TECHNISCHE UNIVERSITÄT DRESDEN

Faculty of Electrical and Computer Engineering

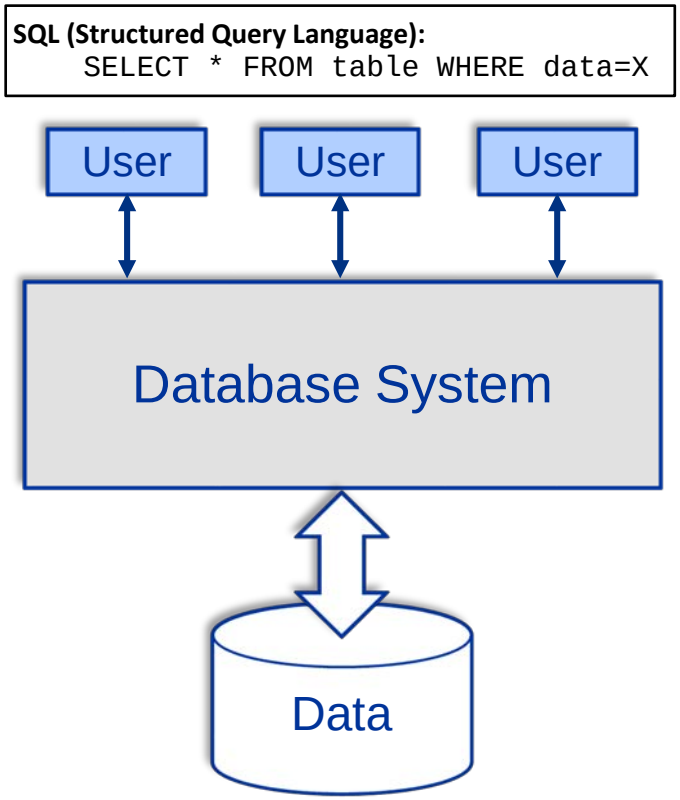
Energy-Efficient Hash Join Implementations in Hardware-Accelerated MPSoCs

8th International Workshop on Accelerating Analytics and Data Management Systems Using Modern Processor and Storage Architectures (ADMS 2017)

Session 2

Sebastian Haas
Gerhard Fettweis

Munich, September 1, 2017



Query Processing

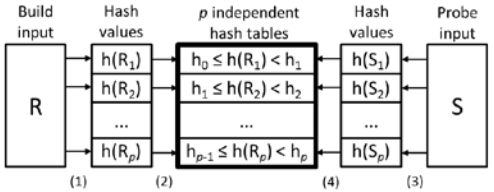
- Demand for
  - Low query latency (fast answer)
  - High throughput
  - Big data
  - Advanced analytics
- Growing demand, data, work, and complexity
- No sufficient improvement in energy efficiency of general-purpose processors

➔ **Custom-made processor**

Paper Contribution

Description of hash join algorithms, focus on

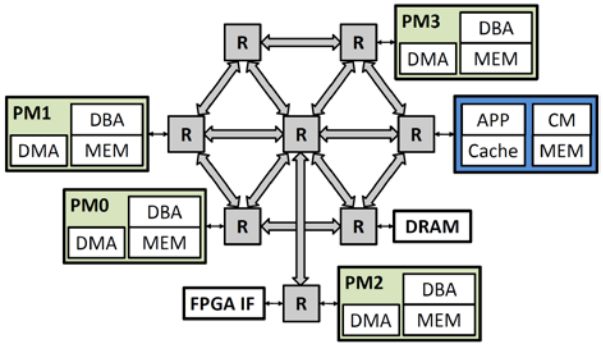
- Reuse of state-of-the-art solutions
- Input data partitioning
- Adapted hash table design



Hash Join

Adaption of join algorithms regarding

- MPSoC architecture (Tomahawk4 chip)
- HW-accelerated processing elements



Tomahawk4 MPSoC

Performance and power measurements

- Different number of cores
- Impact on hashing instructions

Background

TECHNISCHE UNIVERSITÄT DRESDEN

Starting point

- Relational database
- SQL join operator (2-way equi-join)

```
SELECT *
FROM R, S
WHERE R.A = S.B
```

