

# Hardware Challenges in PQC



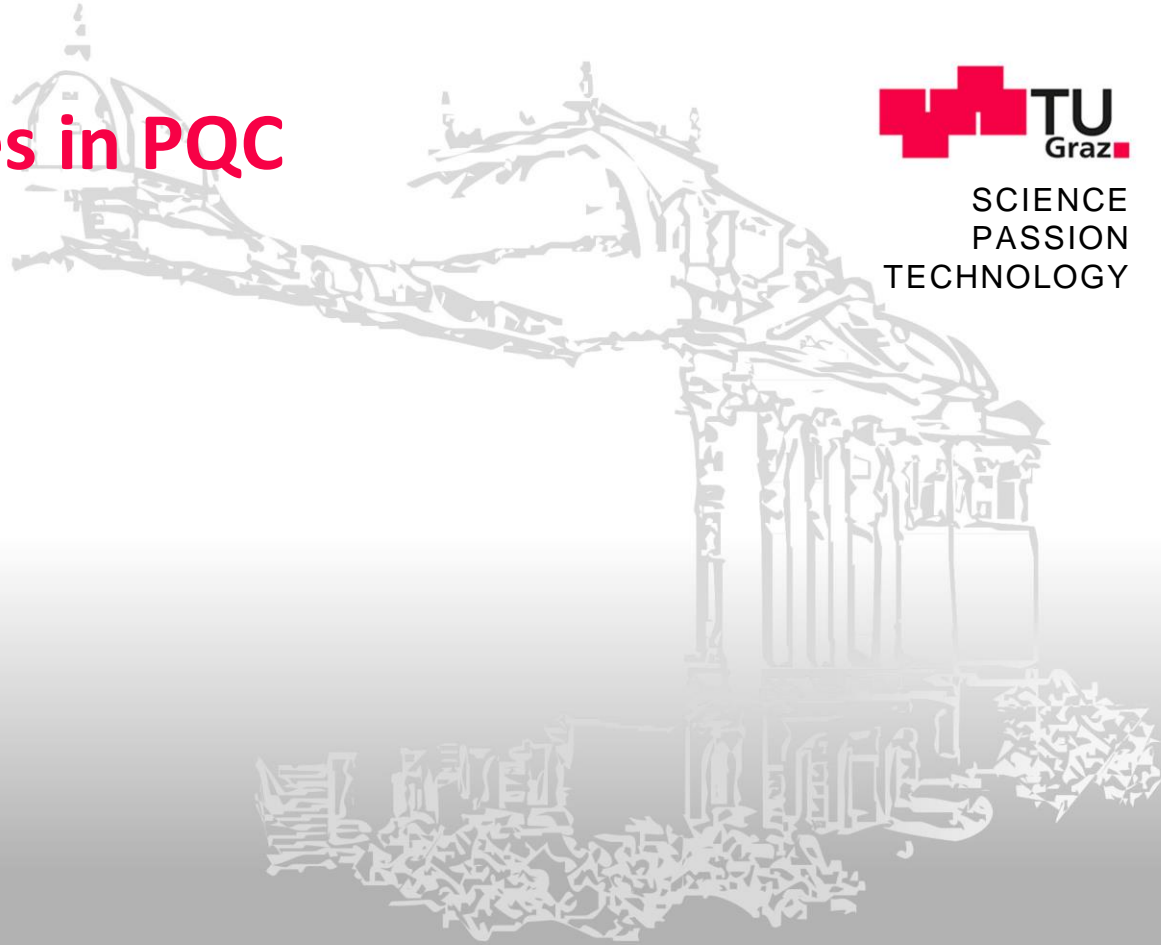
SCIENCE  
PASSION  
TECHNOLOGY

27<sup>th</sup> March 2025

OPTIMIST

Sujoy Sinha Roy

Graz University of Technology



# Customized hardware for PQC

Small survey asking two questions:

1. What makes hardware design for PQC challenging?
2. Why is it non-trivial to reuse or port PQC hardware designs?

# 1. Mathematical Complexity and Diversity

PQC mathematical foundations

- Lattice-based
  - Multivariate-based
  - Hash-based
  - Code-based
  - Isogeny-based
- + PQC schemes add various optimizations

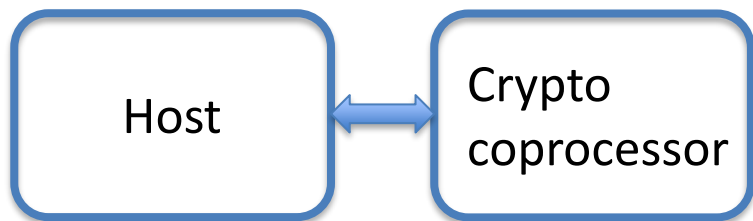
# 1. Mathematical Complexity and Diversity

