

Improving Phase Change Memory Performance via Data Content Aware Access

DATACON

Anup Das

Shihao Song, Onur Mutlu, and Nagarajan Kandasamy

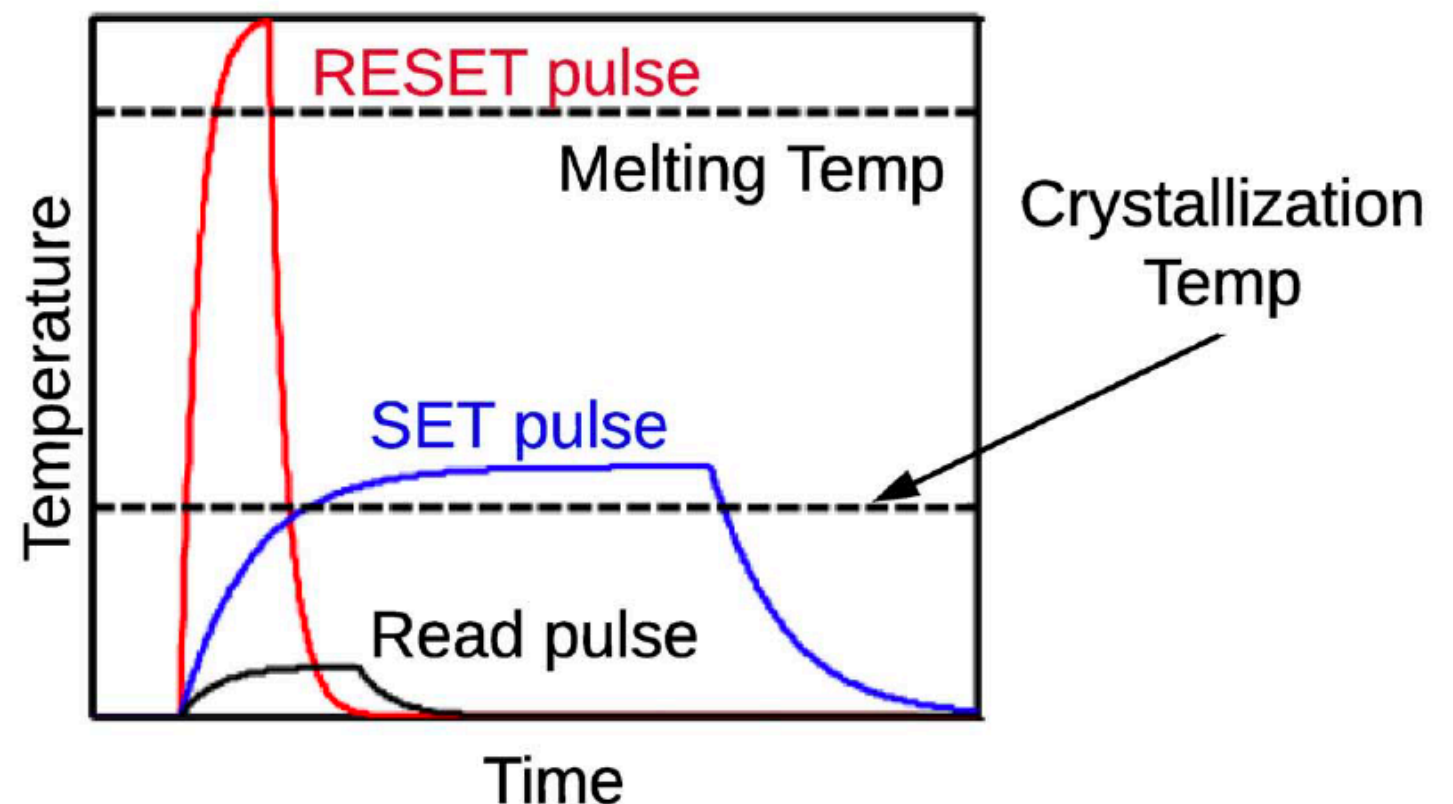
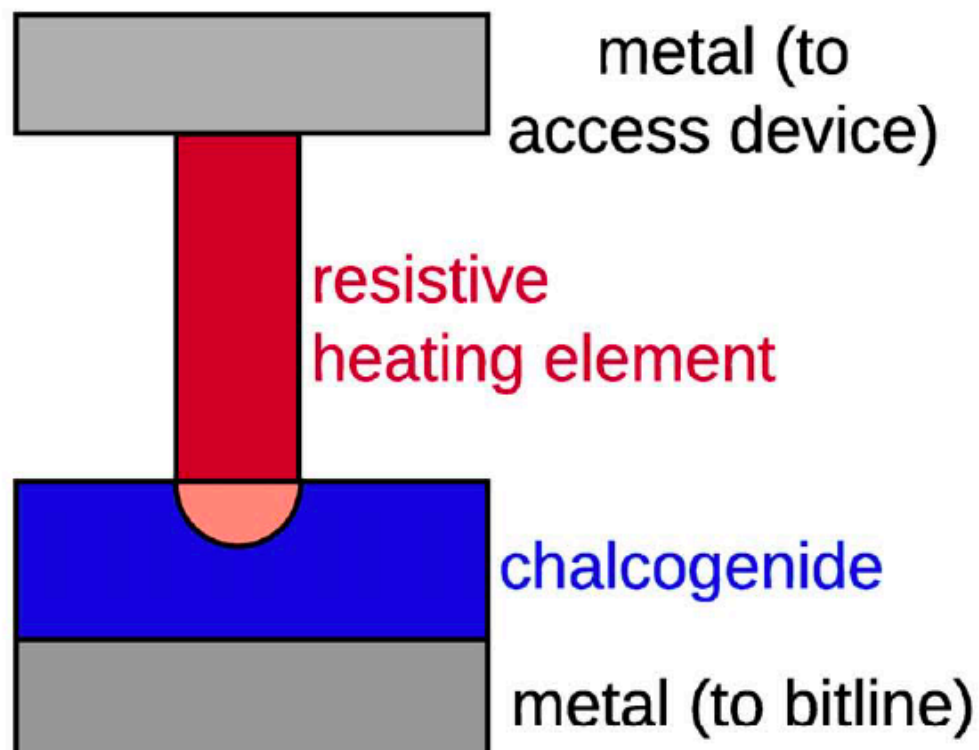


Executive Summary

- Observations
 - During PCM write, latency and energy are sensitive to the *data to be written as well as the content that is overwritten*, which is *unknown* at the time to write
 - Overwriting *known* all-0s or all-1s content improves both latency and energy by programming the PCM cells *only* in one direction, i.e., using SET or RESET operations, not both
- Idea: Data Content Aware PCM Writes
 - Overwrite unknown content *only when necessary*, and otherwise overwrite all-0s and all-1s content ➡ reduces latency and energy
- Two Key Mechanisms
 - **Address translation**: to translate the write address to a physical address within memory that contains the best type of content to overwrite
 - **Re-Initialization**: to methodically re-initialize *unused* memory locations with known all-0s and all-1s content in a manner that does not interfere with regular read and write accesses
- Performance Evaluation
 - Significant **improvement of system performance (27%) and reduction of energy (43%)** for SPEC CPU2017, NAS Parallel, and Tensorflow Benchmarks

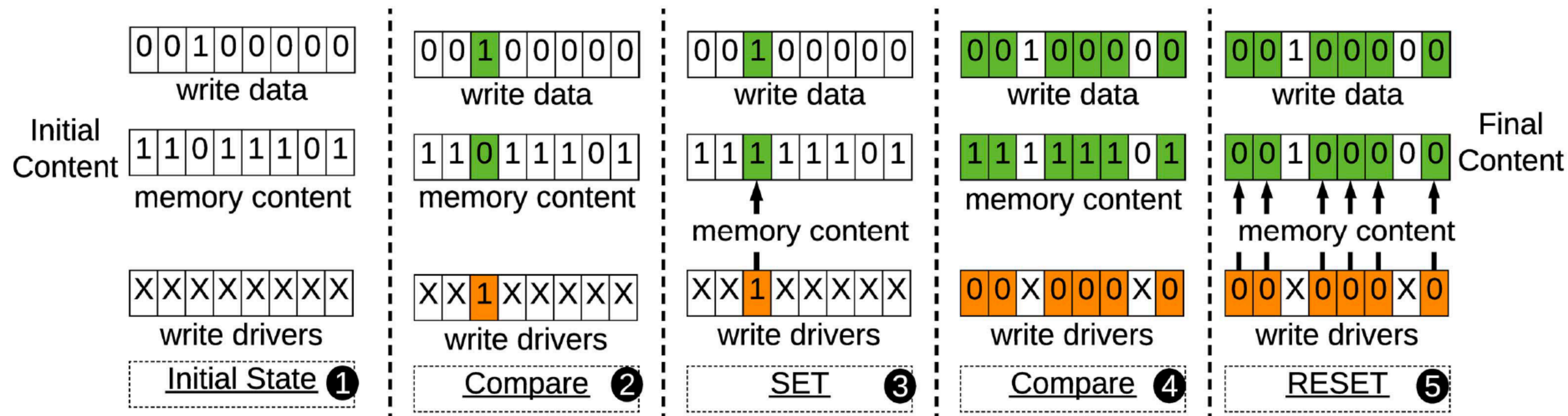
Asymmetry in PCM SET and RESET Operations

- PCM stores data as resistance of its cells
- RESET operation (1 \rightarrow 0): **fast** and **high energy**
- SET operation (0 \rightarrow 1): **slow** and **low energy**



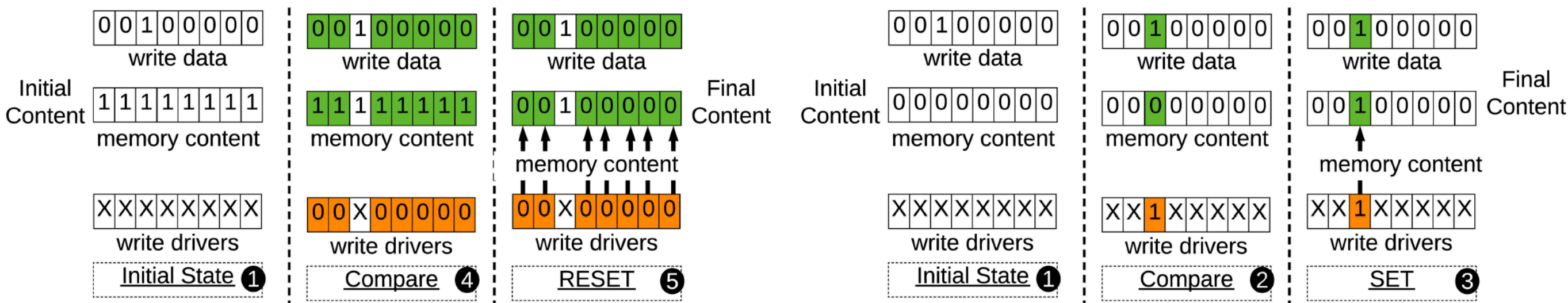
Baseline PCM Write Operation

- Use **both** SET and RESET operations
- Program and Verify (P&V) Scheme



PreSET and PreRESET

- Use **either** SET and RESET operations, not both
- PreSET or PreRESET is decided at **design-time** and used for *all* PCM writes, independent of the write data



