18-447
Computer Architecture
Lecture 30: In-memory Processing

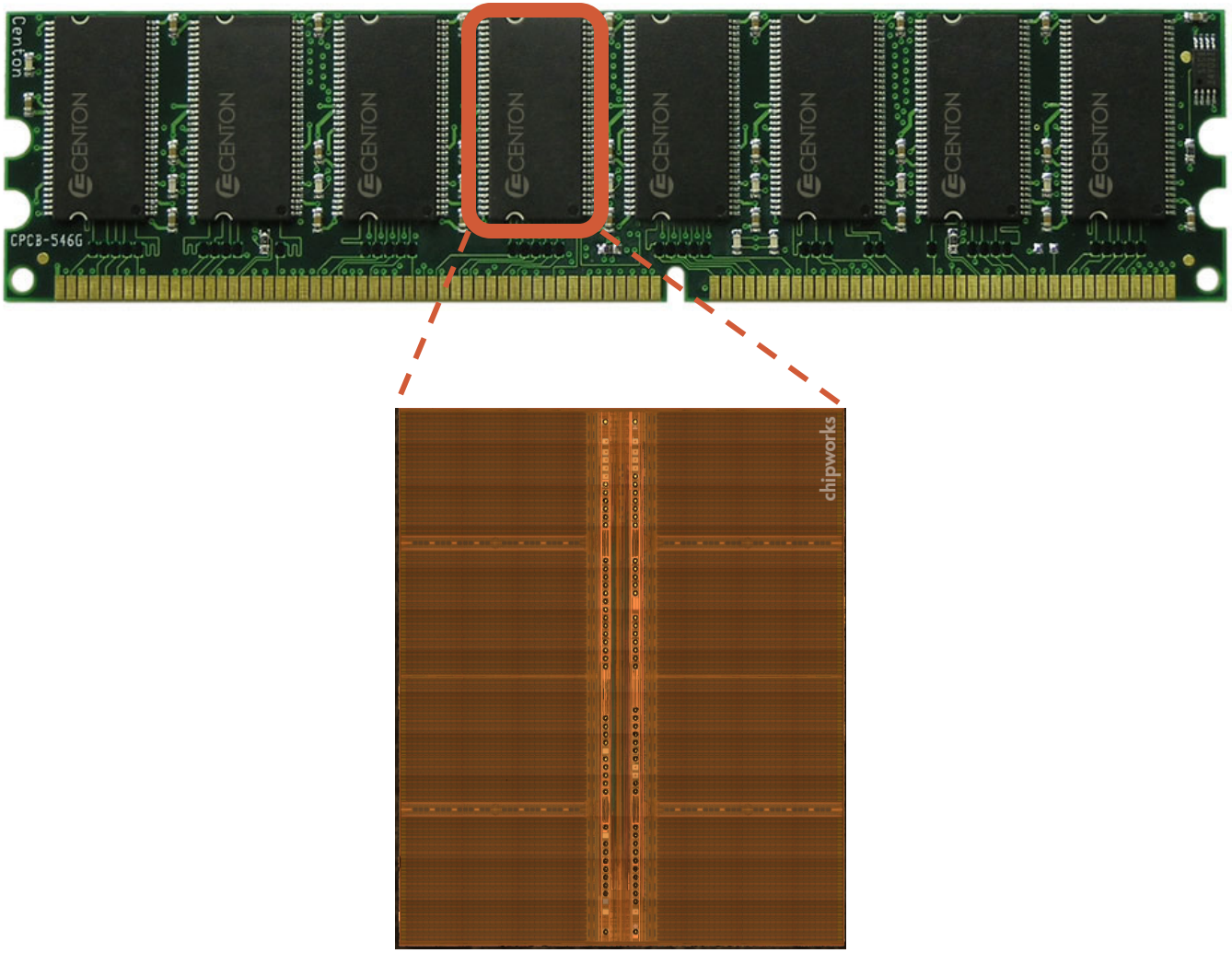
Vivek Seshadri
Carnegie Mellon University
Spring 2015, 4/13/2015

Goals for This Lecture

- Understand DRAM technology
  - How it is built?
  - How it operates?
  - What are the trade-offs?

- Can we use DRAM for more than just storage?
  - In-DRAM copying
  - In-DRAM bitwise operations

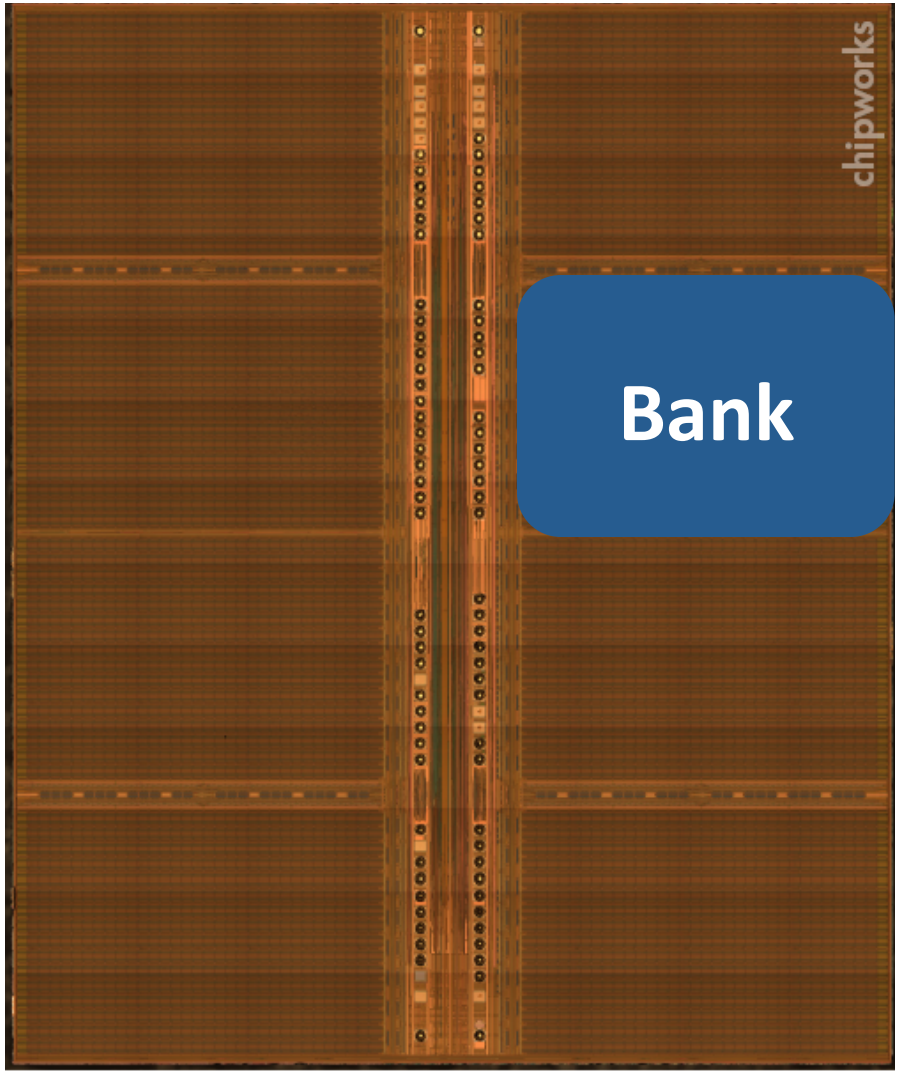
DRAM Module and Chip



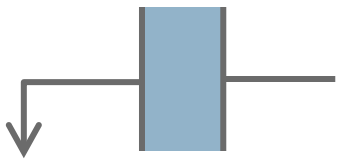
Goals of DRAM Design

- Cost
- Latency
- Bandwidth
- Parallelism
- Power
- Energy
- Reliability

DRAM Chip



DRAM Cell – Capacitor



Empty State

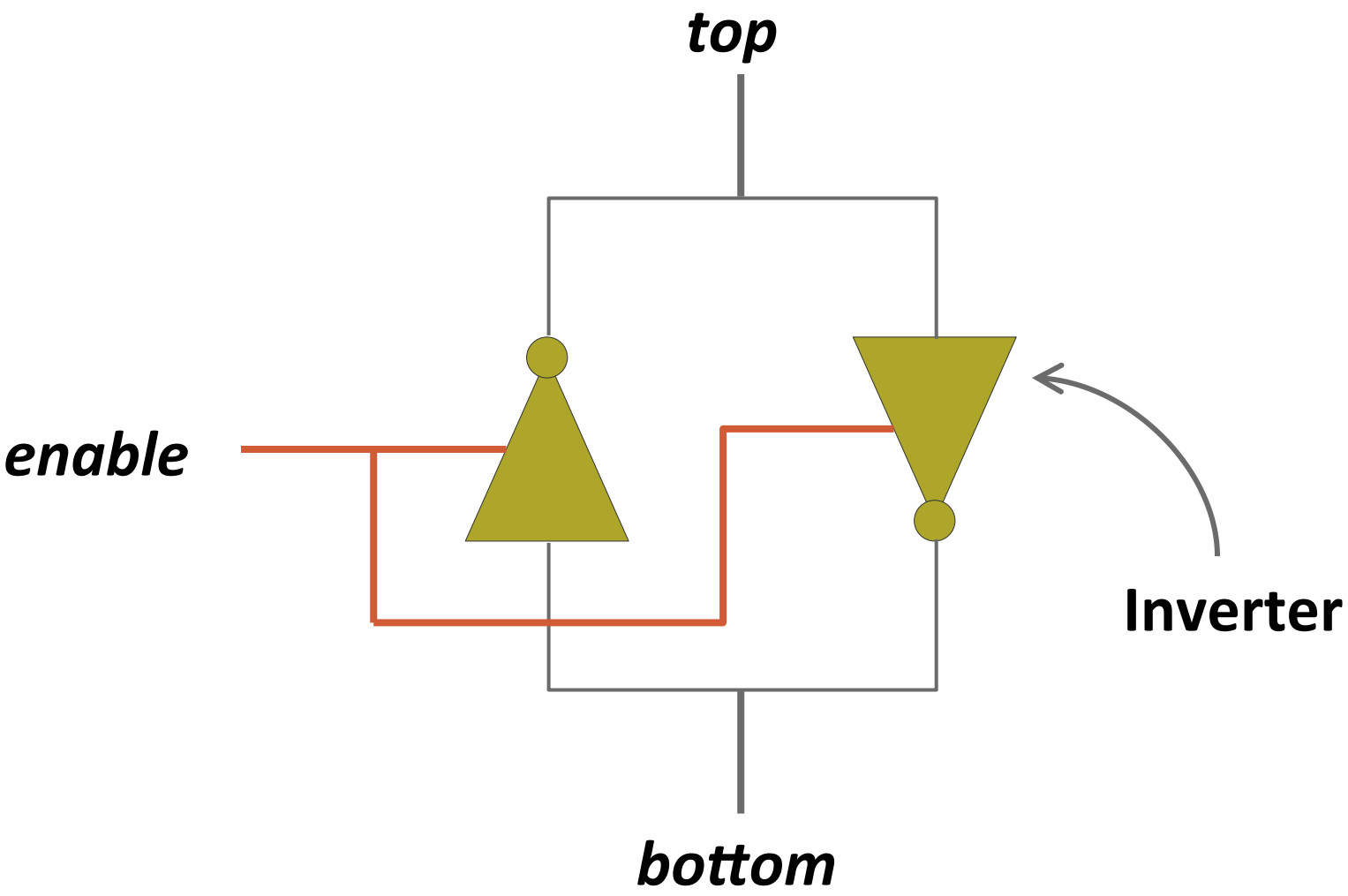
Logical "0"