Implementations approaches

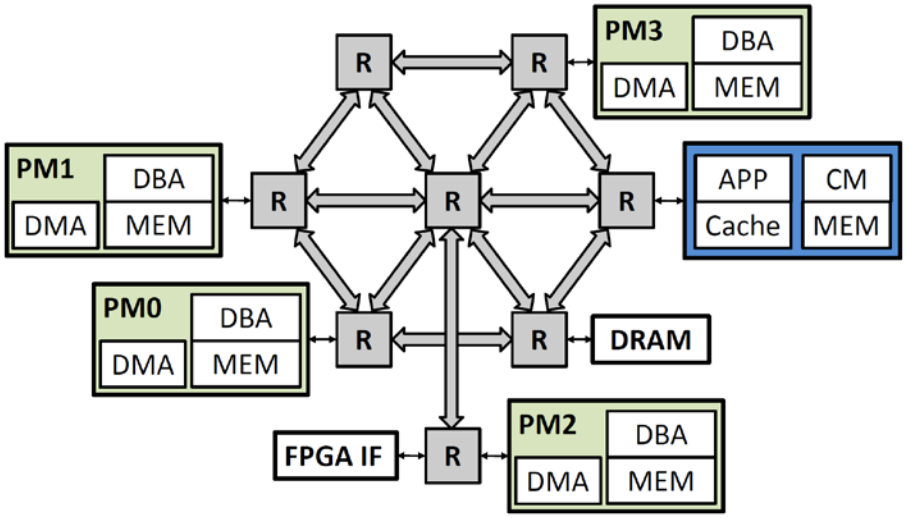
- Nested-loop join
  - Trivial partitioning and parallelization
  - Time complexity: $O(|R||S|)$
- Sort-merge join
  - Benefits from efficient sorting algorithm
  - Single run over both sorted relations
- Hash join
  - Operates on unsorted data
  - Memory bound

MPSoC Platform

Tomahawk4

- Heterogeneous multi-core platform combining SDR capabilities and database processing
- Hexagonal network-on-chip
- 4x Processing Module (PM)
  - Tensilica LX5 with hashing-specific ISA

    → Database Accelerator (DBA)
  - 128 kB local SRAM
- Application Control Module
  - Tensilica 570T (App-Core), 16 kB DCache, 16 kB ICache
  - Tensilica LX5 (CoreManager), 96 kB SRAM
- 128 MB DRAM (LPDDR2), FPGA interface to host-PC

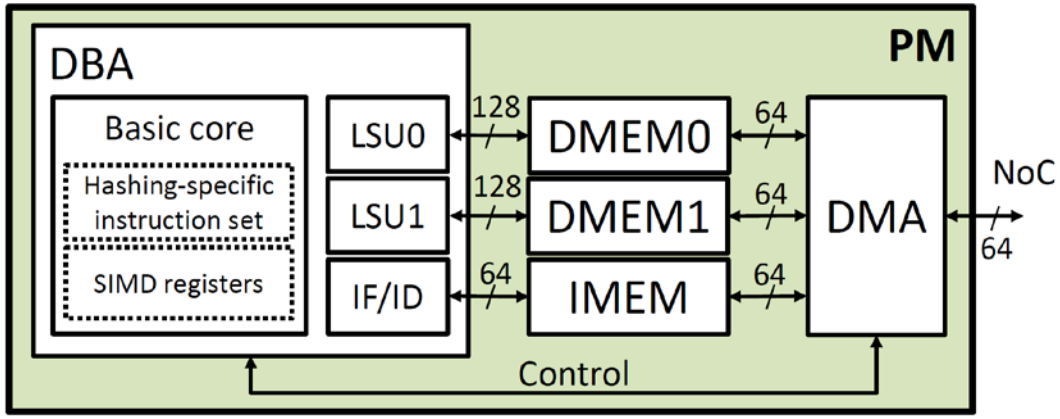


Simplified Tomahawk4 MPSoC platform [1]

[1] S. Haas, T. Seifert, B. Nöthen, S. Scholze, S. Höppner, et al. *A Heterogeneous SDR MPSoC in 28nm CMOS for Low-Latency Wireless Applications*. In Proceedings of the 54th Annual Design Automation Conference (DAC'17), pages 47:1-47:6, 2017.

