An MPSoC for Energy-Efficient Database Query Processing

TensilicaDay 2016

Sebastian Haas
Emil Matúš
Gerhard Fettweis

09.02.2016

Introduction

Challenges

- 5G Mobile Networks
- Mobile Edge Computing
- Tactile Internet
- Internet of Things

Database System

Requirements

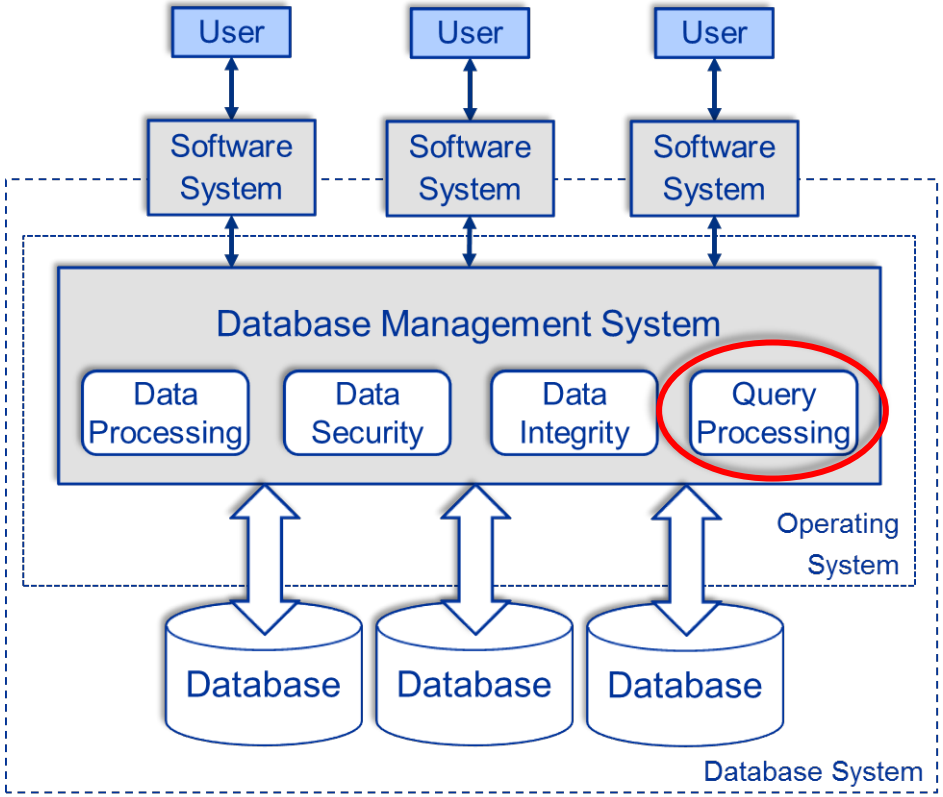
- Throughput
- Latency
- Energy

MPSoC

- Application-Specific
- Customizable
- Energy-Efficient

Database Systems

```
SQL (Structured Query Language):
BEGIN
    FOR X IN 0 .. 255 Loop
        SELECT * WHERE data=X
    END Loop
END;
```



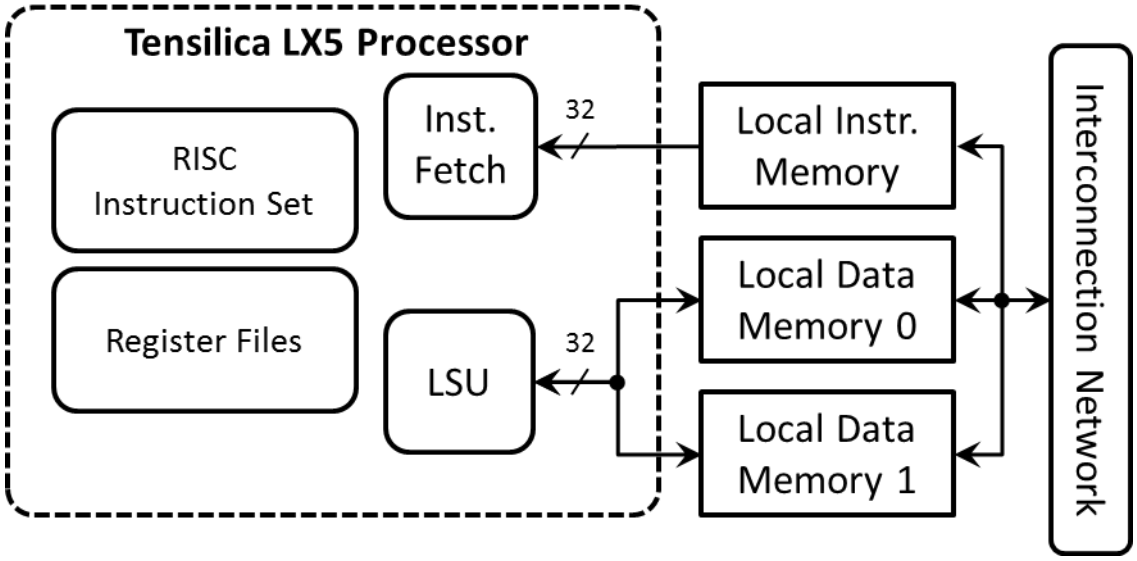
Query Processing

- Big Data
  → **Query Throughput**
- Direct interconnection to user and storage
  → **Query Latency**

→ Database Accelerator (DBA)

