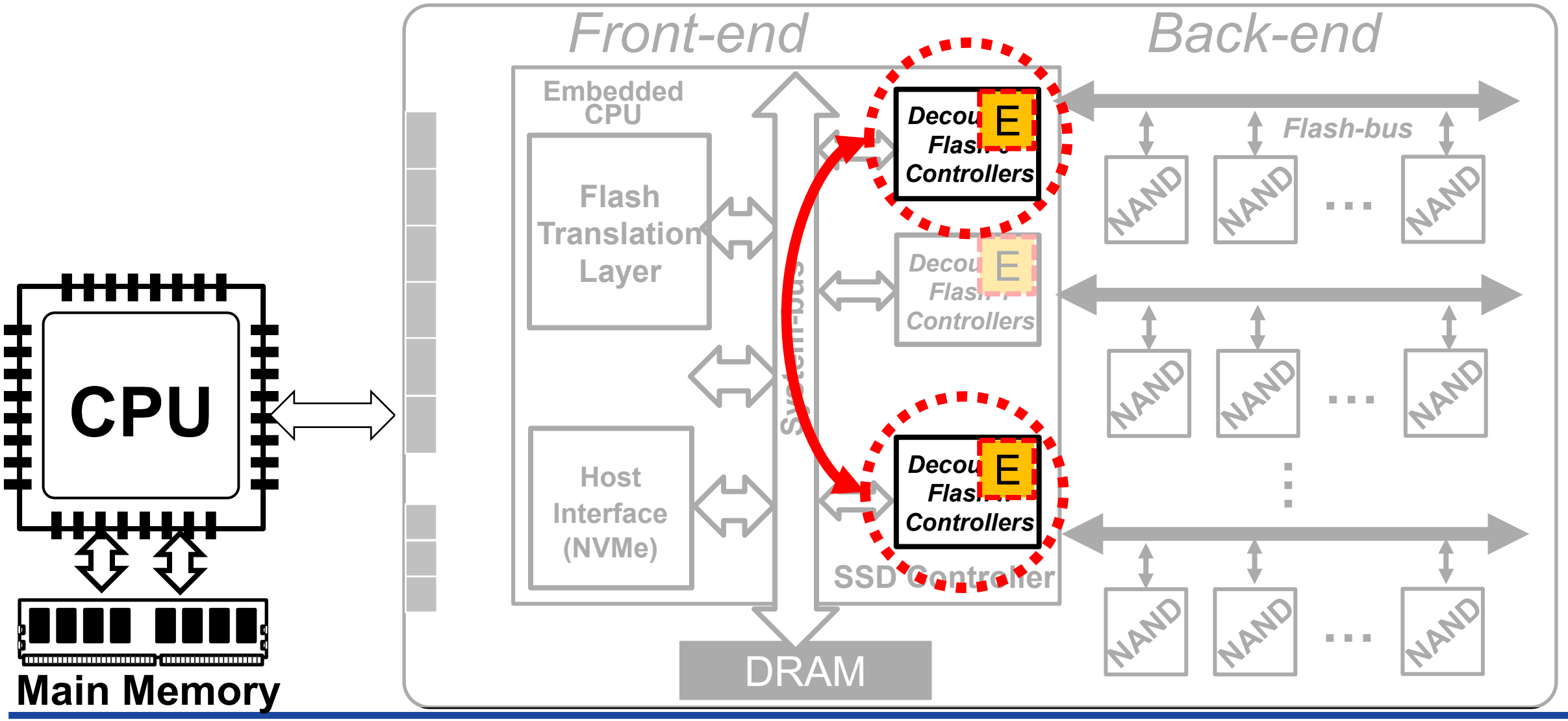
Extend ***local*** copy-back command …



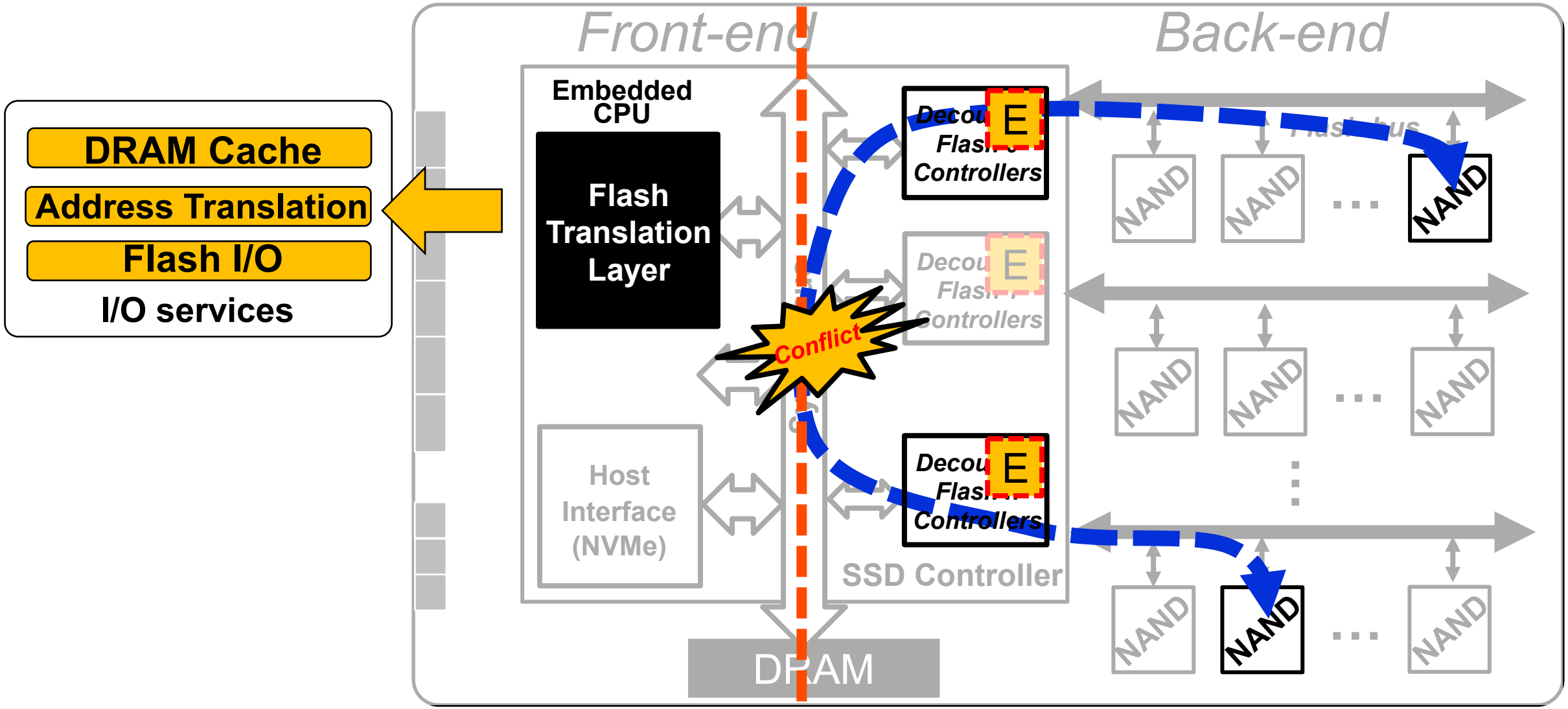
