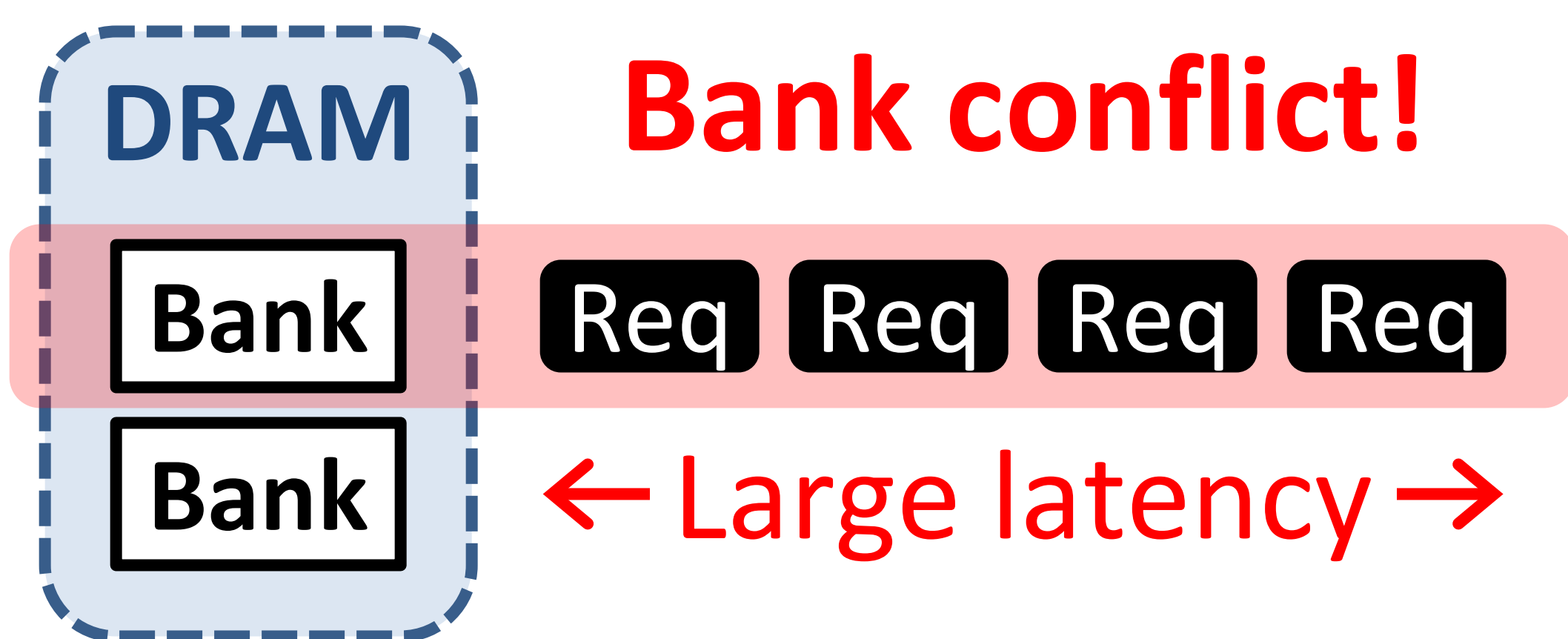


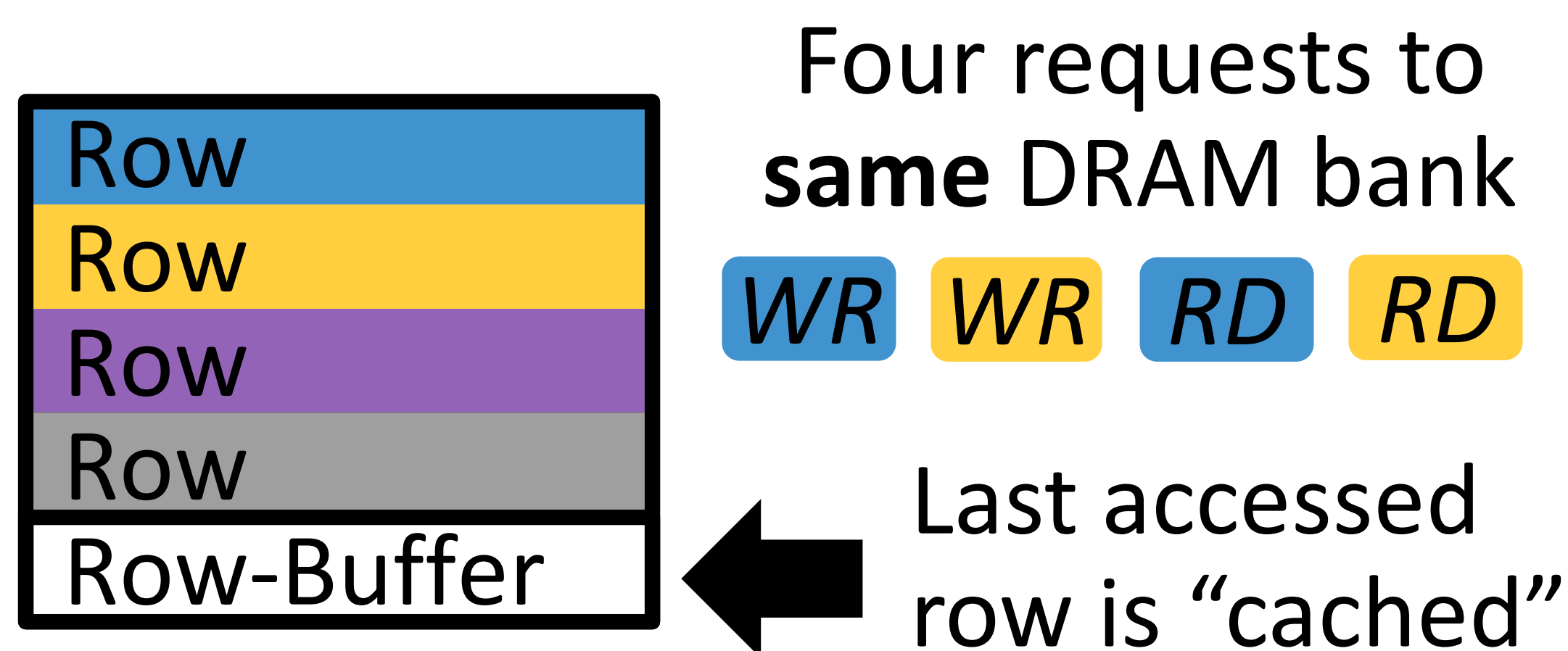
# A Case for Subarray-Level Parallelism (SALP) in DRAM

Yoongu Kim, Vivek Seshadri, Donghyuk Lee, Jamie Liu, Onur Mutlu (CMU)

