Architecting Phase Change Memory as a Scalable DRAM Alternative

Benjamin C. Lee Engin Ipek Onur Mutlu Doug Burger

June 2009 ISCA

Presented by Moritz Herting

19.03.2020

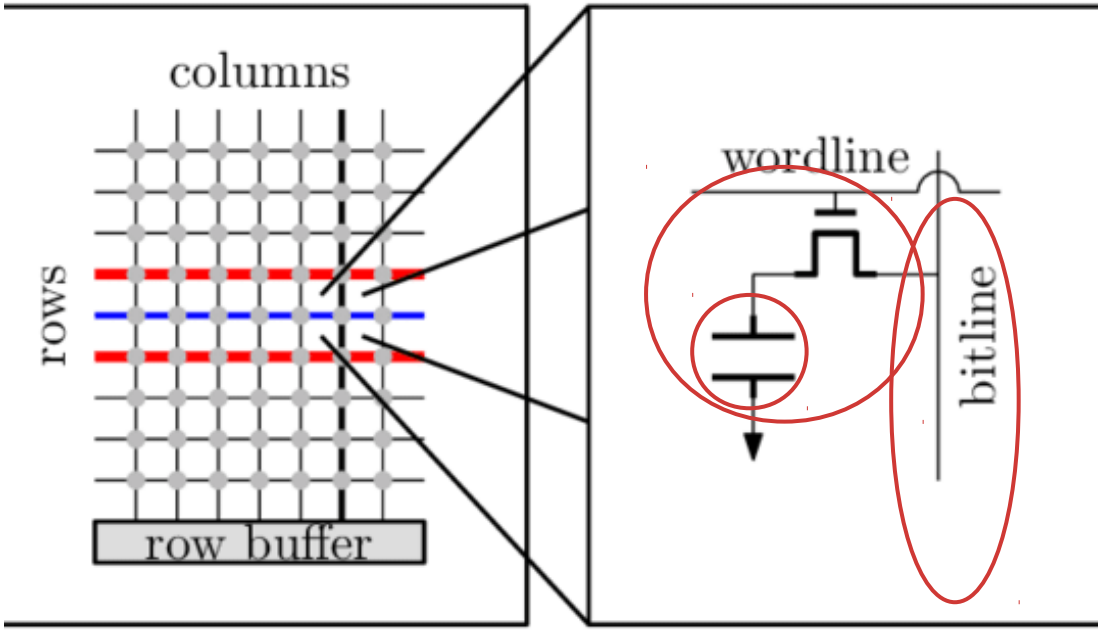
Summary

- **DRAM is hard to scale down**
  - Scaling down decreases Power Consumption and increases Capacity
- **Can we replace DRAM with PCM?**
  - PCM is easy to scale down
- **Get Latency, Power Consumption and Area onto the same Level**
- **Rearrange Buffer and introduce Partial Writes**
- **Evaluate different Configurations which use the same Area and compare Latency, Power draw and Endurance**
- **First to show how to use PCM Technology to architect main Memory that is close to DRAM Performance and has other Advantages as well**

Outline

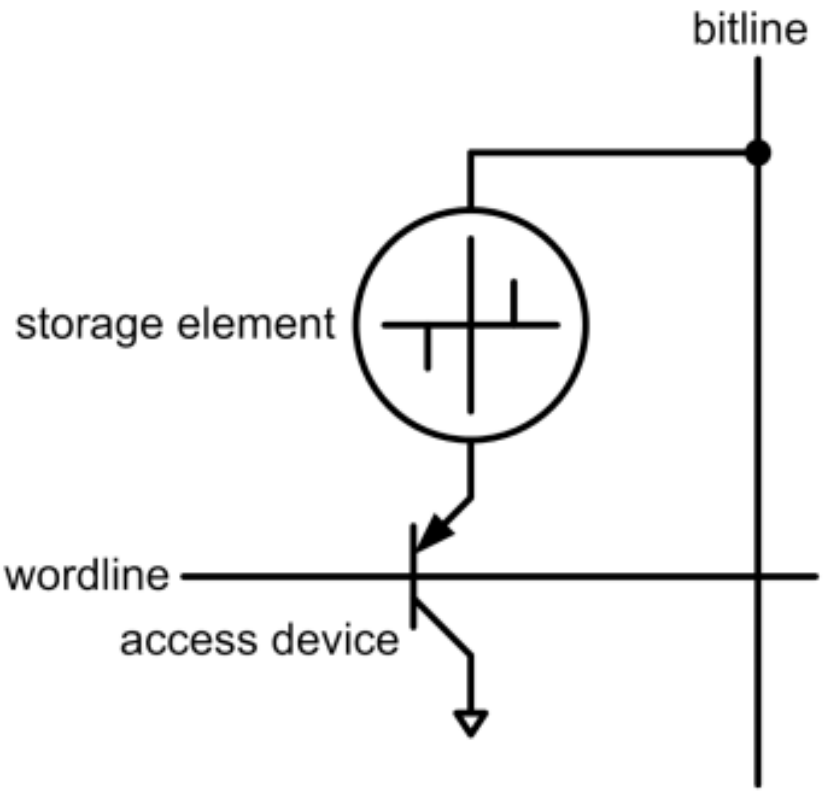
- Basics of DRAM and PCM
- Experimental Methodology
- Architectural Changes
- Process Scaling Improvements
- Conclusion
- Discussion

DRAM Structure



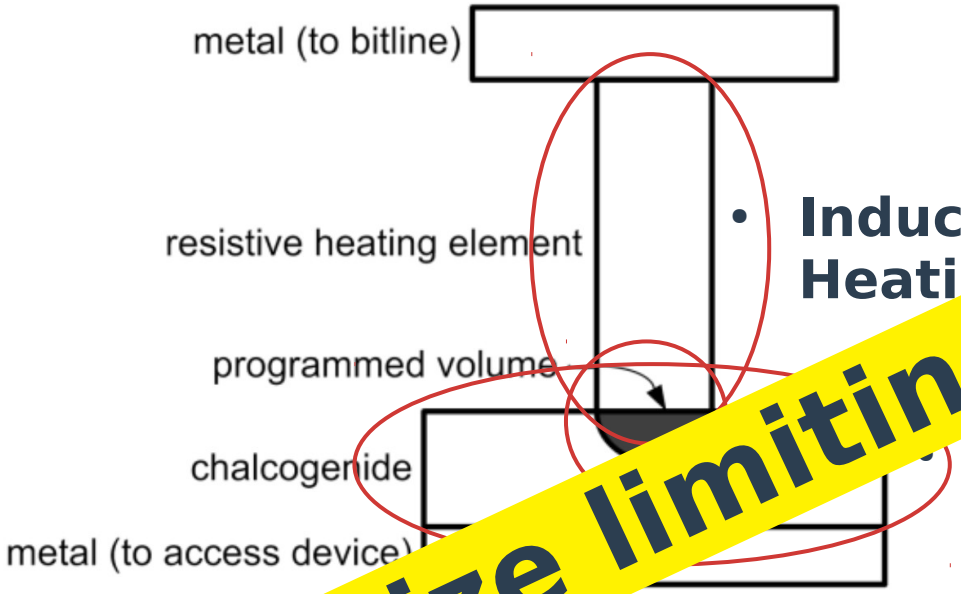
- **DRAM Cell consists of one Capacitor and one Transistor**
- **Store a Bit**
  - Charge/Discharge the Capacitor
- **Read a Bit**
  - The Charge of the Capacitor gets directly to the Buffer via the Bitline
- **Not easily scalable**
  - Smaller Capacitors have smaller Charge Capacity
  - Smaller Access Transistors increase Charge Leakage
  - Harder to store Charge for a long Time

