

Correlation Power Analysis of Hamming-Quasi Cyclic



Huizhen Zhou¹ Luke Beckwith^{1,2} Sanjay Deshpande³ Xiaokuan Zhang¹
Kris Gaj¹ Jakub Szefer³



<https://caslab.io>



Northwestern

¹George Mason University, Fairfax, VA, USA

²PQSecure Technologies, Boca Raton, FL, USA

³Northwestern University, Evanston, IL, USA

¹{hzhou9, lbeckwit, xiaokuan, kgaj}@gmu.edu

²luke.beckwith@pqsecurity.com

³{sanjay.deshpande1, jakub.szefer}@northwestern.edu

1 Motivation

- The transition of **Post-Quantum Cryptography (PQC)** into real-world deployments demands both secure math and **secure implementations**.
- NIST selected **Hamming Quasi-Cyclic (HQC)** [1] for standardization as the alternative key encapsulation mechanism, expanding the post-quantum portfolio beyond lattice-based ML-KEM.
- For real-world deployments, **HQC's implementation security is now a critical priority**.

Contribution: This work builds upon [2] and demonstrates the first power side-channel attack on HQC in hardware and successfully recovers the full secret key.

2 Hamming-Quasi Cyclic

- HQC, like any other key encapsulation mechanism, consists of three core algorithms; **Key Generation, Encapsulation, and Decapsulation**.
- HQC relies on polynomial multiplication between a dense and a sparse polynomial:** for e.g., computing $h \cdot y + x$, where h is dense and y is sparse with a fixed Hamming weight w .
- Sparse polynomial multiplication reduces to cyclic rotations and XOR. $h \cdot y = \bigoplus_{i=0}^{w-1} \text{rot}(h, y_i)$
- Hardware implementation of the **dense-sparse polynomial multiplier leaks secret-dependent power patterns**.
- Decapsulation is the ideal attack target** since it uses the same secret key many times. The dense polynomial is a known public ciphertext and the sparse polynomial is part of the secret key.

