DRAF: A Low-Power DRAM-based Reconfigurable Acceleration Fabric

Mingyu Gao, Christina Delimitrou, Dimin Niu, Krishna Malladi, Hongzhong Zheng, Bob Brennan, Christos Kozyrakis

Stanford | ENGINEERING
Electrical Engineering

ISCA – June 22, 2016

FPGA-Based Accelerators

- Improve performance and energy efficiency
- Good balance between flexibility (CPUs) and efficiency (ASICs)

- Recently used for many datacenter apps
  - Image/video processing, websearch, neural networks, …

Pictures: Putnam, et al. A Reconfigurable Fabric for Accelerating Large-Scale Datacenter Services. ISCA’14

Motivation

- Deploy FPGAs in cost & power constrained systems

- Datacenter systems
  - *High-density* FPGAs for large accelerators for multiple apps
  - *Low-power* FPGAs to simplify integration in servers and racks

- Mobile systems
  - *High-density* FPGAs for accelerators for multiple apps
  - *Low-power* FPGAs for low cost and long battery life

DRAF in a Nutshell

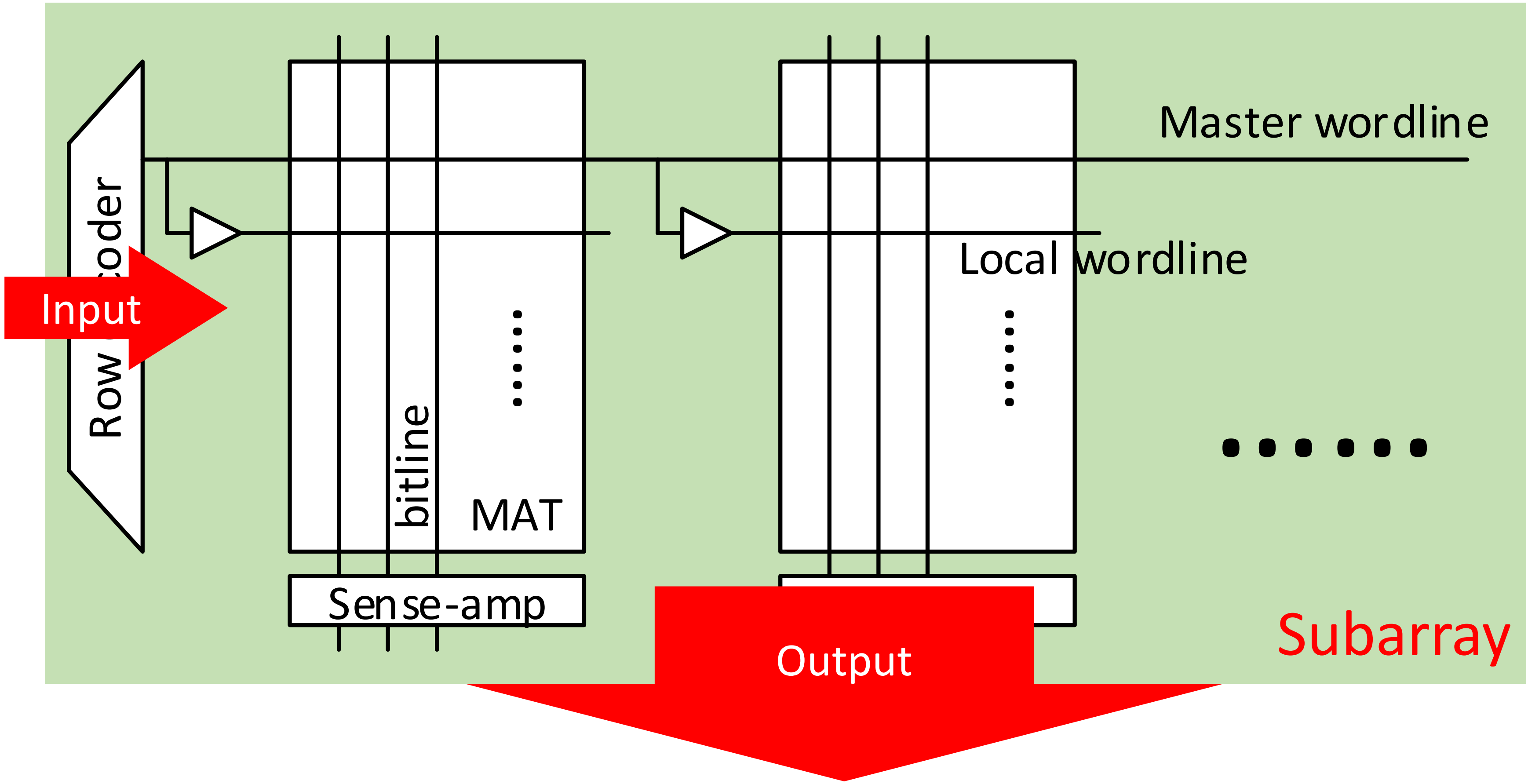
- A *high-density & low-power* FPGA
  - Bit-level reconfigurable, just like conventional FPGAs

- Uses dense *DRAM technology* for lookup tables
  - Replacing the SRAM technology in conventional FPGAs

- DRAF vs. FPGA
  - 10 – 100x logic density
  - 1/3 power consumption
  - Multi-context support with fast context switch

