BOSTON UNIVERSITY

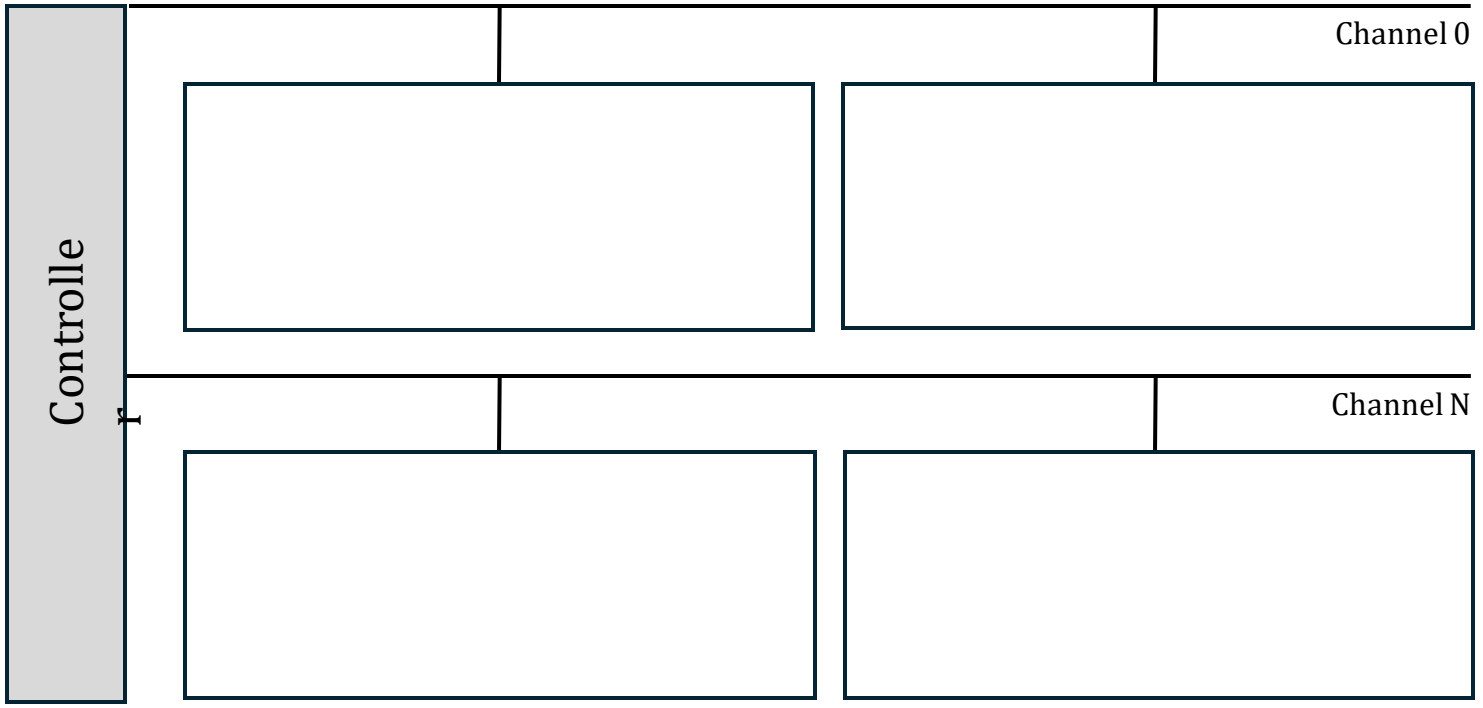
SSD Design Elements

CS 561: Data Systems Architecture

Teona Bagashvili

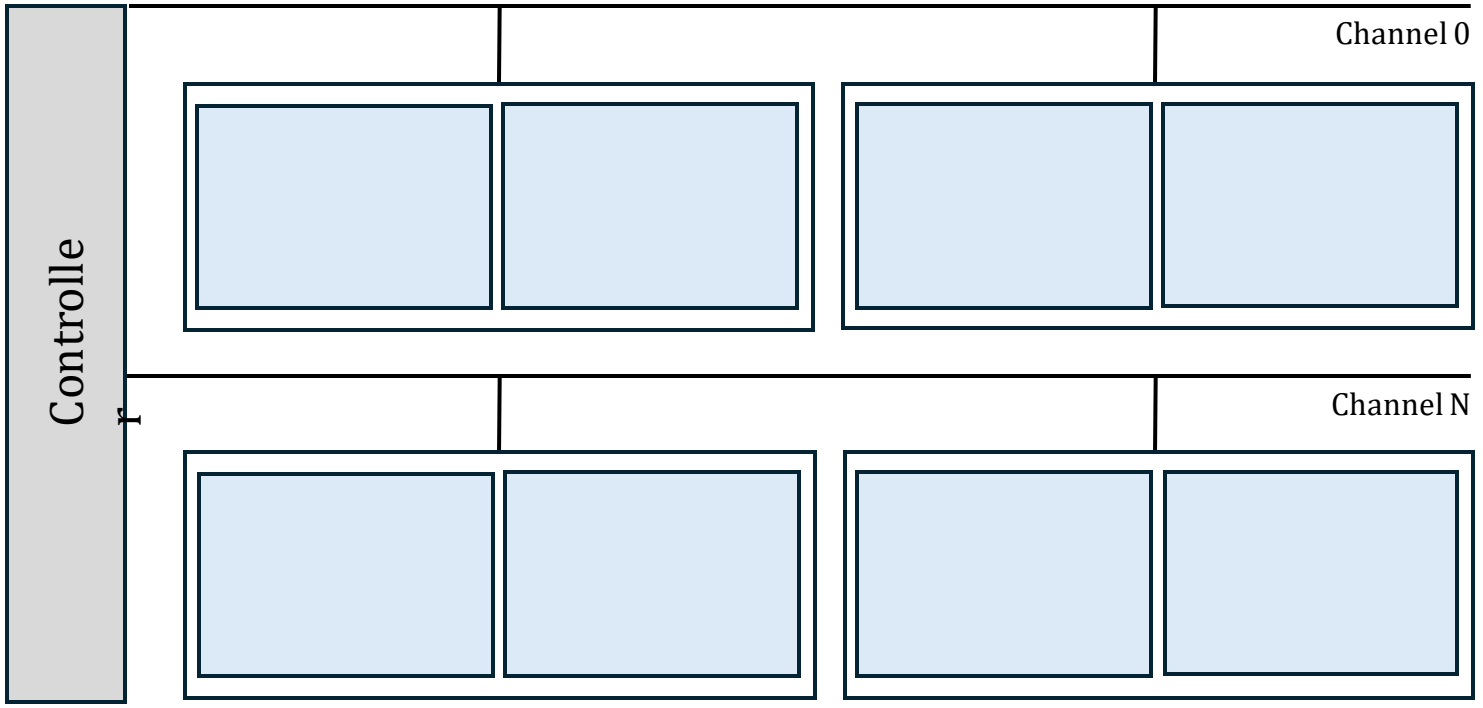
DiSC lab

SSD Architecture



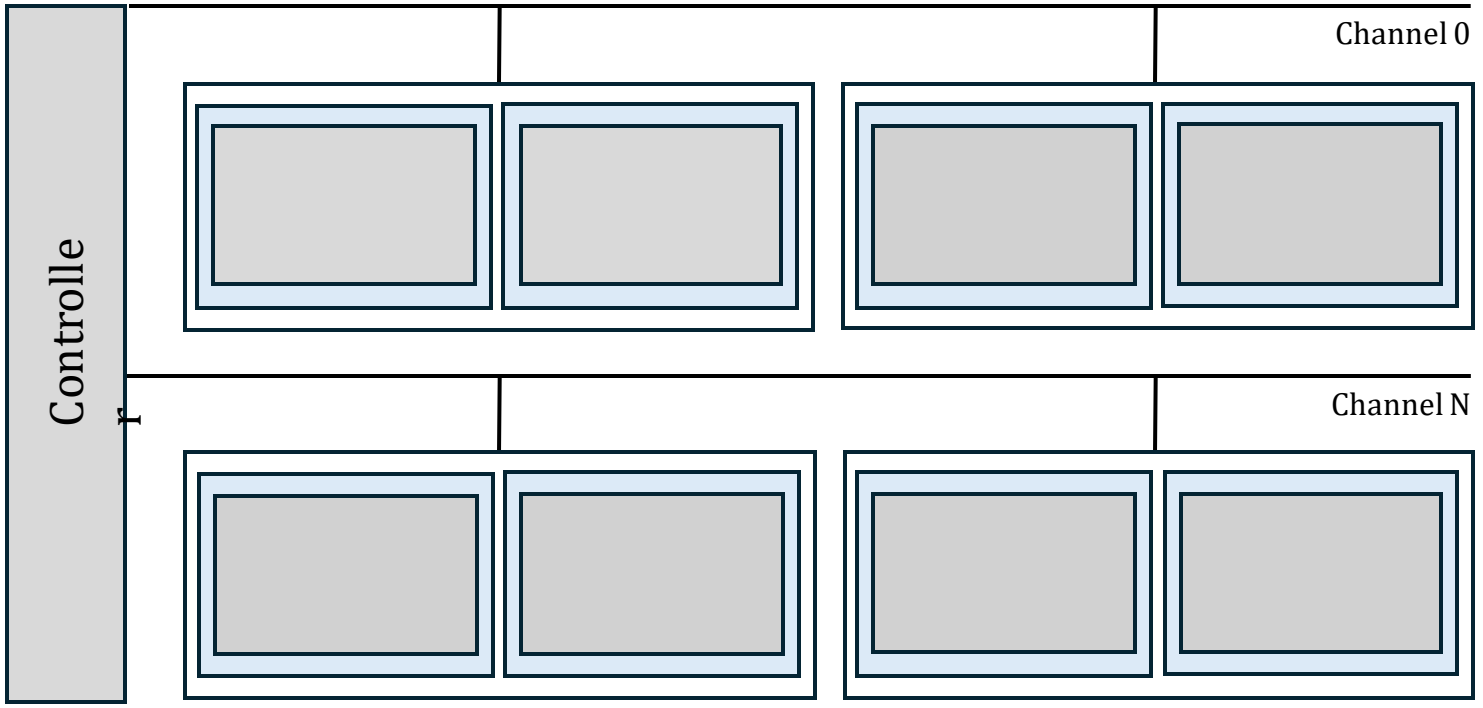
Flash Package -> Chip

SSD Architecture



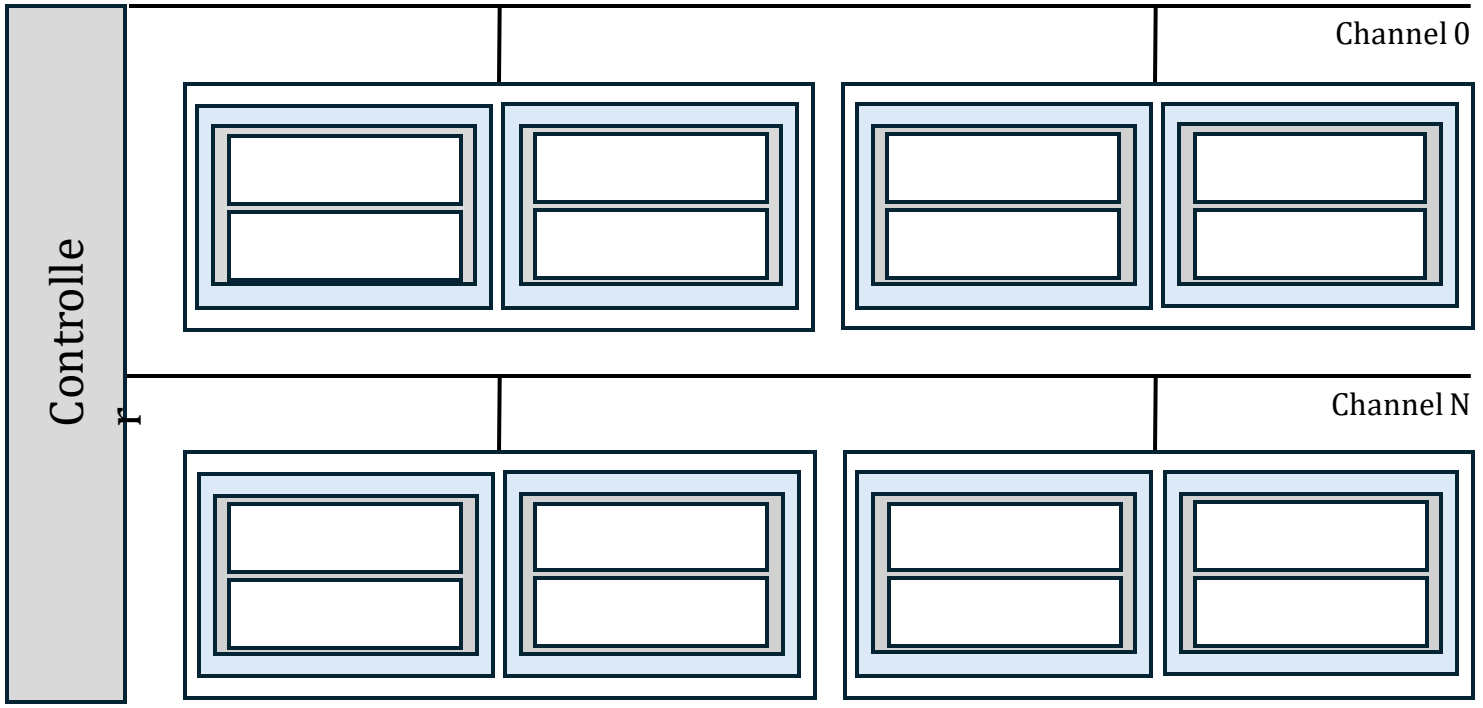
Flash Package -> Chip -> Die

SSD Architecture



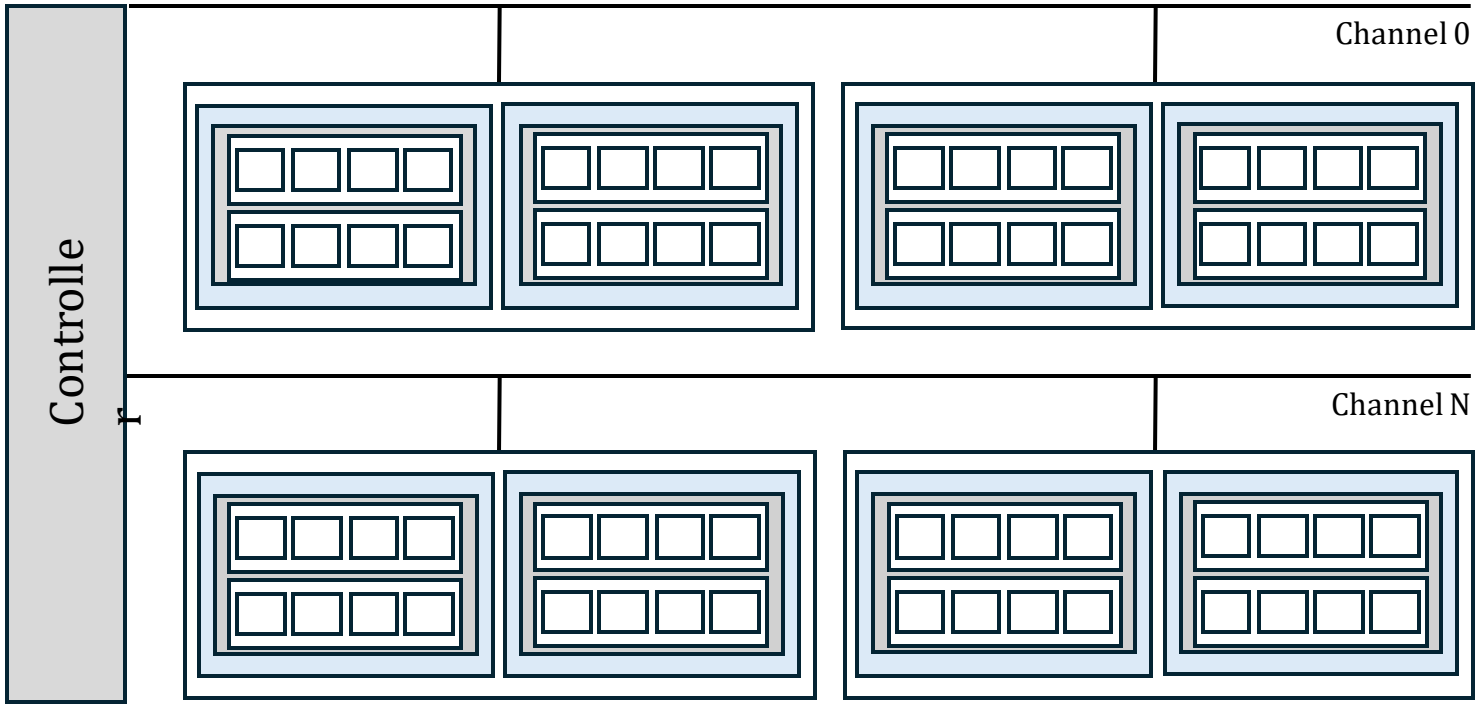
Flash Package -> Chip -> Die -> Plane

SSD Architecture



Flash Package -> Chip -> Die -> Plane -> Erase Block

SSD Architecture



Flash Package -> Chip -> Die -> Plane -> Erase Block -> Page -> Nand Cells

SSD Properties

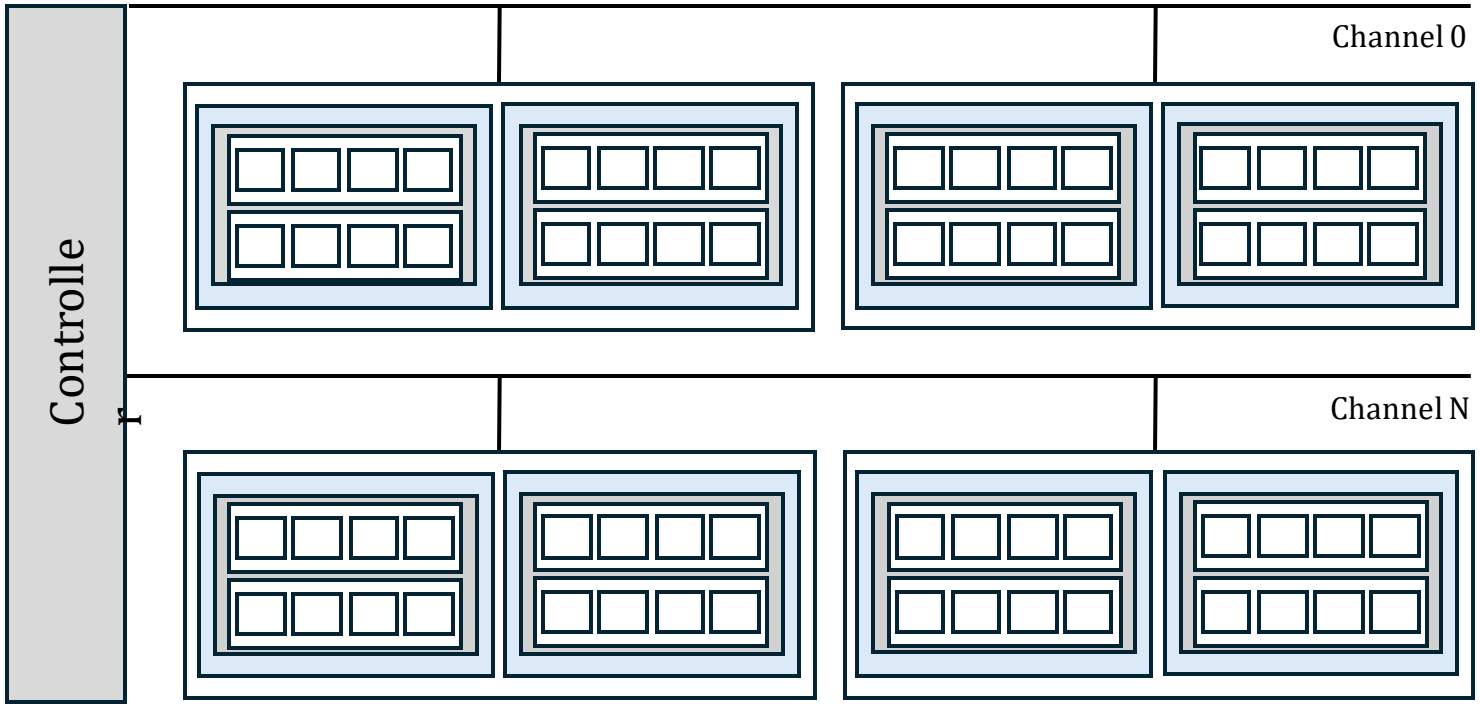
Reads at page level

Writes at page level

Out of place updates

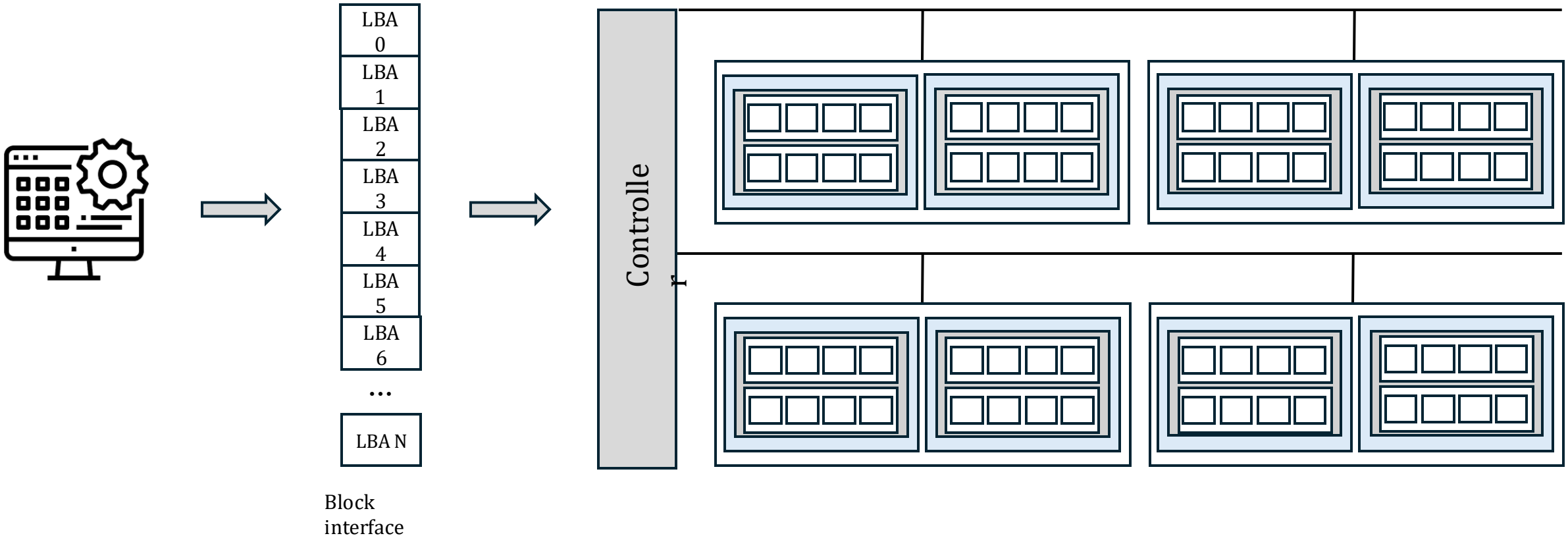
Erasure at block level

Wear leveling

