RowClone: Fast and Energy-efficient In-DRAM Bulk Data Copy and Initialization

Vivek Seshadri, Yoongu Kim, Chris Fallin, Donghyuk Lee, Rachata Ausavarungnirun, Gennady Pekhimenko, Yixin Luo, Onur Mutlu, Phillip B. Gibbons*, Michael A. Kozuch*, Todd C. Mowry (CMU, *Intel Labs)

MEMORY CHANNEL – PERF./ENERGY BOTTLENECK

- **Limited Bandwidth**
- **Increasing number of cores**
- **High energy consumption (20-42% per access)**

Goal: **Reduce unnecessary data movement on the memory channel to improve performance and energy-efficiency**

SHORTCOMINGS OF CURRENT APPROACH

Large data transfer over the memory bus
1. High latency (data transferred 64B at a time)
2. High bandwidth (contention for other applications)
3. High energy

Our Approach: **Perform copy/initialization in DRAM**

ROWCLONE – FAST PARALLEL MODE

Use row-buffer to copy entire row of data
1. Copy from source row to row-buffer
2. Copy from row-buffer to destination row
Can be implemented using back-to-back ACTIVATE command

Limitations:

- **Requires source and destination to share row-buffer**
- **Cannot partially copy data from a row**

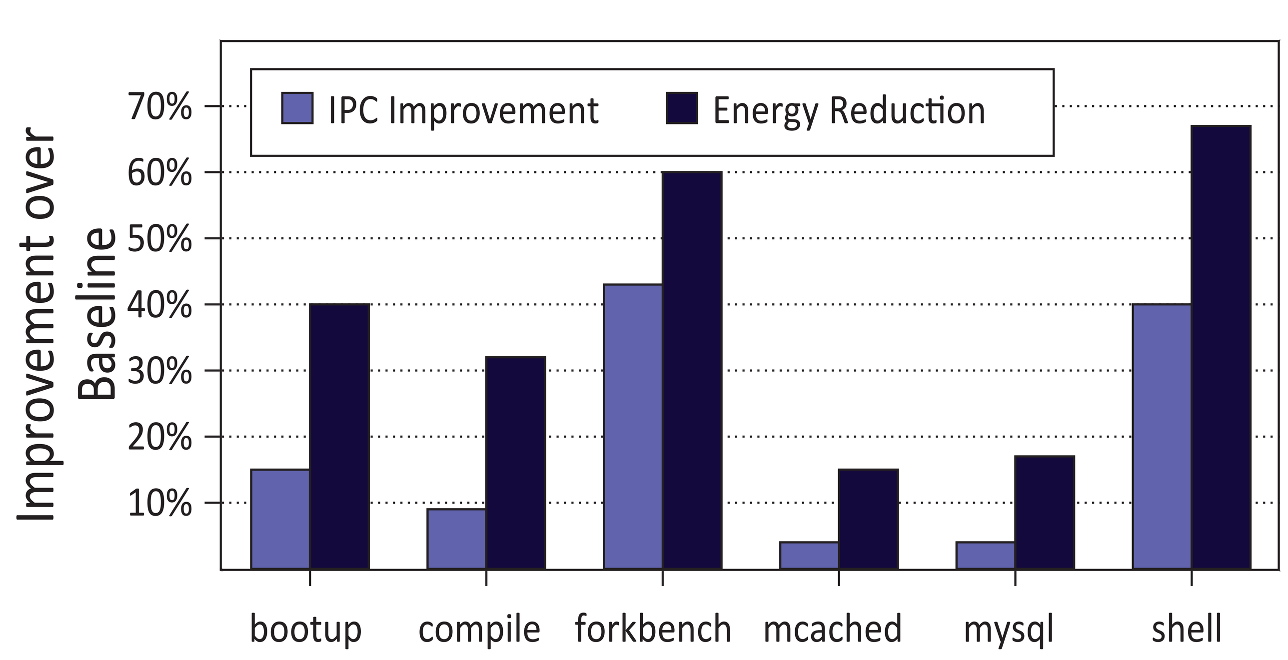
ROWCLONE – PIPELINED SERIAL MODE

Use shared internal bus to copy cache lines
1. Put source bank in read mode
2. Put destination bank in write mode
3. Transfer a cache line using internal bus

