A 22nm ASIC for Flexible Post-Quantum Cryptography

Patrick Karl[1], Jonas Schupp[1], Debapriya Basu-Roy[1,3], Maximilian Schöffel[4], Johannes Feldmann[4], Norbert Wehn[4], Georg Sigl[1,2]

[1]Technical University of Munich
[2]Fraunhofer Institute for Applied and Integrated Security
[3]Indian Institute of Technology Kanpur
[4]Technical University of Kaiserslautern

September 18, 2022

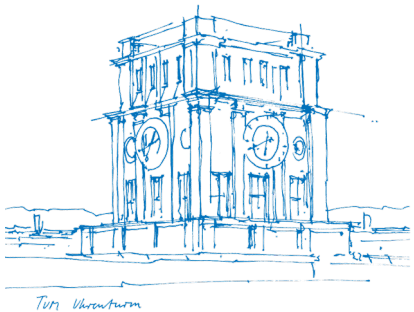


Table of contents

Introduction

- **Motivation**
  - Design PQC HW accelerators and ISEs for efficiency gain on RISC-V platforms
  - Smaller technology nodes (65 nm $\rightarrow$ 22 nm)
  - Bring design to silicon, gain experience

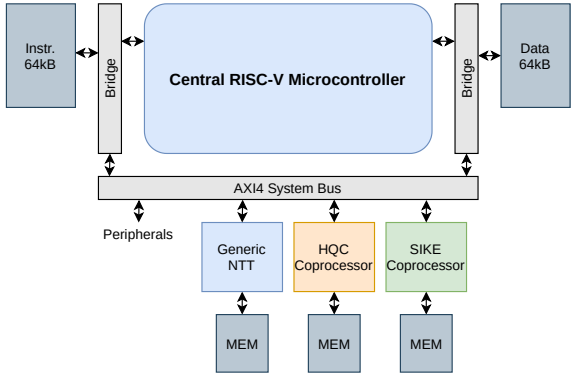
- Target NIST PQC finalists
  - **Lattices:** Generic RISC-V processor [Fri+21]
  - **Codes:** HQC co-processor
  - **Isogenies:** SIKE co-processor

$\rightarrow$ **Main objective:** High performance with flexible PQC support

Architecture

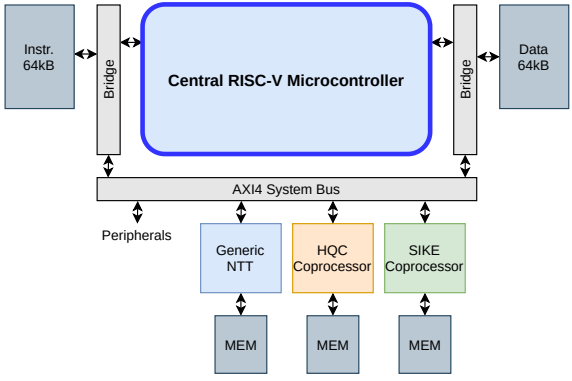
System Overview

- **Central MCU**
  - Basic computing
  - Data transfer and communication
  - *Lattice*-based cryptography
- **Co-processors**
  - Standalone processors
  - Directly accessible from peripherals
  - *Code-* and *isogeny*-based



System Overview

- **Central MCU**
  - Basic computing
  - Data transfer and communication
  - *Lattice*-based cryptography
- **Co-processors**
  - Standalone processors
  - Directly accessible from peripherals
  - *Code-* and *isogeny*-based