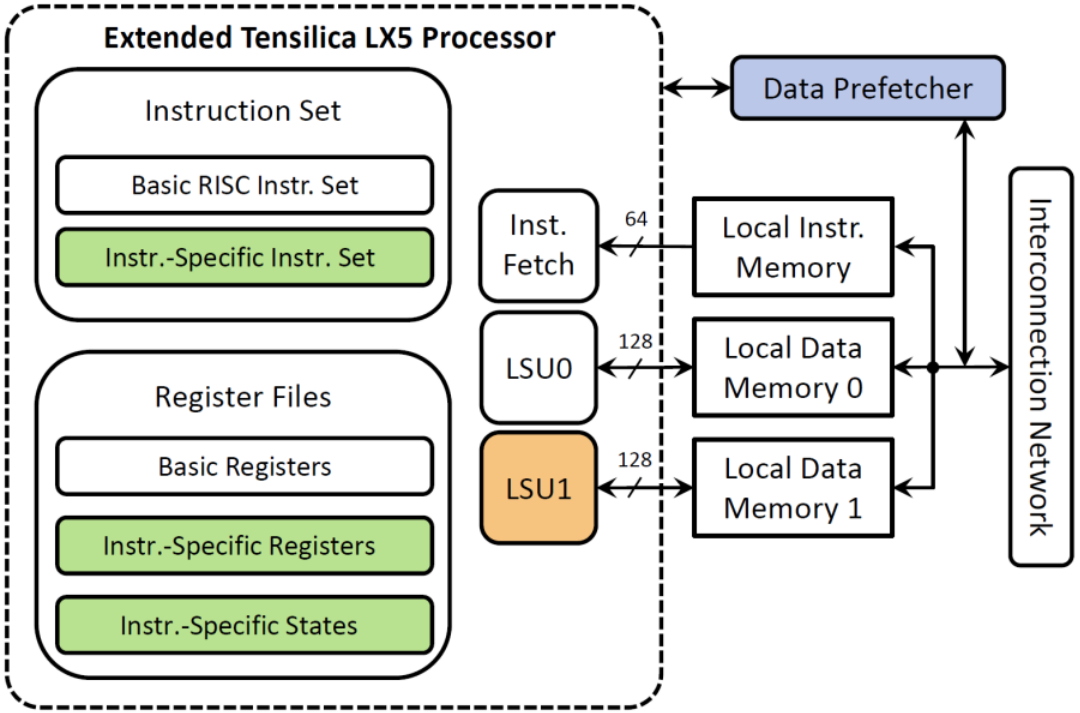
Base Processor

- Tensilica Xtensa LX5 RISC Processor
- Local RAM
  - 2x 32kB data
  - 1x 32kB instruction
- XLMI
  - 1 Load-Store unit (LSU)
  - 32 bit data
  - 32 bit instruction



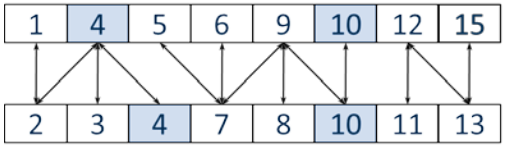
Extended Processor

- Tensilica Xtensa LX5 RISC Processor
- Database-Specific Instruction Set
- Local RAM
  - 2x 32kB data
  - 1x 32kB instruction
- XLMI
  - 2 Load-Store units (LSU)
  - 2x 128 bit data
  - 64 bit instruction
- 64 bit FLIX
- Data Prefetcher

