# Block Disabling Characterization and Improvements in CMPs Operating at Ultra-low Voltages

**A. Ferrerón**[1], D. Suárez-Gracia[2], J. Alastruey-Benedé[1],
T. Monreal[3], V. Viñals[1]

[1]Universidad de Zaragoza, Spain

[2]Qualcomm Research Silicon Valley, USA

[3]Universidad Politécnica de Cataluña, Spain

SBAC-PAD Oct-2014

# Operation near the threshold voltage ($V_{th}$)

### $V_{dd}$ and $V_{th}$ scaling has stopped

Power density no longer stays constant among technology generations and dark silicon appears

# Operation near the threshold voltage ($V_{th}$)

## $V_{dd}$ and $V_{th}$ scaling has stopped

Power density no longer stays constant among technology generations and dark silicon appears

## Operation at ultra-low $V_{dd}$

- Reduce the power and energy consumption
- Switch on more cores to exploit parallelism

# Operation near the threshold voltage ($V_{th}$): Challenges

Delay increases: lower voltage → lower frequency

- Compensate with parallelism: more active cores with the same power budget

# Operation near the threshold voltage ($V_{th}$): Challenges

Delay increases: lower voltage → lower frequency

- Compensate with parallelism: more active cores with the same power budget

Increasing sensitivity to process variation (deviation of device parameters from their nominal values)

- Memory structures especially sensitive to variation
  - Conventional 6T cells: read, write, access, and hold failures
  - Lower voltages → stability margins decrease → increasing cell failure rate
  - $V_{dd_{min}}$ of memory blocks to guarantee reliable operation

## Objective

Lower $V_{dd}$ to near-threshold voltages → energy efficient operation

### Objective

Lower $V_{dd}$ to near-threshold voltages → energy efficient operation

### Problem

High sensitivity of SRAM structures to variation at ultra-low $V_{dd}$

### Objective

Lower $V_{dd}$ to near-threshold voltages → energy efficient operation

### Problem

High sensitivity of SRAM structures to variation at ultra-low $V_{dd}$

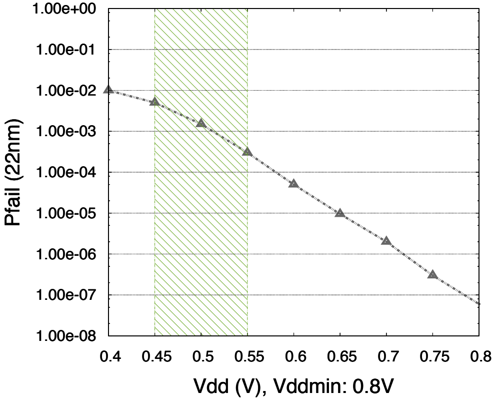
### Our proposal

Mitigate the impact of SRAM cell failures at ultra-low $V_{dd}$ using low complexity techniques:
Block Disabling with Operational Tags and Block Disabling with Operational Tags and Cache-to-cache Transfers