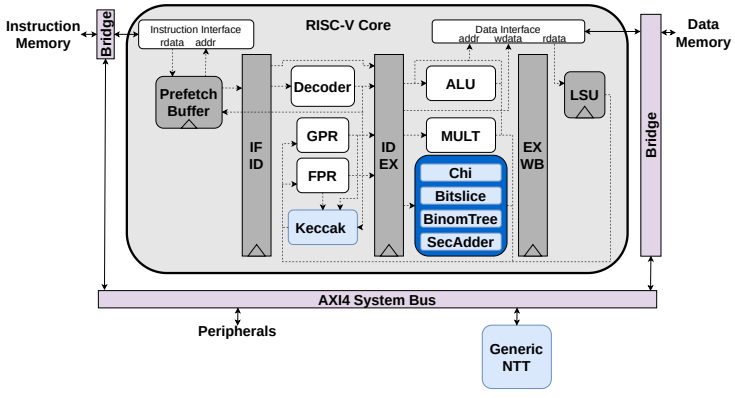


RISC-V core with accelerators for Lattices

- Base design based on [Fri+21] to be presented at CHES2022!
- Provides acceleration for
  - Generic accelerator for Number Theoretic Transform (NTT)
  - Hashing, binomial sampling, A2B/B2A conversions, compression (masking support)
- Targeting Kyber/Saber
  - Strong competitors in NIST competition
  - Comparison of overhead for SCA resilience

RISC-V core with accelerators for Lattices

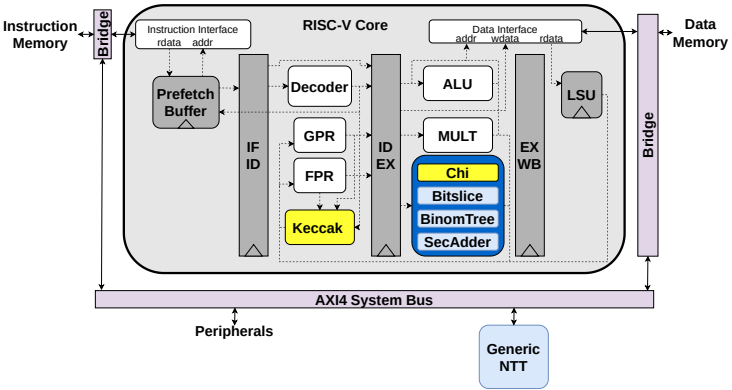
- **PULPino** [Gau+17]
  - cv32e40p: 4-stage pipeline
  - RV32IMFC



RISC-V core with accelerators for Lattices

- **Keccak**
  - Full round unmasked
  - Masked non-linear layer χ

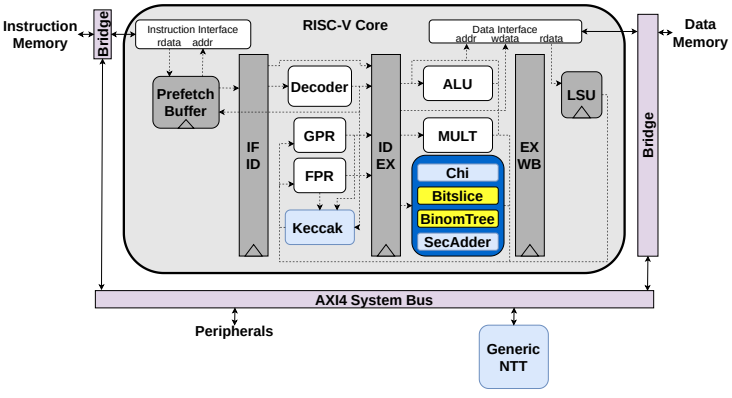
Func7	Func3	Name
0x08	0	`keccak-f1600`
0x16	0	`pq.mchiw`
	1	`pq.mchic`
	2	`pq.mchir`