… to ***global*** copy-back command



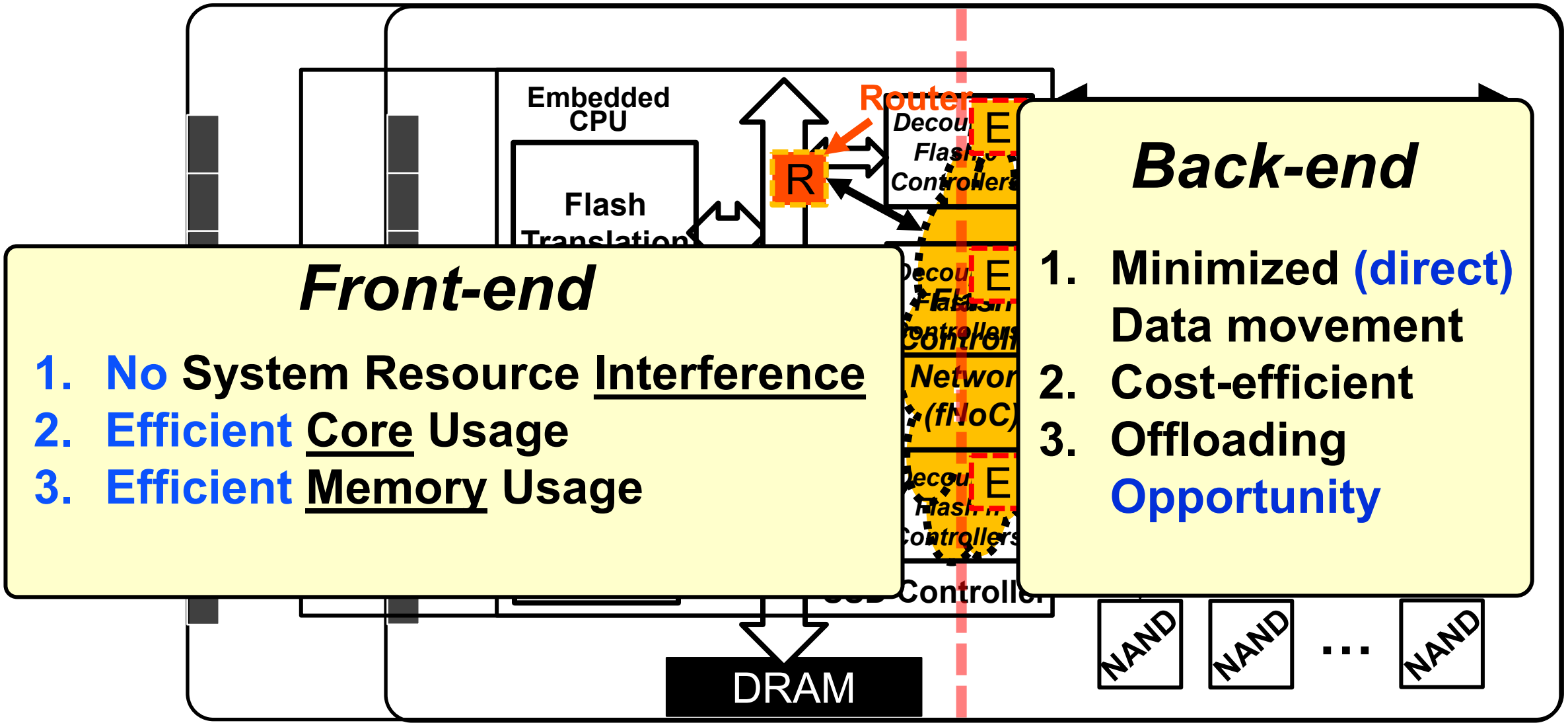
Enable direct **inter-flash controller** communication