Processing Module

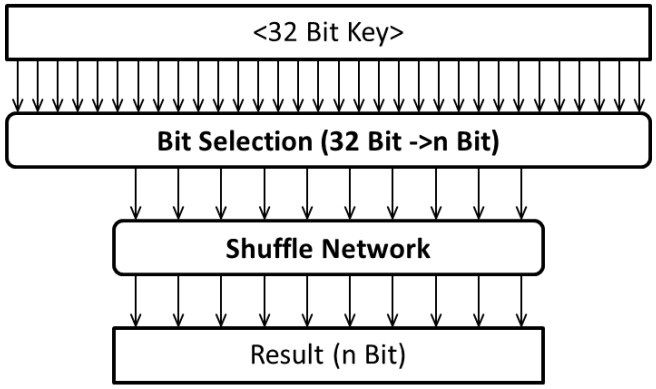
- Tensilica LX5 extended to support basic hashing operations (ASIP)
  - Hashing-specific ISA
  - 128-bit SIMD registers
  - 2x 128-bit data memory interface
  - 64-bit VLIW
- 64kB data memory
- 64kB instruction memory
- DMA controller
- Power management: DVFS, AVS
- Network-on-chip interface



Hash Join

Specification

- Equi-join on two relations *R* and *S* with $|R| \leq |S|$
- 64-bit tuples (32-bit key, 32-bit payload) unsorted
- *R*, *S* and hash table in DRAM
- Hash function: integer bit selection with bit mask



Integer hash function [2]

Implementation challenges

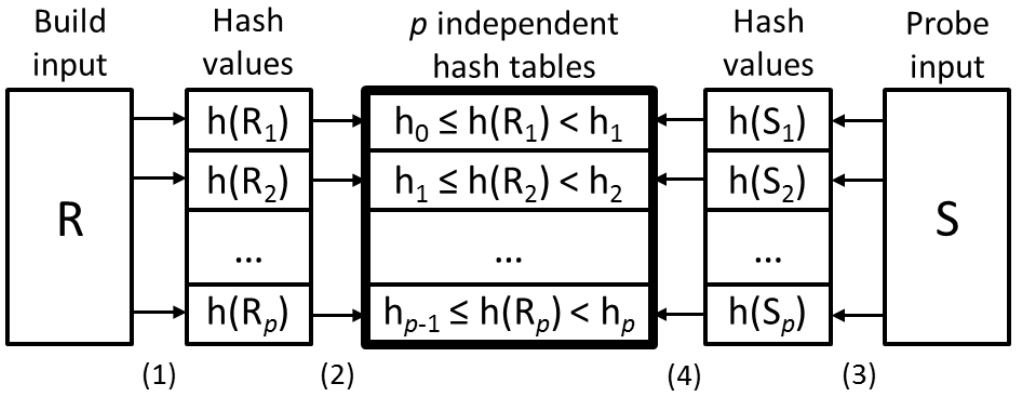
- High DRAM access latency compared to local SRAM
- DRAM only accessible via DMA transfers
- No prior knowledge of input data

[2] O. Arnold, S. Haas, G. Fettweis, B. Schlegel, T. Kissinger, T. Karnagel, and W. Lehner. *HASHI: An Application-Specific Instruction Set Extension for Hashing*. In ADMS, pages 25-33, 2014.

Hash Join (1)

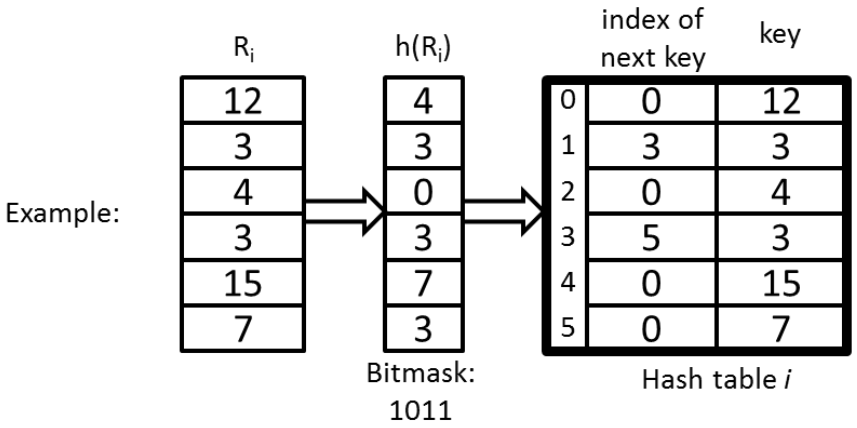
Hash table design

- Partitioning based on hash values:
  → *p* processors
  → *p* independent hash tables
- Buckets connected by linked lists, each bucket stores
  - Key
  - Pointer to next entry of this bucket