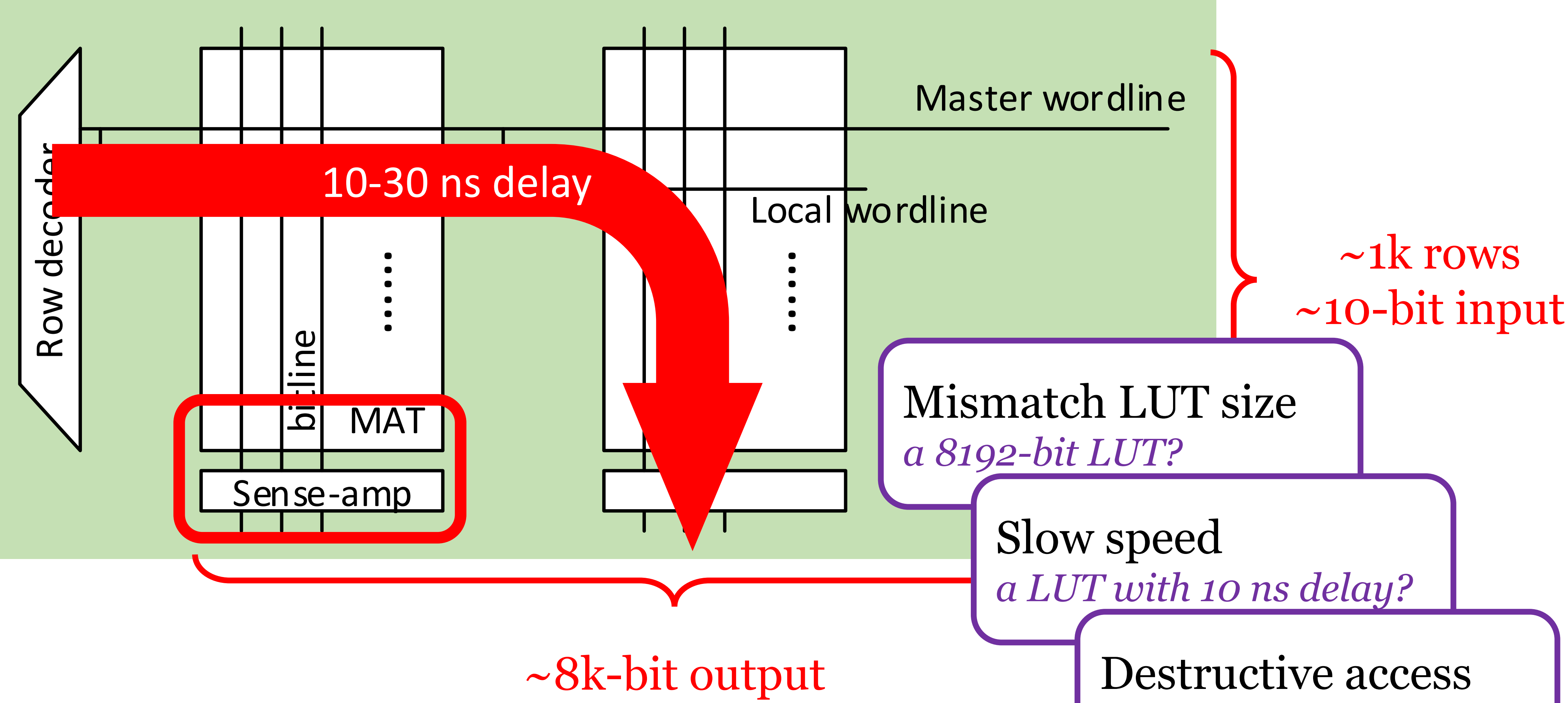
Challenges of Building DRAM-based FPGAs

DRAM Array Structure



A DRAM subarray is naturally a lookup-table

Challenges



Mismatch LUT size
a 8192-bit LUT?