Fully Charged State

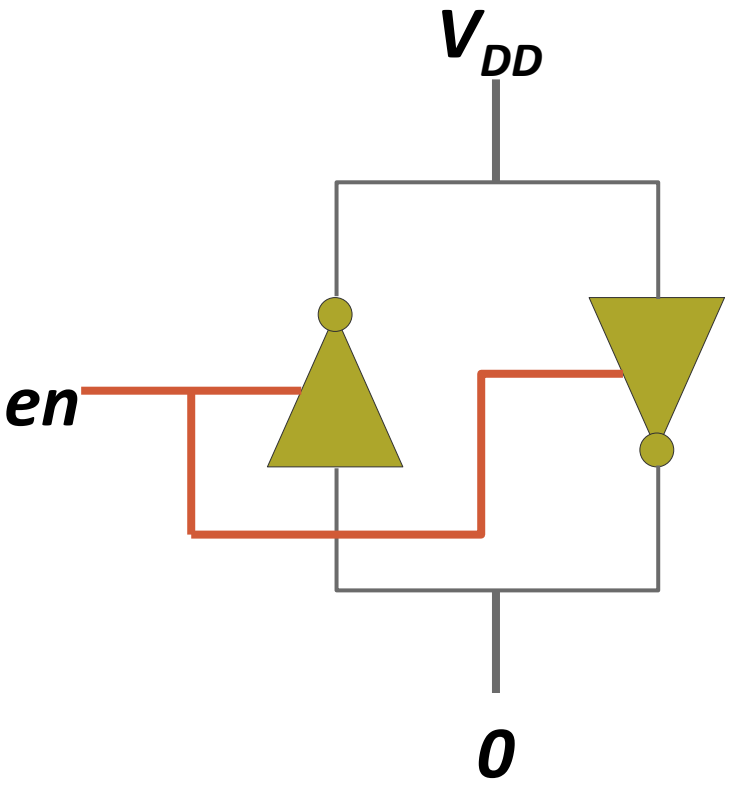
Logical "1"

1. **Small – Cannot drive circuits**
2. **Reading destroys the state**

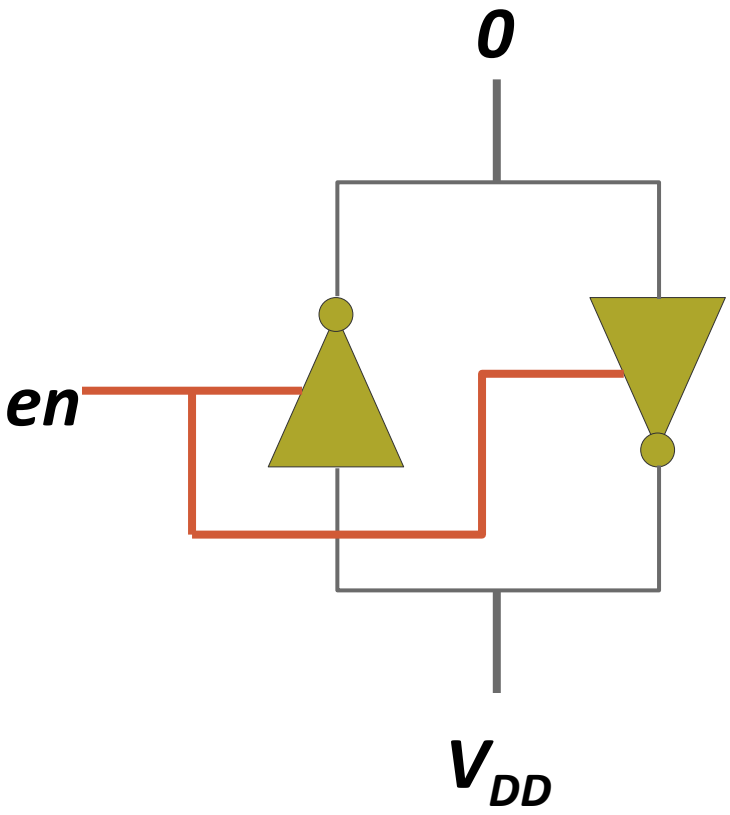
Sense Amplifier



Sense Amplifier – Two Stable States

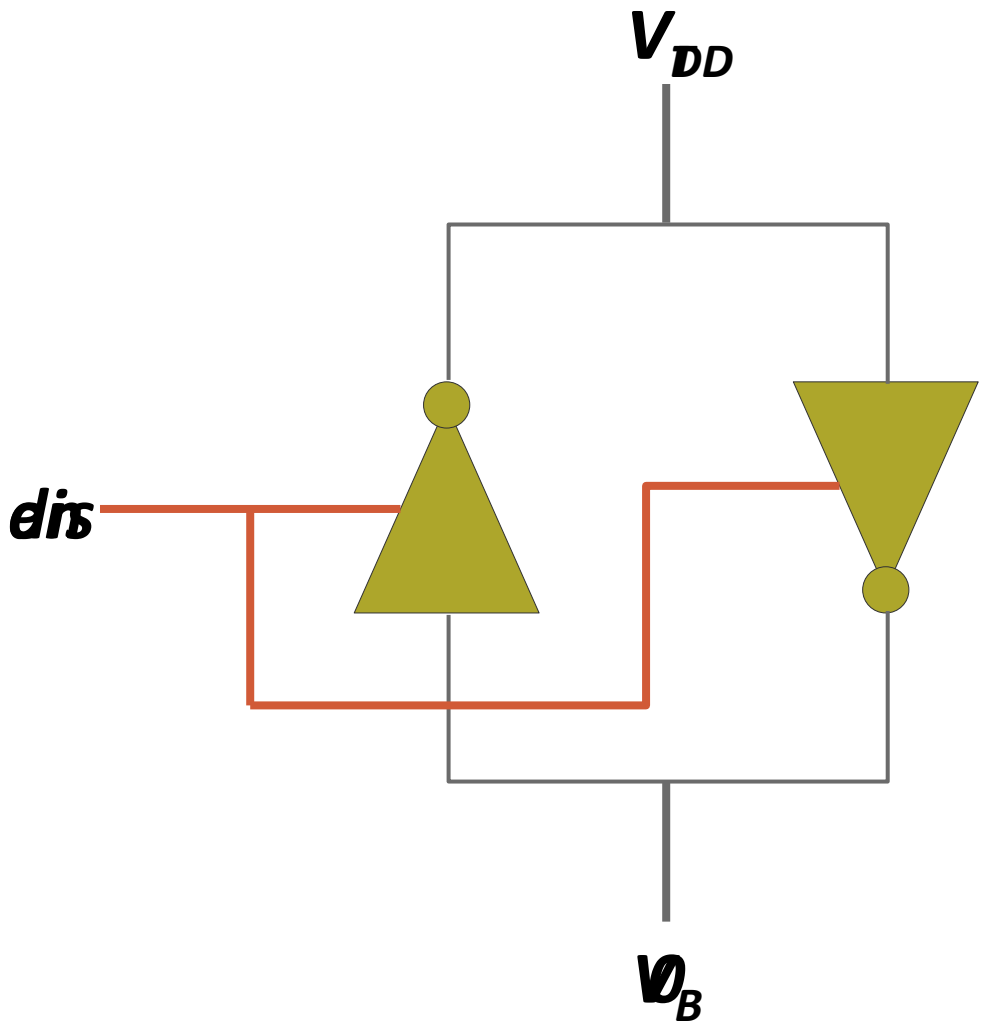


Logical "1"



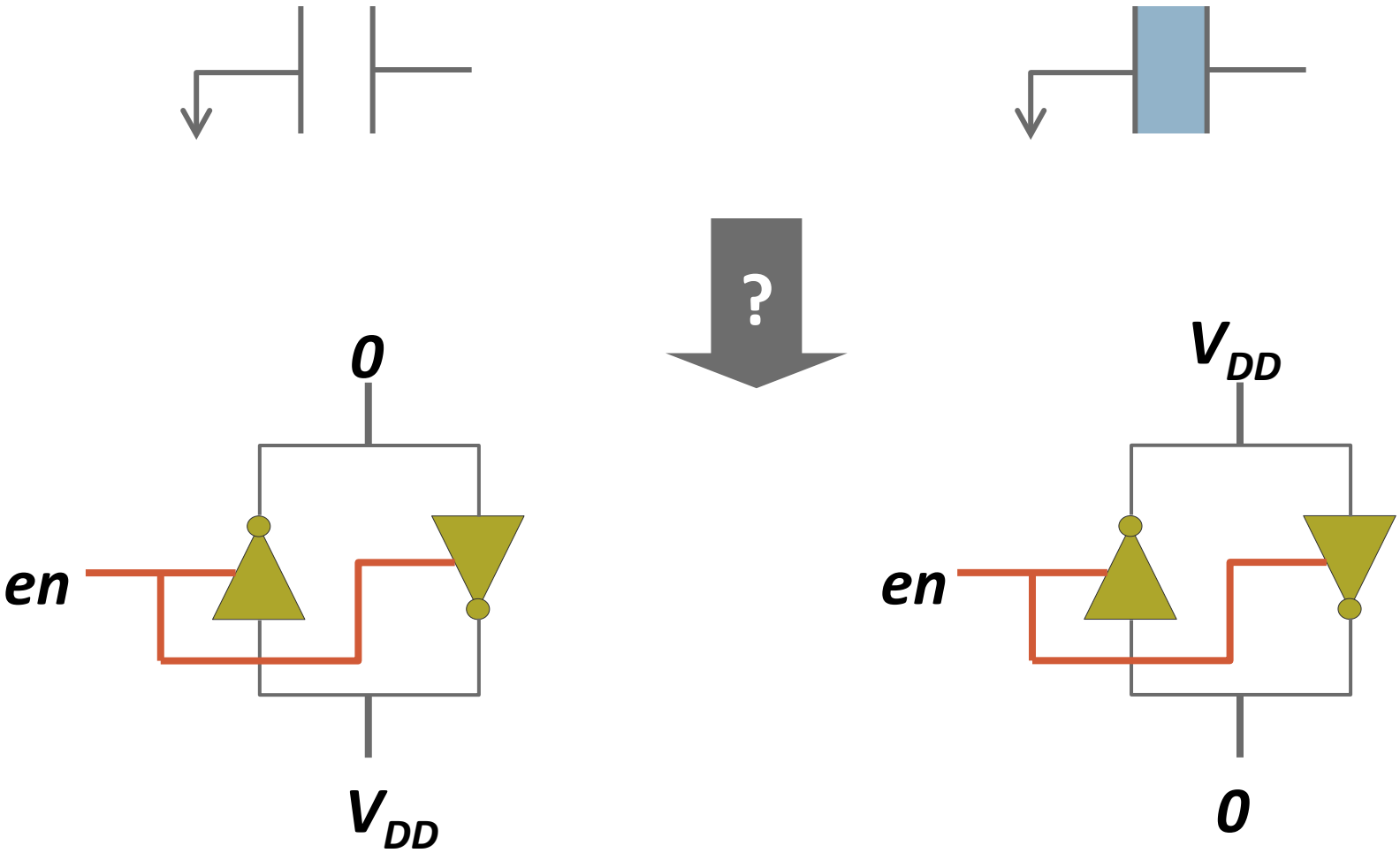
Logical "0"