PCM Structure



- **Similar general Structure as DRAM**
- **Different Storage Elements**
- **Needs a special Sense Amplifier and can’t connect directly to the Buffer**
- **Data gets stored by changing the physical Property of a Material**

PCM Storage Element



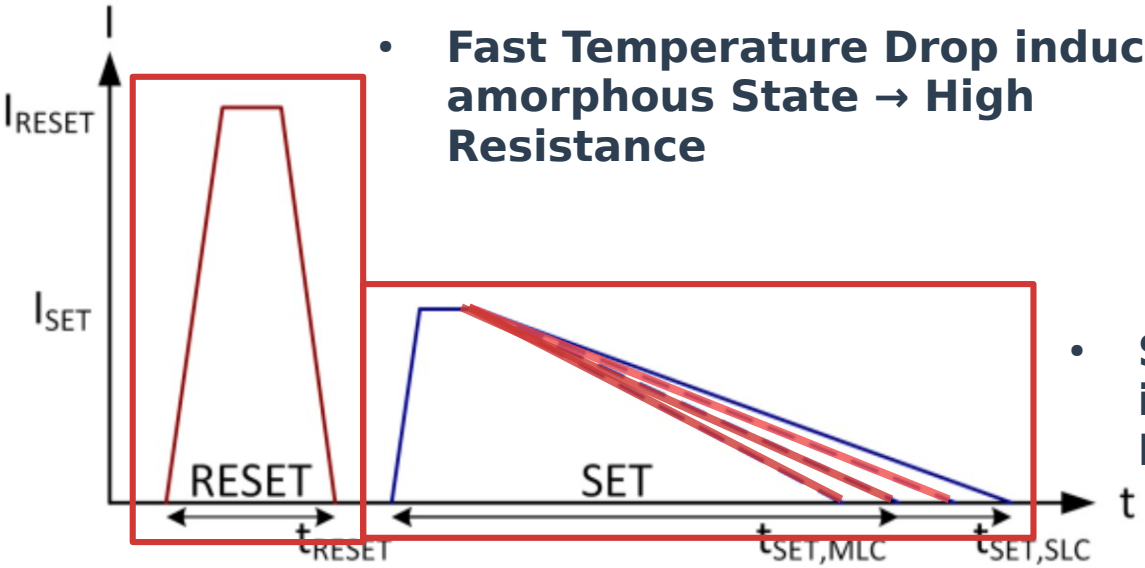
- **Induce Curr… the Heatin…**

Heat up the Chalcogenide to 650°C

…down Time determines …Resistance of the Chalocogenide

No size limiting element!

Writing to a Cell



- **Fast Temperature Drop induces amorphous State → High Resistance**
- **Slow Temperature Drop induces crystalline State → Low Resistance**
- **Intermediate States are possible → Multi Level Cells**

Reading from a Cell

- **Single Level Cells**
  - Low Resistance => 0
  - High Resistance => 1
- **Multi Level Cells**
  - Lowest Resistance => 00
  - Higher Resistances => 01, 10 or 11

DRAM vs PCM Cell

	DRAM [DDR2]		PCM
• Read Latency	5 cycles	4.4x	22 cycles
• Write Latency	5 cycles	12x	60 cycles
• Read Energy	1.17 pJ/bit	2.11x	2.47 pJ/bit
• Write Energy	0.39 pJ/bit	43.12x	16.82 pJ/bit
• Area	6 F^2 /ce	Multi level cells	F^2 /cell
• Scalable	Not easily		Yes
• Volatile	Yes		No
• Refreshes	Yes		No

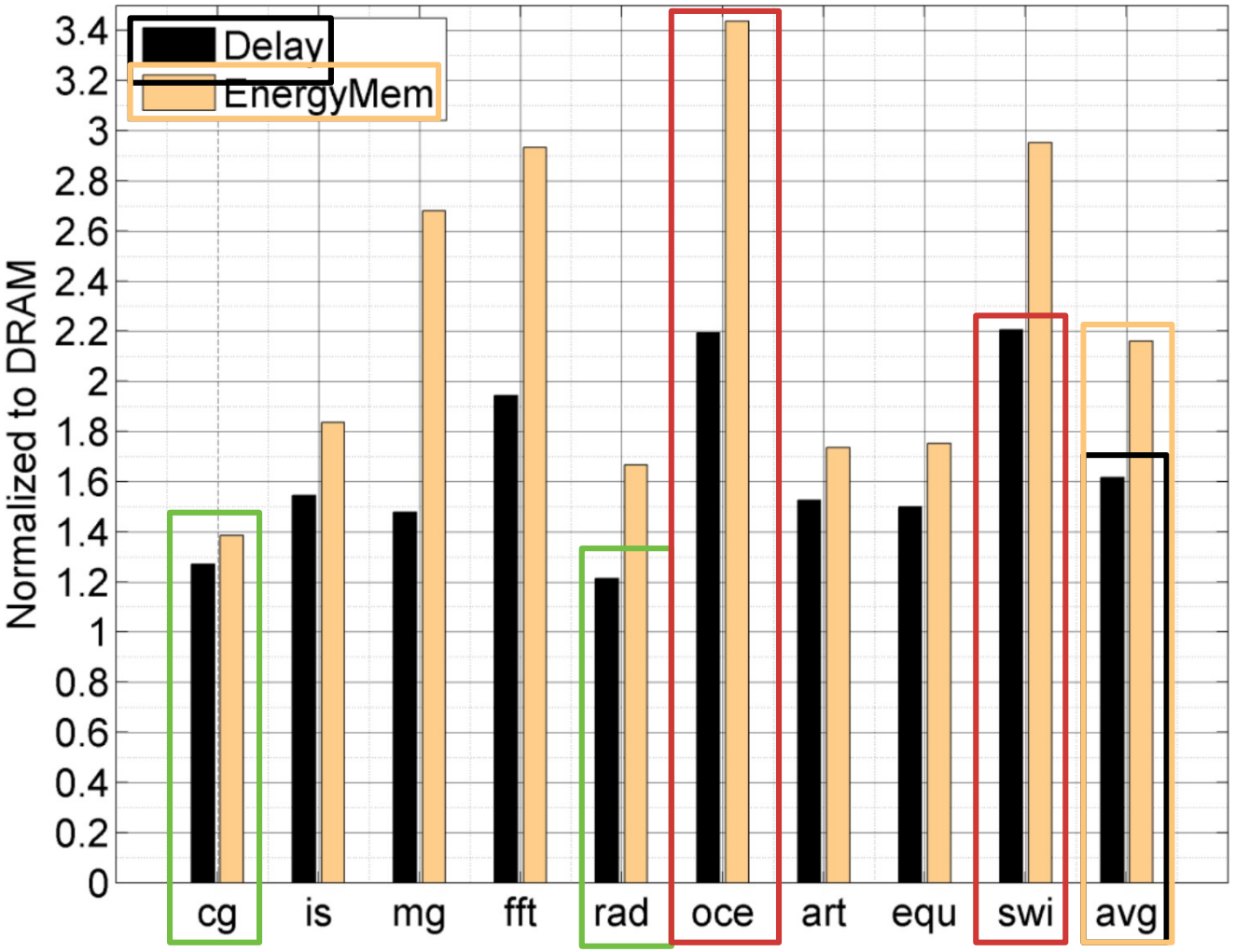
Outline

- Basics of DRAM and PCM
- Experimental Methodology
- Architectural Changes
- Process Scaling Improvements
- Conclusion
- Discussion

