Open-Source Cache Architectures in Chipyard

Phaedra Curlin and Tamara Lehman

OSCAR'26

University of Colorado
Boulder

Agenda

- Open-Source Cache Designs
- Cache Architectures
- Implementation
- Results
- Conclusion

Agenda

- **Open-Source Cache Designs**
- Cache Architectures
- Implementation
- Results
- Conclusion

Secure Cache Architectures

- Cache attacks are a threat to modern computer systems
- Many secure cache architectures proposed to mitigate attacks
- Main focus on performance results
  - Rarely implemented beyond simulators
  - Area and power results omitted/discussed at high level
- Few secure cache designs are open-sourced

How can we facilitate tradeoff analyses between designs to justify overheads introduced by security guarantees?

Open-Source Cache Designs

- Implement and evaluate current secure caches in hardware
- Use complete open-source flow
  - Publicly available cores
    - Rocket Chip[1], BOOM[2], CVA6[3]
    - Rocket Chip Inclusive Cache[4]
  - Design inside of Chipyard[5]
  - Physical implementation using OpenROAD[6]

[1] Asanović et al, “The Rocket Chip Generator”, Technical Report UCB/EECS-2016-17, 2016.

[2] Zhao et al, “SonicBOOM: The 3rd Generation Berkeley Out-of-Order Machine”, Fourth Workshop on Computer Architecture Research with RISC-V, 2020.

[3] Zaruba and Benini, “The Cost of Application-Class Processing: Energy and Performance Analysis of a Linux-Ready 1.7-GHz 64-Bit RISC-V Core in 22-nm FDSOI Technology”, *TVLSI*, 2019.

[4] CHIPS Alliance, “rocket-chip-inclusive-cache”, https://github.com/chipsalliance/rocket-chip-inclusive-cache.

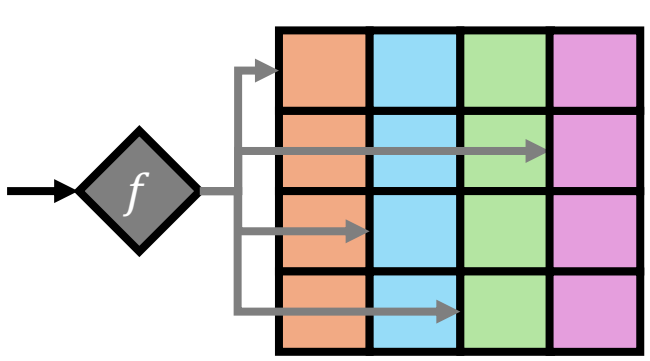
[5] Alon et al, “Chipyard Integrated Design, Simulation, and Implementation Framework for Custom SoCs,” *IEEE Micro*, 2020.

[6] Ajayi et al, “Toward an Open-Source Digital Flow: First Learnings from the OpenROAD Project,” *DAC*, 2019..

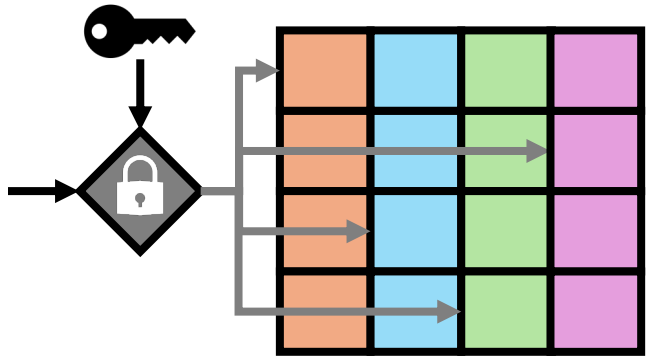
Agenda

- Open-Source Cache Designs
- **Cache Architectures**
- Implementation
- Results
- Conclusion

Cache Architectures



Skewed Associative[1]



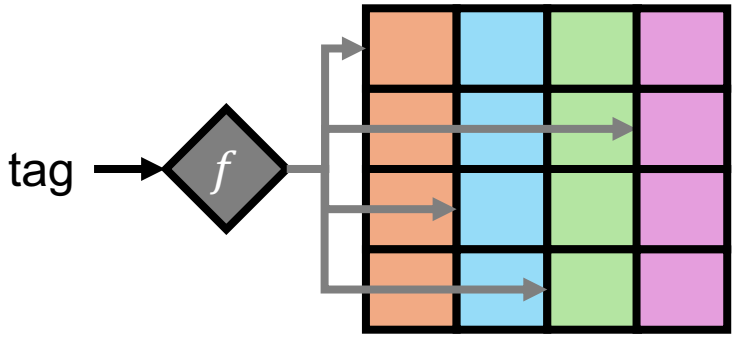
ScatterCache[2]

[1] Seznec and Bodin, “Skewed-associative Caches”, *PARLE*, 1993.

[2] Werner et al, “ScatterCache: Thwarting Cache Attacks via Cache Set Randomization”, *USENIX Security*, 2019.