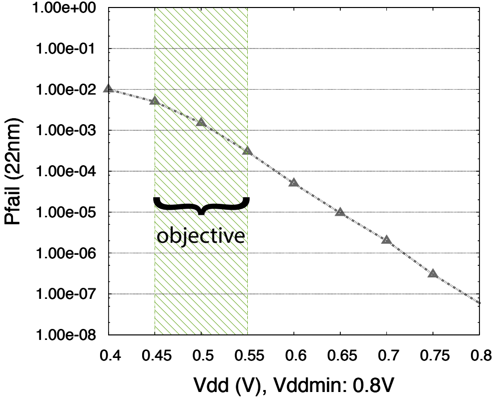
# Outline

# Outline

# Example of Probability of Failure of SRAM Cells at 22nm

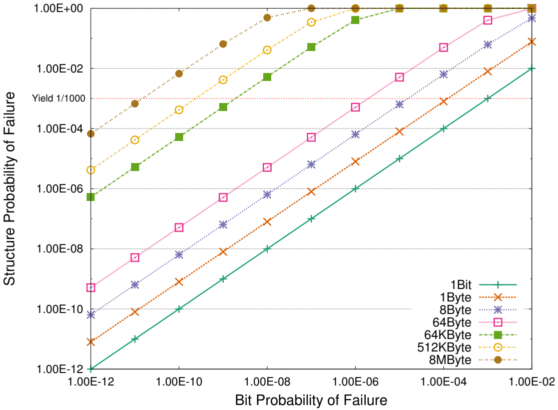


# Example of Probability of Failure of SRAM Cells at 22nm

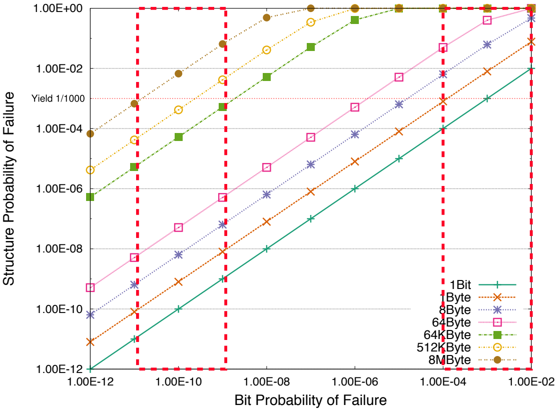


Universidad Zaragoza

# Bit Probability of Failure Affects Yield

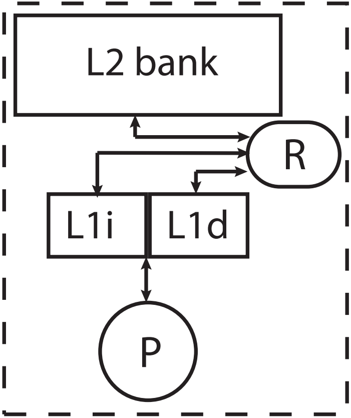


# Bit Probability of Failure Affects Yield

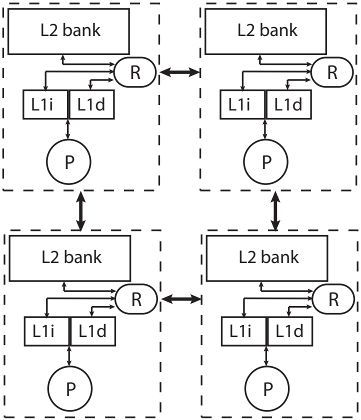


# Outline

# Traditional Cache Hierarchy

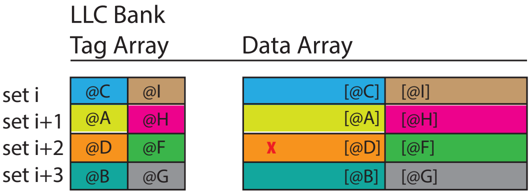


# Traditional Cache Hierarchy



# Block Disabling Fundamentals

SRAM cell failure detected:
Block Disabling (BD) deactivates entry (tag and data)
Simple implementation and low overhead: 1 bit per cache entry

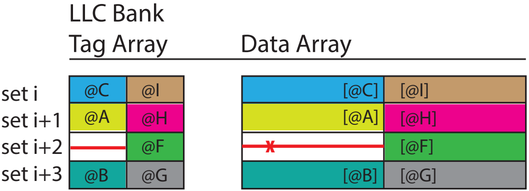


LLC Bank

|  | Tag Array |  | Data Array |  |
|---|---|---|---|---|
| set i | @C | @I | [@C] | [@I] |
| set i+1 | @A | @H | [@A] | [@H] |
| set i+2 | @D | @F | X [@D] | [@F] |
| set i+3 | @B | @G | [@B] | [@G] |

# Block Disabling Fundamentals

SRAM cell failure detected:
Block Disabling (BD) deactivates entry (tag and data)
Simple implementation and low overhead: 1 bit per cache entry



LLC Bank

Tag Array

| | | |
|---|---|---|
| set i | @C | @I |
| set i+1 | @A | @H |
| set i+2 | | @F |
| set i+3 | @B | @G |

Data Array

| | | |
|---|---|---|
| set i | [@C] | [@I] |
| set i+1 | [@A] | [@H] |
| set i+2 | ✗ | [@F] |
| set i+3 | [@B] | [@G] |