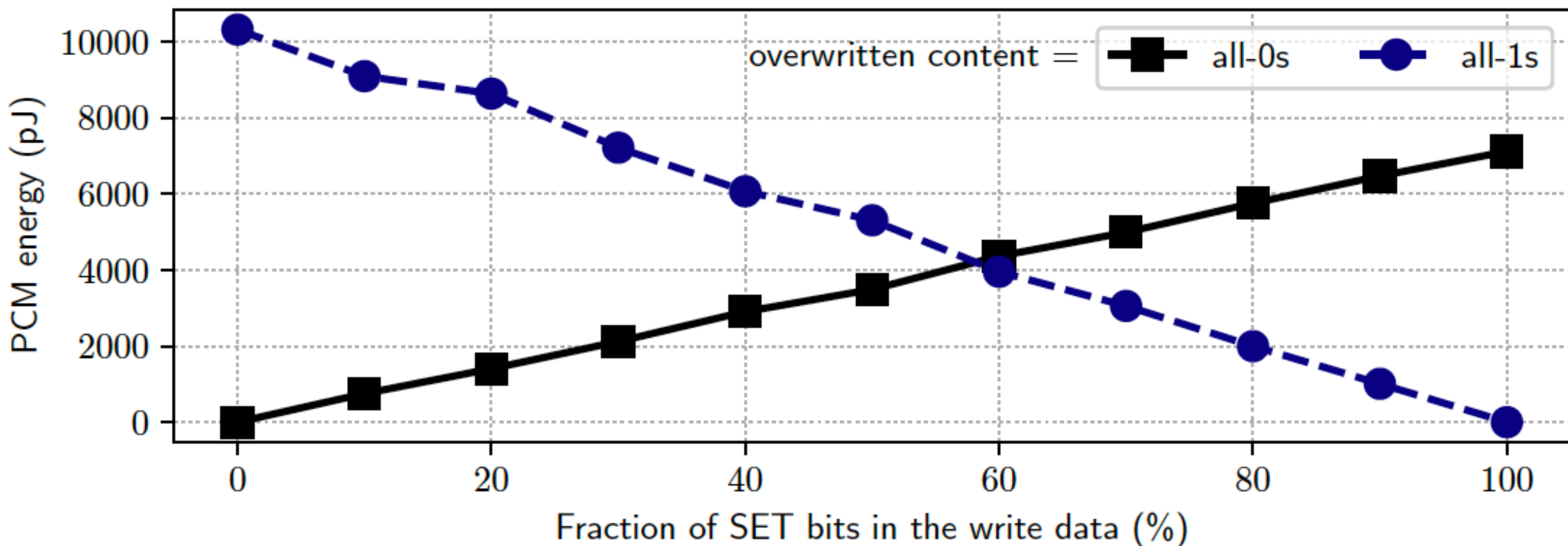
PreSET: Overwrite all-1s

PreRESET: Overwrite all-0s

Qureshi et al., “PreSET: Improving performance of phase change memories by exploiting asymmetry in write times,” in ISCA, 2012

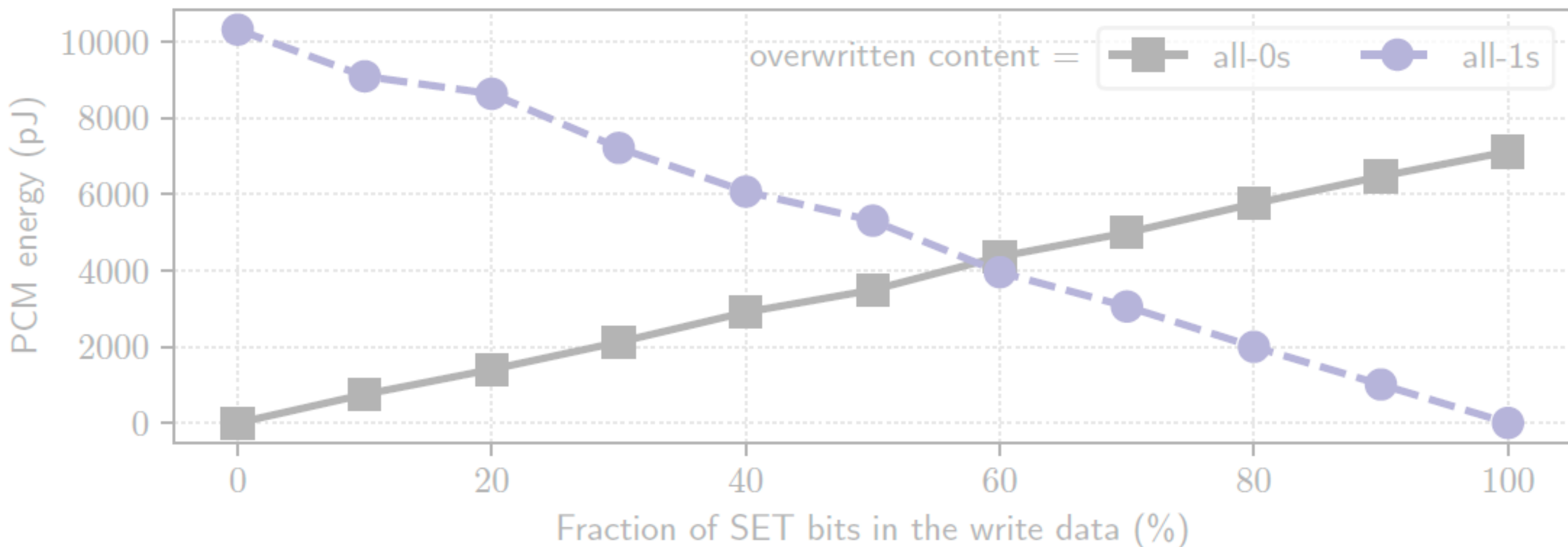
PCM Write Related Key Observations (I)

- Energy benefit of PreSET vs. PreRESET depends on the **fraction of SET bits in the write data**



PCM Write Related Key Observations (I)

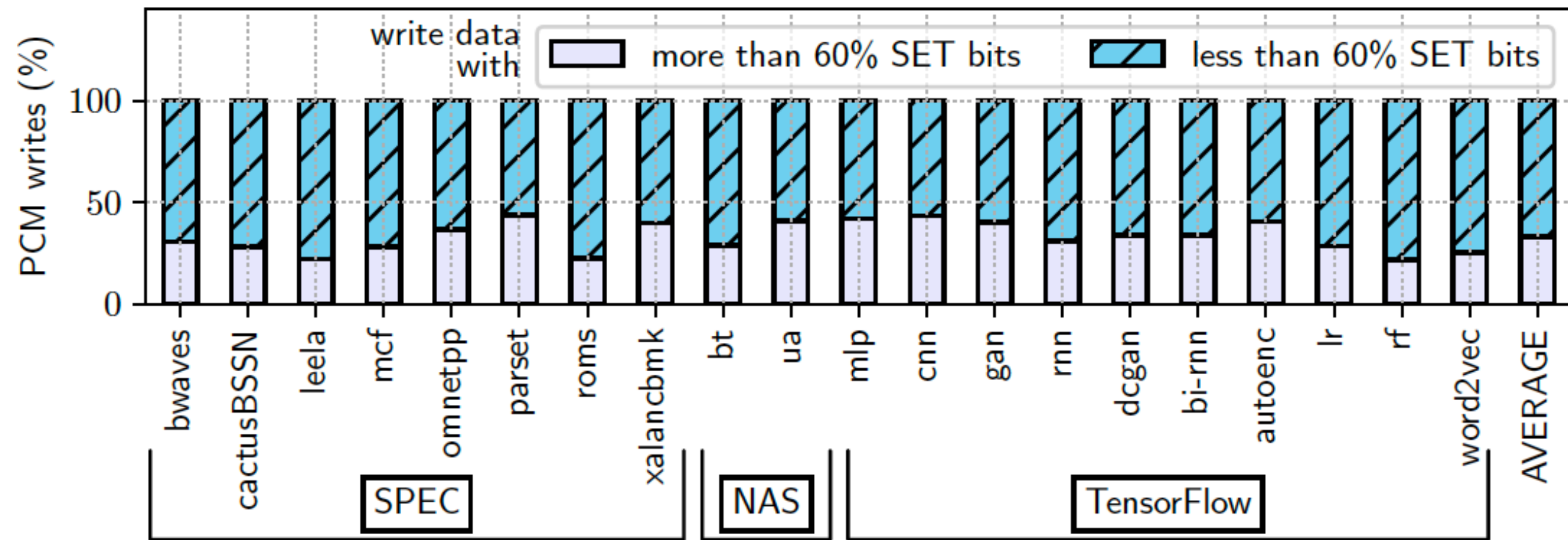
- Energy benefit of PreSET vs. PreRESET depends on the **fraction of SET bits in the write data**



Idea: Use PreRESET when the fractions of SET bits in the write data is **less than 60%**, otherwise use PreSET

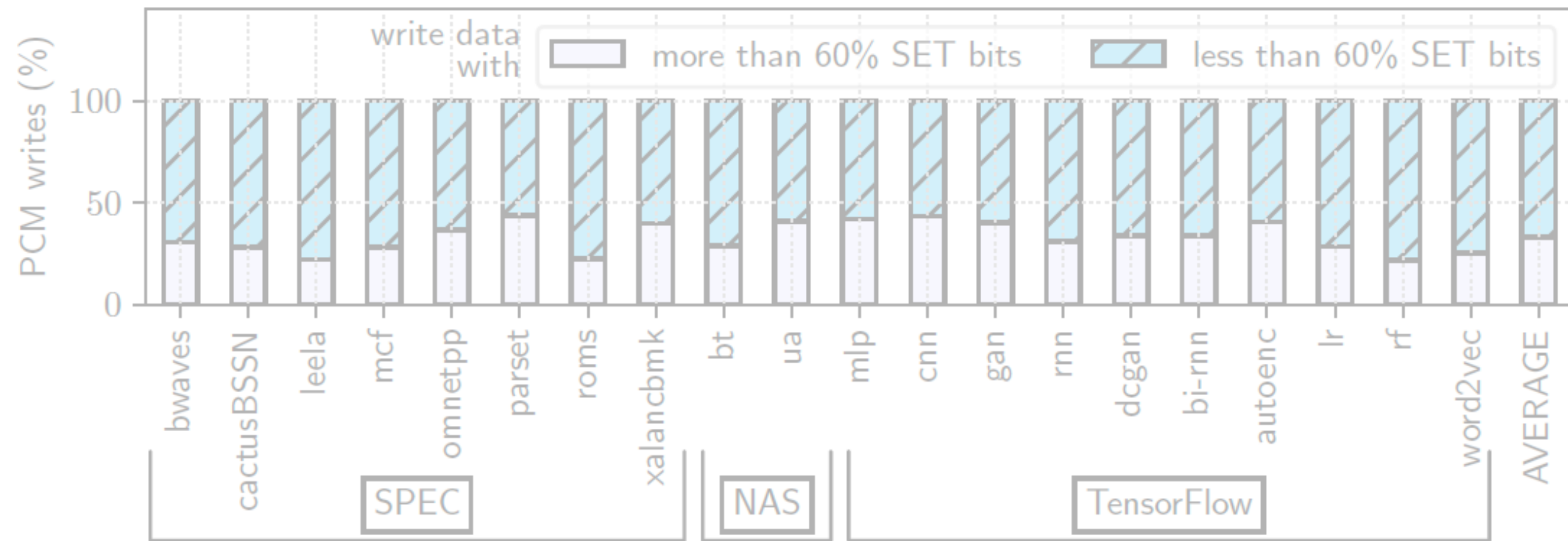
PCM Write Related Key Observations (II)

- The fraction of SET bits in the write data is **workload dependent**



PCM Write Related Key Observations (II)

- The fraction of SET bits in the write data is **workload dependent**



Idea: Re-initialize **unused** memory locations with all-0s or all-1s based on the the PCM writes in a workload

Data Content Aware PCM Writes

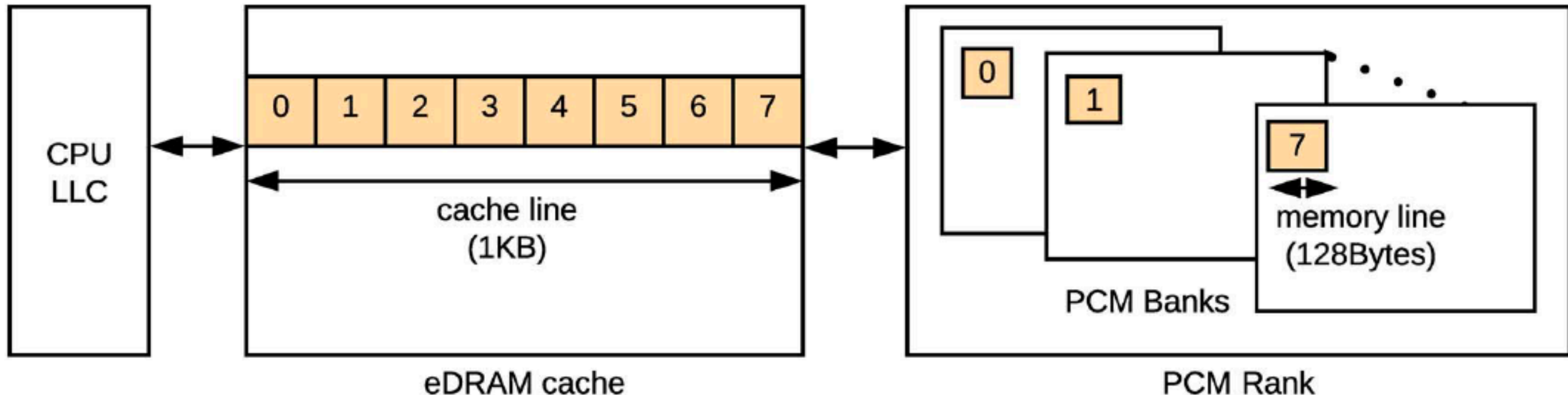
- Key Idea
 - To serve PCM writes
 - ▶ Overwrite known all-0s or all-1s content: to improve performance and energy
 - ▶ Overwrite unknown content: only when is no all-0s or all-1s initialized location available in PCM
- Three components
 - Analysis of write data
 - ▶ Analyze the fraction of SET bits in write data and estimate the energy-latency trade-offs for SET and RESET operations in PCM
 - Address translation
 - ▶ Translate the write address to a physical address within memory that contains the best type of content to overwrite, and record this translation in a table for future accesses.
 - Re-initialization
 - ▶ Re-initialize unused memory locations with known all-0s or all-ones content in a manner that does not interfere with regular read and write accesses