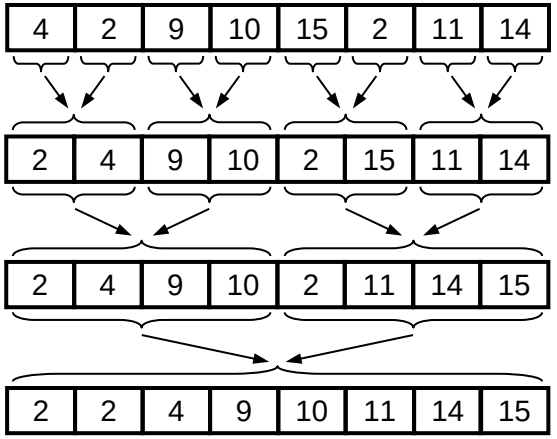


DBA Applications

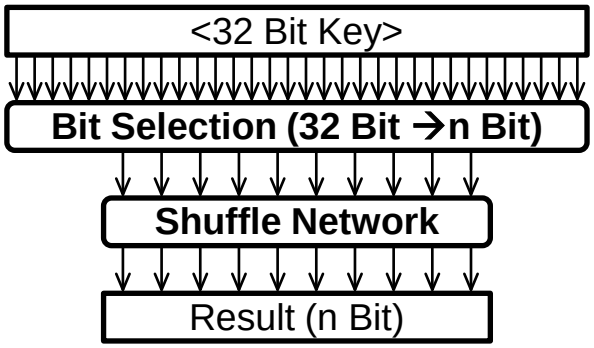
Intersection



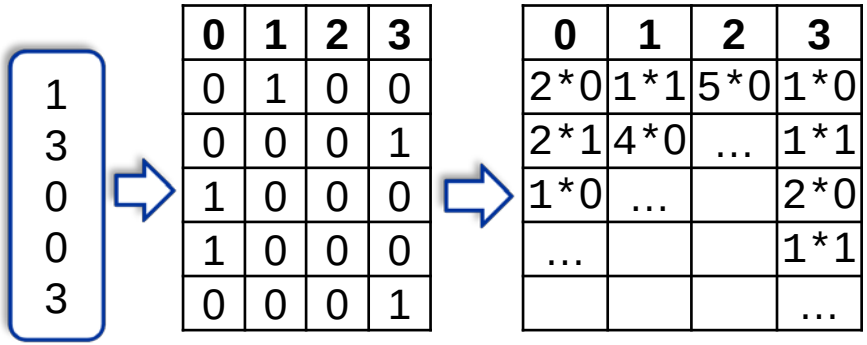
Merge Sort



Hashing



Bitmap Index Compression



Hashing: TIE Development

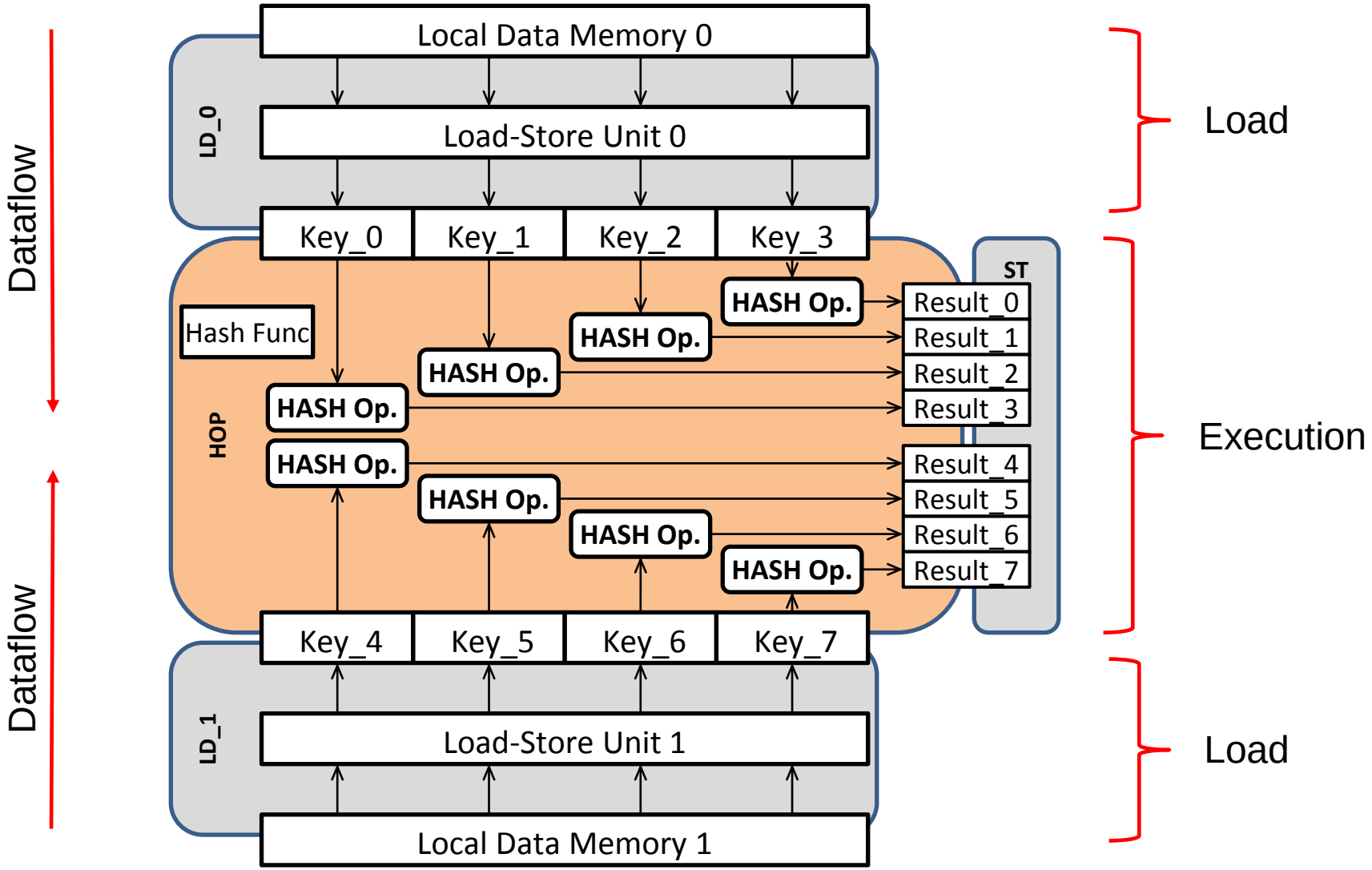

```
unsigned int hash, shVal, shVal_neg;
unsigned int mask = 0xFFFFFFFF;

for(i=0; i<keySize; i++){
    //load key, bit selection
    hash = key[i] & hashFunc;

    //extract bits
    for(j=30; j>=0; j--){
        if(!(hashFunc & (0x1<<j))){
            //partial shift right
            shVal = hash & (mask<<j);
            shVal_neg = hash & ~(mask<<j);
            hash = (shVal>>1) | shVal_neg;
        }
    }
    //store hash value
    hashValue[i] = hash;
}
```

Pure C code

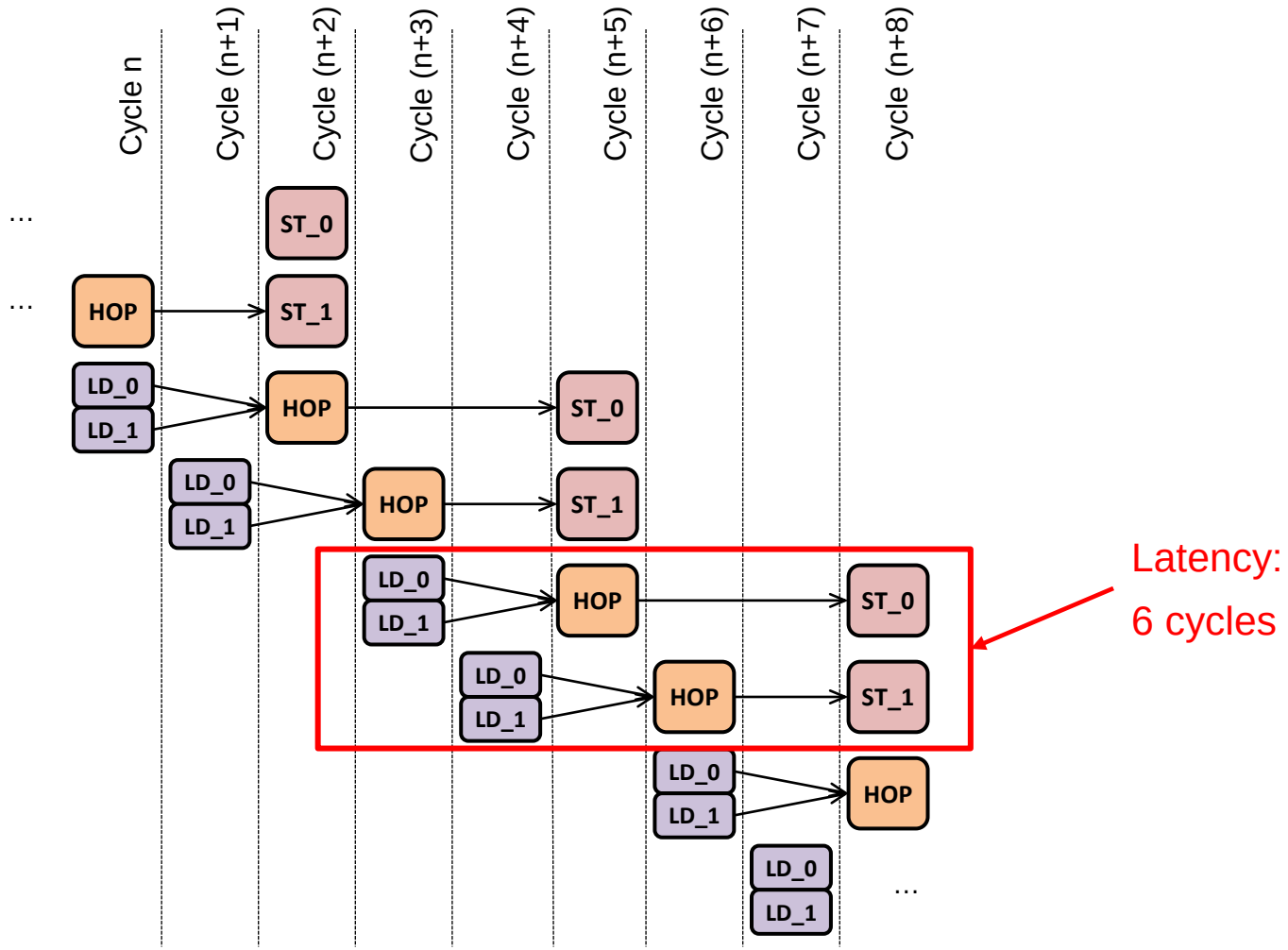
```
//init pointer, variables
init_states(key, hashValue, hashFunc);

LD_0(); LD_1();

//load keys, extract bits, store hash values
for(i=0; i<(keySize/16); i++){
    LD_0(); LD_1(); HOP();          1 cycle
    LD_0(); LD_1();                 1 cycle
    HOP(); ST_0(); ST_1();          1 cycle
}

HOP();
ST_0(); ST_1();
```