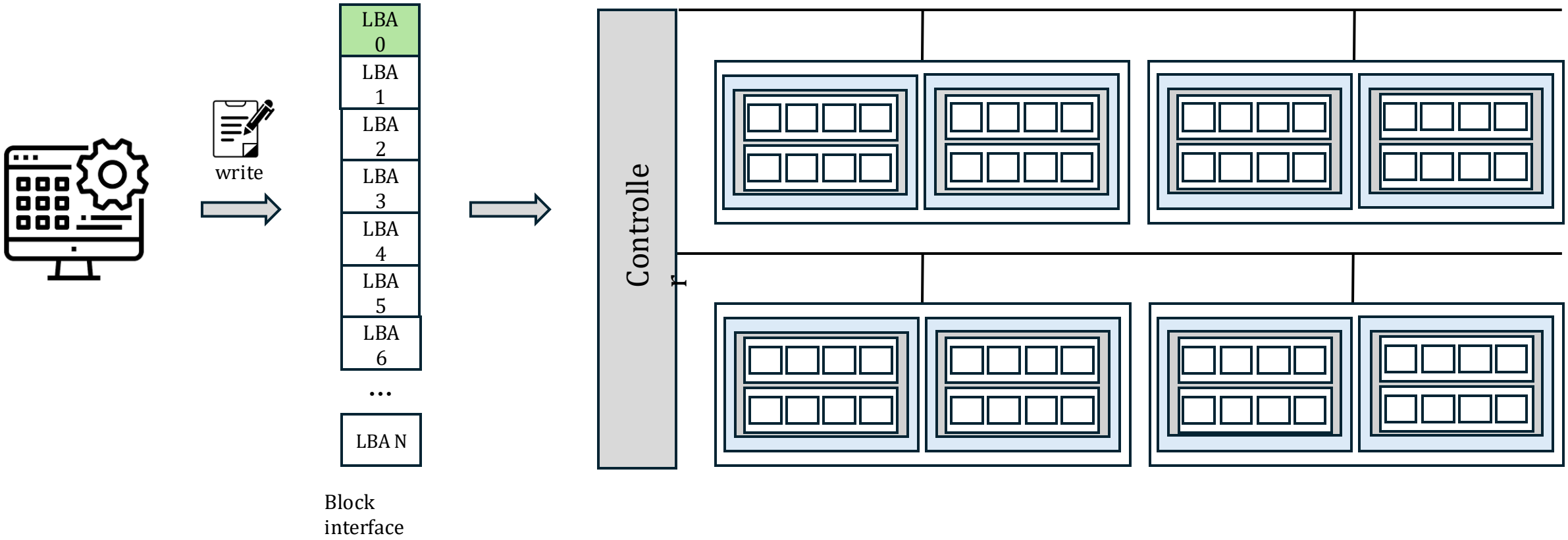


Flash Package -> Chip -> Die -> Plane -> Erase Block -> Page -> Nand Cells

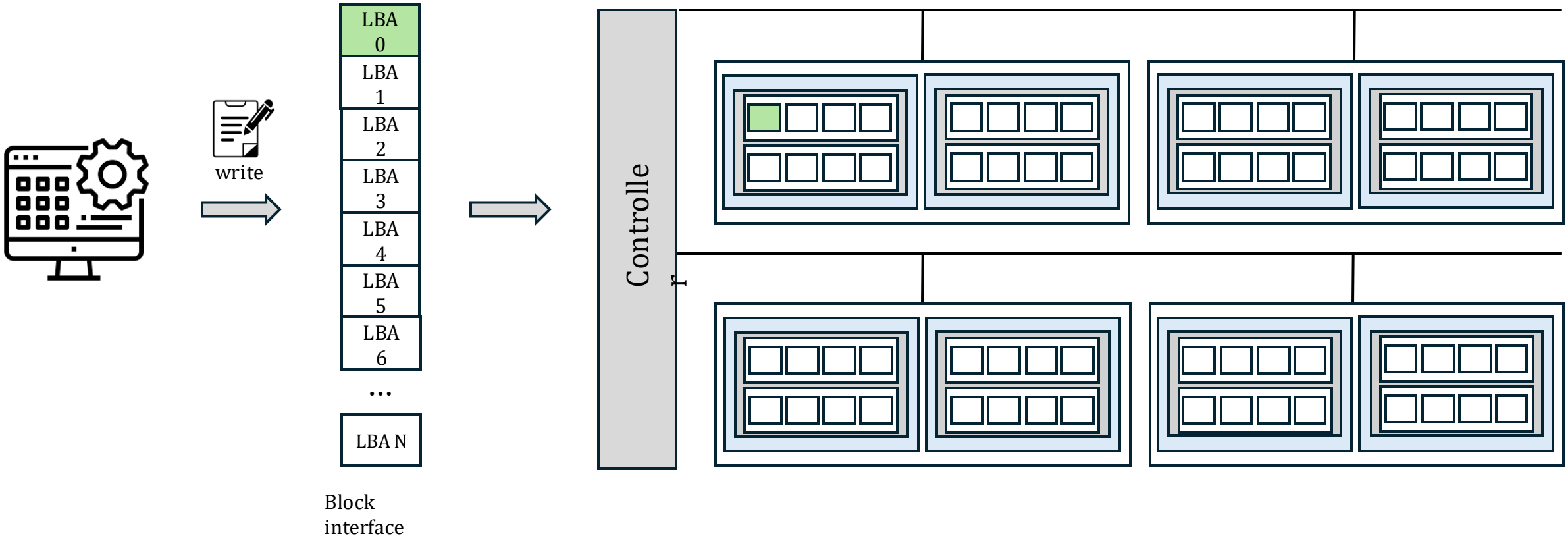
Black-Box SSD



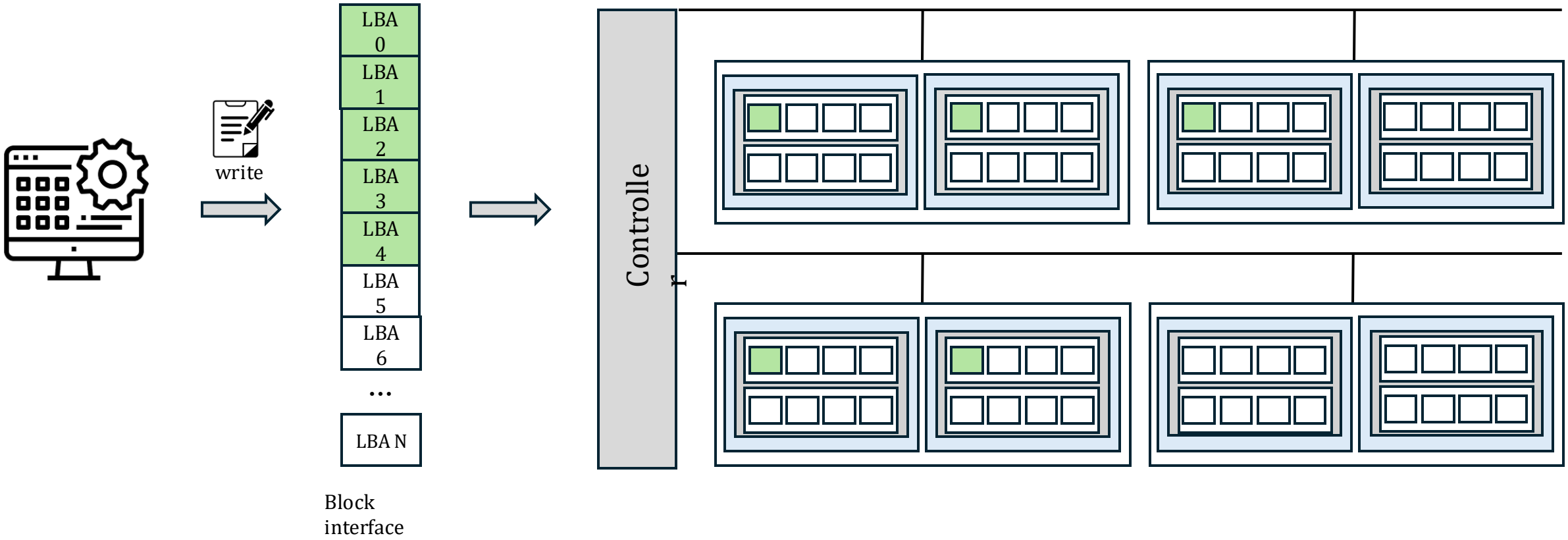
Black-Box SSD



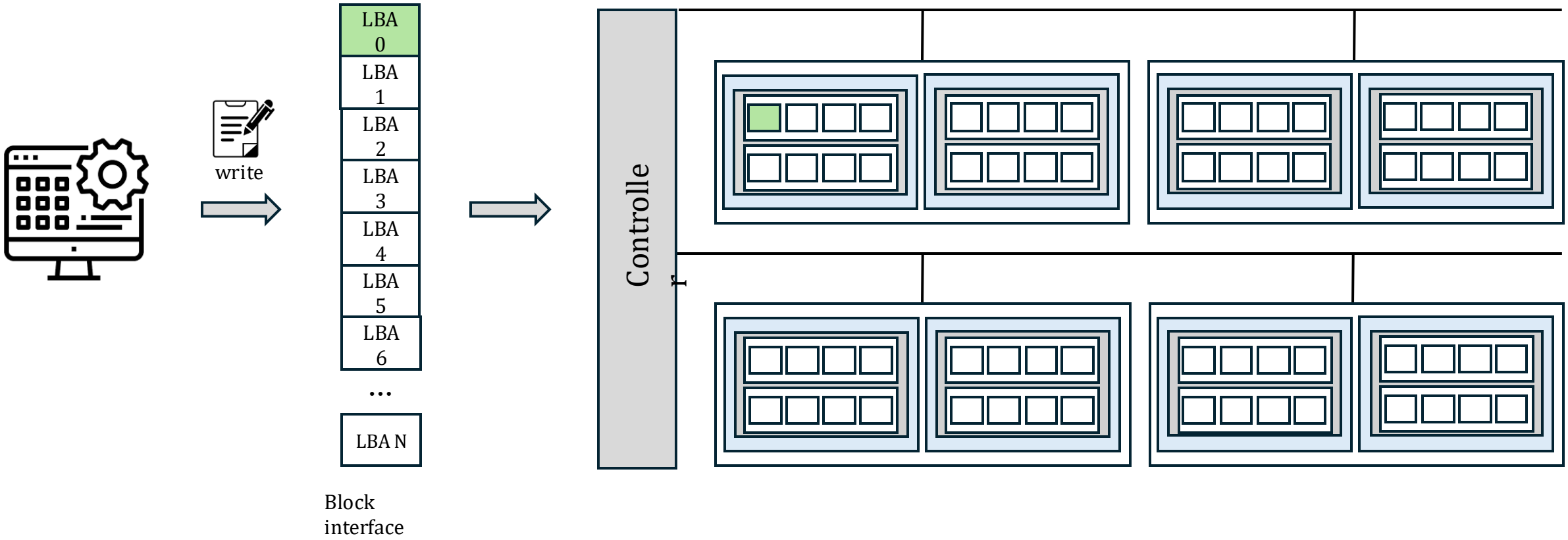
Black-Box SSD: LBA -> PBA



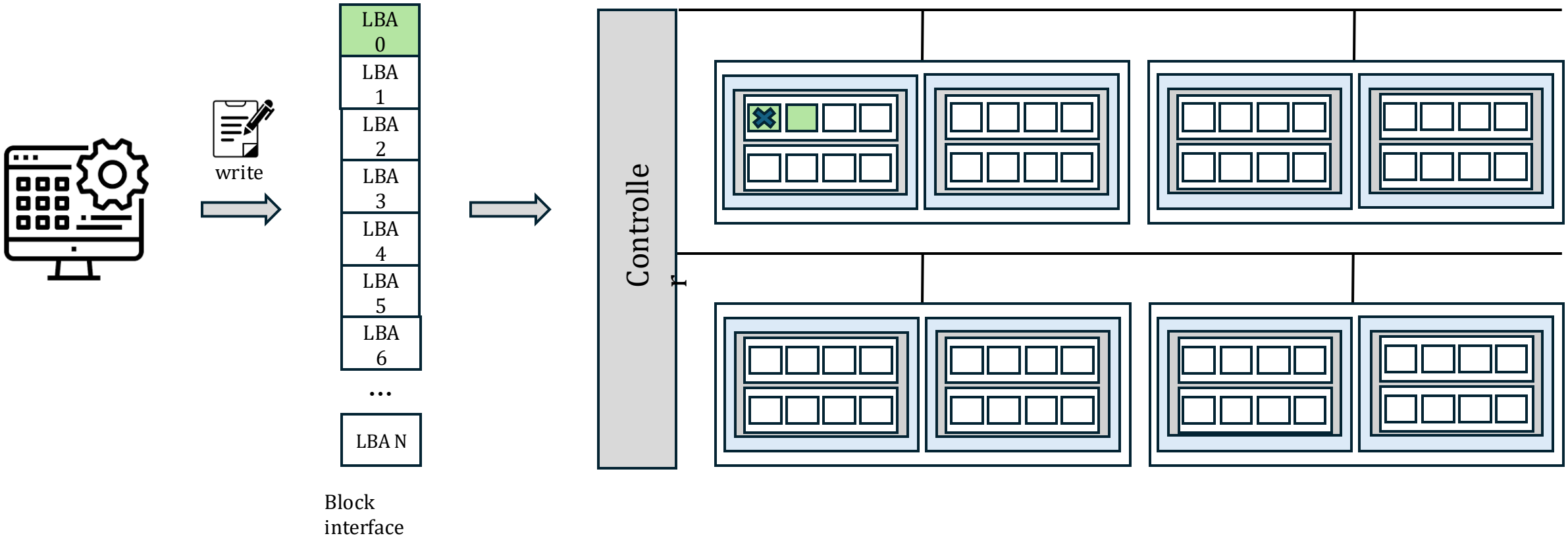
Black-Box SSD: Striping



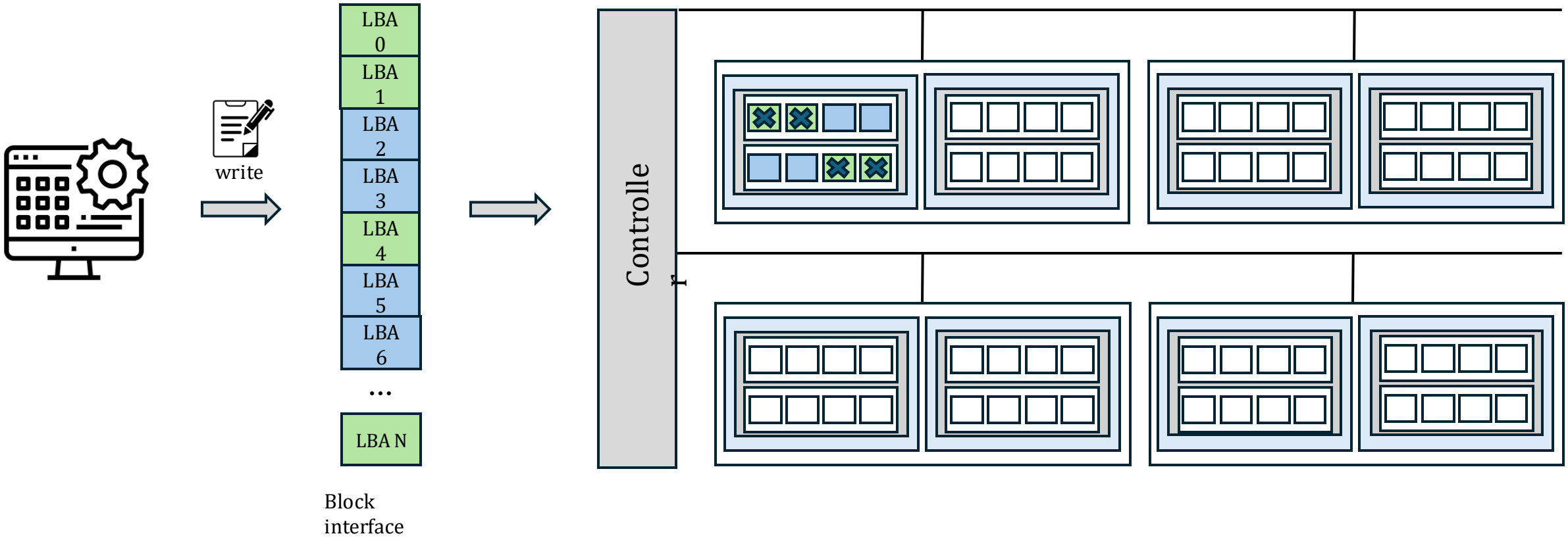
Black-Box SSD: Out-of-place updates



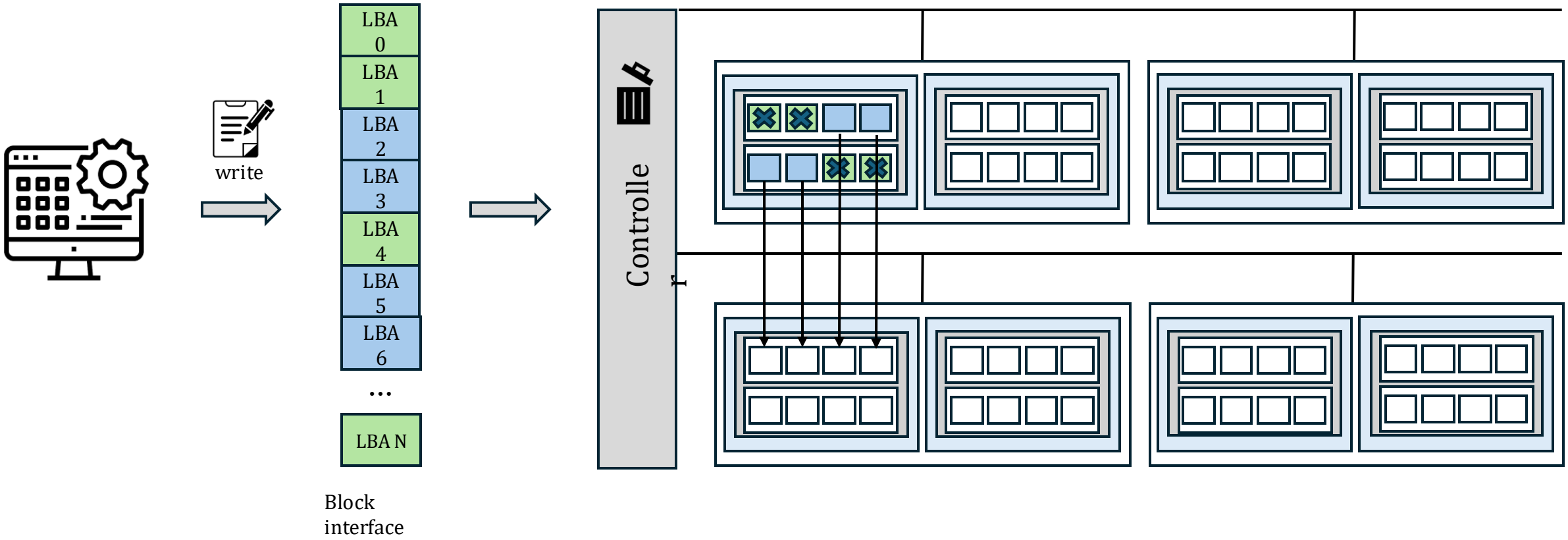
Black-Box SSD: Out-of-place updates



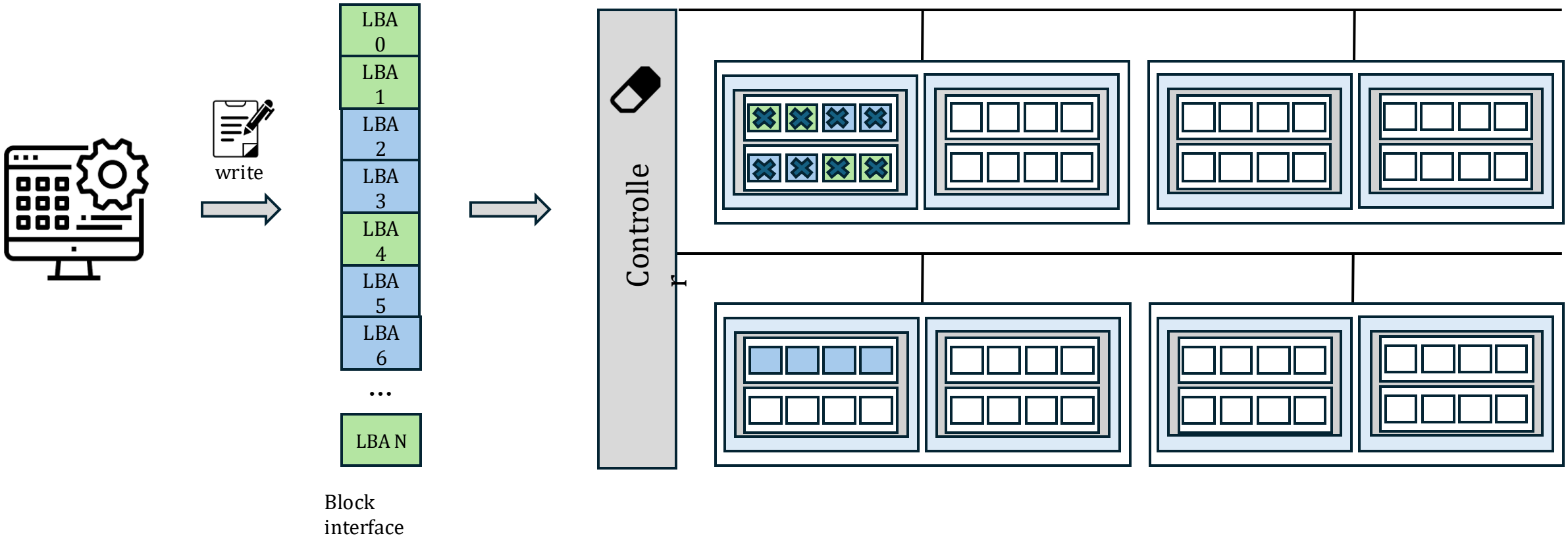
Black-Box SSD: Garbage Collection



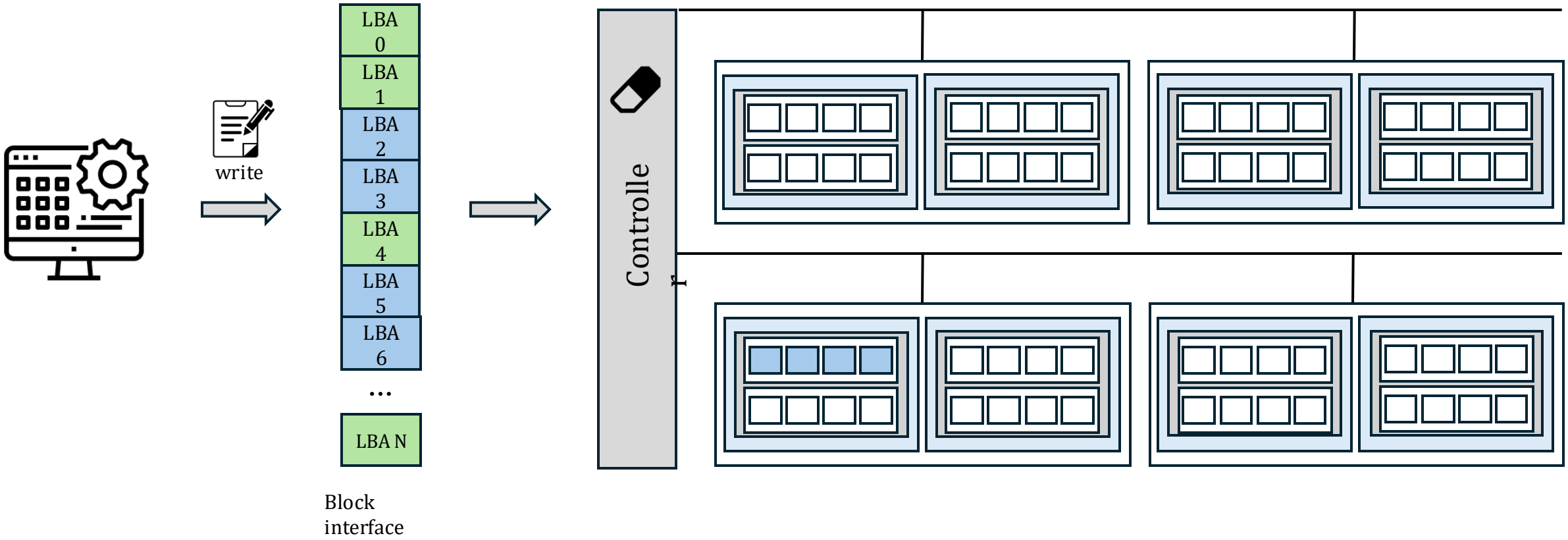
Black-Box SSD: Garbage Collection



Black-Box SSD: Garbage Collection

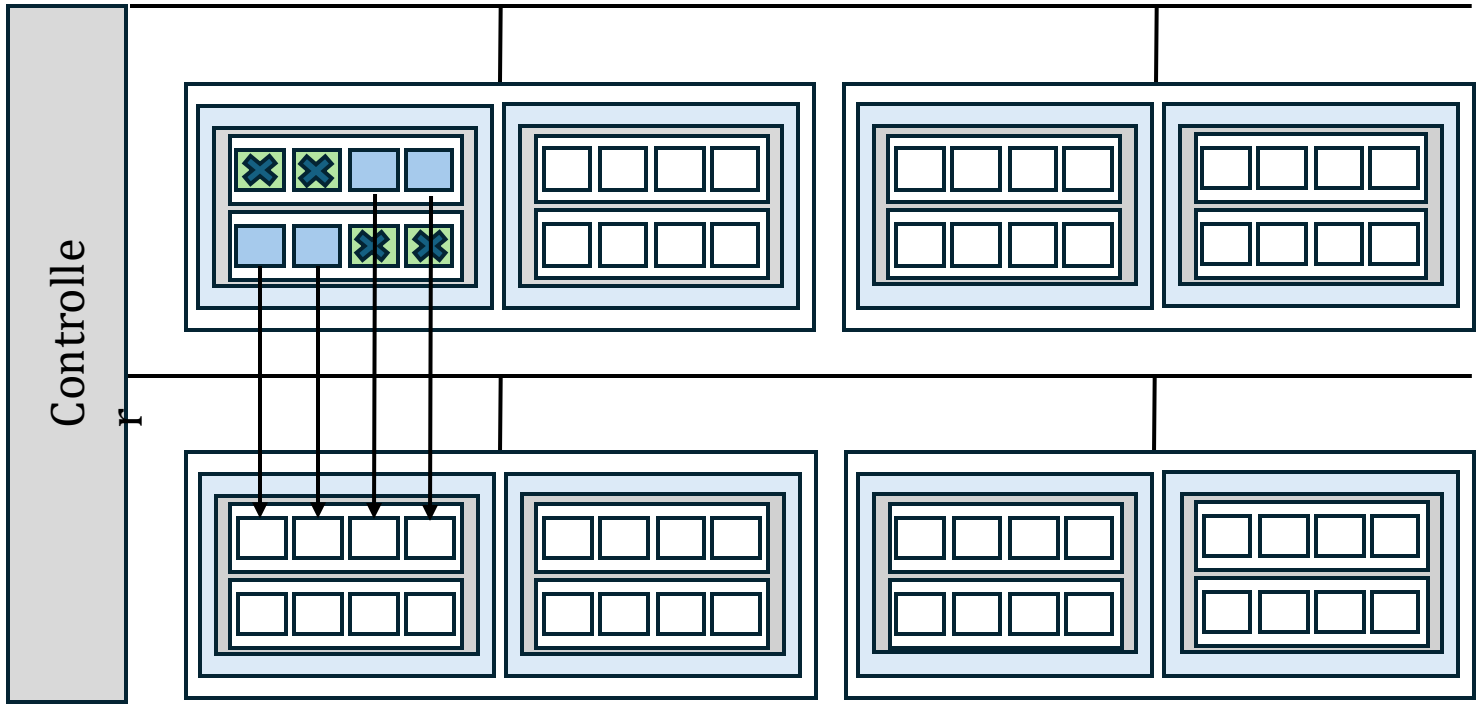


Black-Box SSD: Garbage Collection



Black-Box SSD: Block Interface Tax

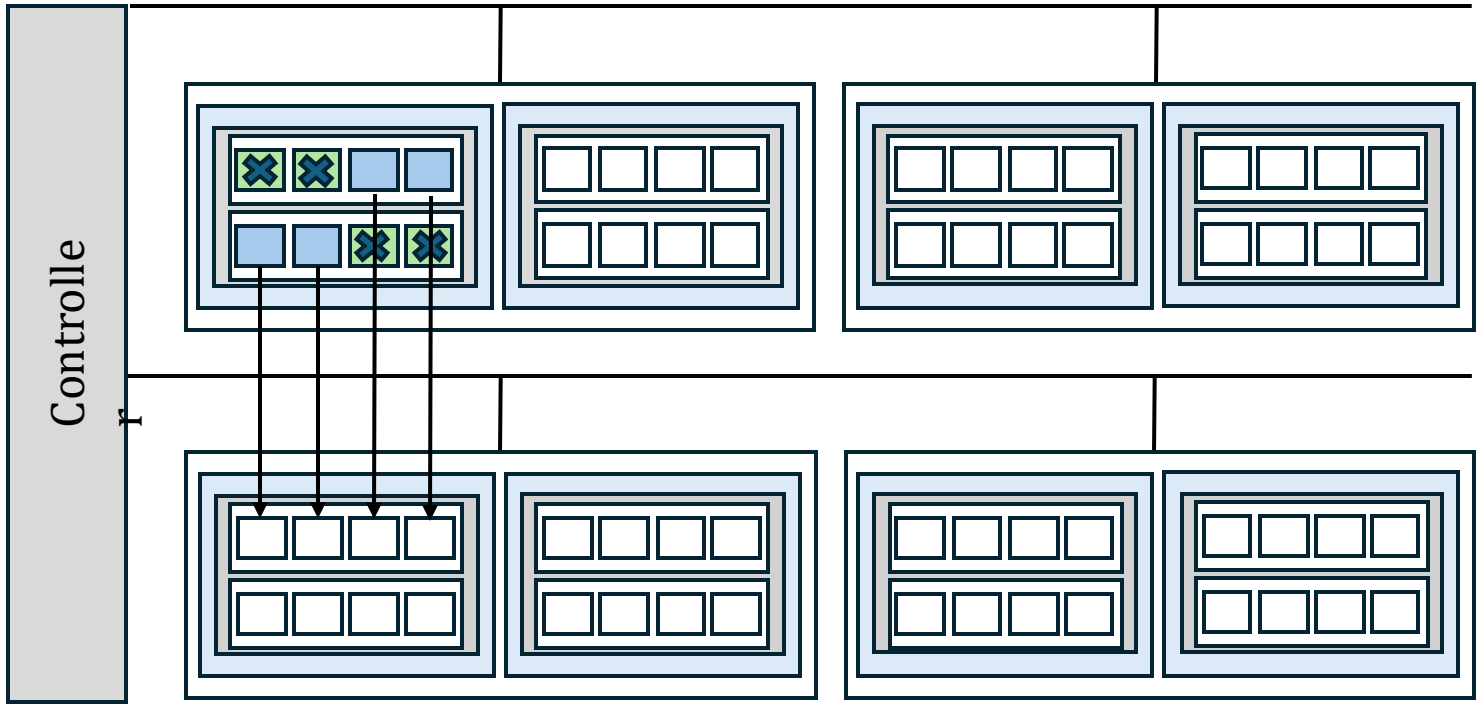