RISC-V core with accelerators for Lattices

- **Bitsliced Sampler**
  - Adder tree for binomial sampling

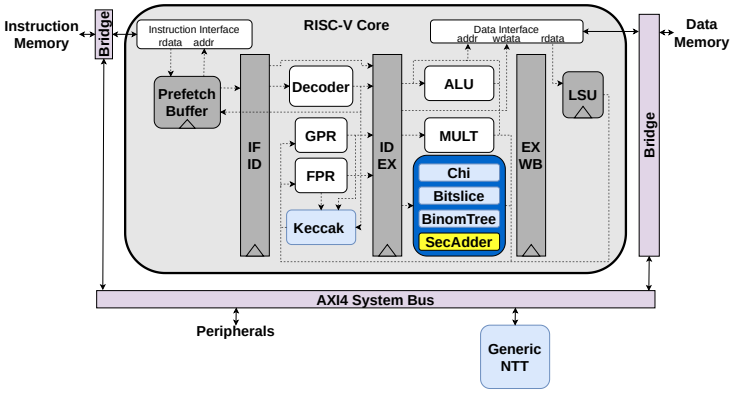
Func7	Func3	Name
0x15	0	pq.slicew
	1	pq.slicer
0x14	0	pq.mbinw
	1	pq.mbinc
	2	pq.mbincinv
	3	pq.mbinr
	4	pq.mbincpy
	5	pq.mbinrst



RISC-V core with accelerators for Lattices

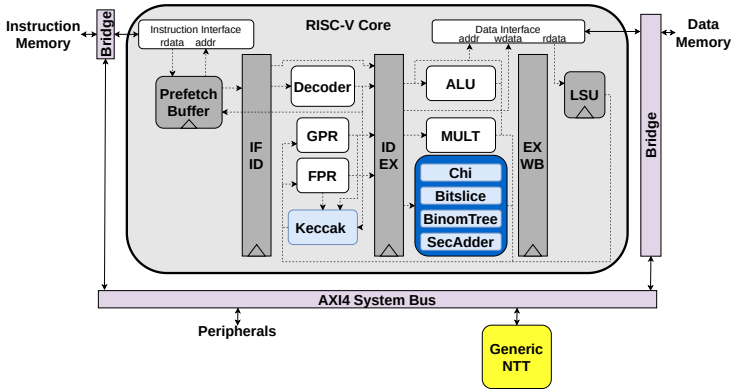
- **SecAdder**
  - Securely adding shares
  - Used for A2B/B2A conversions

Func7	Func3	Name
0x17	0	`pq.maddw`
	1	`pq.maddc`
	2	`pq.maddcc`
	3	`pq.maddr`



RISC-V core with accelerators for Lattices

- **Generic NTT** [Fri+21]
  - NTT-based polynomial arithmetic
  - Wide range of parameters supported
  - Transforms, multiplication, addition etc.



Co-processor: Hamming-Quasi Cyclic (HQC) Key Exchange

- Contributed by Technical University of Kaiserslautern
- Based on syndrome-decoding problem
  - Cyclic codes for smaller keys
  - Concatenation of Reed-Muller Reed-Solomon codes
  - Support for security level 1
- Fully functional crypto peripheral
  - Custom RV32I core
  - ISA Extensions for decoding

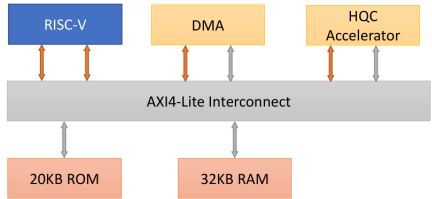


Figure: Co-processor architecture

Co-processor: Hamming-Quasi Cyclic (HQC) Key Exchange

- **Control Unit**
  - Access to memories
  - Manages operation mode, starts submodules
  - Independent from RISC-V (parallel execution)
- **R-Unit**
  - Arithmetic in $F_2[X]/(X^n - 1)$
- **Sampling Unit, RM-Decoder**
  - Keccak-based PRNG unit
  - Decoding of Reed-Muller codeword

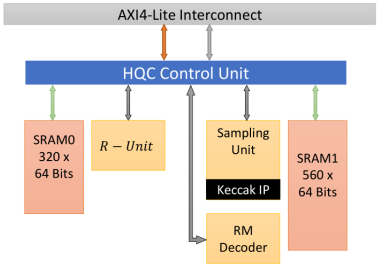


Figure: Accelerator architecture

Co-processor: Supersingular Isogeny Key Exchange (SIKE)

- Based on the computation of isogenies
  - Supports SIKEp434 and SIKEp751
  - Security level 1 and 5
- Main bottlenecks
  - Point trippling
  - Evaluation of degree 3 isogenous curve
- Basic idea similar to [RM19]
  - 2 radix-15 multipliers
  - Each doing 3 multiplications in parallel