C code with TIE instructions

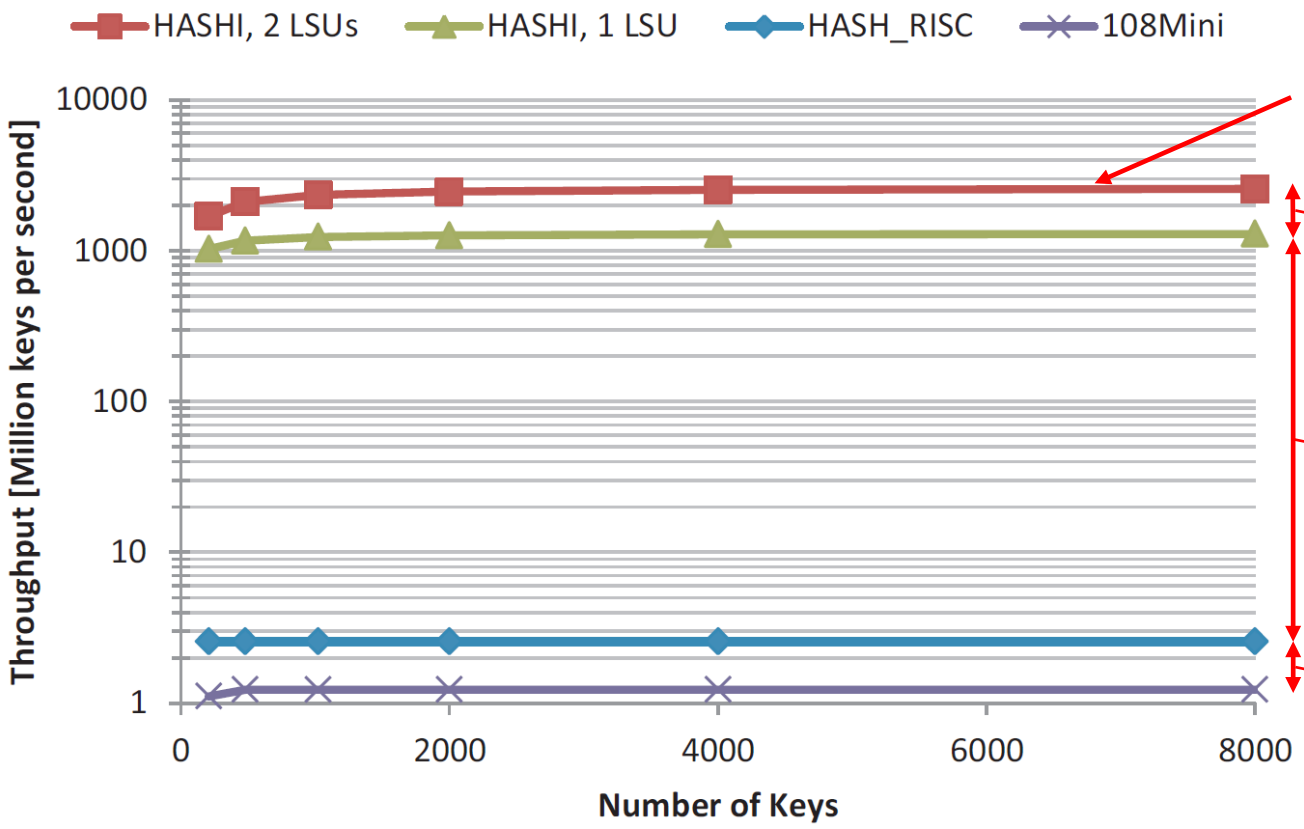
Hashing: Instruction Flow



Hashing: Pipeline Snippet



Hashing: Results



Final processor

+1 Load-Store unit (2x)

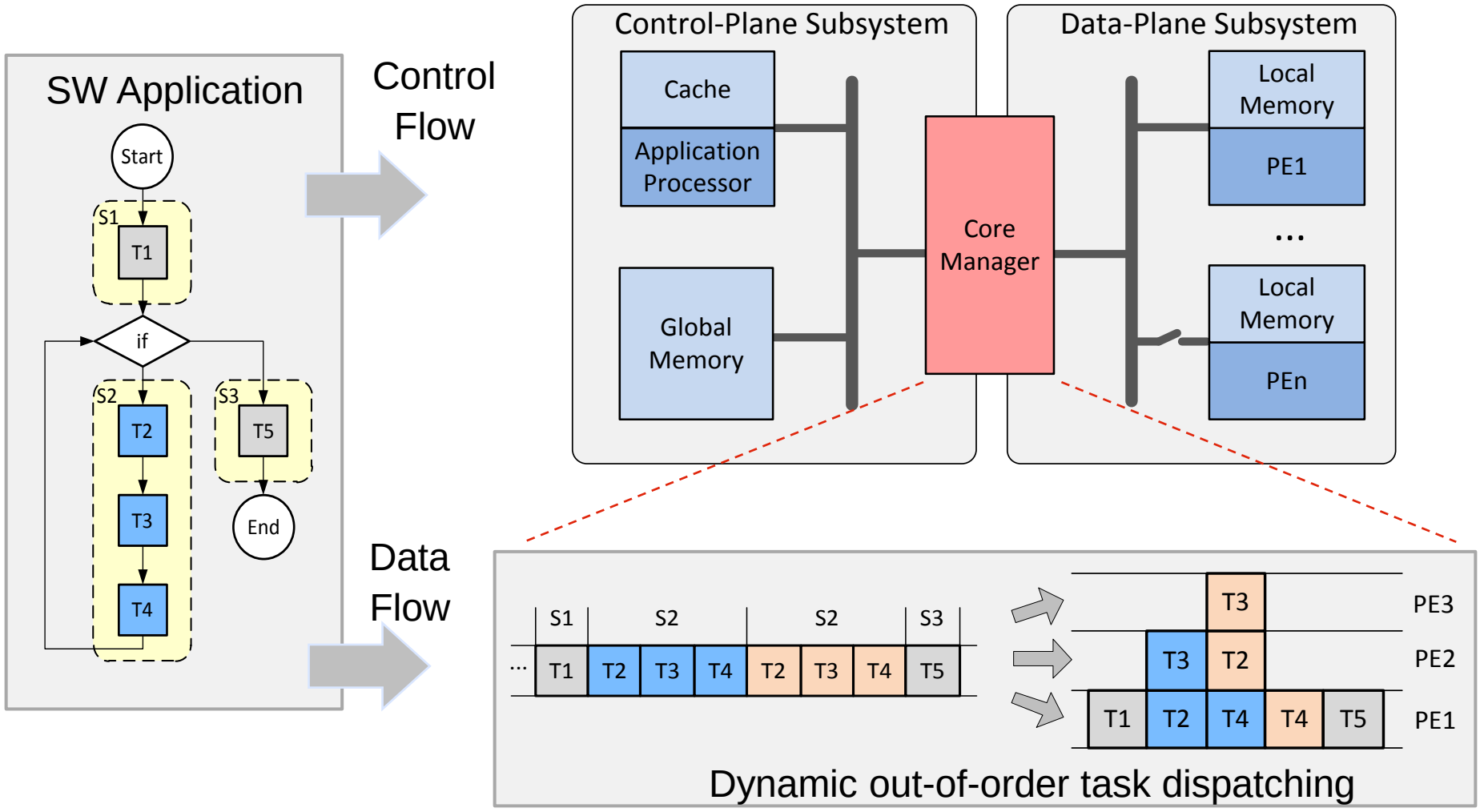
+ TIE instructions (500x)