Sense Amplifier Operation

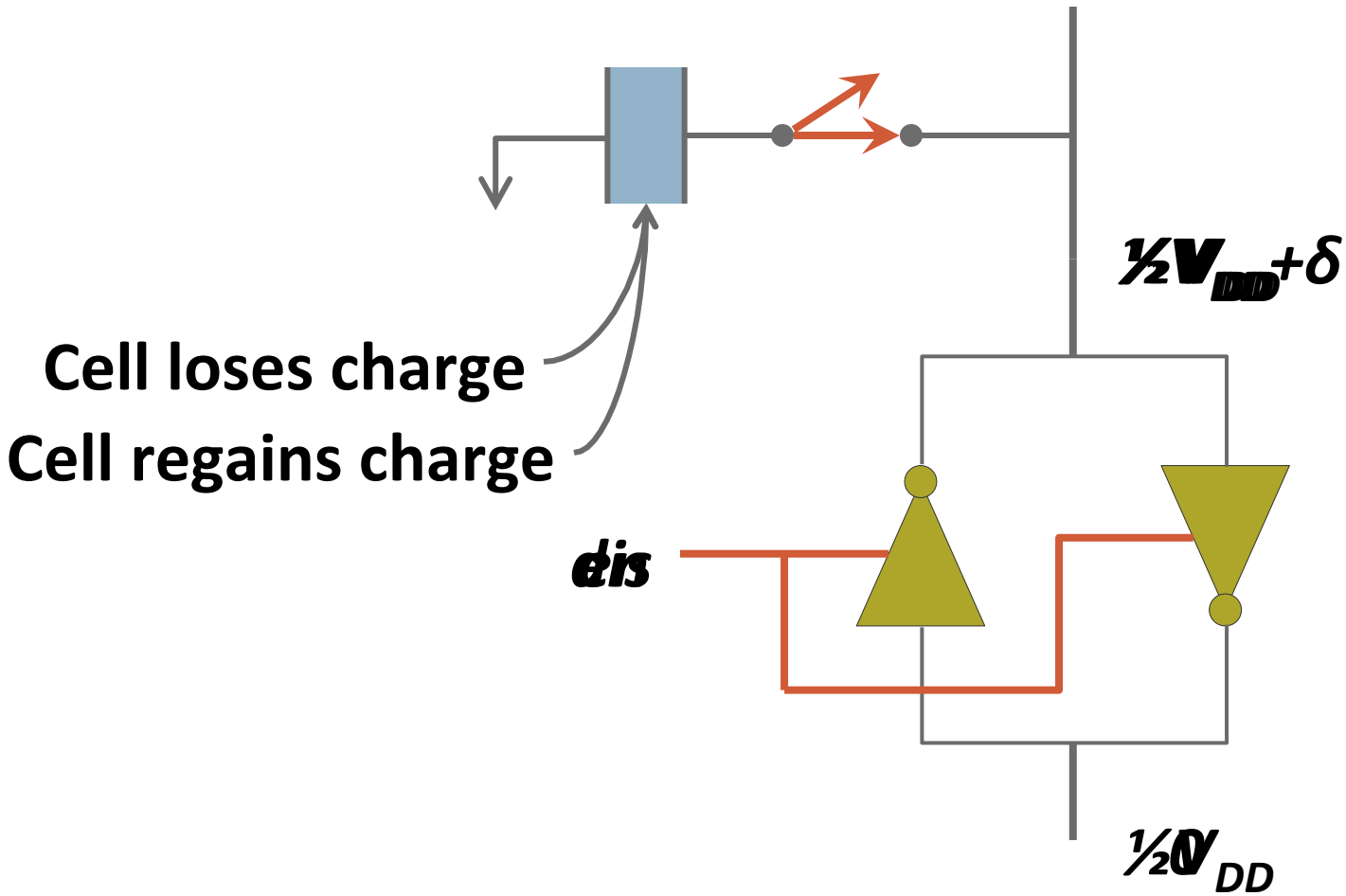


$V_T > V_B$

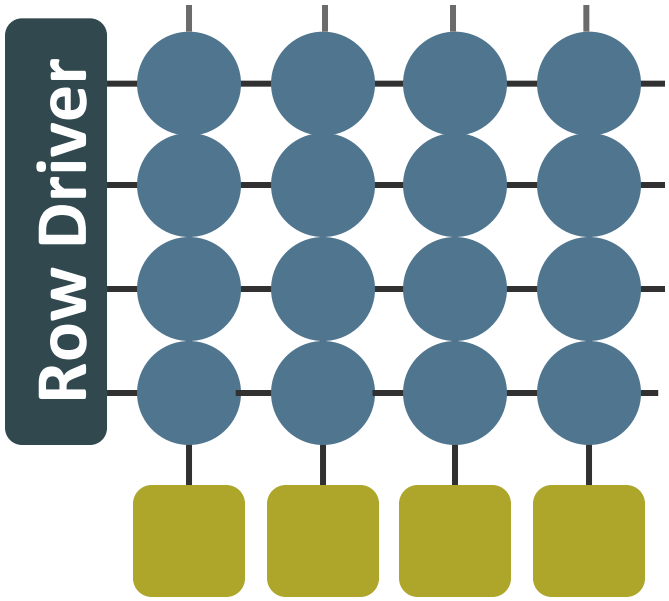
Capacitor to Sense Amplifier



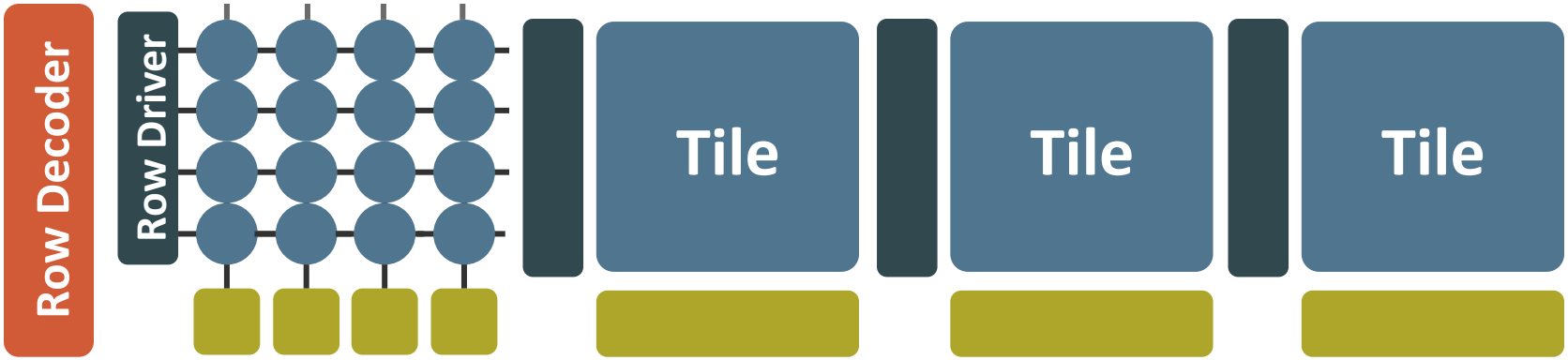
DRAM Cell Operation



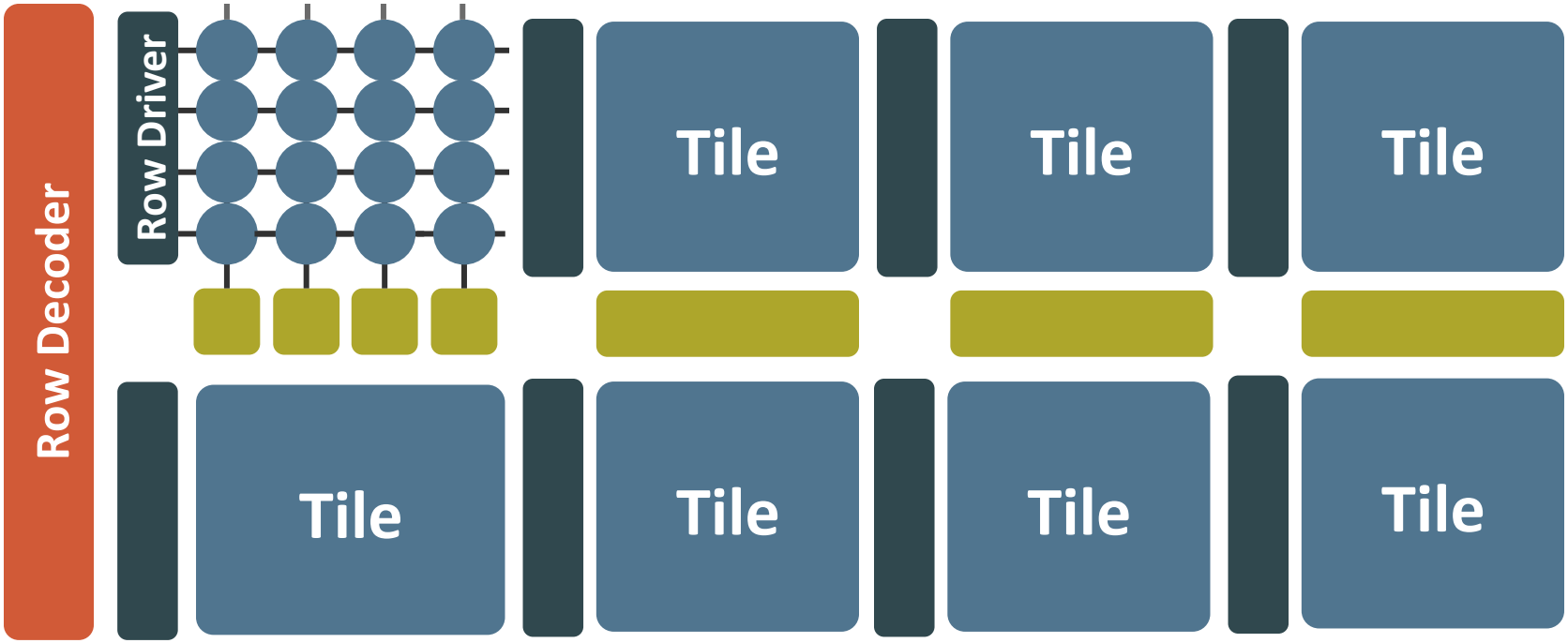
Amortizing Cost – DRAM Tile



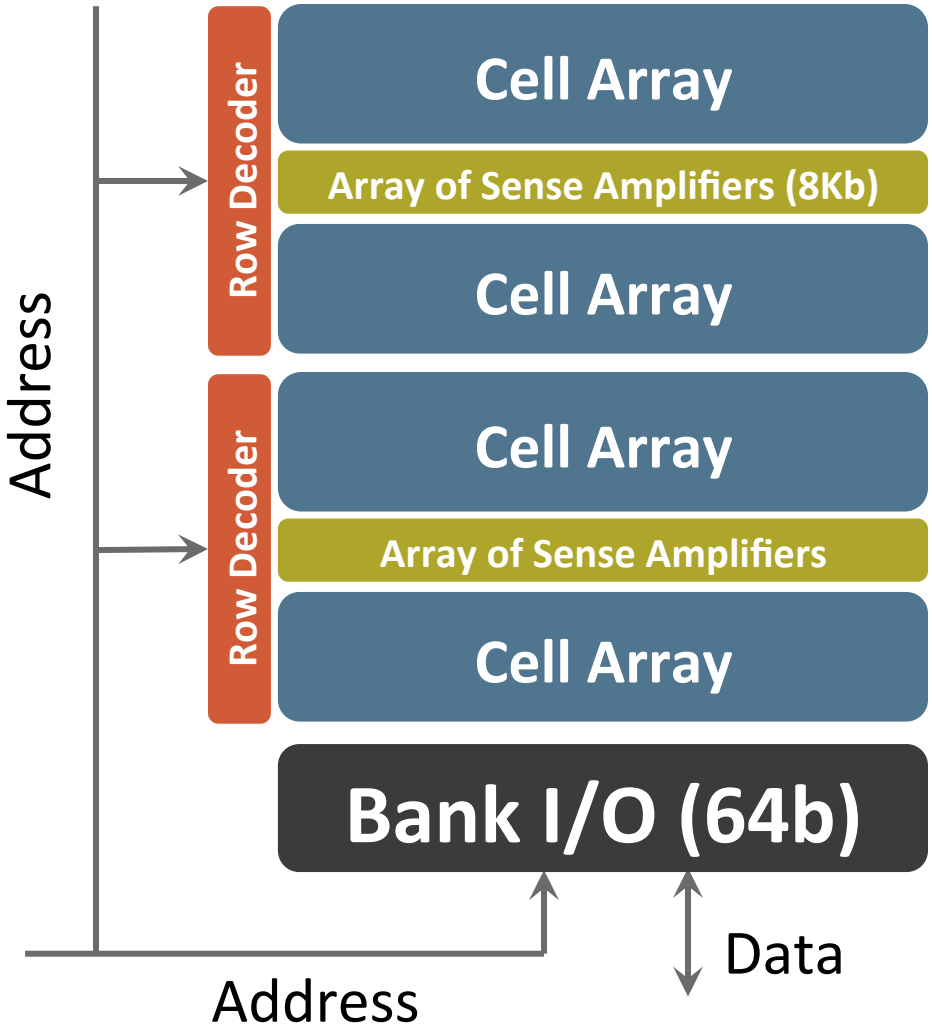
DRAM Subarray



DRAM Subarray

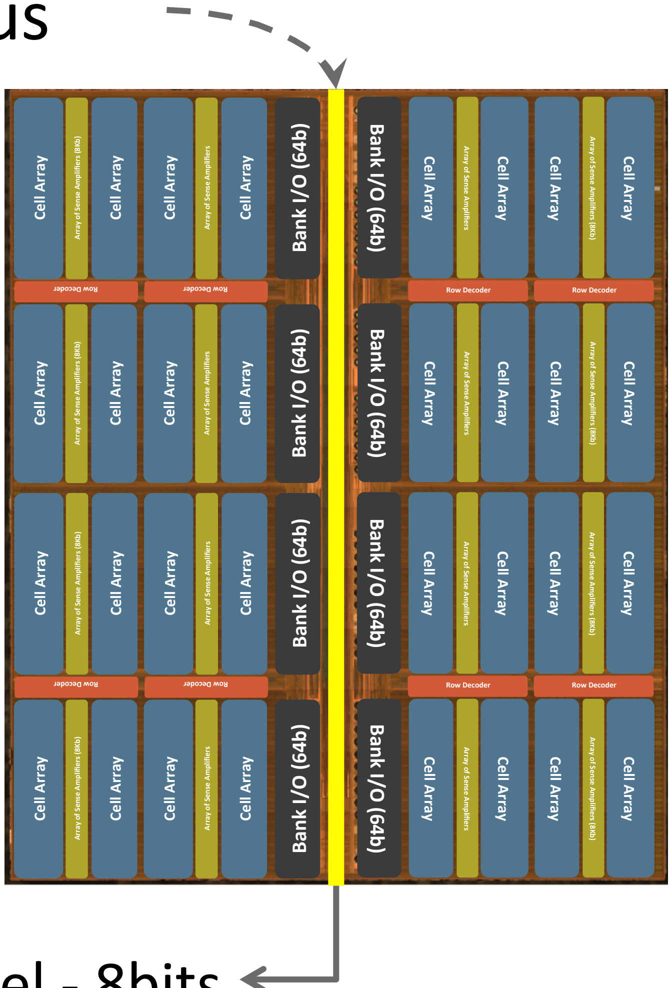


DRAM Bank


Address
Row Decoder
Cell Array
Array of Sense Amplifiers (8Kb)
Cell Array
Row Decoder
Cell Array
Array of Sense Amplifiers
Cell Array
Bank I/O (64b)
Address
Data

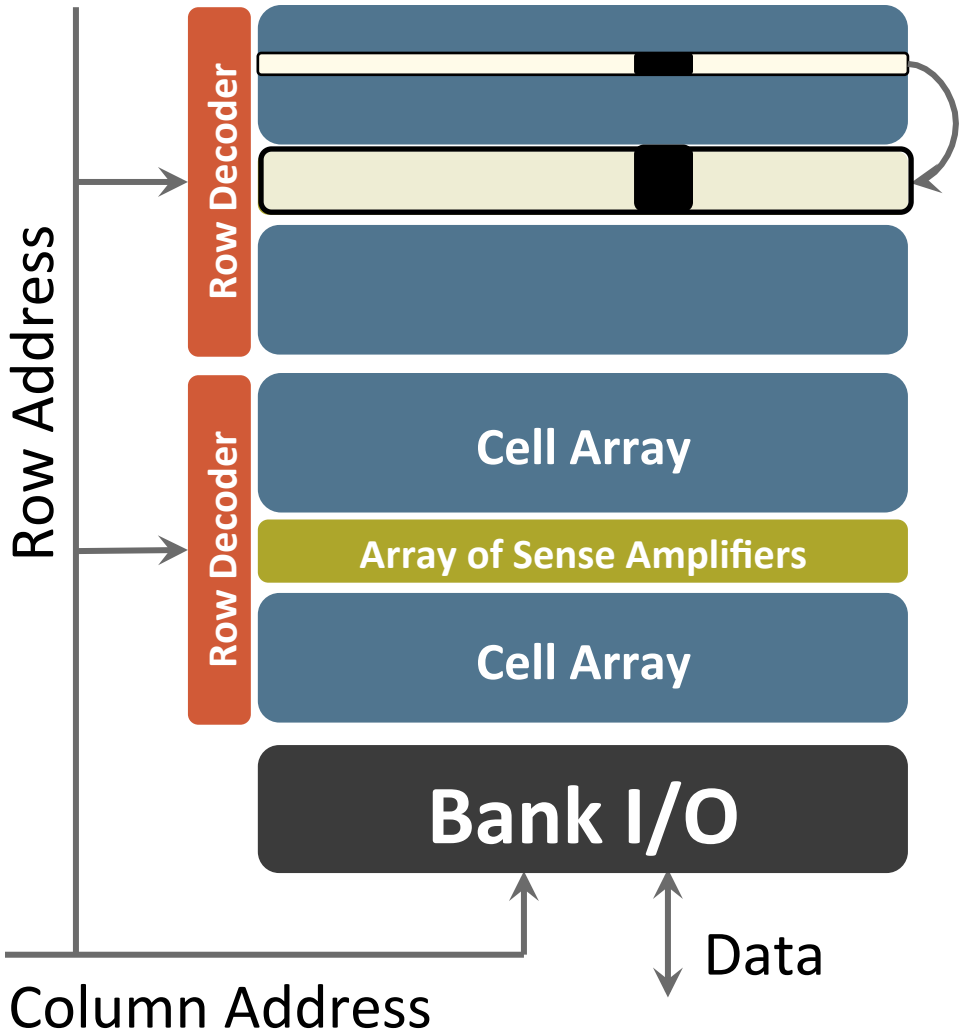
DRAM Chip

Shared internal bus



Memory channel - 8bits

DRAM Operation



1. ACTIVATE Row
2. READ/WRITE Column