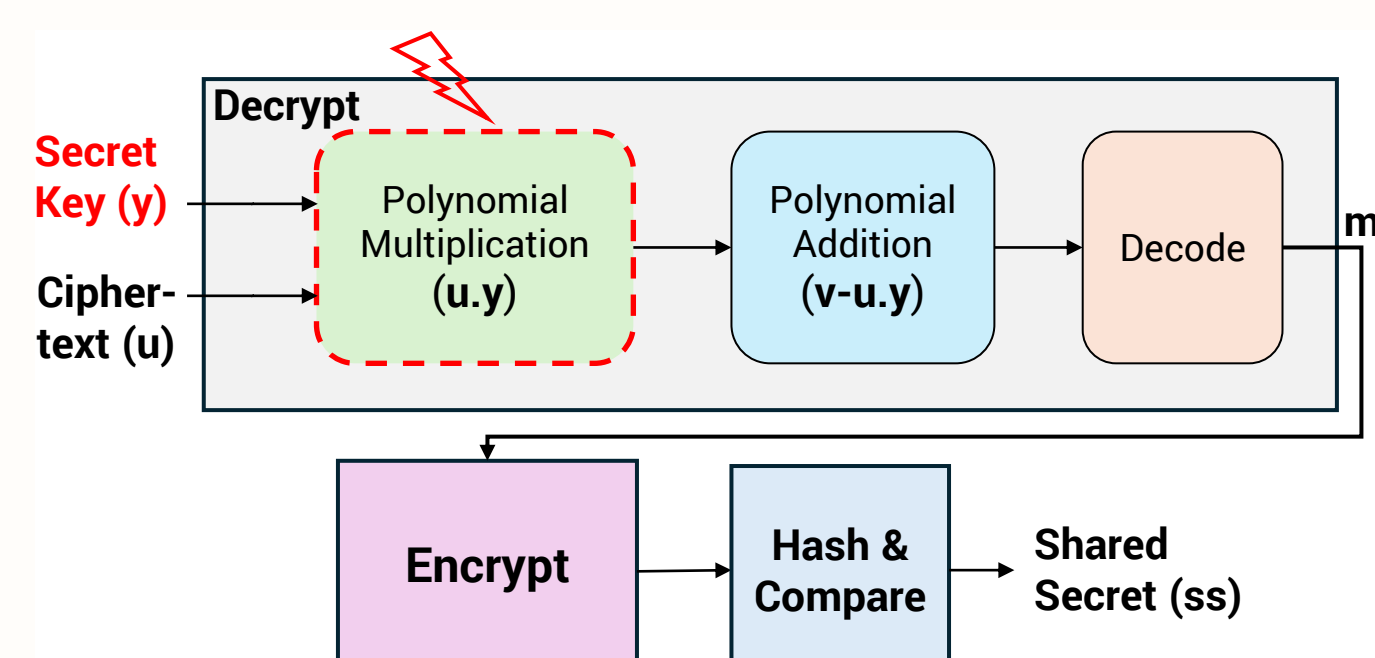
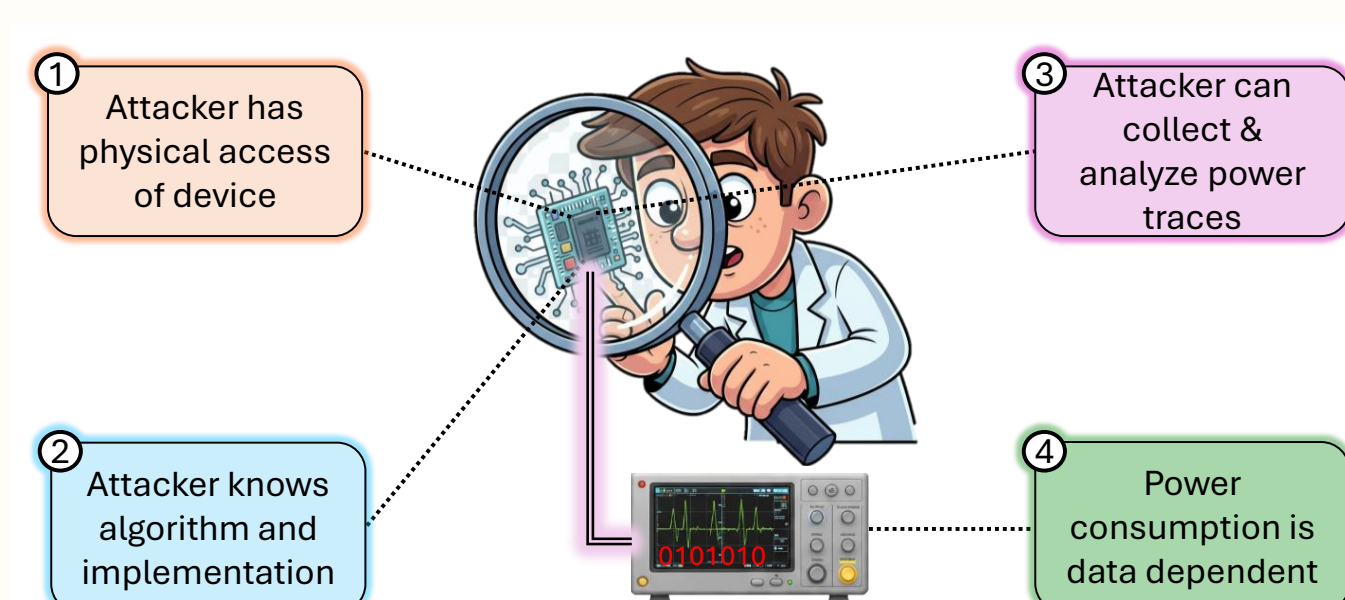


Figure 1: Block diagram representation of HQC decapsulation algorithm.