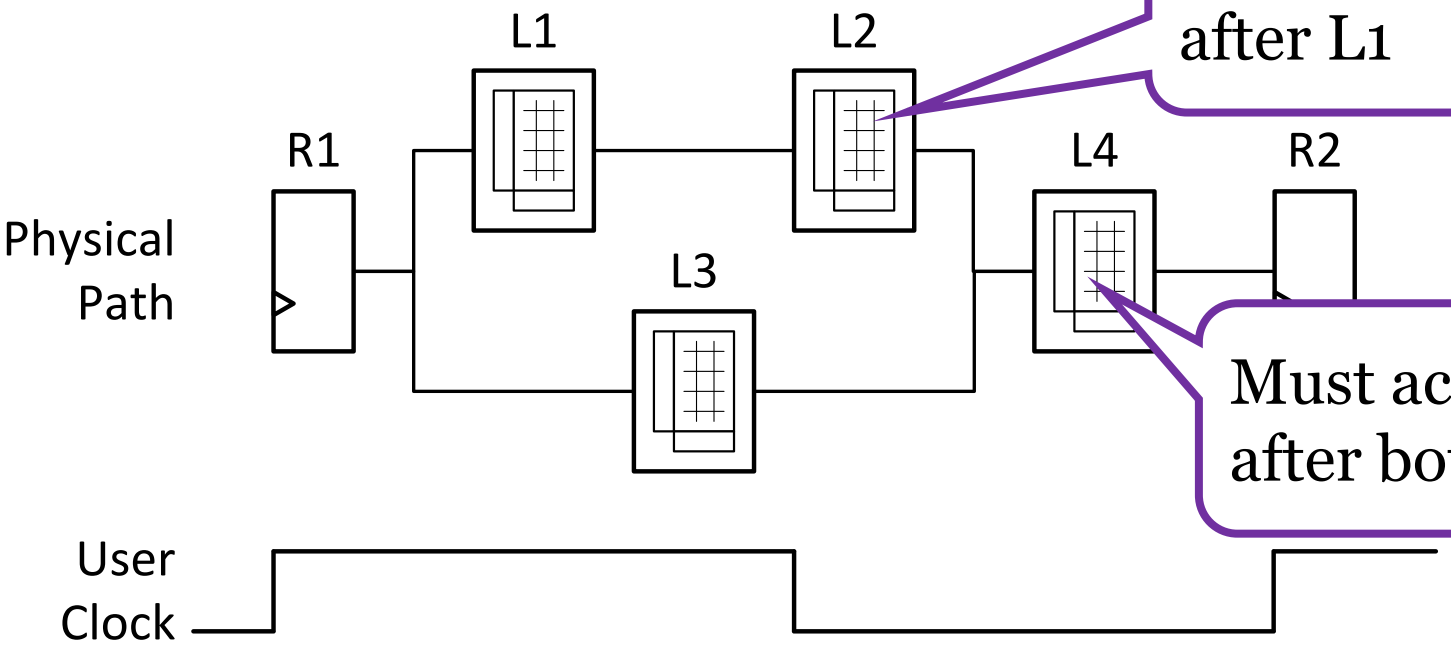
Slow speed
a LUT with 10 ns delay?

Destructive access
data lost after access?

Destructive Access

- Explicit activation, restoration, and precharge operations
  - Longer access delay due to serialization