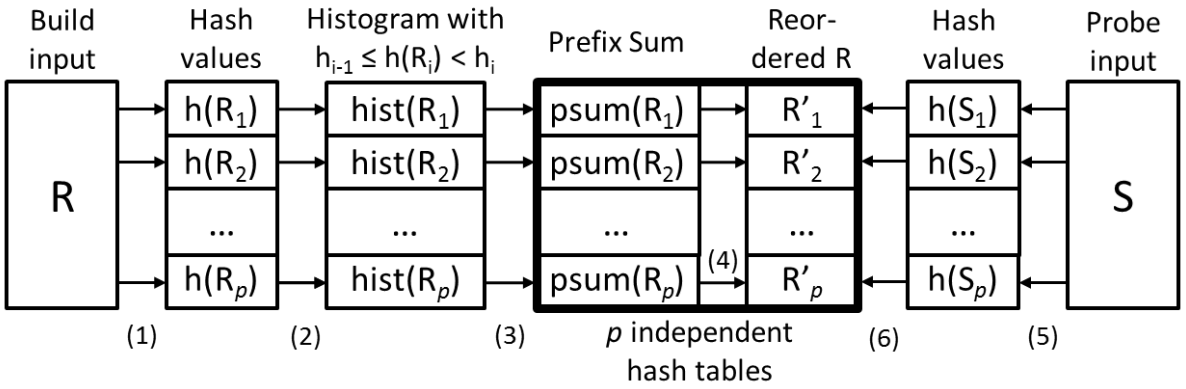
Algorithm

1) Hash *R*
2) Insert *R* to hash table
3) Hash *S*
4) Join



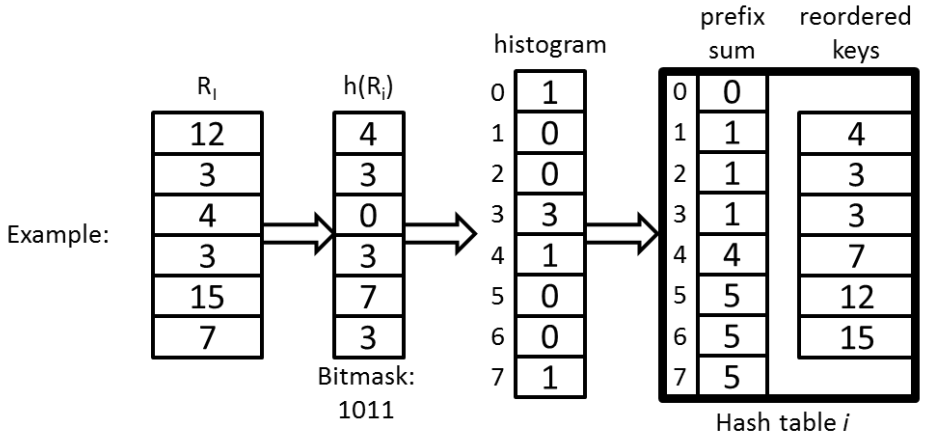
Hash table design

- Partitioning based on hash values
- Buckets identified by histogram and prefix sum of keys in *R*



Algorithm

1) Hash *R*
2) Determine histogram from *R*
3) Determine prefix sum from *R*
4) Insert *R* to hash table
5) Hash *S*
6) Join



