Chair for Security in Information Technology
TUM School of Computation, Information and Technology
Technical University of Munich

The Performance and (Hidden) Communication Cost of Hardware Accelerators for Hash Primitives Used in Post-Quantum-Cryptography

Patrick Karl[1], Jonas Schupp[1], and Georg Sigl[1,2]

Motivation

- Post-Quantum Cryptography (PQC) uses computationally intensive subroutines
  - Huge percentage for hashing and PRNG (mostly SHA-3)
  - Other examples NTT-based polynomial multiplication (mostly for lattice schemes)

→ HW acceleration obviously makes sense

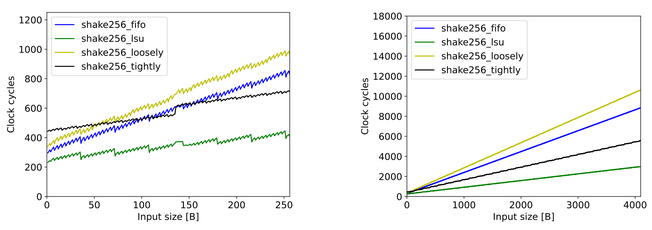
✗ BUT accelerators often under-utilized!

Contribution

- Benchmarks of architectures for **Hash** (variable input, fixed output length) and **CSPRNG** scenario (fixed input, variable output length)
- Case-Study of **SPHINCS+**, **Kyber**, **Dilithium** and **Falcon**

Benchmark Results - SHAKE256

Hash Scenario

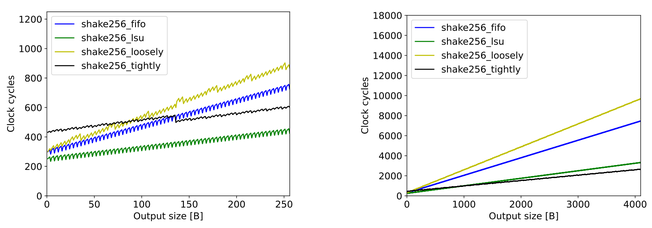


(a) Hash scenario up to 256 Bytes input

(b) Hash scenario up to 4096 Bytes input

- Tightly-coupled efficient for small inputs, suffers from “software absorption”
- LSU-coupled version efficient due to parallel absorption/data transfer

CSPRNG Scenario



(a) CSPRNG scenario up to 256 Bytes output

(b) CSPRNG scenario up to 4096 Bytes output

- LSU-coupled version still efficient in general

Hardware Acceleration[a]

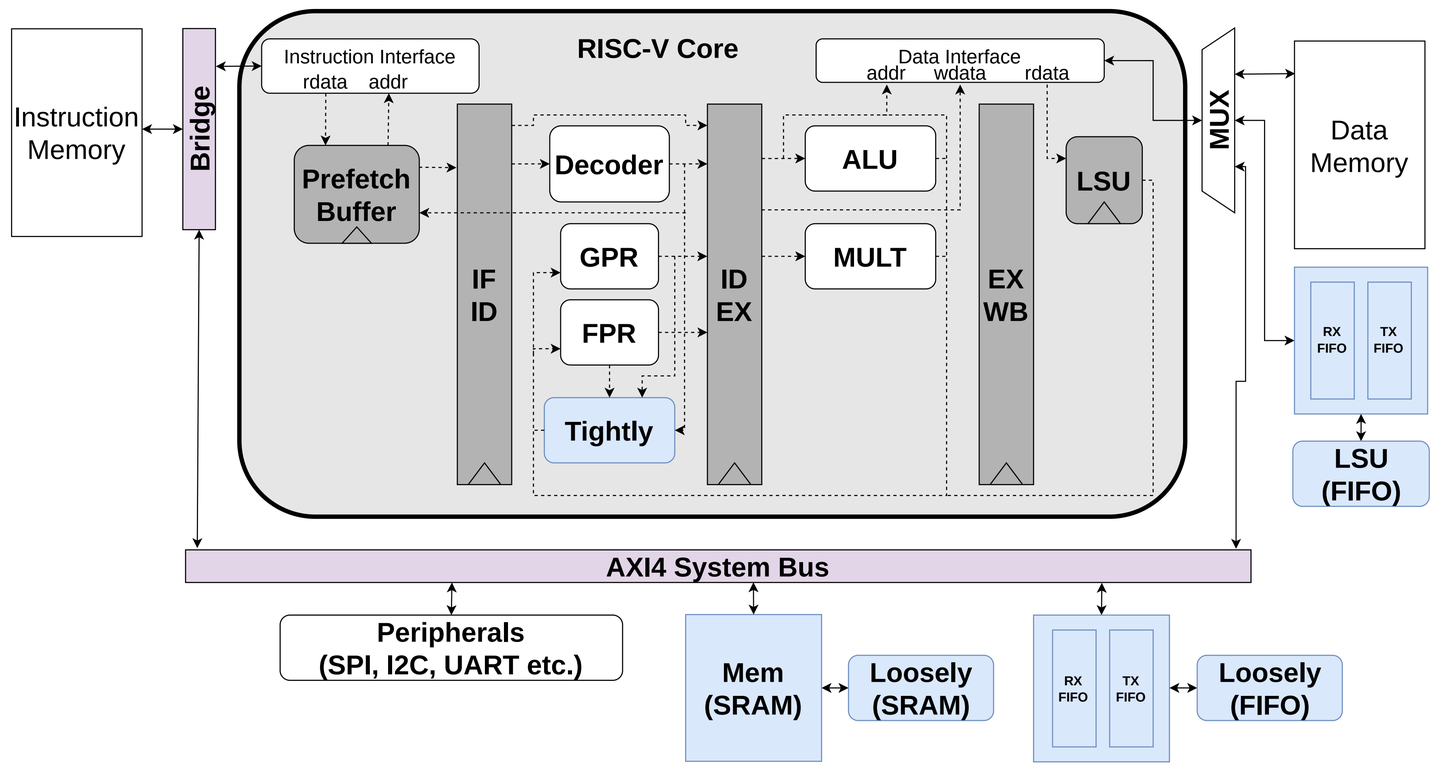
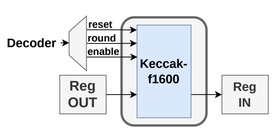
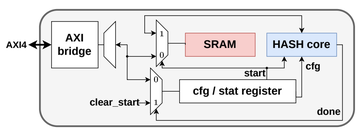