- Issue of LUT chaining: order of LUT access



Must activate L2 after L1

Must activate L4 after both L2 & L3

DRAF Architecture

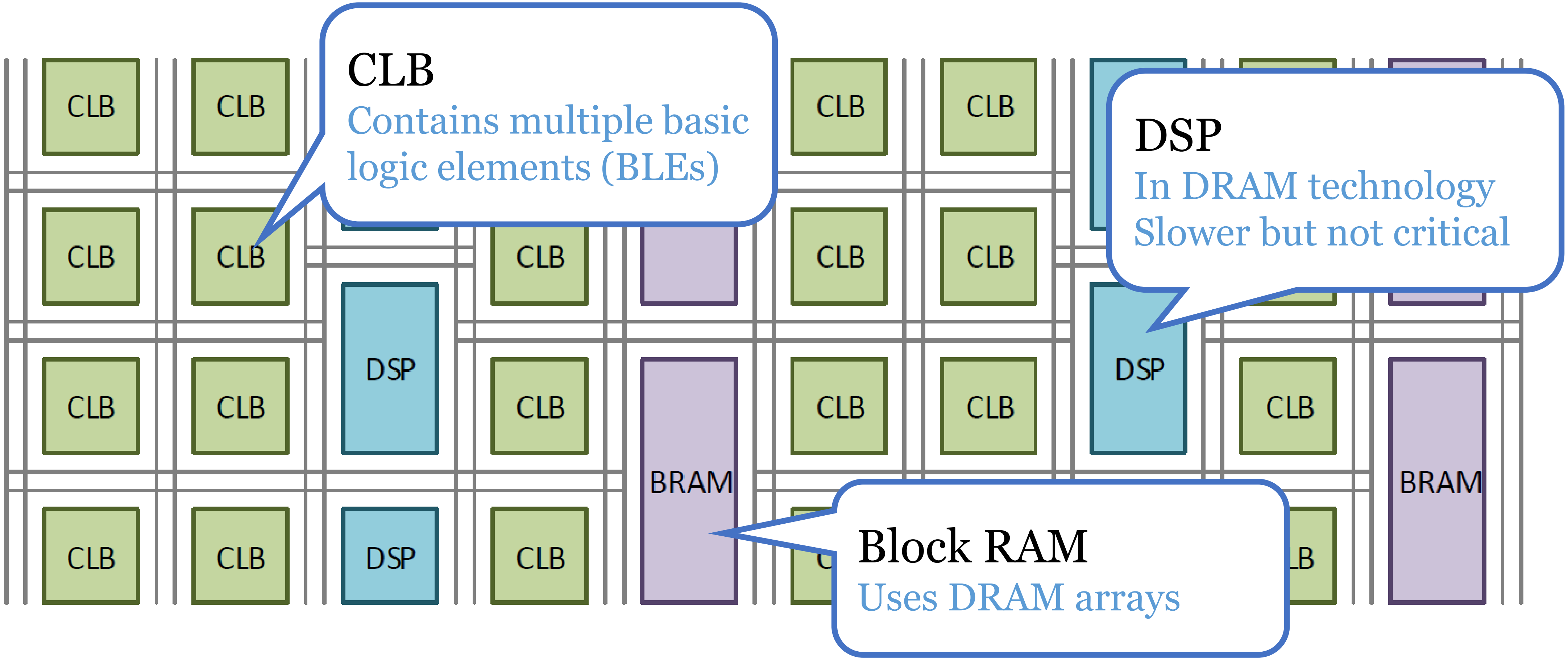
Basic Logic Element

Multi-Context Support

Timing

DRAF Overview

- Same island layout and configurable interconnect as FPGA

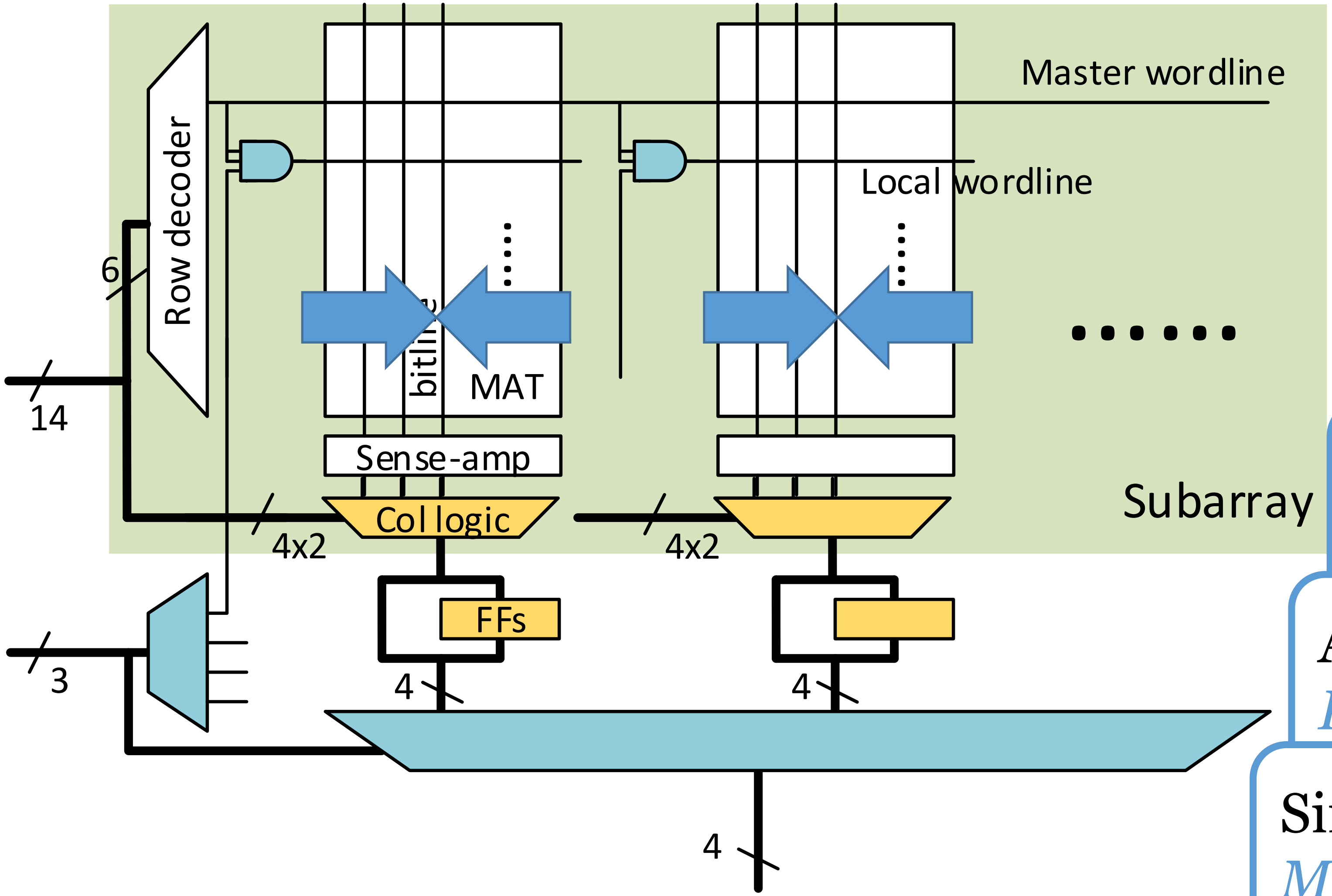


CLB
Contains multiple basic logic elements (BLEs)