Experiments: 4 DBAs

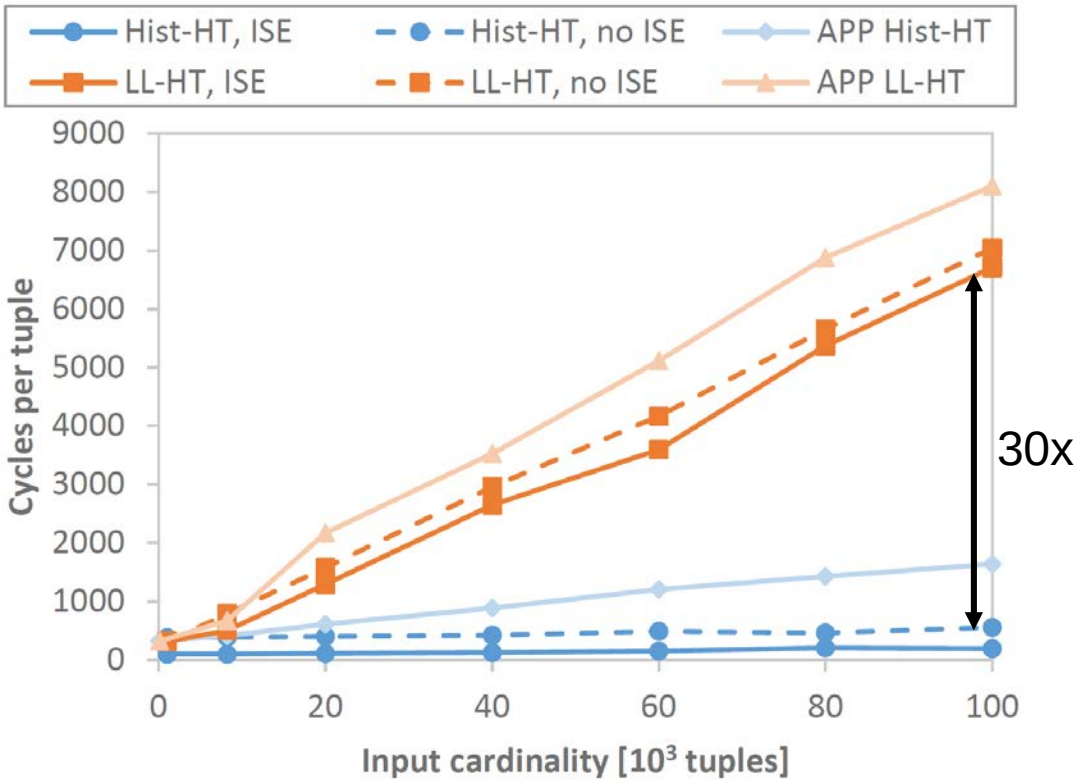
TECHNISCHE UNIVERSITÄT DRESDEN

Configuration

- 4 DBAs execute actual algorithm
- CM initializes data and starts cores
- APP used for comparison

Results

- Histo-HT join 30x better than LinkedList-HT join
- Both joins up to 3.2x faster when using instruction set extensions
- APP up to 5x slower than the 4 DBAs