Methodology

- **Impact on Applications**
  - Latency
  - Power Consumption
- **Endurance**
- **Simulation using SESC**
- **4 Core Superscalar, Out-of-Order CPU @4GHz**
- **Parallel Workloads**
- **Memory intense Workloads**

Performance/Energy Baseline of PCM



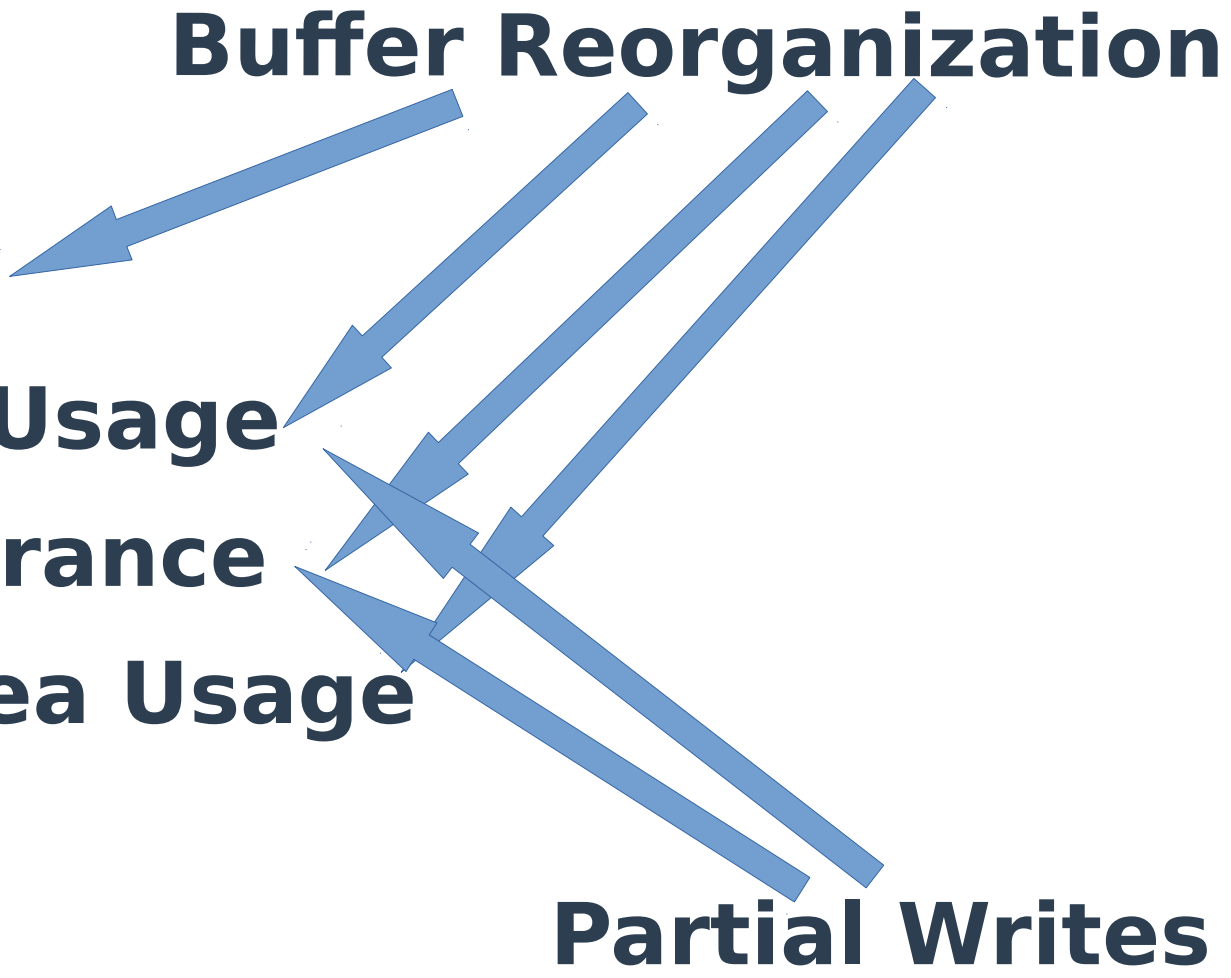
- **Latency**
  - 1.2x up to 2.2x
  - 2.16x on Average
- **Energy**
  - 1.4x up to 3.4x
  - 2.2x on Average

Outline

- Basics of DRAM and PCM
- Experimental Methodology
- Architectural Changes
- Process Scaling Improvements
- Conclusion
- Discussion