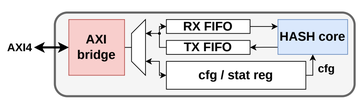
Figure 1 Overview of the platform and the different coupling options



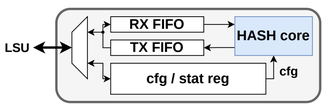
(a) Tightly Coupled Architecture



(b) Loosely (SRAM) Coupled Architecture



(c) Loosely (FIFO) Coupled Architecture



(d) LSU (FIFO) Coupled Architecture

Case-Study: SPHINCS+

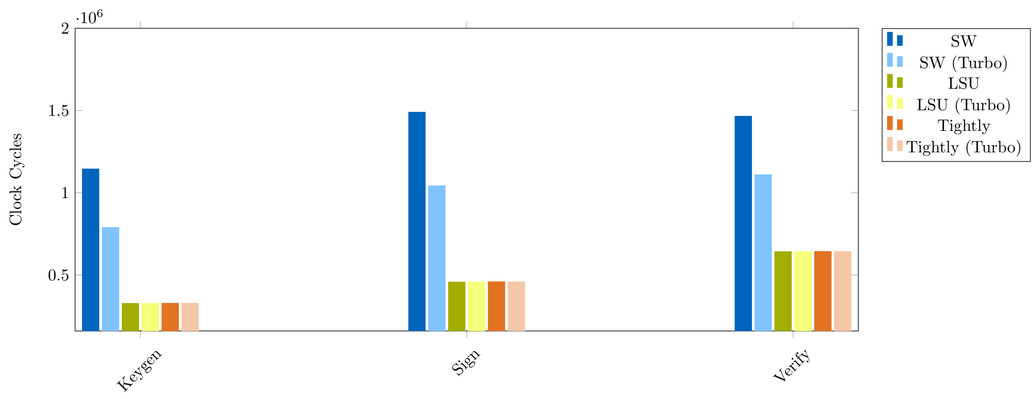
Architecture	Keygen	Sign	Verify
shake256_sw	145 119	3 602 835	201 209
shake256_lsu	1 725	42 598	2 458
shake256_tightly	2 902	71 921	4 091

Table 1 Kilo cycles for SPHINCS+ 128f-simple.

Architecture	Keygen	Sign	Verify
shake256_sw	9 288 740	70 607 335	69 400
shake256_lsu	109 770	837 742	852
shake256_tightly	184 914	1 410 992	1 419

Table 2 Kilo cycles for SPHINCS+ 128s-simple.

Case-Study: Kyber



- Round-reduction directly visible in SW
  - ✓ About 25 % to 30 % speed-up for turbo-shake256
- Limited gain for accelerators
  - Less than 1 % for turbo-shake256

[1]Chair for Security in Information Technology, Technical University of Munich
[2]Fraunhofer Institute for Applied and Integrated Security, Germany

[a]Patrick Karl, Jonas Schupp, and Georg Sigl. “The Impact of Hash Primitives and Communication Overhead for Hardware-Accelerated SPHINCS+”. In: Constructive Side-Channel Analysis and Secure Design. Ed. by Romain Wacquez and Naofumi Homma. Cham: Springer Nature Switzerland, 2024, pp. 221–239.