Outline

- Introduction
- Detailed design of DATACON
- Address translation
- Overwritten content selection
- Re-initialization
- Evaluation
- Conclusion

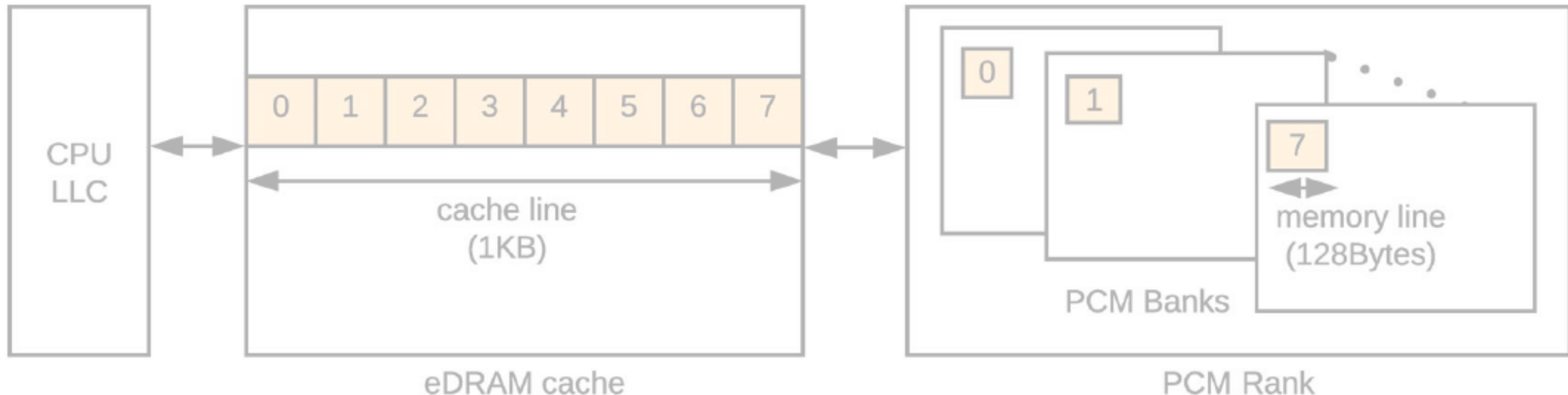
System Overview

- DRAM-PCM **hybrid memory** with embedded DRAM (eDRAM) as write cache to PCM main memory



System Overview

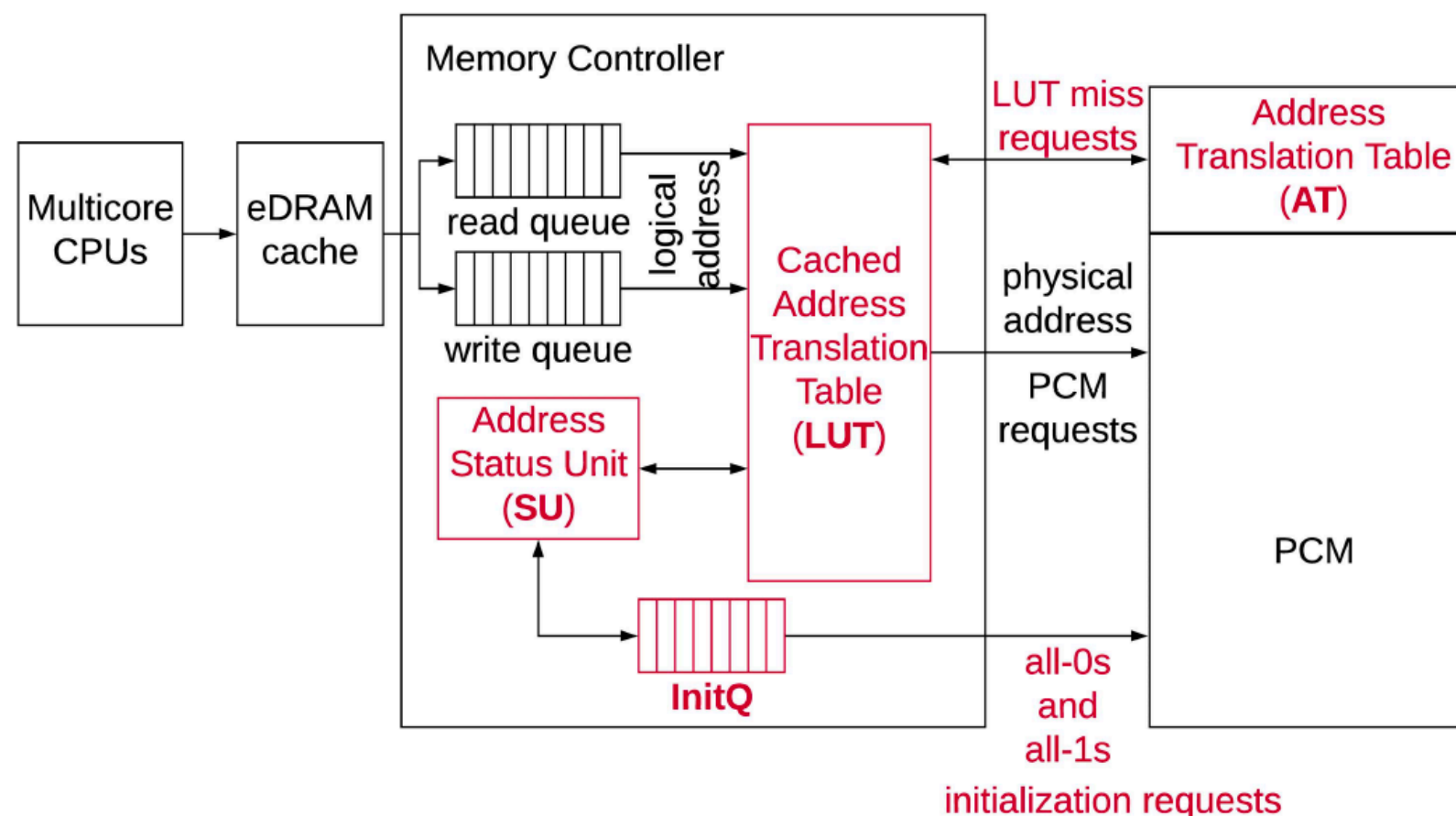
- DRAM-PCM **hybrid memory** with embedded DRAM (eDRAM) as write cache to PCM main memory



One eDRAM cache line maps to **eight** memory lines in a PCM rank

Detailed Design of DATACON

- DATACON adds **four** new components to the baseline design
 - **Address Translation Table (AT)**: to record logical-to-physical address translations, which are needed to redirect write requests to the best overwritten content in PCM
 - **Lookup Table (LUT)**: to cache recently-used address translation information in the memory controller
 - **Address Status Unit (SU)**: to select an all-0s or all-1s initialized physical address for the logical address of a write request
 - **Initialization Queue (InitQ)**: to record unused physical locations in PCM, such that they can be re-initialized methodically



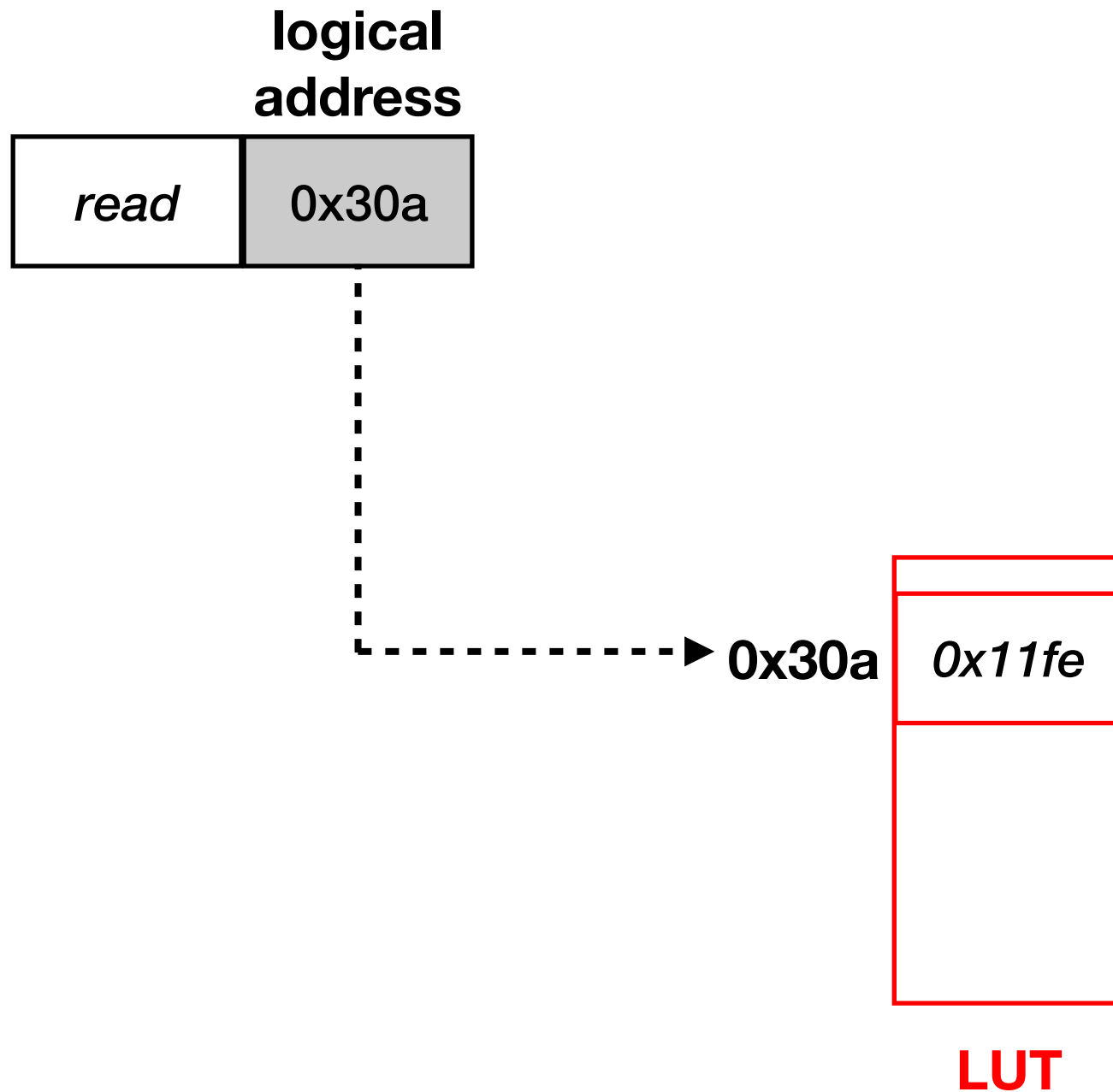
Outline

- Introduction
- Detailed design of DATACON
- **Address translation**
- Overwritten content selection
- Re-initialization
- Evaluation
- Conclusion