Data bus: 32 →128 bit (2x)

Throughput $T = \frac{n_{key}}{t}$

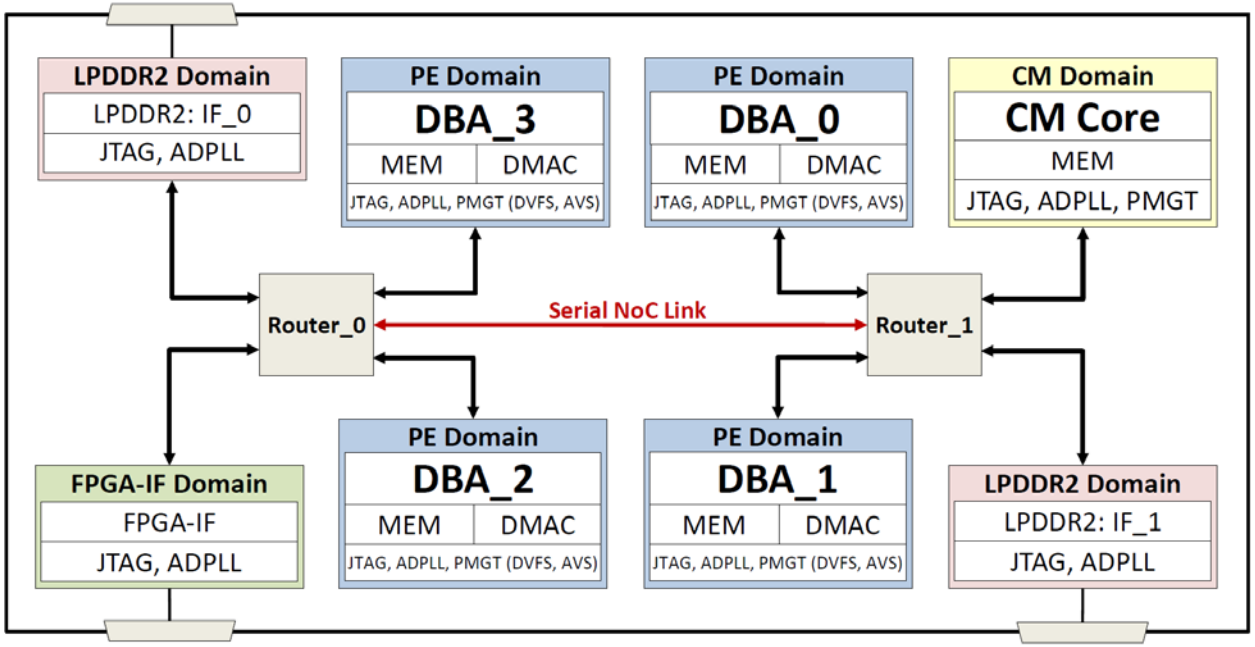
n_{key}: number of keys
t: time to perform the operation

Tomahawk Concept



Tomahawk3 MPSoC

- 28 nm CMOS SLP Globalfoundries
- Die Size: 18 mm²
- Tape-Out: Oct. 2014
- 5 Core heterogeneous MPSoC (2 core types)
- Network-on-Chip: High-Speed Serial Data Link with 2 Routers
- Power Management: DVFS, AVS
- 2x LPDDR2 Memory Interface: 2x 64 MB SDRAM