Issues with garbage collection?



Black-Box SSD: Block Interface Tax

Issues with garbage collection?

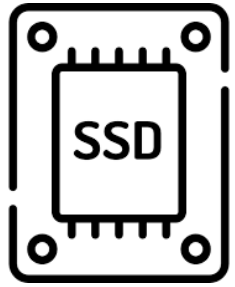
- ✗ Write Amplification
- ✗ Interference with host I/O
- ✗ Increased wear



Block Interface Tax

Random I/O

Treat SSD as black-box



Handle out-of-place updates

Garbage collection to reclaim space

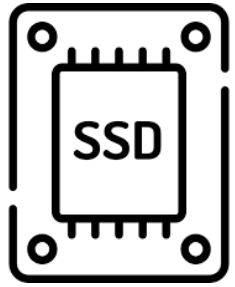
Wear leveling

Optimal data placement

Avoiding Block Interface Tax

Random I/O

Treat SSD as black-box



Handle out-of-place updates

Garbage collection to reclaim space

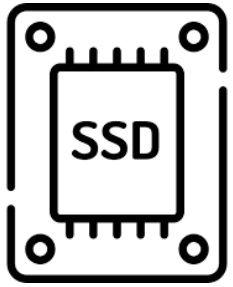
Wear leveling

Optimal data placement

Avoiding Block Interface Tax

Append only interface

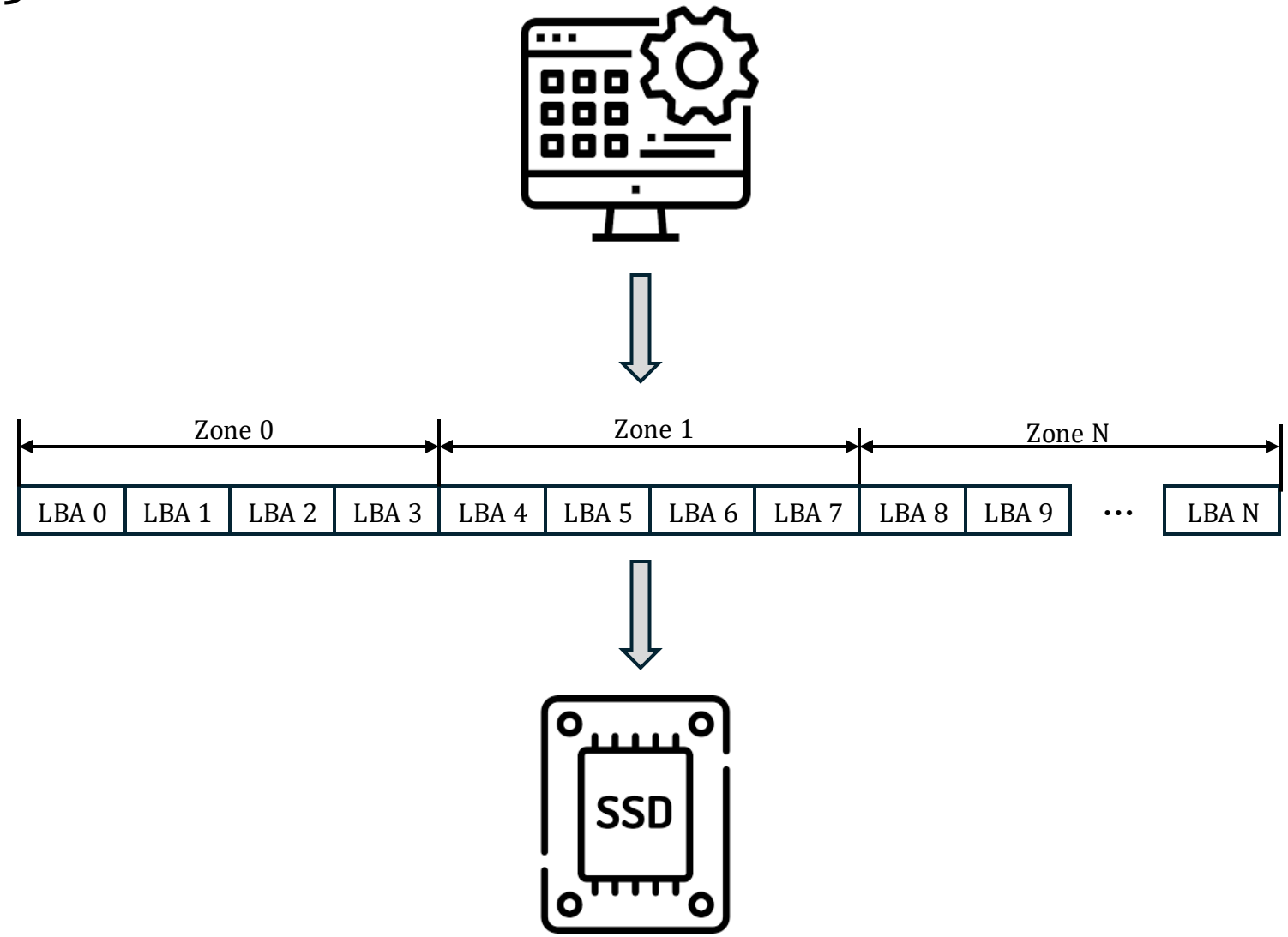
Garbage collection to reclaim space



Wear leveling

Optimal data placement

Zoned Namespace Interface (ZNS)