PQC mathematical foundations

- Lattice-based
- Multivariate-based
- Hash-based
- Code-based
- Isogeny-based

+ PQC schemes add various optimizations

Coprocessor needs to support KEM + DSA




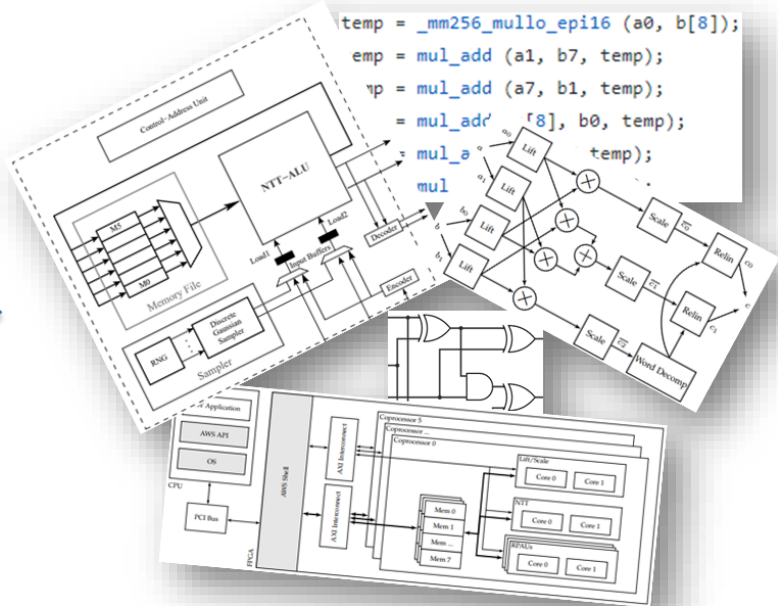
Challenges:

- Diverse math foundations
  - Different parameters
- Complex architectural requirements

# 2. From Crypto Math to Hardware – Sequential Process

$\Lambda_u^\perp(g^T) = \{z \in \mathbb{Z}^k : g^T z = u \pmod q\}$   
 $P_r(E=z) = \frac{1}{\sqrt{e} \sigma} e^{-z^2 / 2\sigma^2}$   
 $\Delta(\tilde{D}_{\mathbb{Z}^m, \sigma}, D_{\mathbb{Z}^m, \sigma}) < 2^{-k} + 2mz_t \epsilon$   
 $\mathcal{R} = \mathbb{Z}[X] / \langle \Phi_M(X) \rangle$   
 $Q = \prod_{i=0}^{\ell-1} q_i$  RNS  
 $seed_A \leftarrow \mathcal{U}(\{0, 1\}^{256})$   
 $s = \beta_\mu(R_q^{\ell \times 1}; r)$   
 $b' = ((As' + h) \pmod q) \gg (\epsilon_q - \epsilon_p) \in R_p^{\ell \times 1}$

From math  
  
to practice



Cryptographers prioritize security

HW designers work within given spec

Feedback loop will improve situation



### 3. Hardware gives Efficiency but lacks Flexibility

- You can't scale up/down like in software
- Every new use case = partial redesign

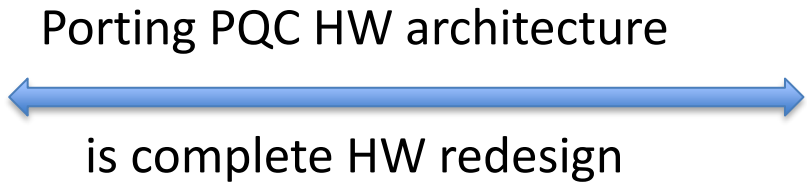
### 3. Hardware gives Efficiency but lacks Flexibility

- You can't scale up/down like in software
- Every new use case = partial redesign



HSM

High-speed crypto  
e.g., 8,000 DSA/s



Contact-less payment



Constrained crypto  
e.g., low power

### 3. Hardware gives Efficiency but lacks Flexibility – Case study MAYO

MAYO<sub>3</sub> signing on SW

- High-end Intel/AMD = 1,200/sec\*
- Constrained Arm M4 = 0.5/sec\*

\*MAYO <https://pqmayo.org/assets/specs/mayo.pdf>



### 3. Hardware gives Efficiency but lacks Flexibility – Case study MAYO

MAYO<sub>3</sub> signing on SW

- High-end Intel/AMD = 1,200/sec\*
- Constrained Arm M4 = 0.5/sec\*

MAYO<sub>3</sub> signing on HW (high-performance)

- Kintex-7 FPGA = 1,500/sec<sup>#</sup>
- 28nm ASIC = 25,000/sec<sup>#</sup>

Can we use the same HW architecture for contactless cards?

\*MAYO <https://pqmayo.org/assets/specs/mayo.pdf>

<sup>#</sup>Whipping MAYO paper from CCS 2024

### 3. Hardware gives Efficiency but lacks Flexibility – Case study MAYO

MAYO<sub>3</sub> signing on SW

- High-end Intel/AMD = 1,200/sec\*
- Constrained Arm M4 = 0.5/sec\*

MAYO<sub>3</sub> signing on HW (high-performance)

- Kintex-7 FPGA = 1,500/sec<sup>#</sup>
- 28nm ASIC = 25,000/sec<sup>#</sup>
- **2 mm<sup>2</sup> area, 4 W power in ASIC**

Can we use the same HW architecture for contactless cards?

→ We need a new design for low area and power

\*MAYO <https://pqmayo.org/assets/specs/mayo.pdf>

<sup>#</sup>Whipping MAYO paper from CCS 2024

Starting from a given git RTL project, how easy it is to get the hardware?

## 4. Design reusability and portability issues

1. Development environment setup
  - Vivado/Vitis installation 50+ GB
  - Cadence/Synopsys licensing
  - Toolchain version alignment
  - Board-specific configurations

## 4. Design reusability and portability issues

### 1. Development environment setup

- Vivado/Vitis installation 50+ GB
- Cadence/Synopsys licensing
- Toolchain version alignment
- Board-specific configurations

### 2. Design implementation and testing

- IP blocks are rather problematic
- Simulation, synthesis, placement, ..., take huge time
- Platform variations

# Closing thoughts

- Hardware = design + optimization + platform-specific tuning
- PQC adds:
  - Complex math,
  - Larger data,
  - Diverse application demands
- Reuse and portability remain open research challenges