# Block Disabling at Ultra-low Voltages

At lower voltages capacity and associativity degrade very fast

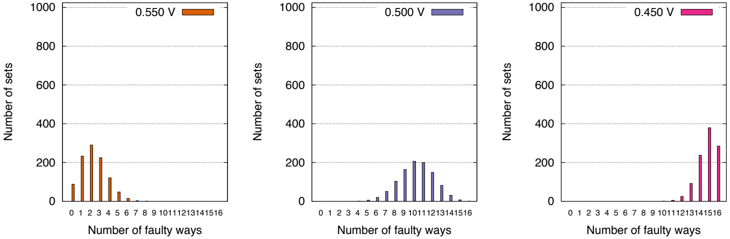
- Available capacity for 16-way, 1MB cache bank with block disabling (block size is 64 bytes):

| **Vdd** | **Available capacity (KB)** |
|---|---|
| 0.55V | 887 KB (86%) |
| 0.50V | 408 KB (40%) |
| 0.45V | 138 KB (13%) |

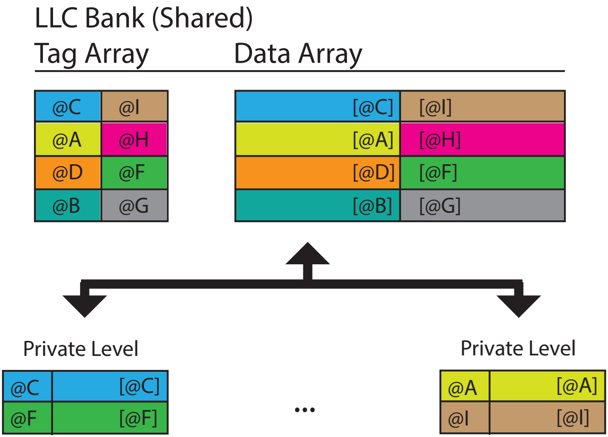
# Block Disabling at Ultra-low Voltages

At lower voltages capacity and associativity degrade very fast

- Associativity degradation for 16-way, 1MB cache bank with block disabling (block size is 64 bytes):



# Inclusive Hierarchies



LLC Bank (Shared)

Tag Array

| | |
|---|---|
| @C | @I |
| @A | @H |
| @D | @F |
| @B | @G |

Data Array

| | |
|---|---|
| [@C] | [@I] |
| [@A] | [@H] |
| [@D] | [@F] |
| [@B] | [@G] |

Private Level

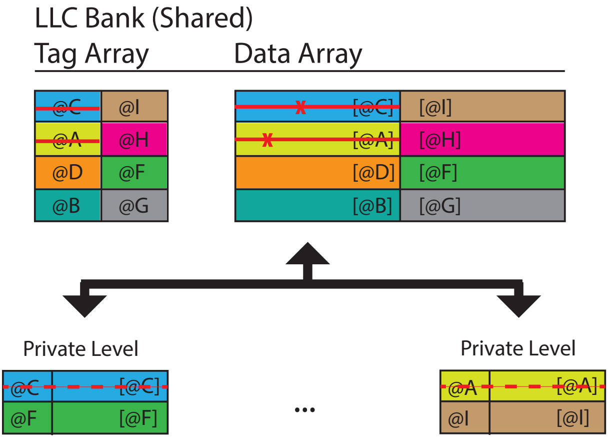
| | |
|---|---|
| @C | [@C] |
| @F | [@F] |

...

Private Level

| | |
|---|---|
| @A | [@A] |
| @I | [@I] |

# Inclusive Hierarchies and Block Disabling Interaction



LLC Bank (Shared)

Tag Array

| | |
|---|---|
| ~~@C~~ | @I |
| ~~@A~~ | @H |
| @D | @F |
| @B | @G |

Data Array

| | |
|---|---|
| ~~[@C]~~ | [@I] |
| ~~[@A]~~ | [@H] |
| [@D] | [@F] |
| [@B] | [@G] |

Private Level

| | |
|---|---|
| ~~@C~~ | ~~[@C]~~ |
| @F | [@F] |

...

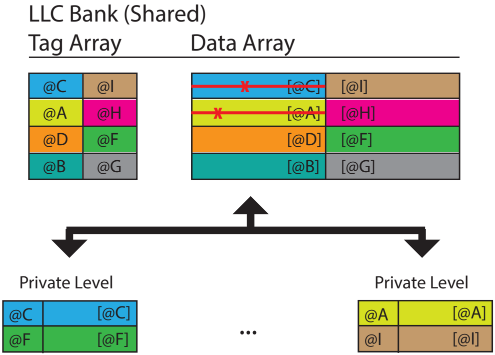
Private Level

| | |
|---|---|
| ~~@A~~ | ~~[@A]~~ |
| @I | [@I] |

# Outline

# BD with operational tags: BDOT

Allow blocks to be allocated as just tags: entries with faulty bits can still be used to allocate tag-only blocks in LLC