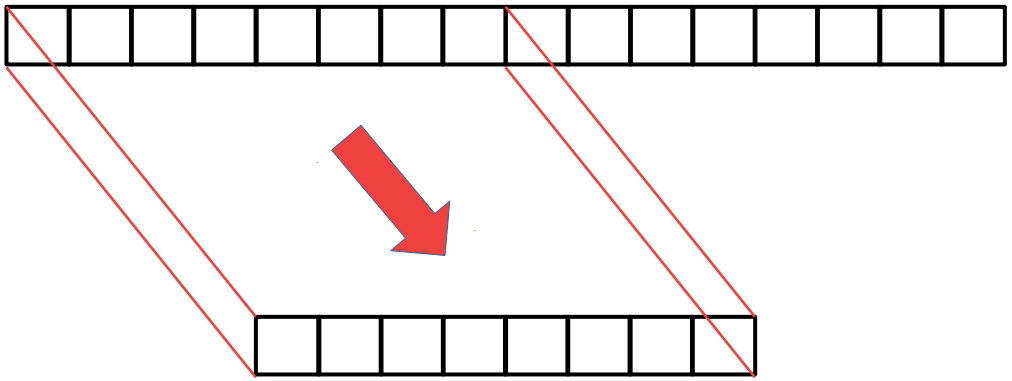
Problems to solve

- High Latency
- High Energy Usage
- Limited Endurance
- Increased Area Usage

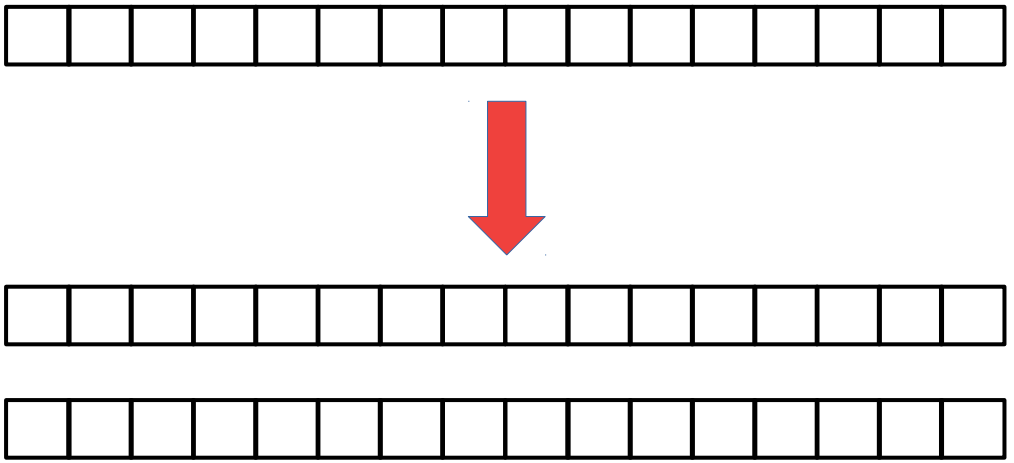


Buffer Reorganization

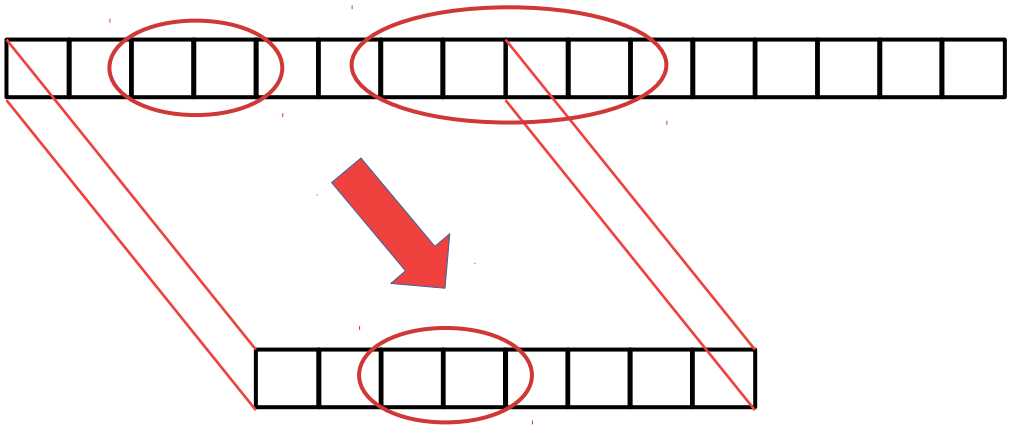
Narrower Rows



Multiple Buffers



Narrow Rows

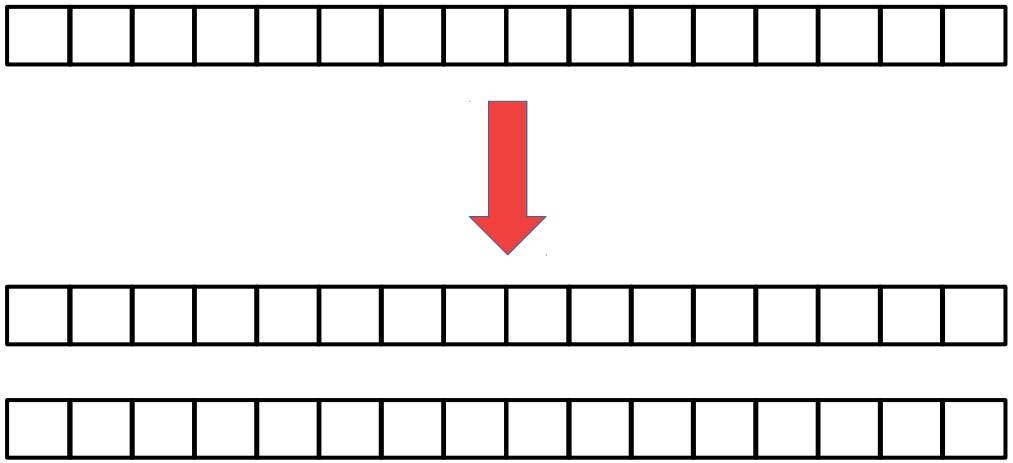


- **Decrease Amount of simultaneous Writes**
  - Decrease Power Draw per Array write
  - Improve Endurance
  - Less write Coalescence
- **Decrease Amount of simultaneous Reads**
  - Decrease Power Draw per Array read
  - Less spacious Coherence
- **Fewer Latches per Buffer**
  - Decrease Power Draw
  - Decreased Area

Multiple Rows

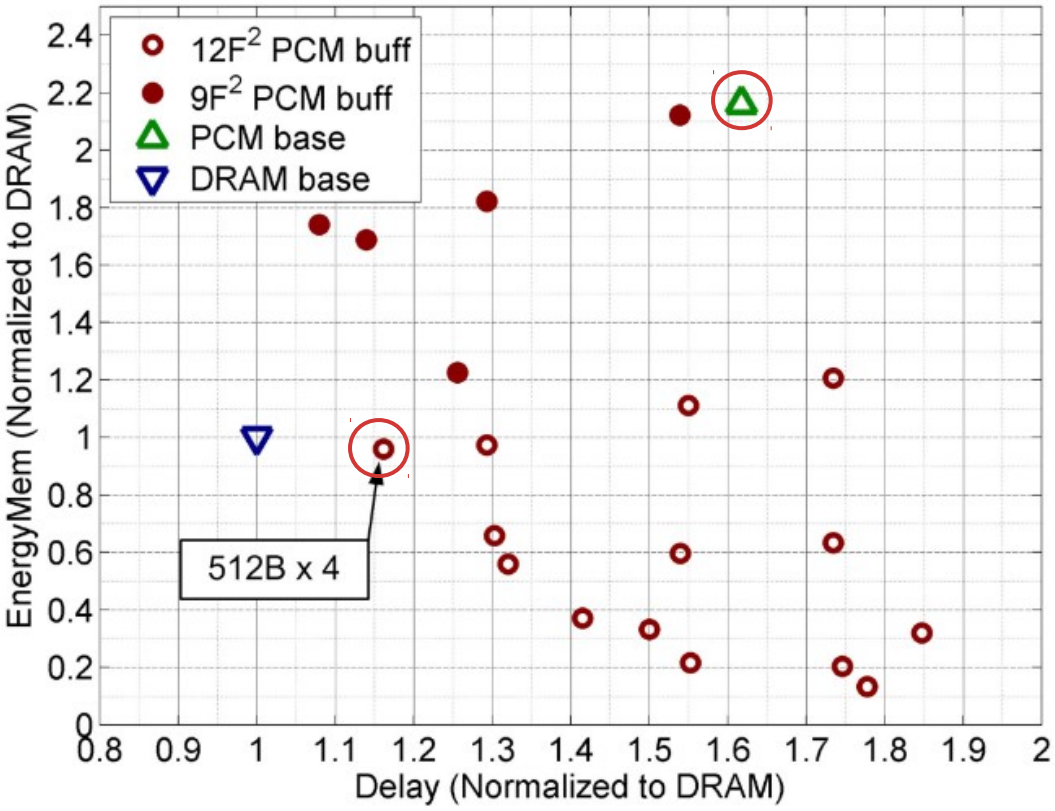
- **Less Conflict Misses**
  - More read and write Coalescence
    - Lower Latency
- **Less frequent Reads and Writes**
  - Lower Power Usage
  - Improves Endurance
- **Increased Area**