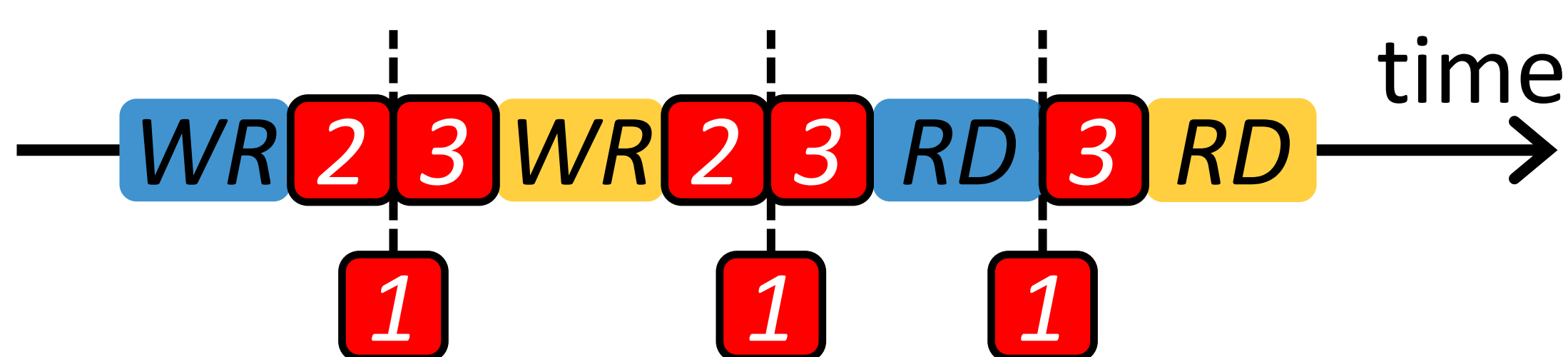
## 1. DRAM Bank Conflicts



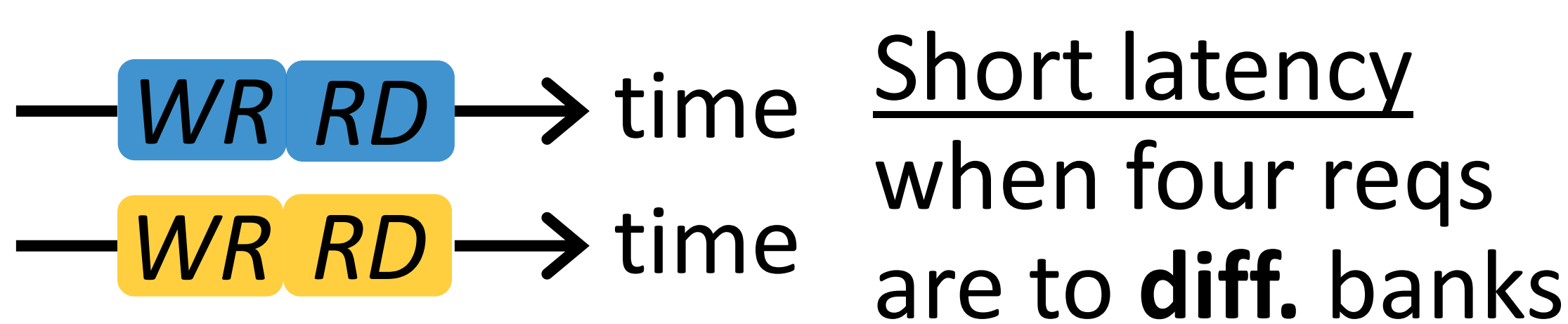
## 2. Timeline of DRAM Bank Conflicts



Large latency due to **3 problems**:

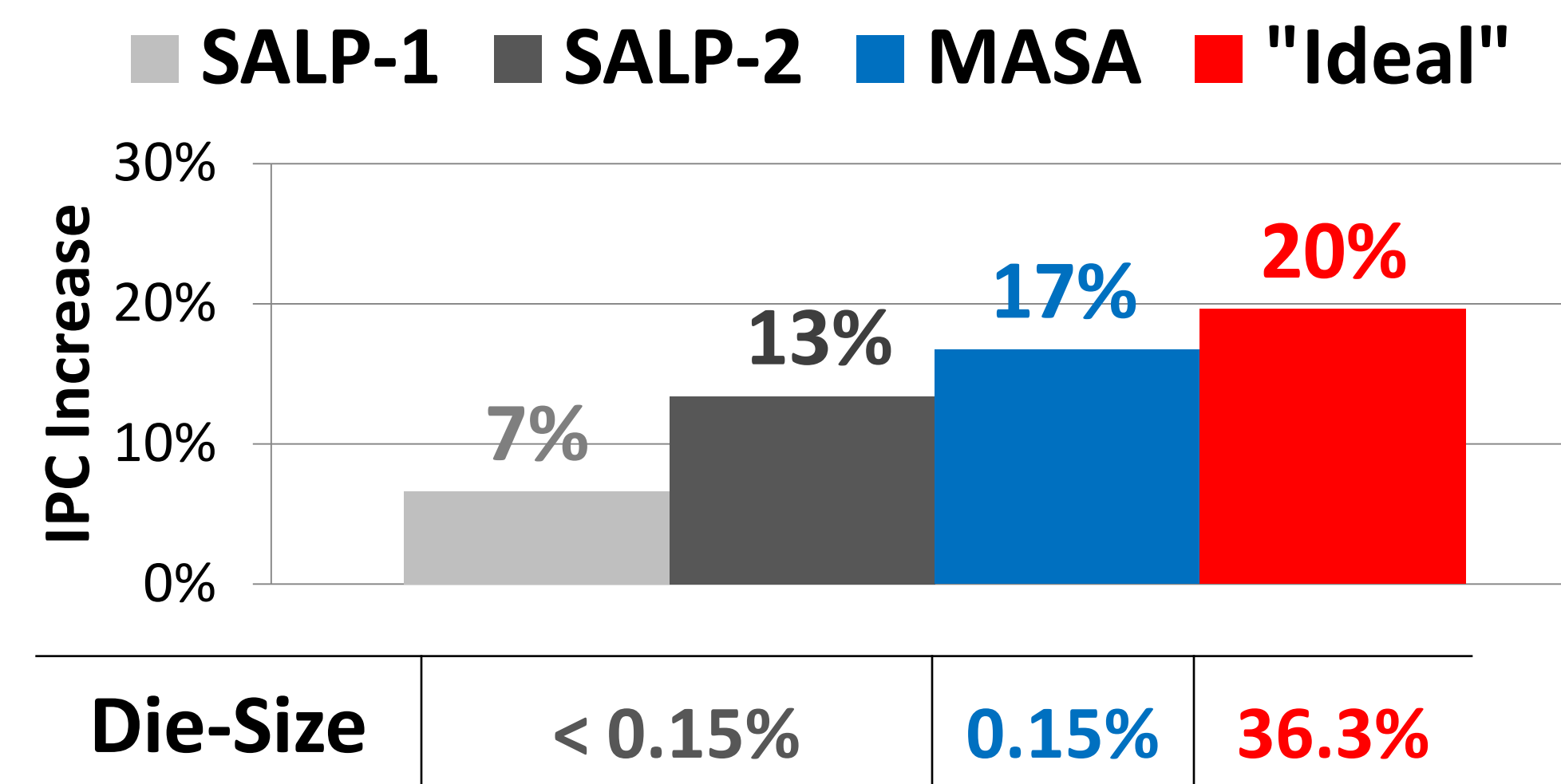
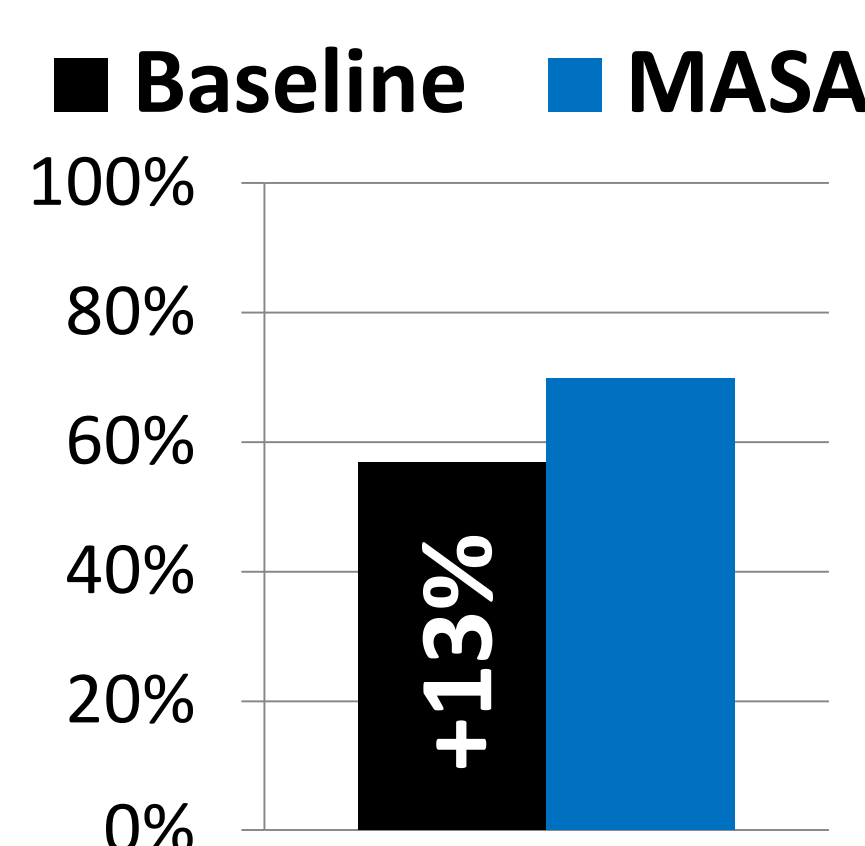