# BD with operational tags: BDOT

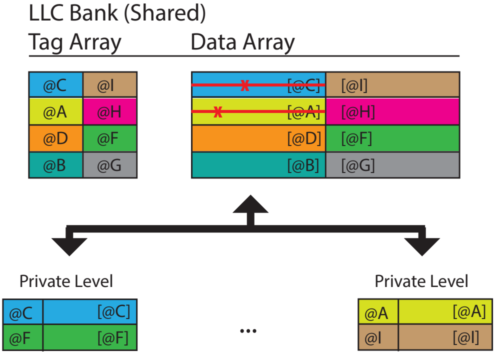
- Protect the tag array
  - Bigger/robust cells: bigger transistors/more transistors per cell (assist circuitry)
  - More complex error correction codes (ECC)

# BD with operational tags: BDOT

- Protect the tag array
  - Bigger/robust cells: bigger transistors/more transistors per cell (assist circuitry)
  - More complex error correction codes (ECC)
- Why not protect the whole cache structure?
  - Area and power increase when using bigger/robust cells
  - Complex ECC require extra storage and checking hardware: might increase access latency
  - Tag array roughly 10% of the cache area (LLC)

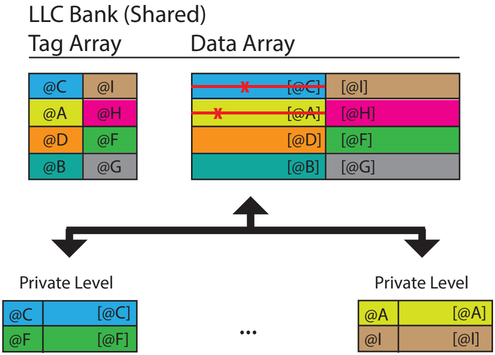
# BDOT with cache-to-cache trasnfers: BDOT-C2C

- Problem: requests to tag-only blocks → off-chip transactions



# BDOT with cache-to-cache trasnfers: BDOT-C2C

- Problem: requests to tag-only blocks → off-chip transactions
- Observation: shared blocks already on-chip (private levels)



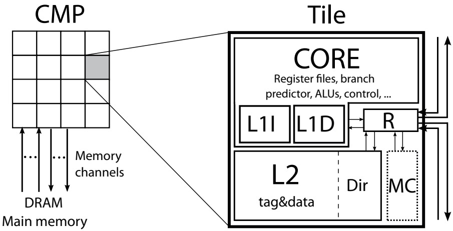
# BDOT with cache-to-cache trasnfers: BDOT-C2C

Provide cache-to-cache transfers of clean blocks: leverage coherence protocol

- The protocol already does cache-to-cache transfers of exclusively owned blocks
- Slight change in the coherence protocol behavior, but no hardware overhead
- Potential gain depends on the applications sharing degree