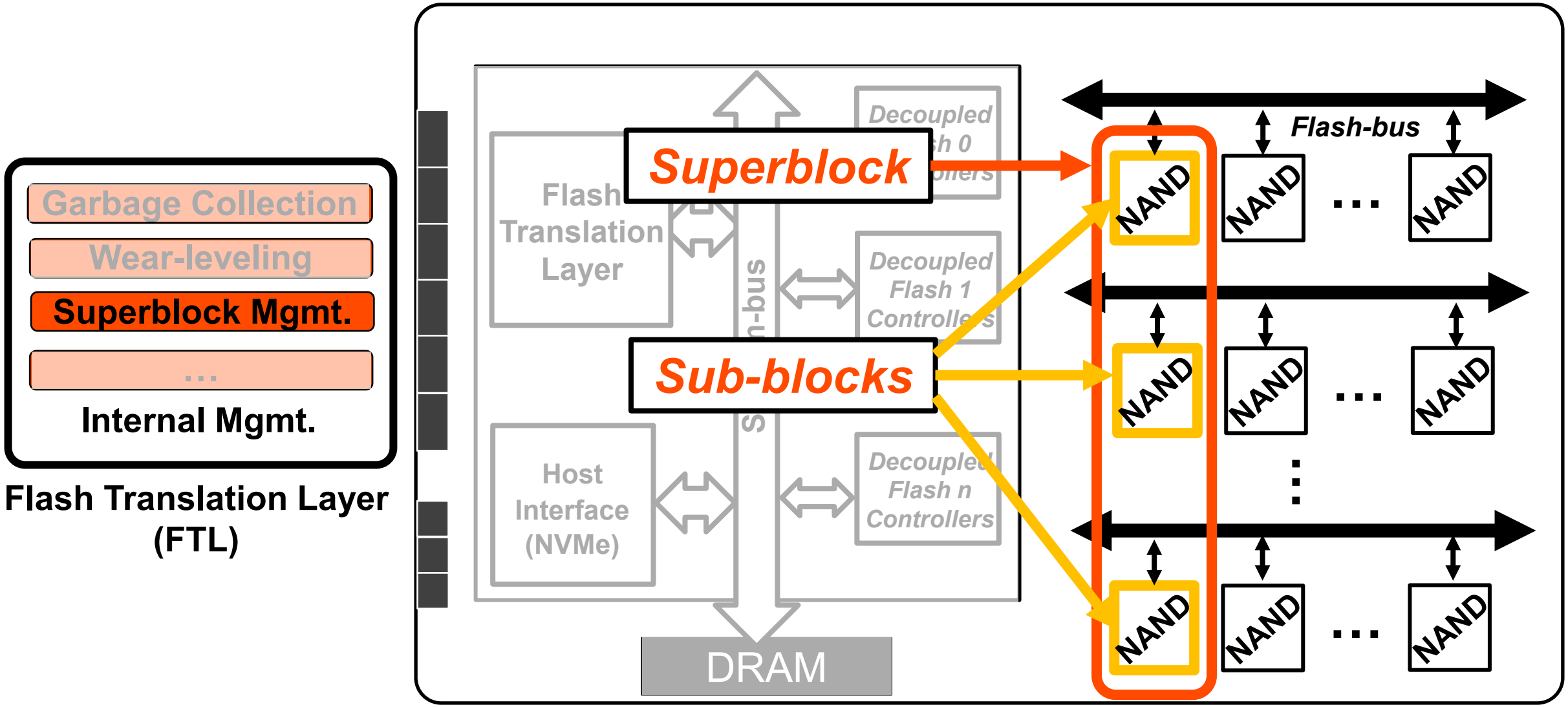
But dSSD still **shares** the system bus