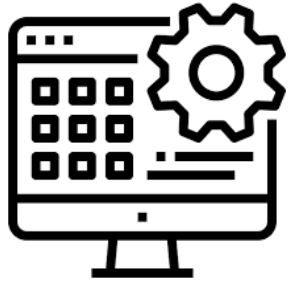


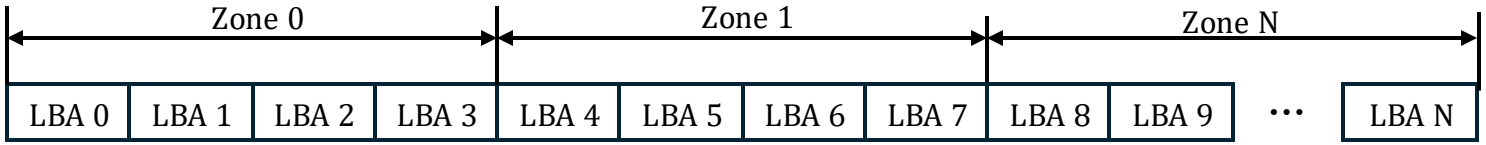
ZNS Properties

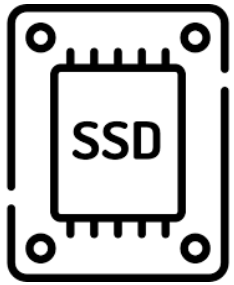
Append-only zones

Reads in any order

Manage zone states

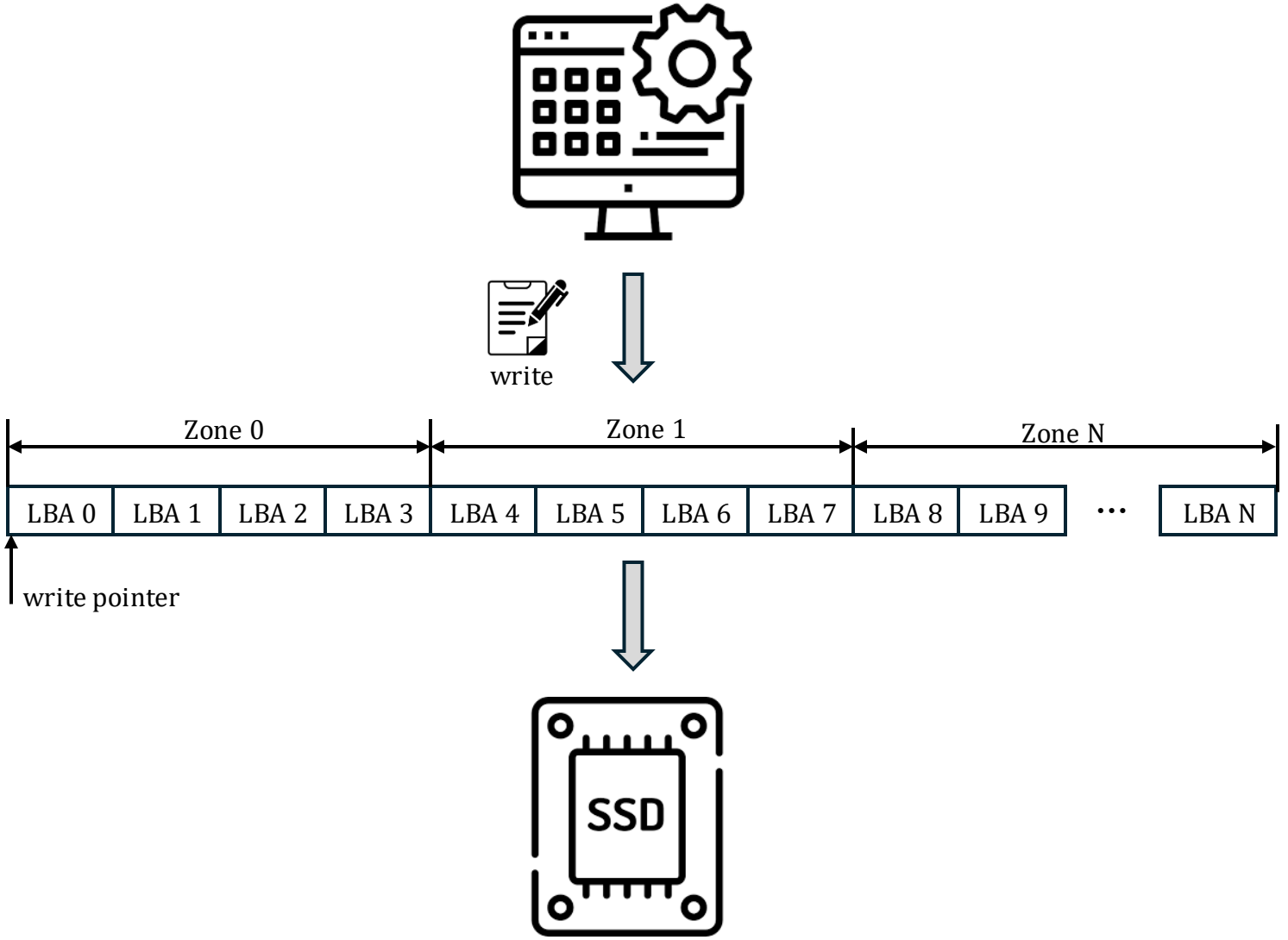




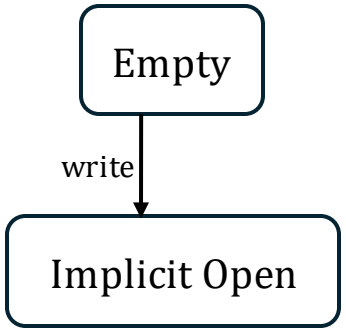


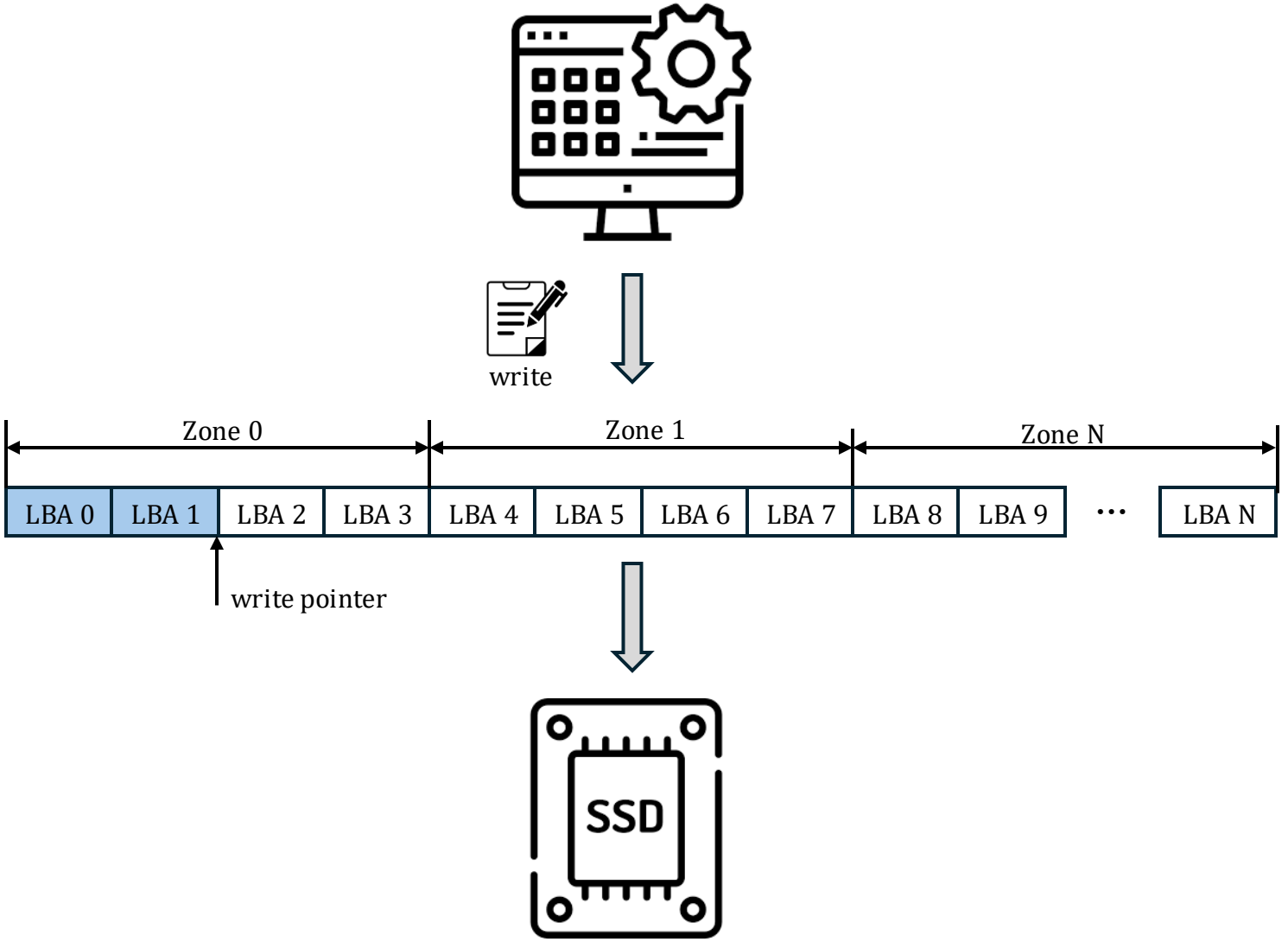
Zone State Management

Empty

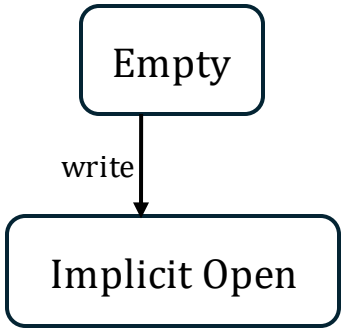


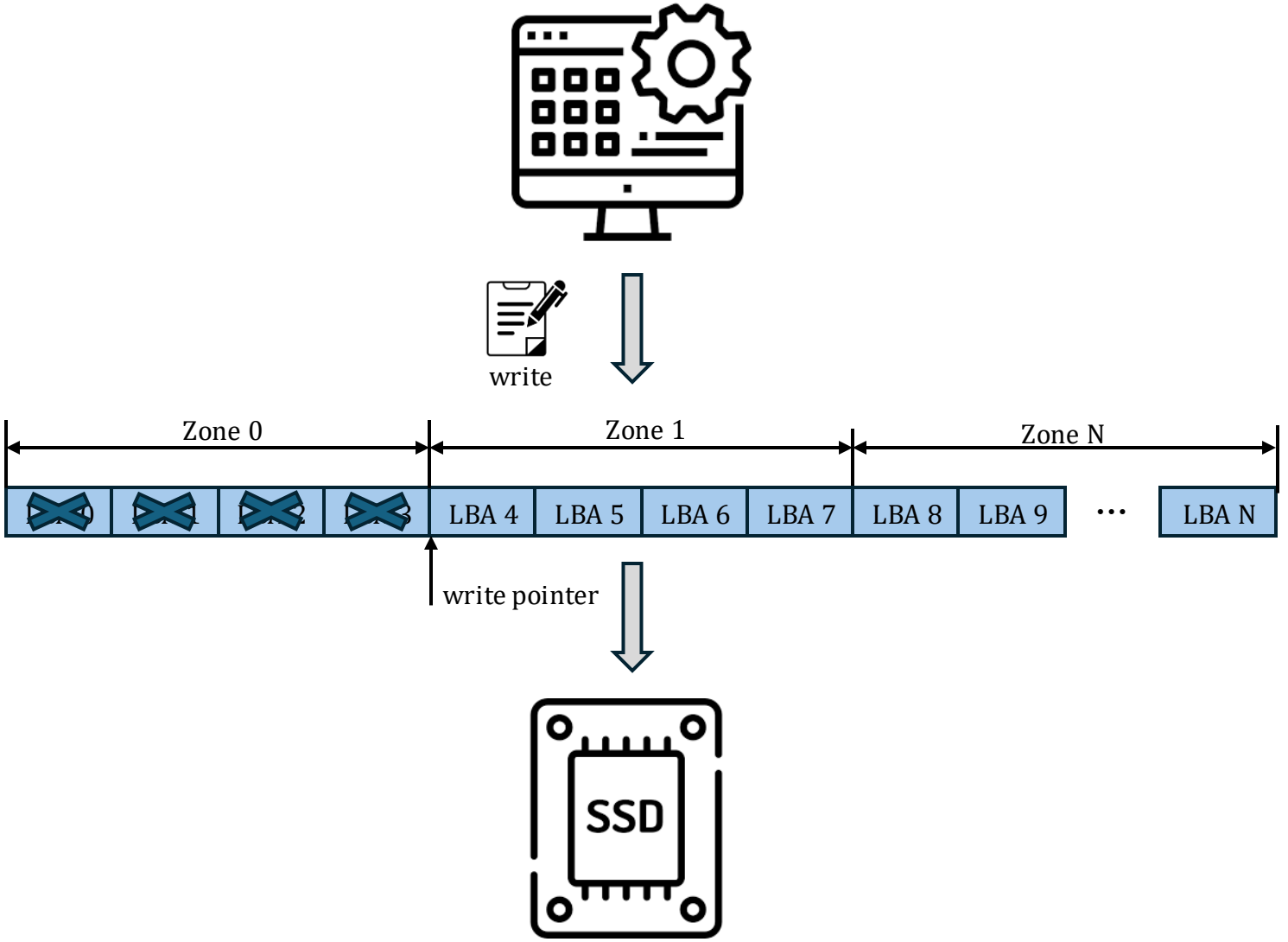
Zone State Management



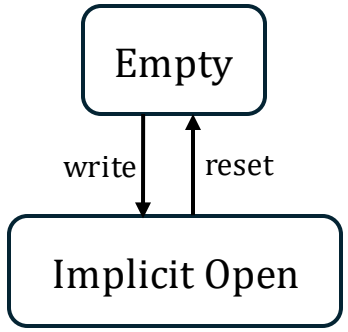


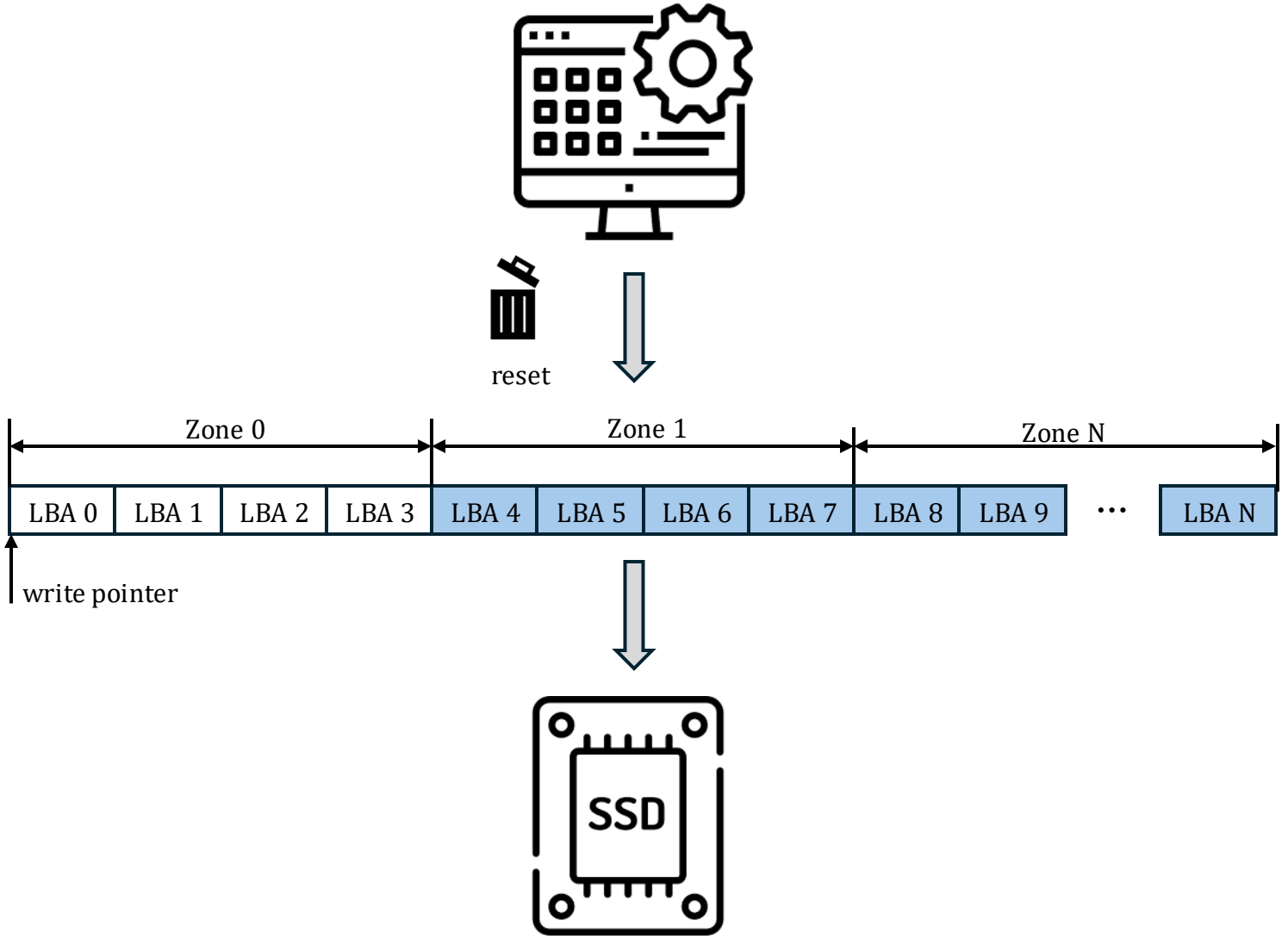
Zone State Management



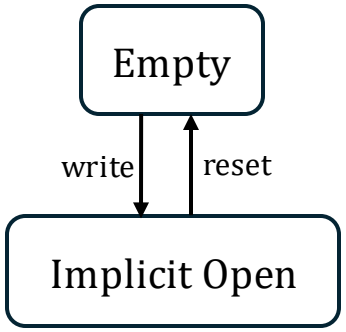


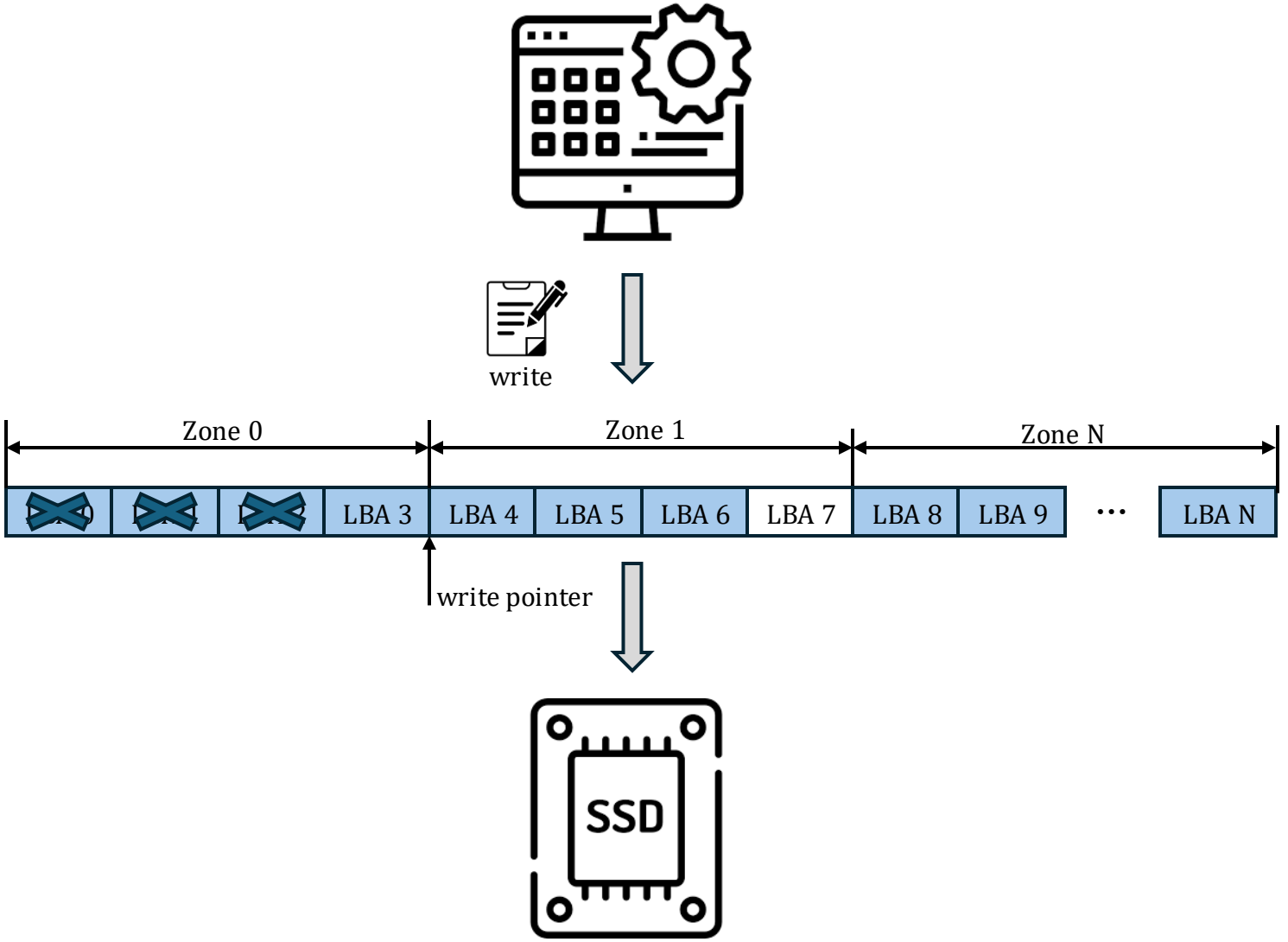
Zone State Management: Reset



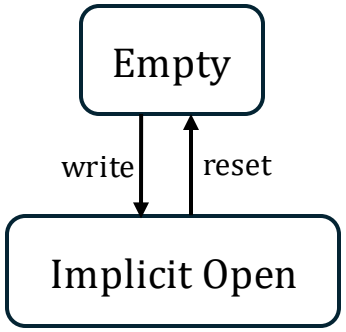


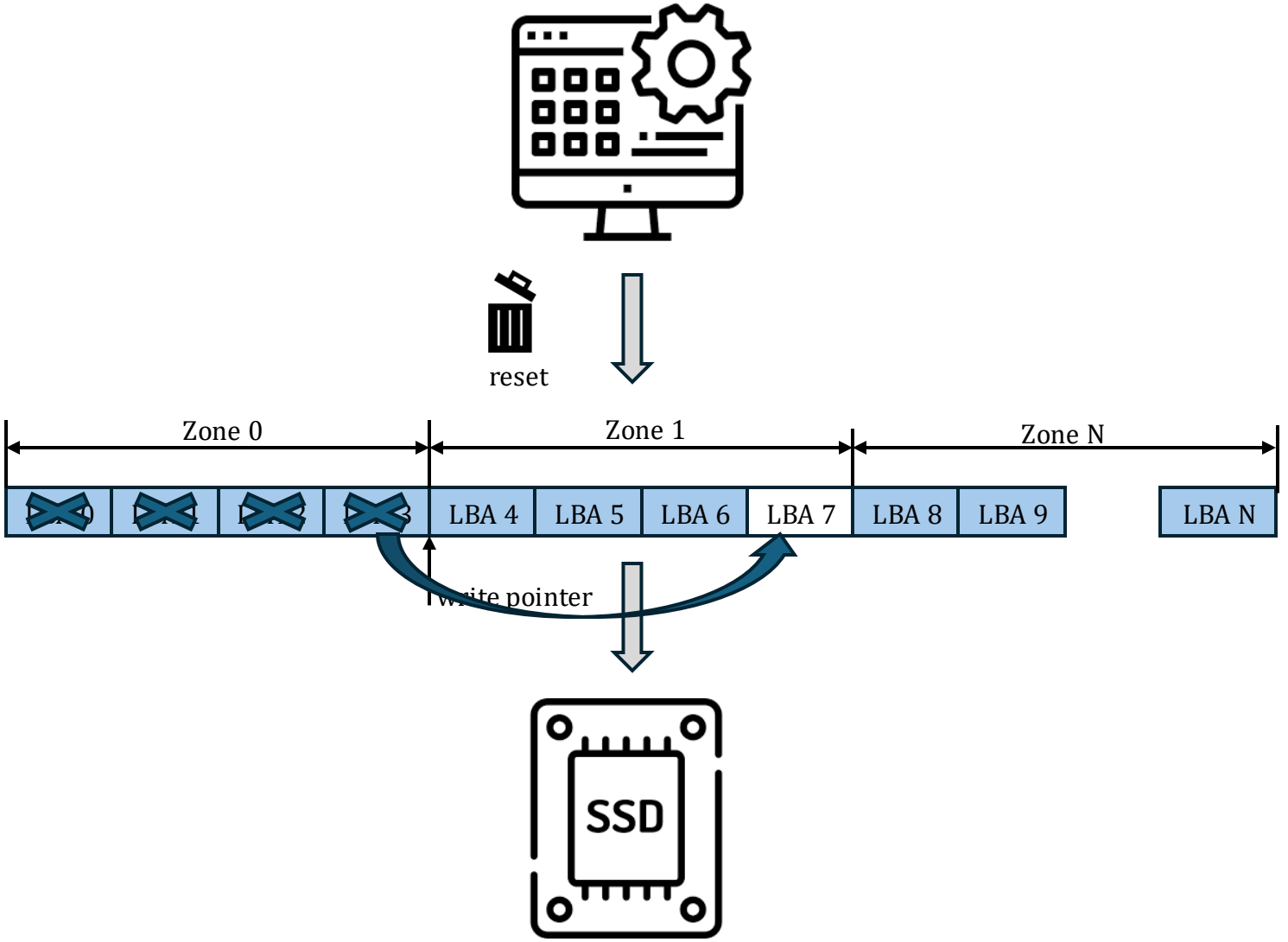
Host-Managed Garbage Collection



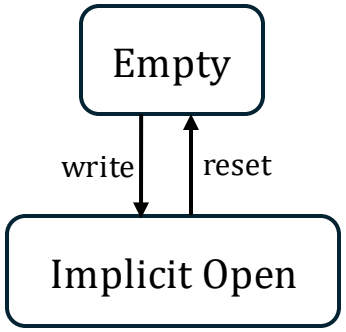


