Carnegie Mellon University ETH zürich NVIDIA TEXAS The University of Texas at Austin SAFARI

What Your DRAM Power Models Are Not Telling You:

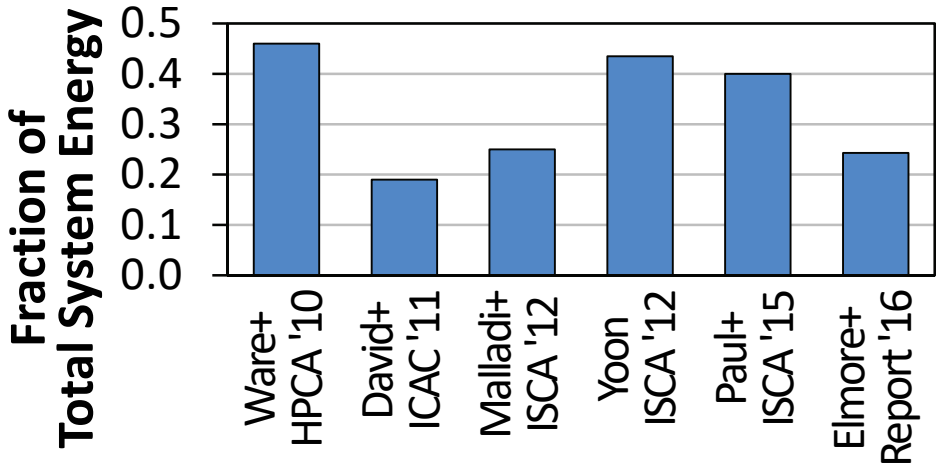
Lessons from a Detailed Experimental Study

Saugata Ghose, A. Giray Yağlıkçı, Raghav Gupta, Donghyuk Lee, Kais Kudrolli, William X. Liu, Hasan Hassan, Kevin K. Chang, Niladrish Chatterjee, Aditya Agrawal, Mike O'Connor, Onur Mutlu

June 21, 2018

DRAM Power Is Becoming a Major Design Concern

- Main memory in computers consists of DRAM modules
- DRAM consumes **up to half of total system power**



- State-of-the-art DRAM power models are not adequate
  - Based on **IDD values:** standard current measurements provided by vendors
  - Often have a **high mean absolute percentage error**
    - » 32% for DRAMPower
    - » 161% for Micron power model

OUR GOAL

Measure and analyze the power used by real DRAM, and build an accurate DRAM power model

Background: DRAM Organization & Operation

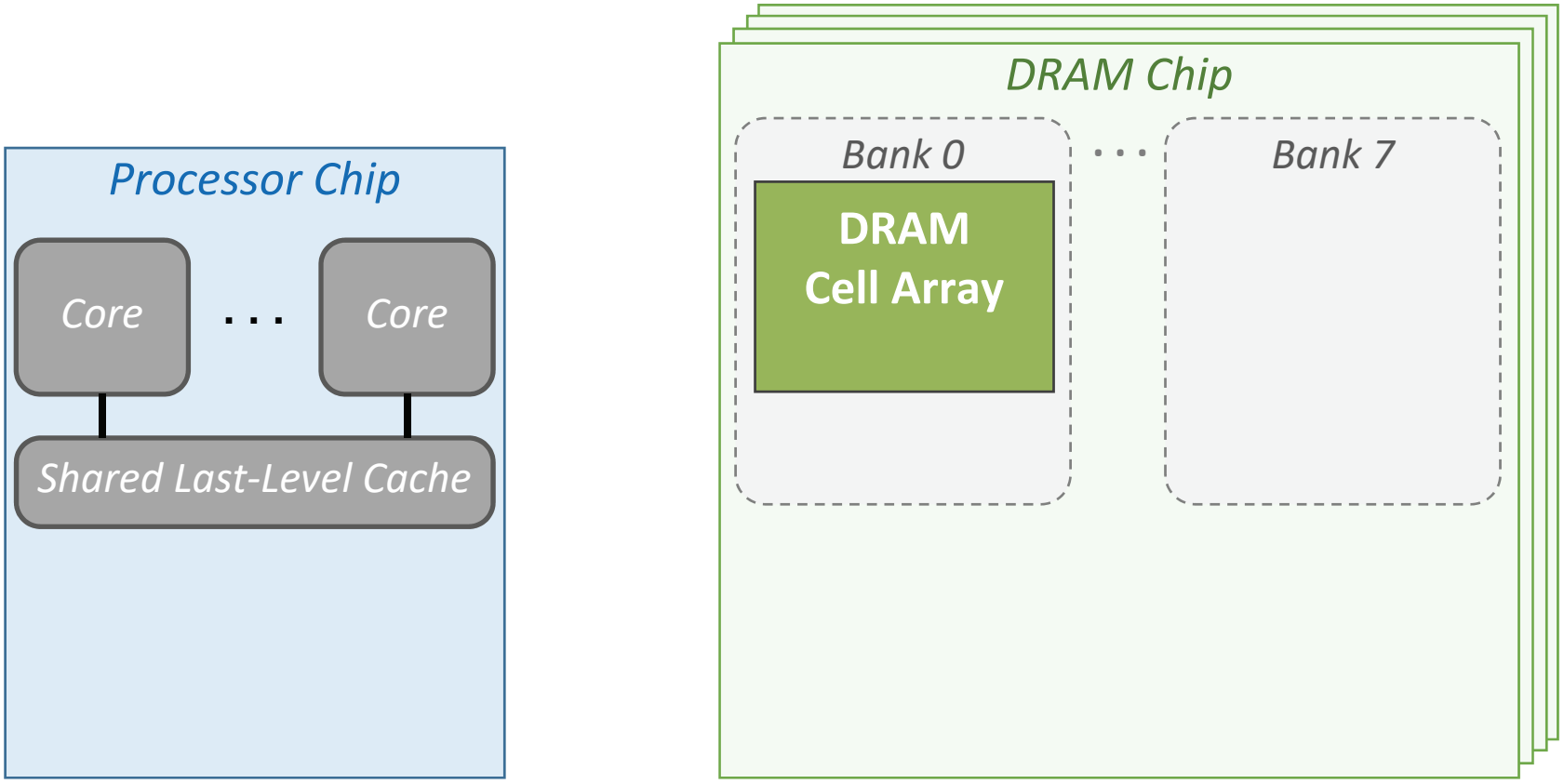
Characterization Methodology

New Findings on DRAM Power Consumption

VAMPIRE: A Variation-Aware DRAM Power Model

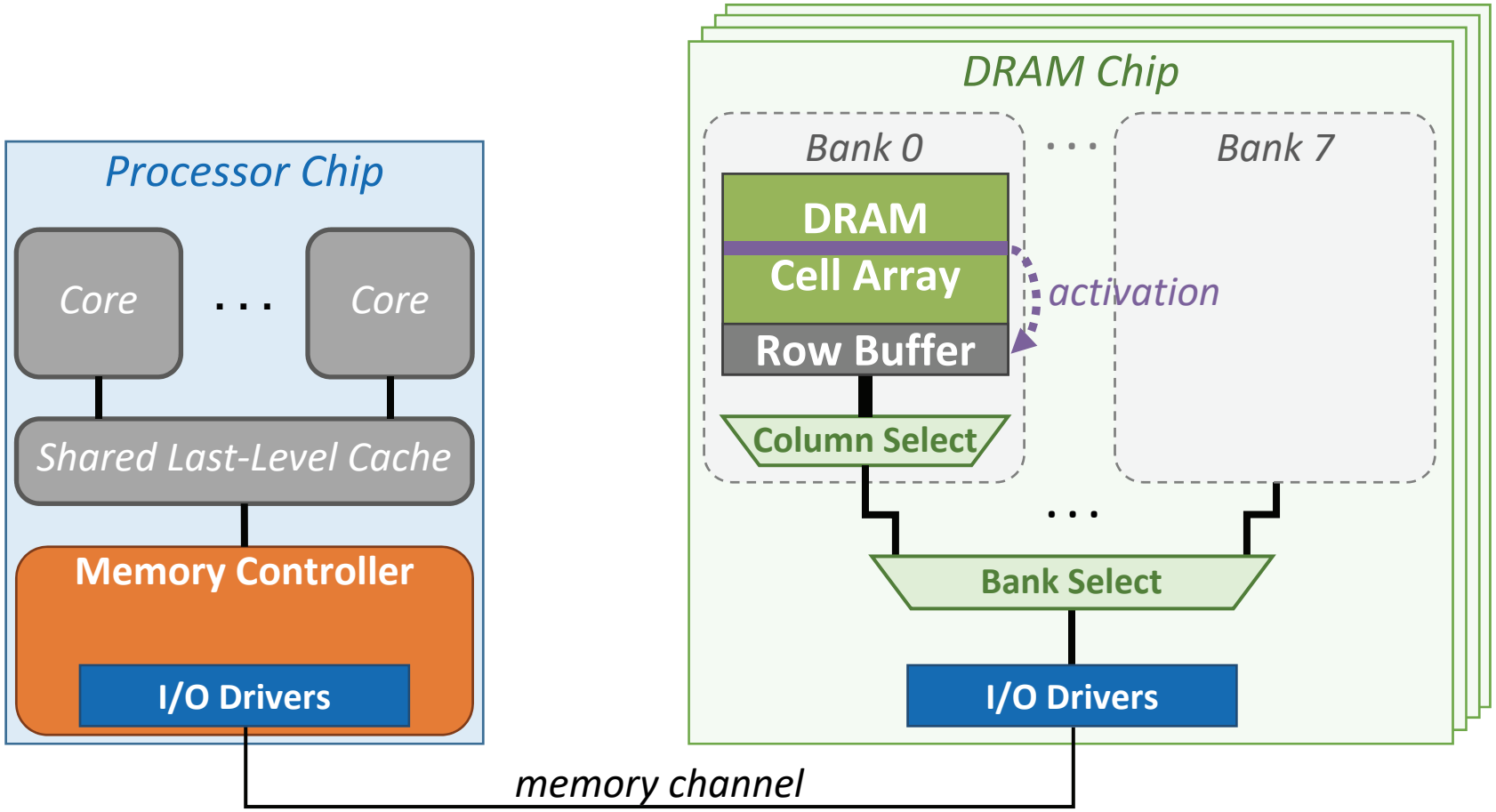
Conclusion

Simplified DRAM Organization and Operation



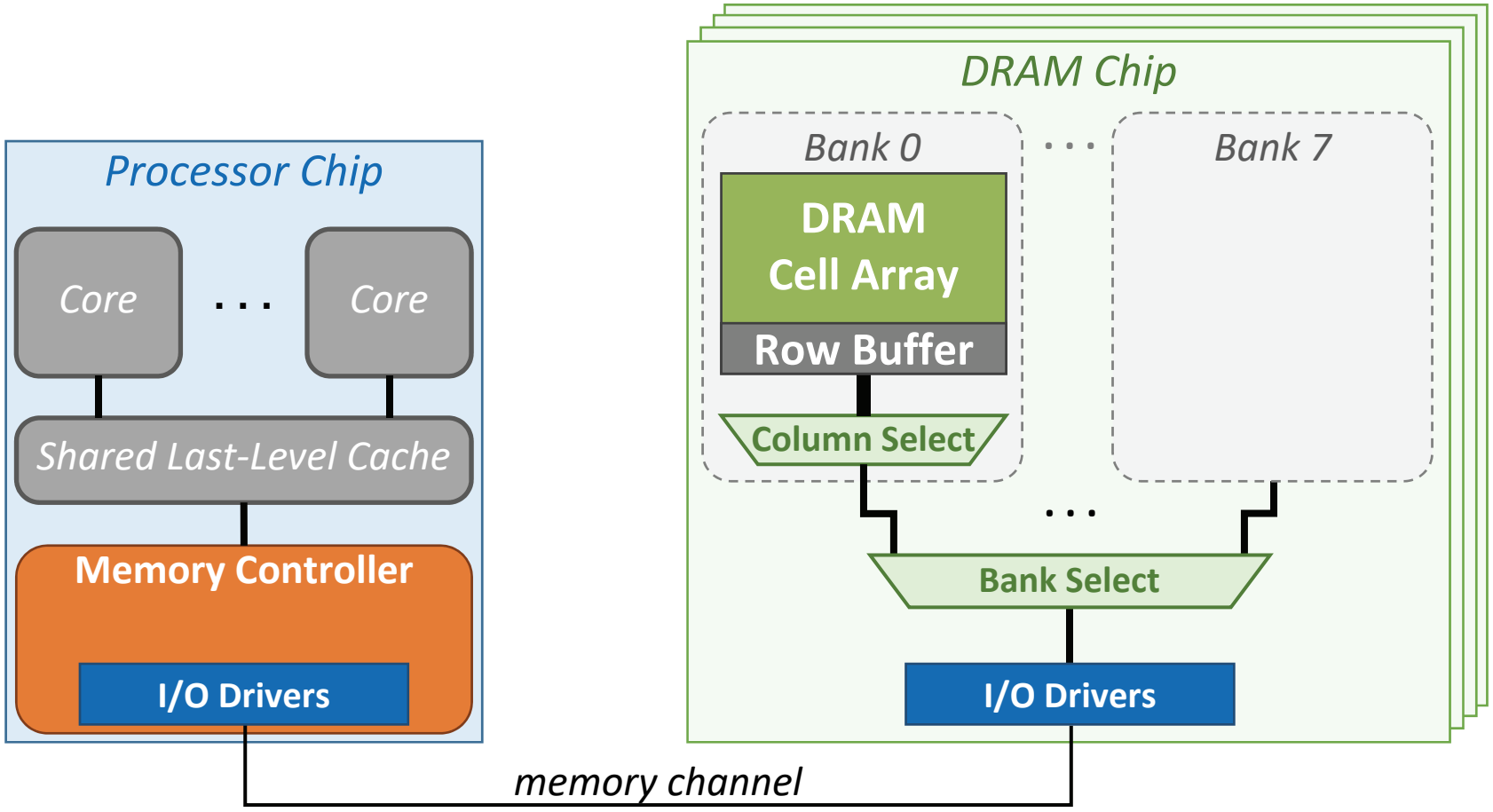
- Fundamental DRAM commands: activate, read, write, precharge

Simplified DRAM Organization and Operation



- Fundamental DRAM commands: activate, read, write, precharge
- One row of DRAM: 8 kB
- One cache line of data: 64 B

Simplified DRAM Organization and Operation



- Fundamental DRAM commands: activate, read, write, precharge
- One row of DRAM: 8 kB
- One cache line of data: 64 B