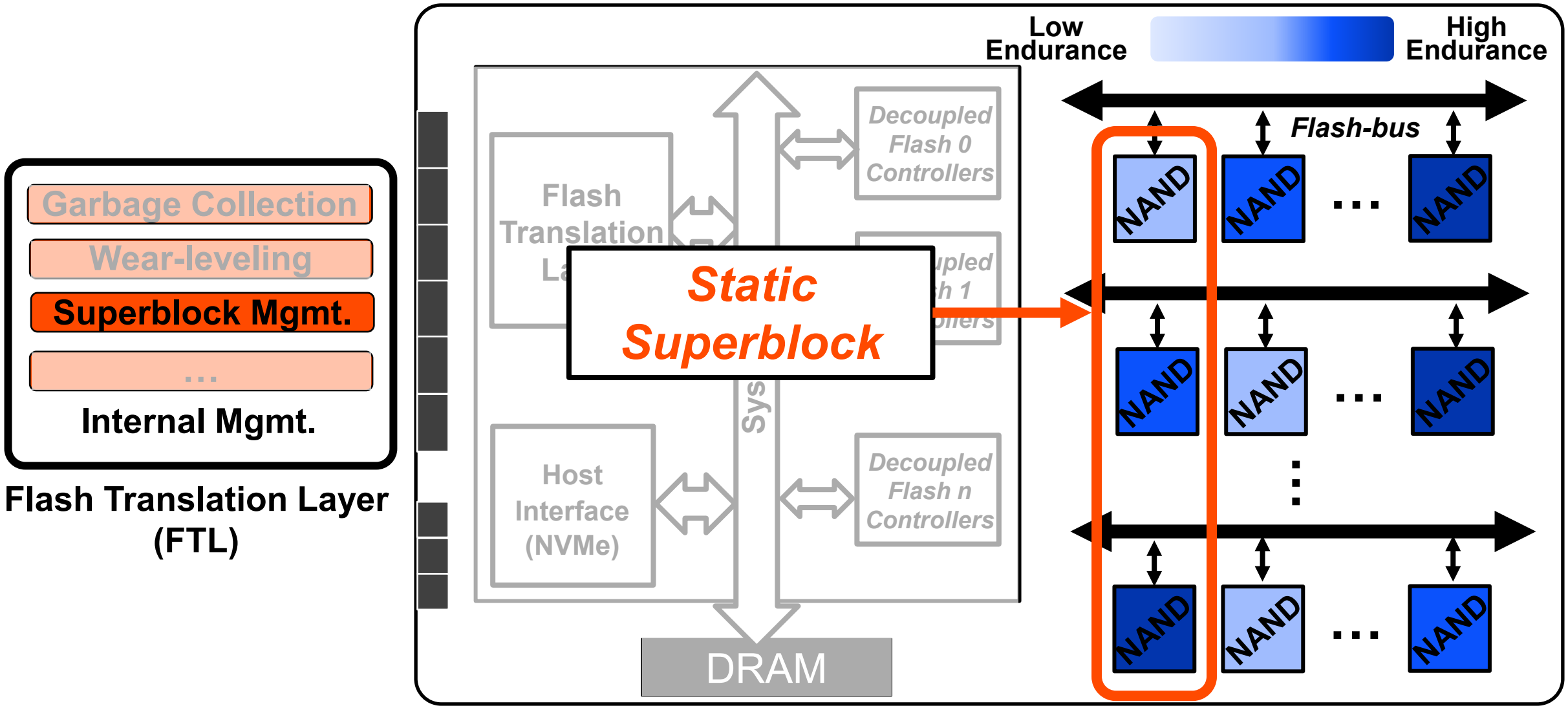
Decoupled SSD on flash network-on-chip (fNoC)