3. PRECHARGE

Goals for This Lecture

- Understand DRAM technology
  - How it is built?
  - How it operates?
  - What are the trade-offs?

- Can we use DRAM for more than just storage?
  - In-DRAM copying
  - In-DRAM bitwise operations

Trade-offs in DRAM Design

- Cost
- Latency —Rows/Subarray
- Bandwidth —Data width, Chips/DIMM
- Parallelism —Banks/Chip
- Power
- Energy
- Reliability

Goals for This Lecture

✓ Understand DRAM technology
  - How it is built?
  - How it operates?
  - What are the trade-offs?

- Can we use DRAM for more than just storage?
  - In-DRAM copying
  - In-DRAM bitwise operations

RowClone

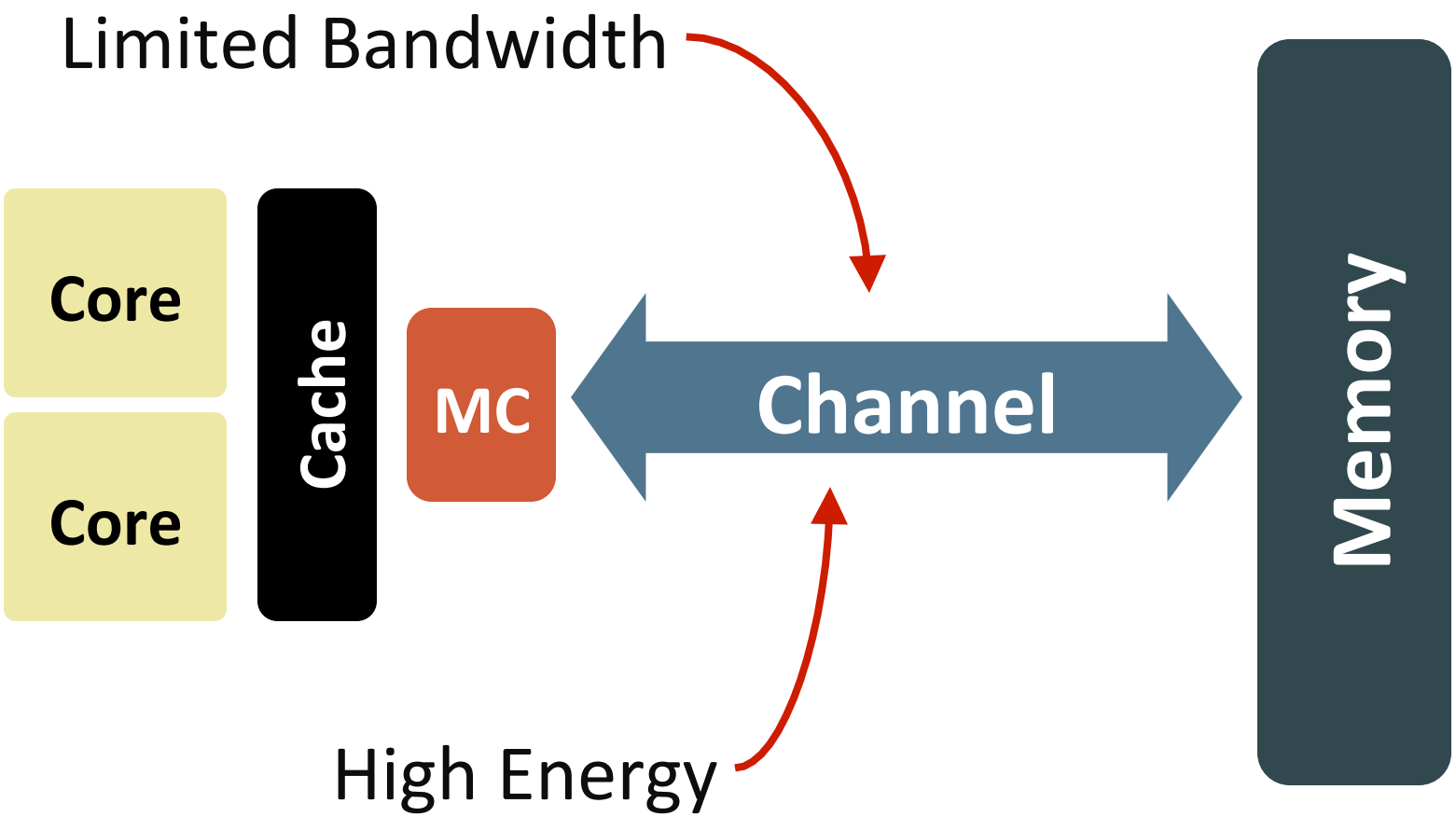
Fast and Energy-Efficient In-DRAM Bulk Data Copy and Initialization