Background: DRAM Organization & Operation

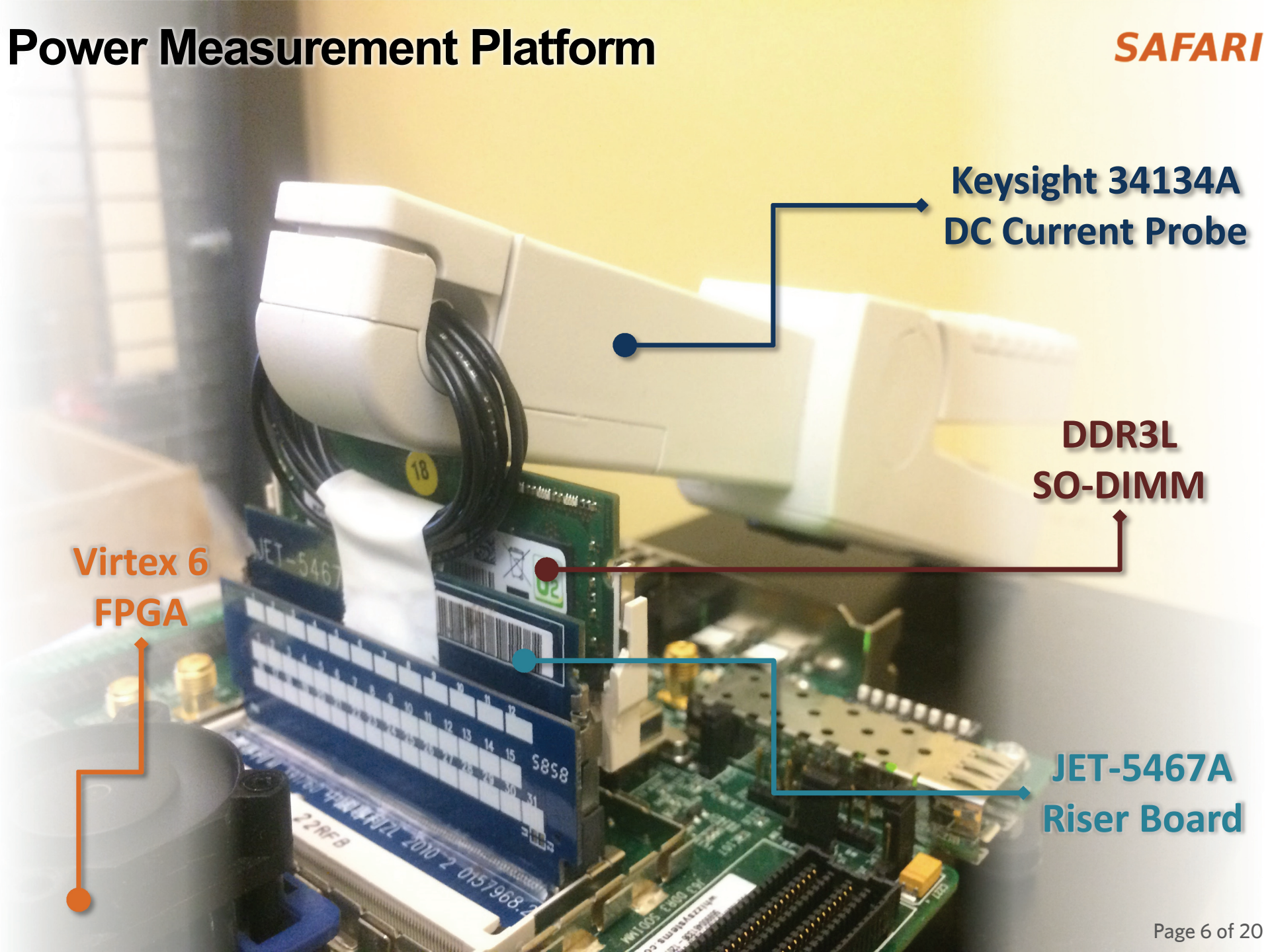
Characterization Methodology

New Findings on DRAM Power Consumption

VAMPIRE: A Variation-Aware DRAM Power Model

Conclusion

Power Measurement Platform



Methodology Details

- **SoftMC: an FPGA-based memory controller** [Hassan+ HPCA '17]
  - Modified to repeatedly loop commands
  - Open-source: https://github.com/CMU-SAFARI/SoftMC

- **Measure current consumed by a module during a SoftMC test**

- **Tested 50 DDR3L DRAM modules** (200 DRAM chips)
  - Supply voltage: 1.35 V
  - **Three major vendors: A, B, C**
  - Manufactured between 2014 and 2016

- **For each experimental test that we perform**
  - 10 runs of each test per module
  - At least 10 current samples per run

Outline

Background: DRAM Organization & Operation

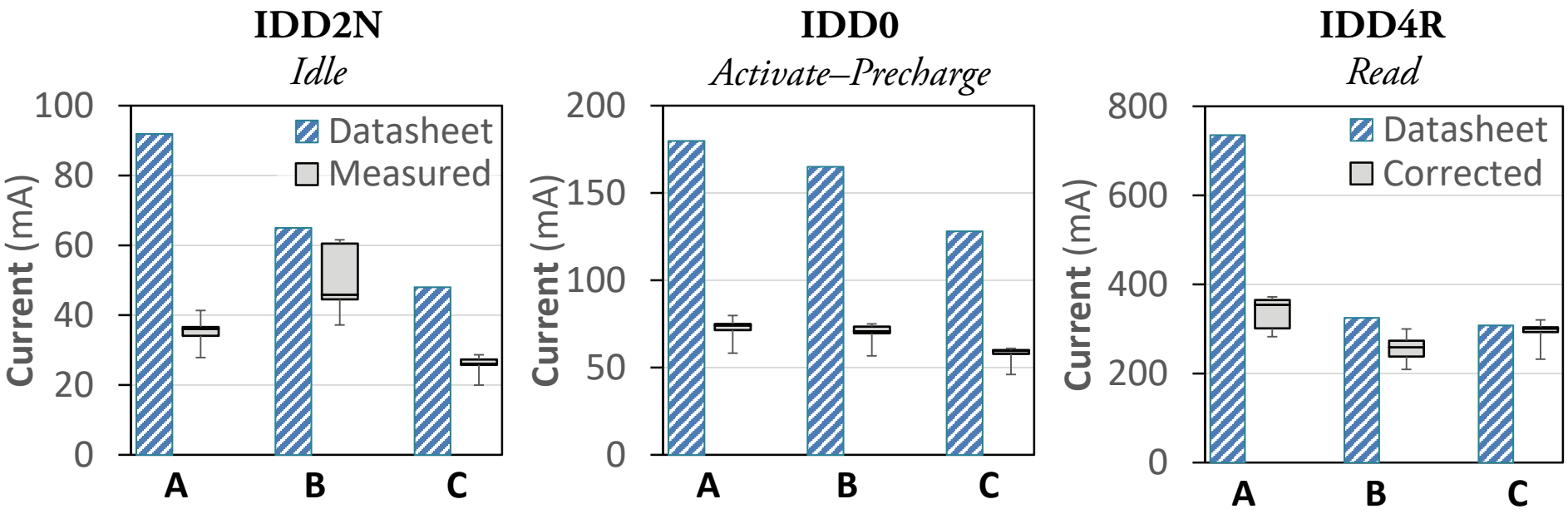
Characterization Methodology

New Findings on DRAM Power Consumption

VAMPIRE: A Variation-Aware DRAM Power Model

Conclusion

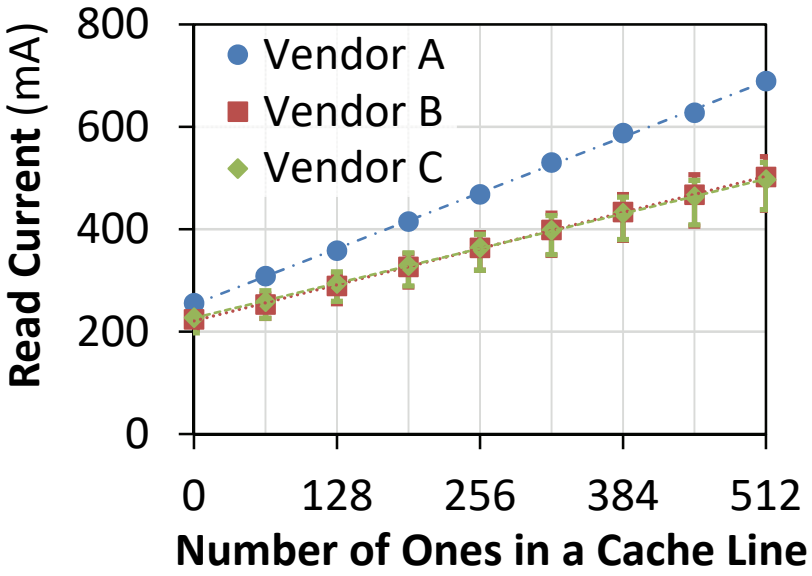
1. Real DRAM Power Varies Widely from IDD Values

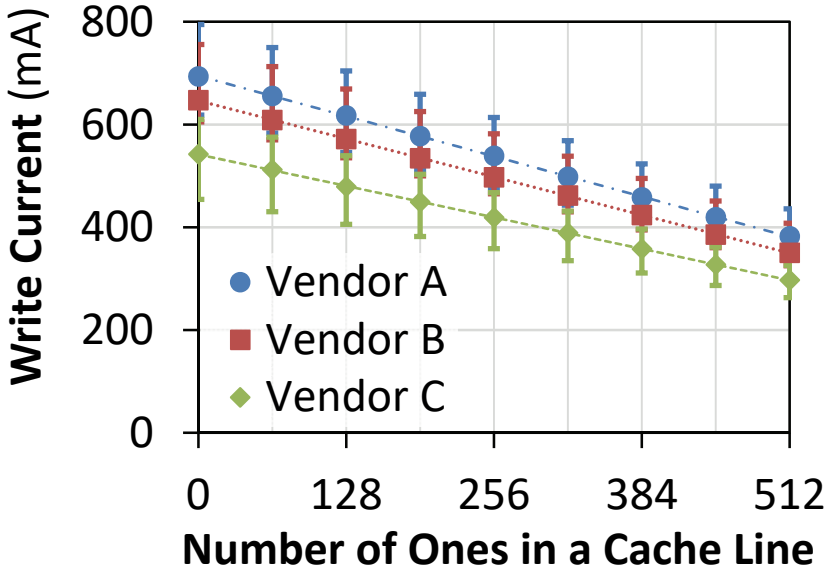


- Different vendors have very different margins (i.e., *guardbands*)
- Low variance among different modules from same vendor

Current consumed by real DRAM modules varies significantly for all IDD values that we measure

2. DRAM Power is Dependent on Data Values

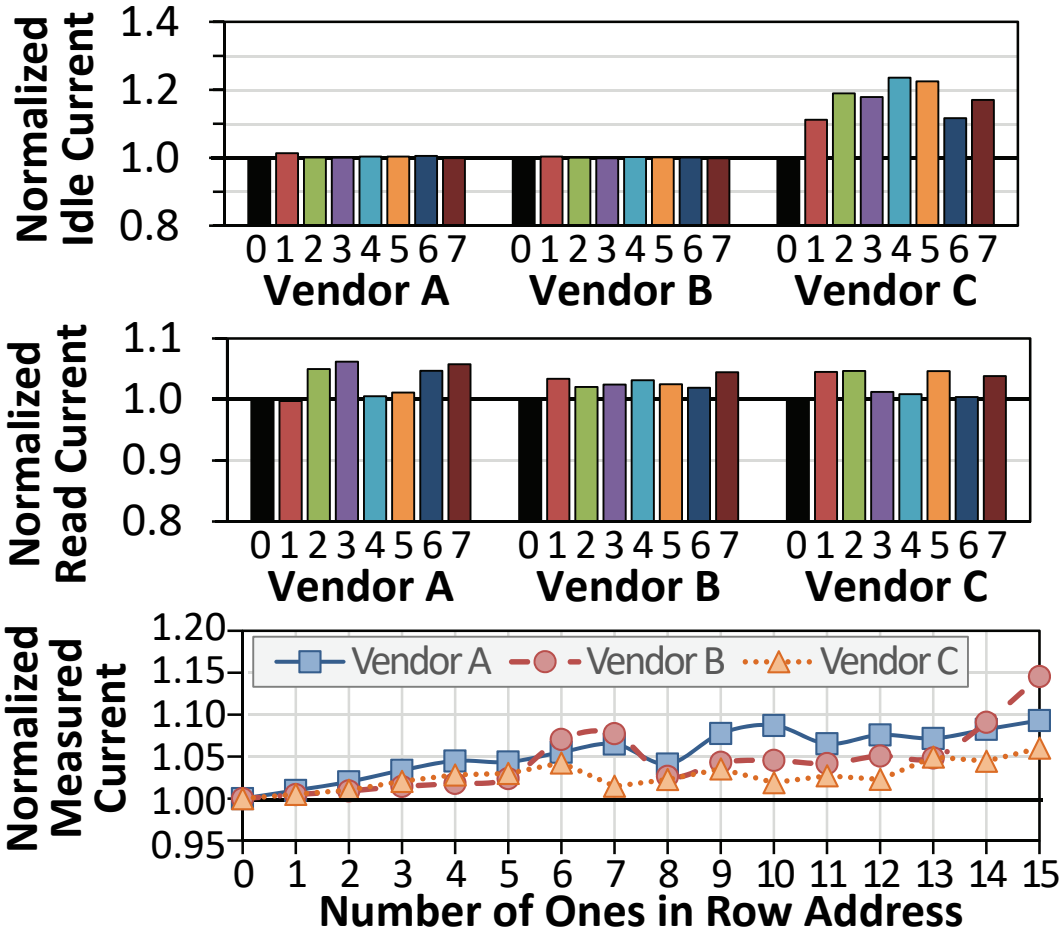




- Some variation due to infrastructure – can be subtracted
- Without infrastructure variation: up to 230 mA of change
- Toggle affects power consumption, but < 0.15 mA per bit

DRAM power consumption depends *strongly* on the data value, but not on bit toggling

3. Structural Variation Affects DRAM Power Usage

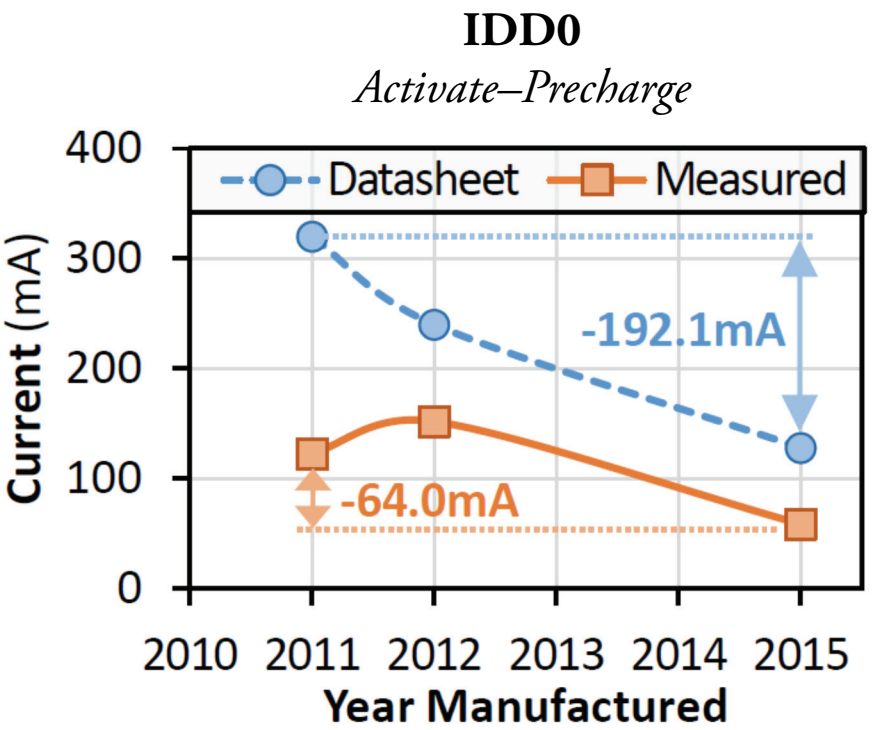


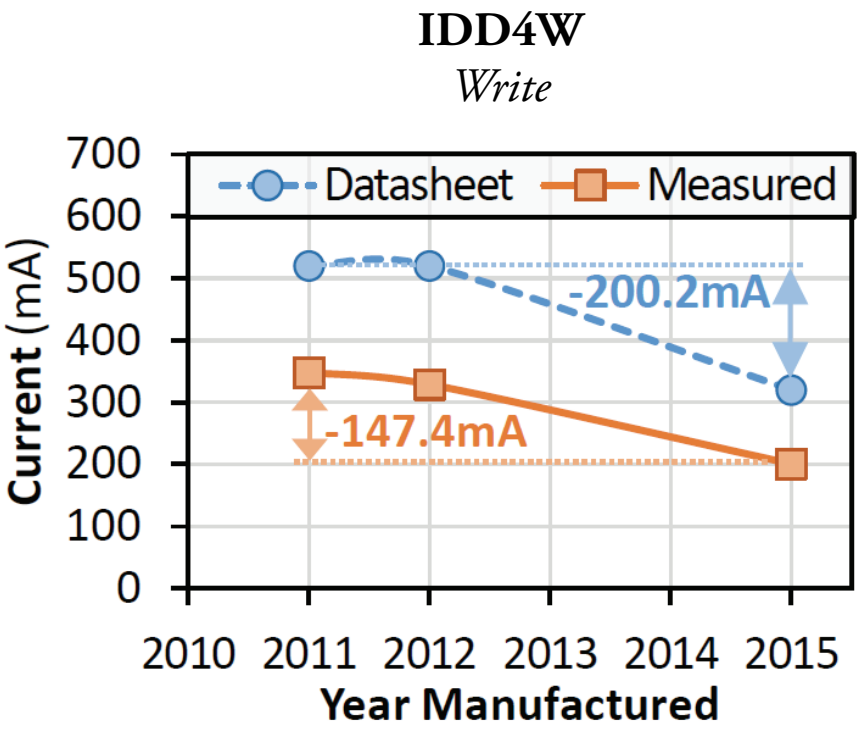
- Vendor C: variation in idle current across banks
- All vendors: variation in read current across banks
- All vendors: variation in activation based on row address