3 Threat Model

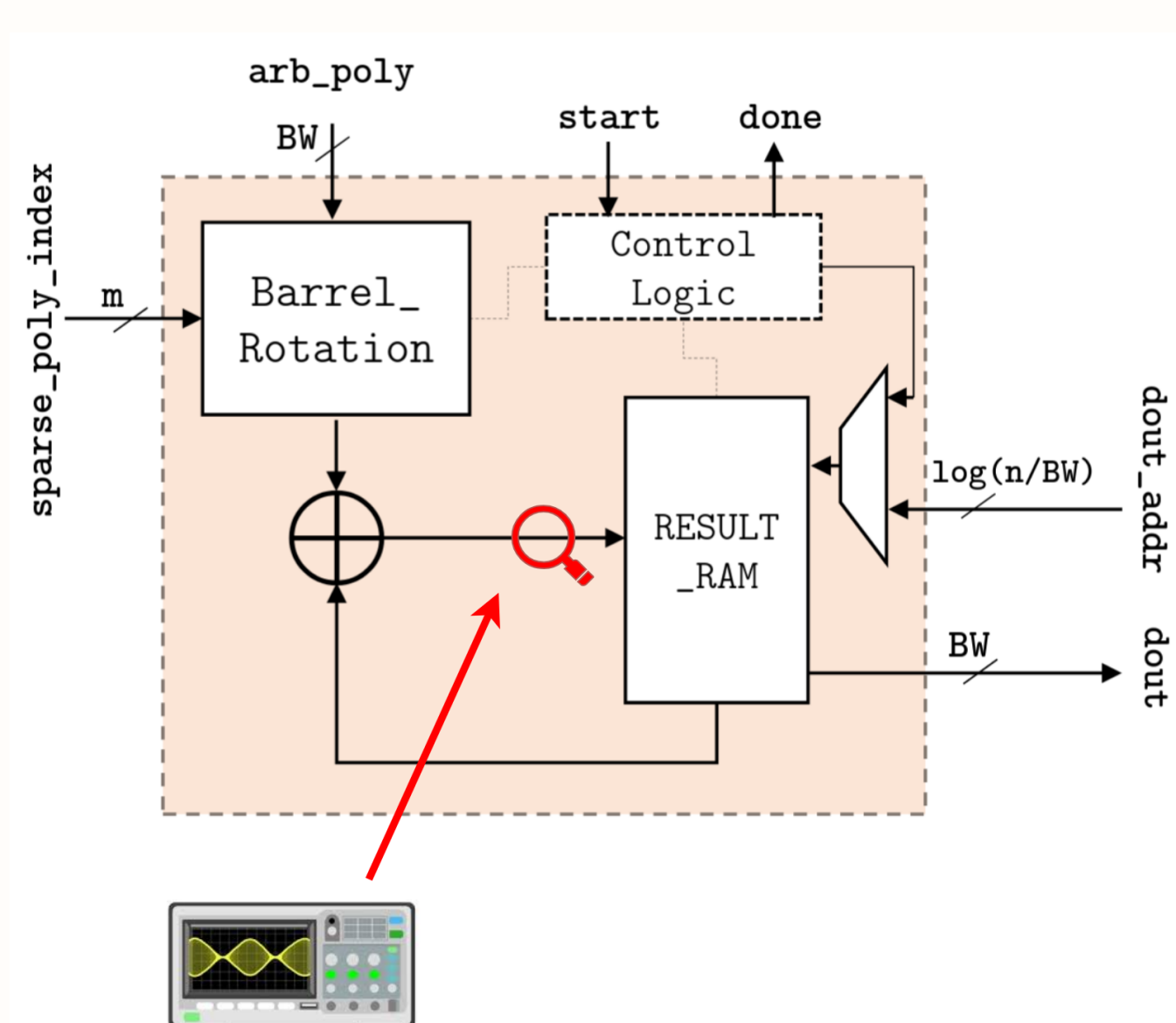


Although the attack is applicable to all HQC parameter sets, we select HQC-1 for this demonstration. Table 1 summarizes the relevant parameters used in the attack.

Table 1: Parameters of the attacked HQC-1 implementation.

Parameter	Value
Ring degree N	17,669
Secret weight w	66
Datapath width	{8,16,32,64,128}
Words per poly	$\lceil N/\text{width} \rceil$
Target clock	10 MHz
ADC sampling rate	200 Msps
Samples per cycle	20
Trace length	100,000 samples