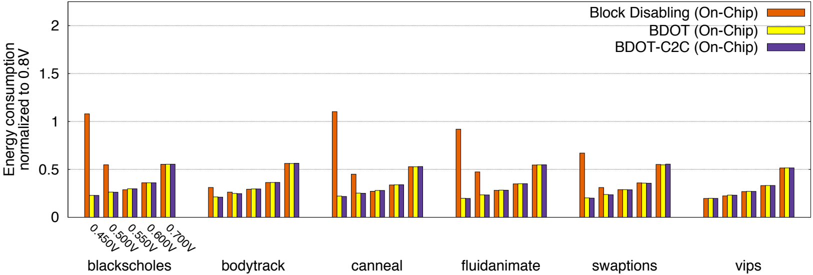
# Outline

# Methodology

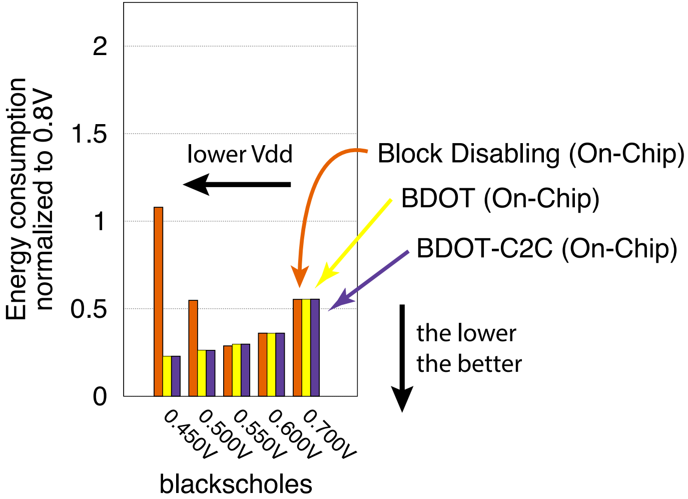


- Experimental set-up:
  Simics + GEMS + GARNET + DRAMSim2 + McPAT
- PARSEC benchmark suite
- Random faults + Monte Carlo simulations

# On-chip Energy Consumption



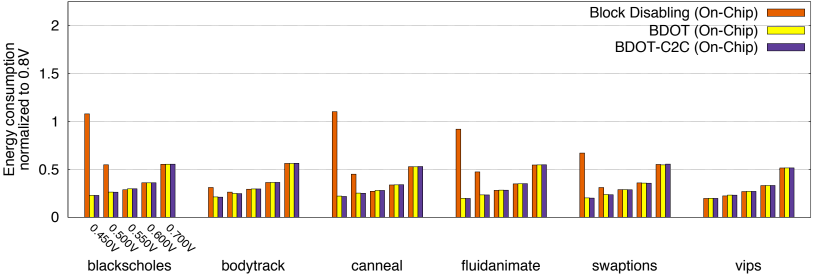
# On-chip Energy Consumption



lower Vdd

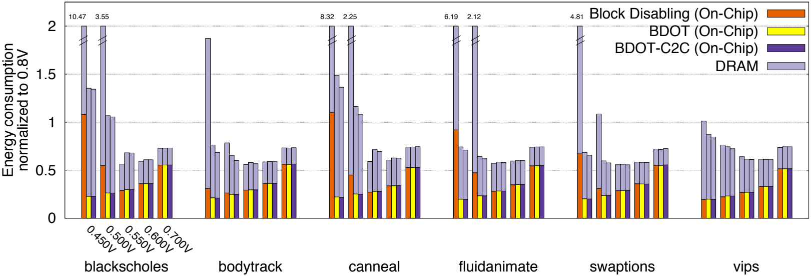
the lower
the better

# On-chip Energy Consumption

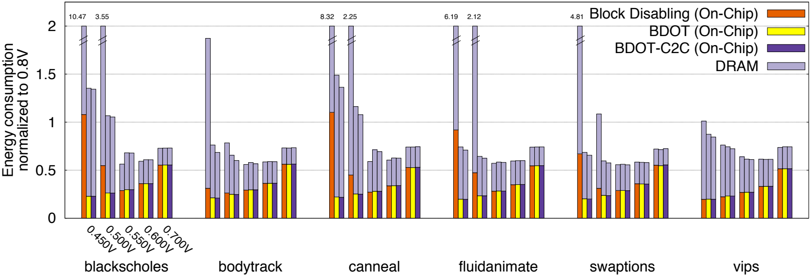


Minimum energy on-chip:
voltages values between 0.45-0.5V more active cores → potential higher performance

# Total Energy Consumption



# Total Energy Consumption



Minimum system energy:

off-chip memory energy consumption main source

higher voltage values (0.55-0.6V)

# Outline

## Conclusions

- Operation near $V_{th}$ for energy efficient operation
  - Switch on inactive cores
  - Reduce the overall energy consumption
- SRAM structures fail when lowering $V_{dd}$
  BD: simple, low overhead, but not effective at ultra-low $V_{dd}$
  Inclusive hierarchies: BD increases inclusion victims
  - BDOT: allow blocks allocated as tag-only → protect inclusion
  - BDOT-C2C: provide cache-to-cache transfers of shared blocks → reduce off-chip transactions
  - BDOT & BDOT-C2C: substantial reduction on-chip power and energy consumption

# Block Disabling Characterization and Improvements in CMPs Operating at Ultra-low Voltages

**A. Ferrerón**[1], D. Suárez-Gracia[2], J. Alastruey-Benedé[1], T. Monreal[3], V. Viñals[1]

[1]Universidad de Zaragoza, Spain

[2]Qualcomm Research Silicon Valley, USA

[3]Universidad Politécnica de Cataluña, Spain

SBAC-PAD Oct-2014