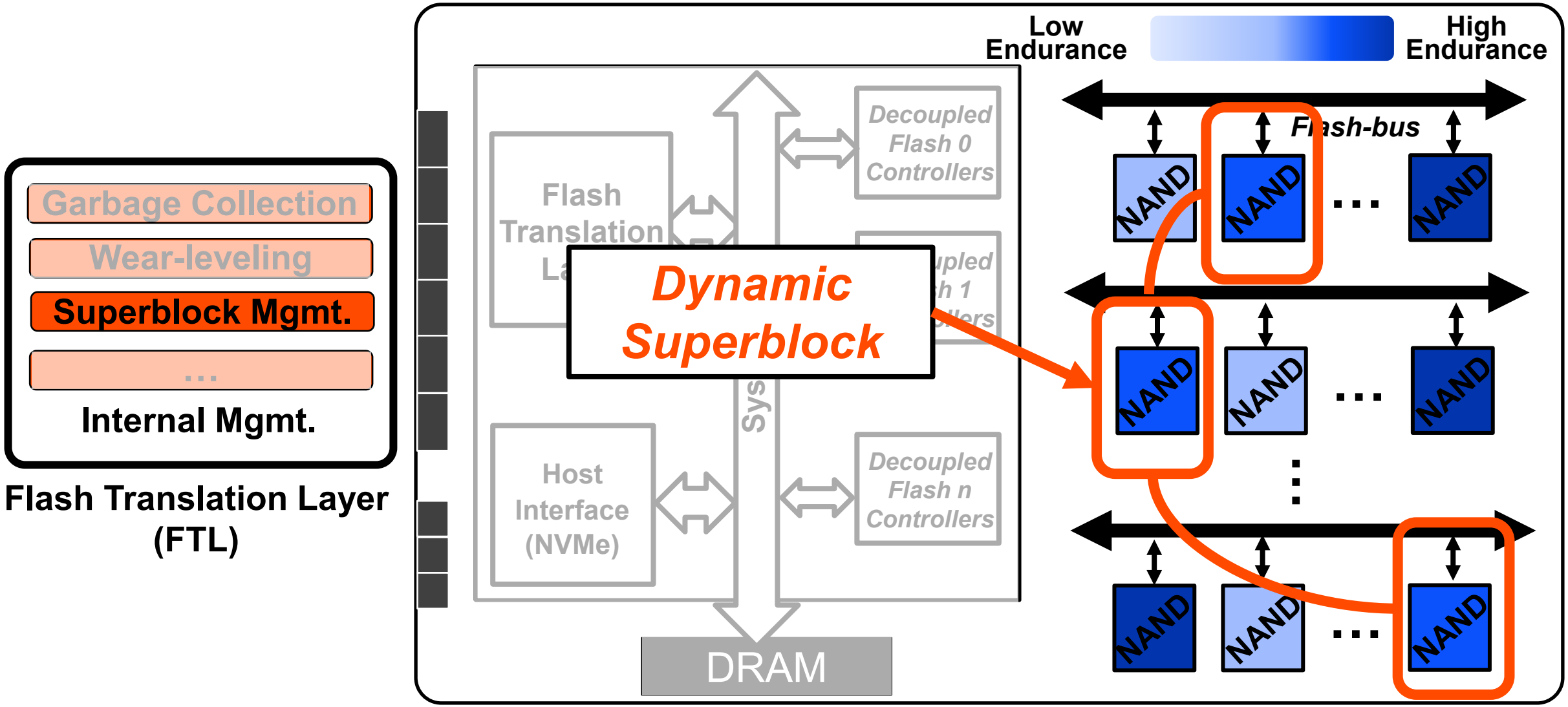
Superblock management in SSD



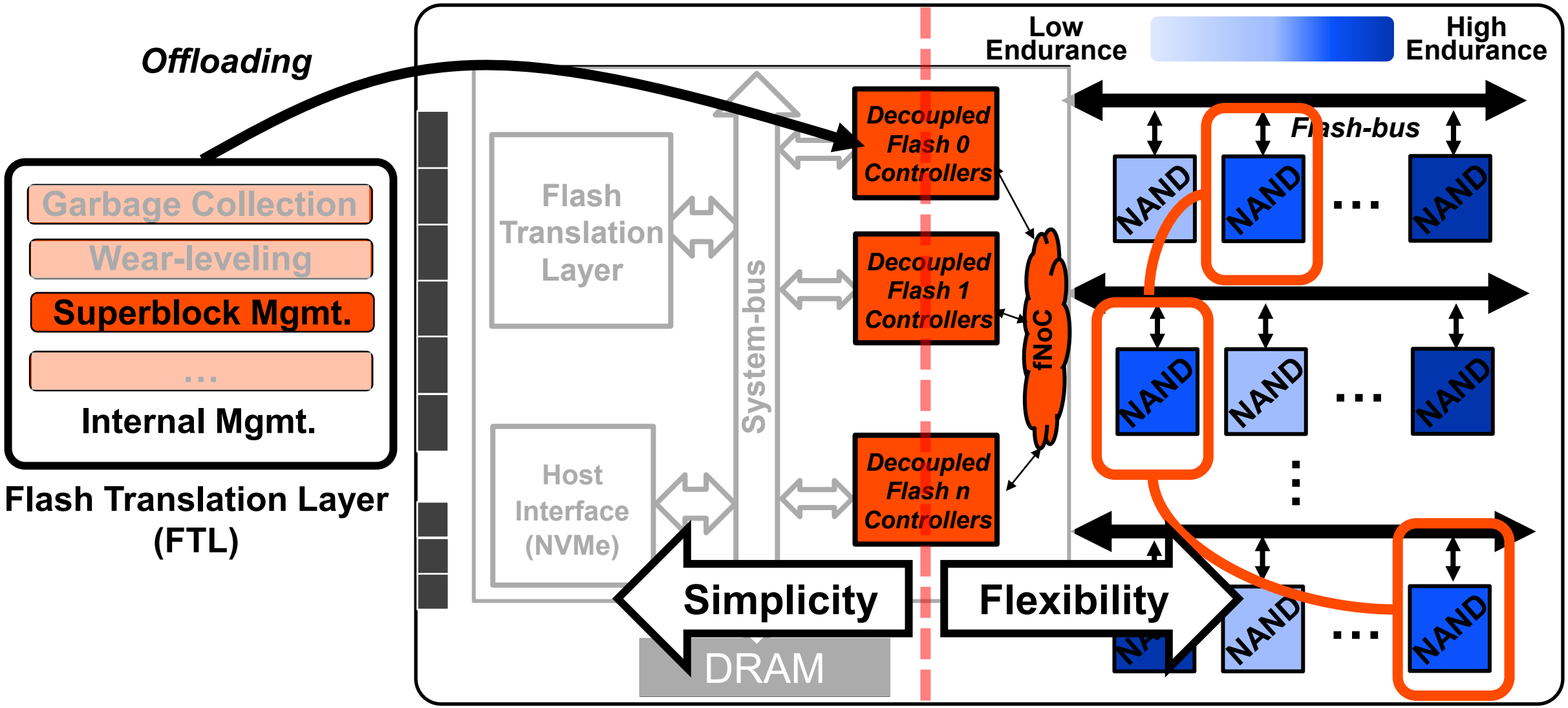
Static Superblock is not flexible



Dynamic Superblock is flexible, **but FTL is complex**