Significant structural variation:
DRAM power varies systematically by bank and row

4. Generational Savings Are Smaller Than Expected

SAFARI





- Similar trends for idle and read currents

Actual power savings of newer DRAM is *much lower* than the savings indicated in the datasheets

Summary of New Observations on DRAM Power

1. Real DRAM modules often consume less power than vendor-provided IDD values state

2. DRAM power consumption is dependent on the data value that is read/written

3. Across banks and rows, structural variation affects power consumption of DRAM

4. Newer DRAM modules save less power than indicated in datasheets by vendors

Detailed observations and analyses in the paper

Outline

Background: DRAM Organization & Operation

Characterization Methodology

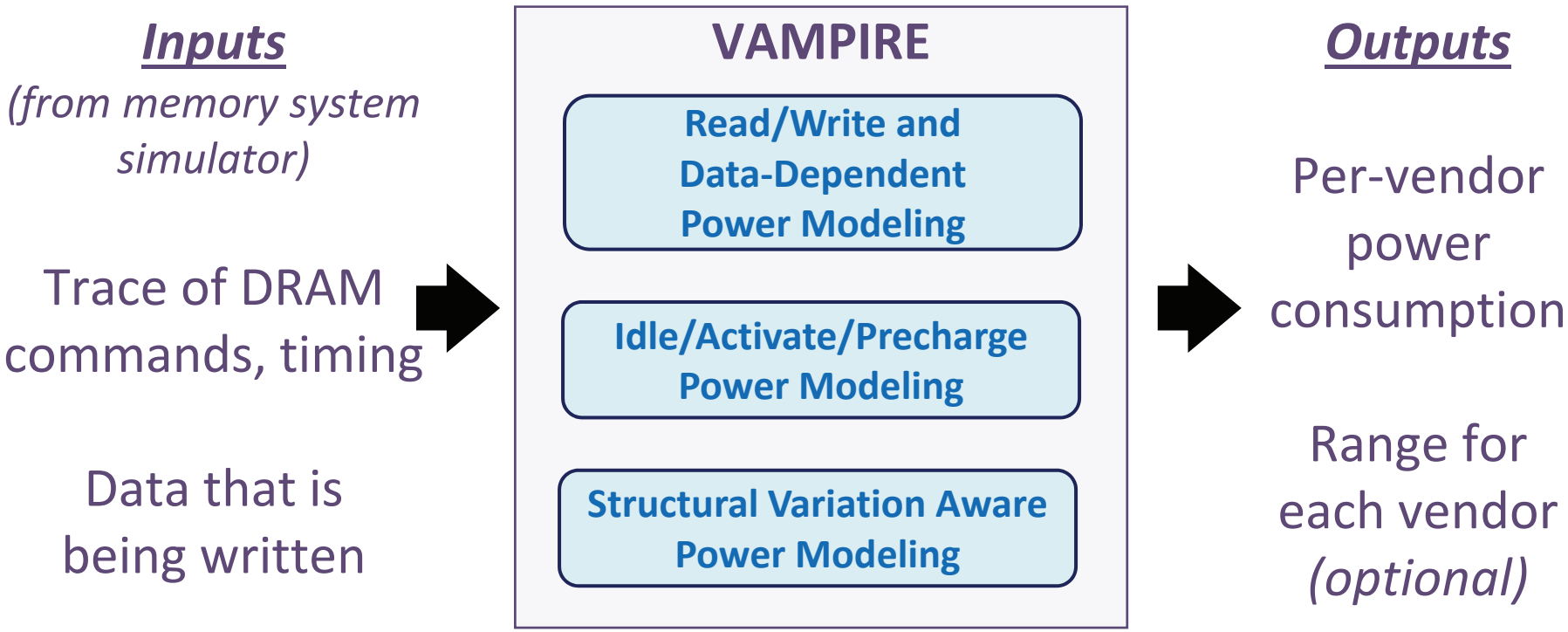
New Findings on DRAM Power Consumption

VAMPIRE: A Variation-Aware DRAM Power Model

Conclusion

A New Variation-Aware DRAM Power Model

- VAMPIRE: Variation-Aware model of Memory Power Informed by Real Experiments

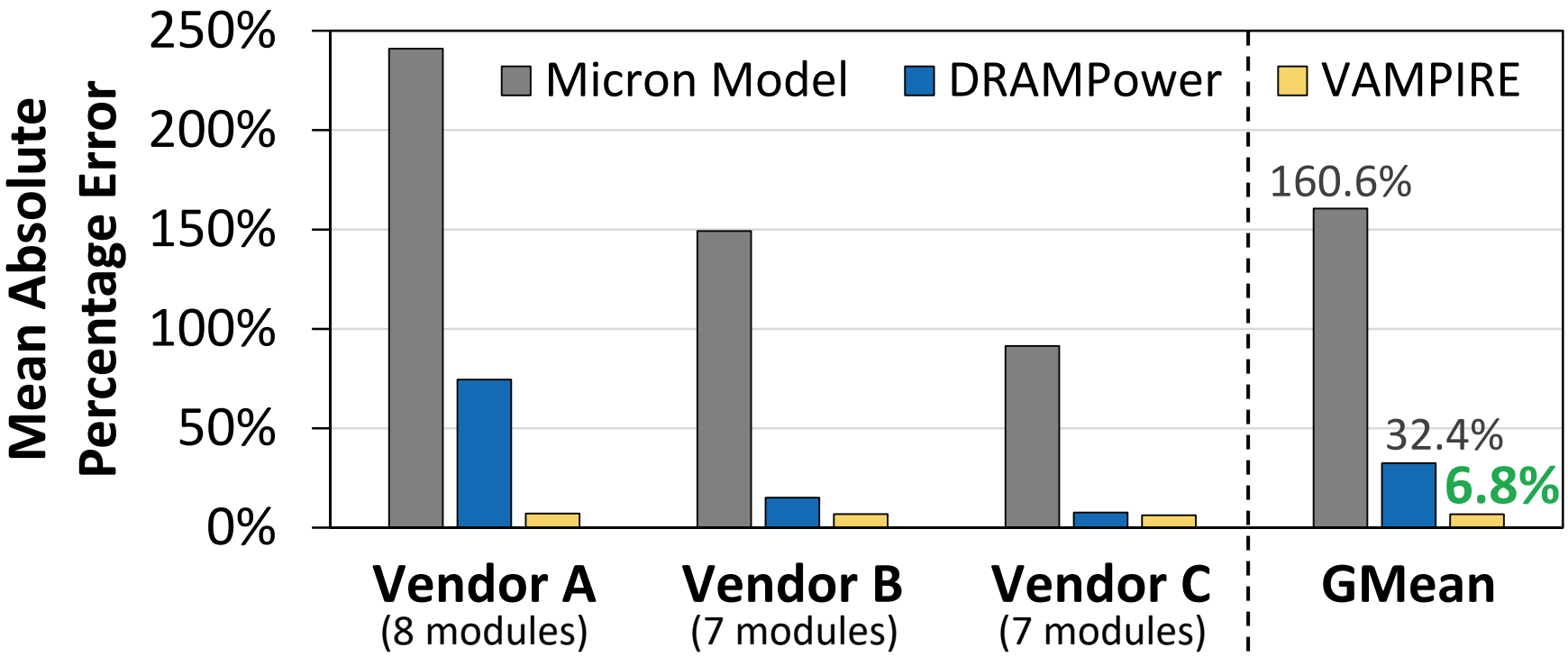


- VAMPIRE and raw characterization data will be open-source: https://github.com/CMU-SAFARI/VAMPIRE (August 2018)

VAMPIRE Has Lower Error Than Existing Models

SAFARI

- Validated using new power measurements: details in the paper

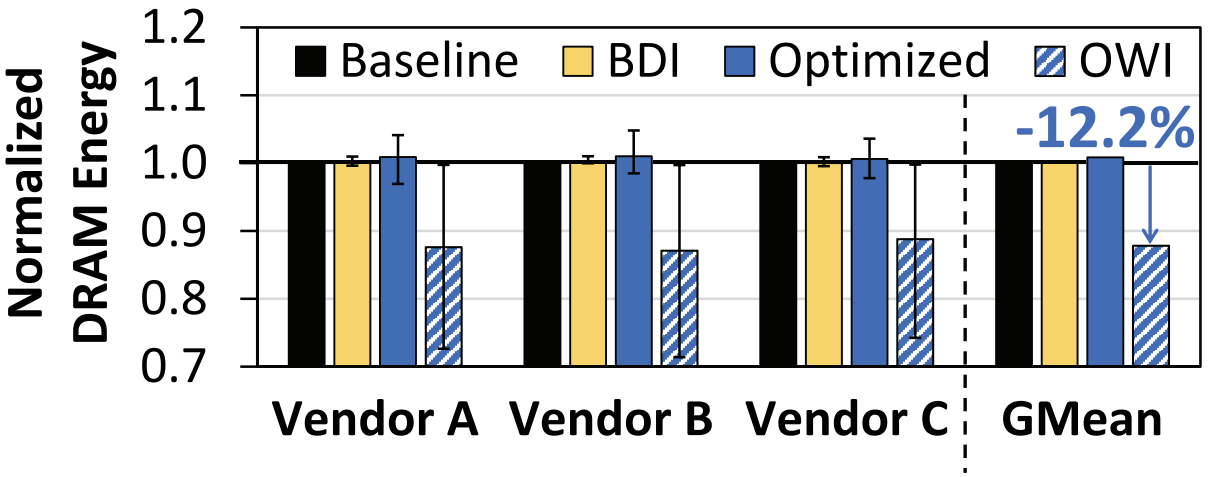


VAMPIRE has very low error for *all* vendors: 6.8%

Much more accurate than prior models

VAMPIRE Enables Several New Studies

- Taking advantage of structural variation to perform variation-aware physical page allocation to reduce power
- Smarter DRAM power-down scheduling
- Reducing DRAM energy with data-dependency-aware cache line encodings
  - 23 applications from the SPEC 2006 benchmark suite
  - Traces collected using Pin and Ramulator



- We expect there to be many other new studies in the future

Outline

Background: DRAM Organization & Operation

Characterization Methodology

New Findings on DRAM Power Consumption

VAMPIRE: A Variation-Aware DRAM Power Model

Conclusion

Conclusion

- DRAM consumes **up to half of total system power:** need to develop new low-power solutions
- State-of-the-art DRAM power models are based only on IDD values, and **have a high error**
- We make **four new observations** on DRAM power consumption using 50 real DRAM modules from three major vendors
  - Real DRAM modules often **consume less power** than IDD values state
  - Power consumption is **dependent on the data value** being read/written
  - Across banks and rows, **structural variation affects power** consumption
  - **Newer DRAM modules save less power** than indicated in datasheets
- **VAMPIRE:** a new DRAM power model built on our observations
  - Mean absolute percentage **error of only 6.8%**
  - Case study: dependency-aware data encoding **reduces DRAM power by 12%**

More information: https://github.com/CMU-SAFARI/VAMPIRE

Carnegie Mellon University ETH zürich NVIDIA TEXAS The University of Texas at Austin SAFARI

What Your DRAM Power Models Are Not Telling You:

Lessons from a Detailed Experimental Study

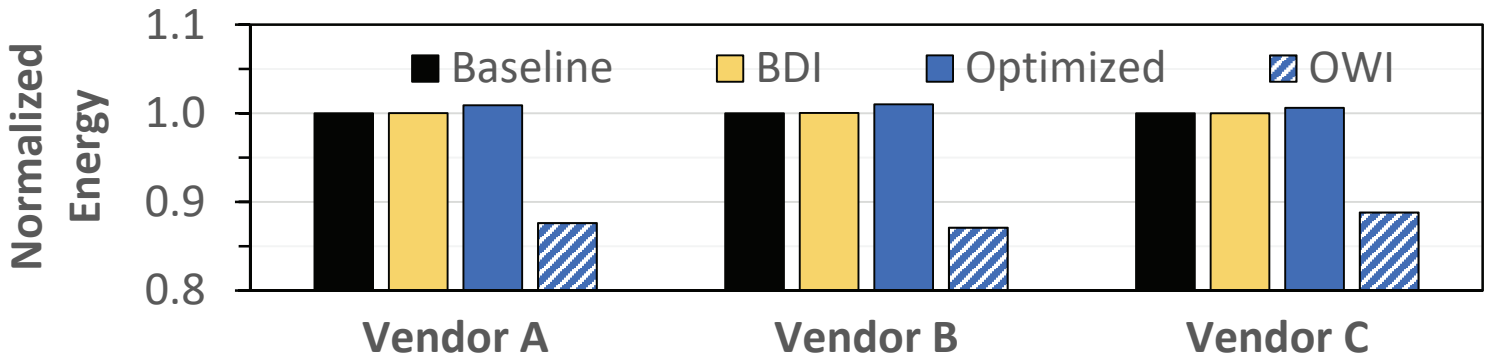
Saugata Ghose, A. Giray Yağlıkçı, Raghav Gupta, Donghyuk Lee, Kais Kudrolli, William X. Liu, Hasan Hassan, Kevin K. Chang, Niladrish Chatterjee, Aditya Agrawal, Mike O'Connor, Onur Mutlu

More information: https://github.com/CMU-SAFARI/VAMPIRE

Backup Slides

More Information in the Paper…

- Full characterization analysis
- Application-level comparison to existing power models
- Case study: dependency-aware data encoding