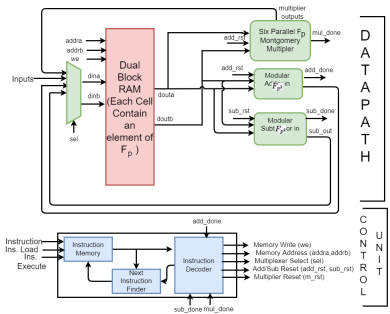
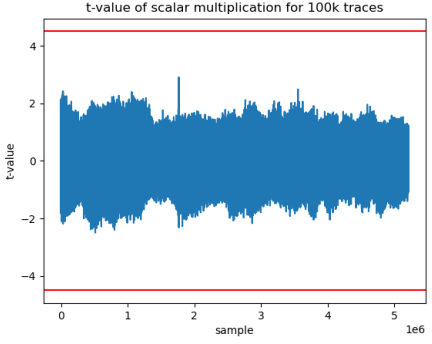


Figure: Co-processor architecture

Co-processor: Supersingular Isogeny Key Exchange (SIKE)

- Countermeasures for scalar multiplication
  - Scalar splitting
  - Point randomization
- Can be used for acceleration of ECC
  - Brainpool, Curve25519, NISTp256 etc.
- Technically isogeny computation also secured



Physical Characteristics

Some Physical Characteristics

- 22 nm FDSOI Globalfoundries via Europractice
  - SLVT cells for high performance
  - LVT for buffers (slower, less leakage)
  - Total of 10 layers
- Chip size $2.5\,\text{mm} \times 1.25\,\text{mm} = 3.125\,\text{mm}^2$
- Realized with Cadence toolchain, i.e. Genus, Innovus, Tempus etc.
  - DRC checks with Siemens Calibre

Area Consumption obtained after Synthesis

Module	Combinational [mm^2]	Sequential [mm^2]	Memory [mm^2]	Total [mm^2]
SIKE	0.102	0.105	0.463	0.670
HQC	0.137	0.015	0.150	0.302
NTT	0.006	0.007	0.101	0.114
RISC-V	0.025	0.019	0.151	0.195
Total	0.152	0.157	0.865	1.174

Table: Area consumption in mm^2

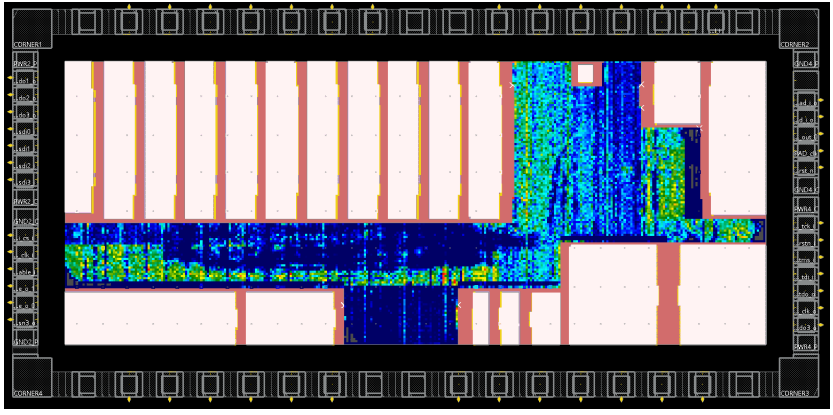
Main Difficulties

- Number and form factor of memories
  - Difficult place & route
  - Long paths from address decoders