DSP
In DRAM technology
Slower but not critical

Block RAM
Uses DRAM arrays

Basic Logic Element



7-10 bits input

2-4 bits output

Narrower MAT
1k bits to 8-16 bits

Specialized column logic
Better flexibility

Additional FFs & MUXs
Registering & retiming

Single-MAT access
Multi-context

Multi-Context Support

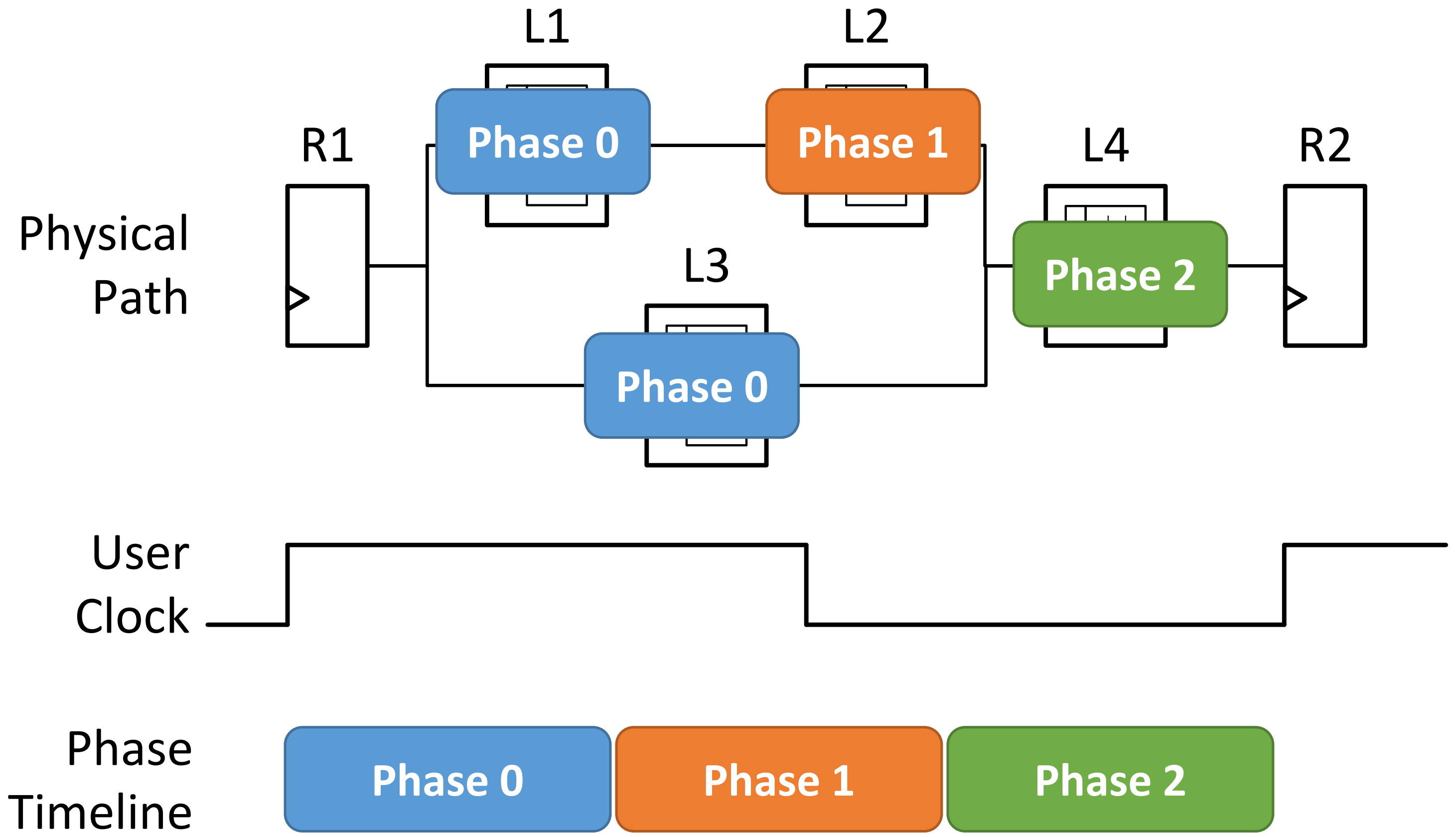
- DRAF supports 8-16 contexts per chip
    - Context: one MAT per BLE
    - Efficient use of MATs with little area and power overhead

- Instant switch between active contexts
    - Similar to context-switch between processes on CPU

- Context uses
    - One context per accelerator design or application
    - One context per part of a very large accelerator design

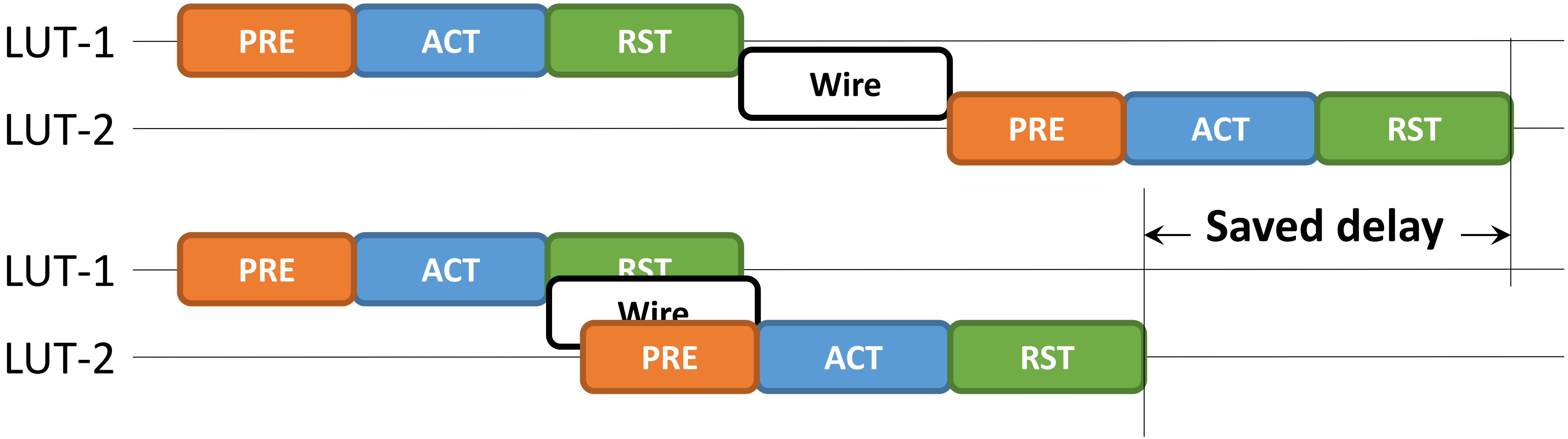
Timing – Destructive Access

- Issue of LUT chaining: order of LUT access
- Solution: *phase* – similar to critical path finding