Overlaps latency of read and write
0.01% DRAM die area overhead

RESULTS: SINGLE-CORE

Simulation using cycle-level CPU simulator coupled with DDR3 DRAM simulator



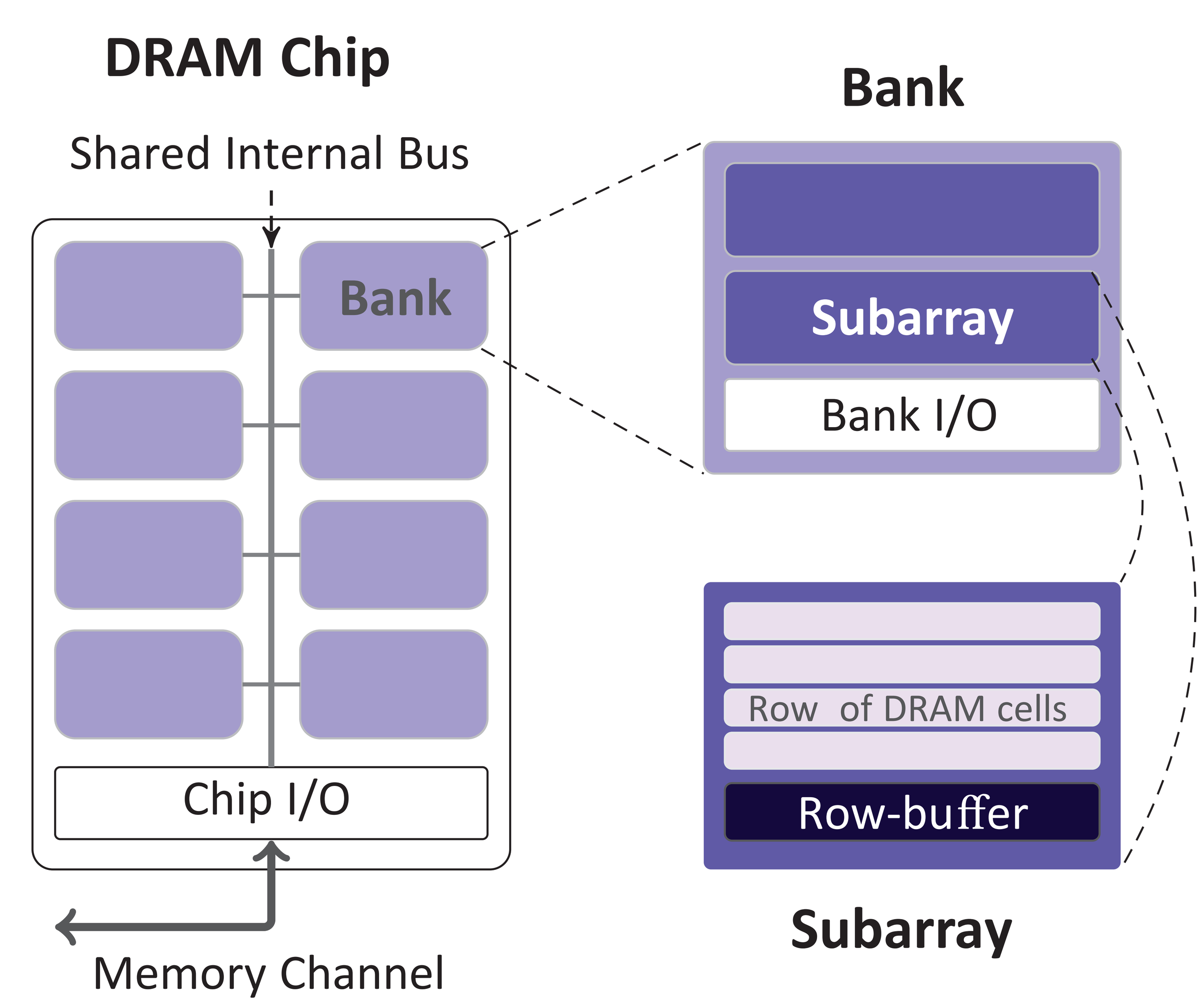
International Symposium on Microarchitecture, 2013