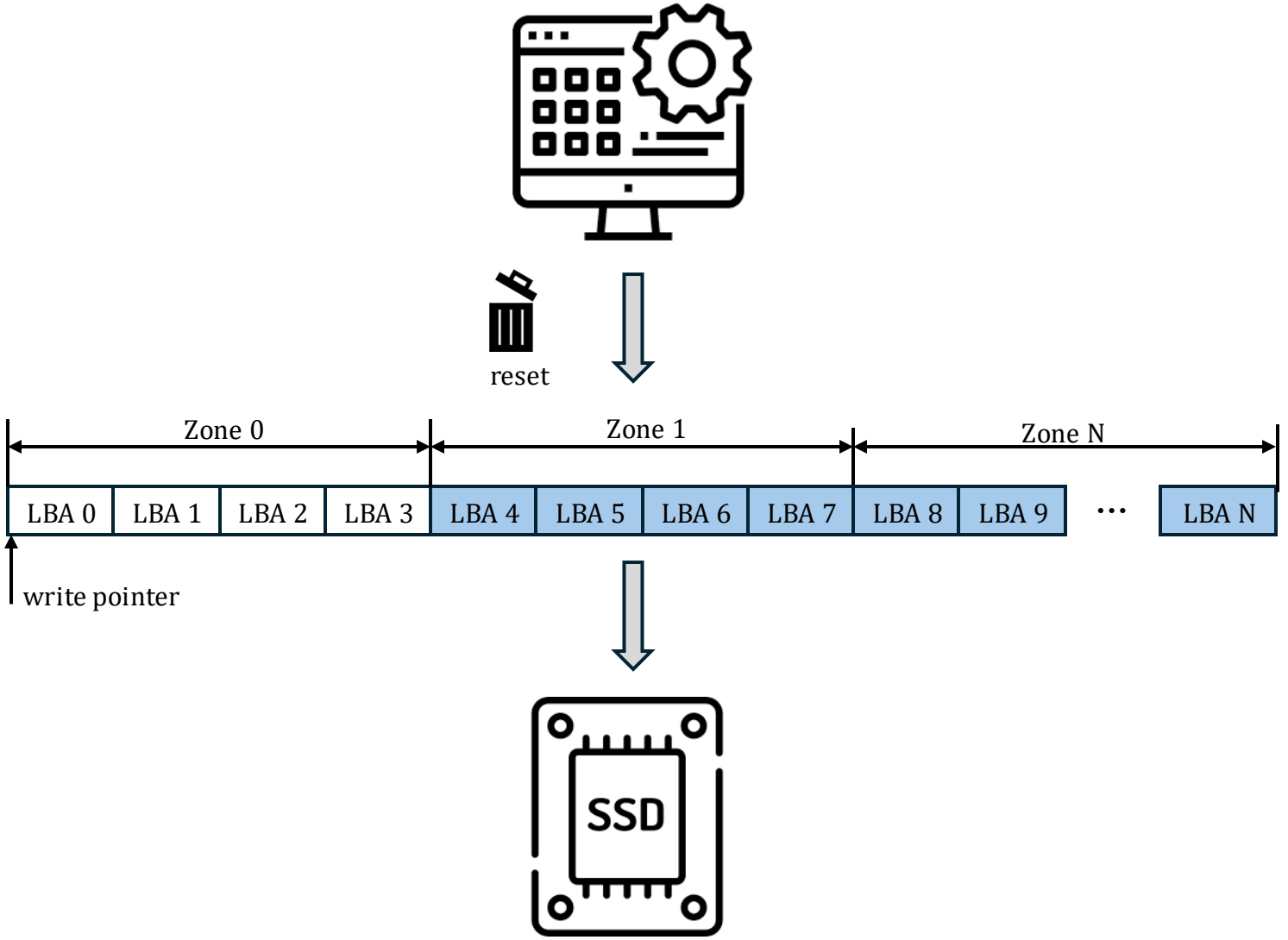
Host-Managed Garbage Collection



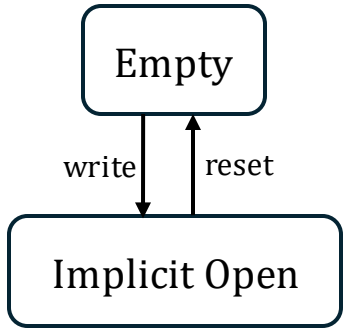


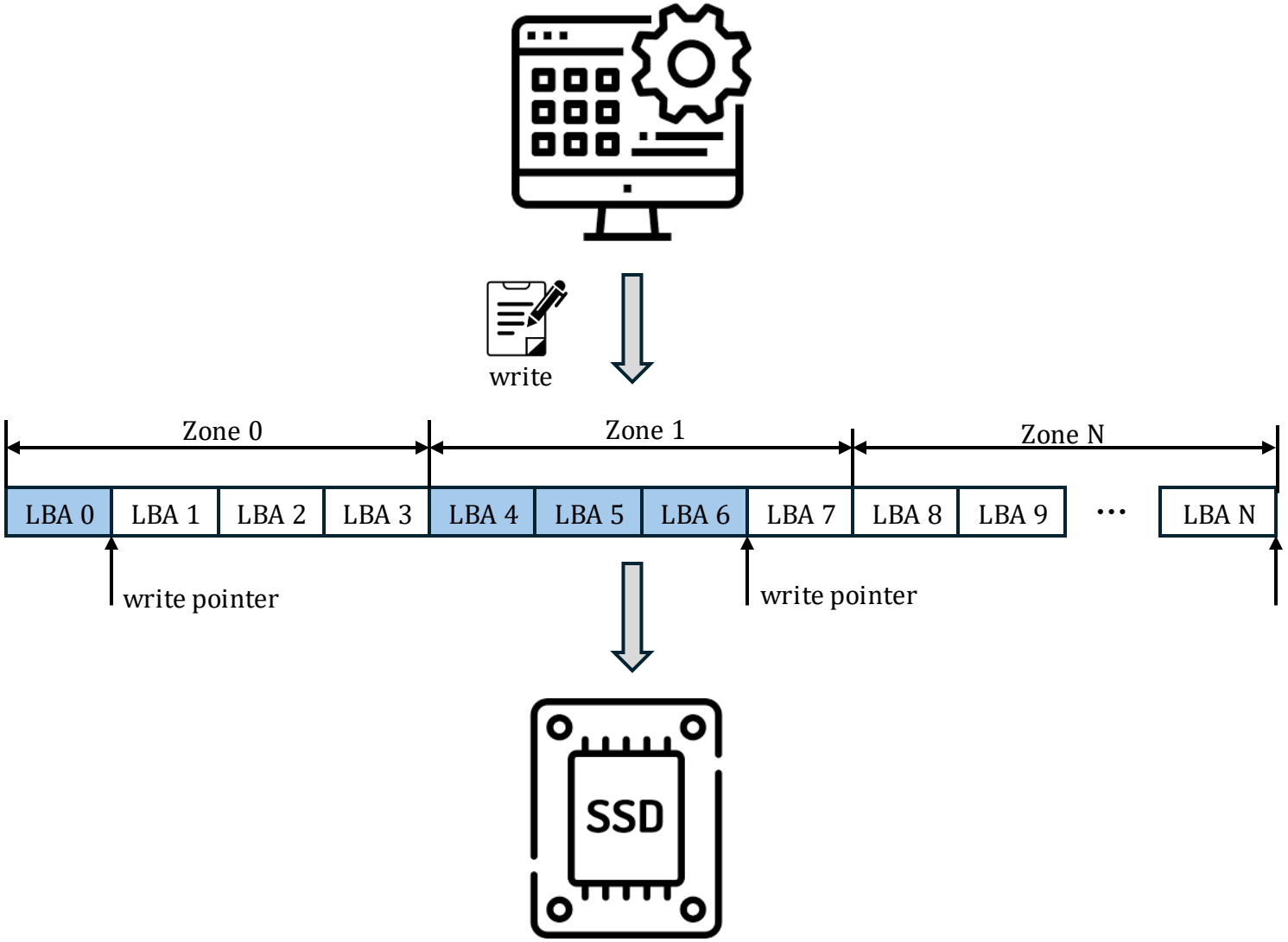
Host-Managed Garbage Collection



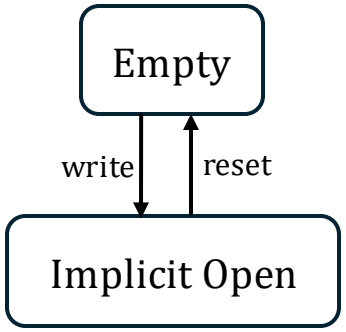


Actively written zones limit

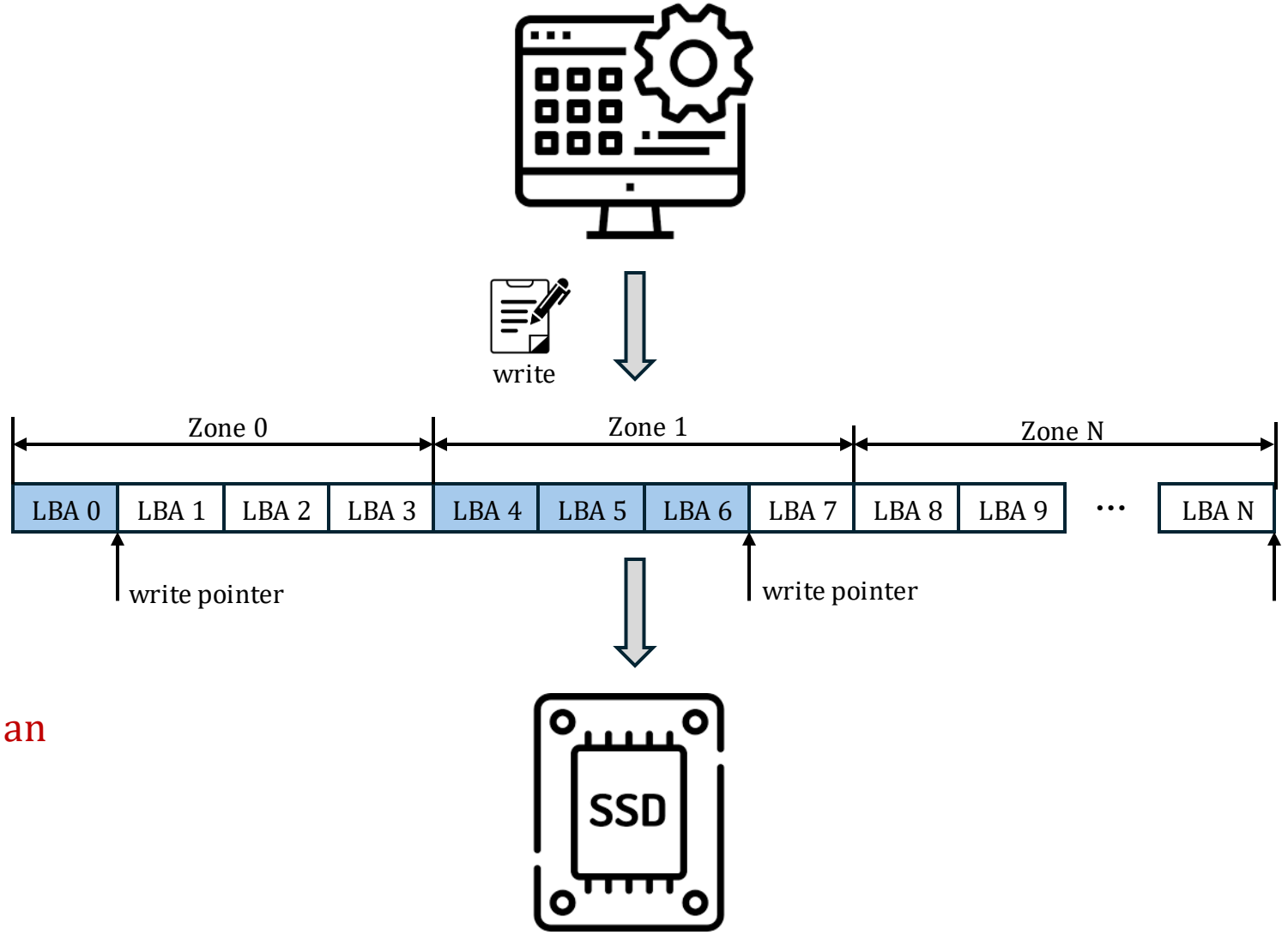




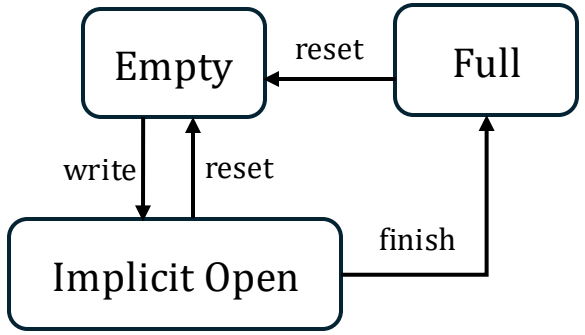
Actively written zones limit

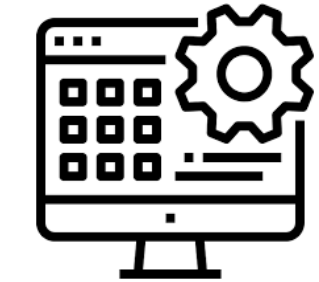


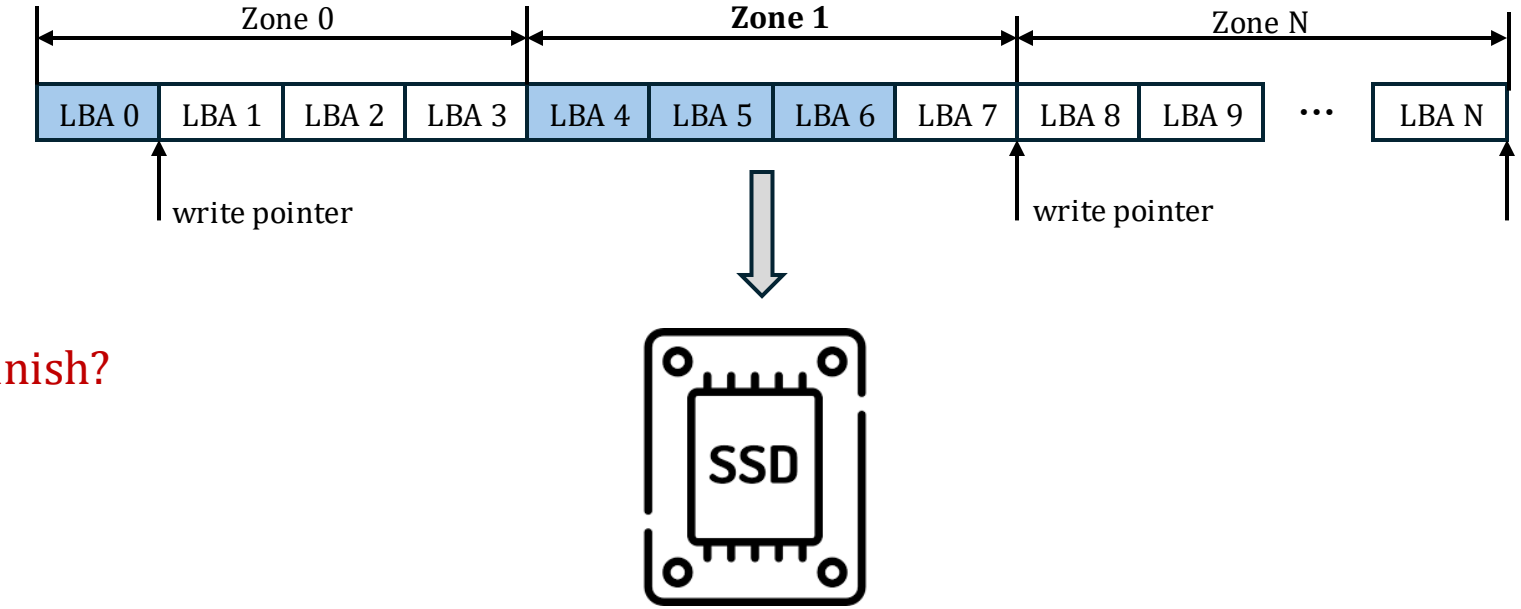
What if resetting is not an option?



Zone Management: Finish

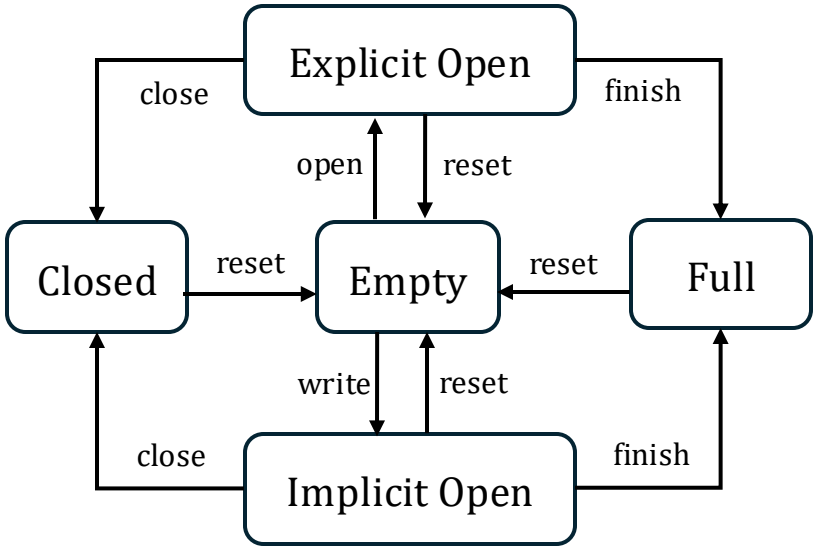


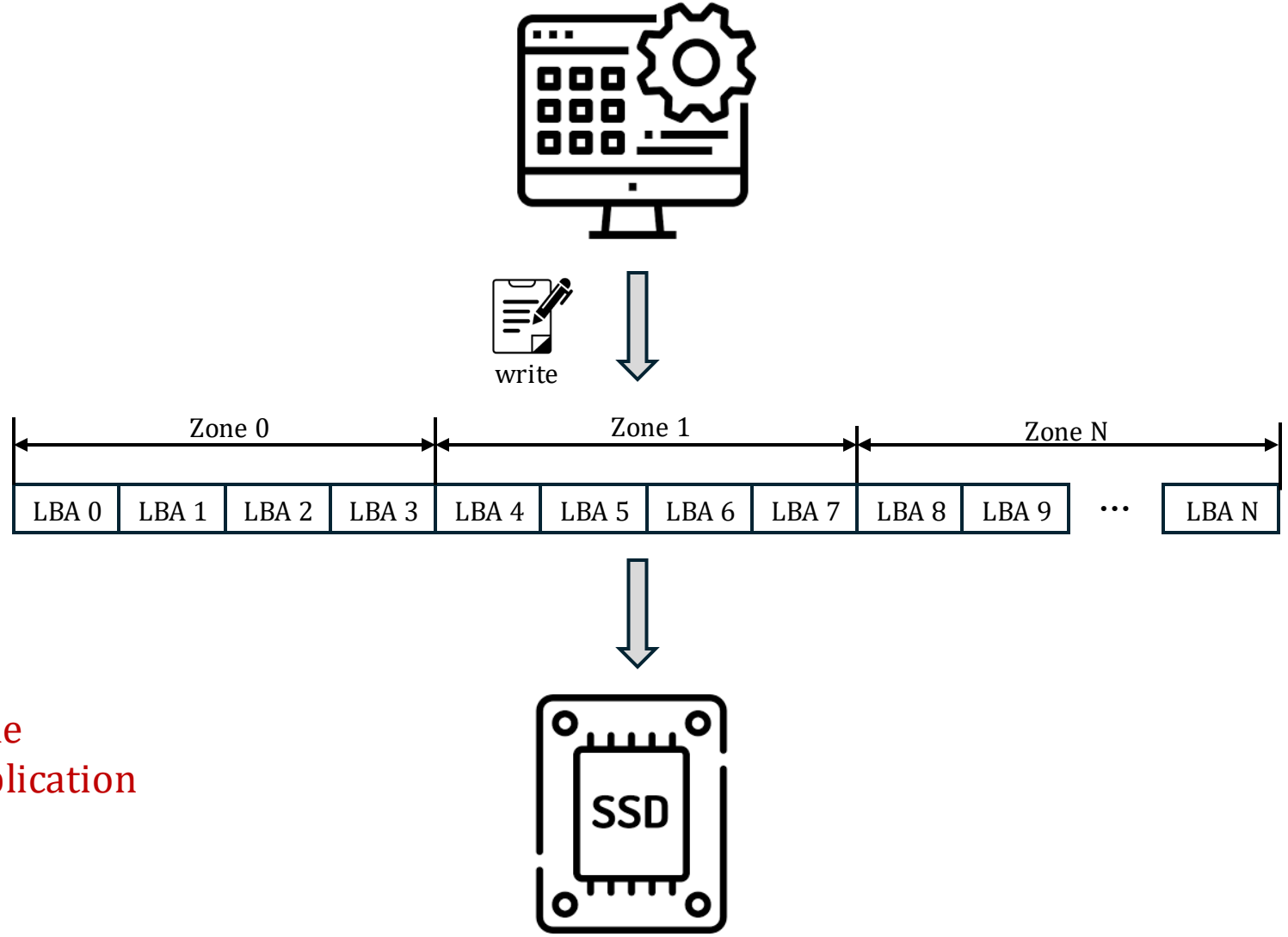




Which zone should I finish?

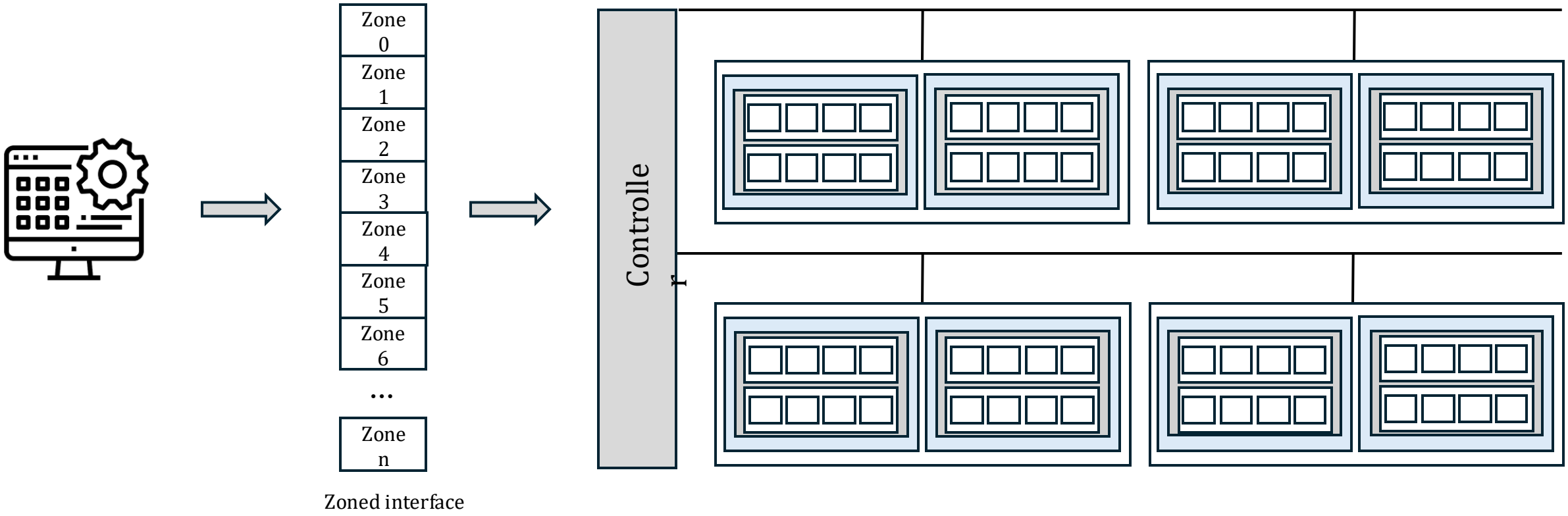
Zone State Management





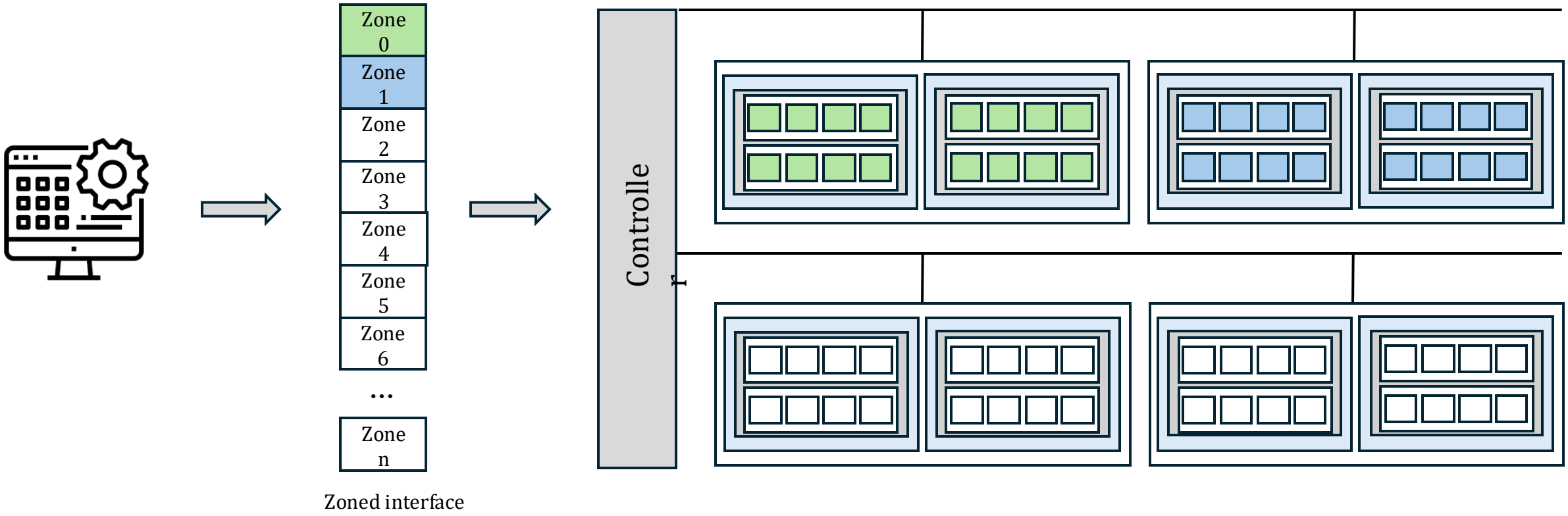
How does state machine management affect application complexity?