Experiments: Impact on multiple cores

TECHNISCHE UNIVERSITÄT DRESDEN

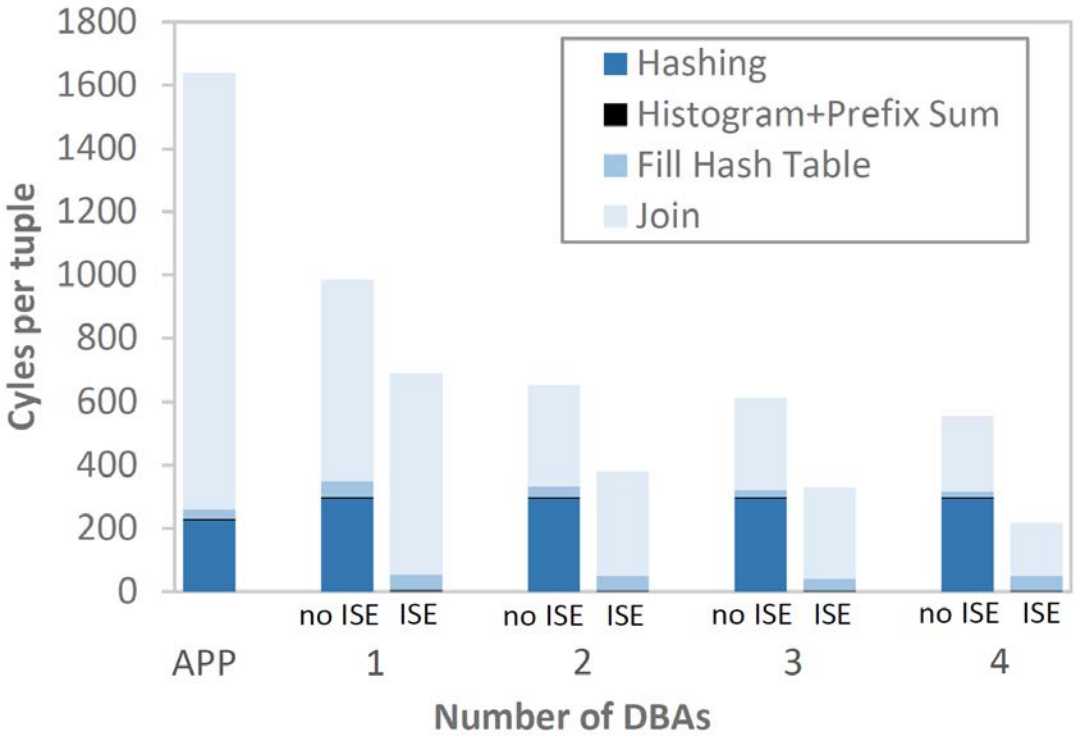
Configuration

- Varying number of DBAs
- Build phase
  - Histogram + prefix sum
  - Fill hash table
- Probe phase
  - Join
- Hashing step includes hashing of *R* and *S*

Results

- Cycle count for hashing stays constant for different number of cores
- Join takes at least 70% of total execution time

Hash join with histogram-based hash table
$|R| = |S| = 100{,}000$ tuples

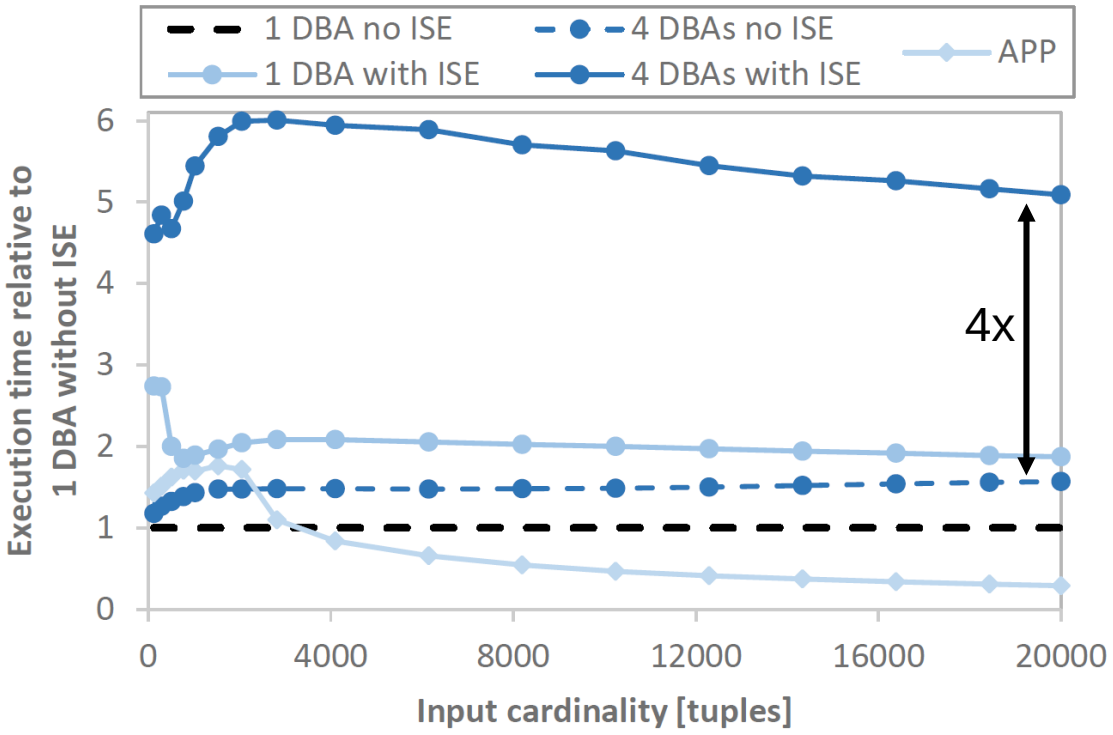


Experiments: Impact on ISE

Results

- Performance advantage of ISE decreases with higher cardinalities
- In comparison to one core, four cores increase the speedup by almost factor 4
- APP increases speedup until cache size of 16 kB is reached