Use multiple Buffers



Evaluating Buffer Reorganizations Performance and Energy

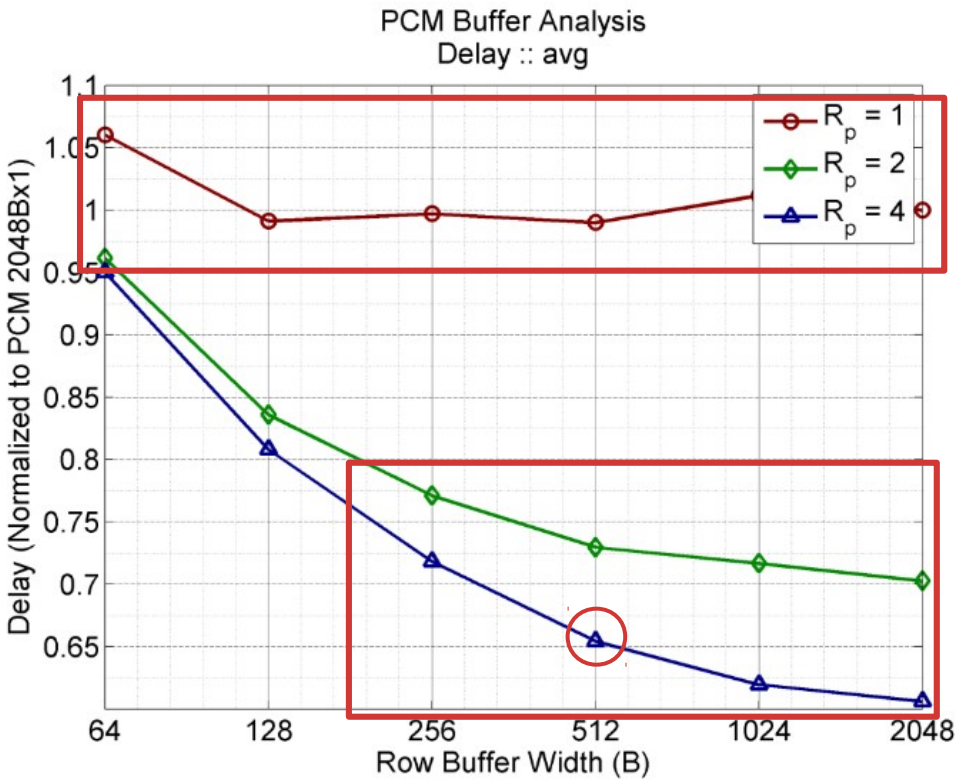
Possible Approaches within the same Area as DRAM



(PCM Baseline would use to much Area)

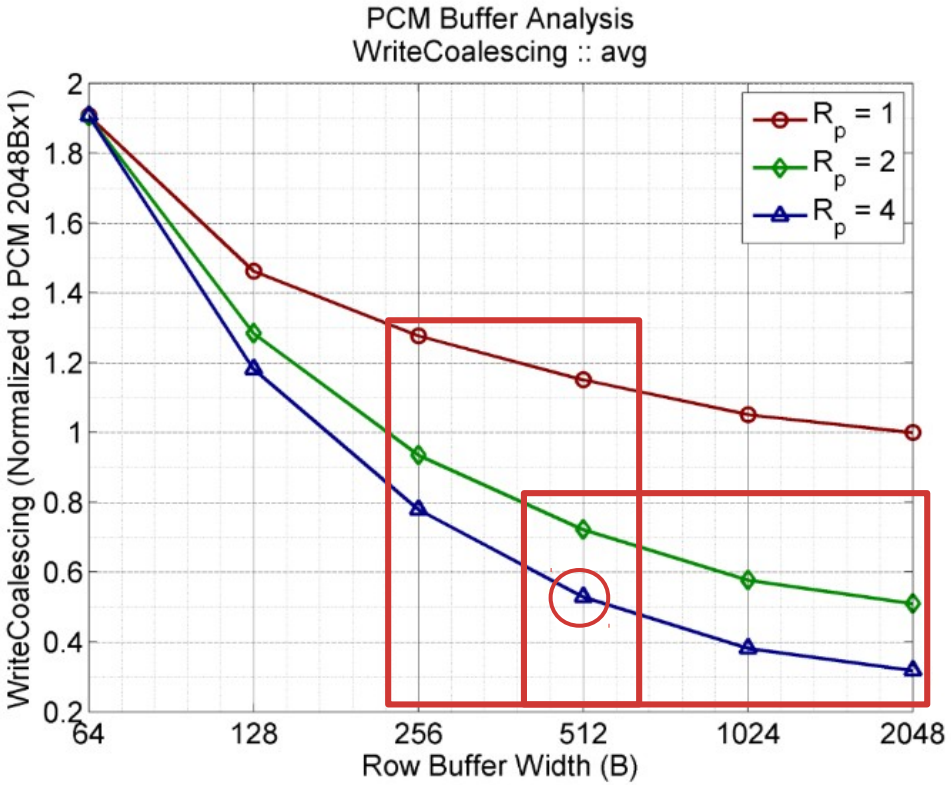
- Huge Difference in Performance and Energy Usage depending on the Approach
- Smaller $9F^2$ Cells wouldn't enable us better Approaches
- 4 x 512B seems like a good Approach
  - Reduced Latency from 1.6x times (Baseline) to 1.16x
  - Reduced Power Consumption from 2.2x to about the same Level

Performance of Row Buffer Configurations



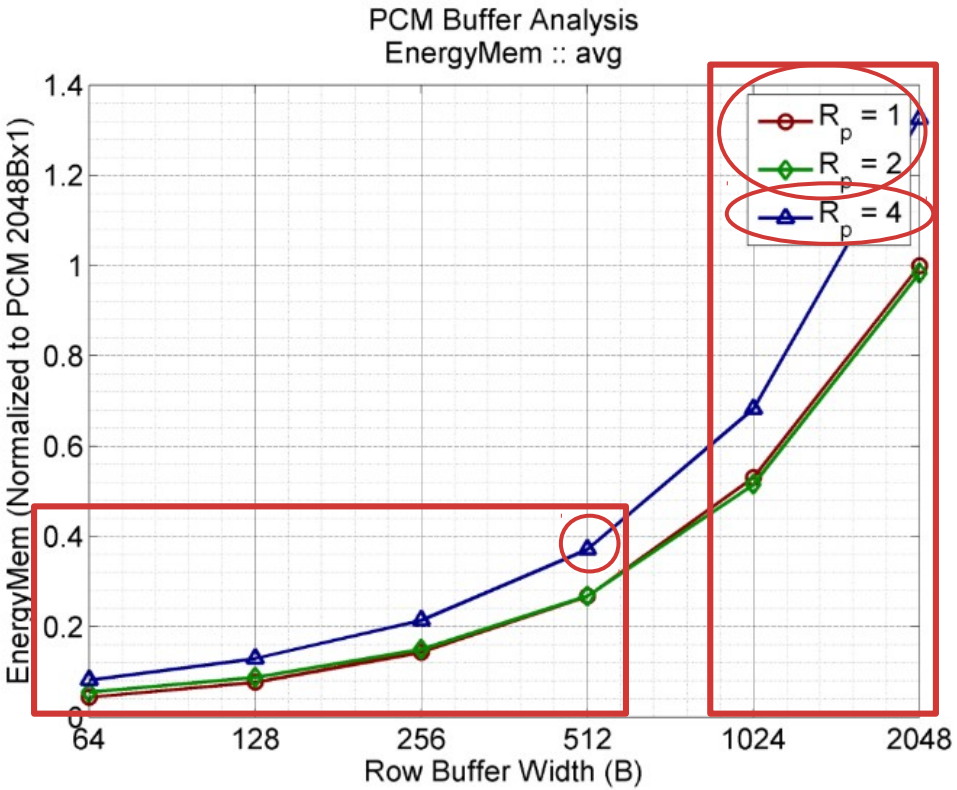
- **Single buffer**
  - Not much spacial Locality
  - Will get evicted to fast for temporal Locality
- **Multiple decently sized Buffers**
  - Able to use temporal Locality
- **4x512B Buffer lead to 66% as much Delay as one 2048B Buffer**

Write Coalescing different Approaches



- Multiple not to small Buffers significantly decrease the Number of Writes
- More/bigger Buffers won’t significantly decrease the Number of Writes
- 4x512B Buffer lead to 53% less Writes compared to one 2048B Buffer