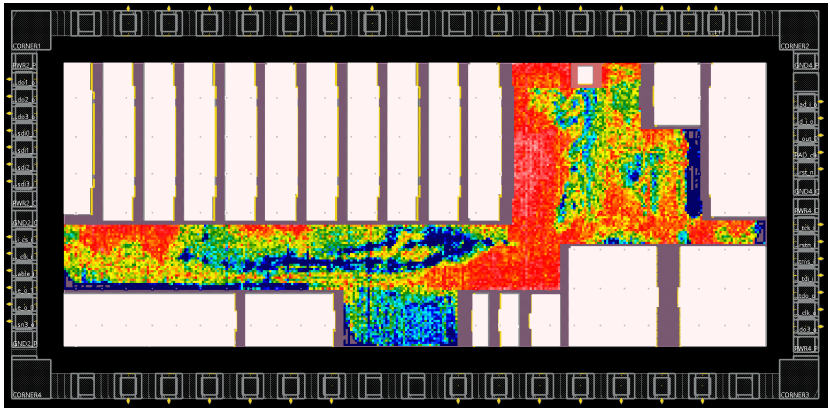
- E.g.: SIKE memories
  - **Instruction:** 1024×192 bit
  - **Data:** 1024×80 bit $\rightarrow$ 10 parallel instances for width of 800 bit

$\rightarrow$ 500 MHz clock generated by Frequency-Locked Loop (FLL) provided by ETH Zürich
  - Critical path from RISC-V to peripherals via system bus
  - Multipliers inside SIKE

Floorplan – Density before buffer insertion

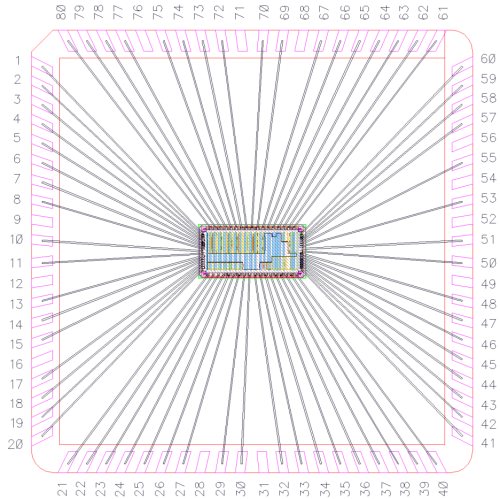


Floorplan – Density after buffer insertion



Bonding Diagram

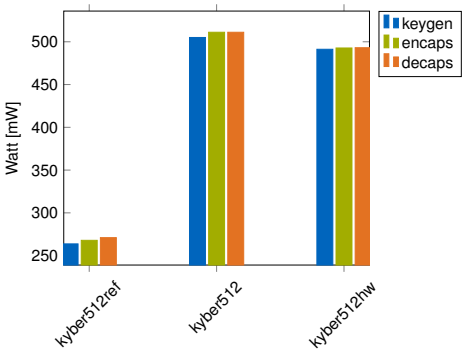
- Quad Flat No-lead (QFN) package
  - 80 pins
  - 1.2 cm $\times$ 1.2 cm
- Core-Voltage 0.8 V
- IO-Voltage 3.3 V



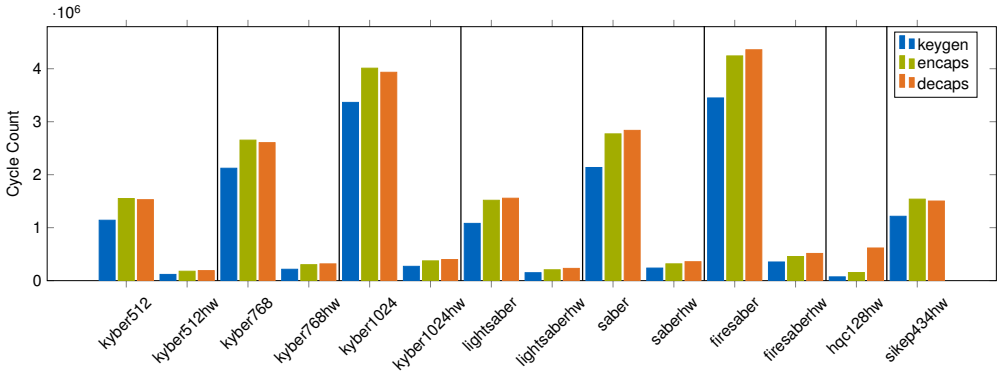
Preliminary Results and Estimations

Total Power Consumption Estimation

- **kyber512ref:** PULPino, no accelerators
- **kyber512:** PULPino, unused accelerators
- **kyber512hw:** PULPino, used accelerators
- $\approx 2\times$ power consumption with our accelerators ($\approx 6\times$ area)