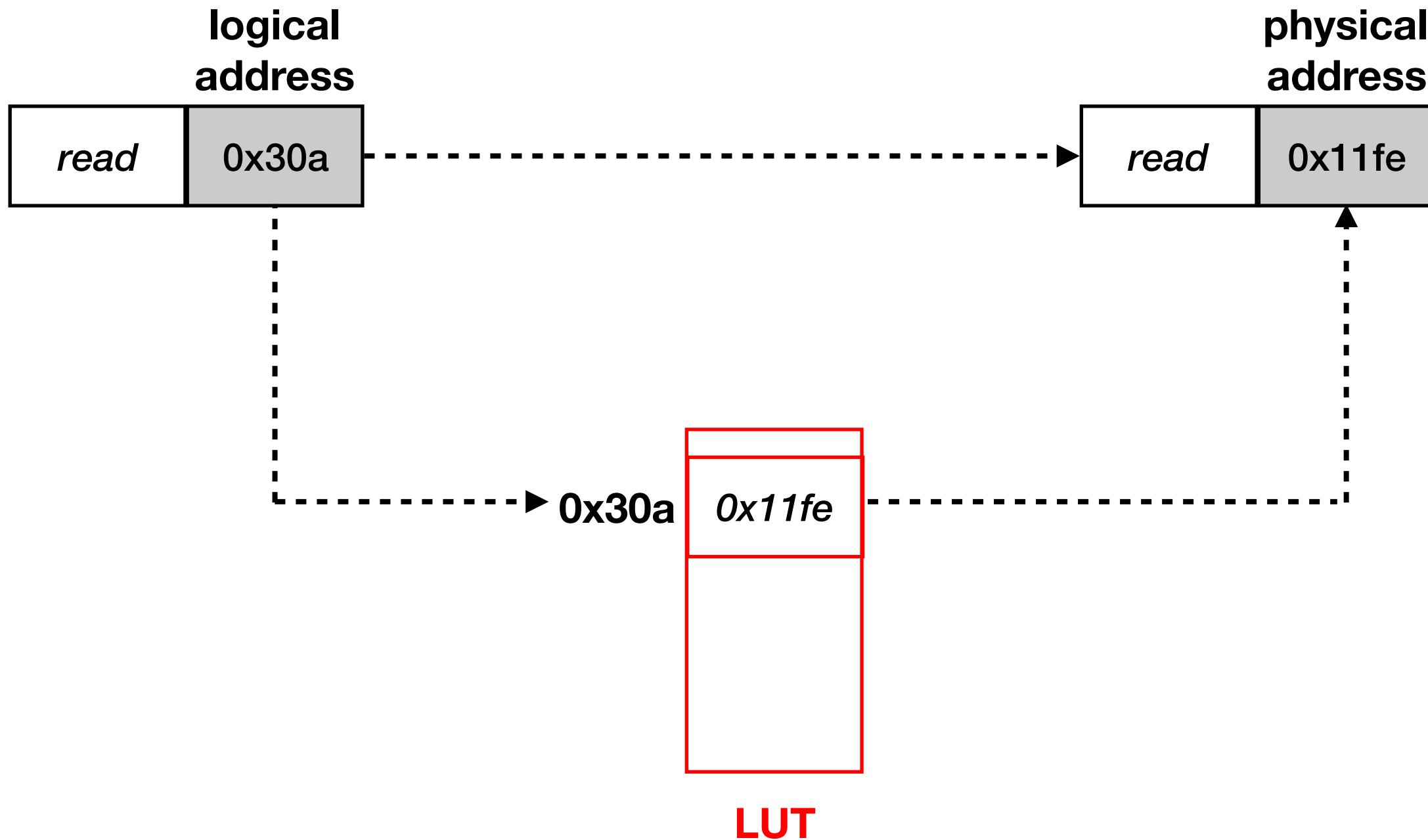
Address Translation (I): Read

logical address	
<i>read</i>	0x30a

Address Translation (I): Read



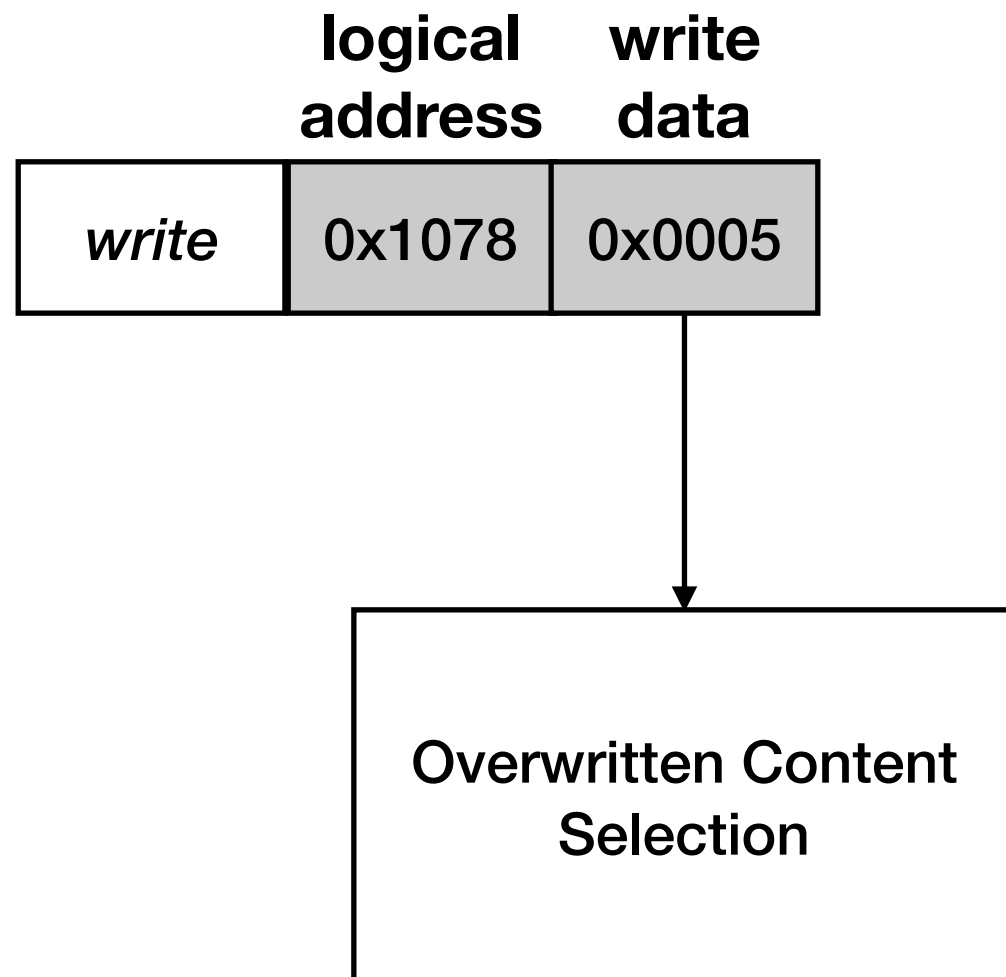
Address Translation (I): Read



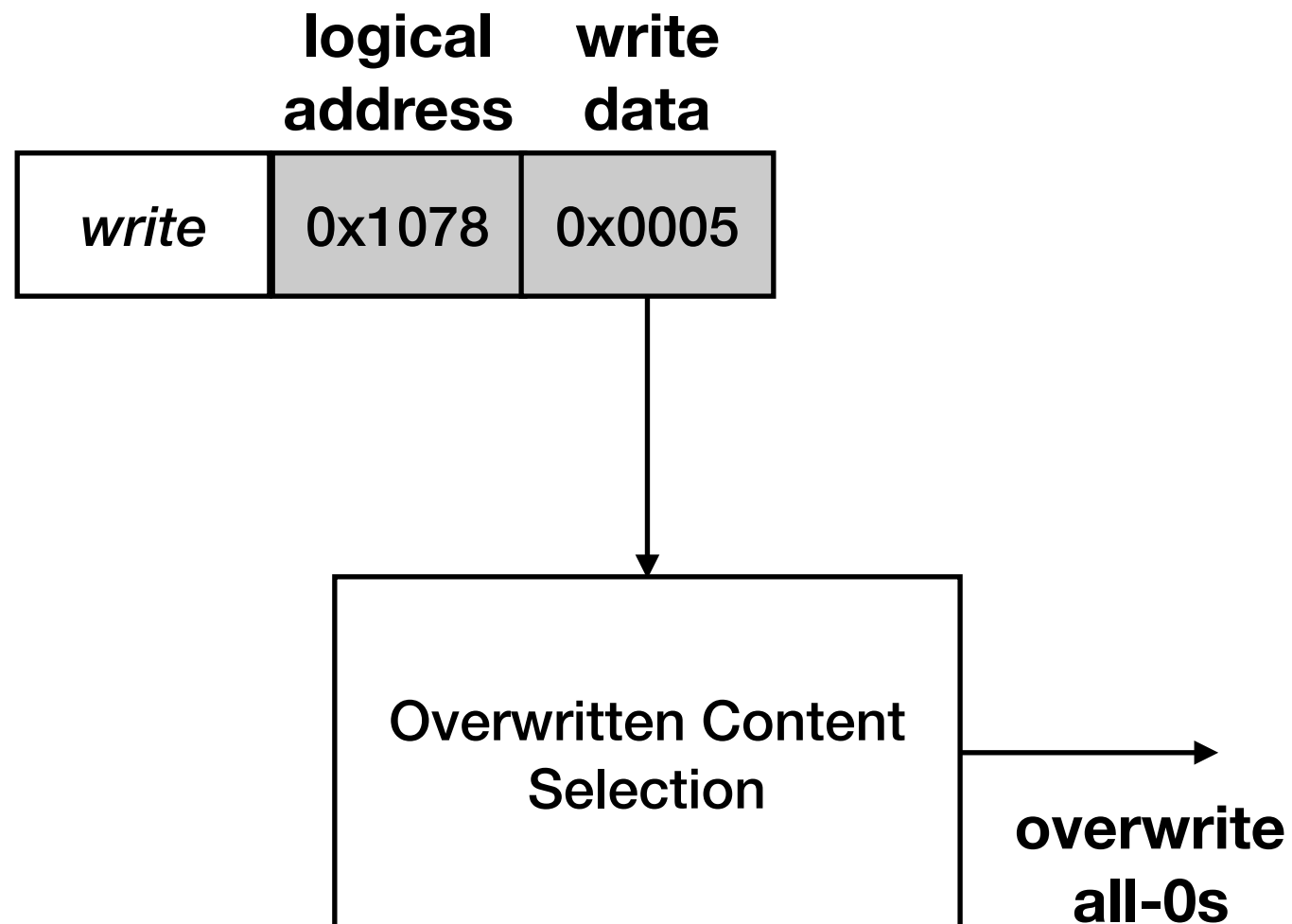
Address Translation (II): Write

	logical address	write data
<i>write</i>	0x1078	0x0005

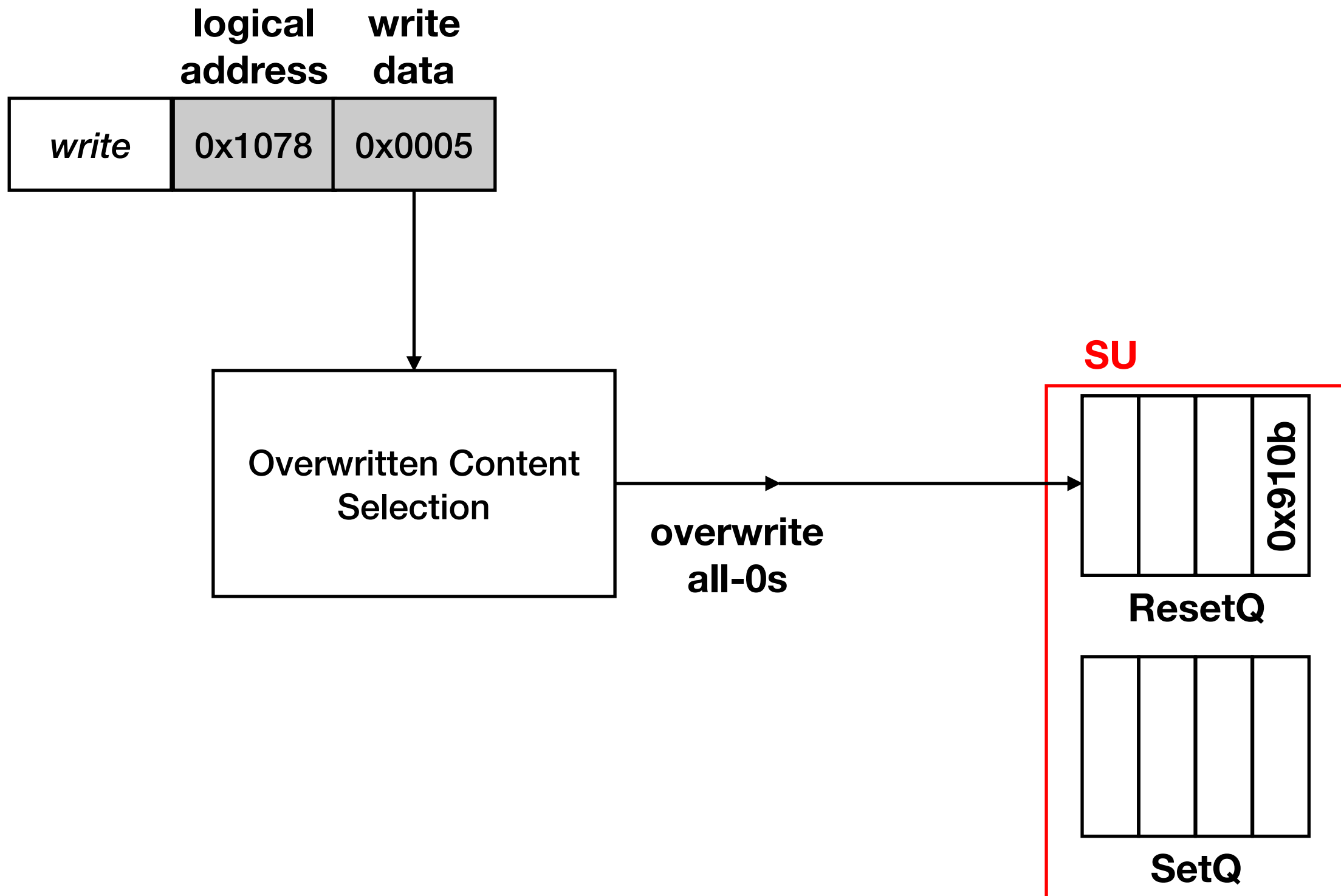
Address Translation (II): Write



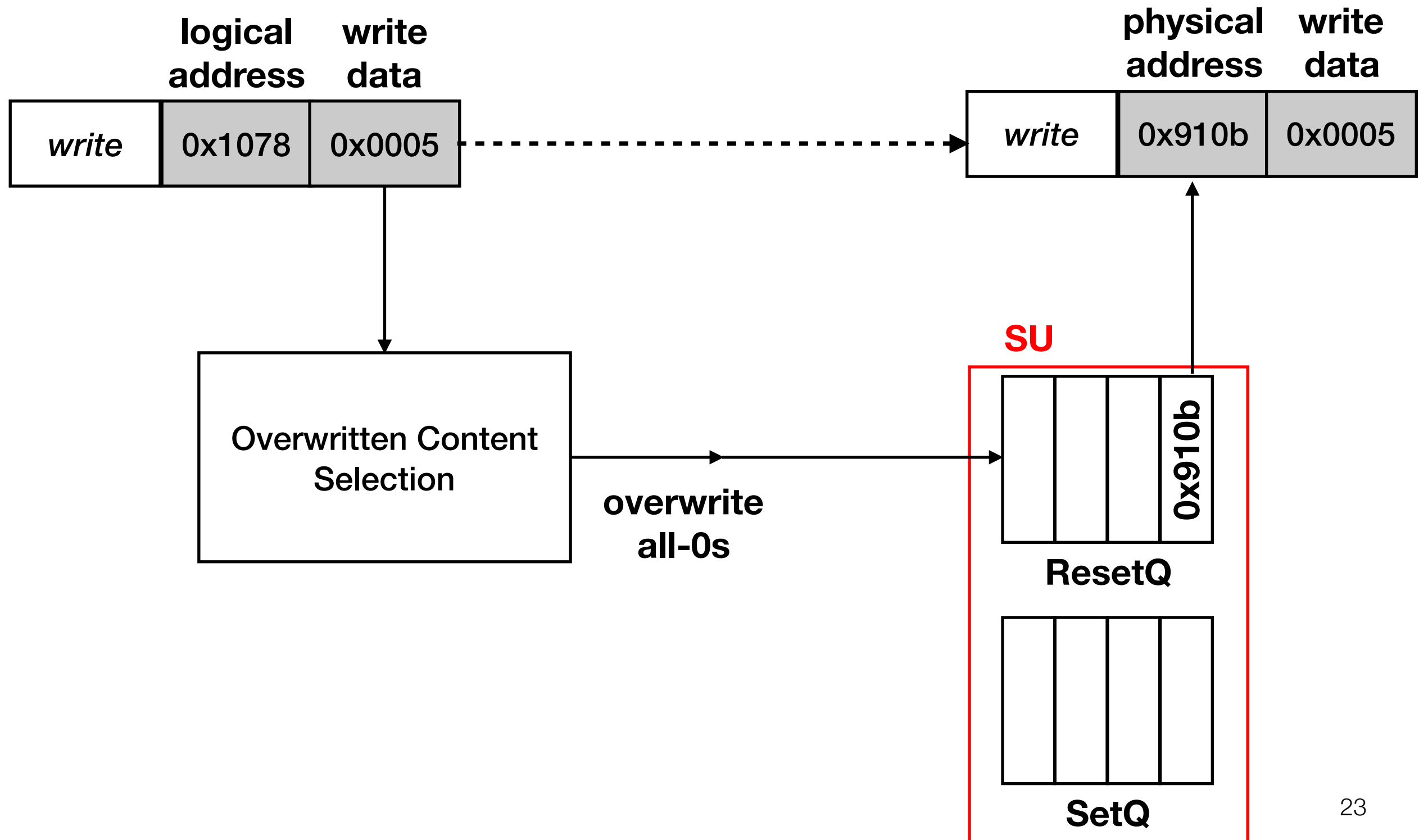
Address Translation (II): Write



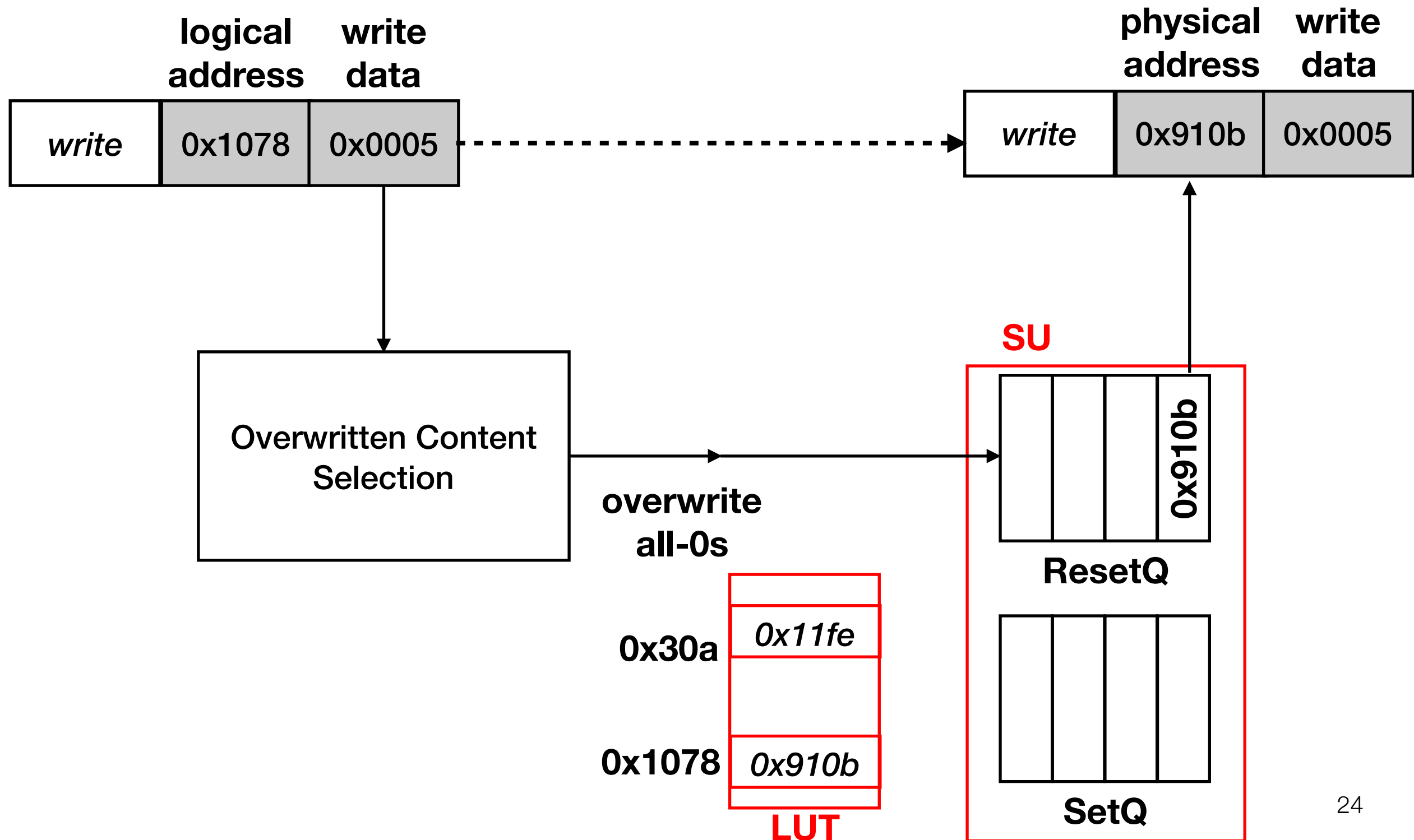
Address Translation (II): Write



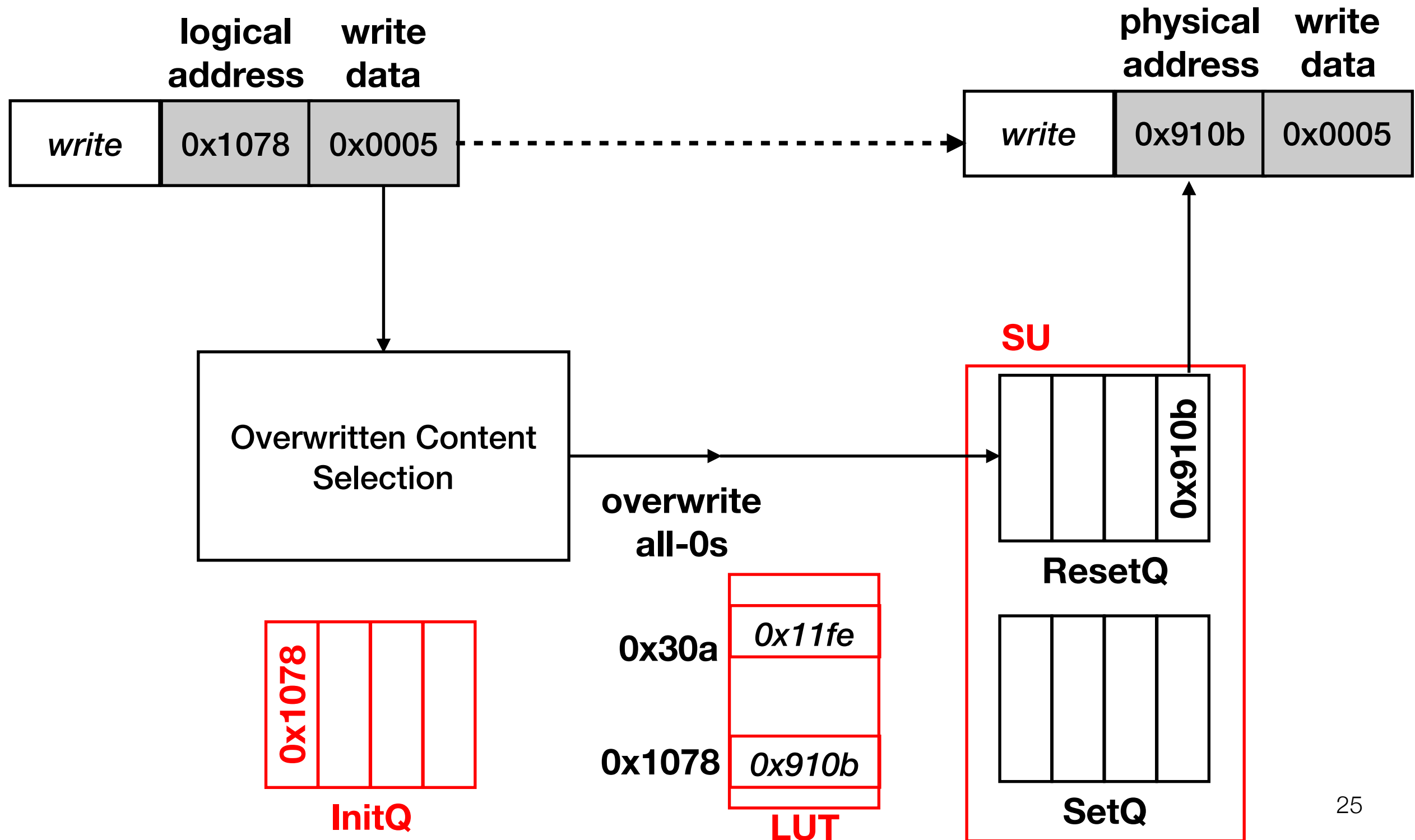
Address Translation (II): Write



Address Translation (II): Write



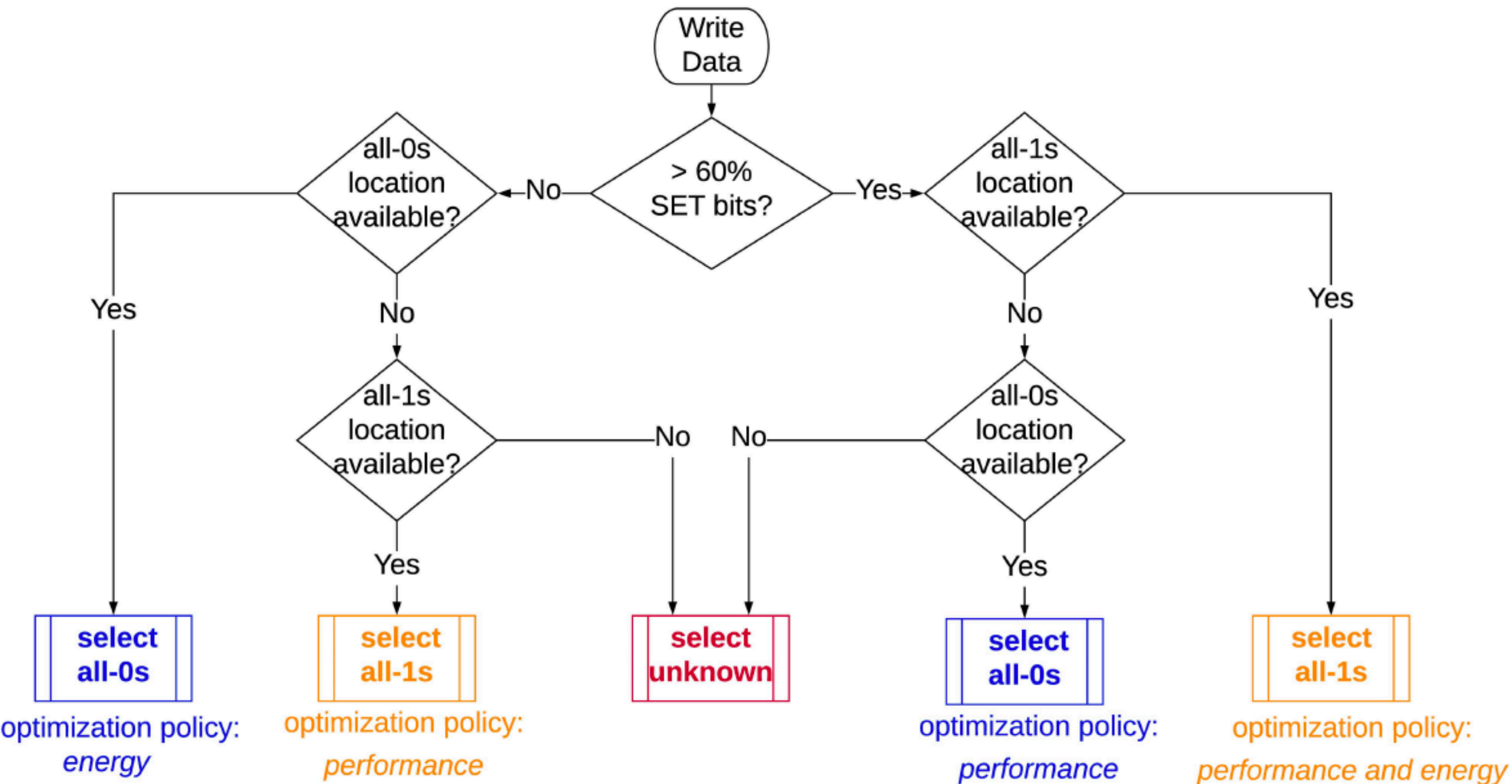
Address Translation (II): Write



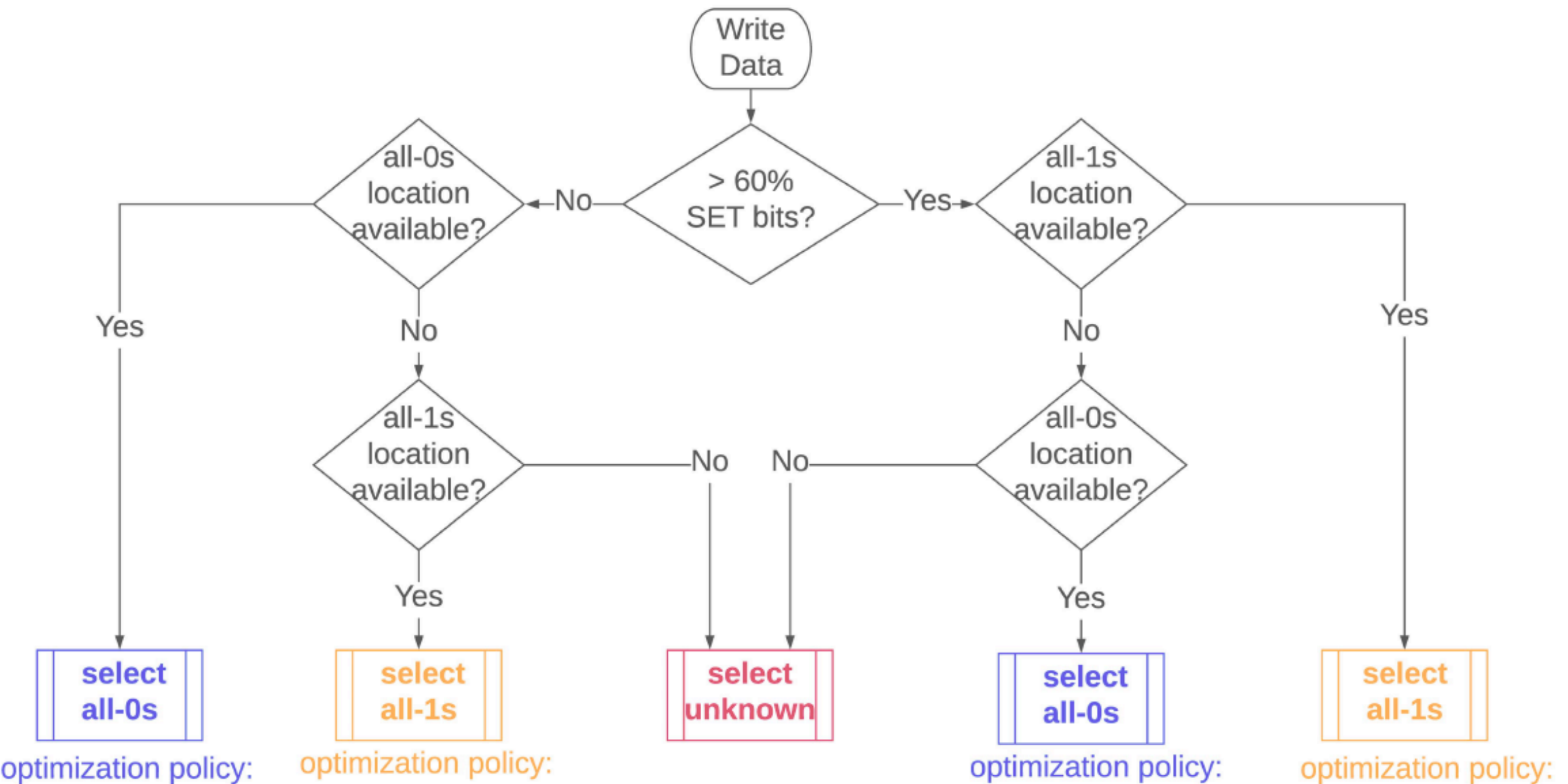
Outline

- Introduction
- Detailed design of DATACON
- Address translation
- **Overwritten content selection**
- Re-initialization
- Evaluation
- Conclusion

Overwritten Content Selection



Overwritten Content Selection



DATACON overwrites unknown content only when there is **no** all-0s or all-1s initialized content available in PCM

Outline

- Introduction