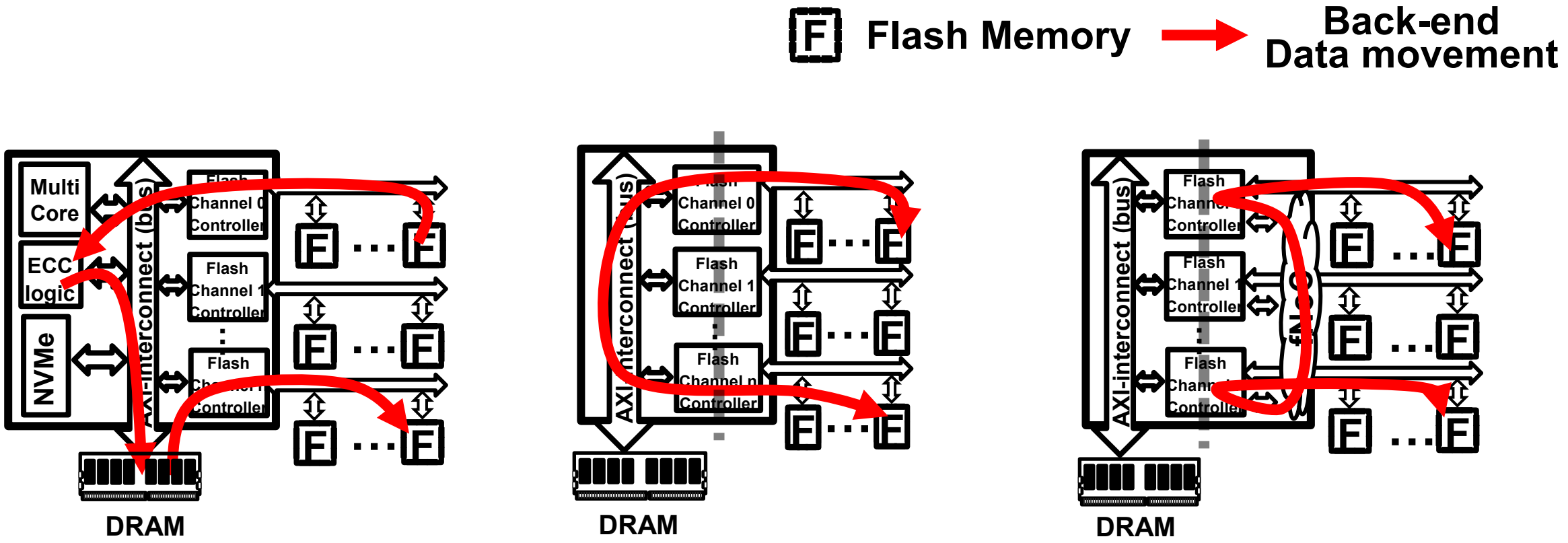
Flexible dynamic superblock with dSSD



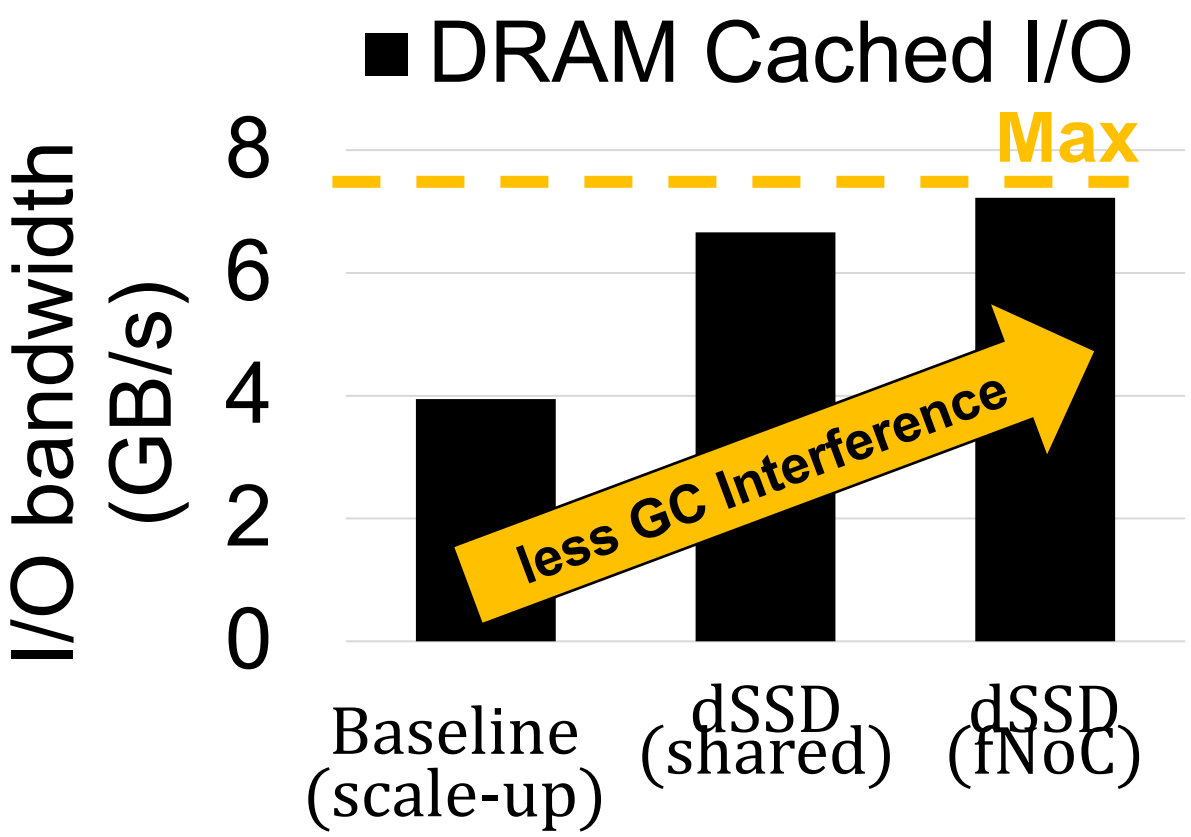
Simulation Setup (Methodology)