Cycle Count

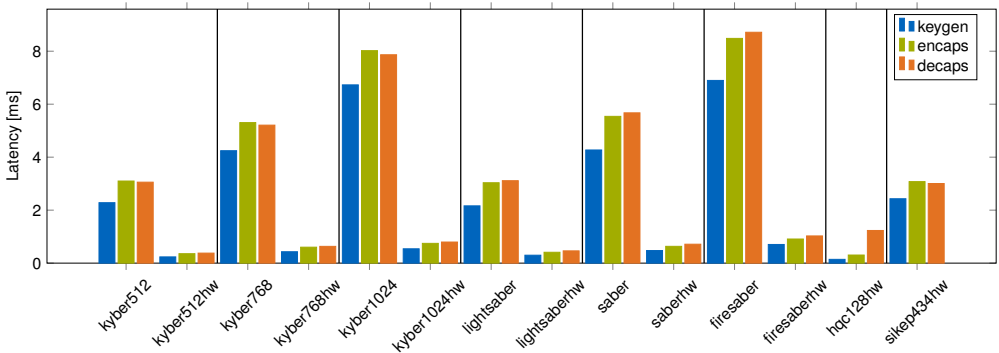


- Improvements up to a factor of ≈ 12

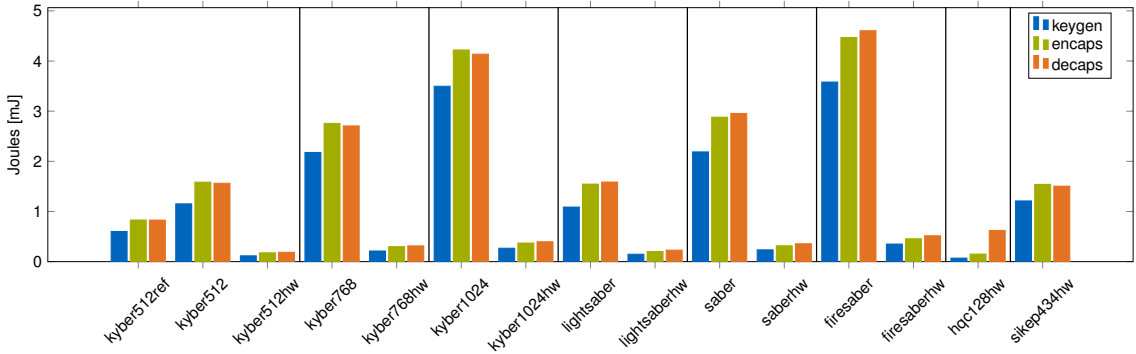
Cycle Count

- Kyber512
  - Keygen: $1,140k \rightarrow 117k$
  - Encaps: $1,549k \rightarrow 178k$
  - Decaps: $1,528k \rightarrow 189k$
- HQC128
  - Keygen: $71k$
  - Encaps: $152k$
  - Decaps: $616k$
- Kyber1024
  - Keygen: $3,364k \rightarrow 270k$
  - Encaps: $4,010k \rightarrow 373k$
  - Decaps: $3,933k \rightarrow 398k$
- Lightsaber
  - Keygen: $1,081k \rightarrow 149k$
  - Encaps: $1,517k \rightarrow 205k$
  - Decaps: $1,555k \rightarrow 231k$
- SIKEp434
  - Keygen: $1,215k$
  - Encaps: $1,538k$
  - Decaps: $1,502k$
- Firesaber
  - Keygen: $3,448k \rightarrow 352k$
  - Encaps: $4,241k \rightarrow 455k$
  - Decaps: $4,357k \rightarrow 513k$