ZNS Controller Design: Mapping



ZNS Controller Design: Mapping

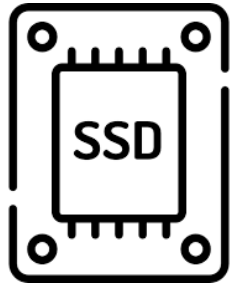
Static Mapping Scheme: Less Striping



Zoned Interface

Append only interface

Garbage collection to reclaim space



Wear leveling

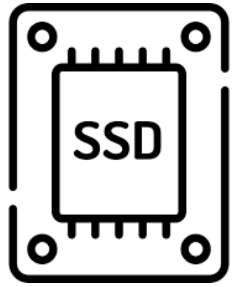
Optimal data placement

Zoned Interface

Append only interface

Garbage collection to reclaim space

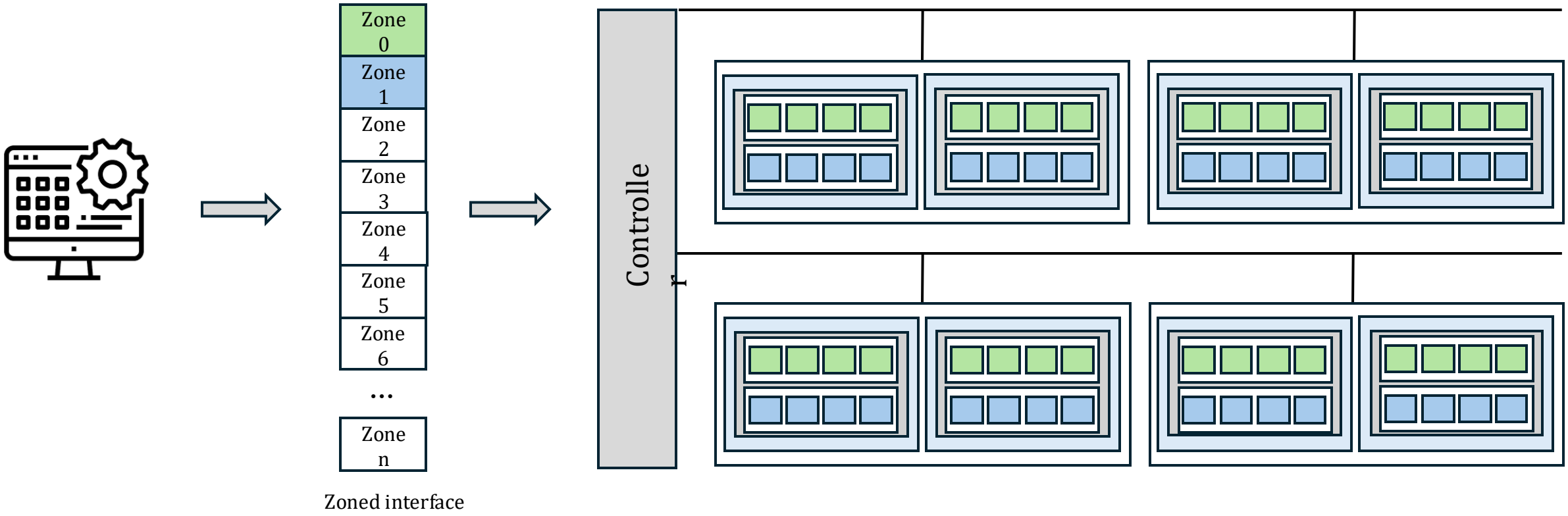
Optimal data placement



Wear leveling

ZNS Controller Design: Mapping

Static Mapping Scheme: Fully Striped

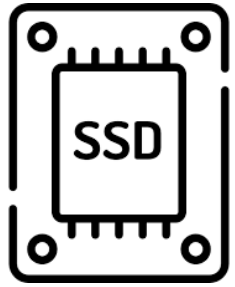


Zoned Interface

Append only interface

Garbage collection to reclaim space

Optimal data placement

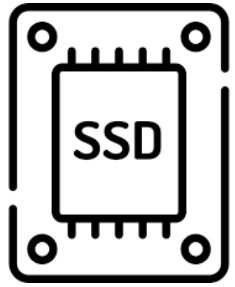


Wear leveling

Zoned Interface

Append only interface

Garbage collection to reclaim space

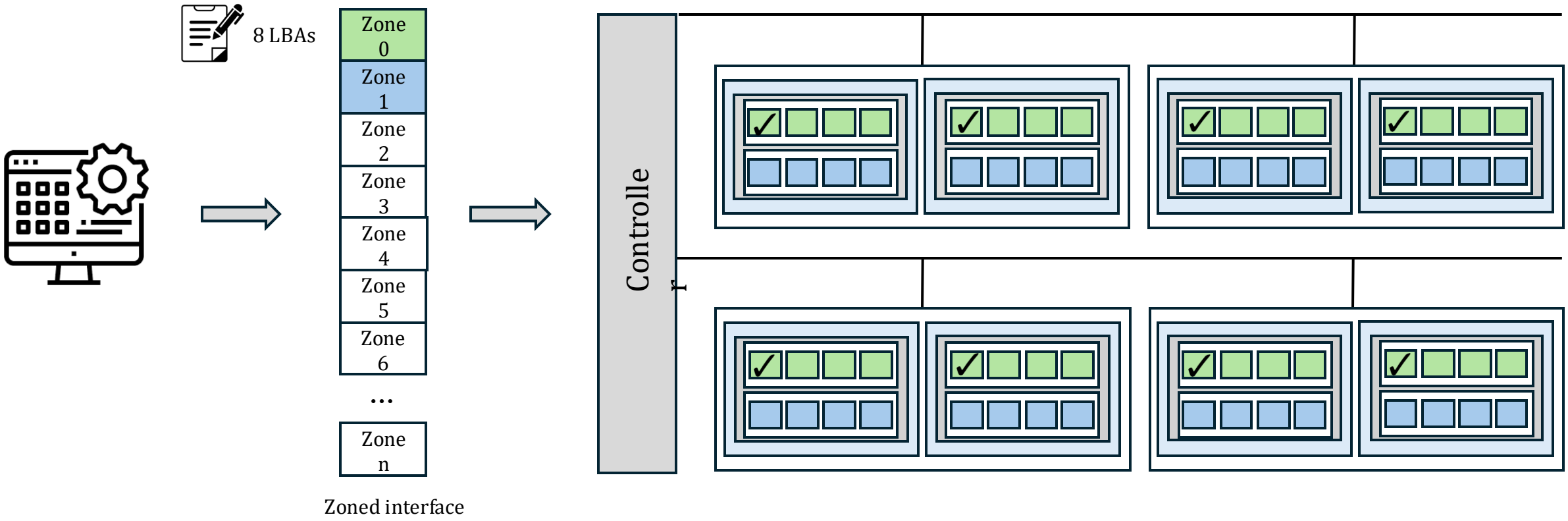


Wear leveling

Optimal data placement

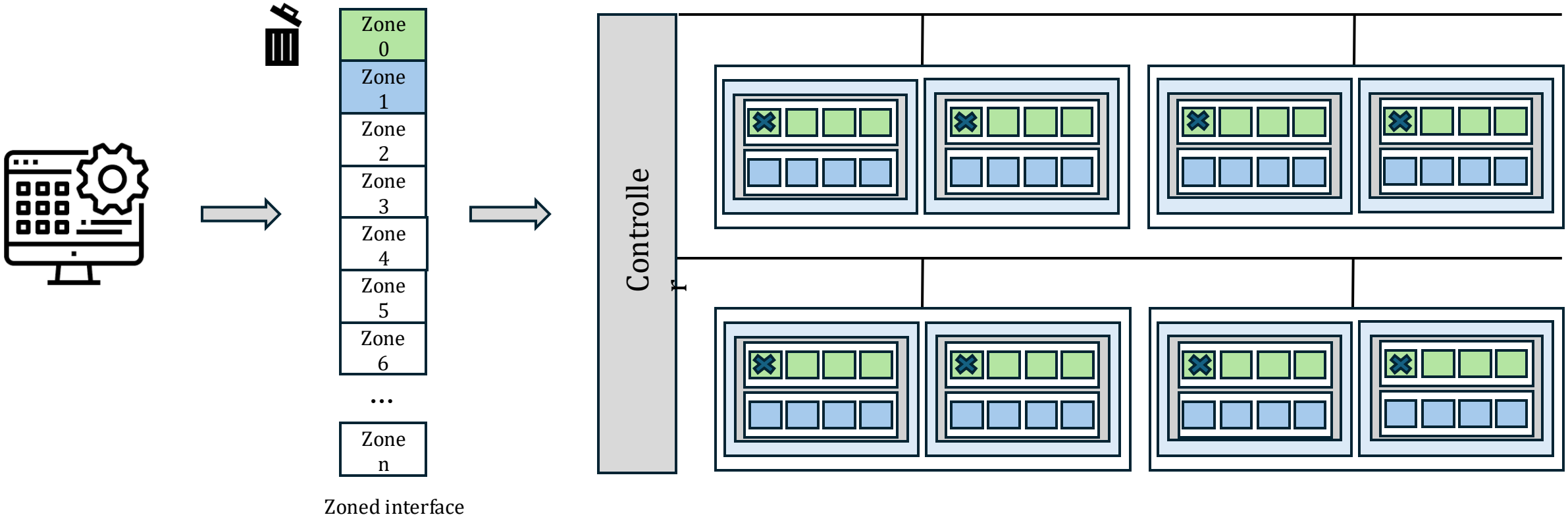
ZNS Controller Design: Write

Static Mapping Scheme



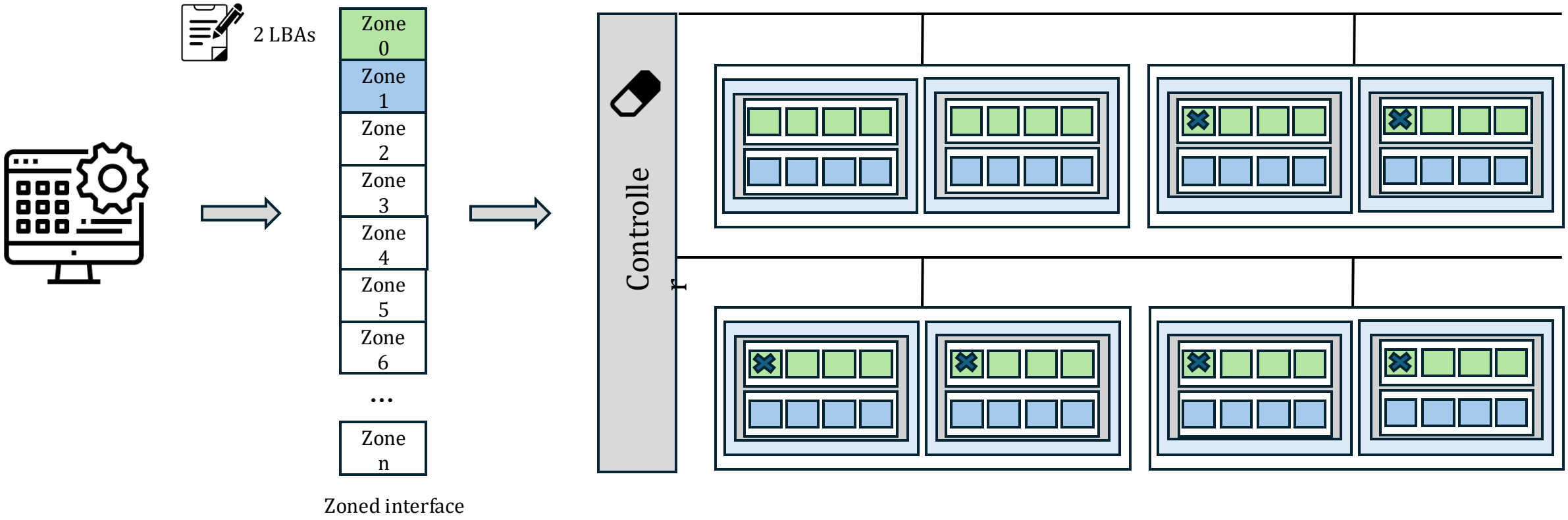
ZNS Controller Design: Reset

Static Mapping Scheme



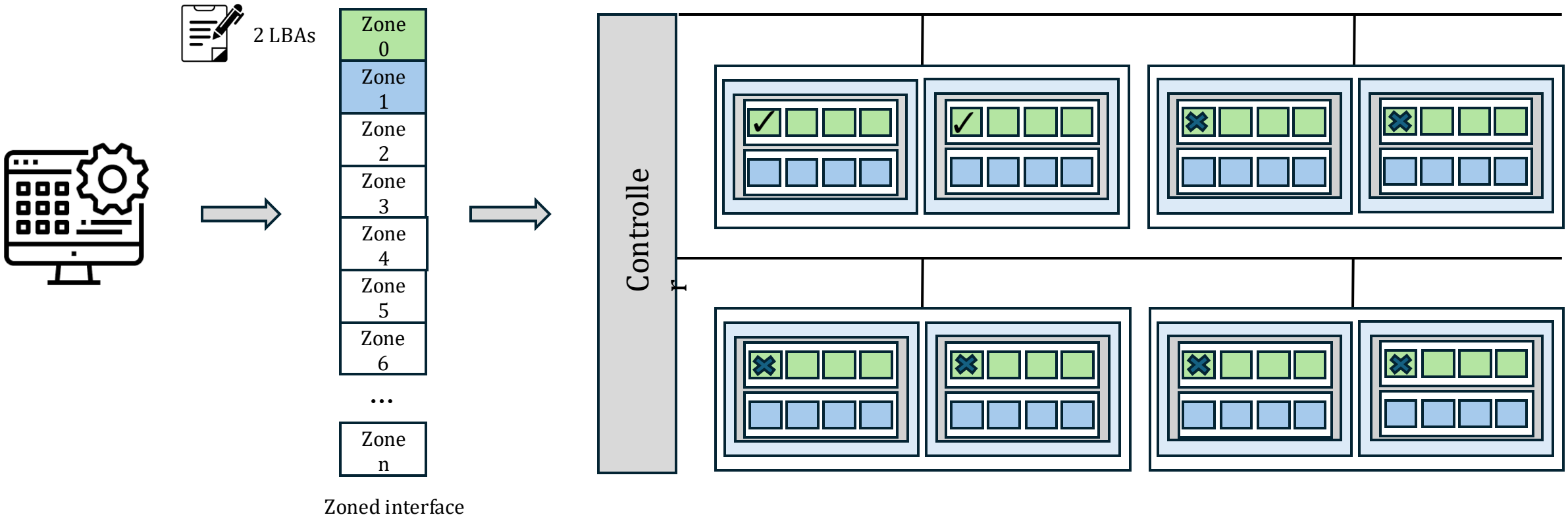
ZNS Controller Design: Reset

Static Mapping Scheme



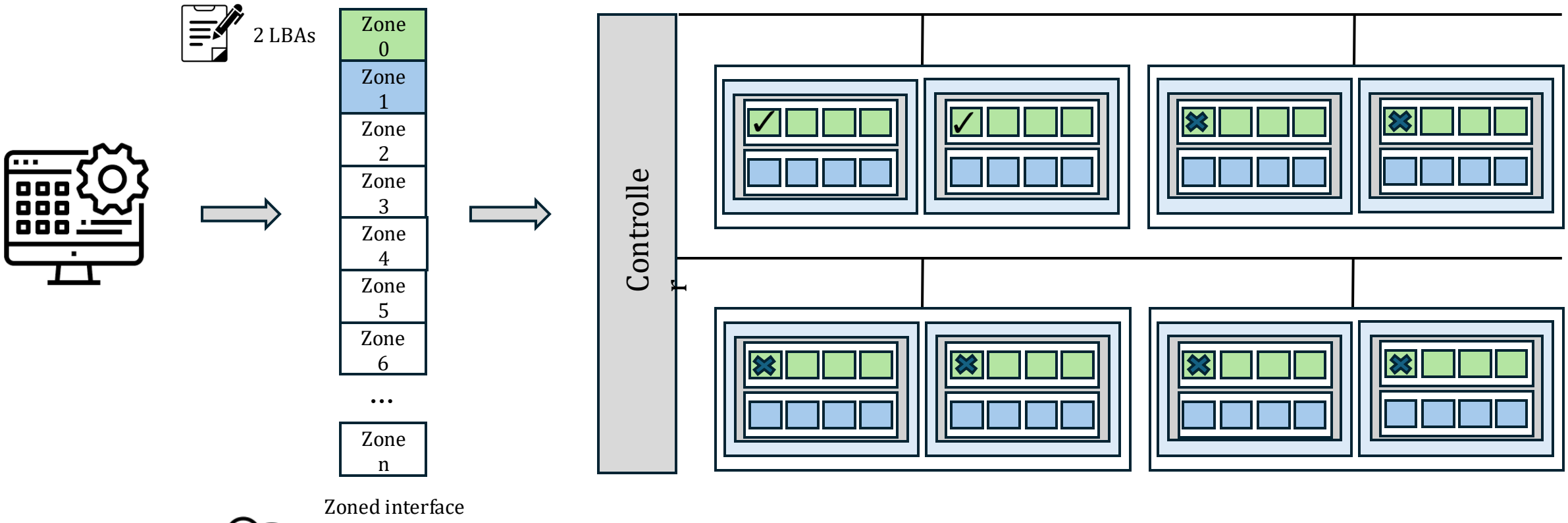
ZNS Controller Design: Reset

Static Mapping Scheme



ZNS Controller Design: Reset

Static Mapping Scheme

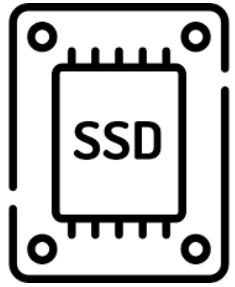


Any issues with static mapping scheme?

