Fast Bulk Bitwise AND and OR in DRAM

Vivek Seshadri, Kevin Hsieh, Amirali Boroumand, Donghyuk Lee, Michael A. Kozuch*, Onur Mutlu, Phillip B. Gibbons*, Todd C. Mowry
Carnegie Mellon University, *Intel Labs

Bulk Bitwise Operations

- Important component of modern-day programming
- Wide variety of applications (e.g., database, graphics)
- Enable more efficient algorithms *

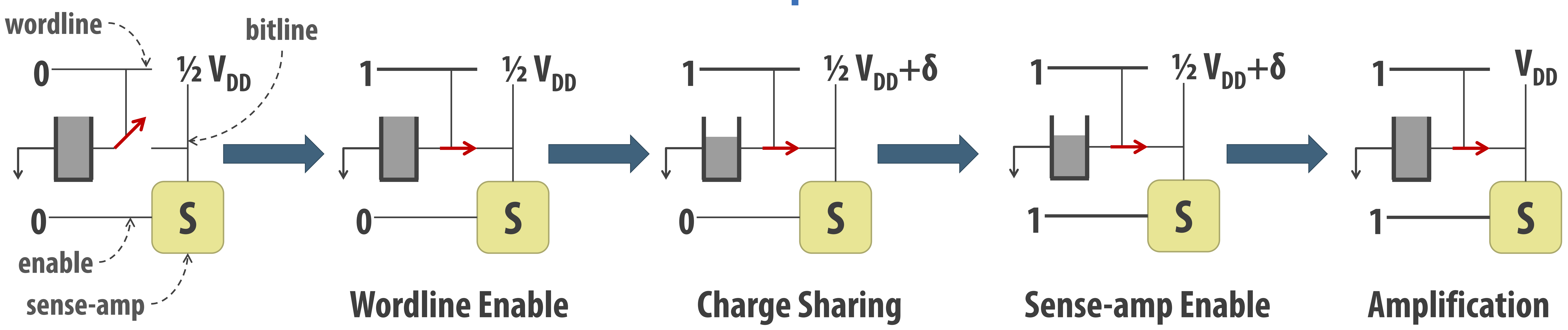
*D.E Knuth, Bitwise Tricks and Techniques, 2009