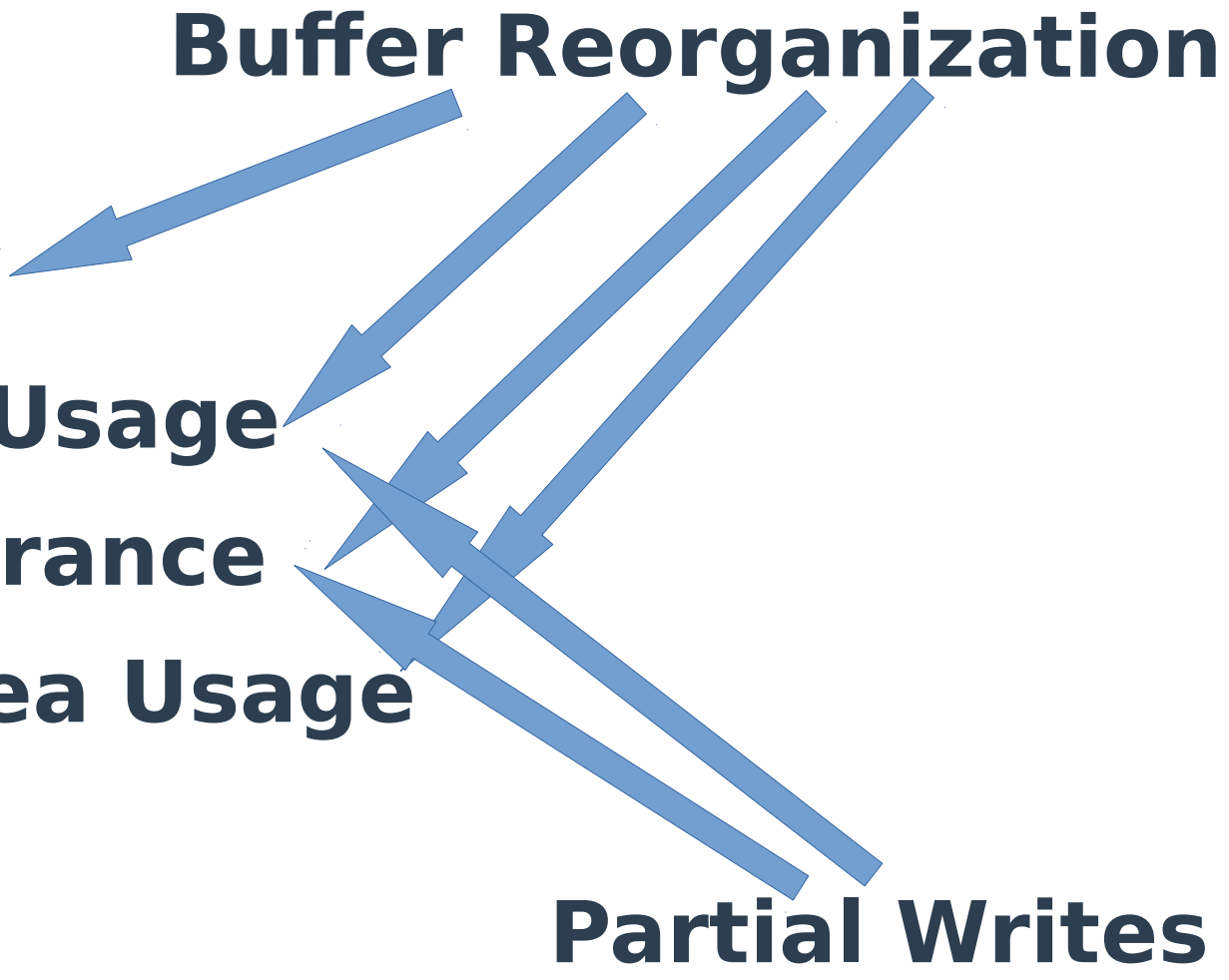
Energy Usage of different Approaches



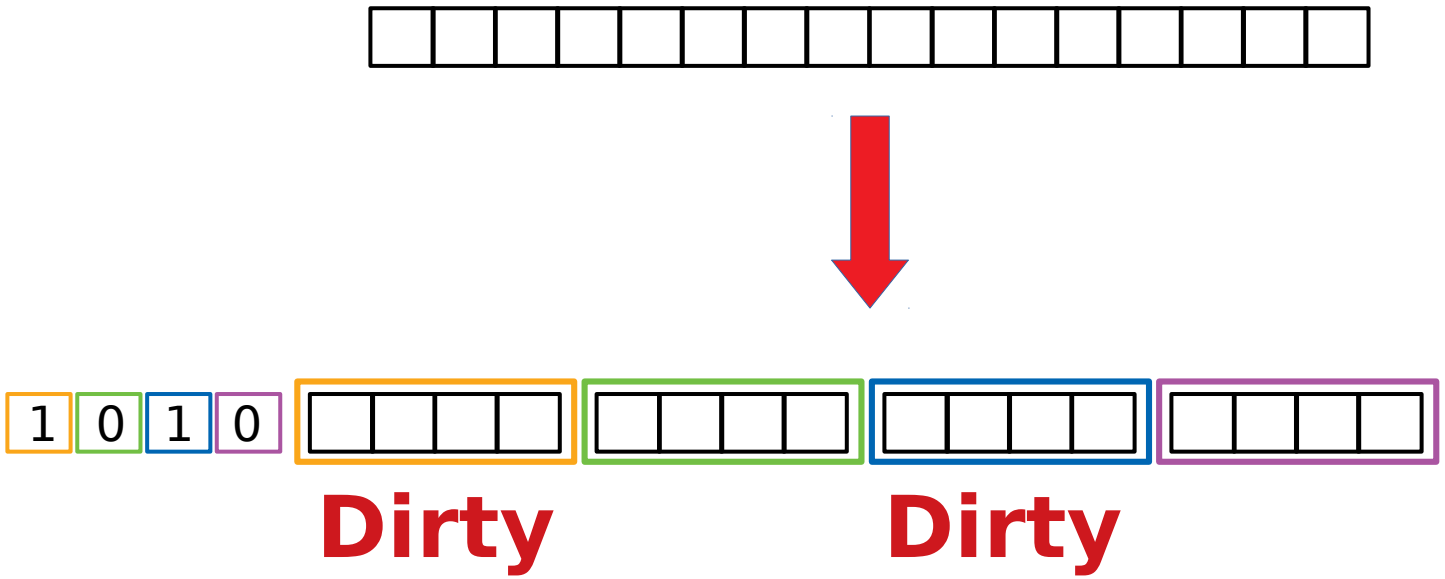
- One and two Buffers use a similar Amount of Energy
  - Fewer Reads and Writes impact Energy Consumption way more than doubling the Row Buffer
- 4 Buffers won't use way more Energy than just two
- Increasing the Width of small Buffers won't use much more Energy than they save from less Cell Accesses
- Increasing the Width of big Buffers won't save us enough Cell Accesses to justify the additional Energy Consumption of the Buffer itself
- 4x512B Buffer is a good Middle Ground