Zoned Interface

Append only interface

Garbage collection to reclaim space



Wear leveling

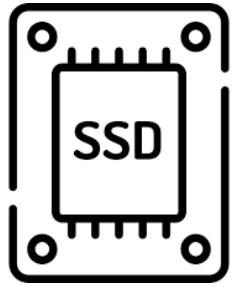
Optimal data placement

Zoned Interface

Append only interface

Garbage collection to reclaim space

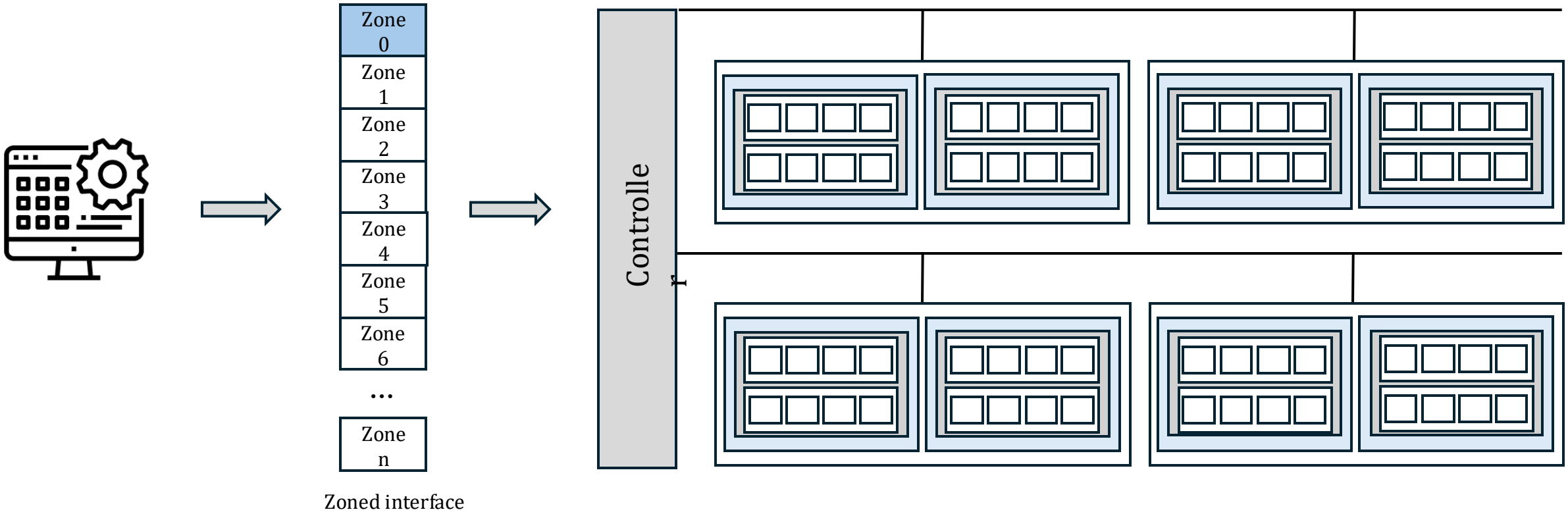
Wear leveling



Optimal data placement

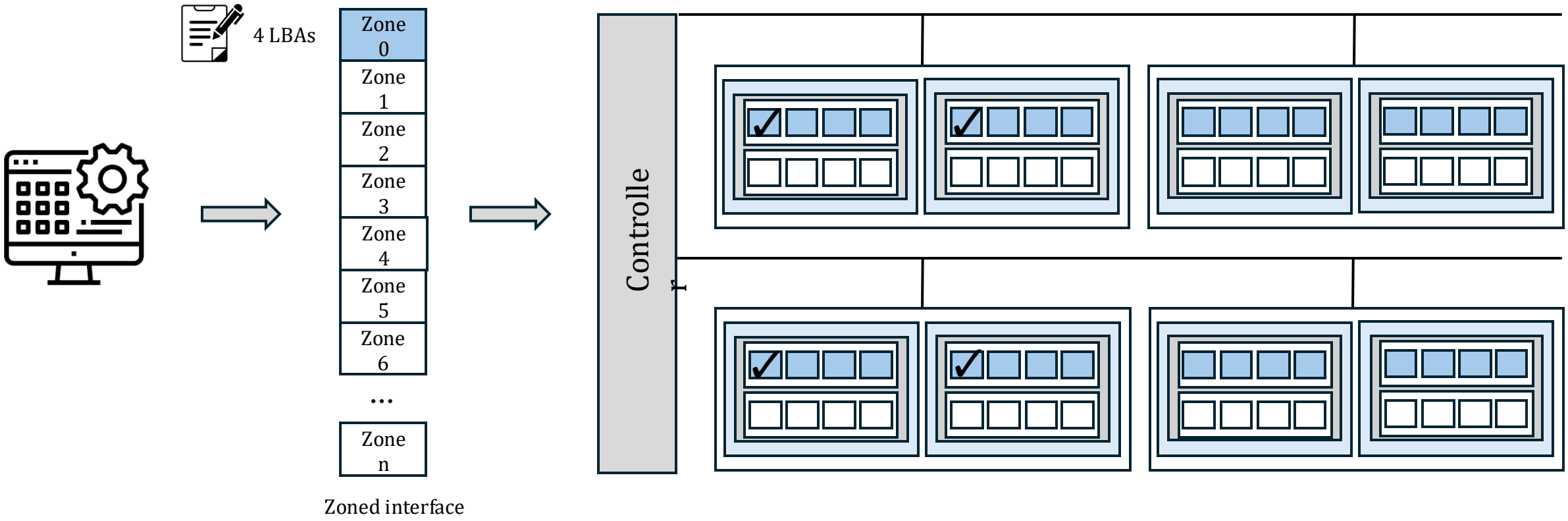
ZNS Controller Design: Mapping

Dynamic Mapping Scheme



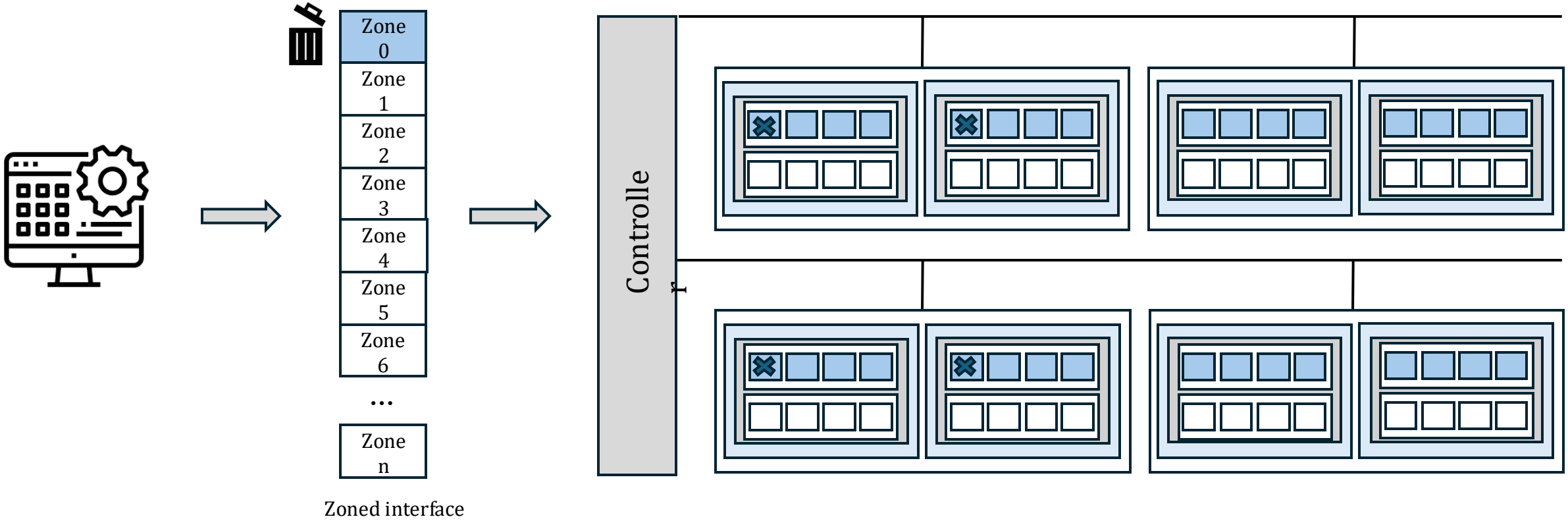
ZNS Controller Design: Mapping

Dynamic Mapping Scheme



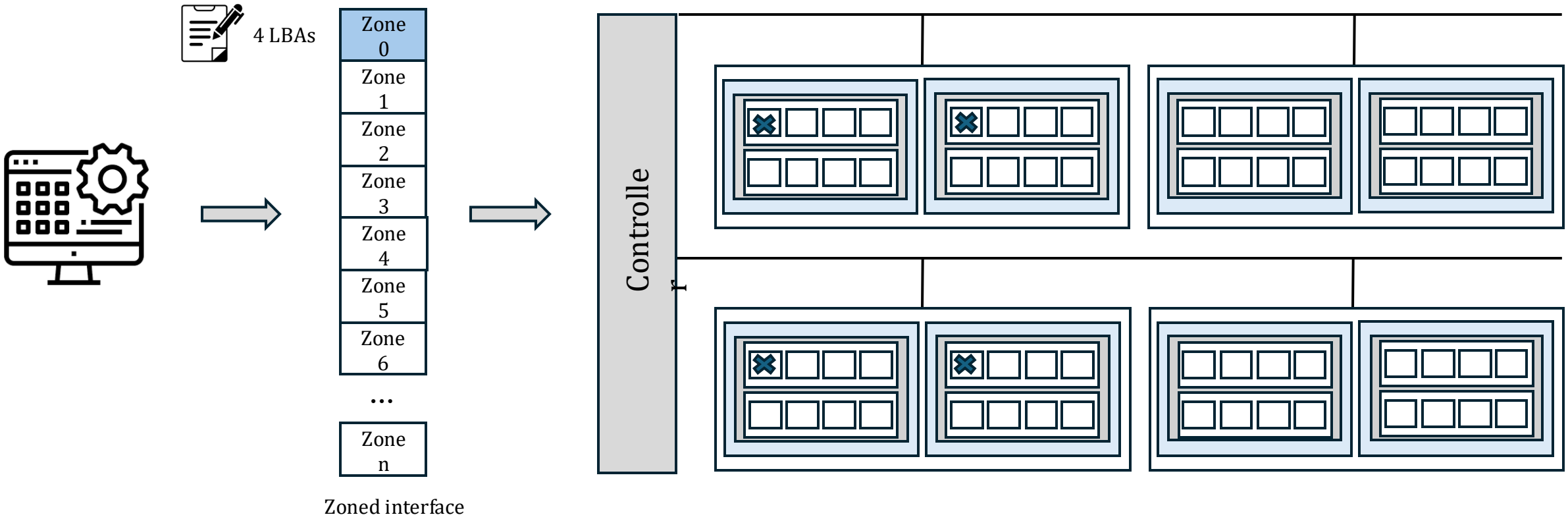
ZNS Controller Design: Mapping

Dynamic Mapping Scheme



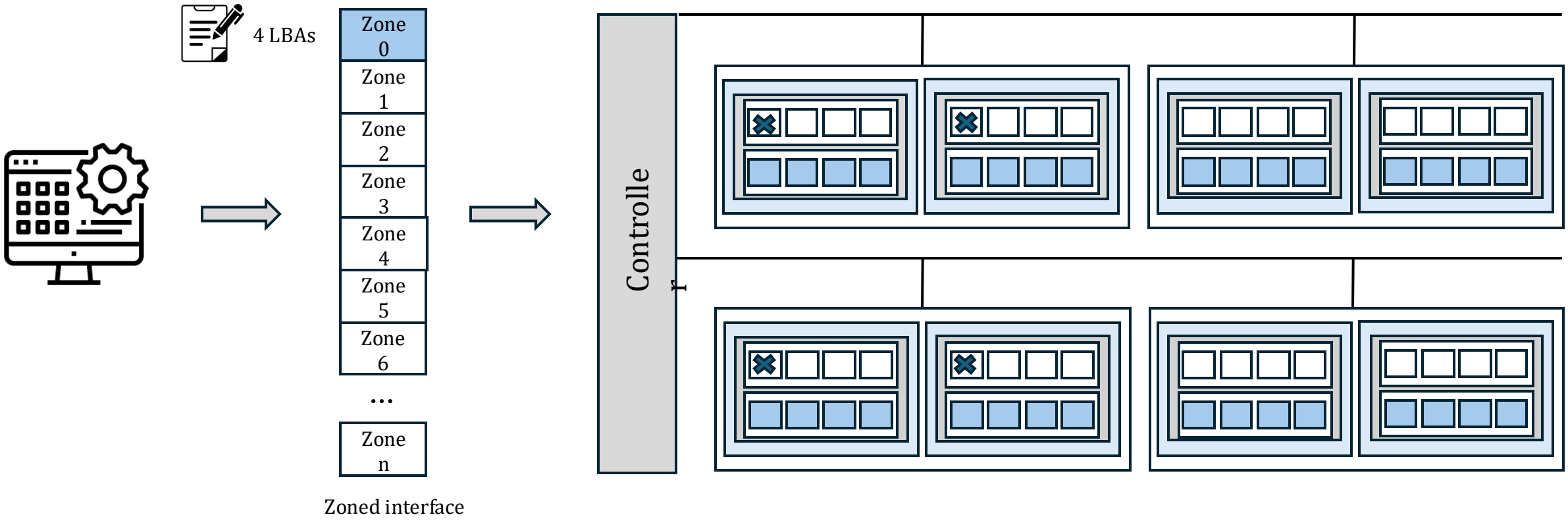
ZNS Controller Design: Mapping

Dynamic Mapping Scheme



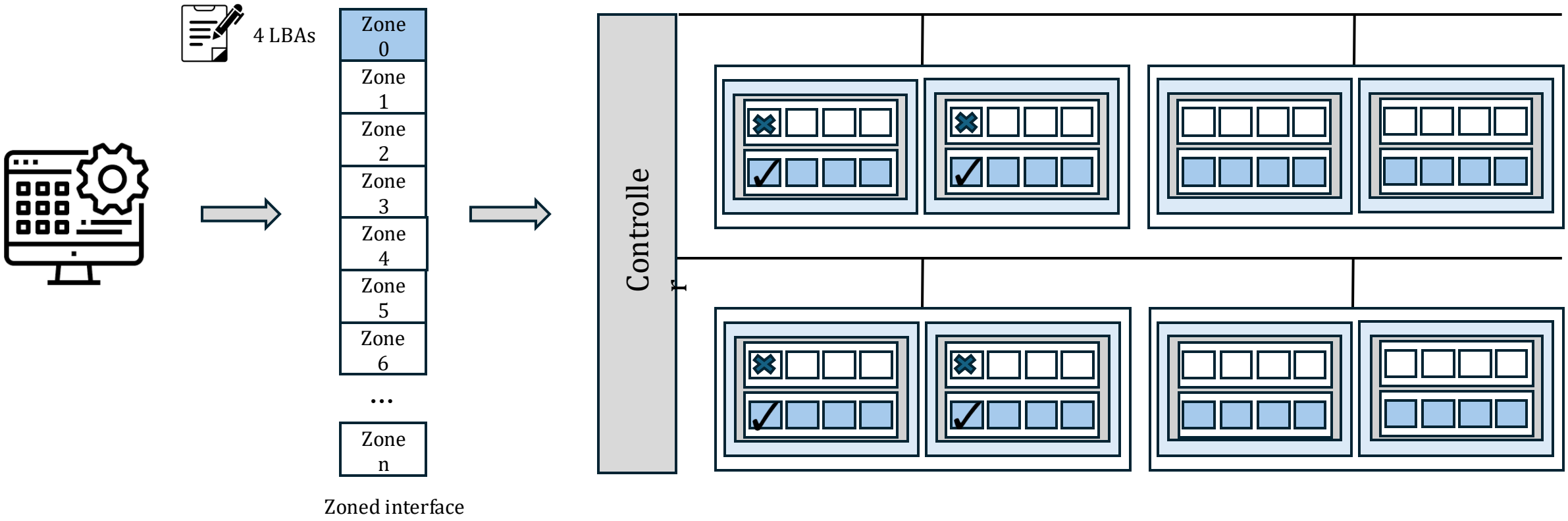
ZNS Controller Design: Mapping

Dynamic Mapping Scheme



ZNS Controller Design: Mapping

Dynamic Mapping Scheme

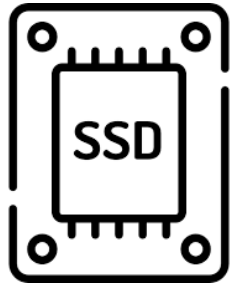


Zoned Interface

Append only interface

Garbage collection to reclaim space

Wear leveling

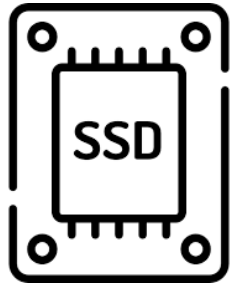


Optimal data placement

Zoned Interface

Append only interface

Garbage collection to reclaim space

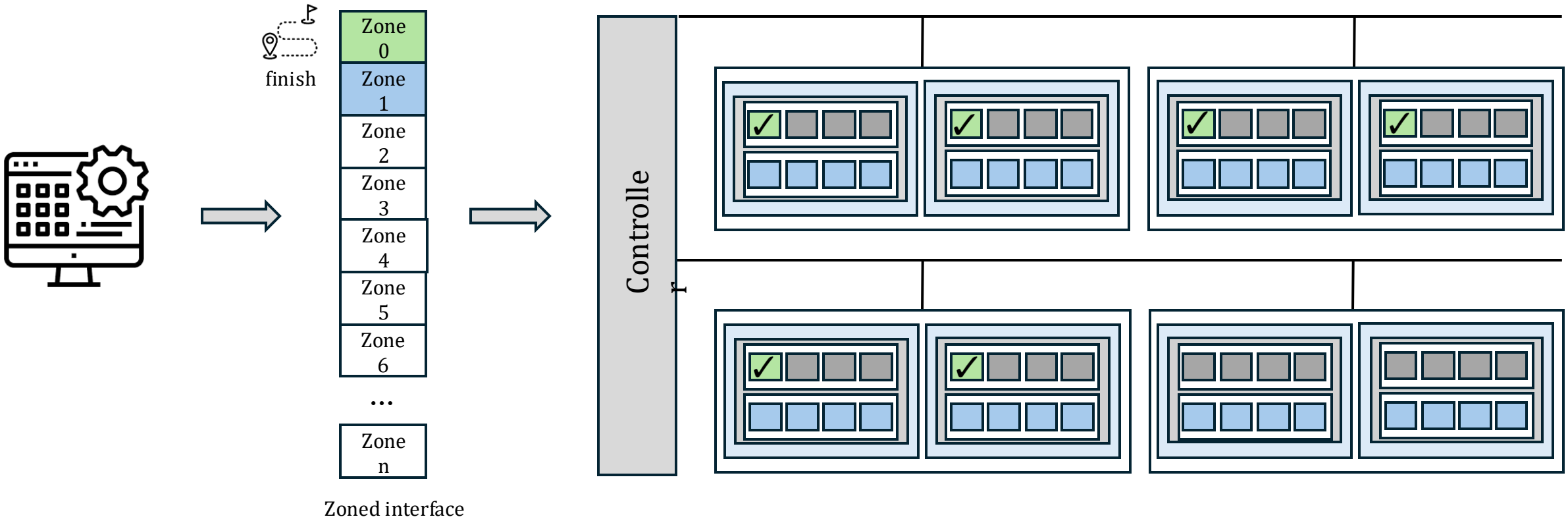


Wear leveling

Optimal data placement

ZNS Controller Design: Finish

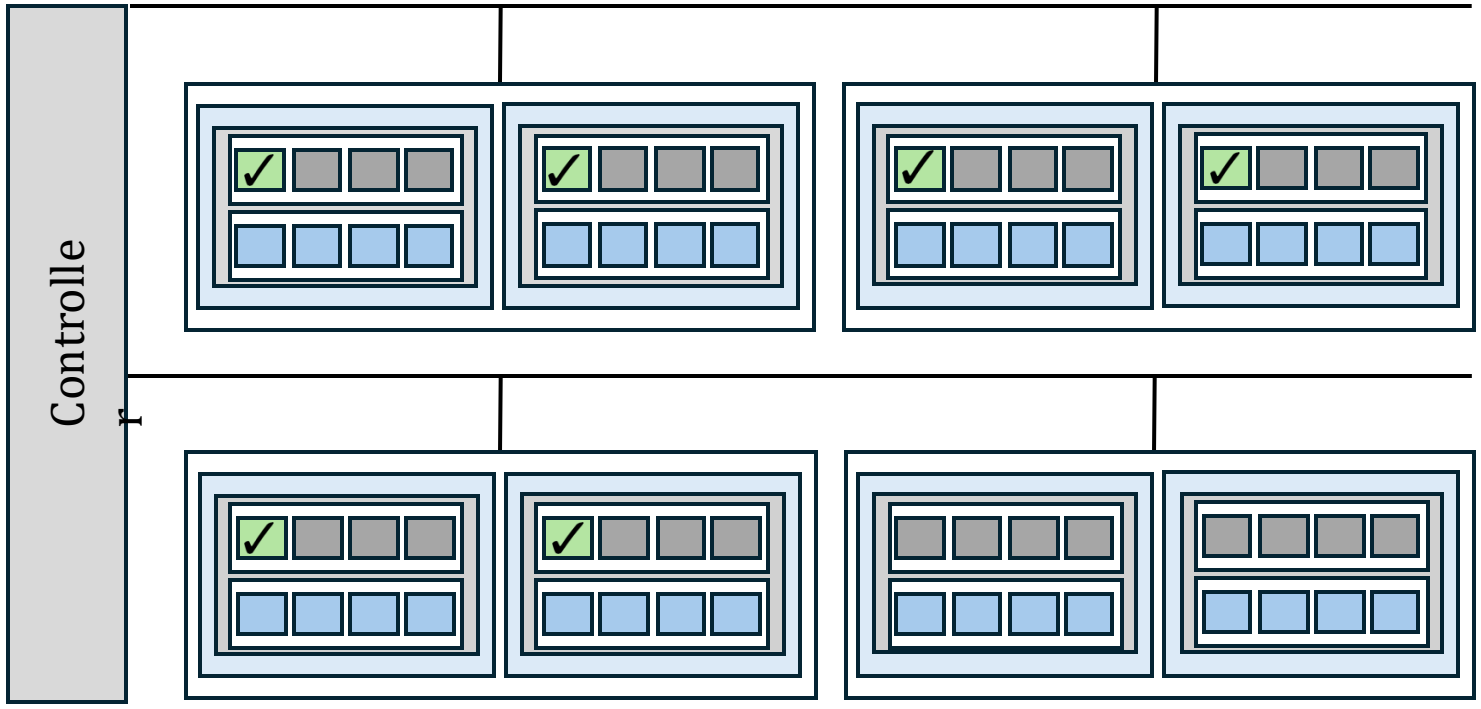
Finish sends dummy writes to all the allocated blocks



ZNS Controller Design: Finish

Finish sends dummy writes to all the allocated blocks

Issues with dummy writes?

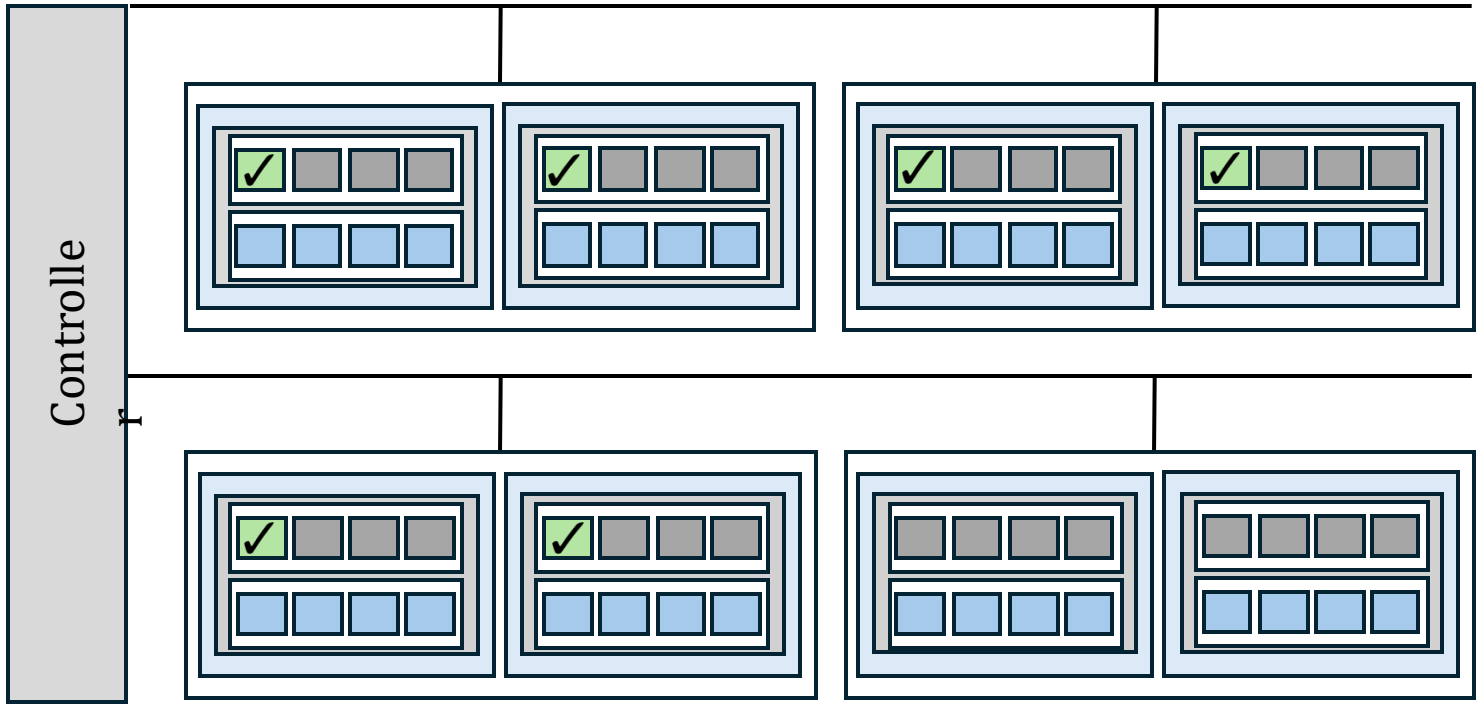


ZNS Controller Design: Finish

Finish sends dummy writes to all the allocated blocks

Issues with dummy writes?

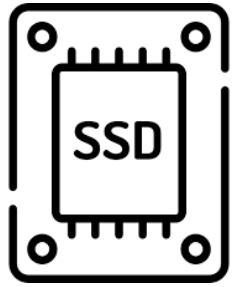
- ✗ Write Amplification
- ✗ Interference with host I/O



Zoned Interface

Append only interface

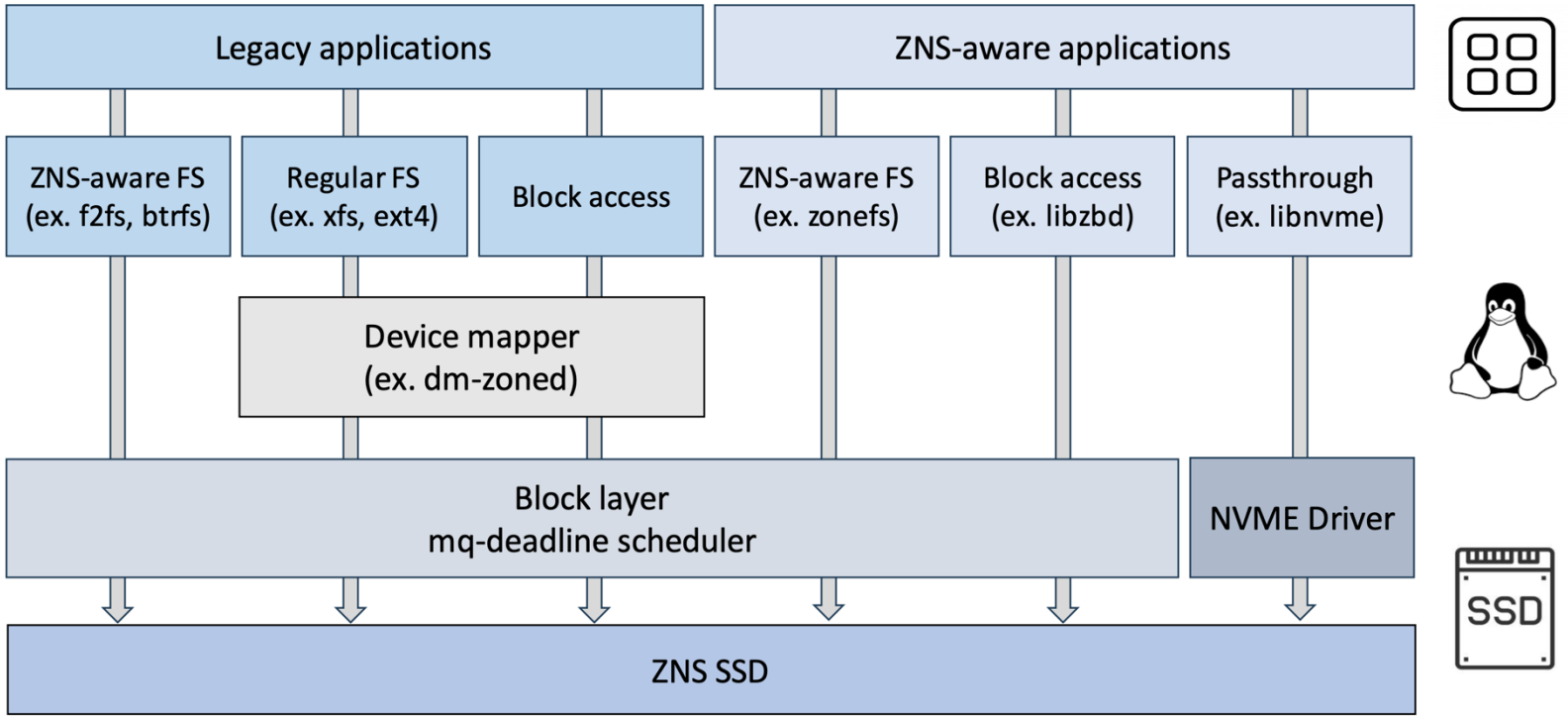
Garbage collection to reclaim space
State machine management