Vivek Seshadri

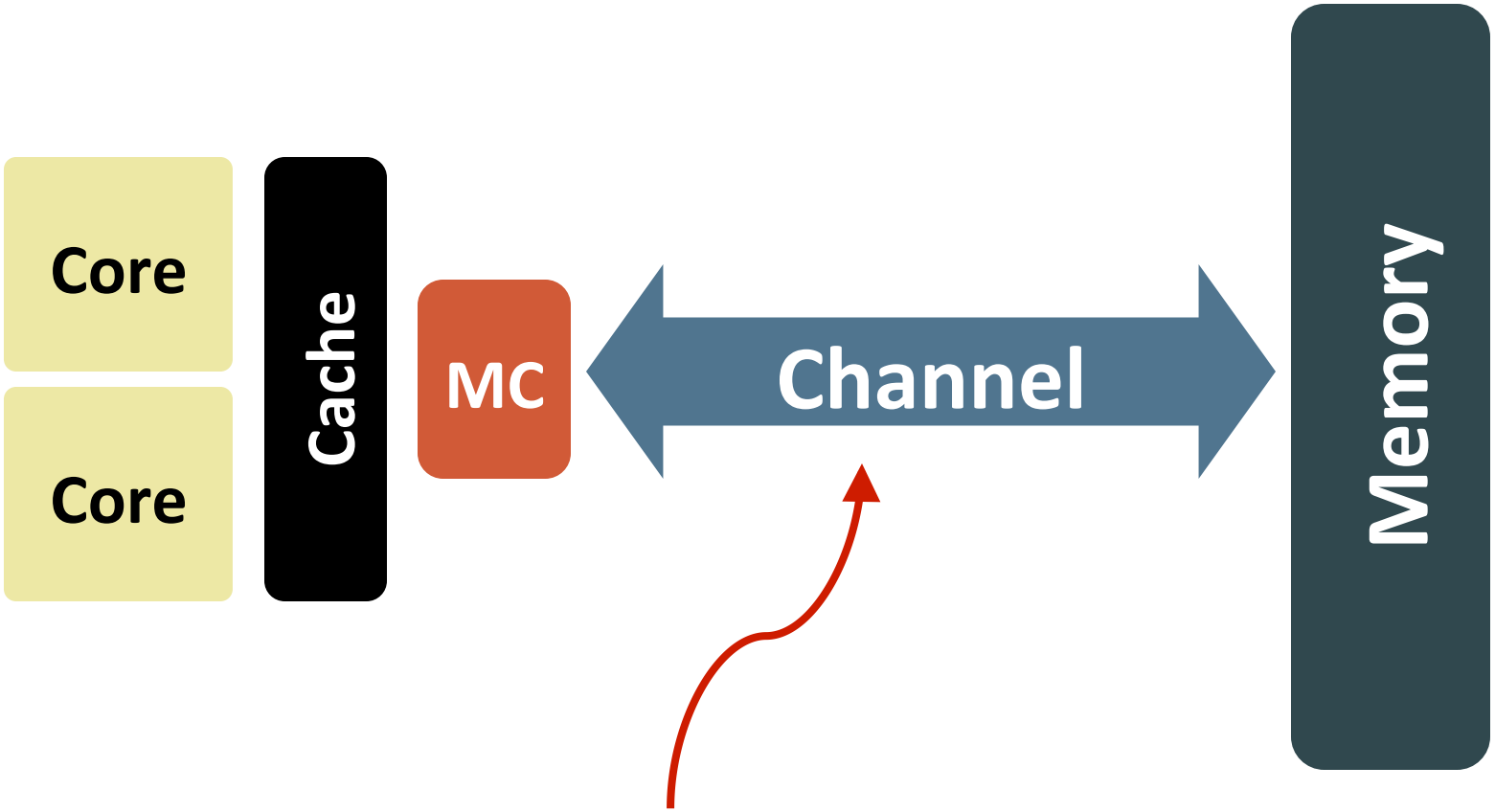
Y. Kim, C. Fallin, D. Lee, R. Ausavarungnirun,
G. Pekhimenko, Y. Luo, O. Mutlu,
P. B. Gibbons, M. A. Kozuch, T. C. Mowry

SAFARI Carnegie Mellon intel

Memory Channel – Bottleneck



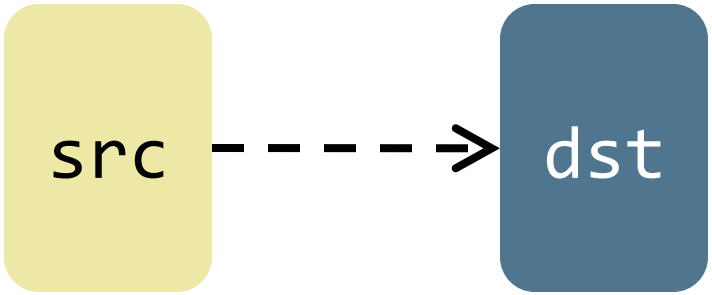
Goal: Reduce Memory Bandwidth Demand



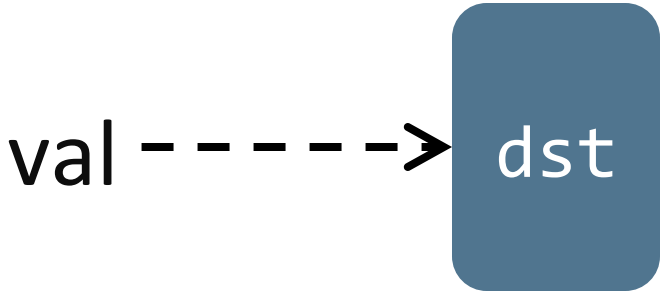
Reduce unnecessary data movement

Bulk Data Copy and Initialization

Bulk Data Copy



Bulk Data Initialization



Bulk Data Copy and Initialization

The Impact of Architectural Trends on Operating System Performance

Mendel Rosenblum, Edouard Bugnion, Stephen Alan Herrod,
Emmett Witchel, and Anoop Gupta

Hardware Support for Bulk Data Movement in Server Platforms

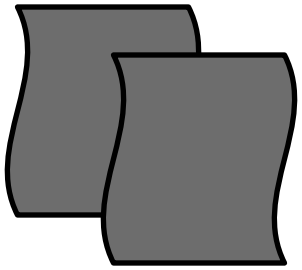
Li Zhao[†], Ravi Iyer[‡] Srihari Makineni[‡], Laxmi Bhuyan[†] and Don Newell[‡]
[†]Department of Computer Science and Engineering, University of California, Riverside, CA 92521
Email: {zhao, bhuyan}@cs.ucr.edu
[‡]Communications Technology Lab, Intel C

Architecture Support for Improving Bulk Memory Copying and Initialization Performance

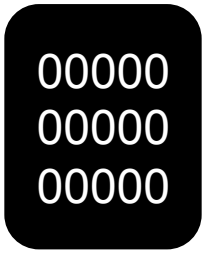
Xiaowei Jiang, Yan Solihin
Dept. of Electrical and Computer Engineering
North Carolina State University
Raleigh, USA

Li Zhao, Ravishankar Iyer
Intel Labs
Intel Corporation
Hillsboro, USA

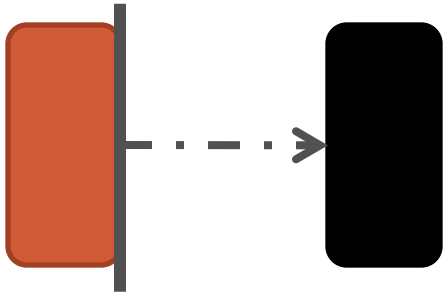
Bulk Copy and Initialization – Applications



Forking

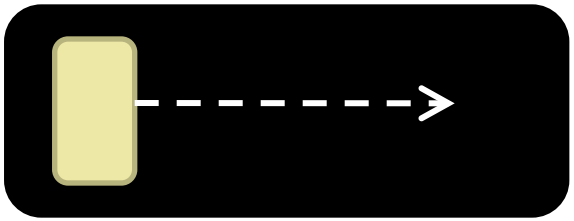


Zero initialization (e.g., security)