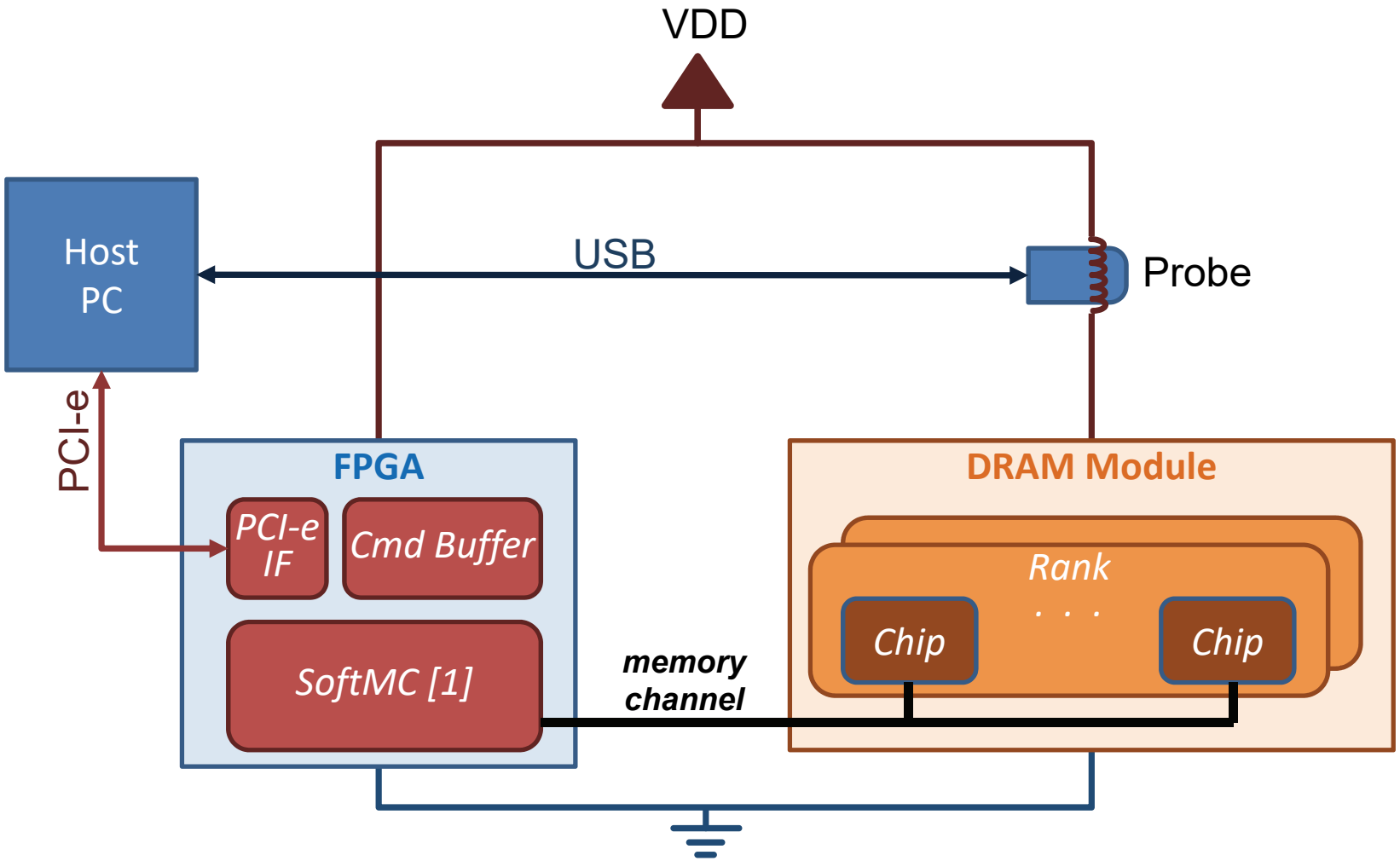


- Paper available at https://github.com/CMU-SAFARI/VAMPIRE

Today's Models Leave a Lot to Be Desired

- **Most models reliant on JEDEC-based IDD values**
  - Micron power calculator
  - DRAMPower
  - gem5/GPGPU-Sim
- **Some rely on circuit-level models**
  - Vogelsang model for memory scaling
  - CACTI
- **None are all that accurate**
  - One value for each DRAM
  - Does not capture any inherent variation (e.g., data, structure)

How Do We Measure Current?



[1] Hassan et al. "SoftMC: A Flexible and Practical Open-Source Infrastructure for Enabling Experimental DRAM Studies," HPCA, 2017.

Foundation of Current Power Models

- Just how bad are current models?

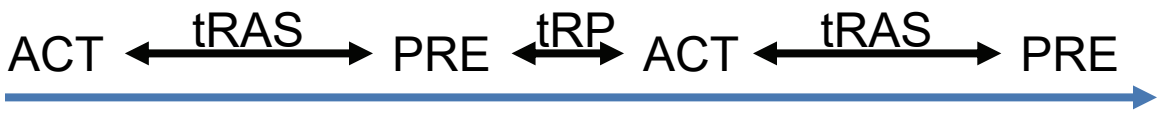
- JEDEC defines a set of IDD values

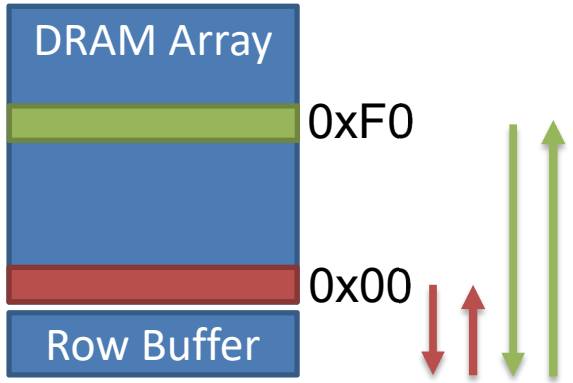
IDD0	Activation and Precharge	
IDD1	Activation – 1 Column Read – Precharge	
IDD2N	Precharge Standby (all banks are precharged/closed)	✓ clk enabled
IDD3N	Active Standby (all banks are active/opened)	✓ clk enabled
IDD2P	Precharge Power-Down (all banks are precharged/closed)	✗ clk disabled
IDD3P	Active Power-Down (all banks are active/opened)	✗ clk disabled
IDD4R/W	Burst mode Read/Write	
IDD5B	Burst mode Refresh	
IDD7	Activate – Column Read w/ Auto Precharge	

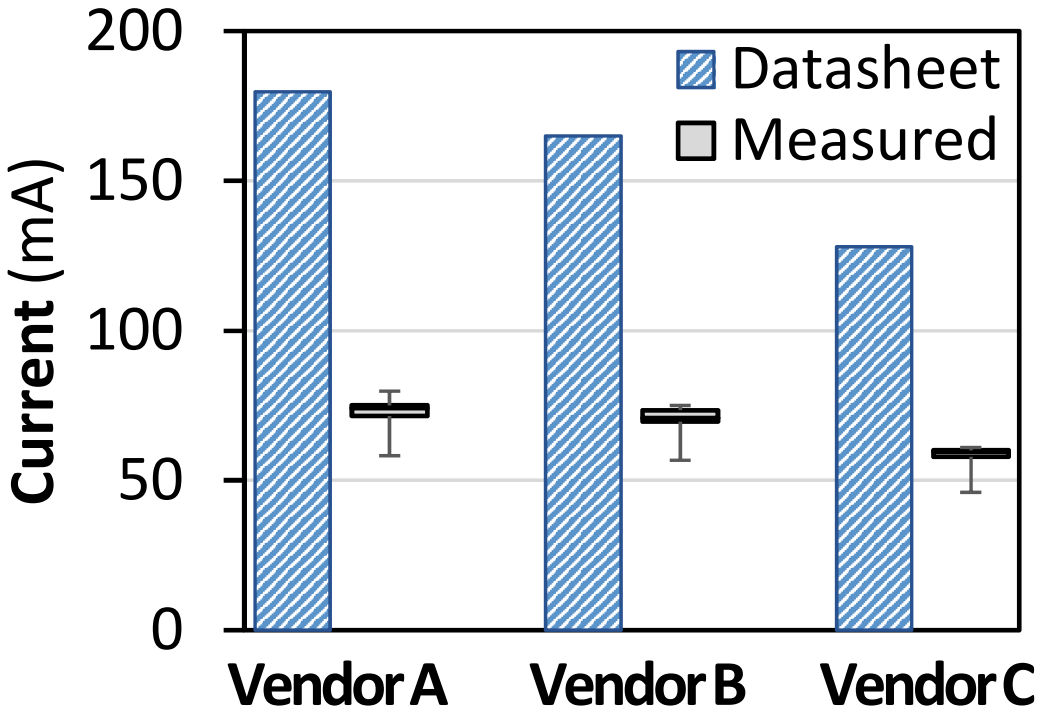
What's So Bad About That?

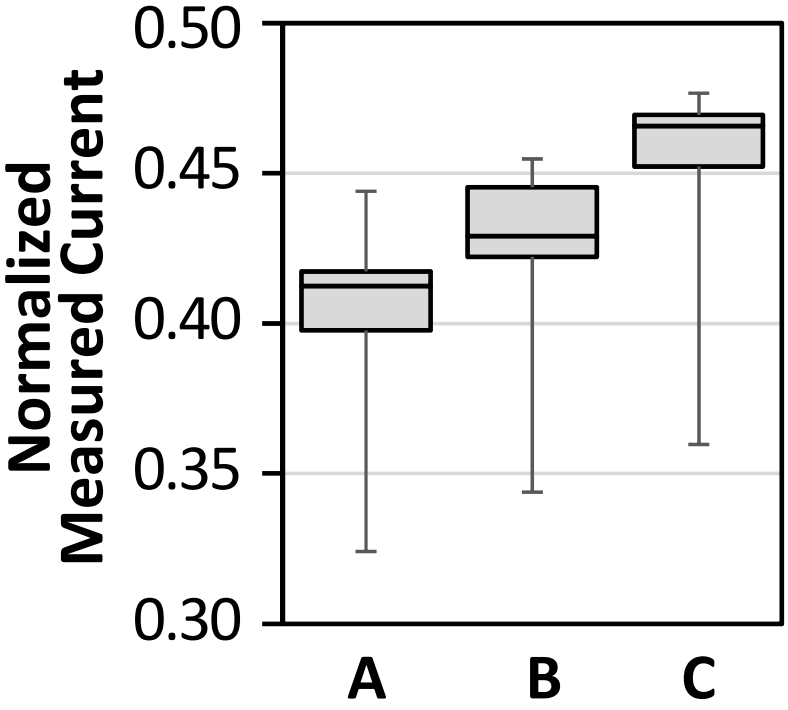
- JEDEC defined IDD measurement loops cover:
  - Average power consumption of all banks
    - » missing variation across banks
  - Average power consumption of only two rows: 00 and F0
    - » missing variation across rows in a subarray
    - » missing variation across subarrays
  - Average power consumption of only two data patterns: 00 and 33
    - » missing effect of number of ones/zeros in data
    - » missing effect of toggling bits

IDD0: Activation and Precharge Energy



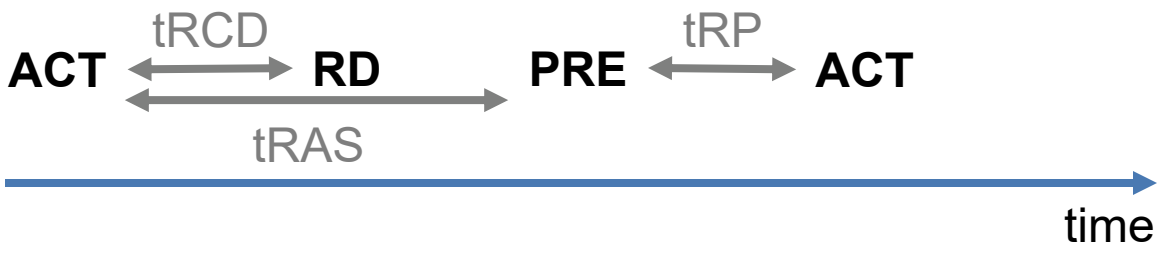


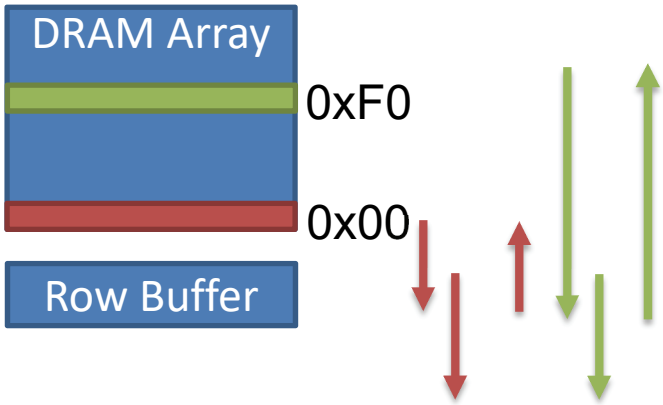


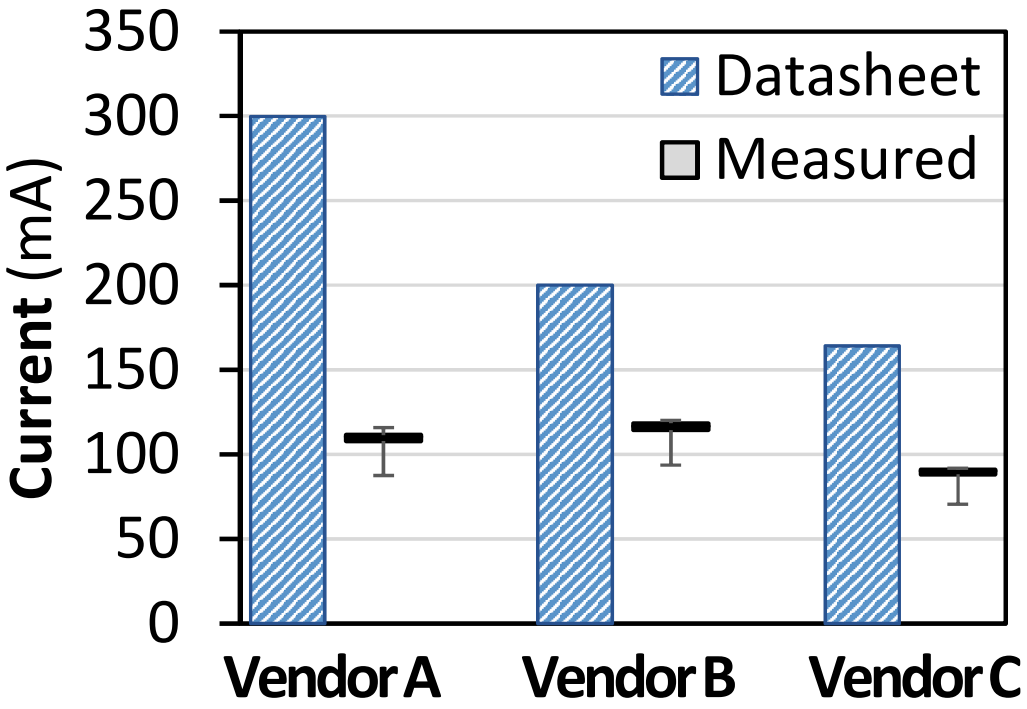


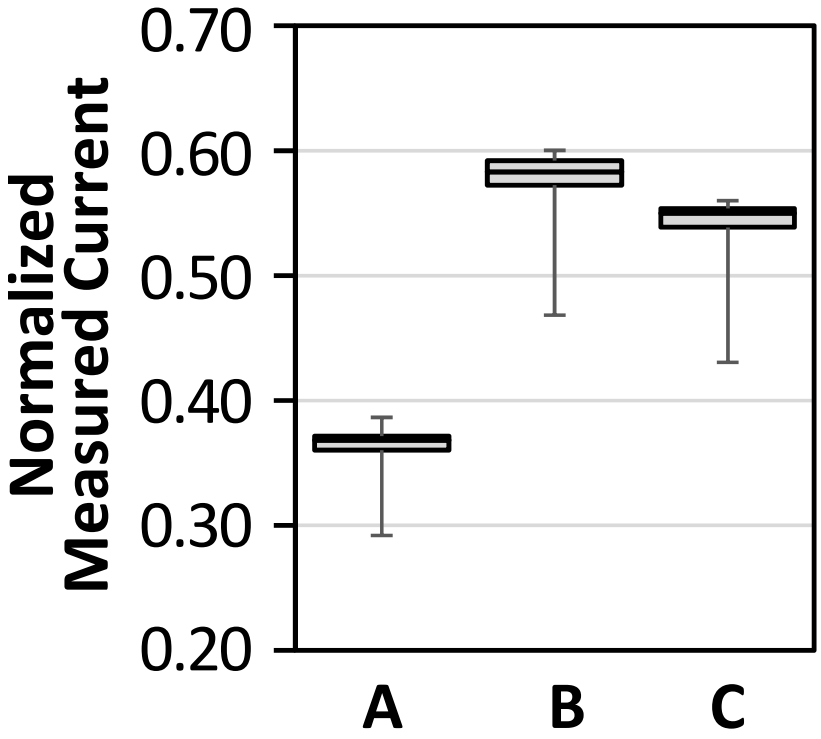
IDD1: Activation, Read, and Precharge Energy