Skewed Associative Cache

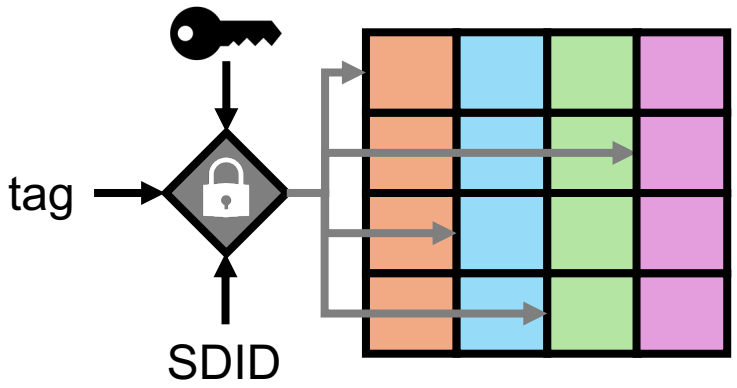
- Different hashing function for different ways
  - Provides illusion of higher associativity in exchange for small area overhead
  - Improves hit ratio



[1] Seznec and Bodin, “Skewed-associative Caches”, *PARLE*, 1993

ScatterCache[1]

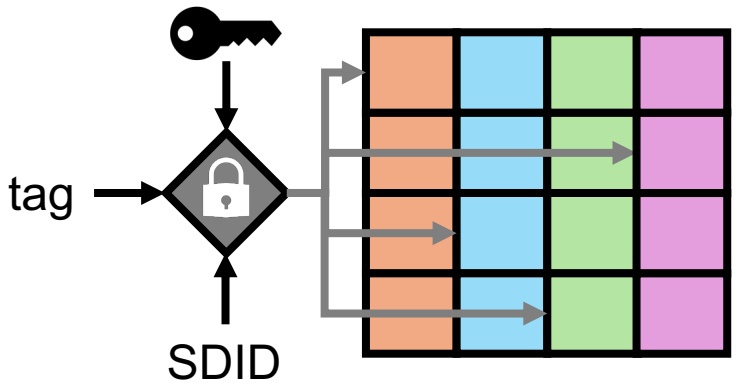
- Last level secure cache architecture
  - Defend against cache attacks
  - Introduce randomness to decouple address from indexing
- Based on skewed associative caches
  - Maximized skewing (1 skew/way)



[1] Werner et al, “ScatterCache: Thwarting Cache Attacks via Cache Set Randomization”, *USENIX Security*, 2019.

ScatterCache[1]

- Cryptographic index mapping function
  - Lightweight cryptographic core
  - Input: tag, secure domain ID, key
  - Output: set index
  - Source of performance overheads



[1] Werner et al, “ScatterCache: Thwarting Cache Attacks via Cache Set Randomization”, *USENIX Security*, 2019.

Agenda

- Open-Source Cache Designs
- Cache Architectures
- **Implementation**
- Results
- Conclusion

Implementation

- Initial design
  - Rocket Chip[1] and Rocket Chip's Inclusive Cache[2]
  - 64KiB, 8 way
- Several modifications needed to achieve
  - Skewed Associative[3]
  - ScatterCache[4]

[1] Asanović et al, "The Rocket Chip Generator", Technical Report UCB/EECS-2016-17, 2016.

[2] CHIPS Alliance, "rocket-chip-inclusive-cache", https://github.com/chipsalliance/rocket-chip-inclusive-cache.

[3] Seznec and Bodin, "Skewed-associative Caches", PARLE, 1993

[4] Werner et al, "ScatterCache: Thwarting Cache Attacks via Cache Set Randomization", *USENIX Security*, 2019.

Implementation – Skewed Associative

- SRAM block does not allow for ways to be accessed independently from the set access.
  - Independent memory block for each way
- Skewing functions use set index, tag, and way
  - Each call to hash function elaborates a separate instance to avoid latency incurred by shared hardware

Implementation – ScatterCache

- Index mapping function can use any cryptographic core
  - Popular choices are QARMA64[1] and PRINCE[2]
- Implement both variants
  - Original work assumes unrolled design
  - Pipeline design (6 stages) due to power overheads introduced
    - Encryption needed for each cache lookup
- Changes made to Directory
  - Base cache assumes single-cycle index derivation
  - Augment Directory with state machine for multi-cycle index calculation

Agenda

- Open-Source Cache Designs
- Cache Architectures
- Modifications to Rocket Chip's Inclusive Cache
- **Results**
- Conclusion

Methodology

- Implemented using Chipyard[1]
  - Rocket Chip[2] and Rocket Chip Inclusive Cache[3]
- Physical design generated with OpenROAD[4]
  - Google/SkyWater Open Source 130nm PDK[5]
- Area and power evaluation
  - Skewed Associative[6]
  - ScatterCache[7]

[1] Alon et al, “Chipyard Integrated Design, Simulation, and Implementation Framework for Custom SoCs,” *IEEE Micro*, 2020.