BULK COPY AND INITIALIZATION

- **Process forking**
- **Initializing large data structures**
- **Secure deallocation**
- **Process checkpointing**
- **Virtual machine cloning/deduplication**
- **Page migration**
- **CPU-GPU communication**

Problem: **These operations degrade overall system performance**

DRAM CHIP ORGANIZATION

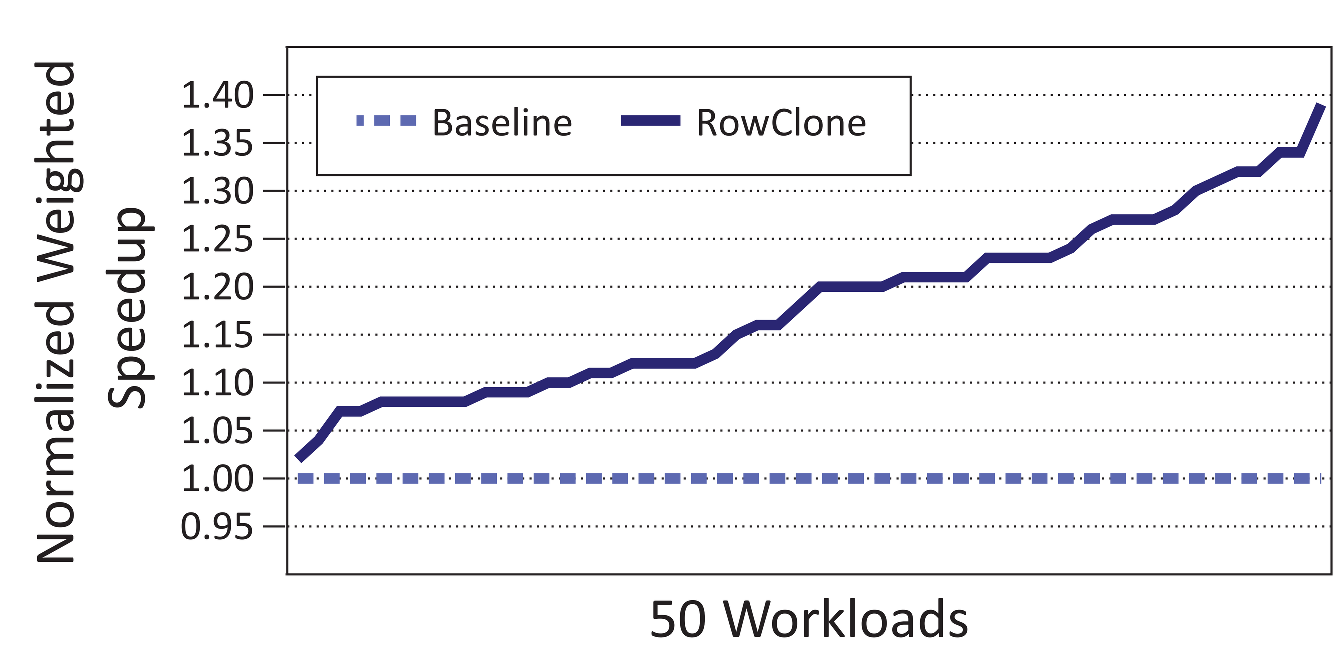


1. Single byte access requires entire row transfer to row-buffer
2. Reads and Writes to all banks use shared internal bus

END-TO-END SYSTEM DESIGN

- **ISA: new instructions - memcpy and meminit**
- **uArch: Ensure cache coherence**
- **OS: Subarray-aware page mapping, minimum size copy/init**

RESULTS: 4-CORE



- **17% reduction in Energy/Instruction**

Intel Science & Technology Center for Cloud Computing

Carnegie Mellon University · Georgia Tech · intel · PRINCETON UNIVERSITY · UC Berkeley · UNIVERSITY of WASHINGTON