SAFARI

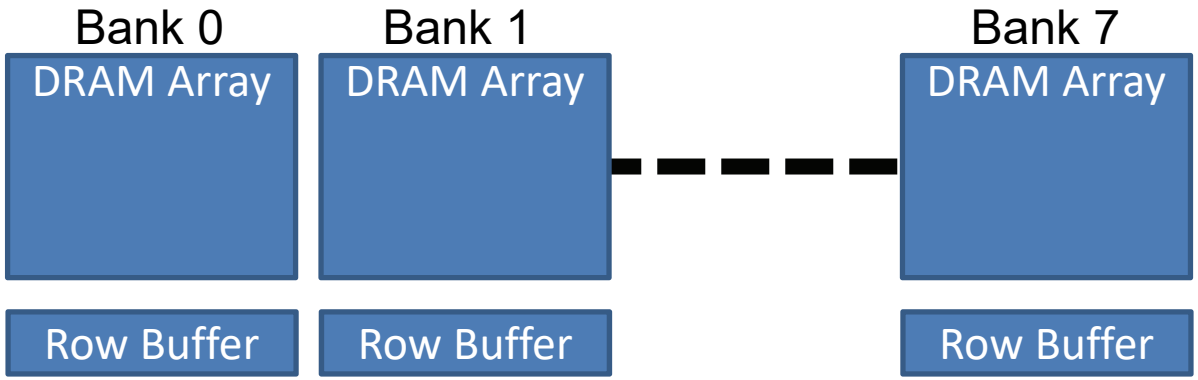






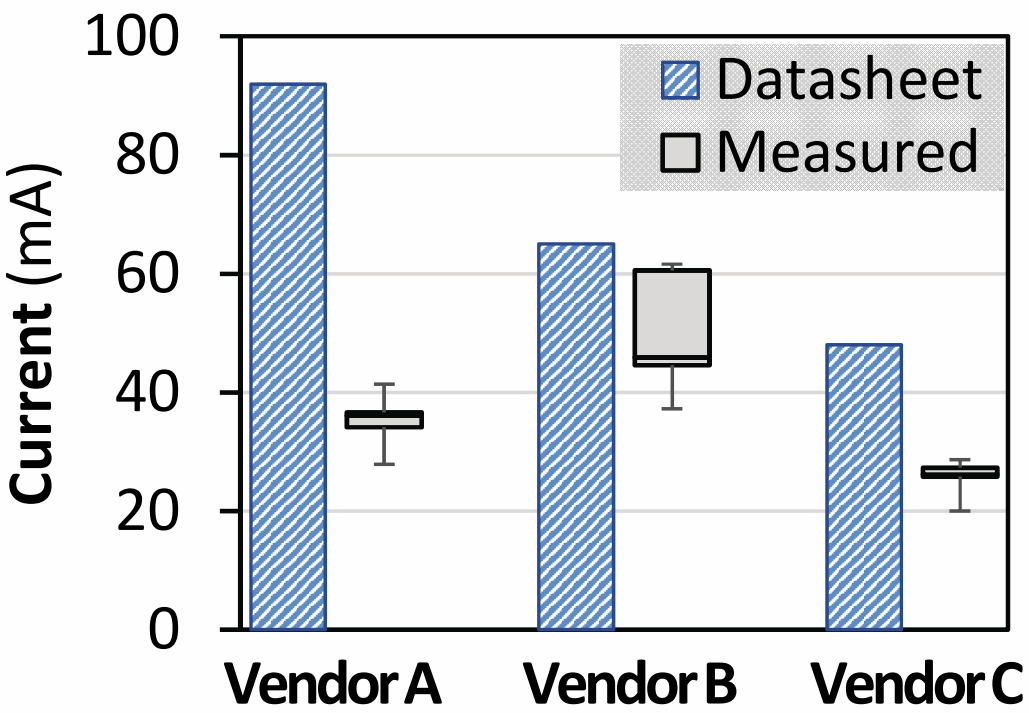


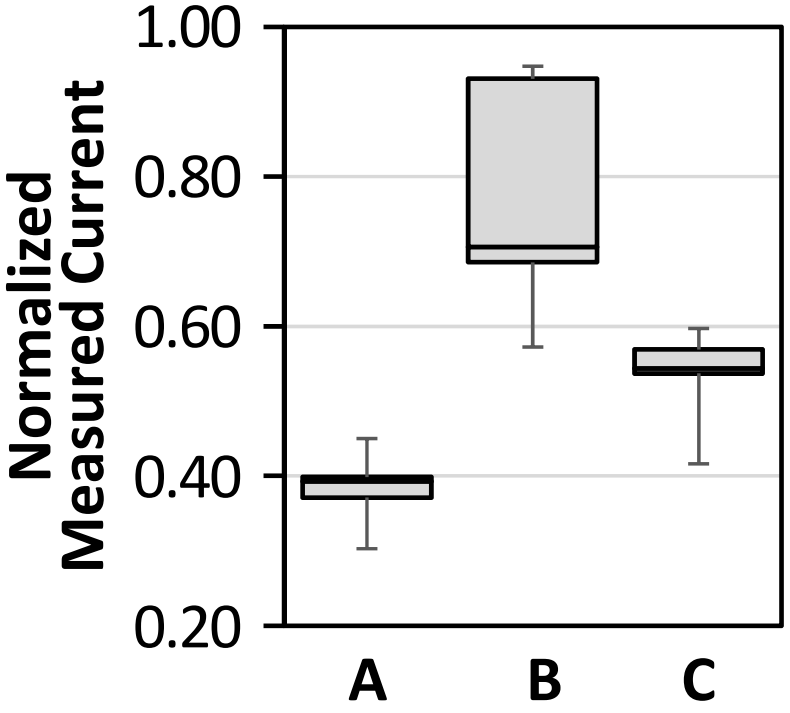
IDD2N: Precharged Standby



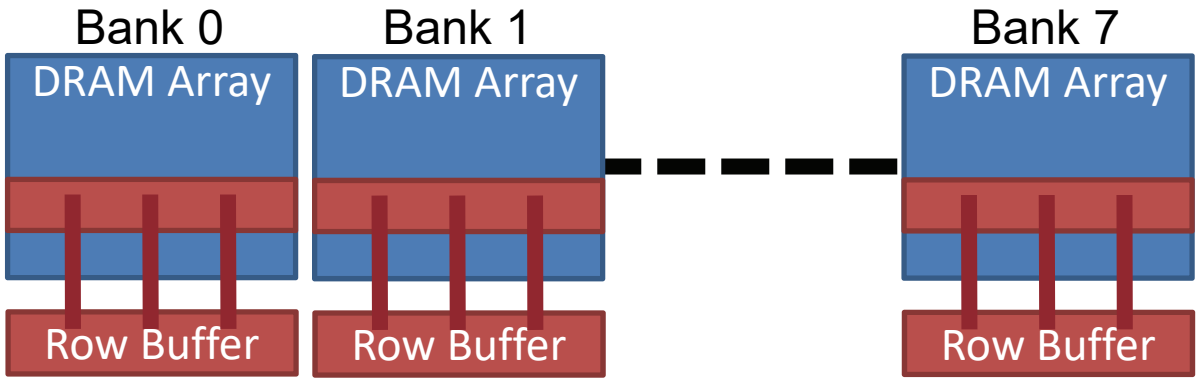
(1) Precharge All Banks
(Close Row Buffers)

(2) Wait



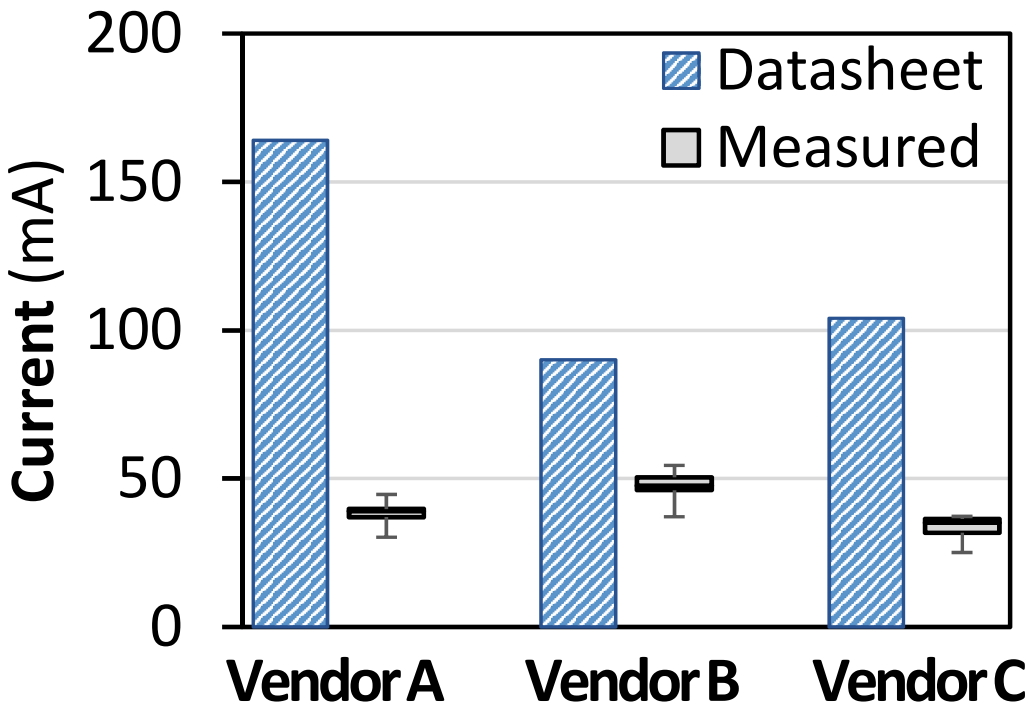


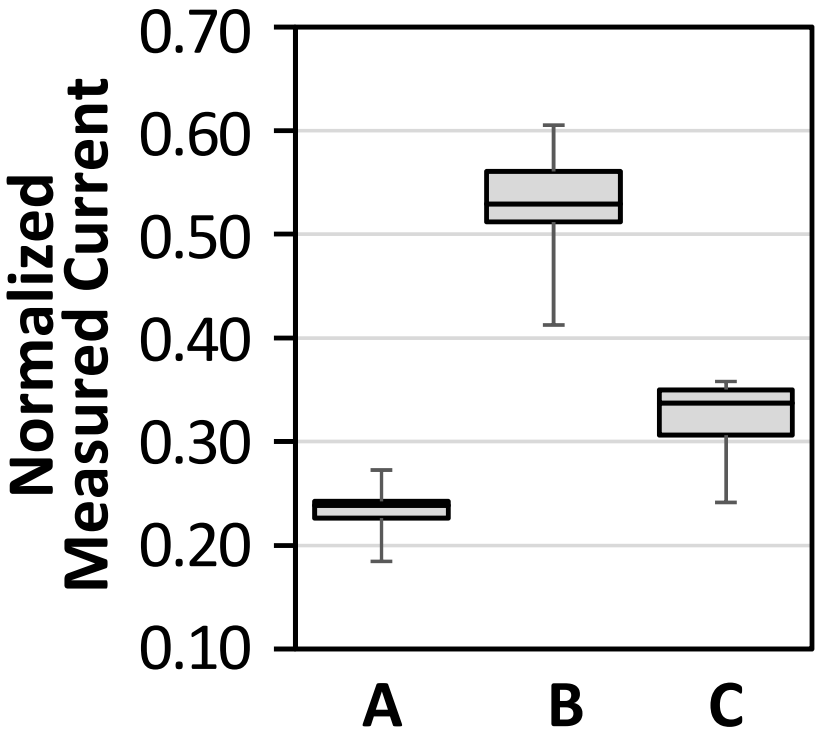
IDD3N: Active Standby



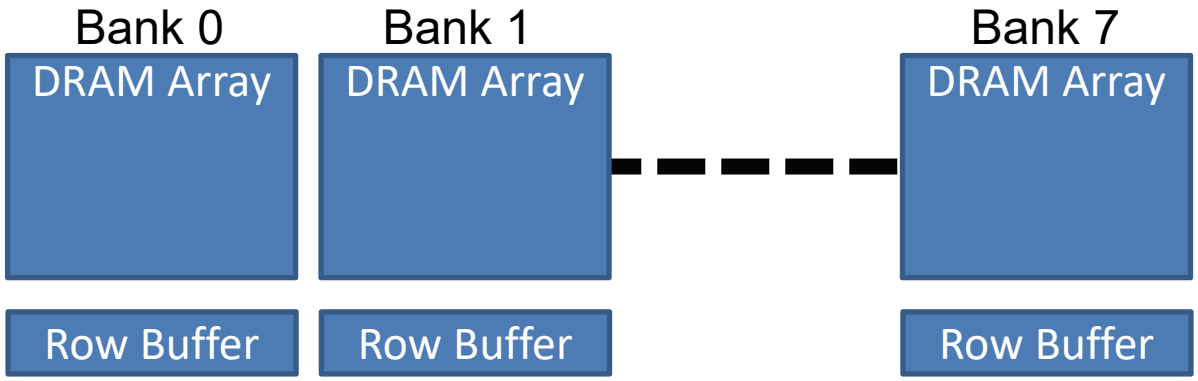
(1) Activate All Banks
(Open Row Buffers)