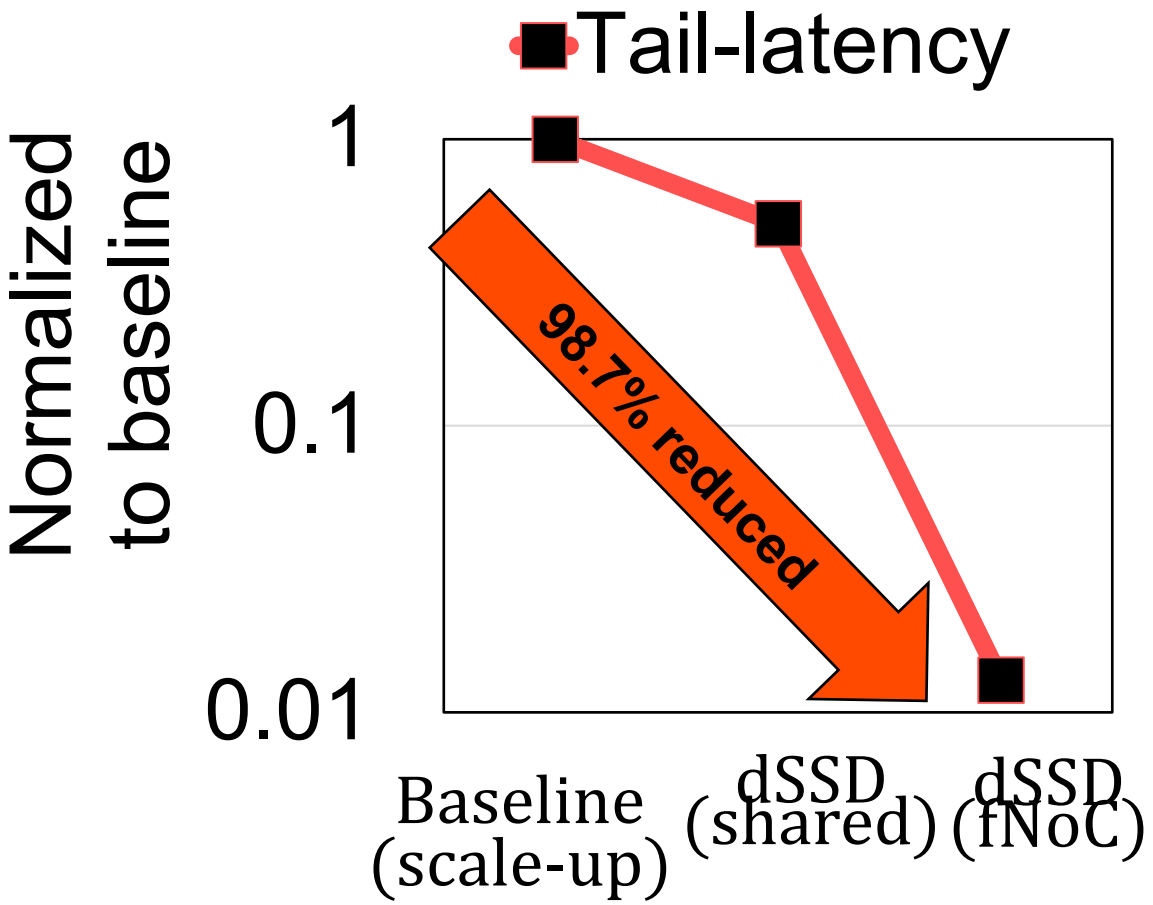
- **SimpleSSD-standalone 2.0 version**
  - PCI-E 3.0 x8 lanes, system-bus = **8GB/s** (x1)
  - DRAM = **8GB/s**, total flash bus BW= **8GB/s** (1000Mhz,8 bits)
  - Ultra-low-latency: read=5us, write=50us, erase=1ms, page size=4KB
  - TLC flash: read=60-95us, write=200-500us,erase=2ms, page size=16KB
  - Garbage collection: victim (greedy) / free (global) selection – parallel GC
- **BookSim (fNoC) : 1D mesh, k = 8, n = 1, routing = dim order,** $B_b = 2GB/s$
- **Workloads**:
  - Synthetic workloads, trace-driven evaluation
- **I/Os**
  - DRAM I/O and Flash I/O cases
  - GC interfered I/O cases for tail-latency