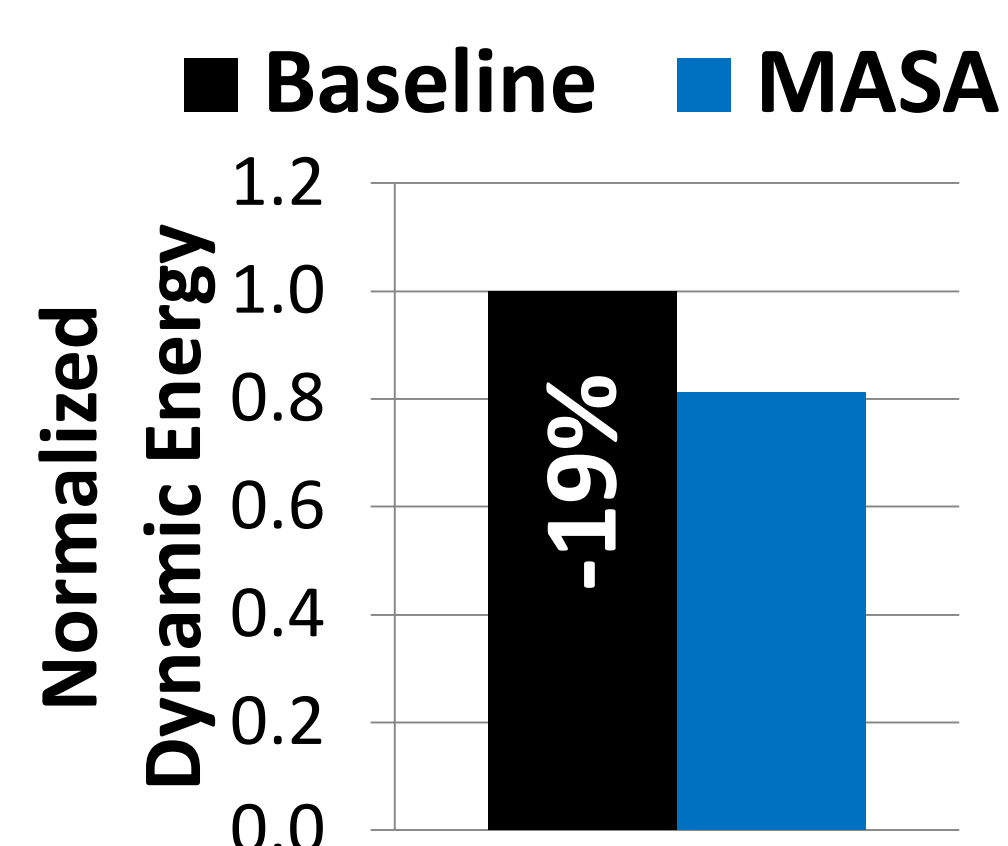