4 Working of the Attack

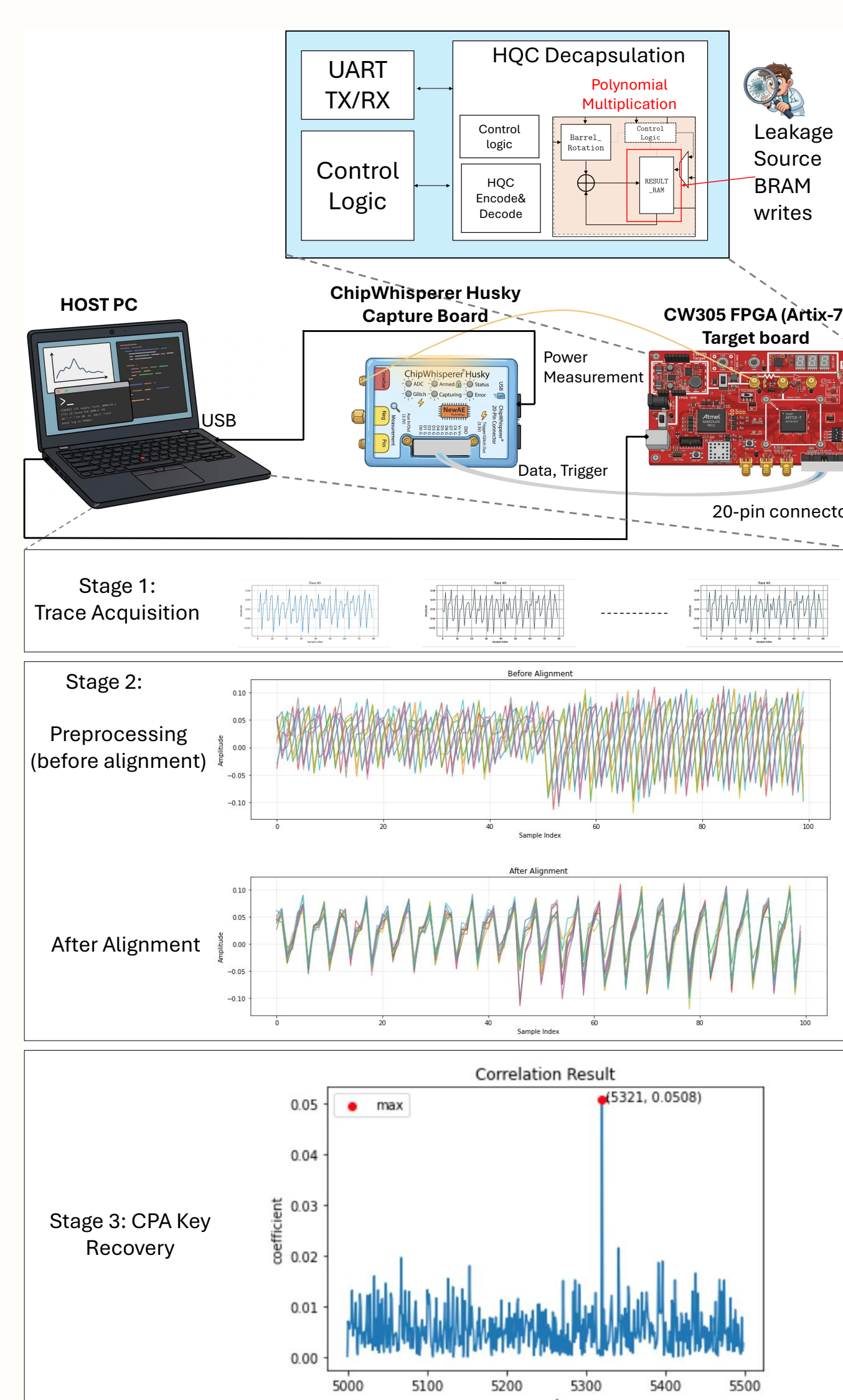


- XOR-accumulator:** each round computes $s^{(i)} = s^{(i-1)} \oplus \text{rot}(h, y_i)$, written to BRAM one 128-bit word at a time.
- Power model:** the secret sparse index determines the rotation amount. The bit-level rotation is

applied to the known dense input, the word rotation is applied by adjusting the state read address. A Python script modeling this behavior generates a Hamming Weight power model for each key guess

- Per-round CPA:** for each round, we compute and correlate the hypothetical HW of each of the $N=17,669$ candidates k with the measured trace. The correct guess y_i yields the highest correlation coefficient.
- Sequential attack procedure:** once $y_0, \dots, y_{i-1}$ are recovered, the partial state $s^{(i-1)} = \bigoplus_{\ell < i} \text{rot}(h, y_\ell)$ is reconstructed from the known ciphertext and fed back into round i 's model.

5 Experimental Setup & Attack



Stage 1 (Trace Acquisition): data is transferred from the host to the FPGA target via UART. Then a trigger signal is issued to initiate the decapsulation computation. During execution, the oscilloscope captures power measurements and streams them back to the host PC.

Stage 2 (Preprocessing & Alignment): the collected traces are aligned and filtered to compensate for trigger jitter.

Stage 3 (CPA Key Recovery): the correlation power analysis computes

the highest coefficients for the attacking subkey. The correct subkey will have the highest coefficient.

Stage 4 (Final Analysis): All multiplier bit-widths were attacked. The correct key ranking at various trace counts is recorded and is shown in the following figure.