Timing – Latency Optimization

- Issue: precharge and restore delays
- Solution: *3-way delay overlapping*
  - Hide PRE/RST delays with wire propagation delay
- Performance gap between DRAF and FPGA reduces from >10x to 2-4x



Summary

- Challenges → solutions
  - Mismatch LUT size → *multi-context BLE*
  - Destructive access → *phase-based timing*
  - Slow speed → *3-way delay overlapping*

- Other design features (see paper)
  - Sense-amp as register
  - Time-multiplexed routing
  - Handling DRAM Refresh

Evaluation

Area, power, performance against FPGA and CPU

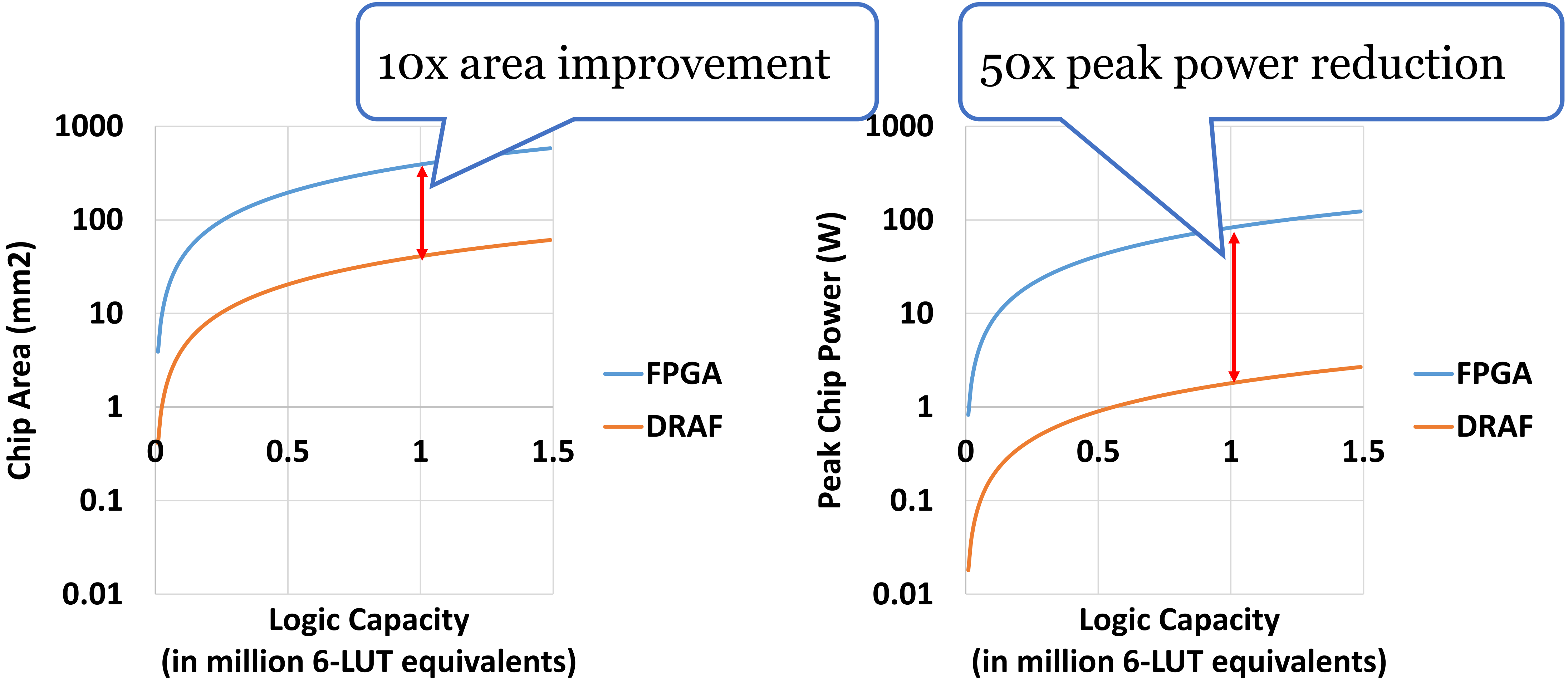
Methodology

- Synthesize, place & route with Yosys + VTR
- CACTI-3DD with 45 nm power and area models

- Comparisons
    - 70 mm^2 FPGA based on Xilinx Virtex-6
    - 70 mm^2 DRAF device, 8-context
    - Intel Xeon E5-2630 multi-core processor (2.3 GHz)

- 18 accelerator designs
    - MachSuite, Sirius, Vivado HLS Video Library, VTR benchsuite
    - Web service, image processing, analytics, neural networks, …