- Detailed design of DATACON
- Address translation
- Overwritten content selection
- **Re-initialization**
- Evaluation
- Conclusion

Re-initialization Requests

- Two queues to record the address of all-0s and all-1s content in PCM
 - SetQ and ResetQ
- Thresholding to initiate re-initialization requests
 - Re-initialization requests stored in the InitQ
- Re-initialization requests serviced when
 - Read and the write queues are **both** empty, or
 - Write queue is empty and read request is to a memory partition **different** from the re-initialization request
 - ▶ Exploit's PCM's partition-level parallelism (PALP)

Song et al., “Enabling and Exploiting Partition-Level Parallelism (PALP) in Phase Change Memories,” in CASES, 2019

Outline

- Introduction
- Detailed design of DATACON
- Address translation
- Overwritten content selection
- Re-initialization
- **Evaluation**
- Conclusion

System Configuration

- A cycle-level in-house x86 multi-core simulator, whose front-end is based on Pin
 - To simulate 8 out-of-order cores
- A main memory simulator, closely matching the JEDEC Nonvolatile Dual In-line Memory Module (NVDIMM)- N/F/P Specifications
 - [Ramulator](#) to simulate DRAM
 - [NVMain](#) to simulate PCM

[Ramulator](#): Kim et al. “*Ramulator: A fast and extensible DRAM simulator*” CAL, 2016.

[NVMain](#): Poremba et al. “*Nvmain 2.0: A user-friendly memory simulator to model (non-) volatile memory systems*” CAL, 2015

Simulation Parameters

Processor	8 cores per socket, 3.32 GHz, out-of-order
L1 cache	Private 32KB per core, 8-way
L2 cache	Private 512KB per core, 8-way
L3 cache	Shared 8MB, 16-way
eDRAM cache	Shared 64MB per socket, 16-way, on-chip
	128GB PCM.
Main memory	4 channels, 4 ranks/channel, 8 banks/rank, 8 partitions/bank, 128 tiles/partition, 4096 rows/tile. Memory interface = DDR4 Memory clock = 1066MHz PCM Timings = See Table 1

Simulation Parameters

Baseline Timing Parameters

	tRCD	tRAS	tRP	tRC	
Read	3.75ns	55.25ns	1ns	56.25ns	
	tRCD	tBURST	tWR	tRP	tRC
Write	75ns	15ns	190ns	1ns	209.75ns

DATACON Timing Parameters

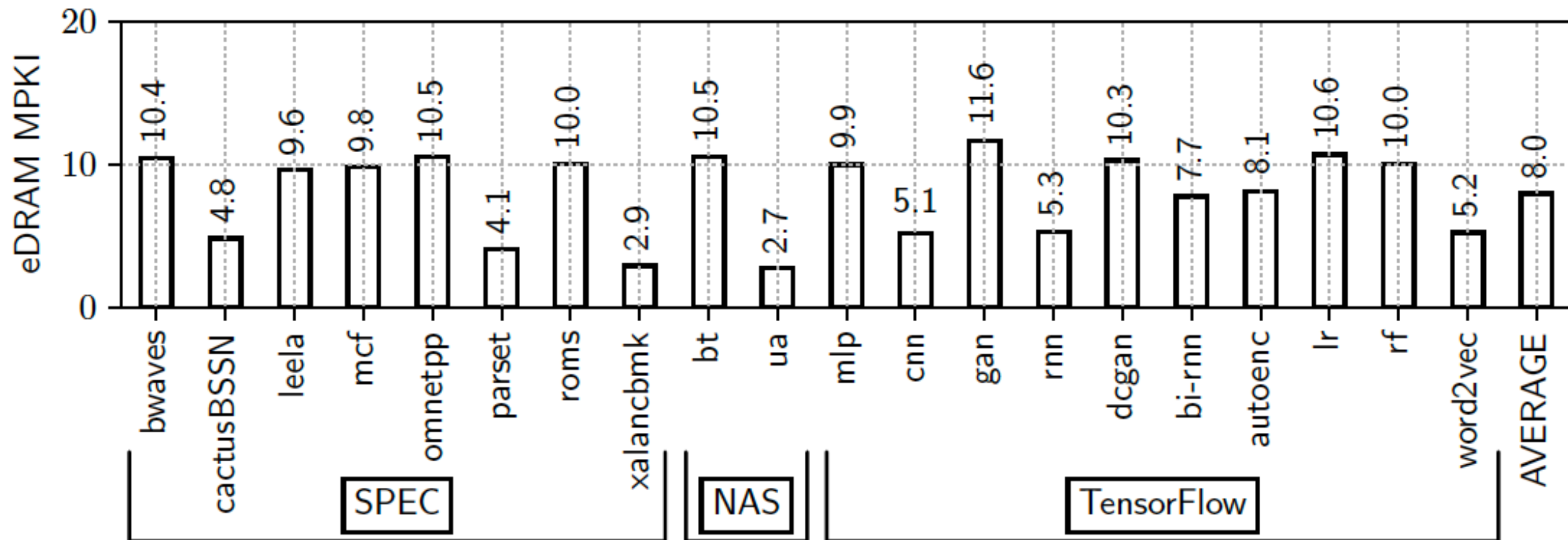
	tRCD	tRAS	tRP	tRC	
Read	3.75ns	55.25ns	1ns	56.25ns	
	tRCD	tBURST	tWR	tRP	tRC
SET (all-0s)	3.75ns	15ns	150ns	1ns	169.75ns
RESET (all-1s)	3.75ns	15ns	40ns	1ns	59.75ns
Write (unknown)	75ns	15ns	190ns	1ns	209.75ns