Latency in [ms] according to 500 MHz



Computed Energy Consumption

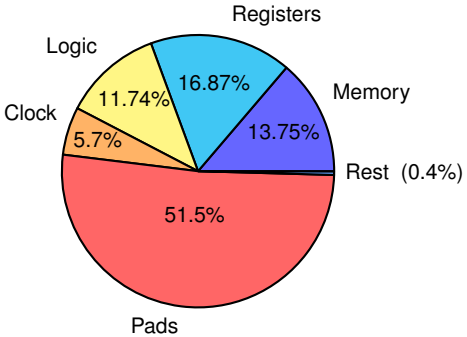


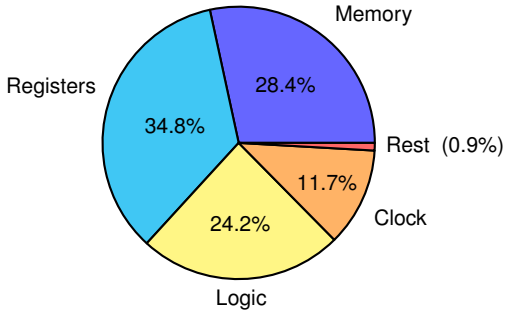
- Factor 7 – 13 of savings depending on algorithm, security level and function

Computed Energy Consumption

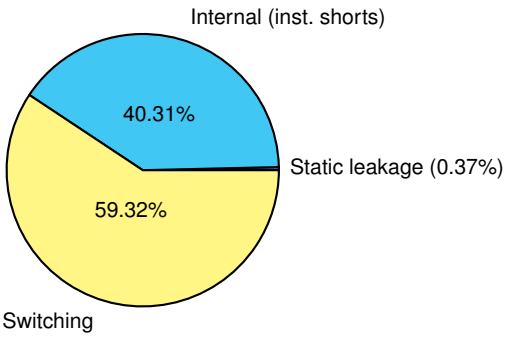
- Kyber512
  - Keygen: 1152 µJ → 115 µJ
  - Encaps: 1584 µJ → 175 µJ
  - Decaps: 1562 µJ → 186 µJ
- HQC128
  - Keygen: 70 µJ
  - Encaps: 150 µJ
  - Decaps: 623 µJ
- Kyber1024
  - Keygen: 3496 µJ → 267 µJ
  - Encaps: 4220 µJ → 370 µJ
  - Decaps: 4136 µJ → 396 µJ
- Lightsaber
  - Keygen: 1088 µJ → 147 µJ
  - Encaps: 1545 µJ → 202 µJ
  - Decaps: 1587 µJ → 229 µJ
- SIKEp434
  - Keygen: 1210 µJ
  - Encaps: 1540 µJ
  - Decaps: 1504 µJ
- Firesaber
  - Keygen: 3580 µJ → 350 µJ
  - Encaps: 4468 µJ → 455 µJ
  - Decaps: 4603 µJ → 515 µJ

Power Analysis Kyber-512 Encapsulation





Power Analysis Kyber-512 Encapsulation



- **Internal**: Small shorts when switching
- **Switching**: (De-) charging capacitive load

$\rightarrow$ Static leakage neglectable

Outlook

- Check functionality
  - Try to verify our estimations (power, energy) with measurements
- Investigate side-channel secured implementations
  - Adjust code of masked implementations to our platform
  - Perform measurements

Thank you for your attention!

Contact: patrick.karl@tum.de

References

[Fri+21] T. Fritzmann et al. “Masked Accelerators and Instruction Set Extensions for Post-Quantum Cryptography”. In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* (Nov. 2021), pp. 414–460.

[Gau+17] M. Gautschi et al. “Near-Threshold RISC-V Core With DSP Extensions for Scalable IoT Endpoint Devices”. In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 25.10 (Oct. 2017), pp. 2700–2713.

[RM19] D. B. Roy and D. Mukhopadhyay. “High-Speed Implementation of ECC Scalar Multiplication in GF(p) for Generic Montgomery Curves”. In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 27.7 (July 2019), pp. 1587–1600.