- 1 Serialization of requests
- 2 Write penalty after WR request
- 3 "Thrashing" of row-buffer



## 3. Our Goal

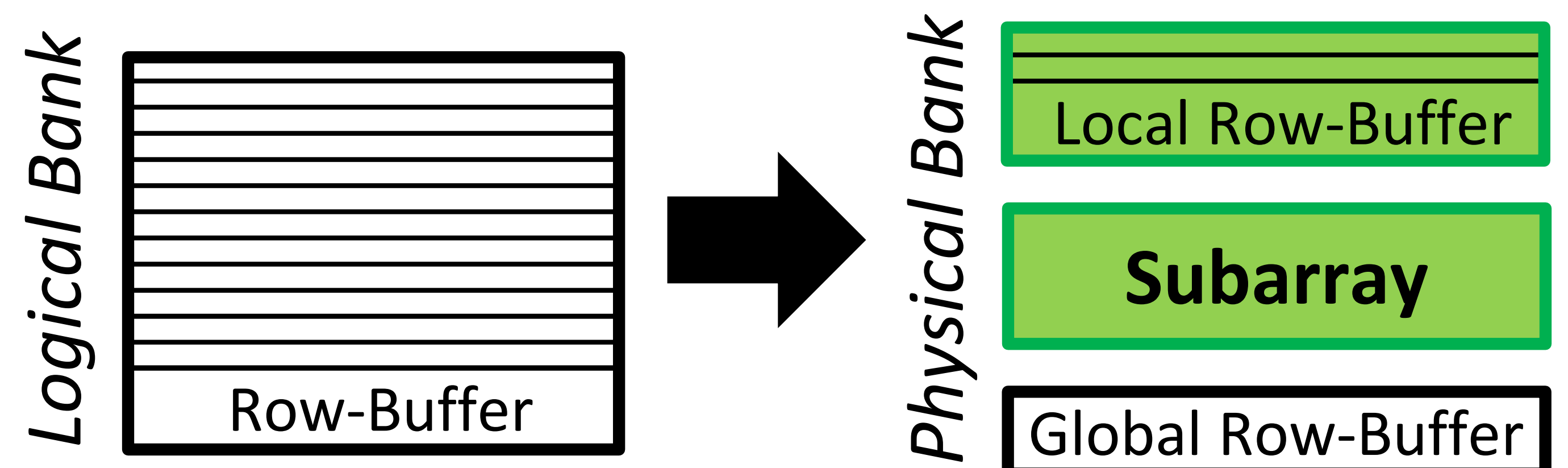
- Goal: Cost-effectively mitigate the detrimental effects of bank conflicts
- **Naïve Solution: Simply add more banks**  
- Very expensive