Evaluated Systems

- **Baseline**: services PCM writes by overwriting *unknown* content
 - Lee et al., “Architecting phase change memory as a scalable DRAM alternative,” in ISCA, 2009
- **PreSET**: services PCM writes by *always* overwriting all-1s
 - Qureshi et al., “PreSET: Improving performance of phase change memories by exploiting asymmetry in write times,” in ISCA, 2012
- **Flip-N-Write**: services PCM writes by *finding out* the memory content using additional reads and programming only bits that are different from the write data
 - Cho et al., “Flip-N-Write: a simple deterministic technique to improve PRAM write performance, energy and endurance,” in MICRO, 2009

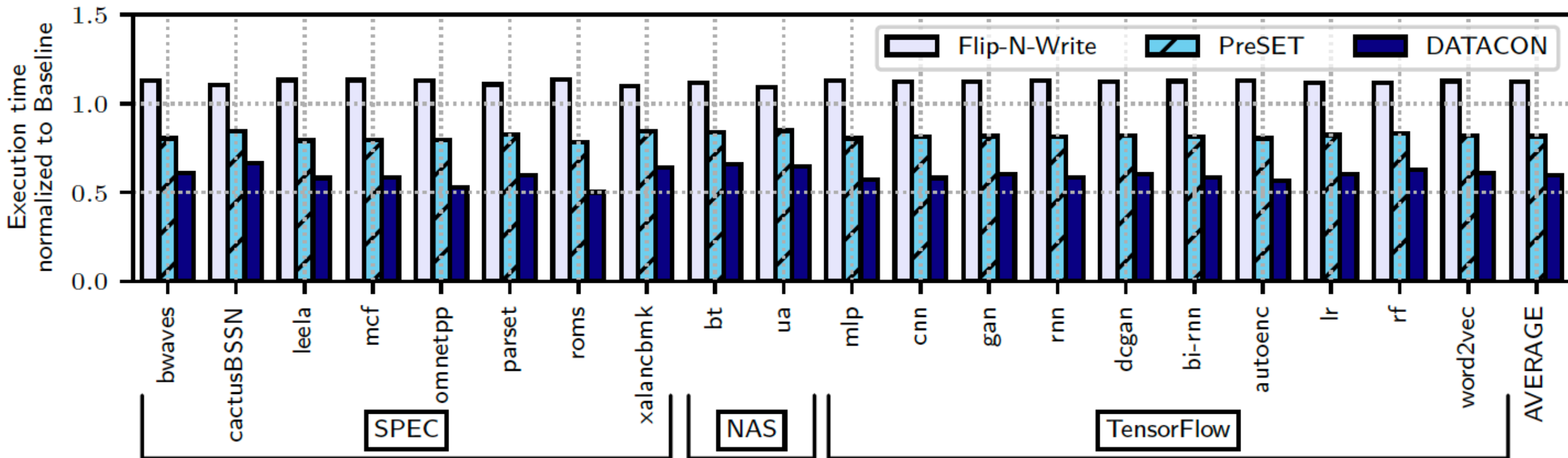
Workloads

- SPEC CPU2017
- NAS Parallel Benchmarks
- Tensorflow workloads

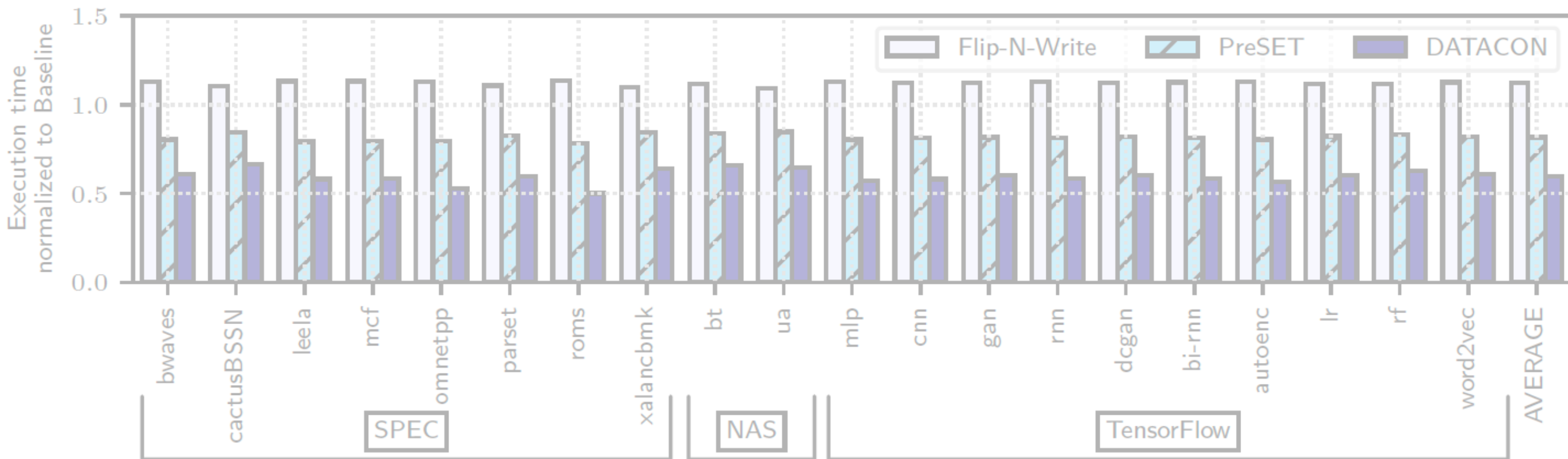


Overall System Performance

- Execution time normalized to Baseline



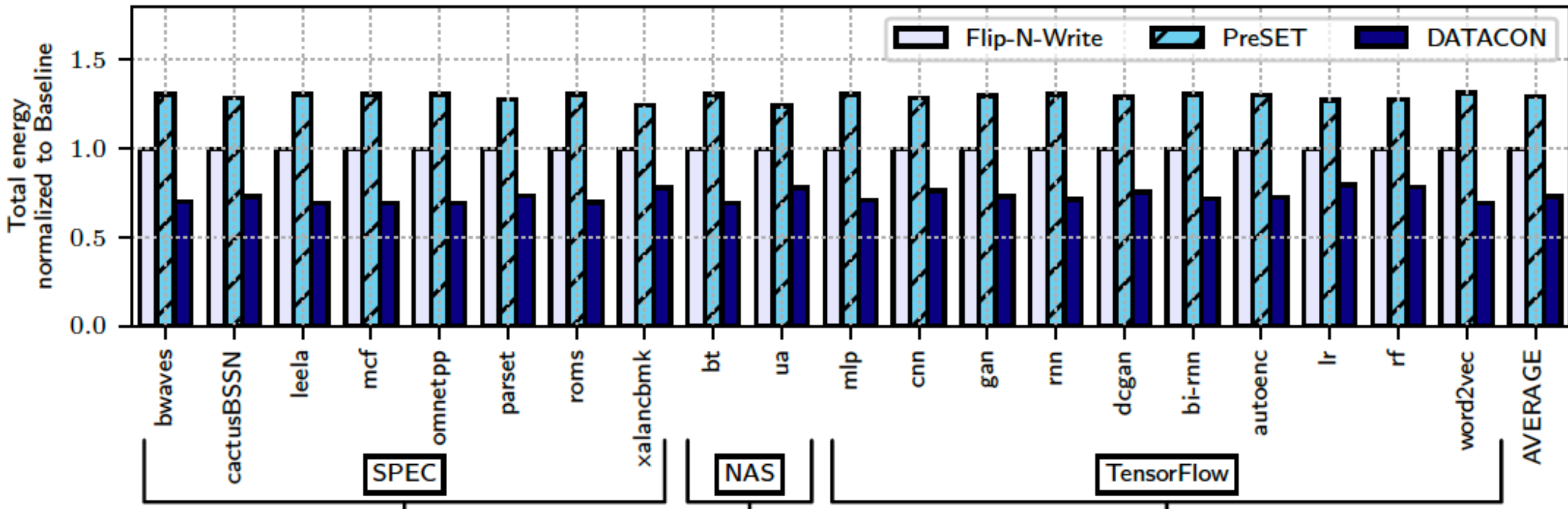
Overall System Performance



Average Execution time of DATACON is
40% lower than Baseline,
47% lower than Flip-N-Write, and
27% lower than PreSET

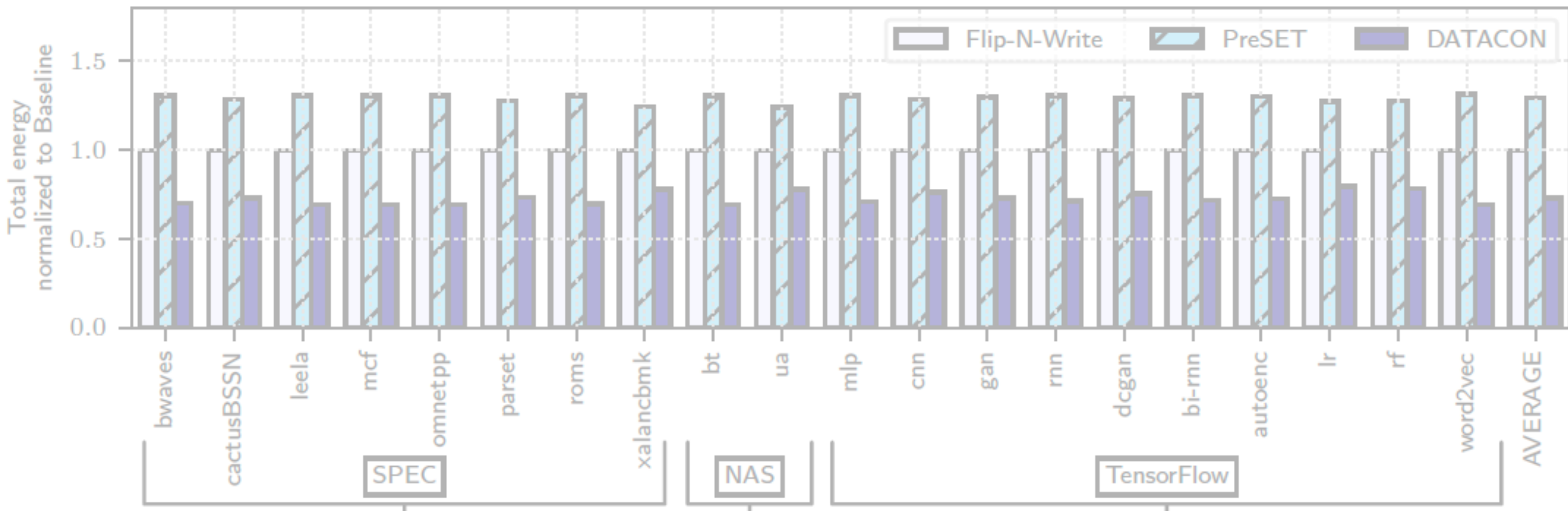
Memory System Energy

- Energy normalized to Baseline



Memory System Energy

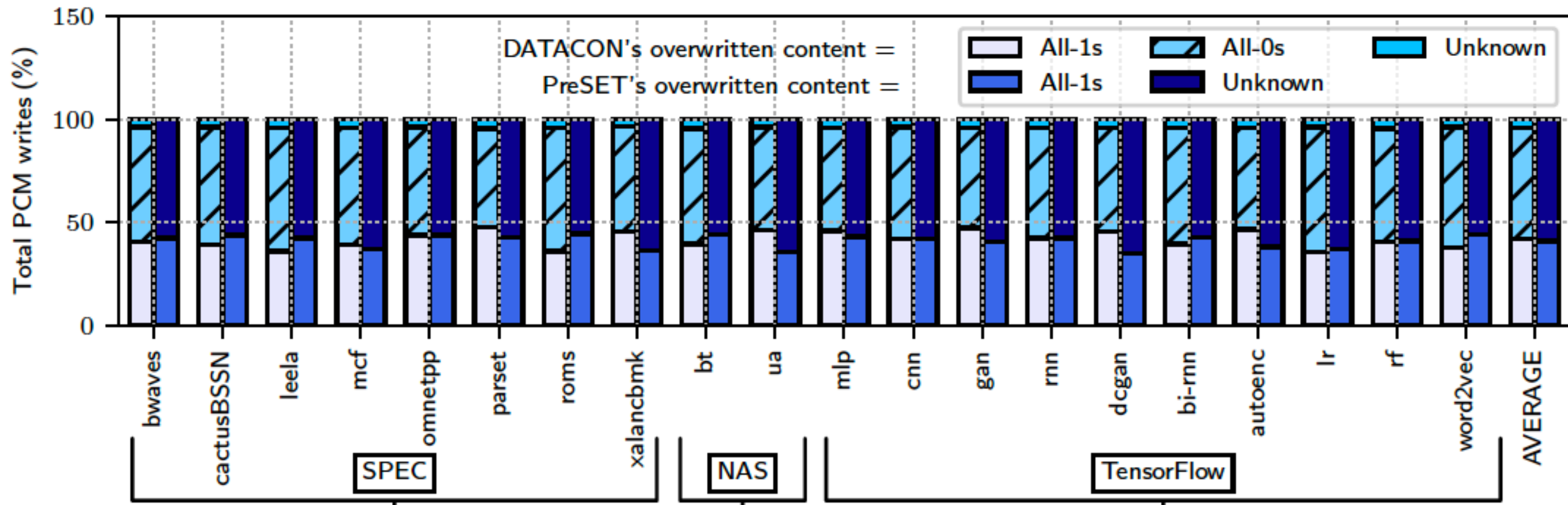
- Energy normalized to Baseline



Average Energy of DATACON is
27% lower than Baseline,
26% lower than Flip-N-Write, and
43% lower than PreSET