Checkpointing

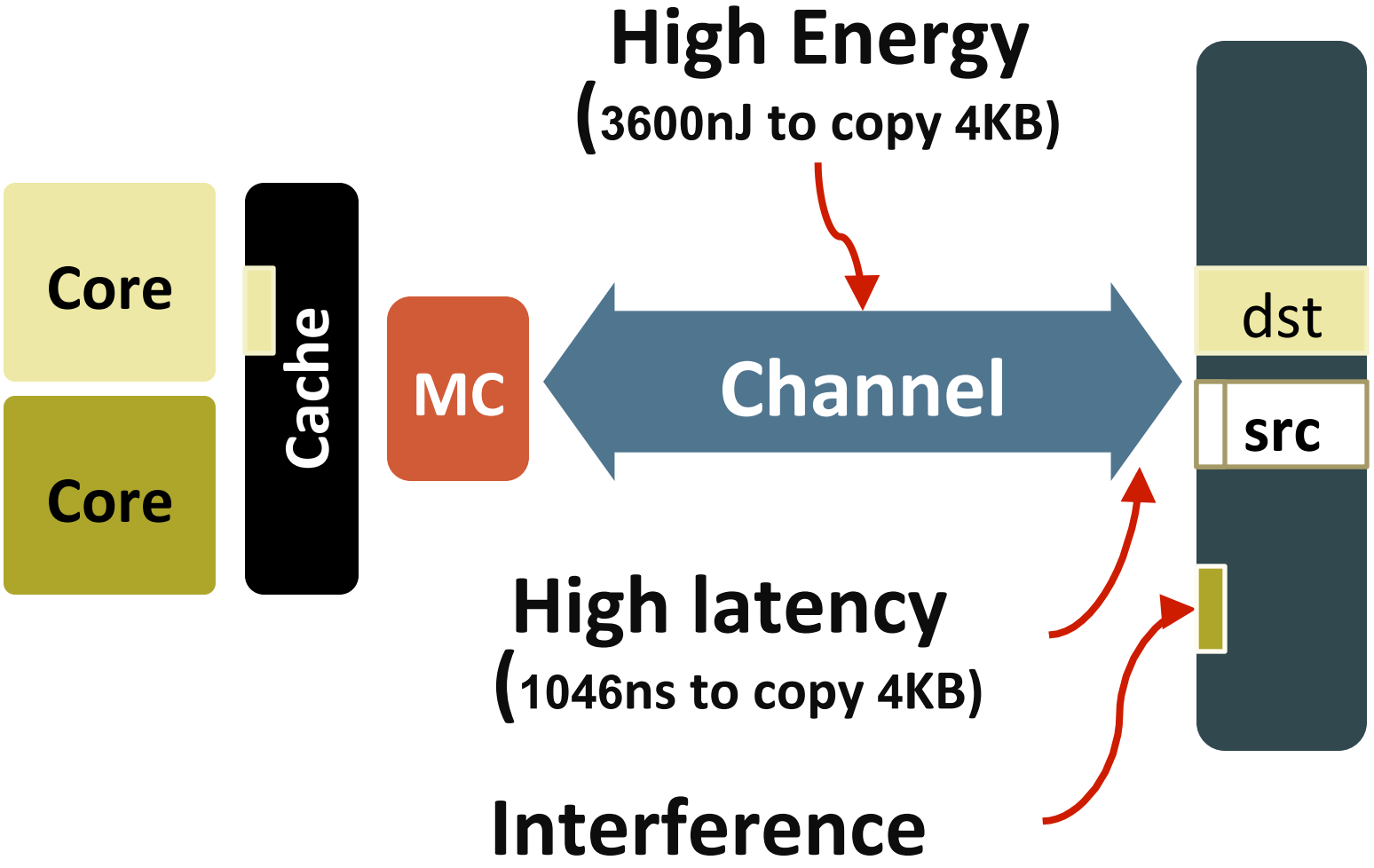
VM Cloning
Deduplication



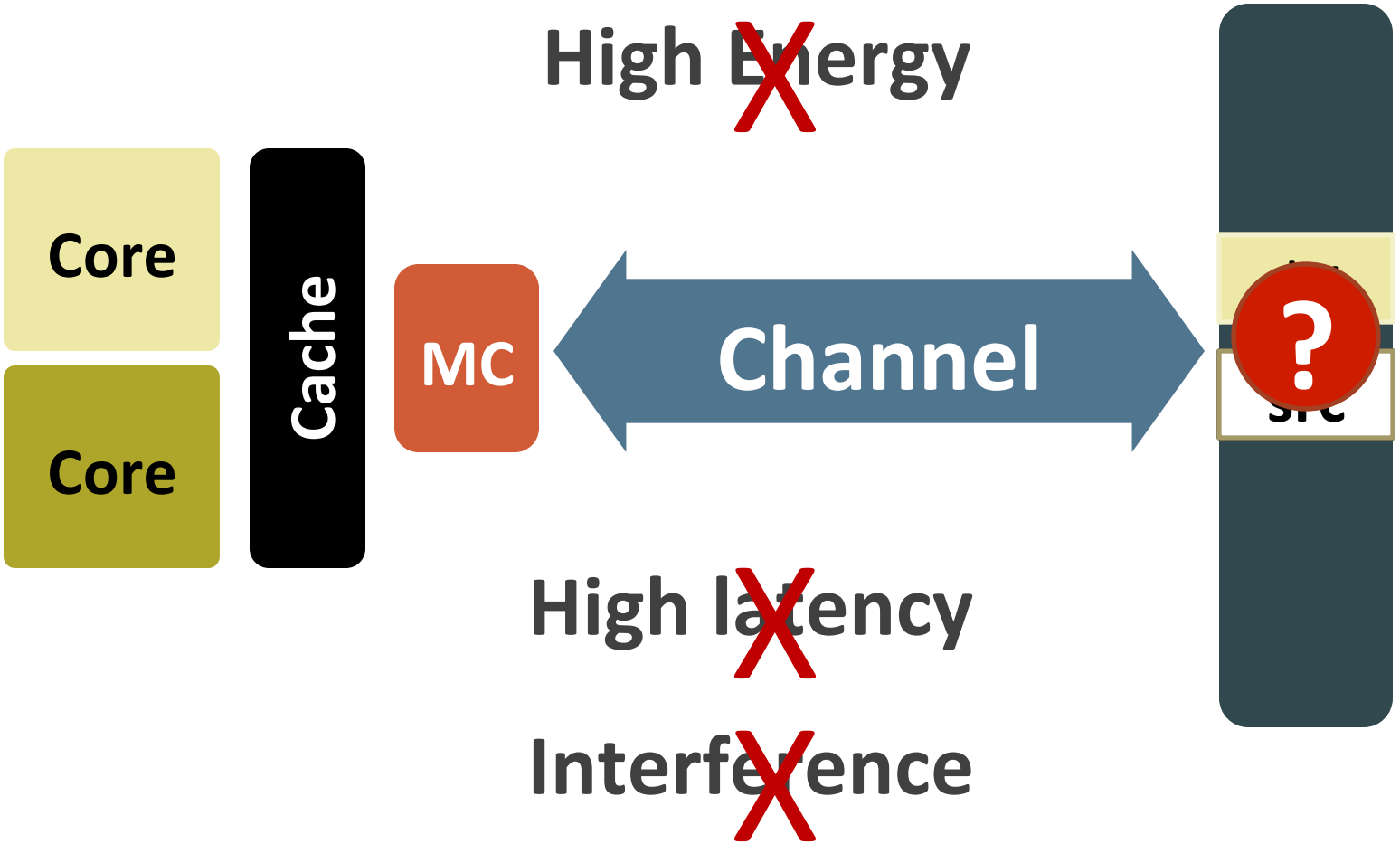
Page Migration

• • •
Many more

Shortcomings of Existing Approach

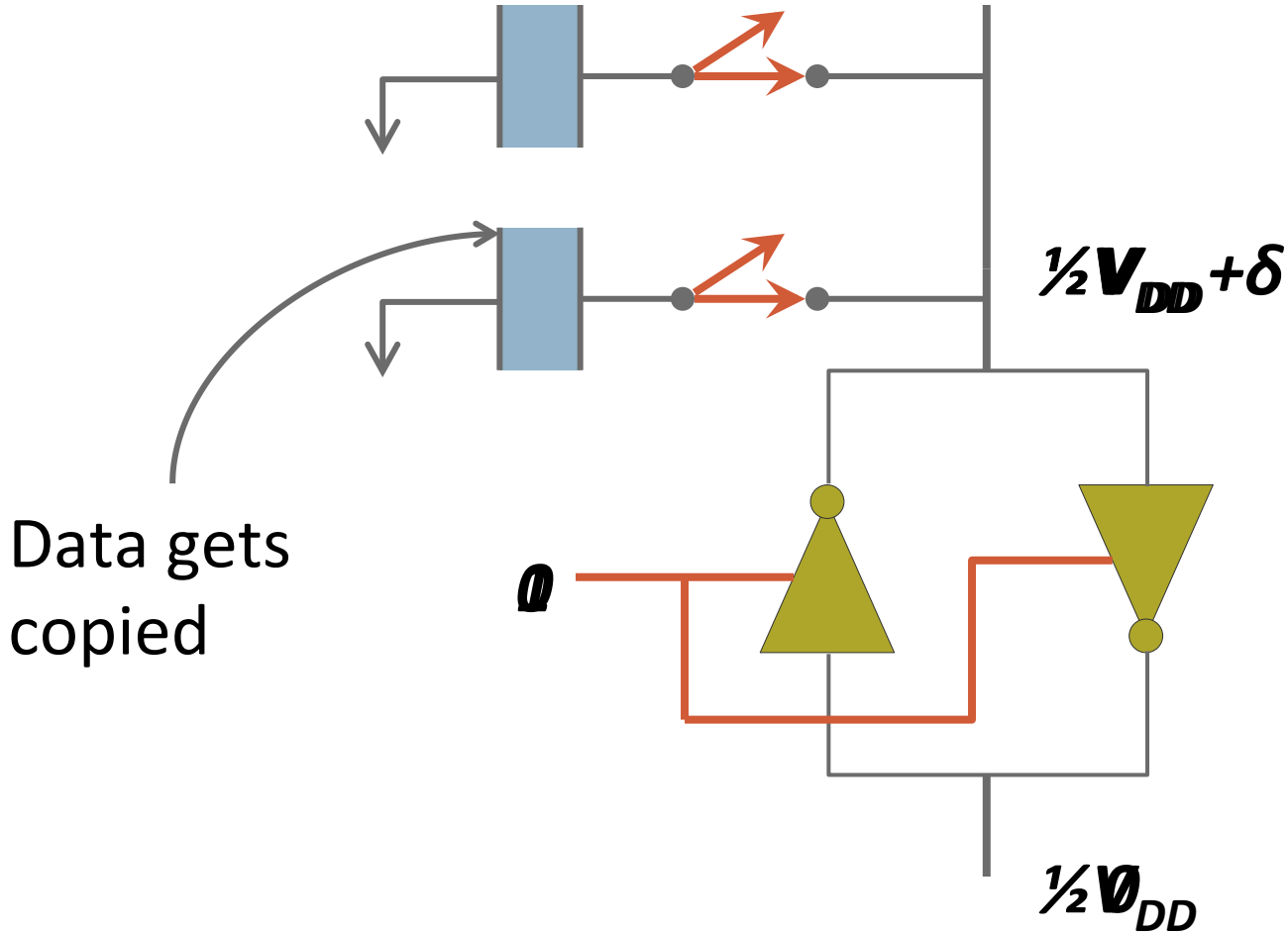


Our Approach: In-DRAM Copy with Low Cost



RowClone: In-DRAM Copy

Bulk Copy in DRAM – RowClone



Data gets copied

Fast Parallel Mode – Benefits

Bulk Data Copy (4KB across a module)

Latency **11X** ↓

1046ns to 90ns

Energy **74X** ↓

3600nJ to 40nJ

No bandwidth consumption

Very little changes to the DRAM chip

Fast Parallel Mode – Constraints

- Location constraint
  - Source and destination in same subarray
- Size constraint
  - Entire row gets copied (no partial copy)