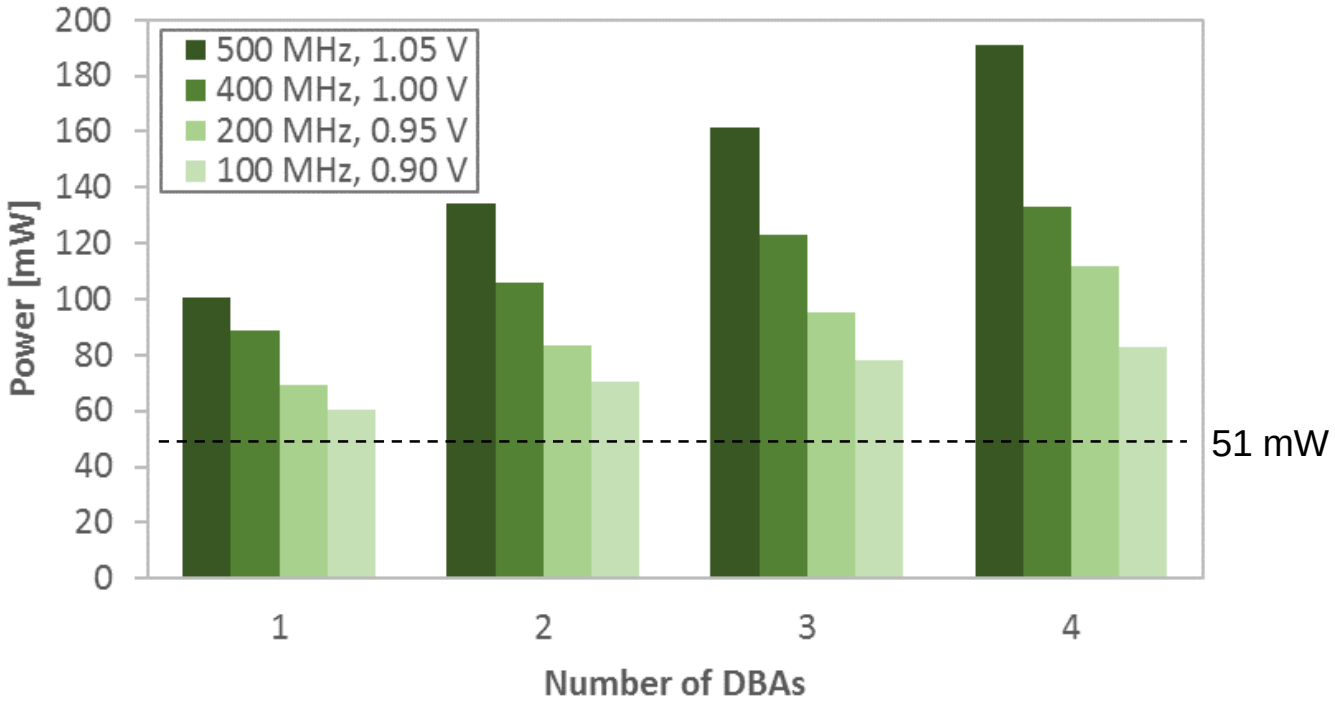
Speedup compared to 1 DBA for histogram-based hash join