(2) Wait





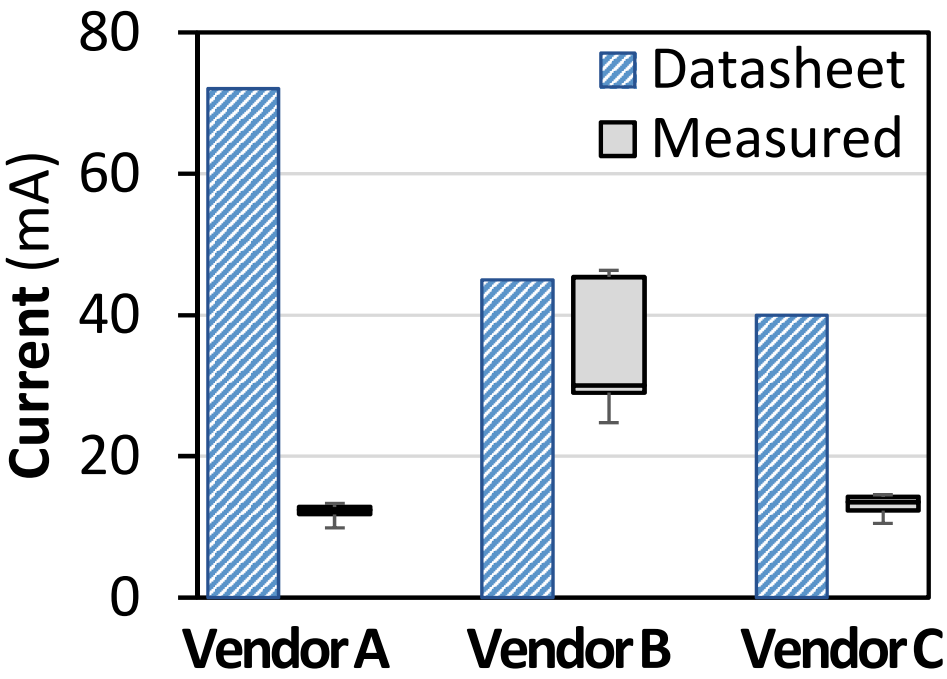
IDD2P: Precharged Power Down

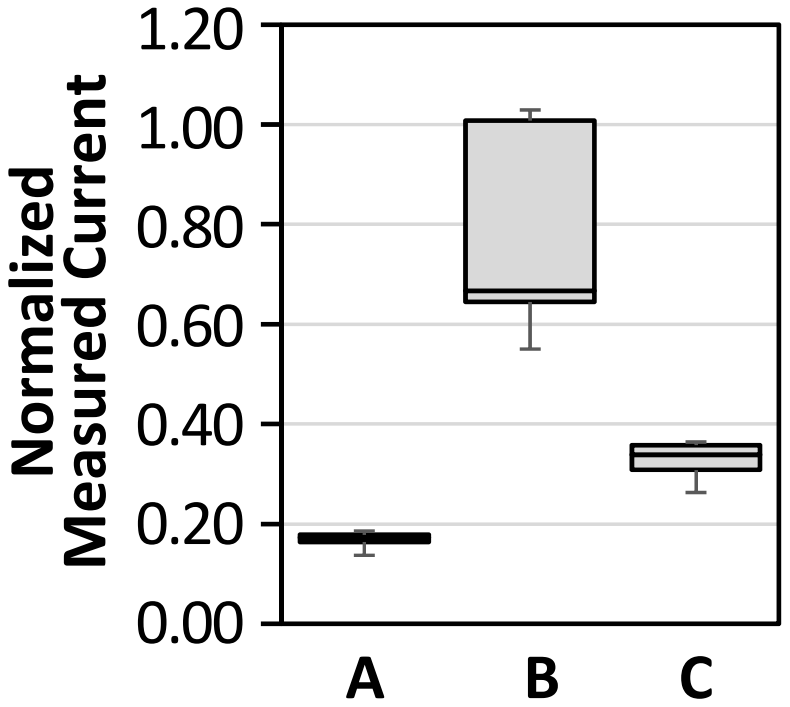


(1) Precharge All Banks
(Close Row Buffers)

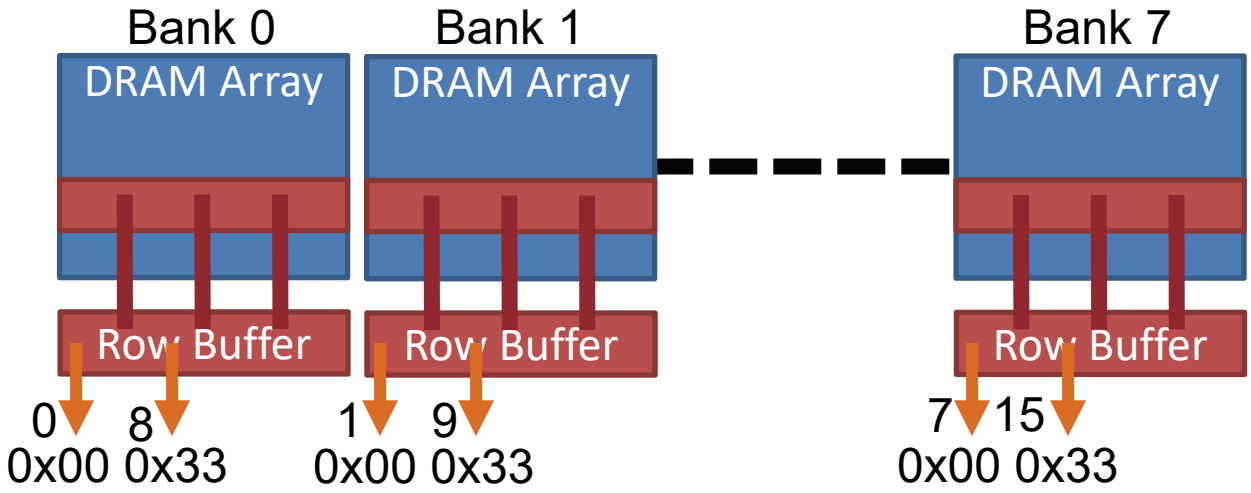
(2) Wait

CLK is Disabled





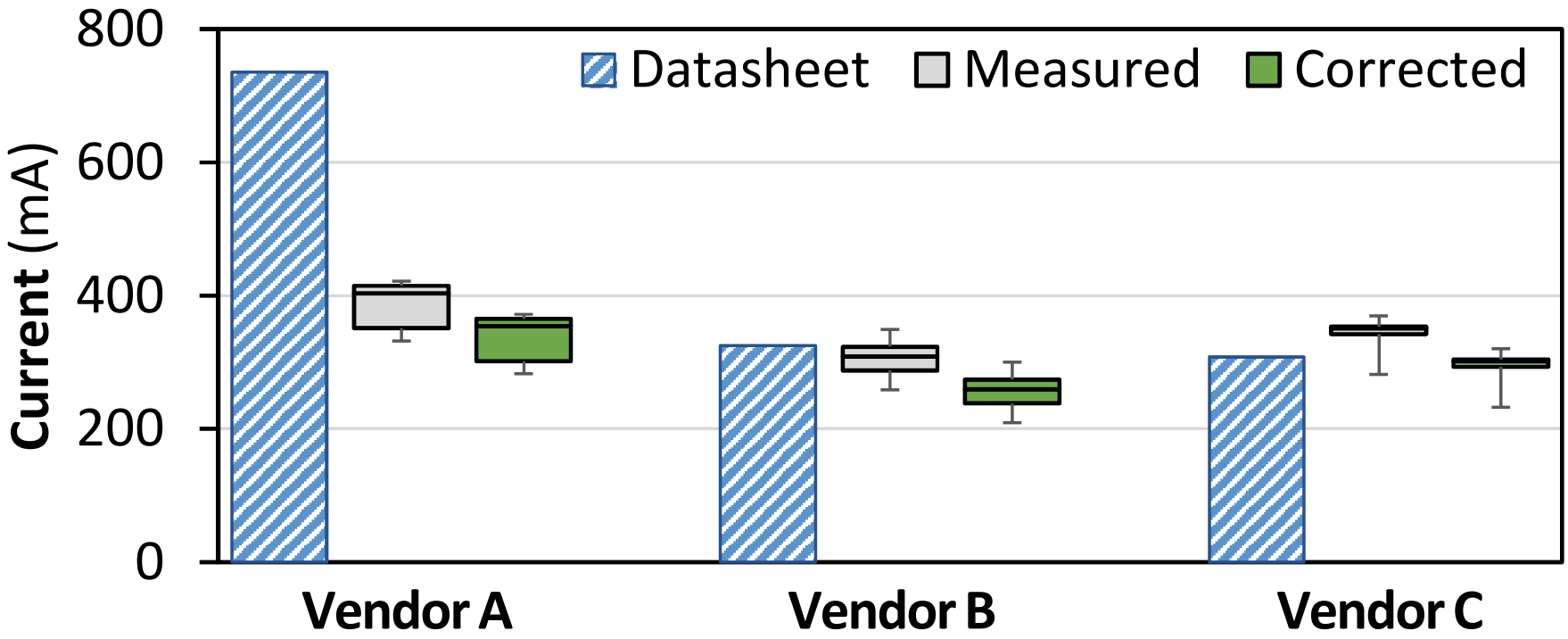
IDD4R: Burst Read Current



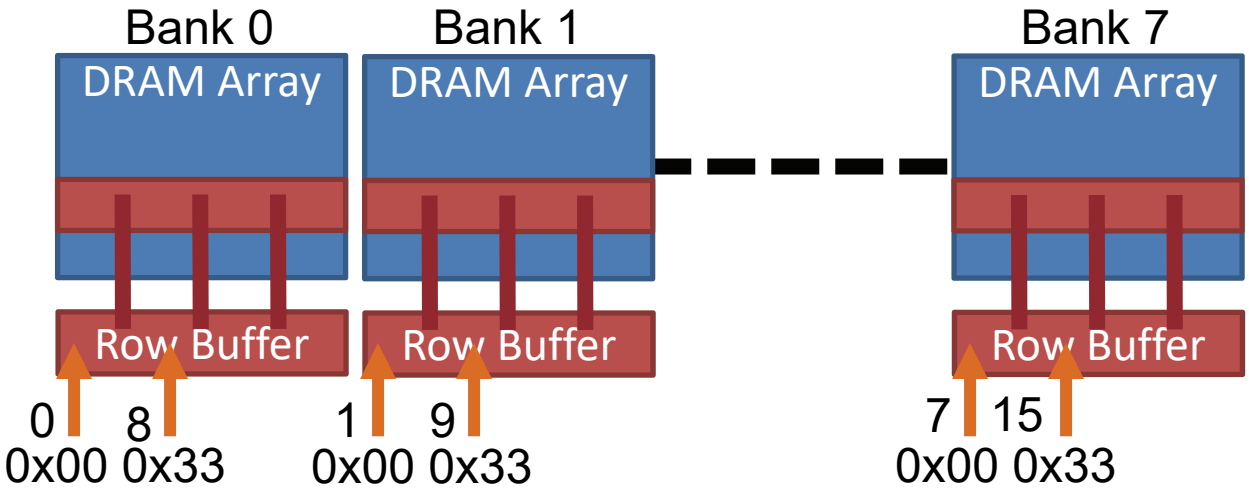
(1) Activate All Banks (Open Row Buffers)

(2) Read one column at a time

(3) Interleave across banks after each read



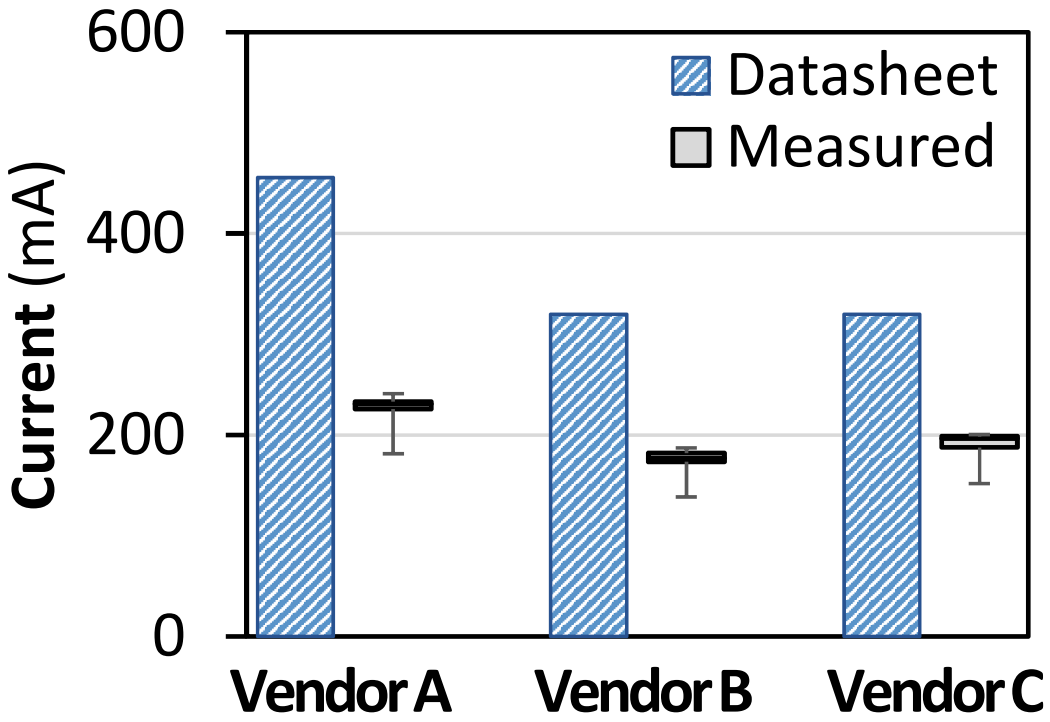
IDD4W: Burst Write Current

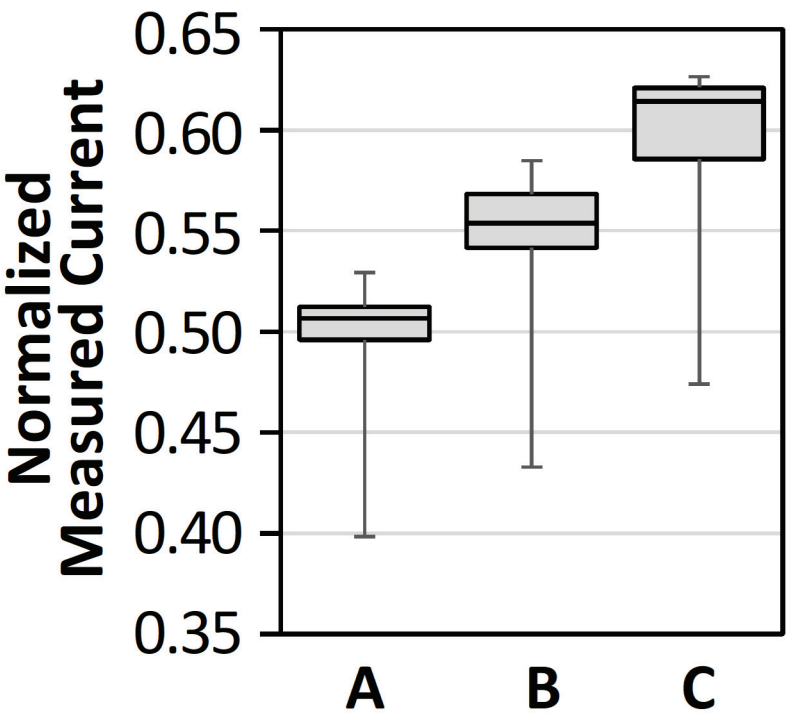


(1) Activate All Banks (Open Row Buffers)