Problems to solve

- High Latency
- High Energy Usage
- Limited Endurance
- Increased Area Usage



Partial Writes Idea



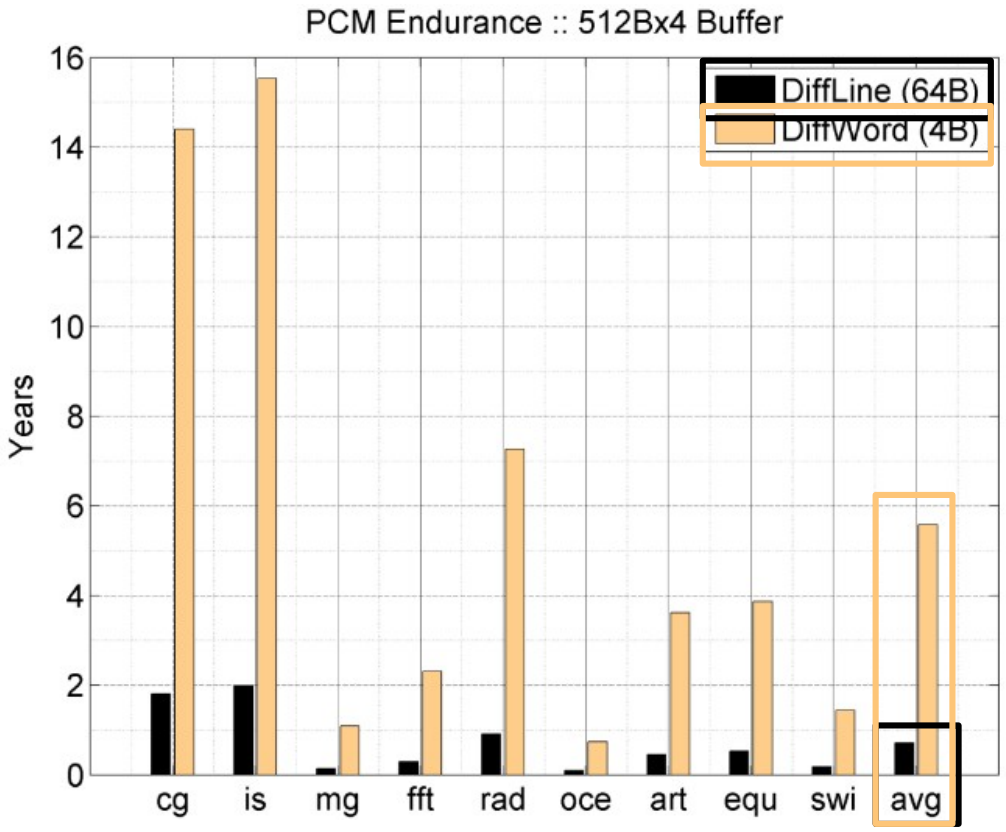
Partial Writes Functionality

- **Decreases the average Amount of written Bits per Array Write**
- **Reduce total number of Cell Writes**
  - Enhance Endurance
  - Decrease Power Consumption
- **Store one dirty Bit per Block**
- **Buffer Reordering will Accommodate for Area Overheads**
- **Requires very small Changes in CPU Cache Structure to include those dirty Bits**
- **64B and 6B Approach**

Partial Writes required Changes

- **64B Blocks**
  - Tracking begins at L2 Cache
  - Requires one Bit per L2 Cache Line
  - 0.2% Overhead in L2 Cache
  - No Change in L1 Cache needed
- **4B Blocks**
  - Tracking begins at L1 Cache
  - Requires 16b per L2 Cache Line
  - Requires 6b per L1 Cache Line
  - 3.1% Overhead in each Cache

Partial Writes Endurance Evaluation



- 0.7 Years with 64B Blocks
- 5.6 Years with 4B Blocks
- Would increase by a Factor of 4 with 32nm Process Size
  - **~700 Years with 64B**
  - **~5’600 Years with 4B**

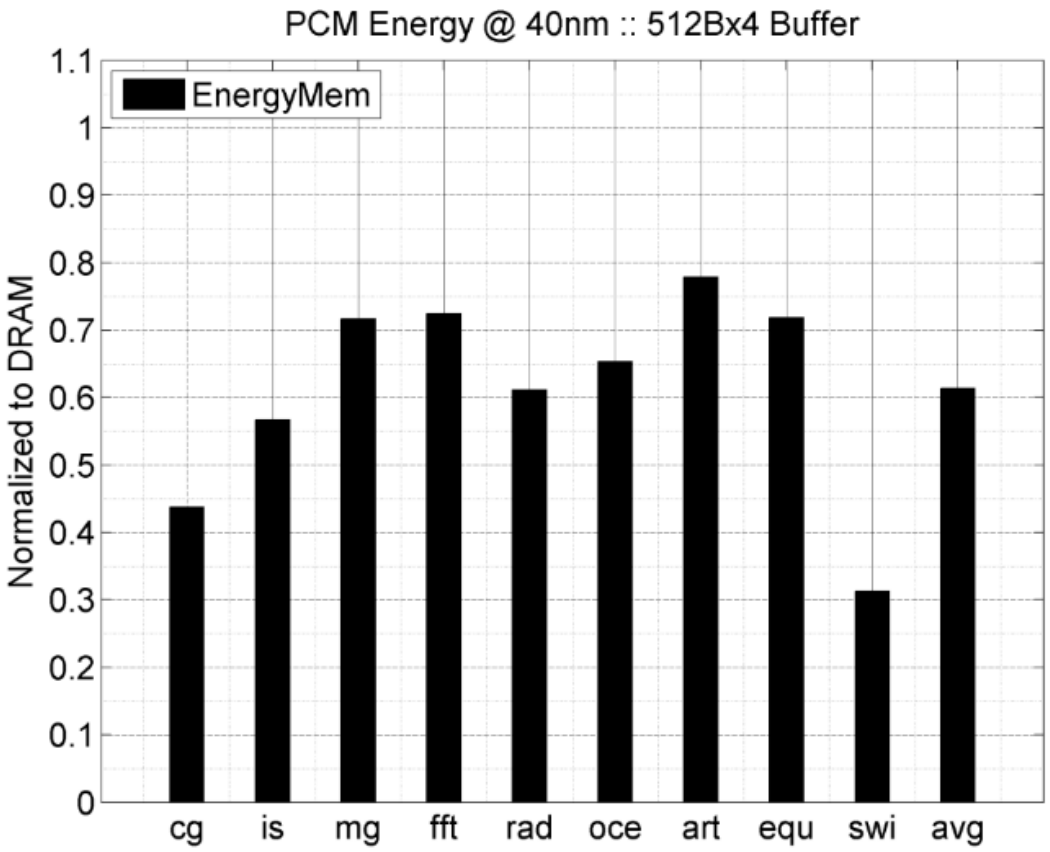
Outline

- Basics of DRAM and PCM
- Experimental Methodology
- Architectural Changes
- Process Scaling Improvements
- Conclusion
- Discussion

Process Scaling Benefits

- **Can further reduce PCM Energy Costs**
- **Improve Endurance further**
- **Increase Density**
  - Increase total Capacity
  - Decrease Price/Capacity Ratio
- **Won’t decrease Latency**

Scaling Improvements from 90nm to 40nm



- **PCM will use 61.3% of Energy compared to DRAM**
- **Will decrease Power by another 2.4x**
- **DRAM only decreases Power by 1.5x**
- **PCM scales 1.6x faster**

Conclusion

- **DRAM is hard to scale down**
  - Scaling down decreases Power Consumption and increases Capacity
- **Can we replace DRAM with PCM?**
  - PCM is easy to scale down
- **Get Latency, Power Consumption and Area onto the same Level**
- **Rearrange Buffer and introduce Partial Writes**
- **Evaluate different Configurations which use the same Area and compare Latency, Power draw and Endurance**