Experiments: Power Consumption

Measured power consumption of histogram-based hash join

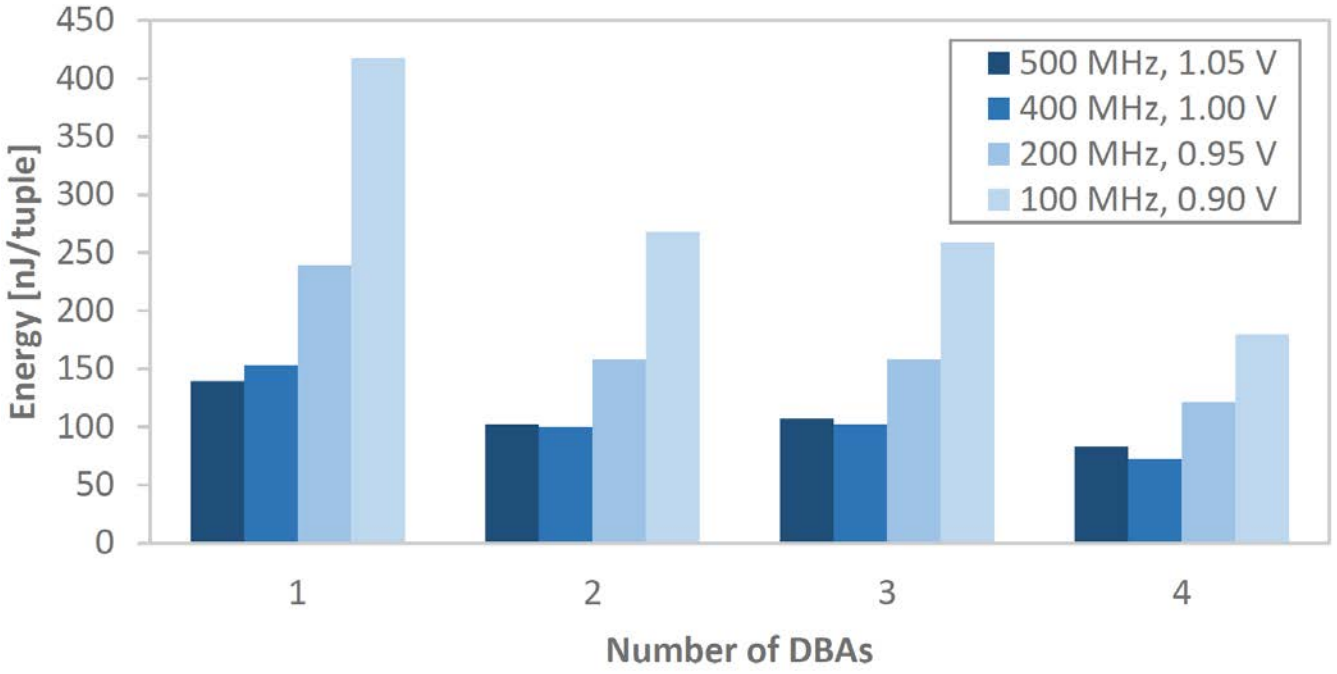
- Vary clock frequencies and supply voltages of processing modules
- CM and NoC run at 500 MHz, 1.05 V and consume about 51 mW



Experiments: Energy Consumption

Energy consumption per tuple of histogram-based hash join

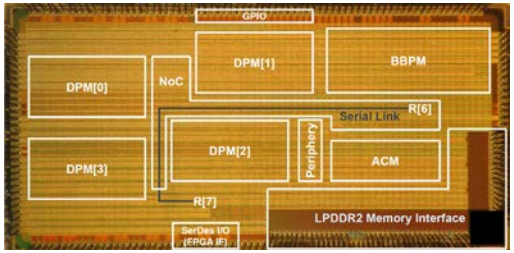
- Vary clock frequencies and supply voltages of processing modules
- CM and NoC run at 500 MHz, 1.05 V with 51 mW



Conclusion

Adapt state-of-the-art hash join algorithms to modern MPSoC platform

- Partitioning of input relations similar to radix clustering
- Two hash table designs with buckets connected by linked lists as well as histogram and prefix sum
- Tomahawk4 with included hashing-specific instruction set processors



Tomahawk4 MPSoC die photo [1]

Combination of instruction set extensions and multi-core performance

- Speedup factor 4x compared to unmodified implementation
- Adapted hash table design with up to 30x throughput improvements

Results provide basis to establish low-power MPSoC in database systems

- Accelerated coprocessor next to high-performance CPU to offload energy-efficient processing of basic database operations

[1] S. Haas, T. Seifert, B. Nöthen, S. Scholze, S. Höppner, et al. *A Heterogeneous SDR MPSoC in 28nm CMOS for Low-Latency Wireless Applications*. In Proceedings of the 54th Annual Design Automation Conference (DAC'17), pages 47:1-47:6, 2017.

Acknowledgement

This work has been supported in part by

- The state of Saxony under grant of the German Research Foundation (DFG) within the Cluster of Excellence "Center for Advancing Electronics Dresden" (cfaed), and SFB912 – HAEC
- The European Union's Horizon 2020 research and innovation program under grant agreement No. 671566 "Superfluidity"
- The ECSEL Joint Undertaking under grant agreement No. 692519 "PRIME"

We also thank the Chair for Highly-Parallel VLSI-Systems and Neuro-Microelectronics of Technische Universität Dresden for backend design and PCB development of the Tomahawk4 chip.