Existing systems require large data movement

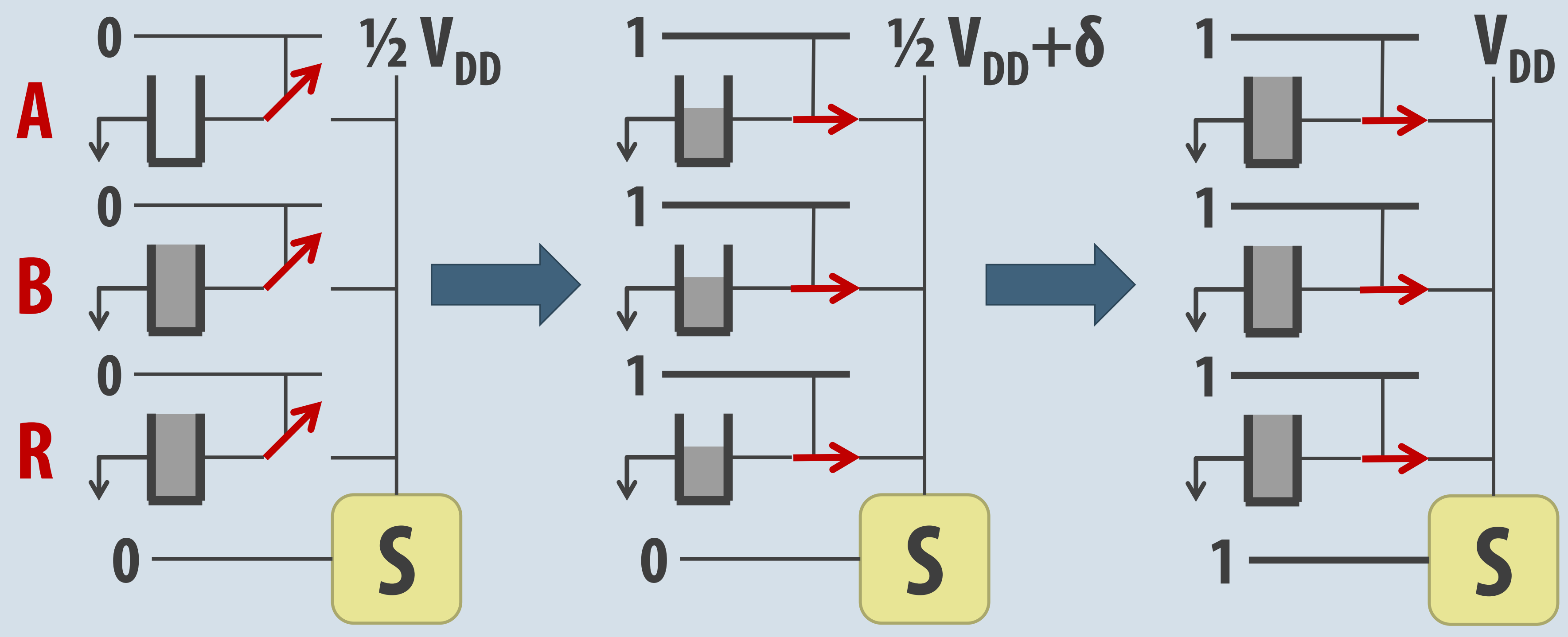
Processor ⇔ Main Memory

1. High Latency 2. High Bandwidth Consumption
3. High Energy Consumption

DRAM Cell Operation



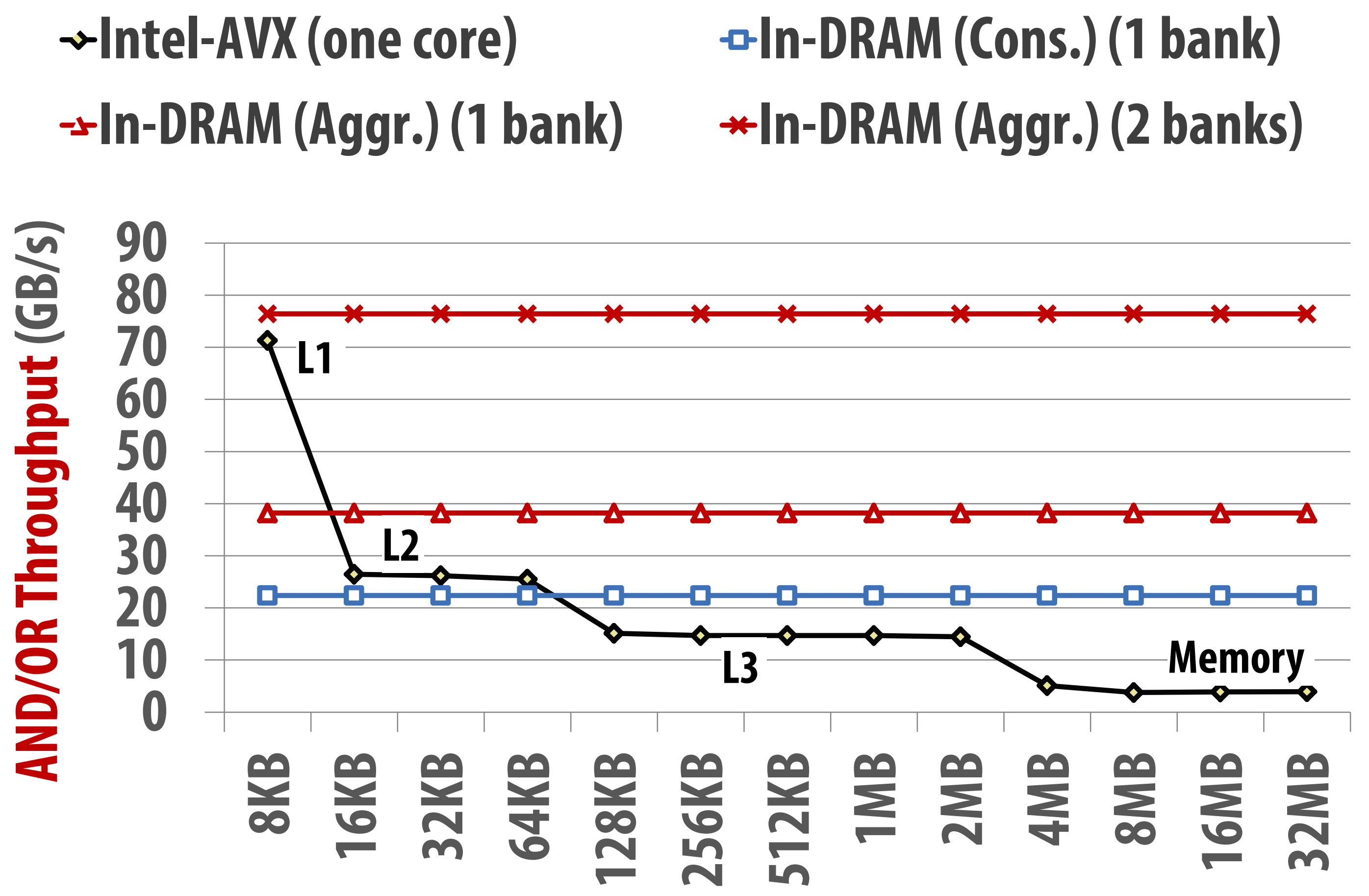
Triple Wordline Activation



Output = R (A **OR** B) + ~R (A **AND** B)

- Enables row-wide bitwise AND/OR operation
- Minimal changes to DRAM row decoder
- Support for RowClone (fast in-DRAM copy/init)

```
Algorithm for C = A (AND/OR) B
1. Copy src row A to temporary row D1
2. Copy src row B to temporary row D2
3. Init temporary row D3 with 0/1
4. Activate D1, D2, D3 simultaneously
5. Copy D1/D2/D3 to result row C
```



In-Memory Bitmap Indices

- Bitmap bins used to represent conditions (e.g., age < 18)
- Bitwise AND/OR of bitmaps used to evaluate query conditions!
- Performance depends on throughput of bitwise AND/OR

FastBit: Real world bitmap index library

- Index-intensive queries: 33% of execution spent in bitwise OR
- Average performance improvement using In-DRAM OR: 30%

Intel Science & Technology Center for Cloud Computing