DRAF Chip Area & Power

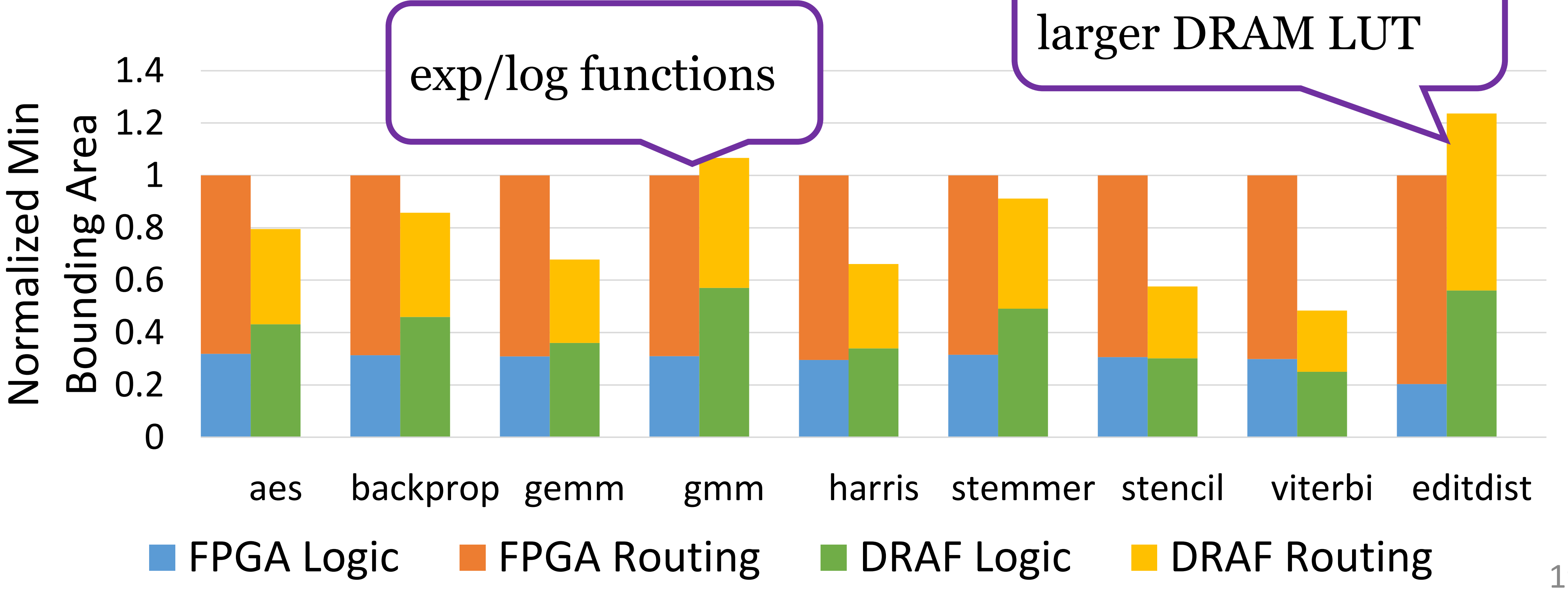


10x area improvement

50x peak power reduction

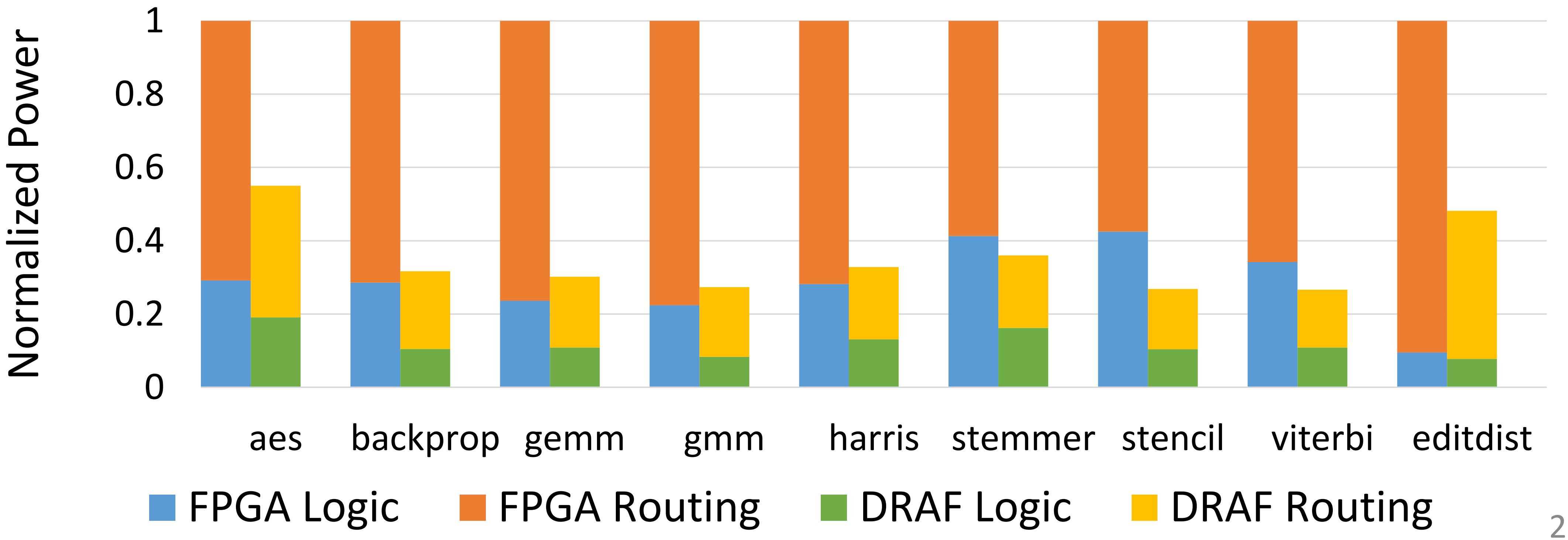
FPGA vs. DRAF (Area)