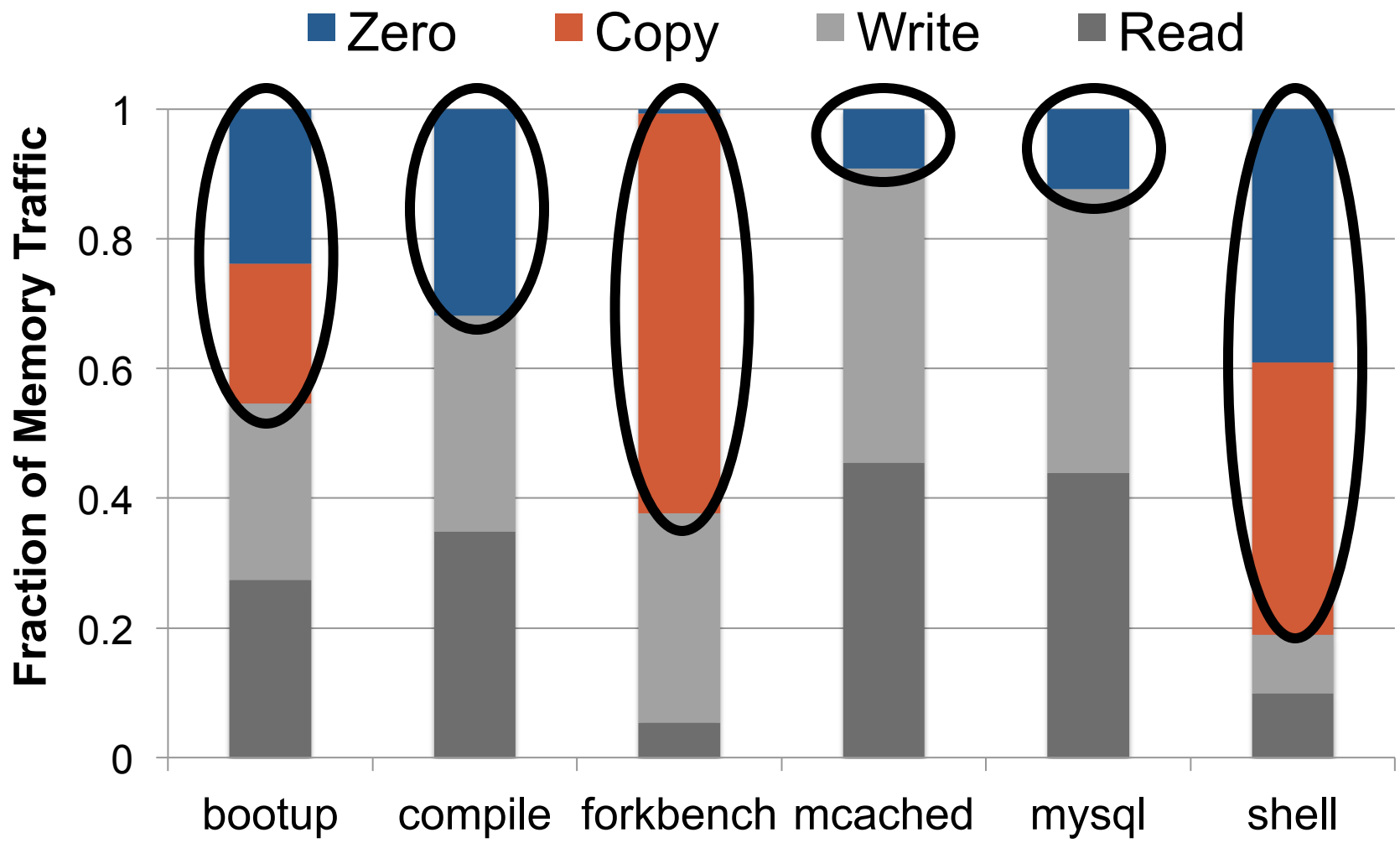
1. Can still accelerate many existing primitives (*copy-on-write, bulk zeroing*)

2. Alternate mechanism to copy data across banks (*pipelined serial mode – lower benefits than Fast Parallel*)

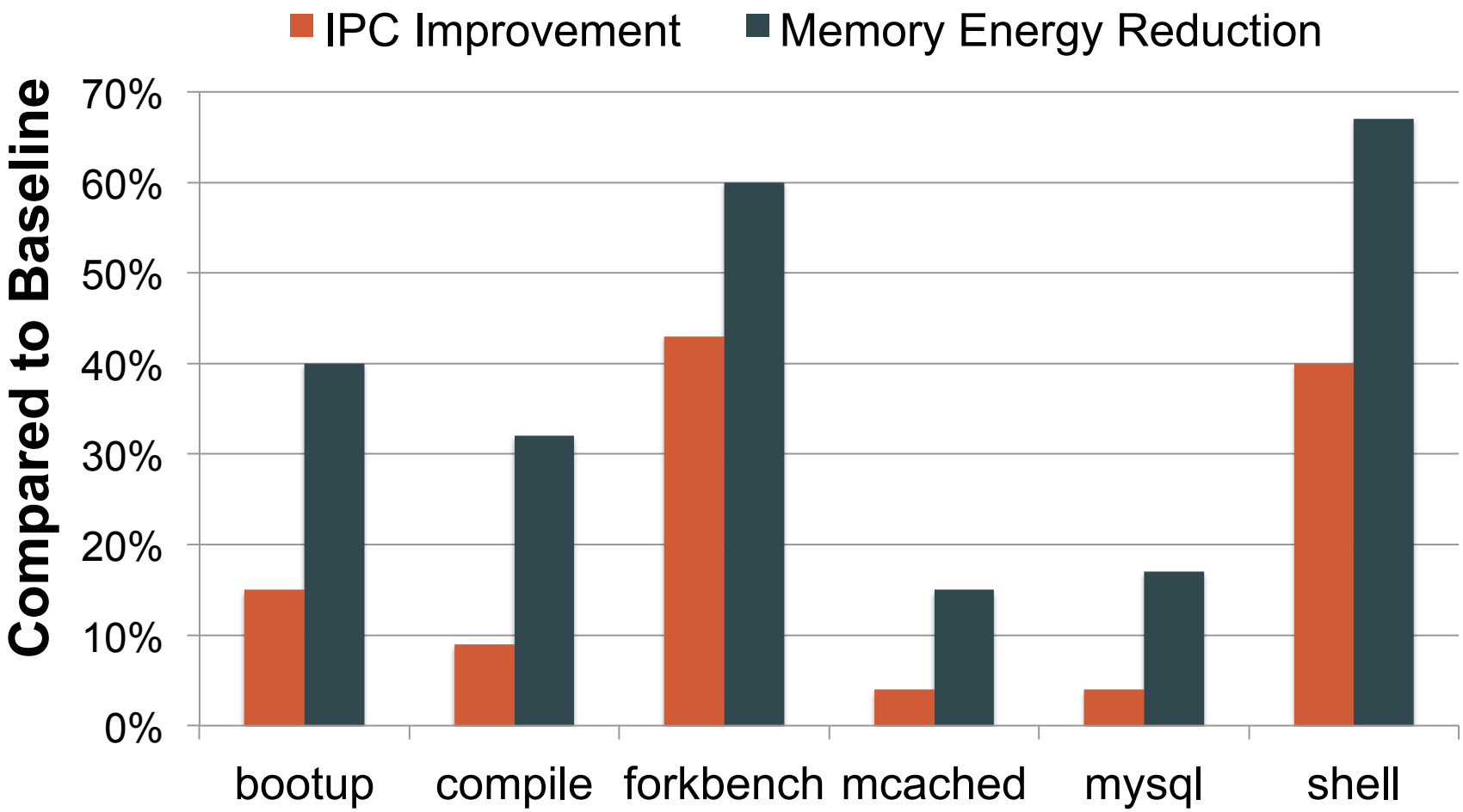
End-to-end System Design

- Software interface
  - memcpy and meminit instructions
- Managing cache coherence
  - Use existing DMA support!
- Maximizing use of Fast Parallel Mode
  - Smart OS page allocation

Applications Summary



Results Summary

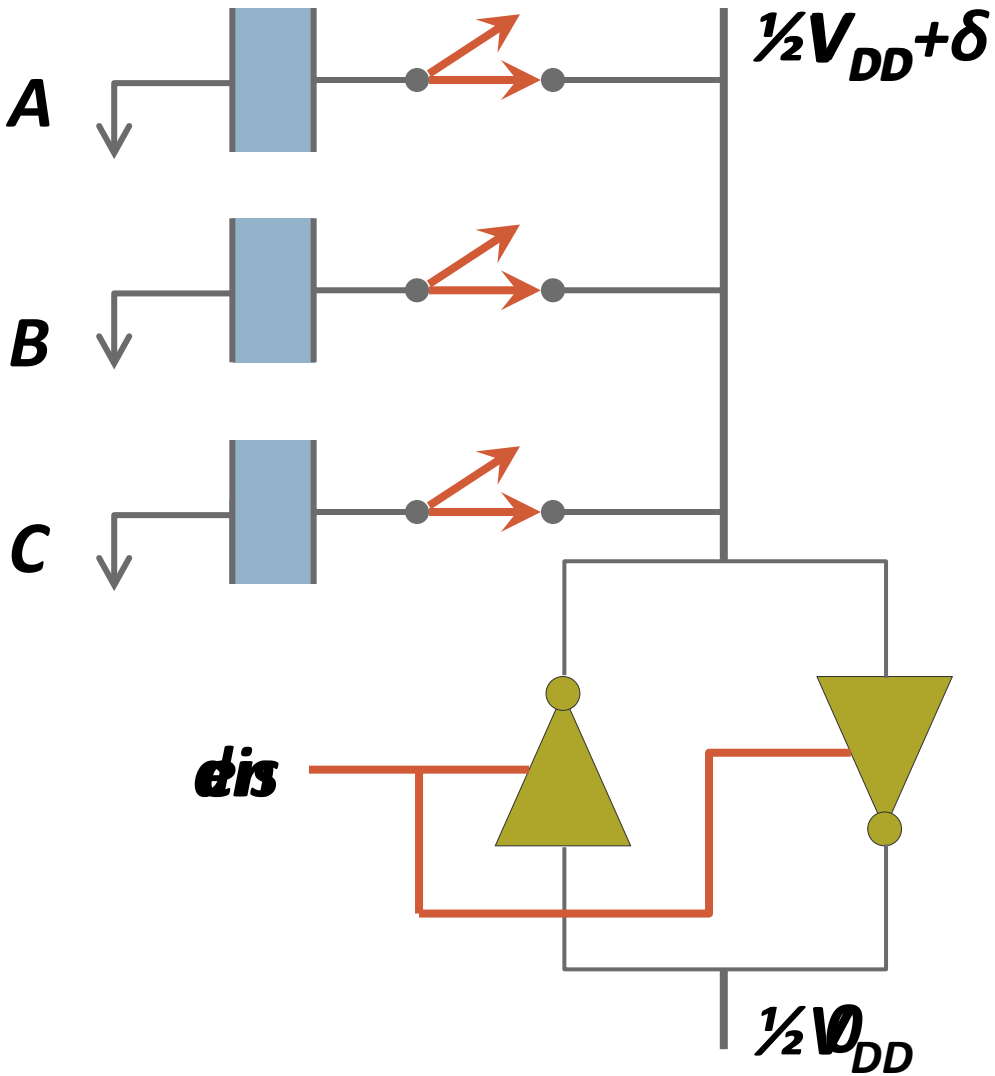


Goals for This Lecture

✓ Understand DRAM technology
  - How it is built?
  - How it operates?
  - What are the trade-offs?