Architecture Comparison

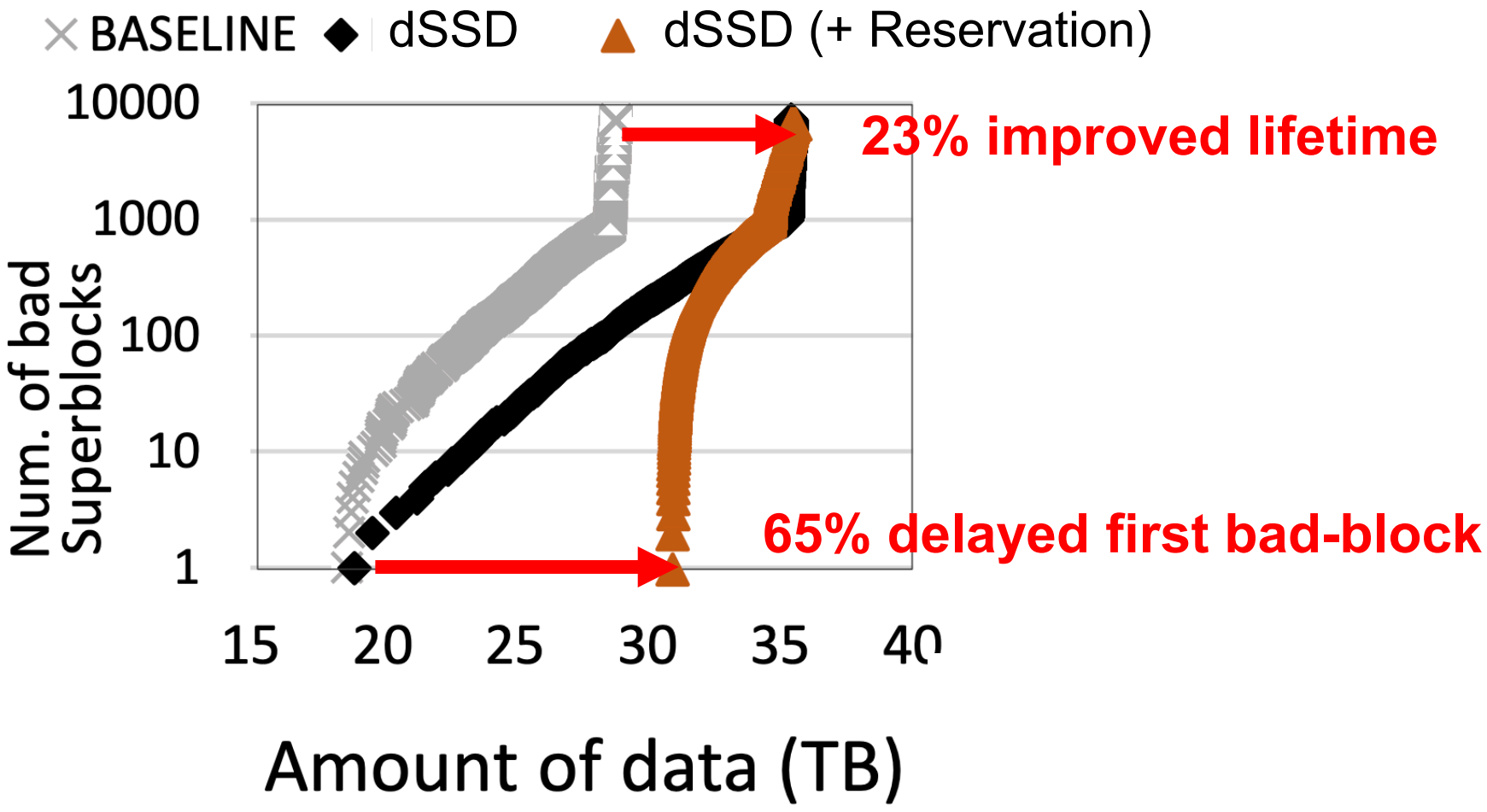


I/O Performance (with GC)





Extension of SSD Lifetime



Summary

- Route Packets, not wires ➔ Send packets, not signals
- SSD can take advantage of interconnection network to improve performance (tail latency) and reliability.
- Network SSD: ***Packetized interface*** within SSD to efficiently utilize the given flash bus bandwidth.
- Decoupled SSD: Effectively ***decouple*** the front-end and back-end of an SSD through the flash channel controllers