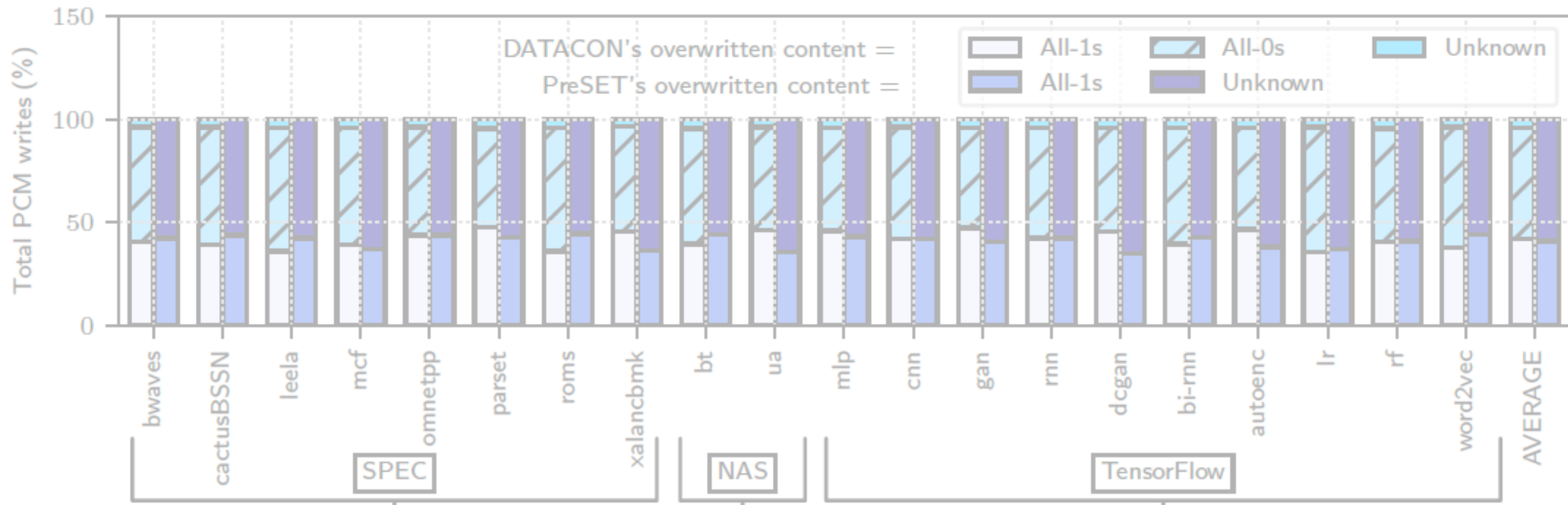
Overwritten Content

- Overwritten content in DATACON and PreSET



Overwritten Content

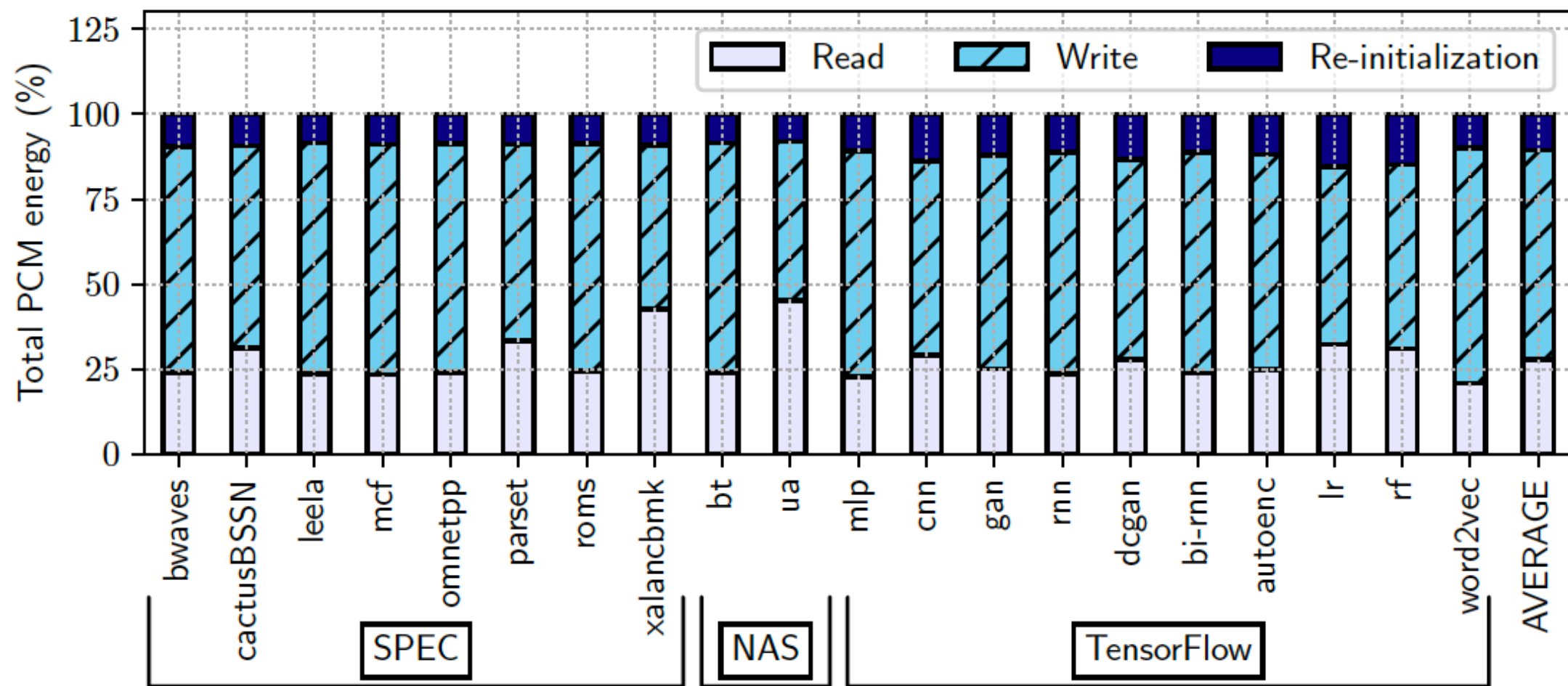
- Overwritten content in DATACON and PreSET



DATACON overwrites unknown content for **only 4%** of PCM writes.
PreSET overwrites unknown content for **59%** of PCM writes.

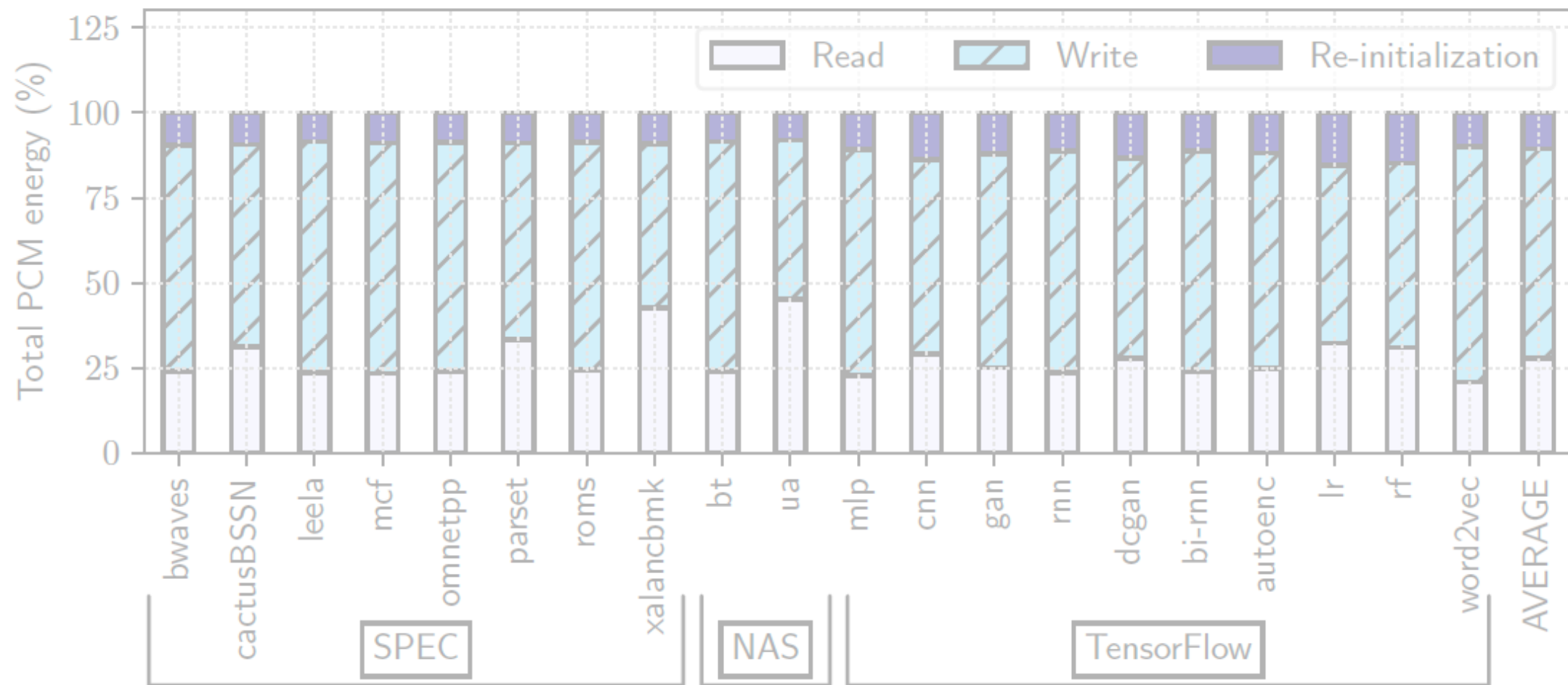
Re-initialization Overhead

- Total PCM energy distributed into energy to service reads, writes, and re-initialization requests in DATACON



Re-initialization Overhead

- Total PCM energy distributed into energy to service reads, writes, and re-initialization requests in DATACON



Re-initialization requests constitute, **on average,**
11% of the total PCM energy

Outline

- Introduction
- Detailed design of DATACON
- Address translation
- Overwritten content selection
- Re-initialization
- Evaluation
- Conclusion

Conclusion

- Observations
 - During PCM write, latency and energy are sensitive to the *data to be written as well as the content that is overwritten*, which is *unknown* at the time to write
 - Overwriting *known* all-0s or all-1s content improves both latency and energy by programming the PCM cells *only* in one direction, i.e., using SET or RESET operations, not both
- Idea: Data Content Aware PCM Writes
 - Overwrite unknown content *only when necessary*, and otherwise overwrite all-0s and all-1s content ➡ reduces latency and energy
- Two Key Mechanisms
 - **Address translation**: to translate the write address to a physical address within memory that contains the best type of content to overwrite
 - **Re-Initialization**: to methodically re-initialize *unused* memory locations with known all-0s and all-1s content in a manner that does not interfere with regular read and write accesses
- Performance Evaluation
 - Significant **improvement of system performance (27%) and reduction of energy (43%)** for SPEC CPU2017, NAS Parallel, and Tensorflow Benchmarks

Improving Phase Change Memory Performance via Data Content Aware Access

DATACON

Anup Das

Shihao Song, Onur Mutlu, and Nagarajan Kandasamy

Contact: shihao.song@drexel.edu