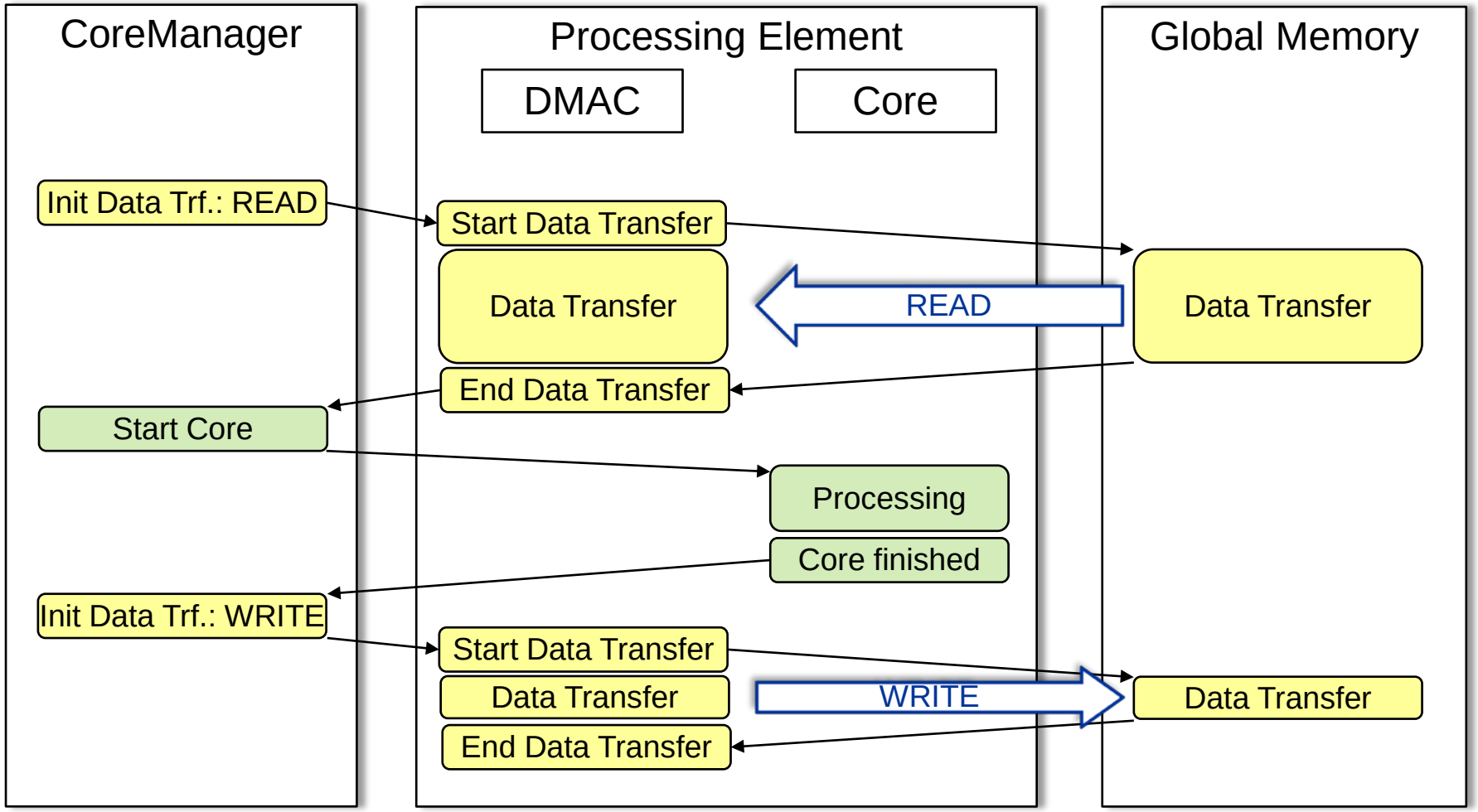
- Processing Elements:
  - Tensilica Xtensa LX5
  - Extended Instruction Set for Database Applications
- CoreManager:
  - Optimized for query processing
  - Tensilica Xtensa LX5

Single Core Performance

Algorithm	Sorted-Set Operations		Sorting		Hashing		Searching				Bitmap Operations		
Operator	Intersection		Merge Sort		Bit Extraction	Sampling	Equality Search				WAH Compression	WAH Decompression	WAH Logical AND Operation
Data width [bit]	32	16	32	16	32	32	4	8	16	32	32	32	32
Area Percentage[1] [%]	4.9	9.0	12.4	23.3	5.6	10.7	0.2	0.3	0.4	0.5	2.8	2.9	3.4
RISC[2]:													
Throughput [Gbit/s]	0.058	0.063	0.004	0.004	0.083	0.064	0.43	0.65	1.86	3.72	0.25	0.5	1.61
Power [mW]	43.3	52.5	58.2	60.3	40.6	53.0	52.5	52.7	56.5	54.4	50.8	47.8	60.4
Power/Throughput [nJ/bit]	742.5	834.2	14934.6	16752.1	487.5	828.4	122.4	81.3	30.1	14.6	200.9	96.6	37.6
DBA-ASIP[3]:													
Throughput [Gbit/s]	1.49	3.17	0.032	0.063	82.13	82.41	63.5	63.6	63.6	63.6	10.3	28.1	48.4
Power [mW]	69.8	78.6	60.9	64.3	73.4	63.5	63.5	61.2	62.1	62.3	55.8	50.8	66.7
Power/Throughput [nJ/bit]	46.9	24.8	1892.5	1022.2	0.9	0.8	1.0	0.96	0.98	0.98	5.4	1.8	1.4
Energy Gain RISC/DBA-ASIP	16x	34x	8x	16x	542x	1036x	122x	85x	31x	15x	37x	54x	27x

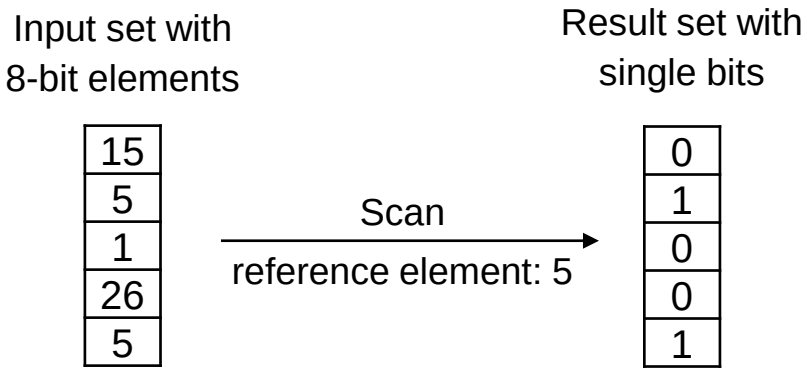
[1] Relative Area Consumption of the functional database units regarding to the complete DBA processor
[2] Measured on the DBA cores of the Tomahawk3 without database-specific instruction set with f_{Max} = 500 MHz @ V_{DD}=1.1V
[3] Measured on the DBA cores with f_{Max} = 500 MHz @ V_{DD}=1.1V