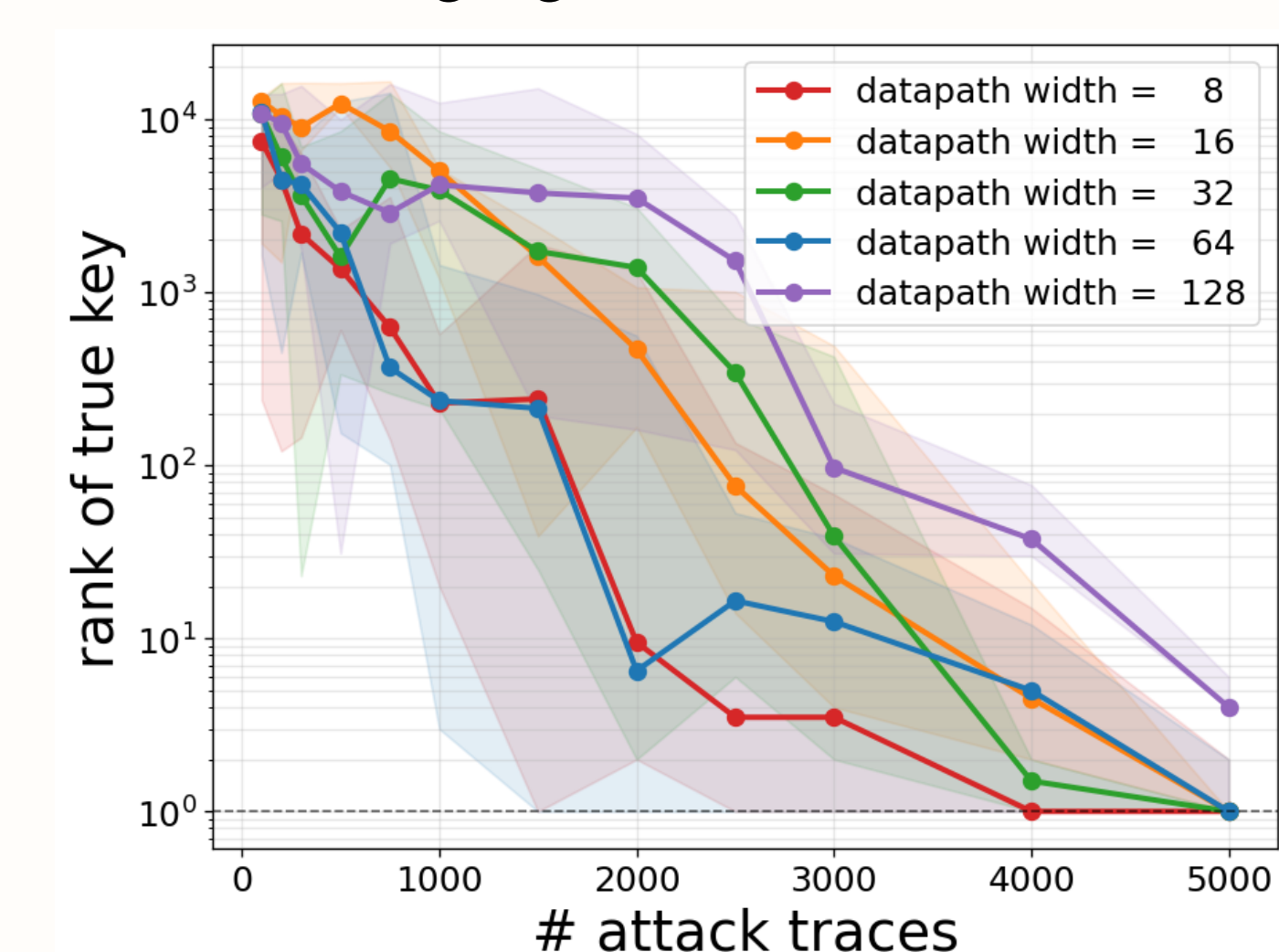


Figure 2: Key rank vs number of traces for each multiplier datapath width. The attack was repeated multiple times: solid lines show the median; shaded bands show the min-max range.

6 Conclusion

- A practical CPA attack against FPGA-based HQC:** this demonstration shows that unprotected hardware implementations of code-based PQC remain vulnerable to side-channel attacks, a timely result given NIST's selection of HQC for standardization.
- Broadly applicable attack techniques:** this attack technique extends beyond HQC to other sparse-polynomial-based PQC schemes such as BIKE [3], highlighting a recurring vulnerability in code-based post-quantum cryptography hardware.

7 References

- [1] P. Gaborit et al., "HQC," tech. rep., National Institute of Standards and Technology, 2025.
- [2] L. Beckwith et al., "Power side-channel key recovery attack on a hardware implementation of bike," in *2024 Asian Hardware Oriented Security and Trust Symposium (AsianHOST)*, pp. 1–6, 2024.
- [3] N. Aragon et al., "BIKE," tech. rep., National Institute of Standards and Technology, 2022.