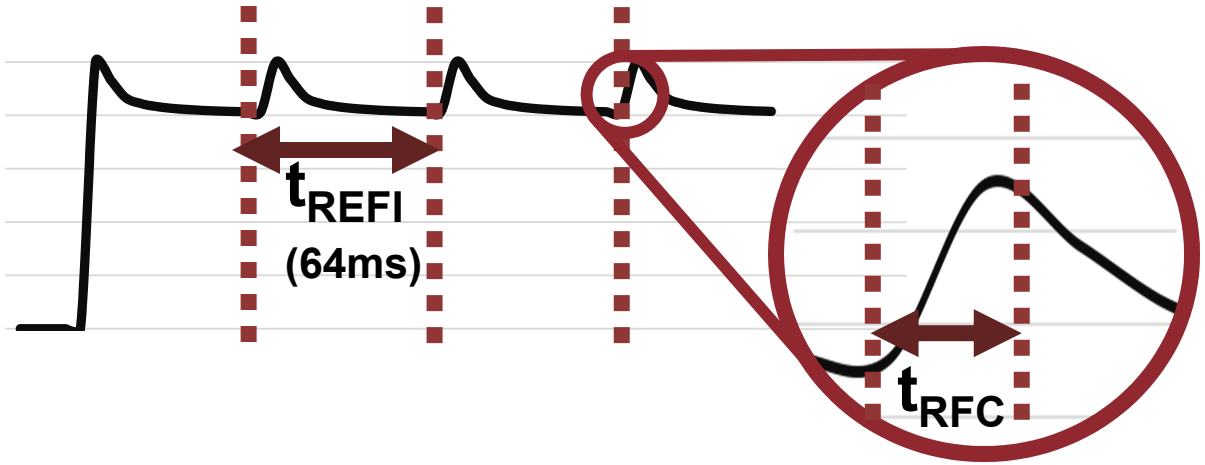
(2) Write one column at a time

(3) Interleave across banks after each read





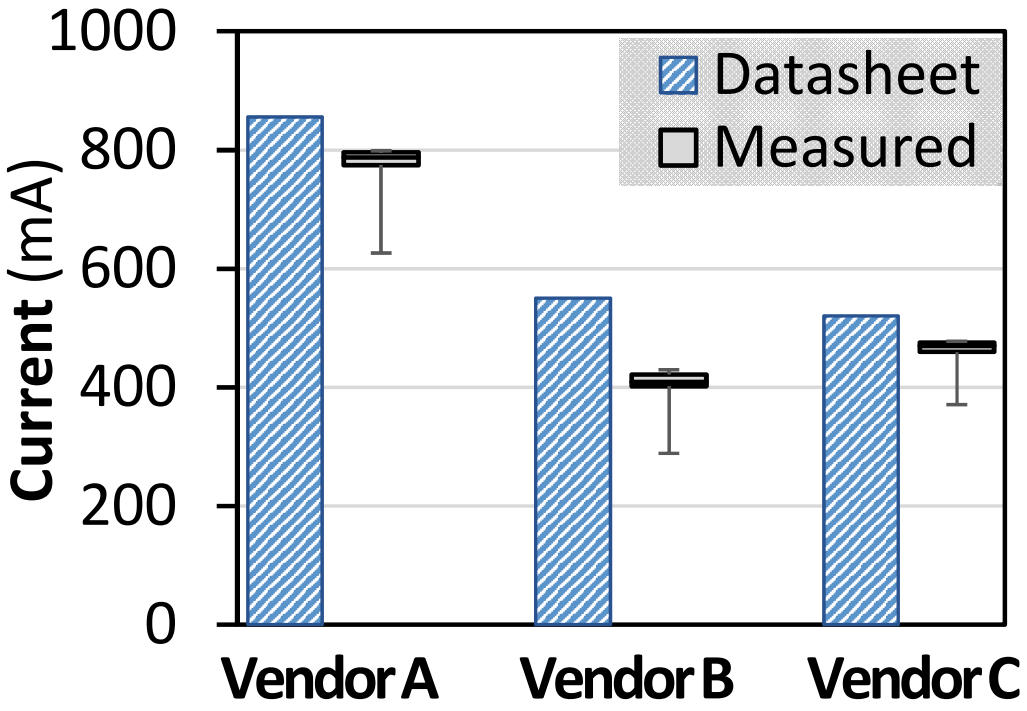
IDD5B: Refresh in Burst Mode

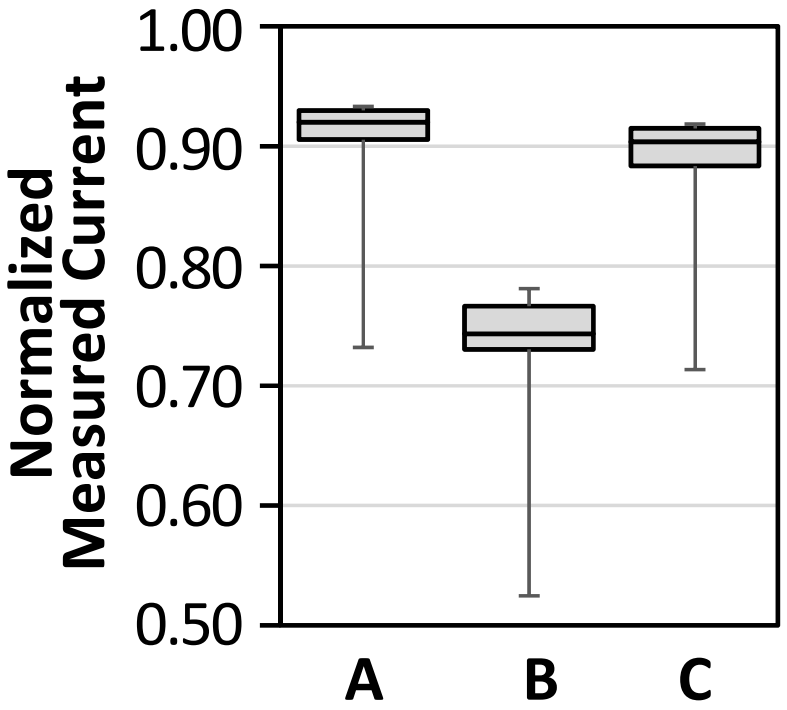


Burst Mode:

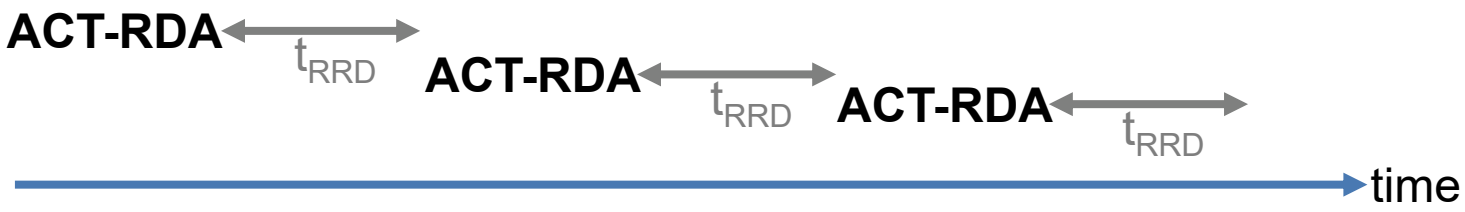
REF t_{RFC} REF t_{RFC} REF

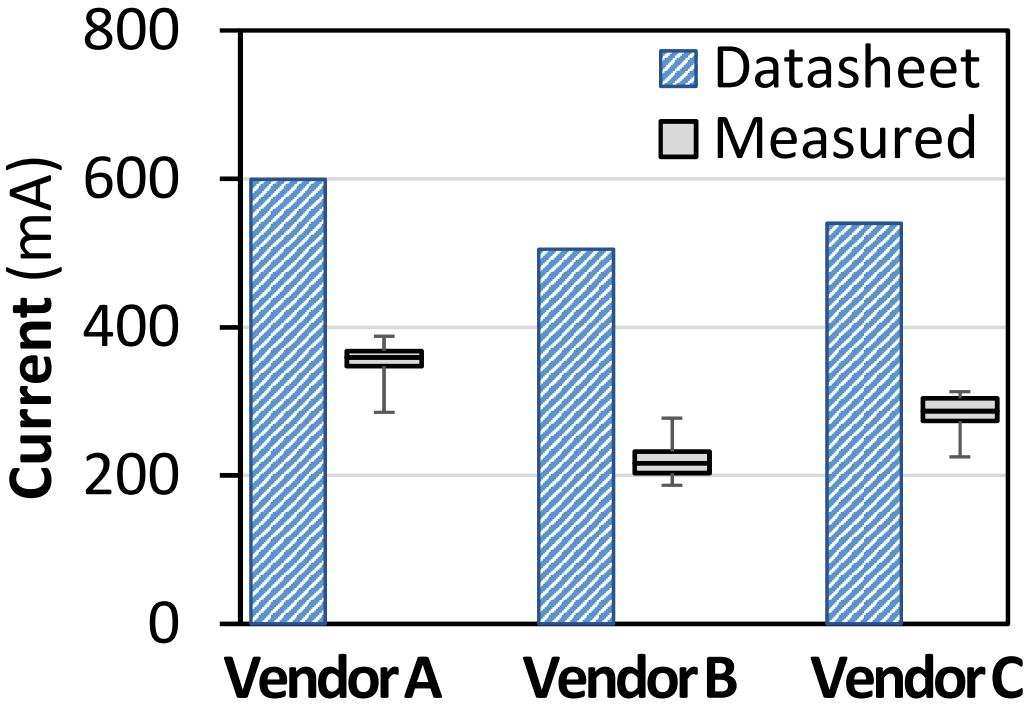
time

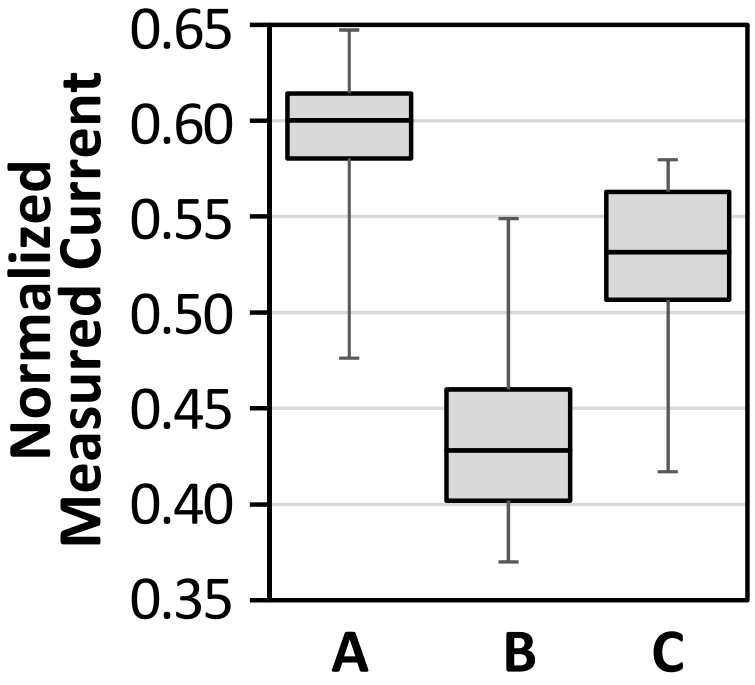




IDD7: Read, Auto-Precharge

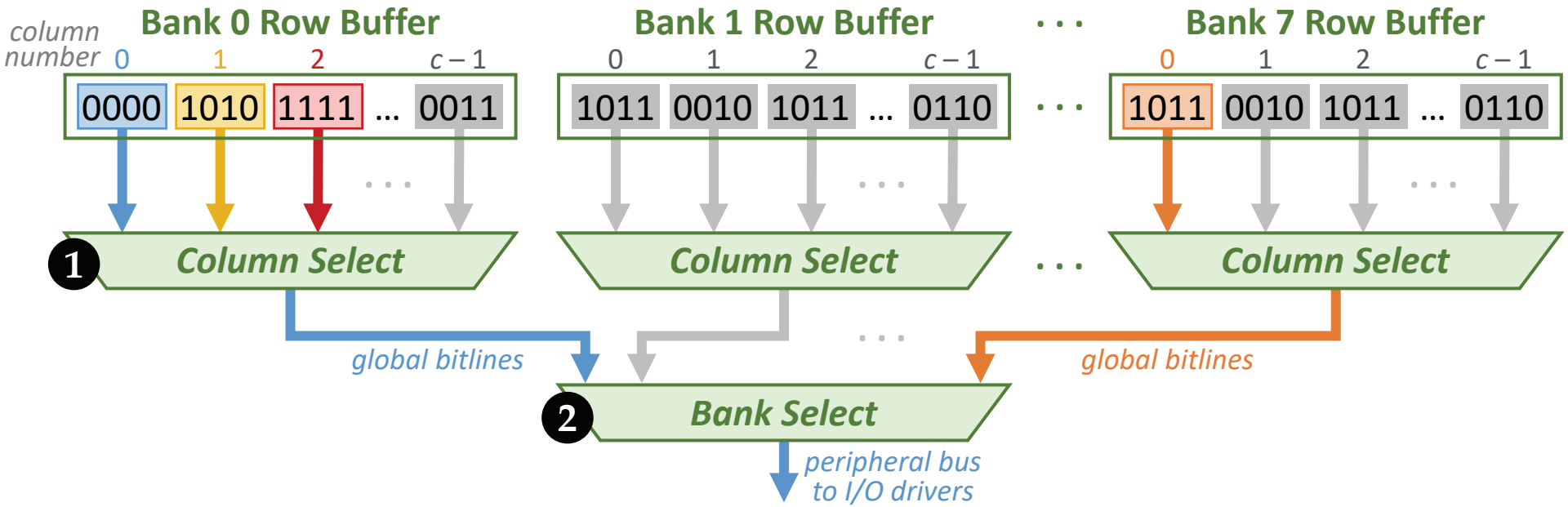


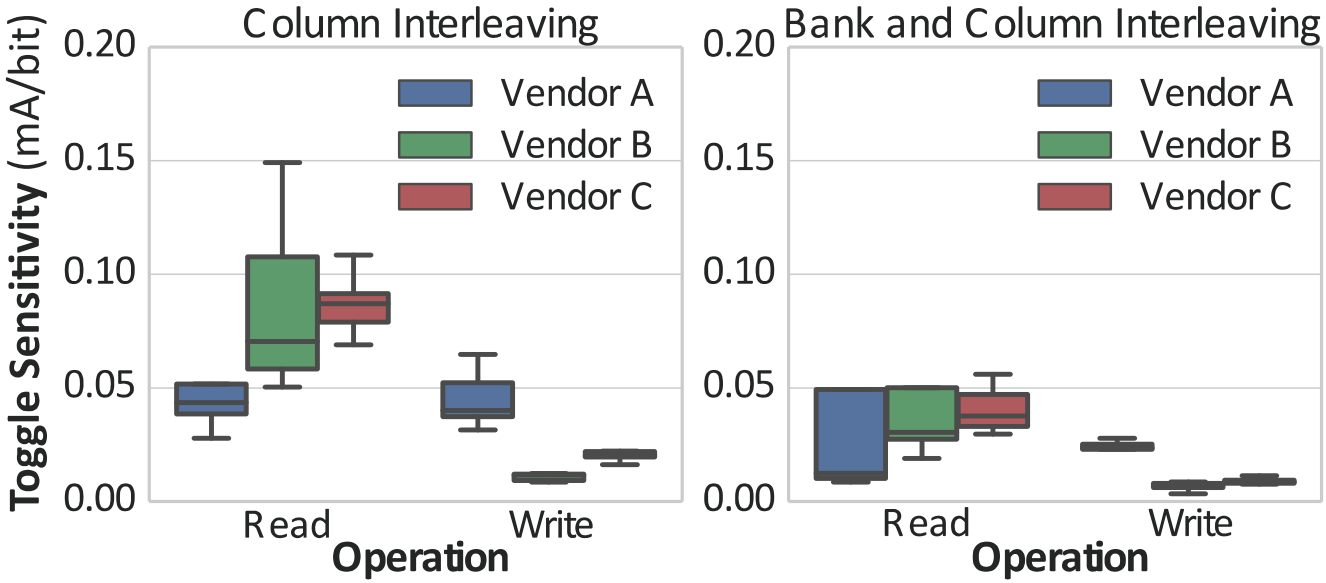




Impact of Bit Toggling on DRAM Power

SAFARI





Data Dependency Model

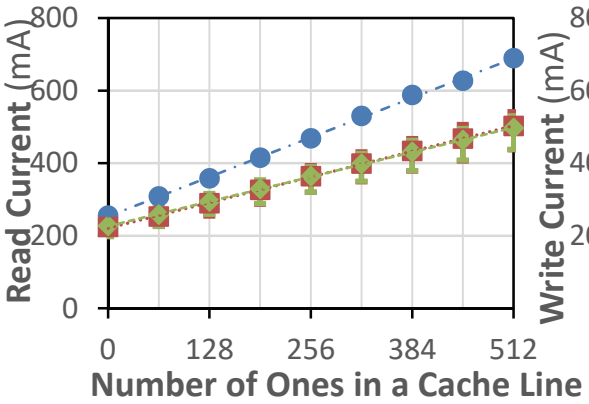
SAFARI

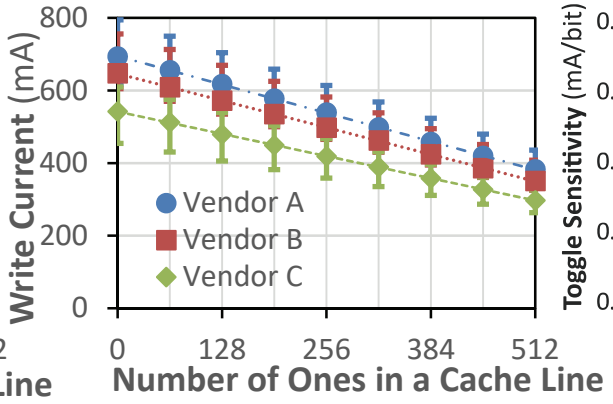
	Read			Write		
	F (mA)	G (mA)	H (mA)	F (mA)	G (mA)	H (mA)
Vendor A	246.44	0.433	0.0515	531.18	-0.246	0.0461
Vendor B	217.42	0.157	0.0947	466.84	-0.215	0.0166
Vendor C	234.42	0.154	0.0856	368.29	-0.116	0.0229

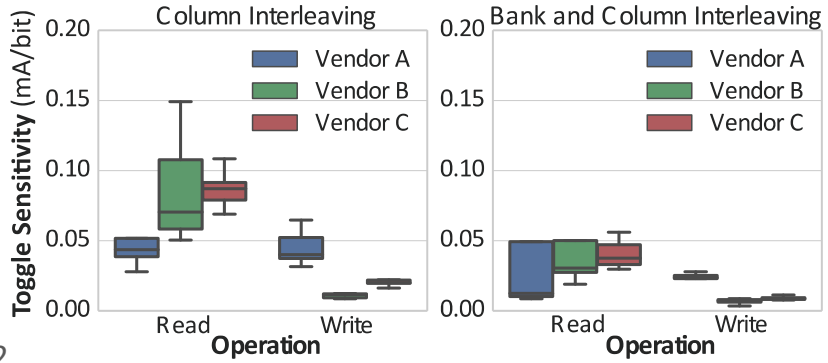
$$y = F + Gn + Ht$$

Additional current per logic-1

Additional current per bit toggle







Models

SAFARI

No Interleaving (Same Bank & Column)

Vendor	Read			Write		
	F	*G*	*H*	*F*	*G*	*H*
A	250.88	0.449	0	489.61	-0.217	0.0000
B	226.69	0.164	0	447.95	-0.191	0.0000
C	222.11	0.134	0	343.41	-0.000	0.0000

Column Interleaving Only

Vendor	Read			Write		
	F	*G*	*H*	*F*	*G*	*H*
A	246.44	0.433	0.0515	531.18	-0.246	0.0461
B	217.42	0.157	0.0947	466.84	-0.215	0.0166
C	234.42	0.154	0.0856	368.29	-0.116	0.0229

Bank Interleaving Only

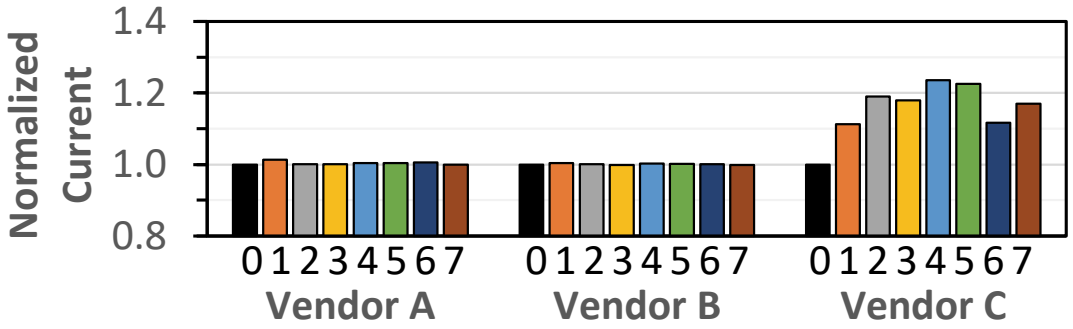
Read			Write		
F	*G*	*H*	*F*	*G*	*H*
287.24	0.244	0.0200	534.93	-0.249	0.0225
228.14	0.159	0.0364	419.99	-0.179	0.0078
289.99	0.034	0.0455	304.33	-0.054	0.0455

Bank and Column Interleaving

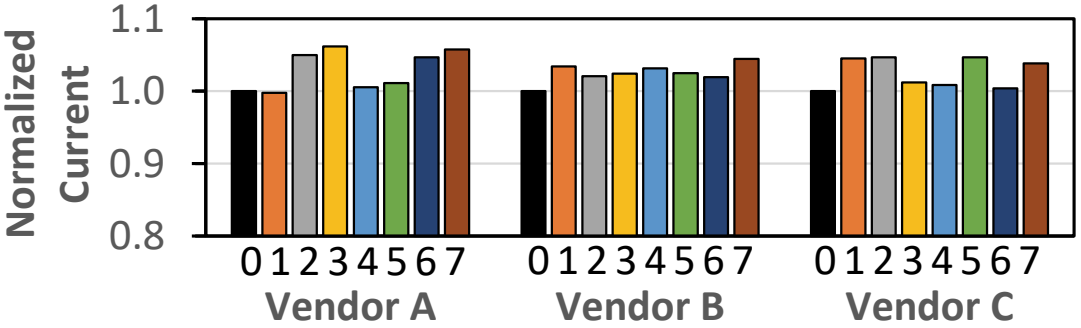
Read			Write		
F	*G*	*H*	*F*	*G*	*H*
277.13	0.267	0.0200	537.58	-0.249	0.0225
223.61	0.152	0.0364	420.43	-0.179	0.0078
266.51	0.099	0.0090	323.22	-0.072	0.0090

https://github.com/CMU-SAFARI/VAMPIRE

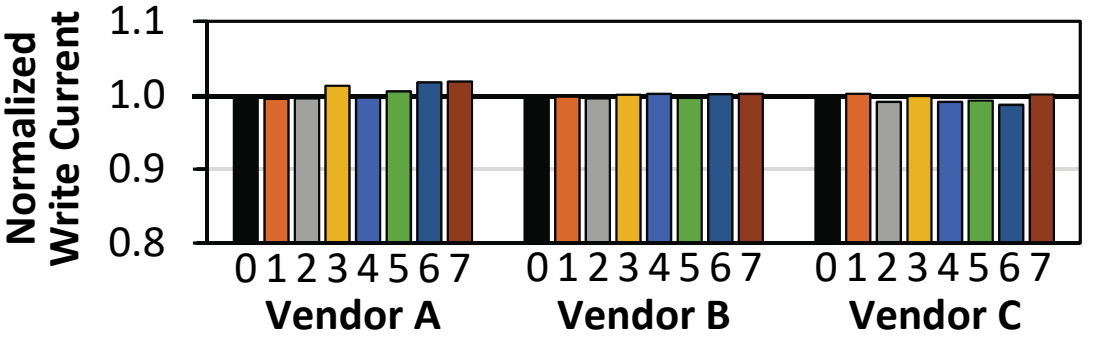
Structural Variation



Normalized **Active Standby** Energy across Banks



Normalized **Read** Burst Energy across Banks



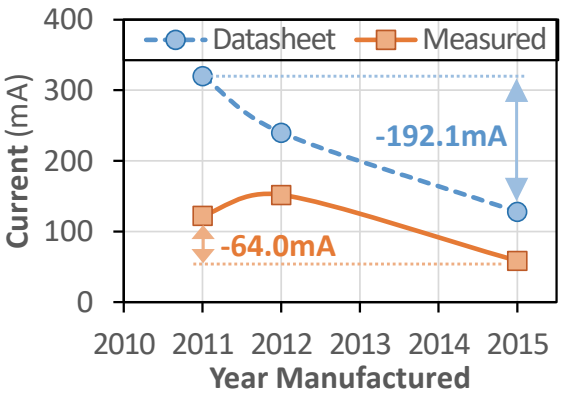
Normalized **Write Burst** Energy across Banks

Evaluated System Configuration

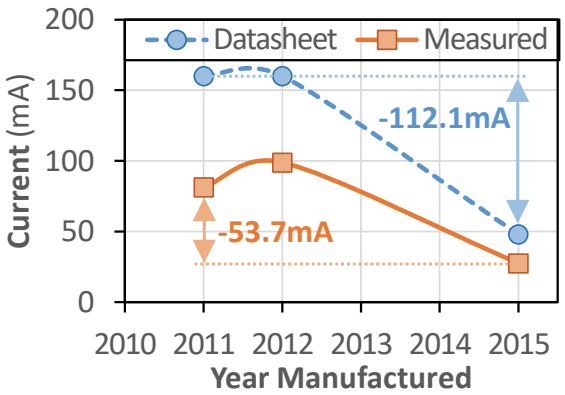
Processor	x86-64 ISA, one core 3.2 GHz, 128-entry instruction window
Cache	L1: 64 kB, 4-way associative; L2: 2 MB, 16-way associative
Memory Controller	64/64-entry read/write request queues, FR-FCFS [119, 149]
DRAM	DDR3L-800 [57], 1 channel, 1 rank/8 banks per channel

- Application traces collected using Pin
- DRAM command timings generated using Ramulator: https://github.com/CMU-SAFARI/ramulator

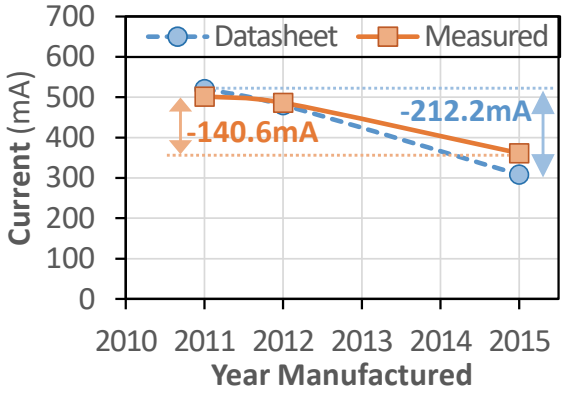
Trends Across Generations



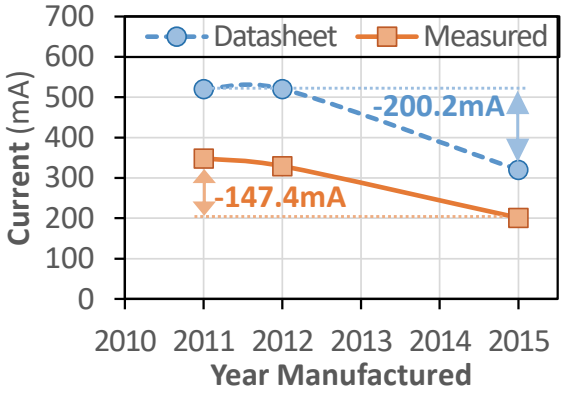
Activation Energy



Precharge Standby Energy



Read Energy

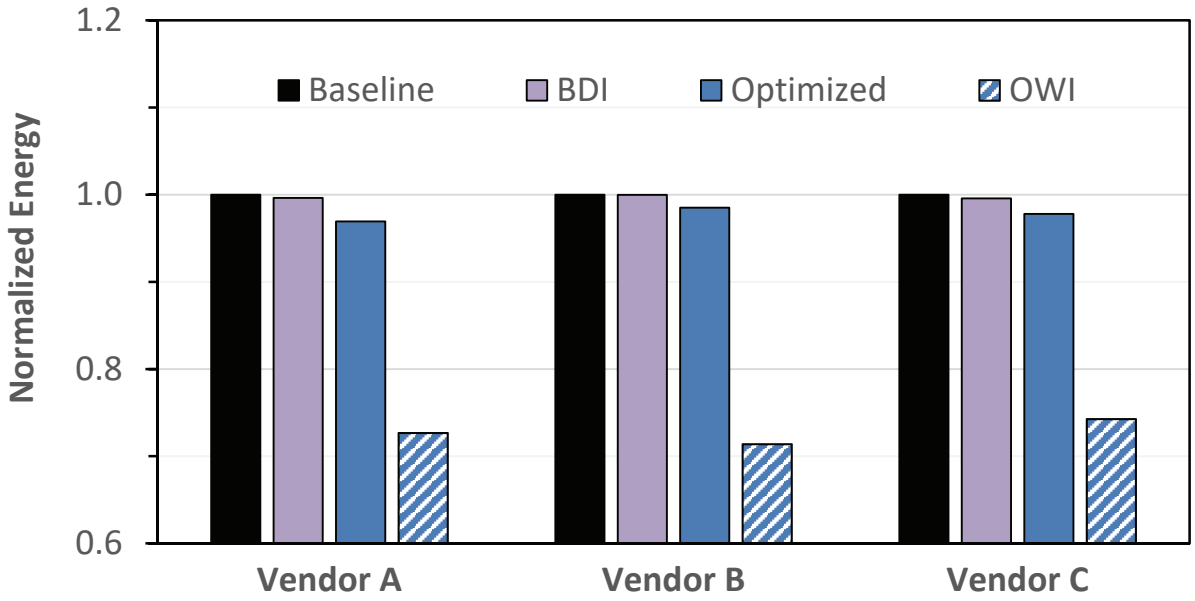


Write Energy

- Basically, if you're building a system, you aren't getting the kinds of savings you were promised

Data Encoding

- Baseline: No coding
- BDI: Base Delta Immediate
- Optimized: Minimize the number of ones
- OWI: Minimize ones for reads, maximize ones for writes



12.5% Energy Reduction

Validating Our DRAM Power Models

- New tests run on 22 of our DDR3L DRAM SO-DIMMs
- Validation command sequence
  - Activate
  - *n* reads
    - » Sweep *n* from 0 to 764
    - » All reads contain data value 0xAA
    - » All reads to Bank 0, Row 128
    - » Column interleaved
  - Precharge
- Error metric: mean absolute percentage error (MAPE)

- Best prior model (DRAMPower): 32.4% MAPE
- VAMPIRE: 6.8% MAPE