[2] Asanović et al, “The Rocket Chip Generator”, Technical Report UCB/EECS-2016-17, 2016.

[3] CHIPS Alliance, “rocket-chip-inclusive-cache”, https://github.com/chipsalliance/rocket-chip-inclusive-cache.

[4] Ajayi et al, “Toward an Open-Source Digital Flow: First Learnings from the OpenROAD Project,” *DAC*, 2019.

[5] Google, “skywater-pdk,” https://github.com/google/skywater-pdk.

[6] Seznec and Bodin, “Skewed-associative Caches”, PARLE, 1993.

[7] Werner et al, “ScatterCache: Thwarting Cache Attacks via Cache Set Randomization”, USENIX Security, 2019.

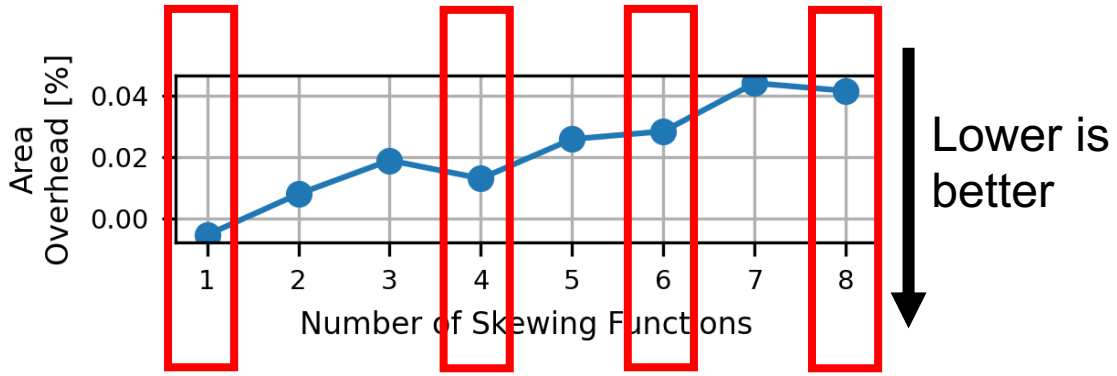
Results – Cache Architectures

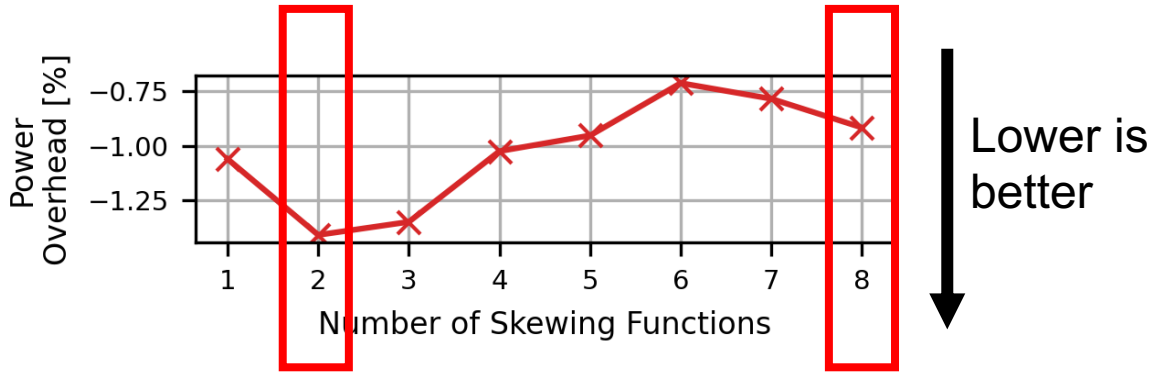
Architecture	Area Overhead [%]	Power Overhead [%]
Skewed Associative	0.008	-1.411
ScatterCache (QARMA64[1])	0.617	23.236
ScatterCache (PRINCE[2])	0.565	28.783

[1] Avanzi, “The QARMA Block Cipher Family,” *ToSC*, 2017.

[2] Borghoff et al, “PRINCE – A Low-Latency Block Cipher for Pervasive Computing Applications,” *ASIACRYPT*, 2012.

Results – Skews





Future Work & Conclusion

- Barriers for secure cache architectures
  - Need to justify tradeoffs of security guarantees
  - Few tradeoffs characterizations for secure caches
- Implemented two cache architectures
  - Skewed Associative
  - ScatterCache

Open-Source Cache Architectures in Chipyard

Few cache architectures have been open-sourced and evaluated for area and power. As a first step towards bridging this gap, we adapted **Rocket Chip's Inclusive Cache** to implement the **Skewed Associative** and **ScatterCache**. We derive area and power results using the SkyWater 130nm and OpernROAD synthesis tool.

Thank you!

Questions?

phaedra.curlin@colorado.edu