- 8-context DRAF occupies 19% less area than 1-context FPGA
  - 10x area efficiency: 8 designs in less silicon area than 1 design before
  - But only one context can be active at a time



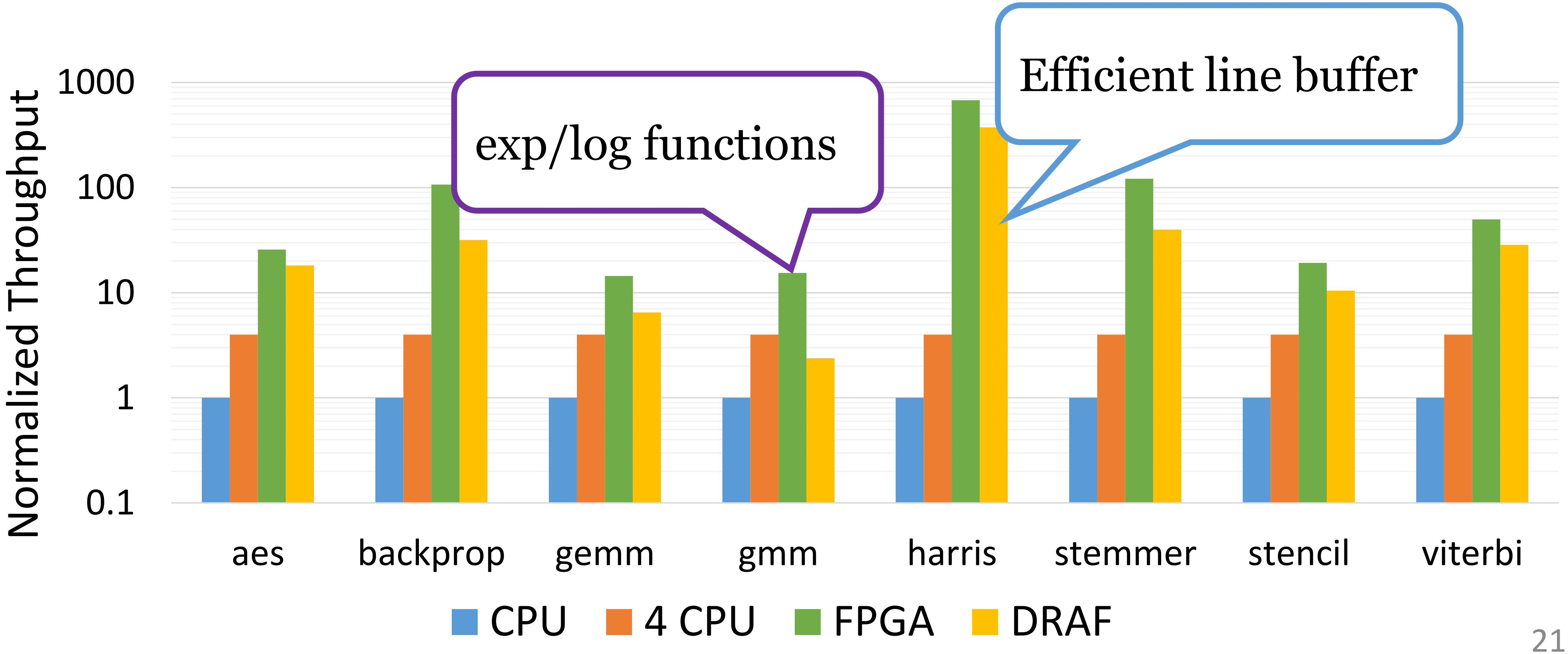
FPGA vs. DRAF (Power)

- Use one context in DRAF
- DRAF consumes 1/3 power of FPGA and 15% less energy
  - Note: current CAD tools are less efficient with DRAF



Performance

- DRAF is 2.7x slower than FPGA
- DRAF is 13.5x faster than CPU, 3.4x faster than ideal 4-core



exp/log functions

Efficient line buffer

Conclusions

- DRAF: high-density and low-power reconfigurable fabric
  - Based on dense DRAM technology
  - Optimized timing + multi-context support

- DRAF targets cost and power constrained applications
  - E.g., datacenters and mobile systems

- DRAF trades off some performance for area & power efficiency
  - 10x smaller area, 3x less power, and 2.7x slower than FPGA
  - Still 13x speedup over Xeon cores

Thanks!

Questions?

Stanford MAST

Ecosystem Research Development MSL

Memory Solutions Lab