Strengths and Weaknesses

- **The Good:**
- **Good Structure**
- **Most of the important Numbers and Assumptions are clear and Sources are easily retractable**
- **Almost all important Aspects are evaluated**
- **The Bad:**
- **No Energy Evaluation with Partial Writes**
- **Only Memory intensive Workloads have been looked at**
  - Maybe some unforeseen Behavior
- **Some more Numbers would have been nice**
  - DRAM Scaling, presumably 90nm as well
  - Expected Run Time per Year for the Endurance Evaluation, presumably 24/7/365
  - IPC for CPU Simulation
- **Rather exact Numbers regarding they are extrapolated from Simulations/Predictions**
  - Maybe some expectable Derivations

Situation Today

- **DDR4**
  - Can be produced in 12nm Process Node Size
  - Similar Cell Latency compared to DDR2
  - Less than half the Power Consumption of DDR2
  - Higher Data Rate
- **Buffer Reorganization for DRAM has been proposed in 2011**
  - 35.8% improved Performance (4 core)
  - 42% Reduction in Energy (4 core)
- **Low-Latency PCM**
  - 119% higher Performance than normal PCM
  - 43% less Energy
- **PDRAM**
  - Hybrid System
  - 30% Energy Savings
- **Optane/3D XPoint**
  - Might be based on PCM
  - Hard to find exact Numbers
  - Similar Latency to DDR4
  - 1/3 of the Bandwith
  - Already in use in Enterprise Solutions as an Addition to DRAM
  - In use by consumers as an HDD Cache to cheaply bring Performance to a similar Level as an SSD

Situation Today – Further Readings

- **Samsung 12nm DDR4 Chip**
  - https://www.golem.de/news/ddr4-speicher-samsung-hat-dritte-10-nm-generation-entwickelt-1903-140184.htm
- **DRAM Buffer Reorganization**
  - https://ieeexplore.ieee.org/document/6113809
- **Low-Latency PCM**
  - https://dl.acm.org/doi/10.1145/3316781.3317853
- **PDRAM (Hybrid System)**
  - https://ieeexplore.ieee.org/abstract/document/5227100
- **Optane**
  - https://www.hardwaretimes.com/what-is-intel-optane-memory-heres-how-it-works-and-why-its-important/
- **NVRAM Standard Proposal (video)**
  - https://youtu.be/xxpF5oVZsrA

Scalable Alternatives

- **Flash (also mentioned in the Paper)**
  - Slow
  - https://de.wikipedia.org/wiki/Flash
- **Static RAM**
  - Only volatile Alternative
  - https://de.wikipedia.org/wiki/Static_random-access_memory
- **Optane/3D XPoint**
  - Not available at the Time of Publication
  - https://www.hardwaretimes.com/what-is-intel-optane-memory-heres-how-it-works-and-why-its-important/
- **Optical PCM**
  - Not yet available
  - https://www.youtube.com/watch?v=UWMEKex6nYA (video)
  - https://www.osapublishing.org/ol/abstract.cfm?uri=ol-44-7-1821
- **Ferroelectric RAM**
  - https://de.wikipedia.org/wiki/Ferroelectric_Random_Access_Memory
- **Resistive RAM**
  - https://de.wikipedia.org/wiki/Resistive_Random_Access_Memory
- **Magnetoresistive RAM**
  - https://de.wikipedia.org/wiki/Magnetoresistive_Random_Access_Memory
- **Nanotube based RAM**
  - https://de.wikipedia.org/wiki/NRAM
  - https://www.youtube.com/watch?v=V1HN0w_aJgg

Own Thoughts

- **Good Addition to volatile DRAM**
- **No full replacement in Performance oriented Devices**
- **Maybe a suitable replacement in Business oriented Ultrabooks/Laptops**
- **Probably coming more to Consumer Products soon**
  - Optane
- **Hybrid Systems**
  - Power Consumption
  - Security
  - Performance
- **Security Implications**
  - Encrypted Hard Drive
    - Volatile Encryption/Decryption Unit
    - Volatile Accessor on the same Die which has to be cryptically unlocked after each Power Loss

- **Others say Holy Grail is Persistence up until CPU Registers**

Discussion Topics

- **What are some use Cases for persistent main Memory in Applications?**
- **What could be the Place of PCM within Today's Computers?**
- **If we could have PCM good enough to replace Memory up until CPU Registers**
  - Would there still be a Reason for Volatile Memory?
  - Which Applications and Use cases could profit from this and how?
- **Can you come up with some Workloads where Persistence could be important/valuable enough to justify some Loss in Performance and/or Power Consumption?**
- **Do you think PCM would have better Chances when looking at Applications which are less Memory intensive**
- **Do you think there is a Difference between Workloads who have a relative higher Amount of**
  - Reads
  - Writes
- **Do you think there would be a Difference when looking at Consumer Workloads and Usage instead of Enterprise Usage**
  - Gaming
  - Operation System Performance
  - Browsing the Eeb
  - Office Work (PowerPoint, Word, Excel)
- **Do you think it would be easy/quick to change Applications to better make use of persistent Memory**
- **Can you come up with some Workloads where Persistence could be a Disadvantage and why?**
  - Security
    - Drive Encryption

Architecting Phase Change Memory as a Scalable DRAM Alternative

Benjamin C. Lee Engin Ipek Onur Mutlu Doug Burger

Discussion Link:

https://moodle-app2.let.ethz.ch/mod/forum/discuss.php?d=45327#p89471

June 2009 ISCA

Presented by Moritz Herting

19.03.2020

Appendix

		PCM	DRAM
Array			
A	bank size (MB)	16	16
C	cell size (F^2)	9MLC, 12MLC	6
Periphery			
S	sense amplifer (T @ $250\lambda^2$/T)	44	14
	sense amplifer (F^2)	2750	875
L	latch (T @ $250\lambda^2$/T)	8	0
	latch (F^2)	500	0
D	decode 2-AND (T @ $1000\lambda^2$/T)	6	0
	decode 2-AND (F^2)	250	0
Buffer Organization			
W	buffer width (B)	64::2x::2048	2048
R	buffer rows (ea)	1::2x::32	1

$$\hat{A}_D = \underbrace{A \cdot C_D}_{\texttt{array}} + \underbrace{W_D \cdot S_D}_{\texttt{sense}}$$

$$\hat{A}_P = \underbrace{A \cdot C_P}_{\texttt{array}} + \underbrace{W_P \cdot S_P}_{\texttt{sense}} + \underbrace{R_P \cdot W_P \cdot L_P}_{\texttt{latch}} + \underbrace{R_P \cdot G(log_2 R, 2) \cdot D_P}_{\texttt{decode}}$$

Appendix

Endurance		
$\hat{W}$	writes per second per bit	calc
$\hat{L}$	memory module lifetime (s)	calc
E	write endurance	1E+08
Memory Module		
C	logical capacity (Gb)	2
Memory Bus Bandwidth		
f_m	memory bus frequency (MHz)	400
M_f	processor frequency multiplier	10
B	burst length (blocks)	8
Application Characteristics		
N_w, N_r	number of writes, reads	sim
T	execution time (cy)	sim
Buffer Characteristics		
W_P, R_P	buffer width (B), rows	512, 4
N_{wb}, N_{wa}	buffer, array writes	sim
δ	fraction of buffer written to array	sim

$$\hat{W} = \underbrace{\frac{f_m}{B/2} \cdot \frac{(N_w + N_r) \cdot (B/2) \cdot M_f}{T}}_{\texttt{memBusOcc}} \times \underbrace{\frac{N_w}{N_w + N_r}}_{\texttt{writeIntensity}} \times \underbrace{8 W_P \cdot \left(\frac{N_{wa}}{N_{wb}}\right) \cdot \delta}_{\texttt{bufferOrg}} \times \underbrace{\frac{1}{C/2}}_{\texttt{capacity}} \quad (3)$$