System Level



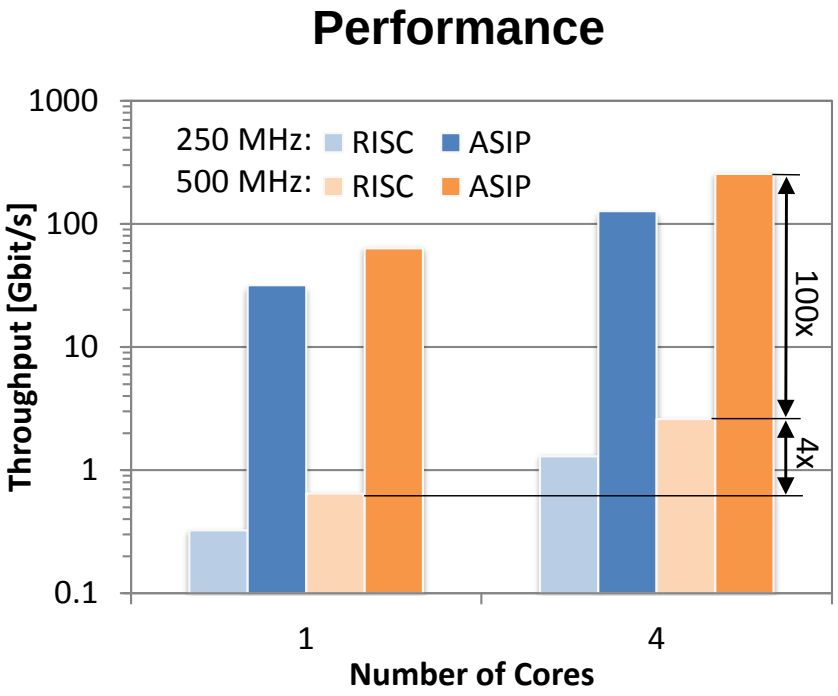
Scan Benchmark

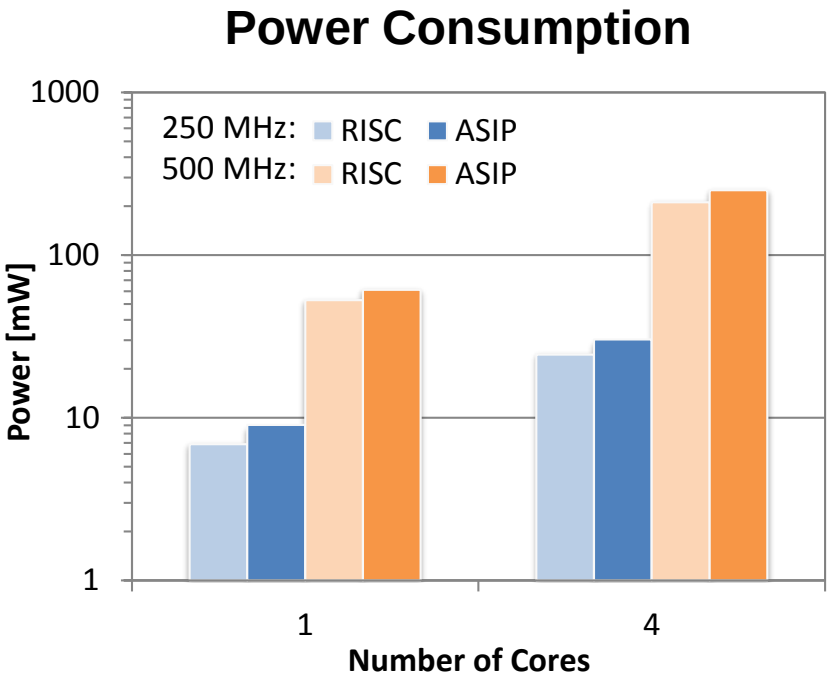
- Scan operation “scans” data set with respect to a reference element (Filtering, Equality Search)
- Result of comparison is one bit
- TIE Instructions available for 4, 8, 16, and 32-bit input values
- Example:



- Advantages
  - High instruction level parallelism
  - High data level parallelism

Scan Benchmark: Results





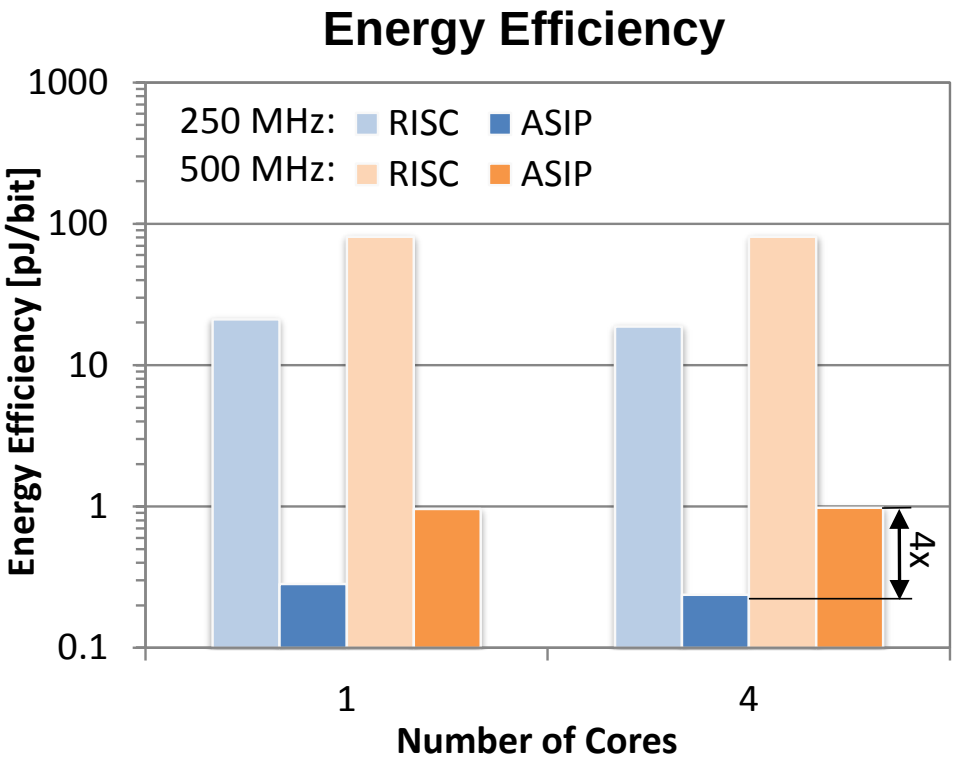
Speedup due to:

- Core Extensions/TIE: 100x
- 1 core → 4 cores: 4x

Power increase due to:

- Core Extensions/TIE: 10mW
- 1 core → 4 cores: 190mW

Scan Benchmark: Results



Energy decrease due to:

- Core Extensions/TIE: 100x
- 1 core → 4 cores: negligible
- 500MHz → 250 MHz 4x

Benchmark Comparisons

TECHNISCHE UNIVERSITÄT DRESDEN

vodafone chair

App. Scenario	Scan					WAH Indexing			
	Tomahawk3		2x Intel Xeon E5-2690 [1]		2x Intel Xeon E5430 [2]	**Tomahawk3**		Intel i7-2600K [3]	NVIDIA GTX-670 [3]
Processing Cores	**4**		16		8	**4**		4	1344
Clock Freq. [GHz]	**0.5**		2.0		2.66	**0.5**		3.4	0.98
Data Deposit	**Loc. Mem**	**DRAM**	Cache	DRAM	DRAM	**Loc. Mem**	**DRAM**	DRAM	DRAM
Total Throughput [Gbit/s]	**254.4**	**25.0**	1281.4	537.0	4.47	**2.26**	**2.26**	0.3	1.26
Total Power [W]	**0.25**	**0.735**	123.2	159.9	190	**0.238**	**0.753**	-	-
DRAM Power [W]	**0.005**	**0.282**	3.151	32.127	40	**0.006**	**0.282**	-	-
Power/Throughput [nJ/bit]	**0.001**	**0.029**	0.096	0.298	42.5	**0.105**	**0.333**	-	-

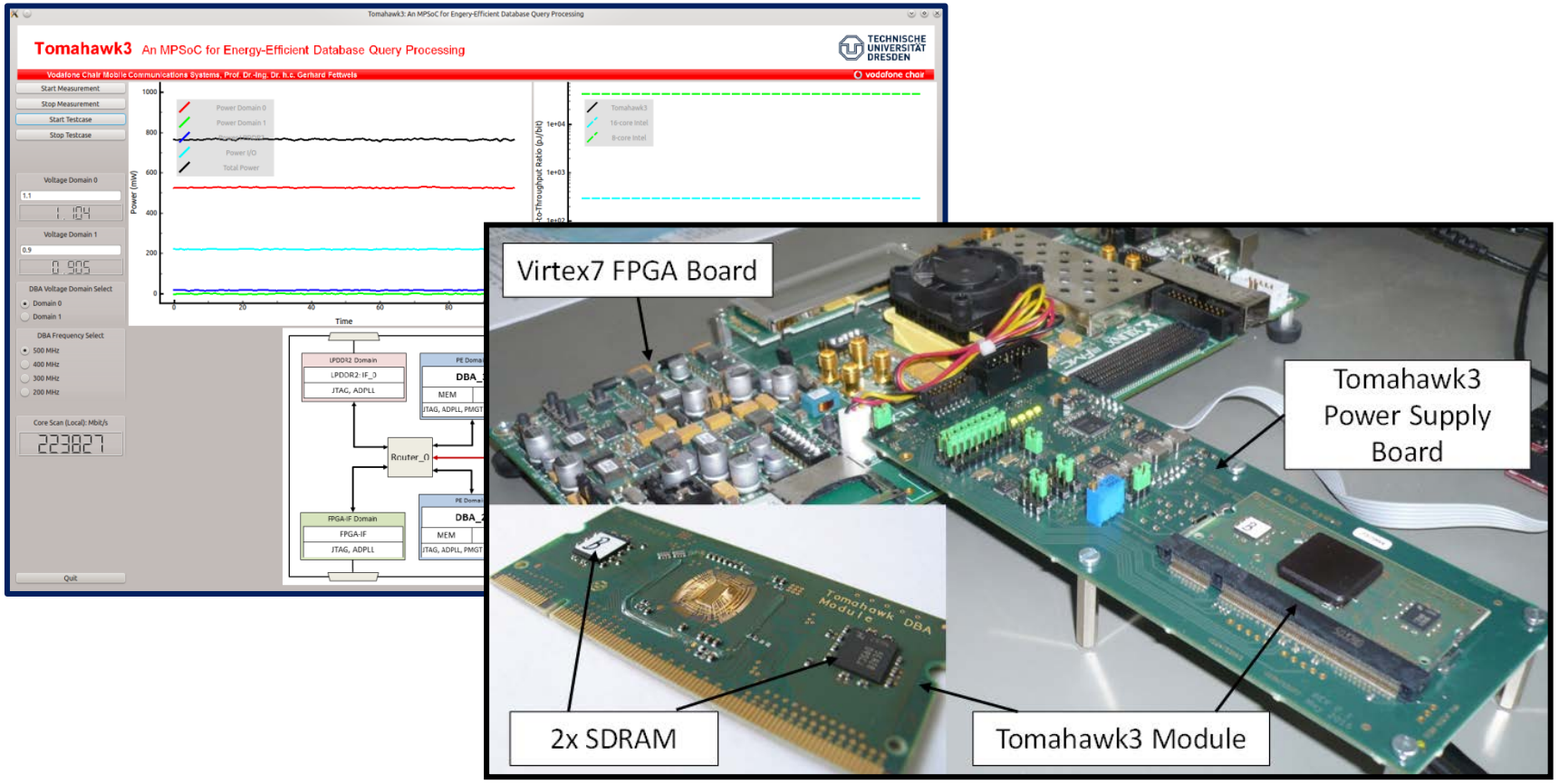
References:

[1] F. Fusco, et al., “Indexing Million of Packets per Second using GPUs,” In Proceedings of the 2013 Conference on Internet Measurement Conference (IMC'13), 2013.

[2] I. Psaroudakis, T. Kissinger, D. Porobic, T. Ilsche, E. Liarou, P. Tözün, A. Ailamaki, and W. Lehner. Dynamic fine-grained scheduling for energy-efficient main-memory queries. In Proceedings of the Tenth International Workshop on Data Management on New Hardware, DaMoN'14, 2014.

[3] D. Tsirogiannis, S. Harizopoulos, and M. A. Shah. Analyzing the energy efficiency of a database server. In Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, SIGMOD'10, 2010.

Tomahawk3 Demonstrator



Acknowledgements: We would like to thank *Cadence* and *Tensilica* for providing software tools an IP as well as the *Chair for Highly-Parallel VLSI-Systems and Neuromorphic Circuits* for Backend-Design and PCB development of the Tomahawk3 chip.

Thank you!

References:

[1] O. Arnold, S. Haas, G. Fettweis, B. Schlegel, T. Kissinger, W. Lehner: *An Application-Specific Instruction Set for Accelerating Set-Oriented Database Primitives*, SIGMOD 2013.

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

[3] B. Nöthen et al.: *A 105GOPS 36mm2 Heterogeneous SDR MPSoC with Energy-Aware Dynamic Scheduling and Iterative Detection-Decoding for 4G in 65nm CMOS*. ISSCC 2014

[4] O. Arnold, E. Matus, B. Nöthen, M. Winter, T. Limberg and G. Fettweis: *Tomahawk - Parallelism and Heterogeneity in Communications Signal Processing MPSoCs*. TECS 2013