- Can we use DRAM for more than just storage?
  - In-DRAM copying
  - In-DRAM bitwise operations

Triple Row Activation



Final State

AB + BC + AC

C(A + B) + ~C(AB)

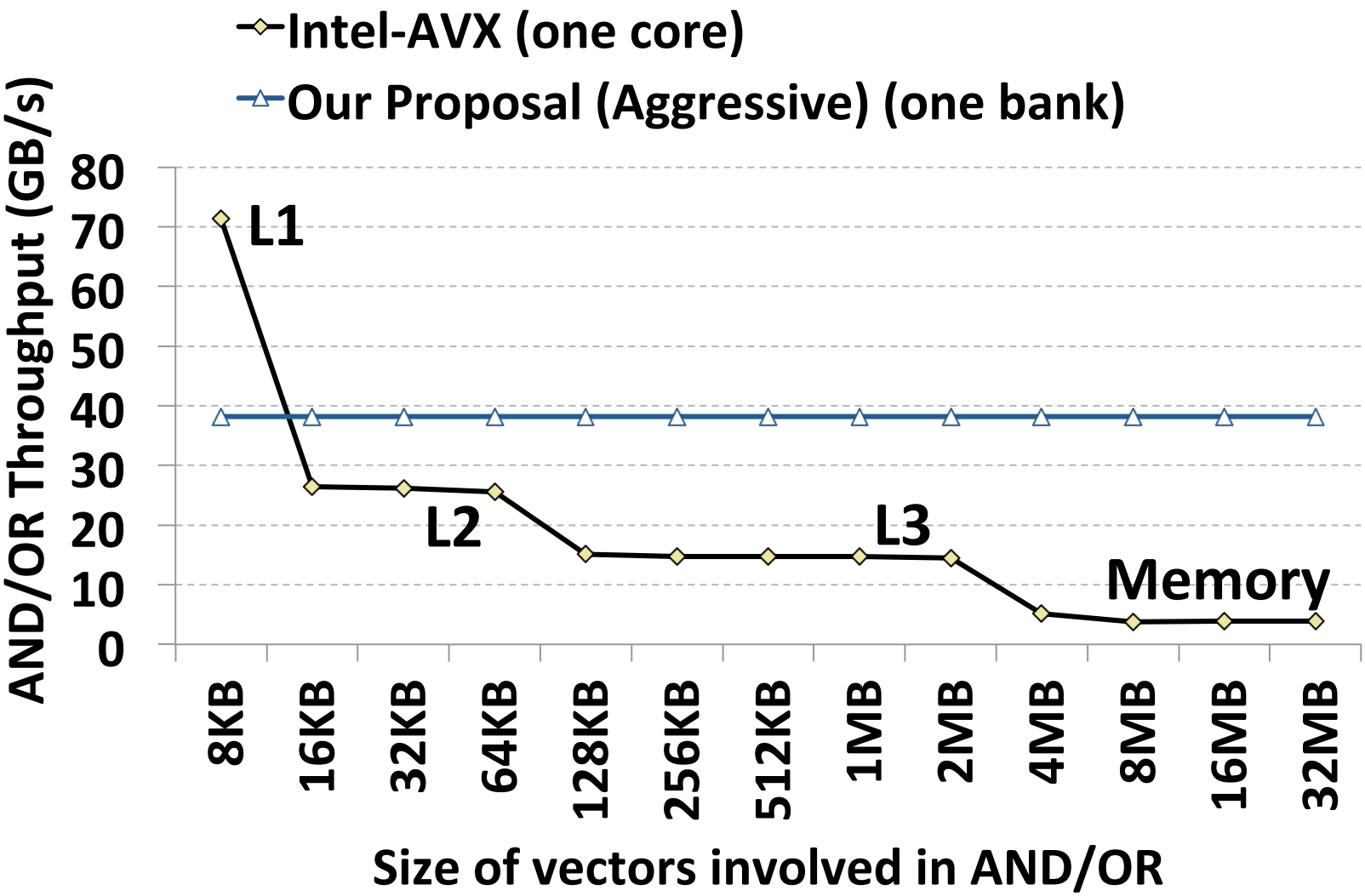
In-DRAM Bitwise AND/OR

Required Operation: Perform a bitwise AND of two rows *A* and *B* and store the result in *C*

- *R0* – reserved zero row, *R1* – reserved one row
- *D1, D2, D3* – Designated rows for triple activation

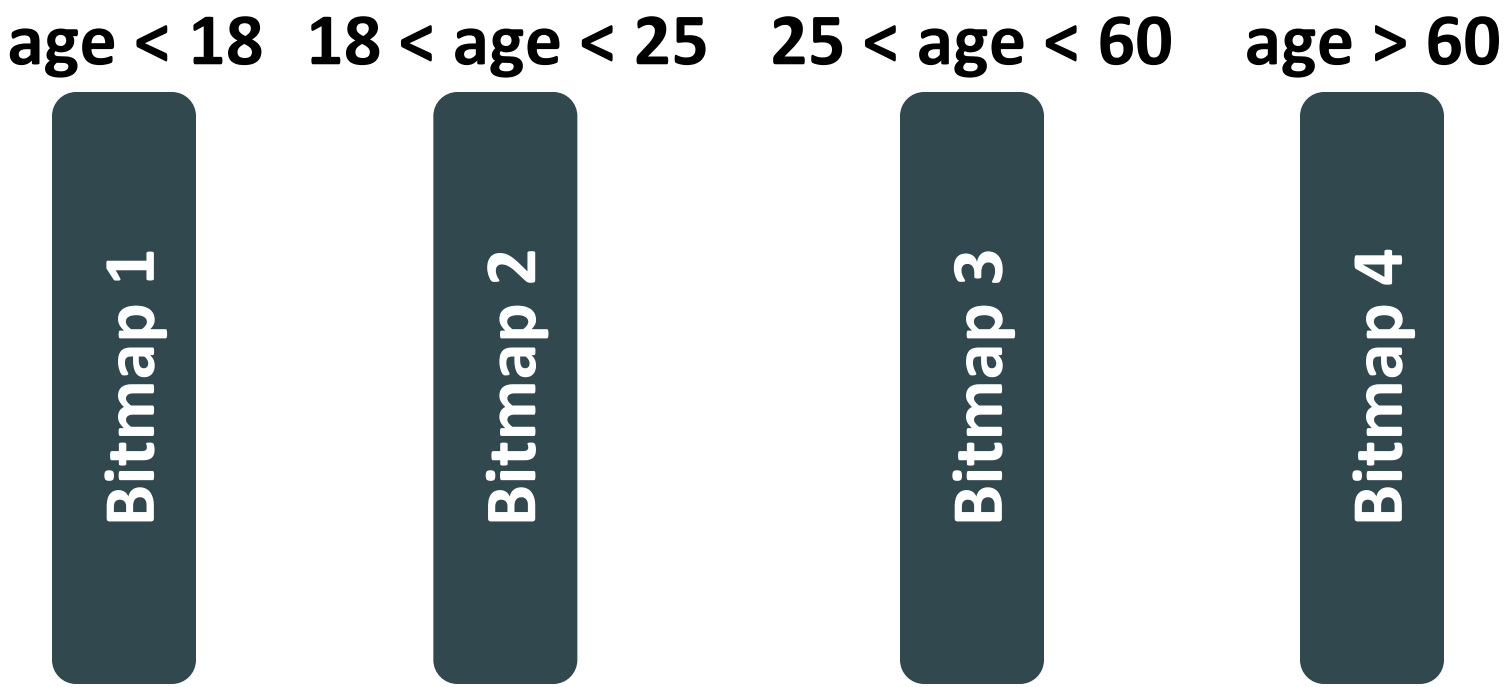
1. RowClone *A* into *D1*
2. RowClone *B* into *D2*
3. RowClone *R0* into *D3*
4. ACTIVATE *D1,D2,D3*
5. RowClone *Result* into *C*

Throughput Results

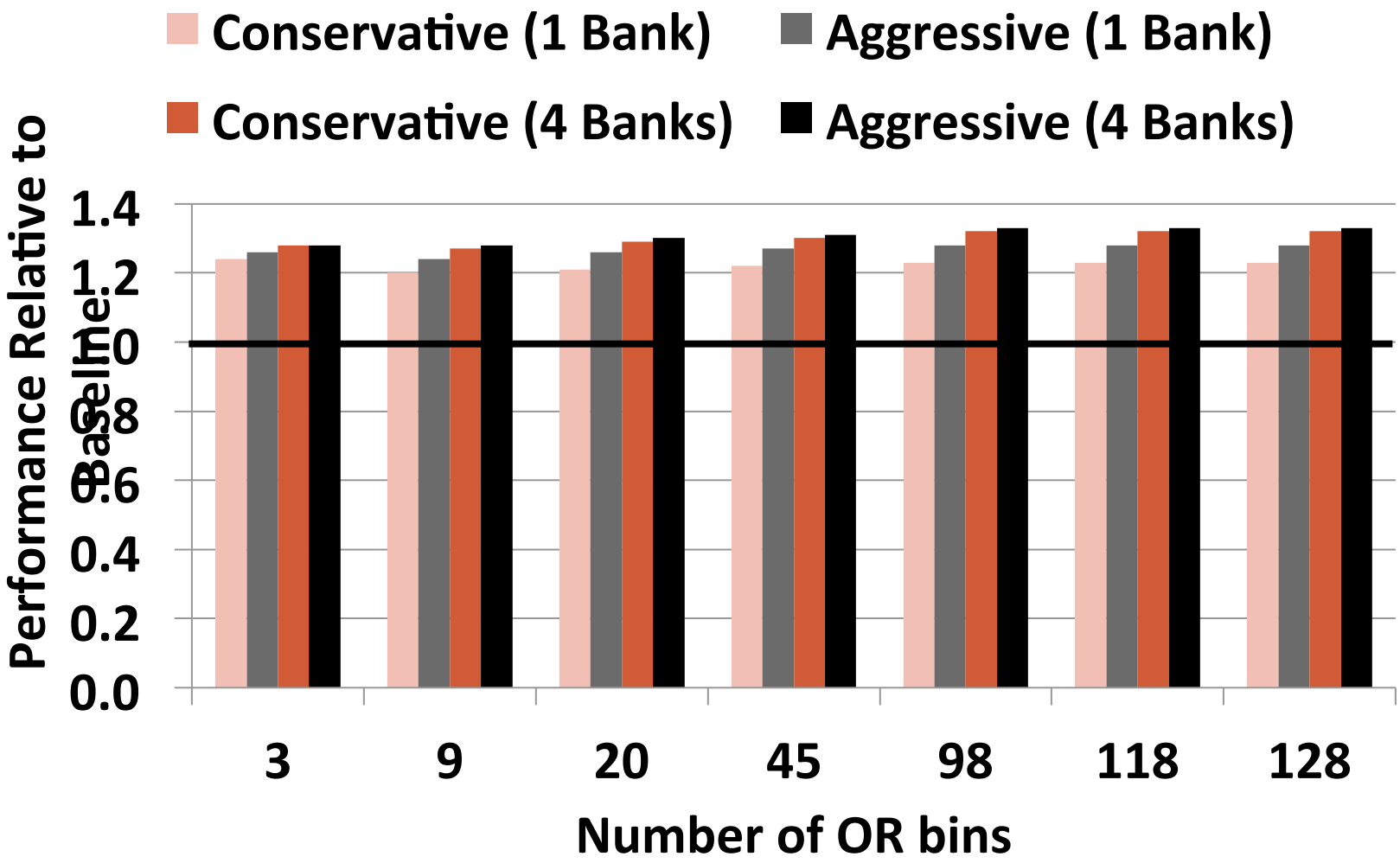


Bitmap Index

- Alternative to B-tree and its variants
- Efficient for performing *range queries* and *joins*



Performance Evaluation



Goals for This Lecture

- ✓ Understand DRAM technology
  - How it is built?
  - How it operates?
  - What are the trade-offs?

- ✓ Can we use DRAM for more than just storage?
  - In-DRAM copying
  - In-DRAM bitwise operations