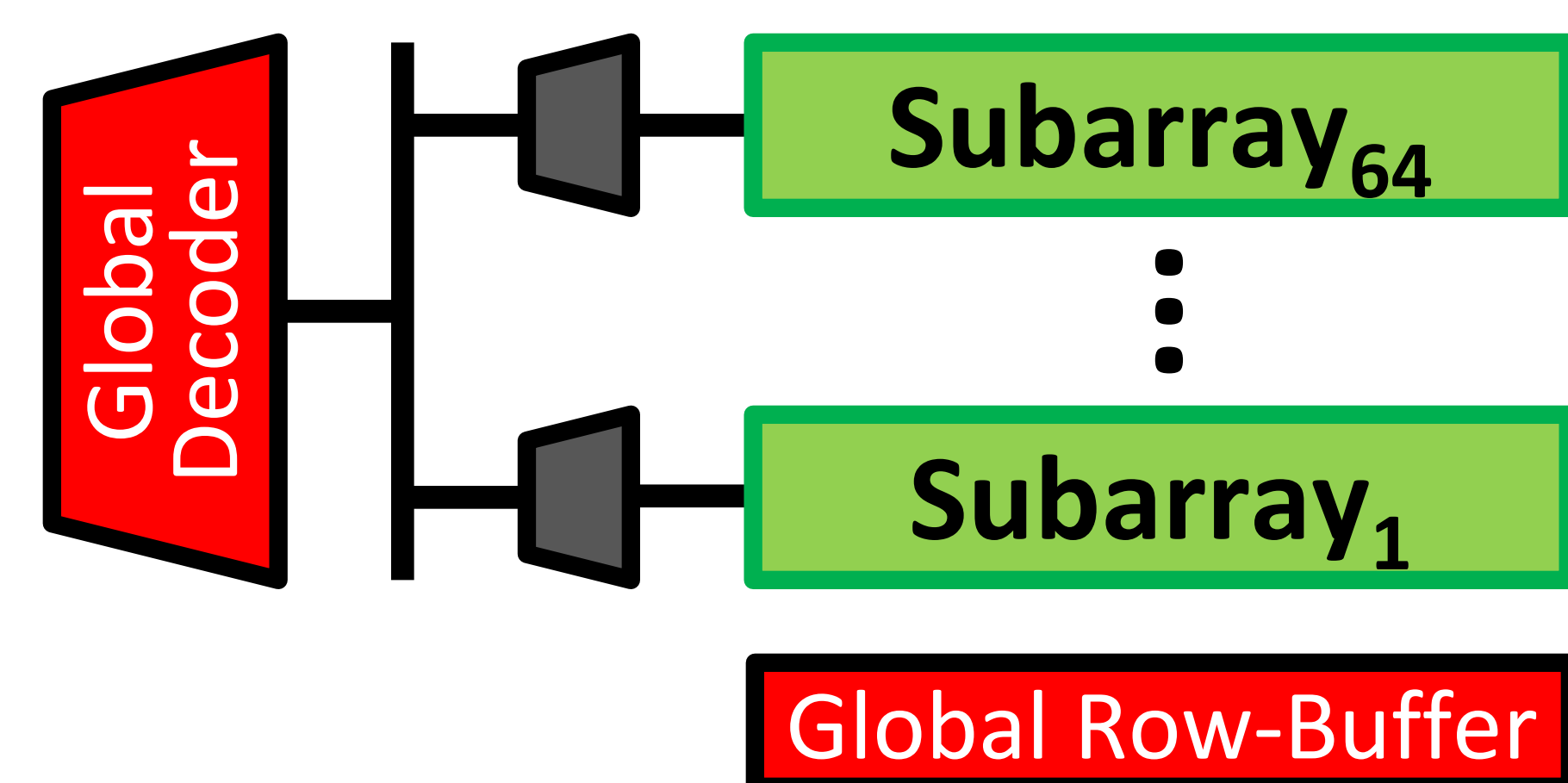
Die-Size: < 0.15%, 0.15%, 36.3%

## 4. Two Key Observations

1. A DRAM bank is divided into **subarrays**  
- Each subarray has a **local row-buffer**



2. **Subarrays are mostly independent...**  
- Except when sharing **global structures**



## 5. Key Idea

Reduce the sharing of...

1. **Global decoder**: enable parallel access to multiple subarrays
2. **Global row-buffer**: utilize multiple local row-buffers concurrently

## 6. Mechanism: MASA

Multitude of Activated SubArays  
Add two latches to each subarray

1. **Subarray Address Latch**  
- Stores per-subarray row-address
2. **Designated-Bit Latch**  
- Connects one subarray's local row-buffer to global row-buffer