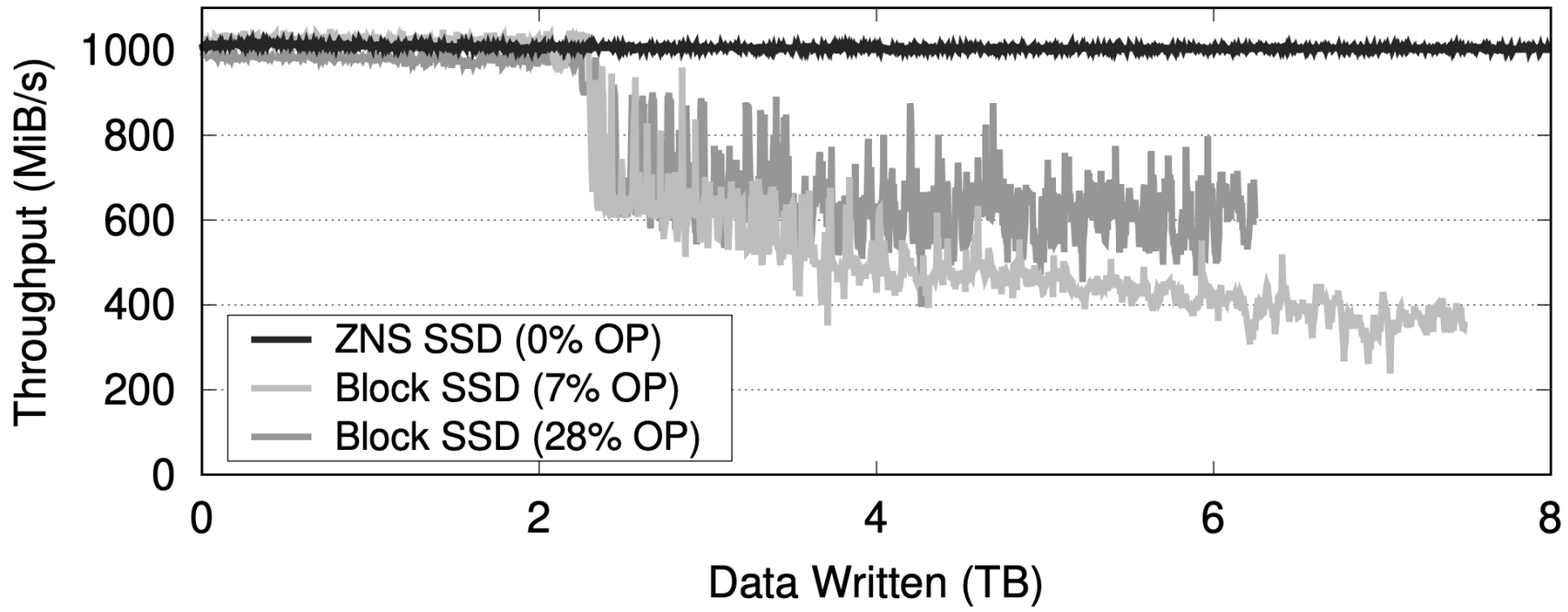
Wear leveling

Optimal data placement

ZNS Access Paths



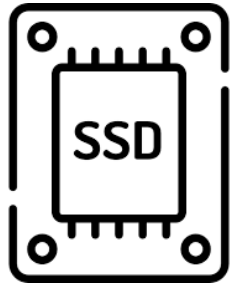
ZNS Access Paths



Avoiding Zoned Interface Tax

Append only interface

Garbage collection to reclaim space
State machine management

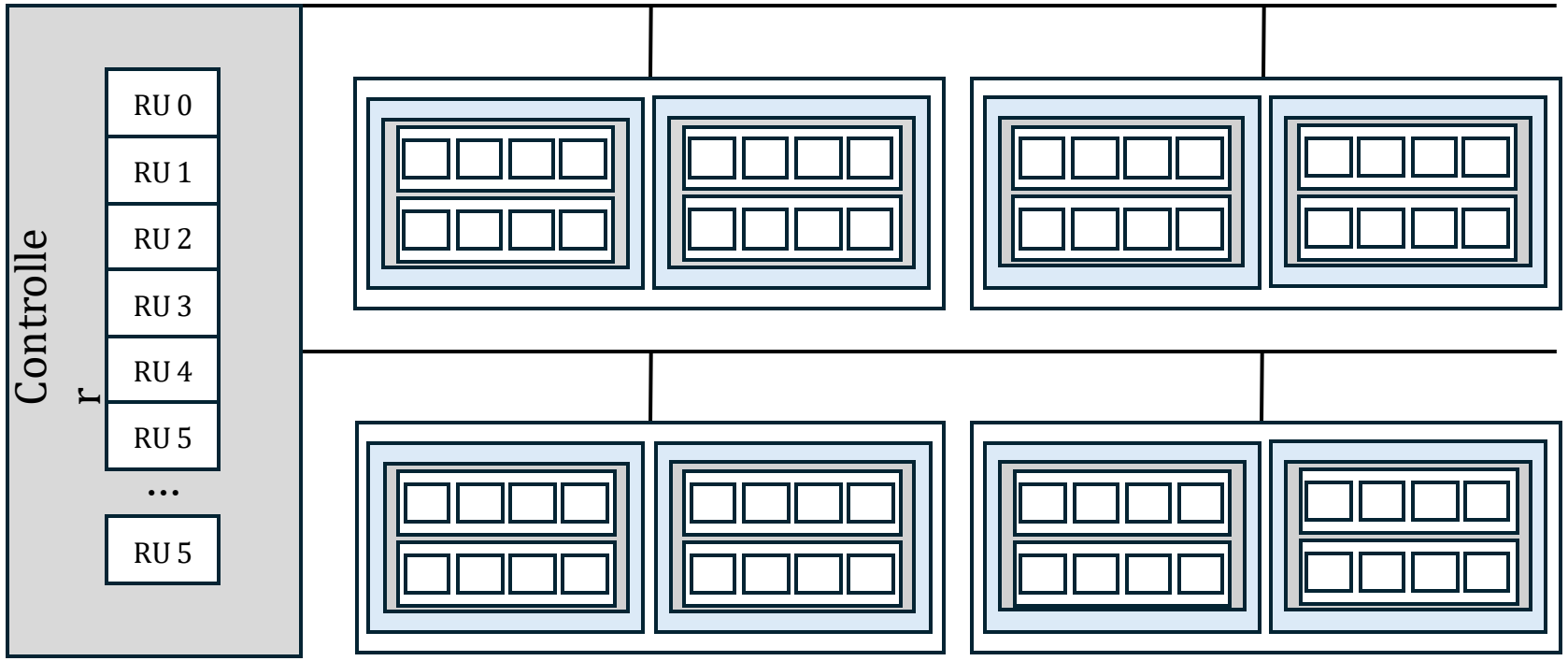


Wear leveling

Optimal data placement

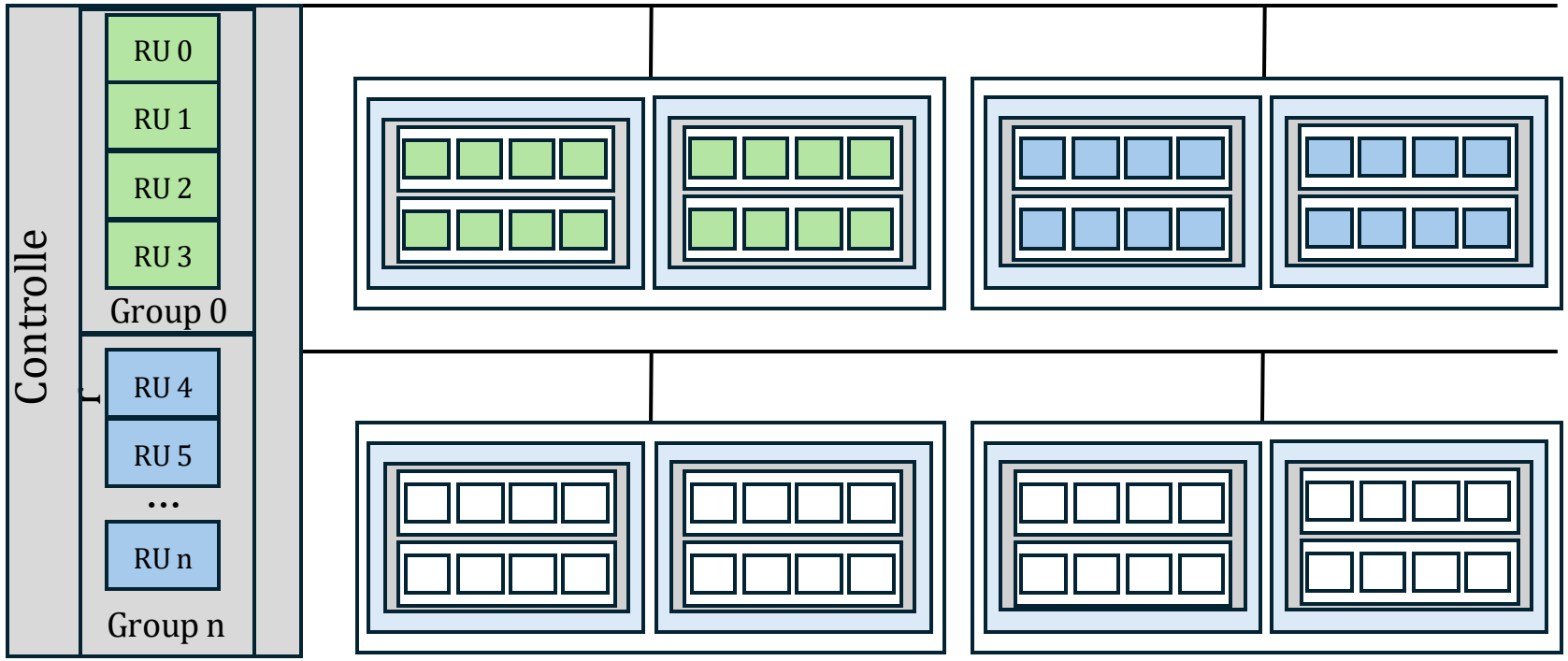
FDP Controller Design: Mapping

Reclaim Unit (RU)



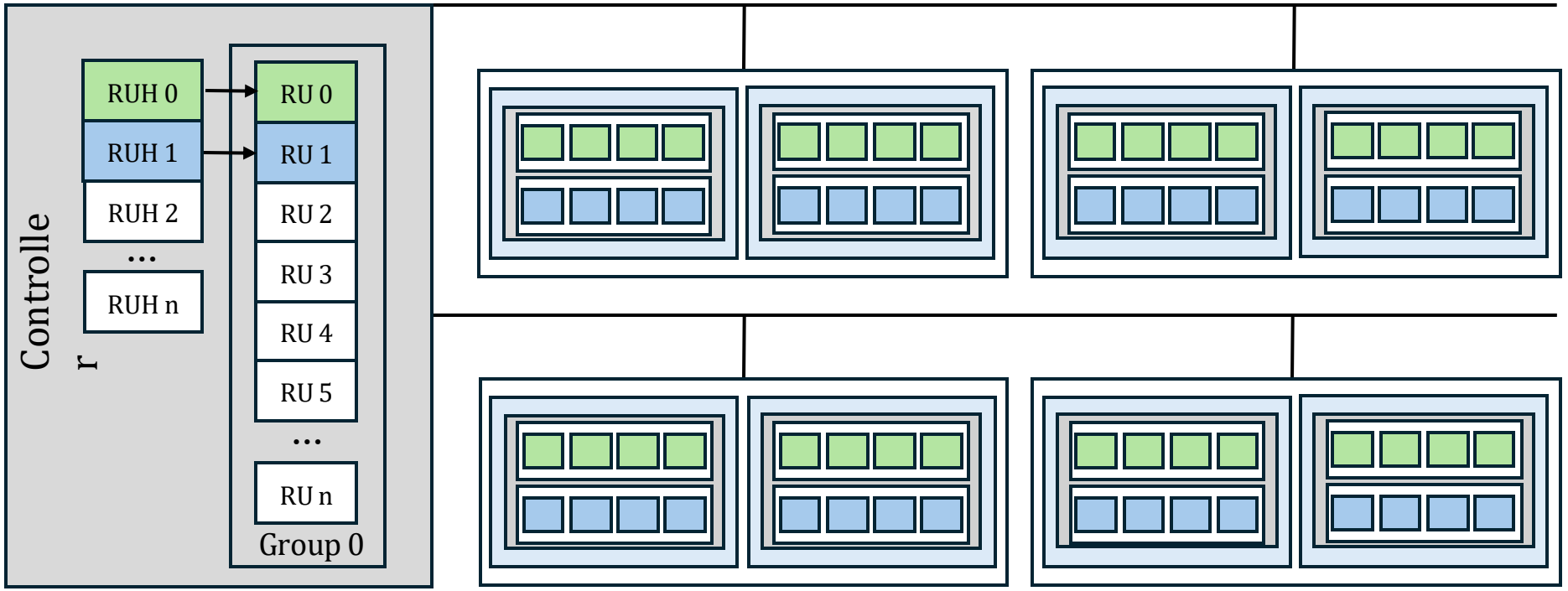
FDP Controller Design: Mapping

Reclaim Unit (RU)



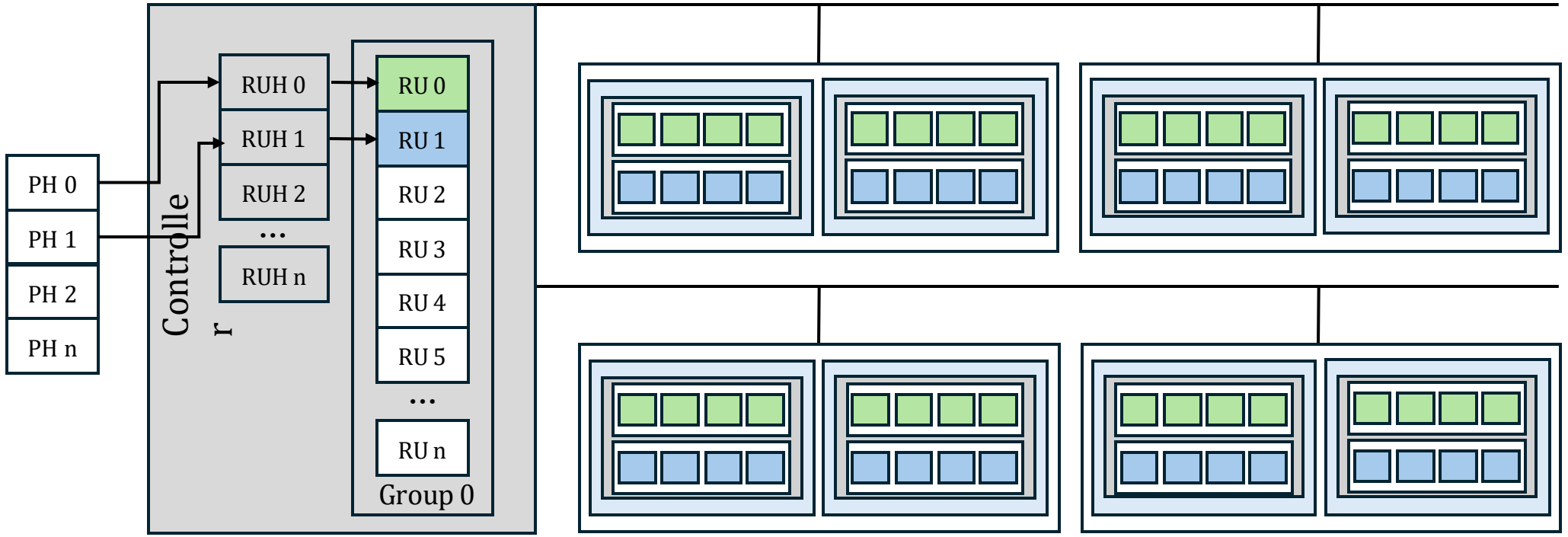
FDP Controller Design: Mapping

Reclaim Unit Handle (RUH)



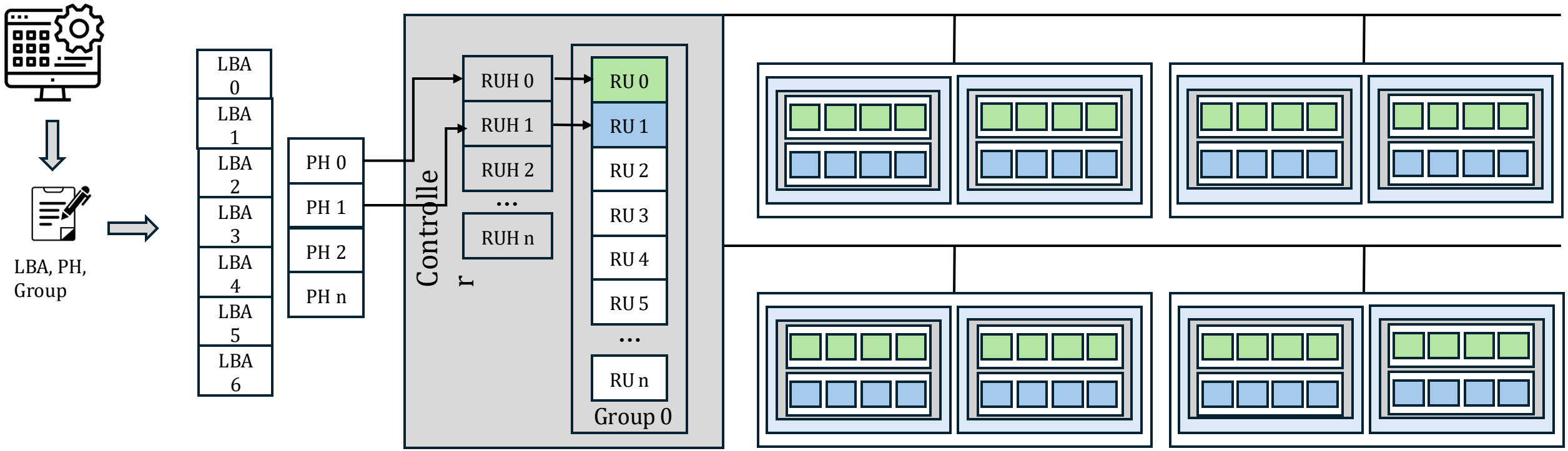
FDP Controller Design: Mapping

Placement Handle (PH)



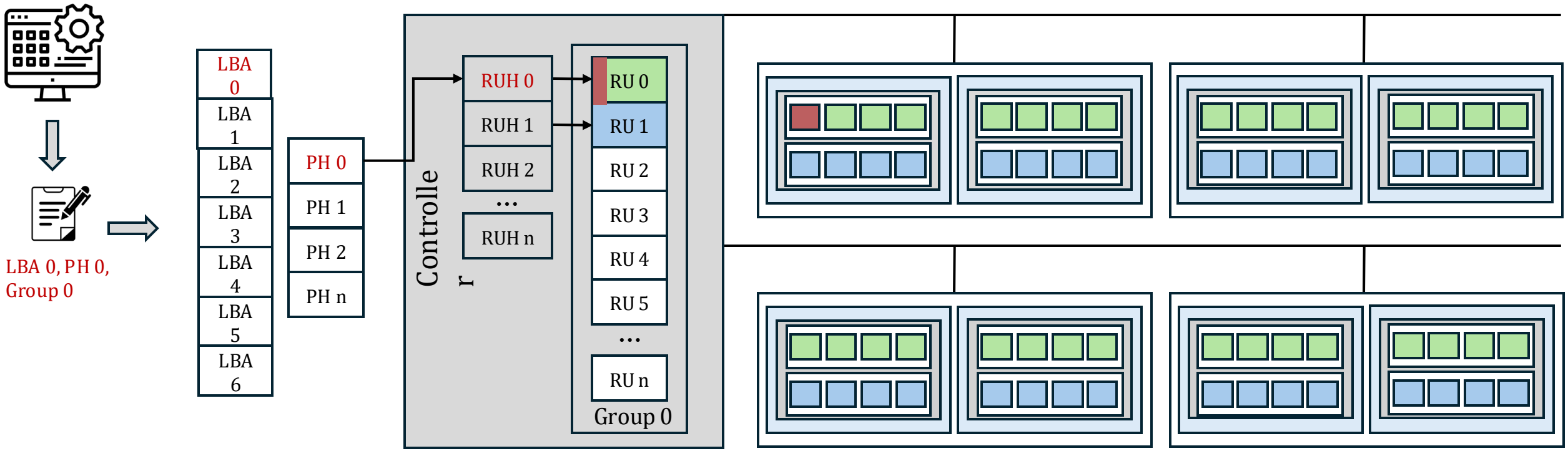
FDP Controller Design: Mapping

Write command has to include LBA, PH and Group



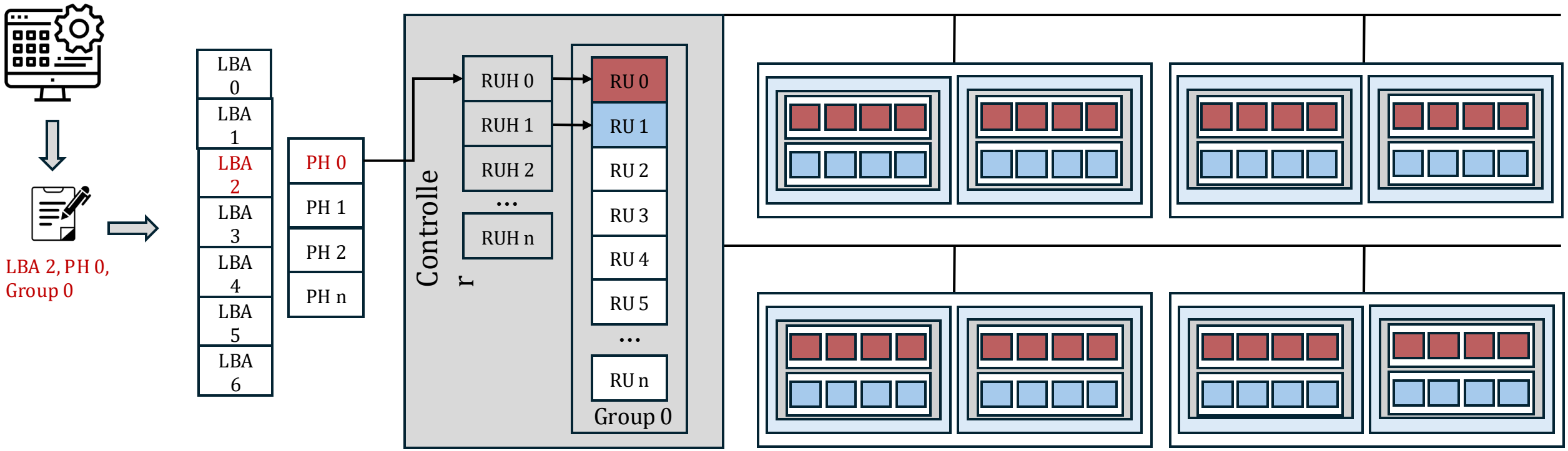
FDP Controller Design: Mapping

Placement Handle (PH)



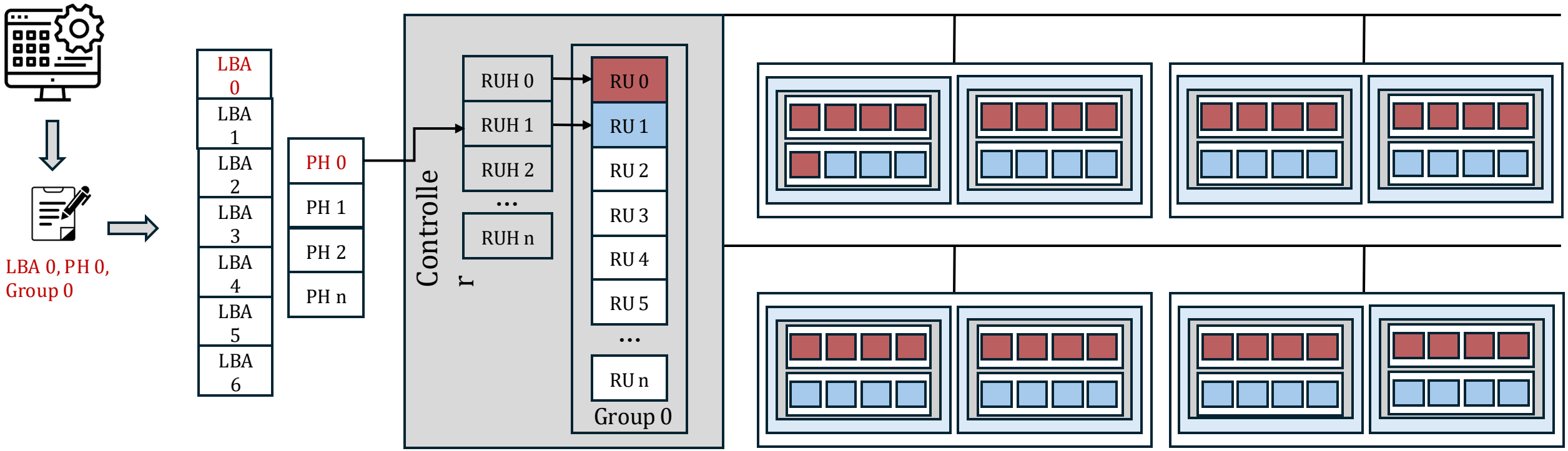
FDP Controller Design: Mapping

Placement Handle (PH)



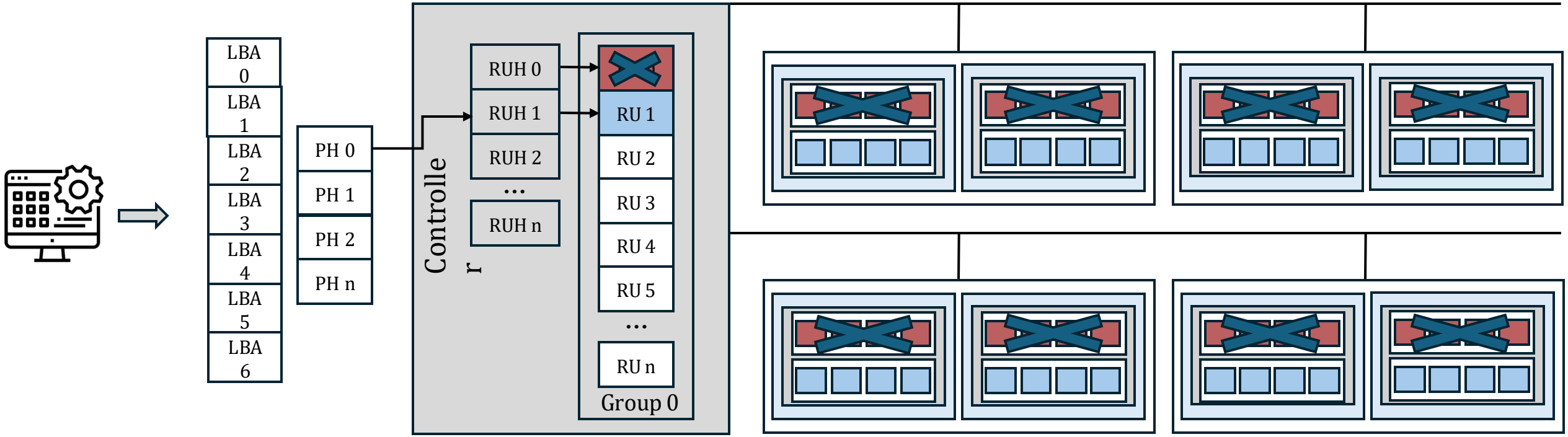
FDP Controller Design: Mapping

Writing beyond RU boundaries



FDP Controller Design: Mapping

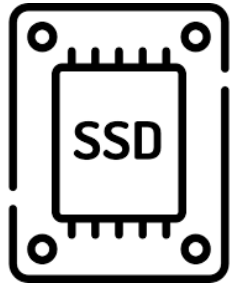
Write amplification benefits



Avoiding Zoned Interface Tax

Append only interface

Garbage collection to reclaim space
State machine management



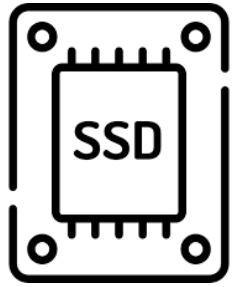
Wear leveling

Optimal data placement

Avoiding Zoned Interface Tax

Append only interface

Garbage collection to reclaim space
State machine management

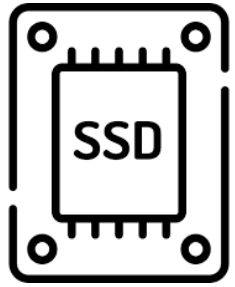


Wear leveling

Optimal data placement

Flexible Data Placement

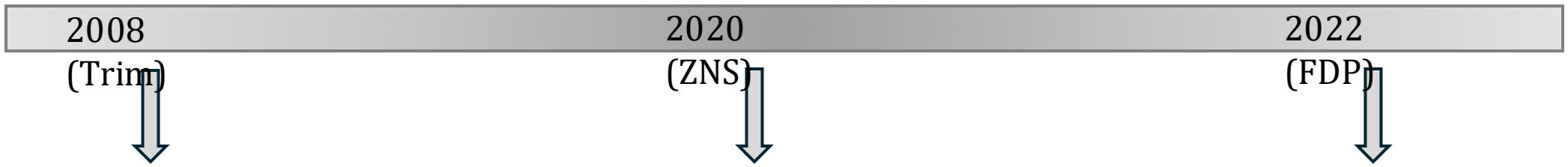
Garbage collection to reclaim space



Wear leveling

Optimal data placement

History of Data Placement on SSD



Black-Box SSD

Block Interface

Over Provisioning

Trim command

Zoned Namespace SSD

Zoned Interface

Zone State Machine

Append-only Zone

No writes beyond boundary

Flexible Data Placement

Block Interface

Random writes

writes beyond boundary

History of Data Placement on SSD