Correlation Power Analysis of Hamming-Quasi Cyclic

Huizhen Zhou[1] Luke Beckwith[1,2] Sanjay Deshpande[3] Xiaokuan Zhang[1]
Kris Gaj[1] Jakub Szefer[3]

[1]George Mason University, Fairfax, VA, USA
[2]PQSecure Technologies, Boca Raton, FL, USA
[3]Northwestern University, Evanston, IL, USA
[1]{hzhou9, lbeckwit, xiaokuan, kgaj}@gmu.edu
[2]luke.beckwith@pqsecurity.com
[3]{sanjay.deshpande1, jakub.szefer}@northwestern.edu

https://caslab.io

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 scheme, 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 $\mathrm{h} \cdot \mathrm{y} + \mathrm{x}$, where h is dense and y is sparse with a fixed Hamming weight w.
- Sparse polynomial multiplication reduces to cyclic rotations and XOR. $\mathrm{h} \cdot \mathrm{y} = \bigoplus_{\mathrm{i}=0}^{w-1} \mathrm{rot}(\mathrm{h}, \mathrm{y_i})$
- Hardware implementation of the **dense–sparse poynomial 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.

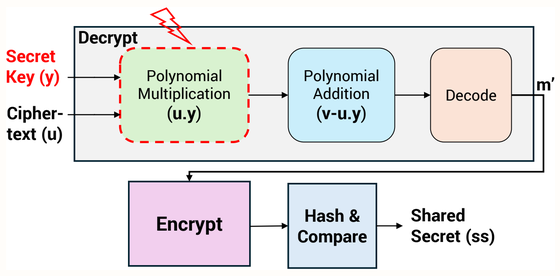
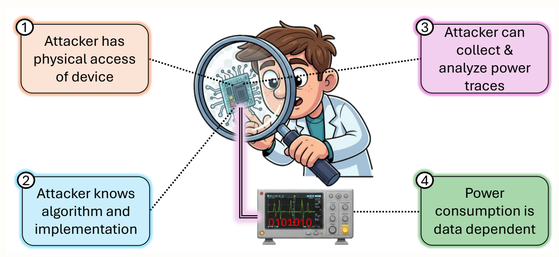


Figure 1: Block diagram representation of HQC decapsulation algorithm.

3 Threat Model

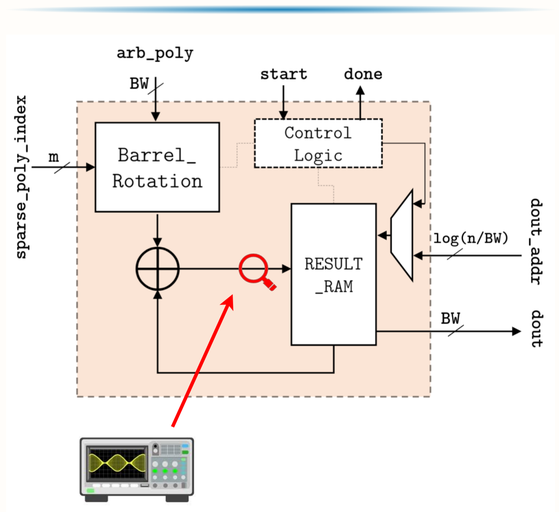


Although the attack is applicable to all HQC parameter sets, we select HQC-1 for this demonstration. Table 1 summarizes the relevant parameters used in the attack.

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

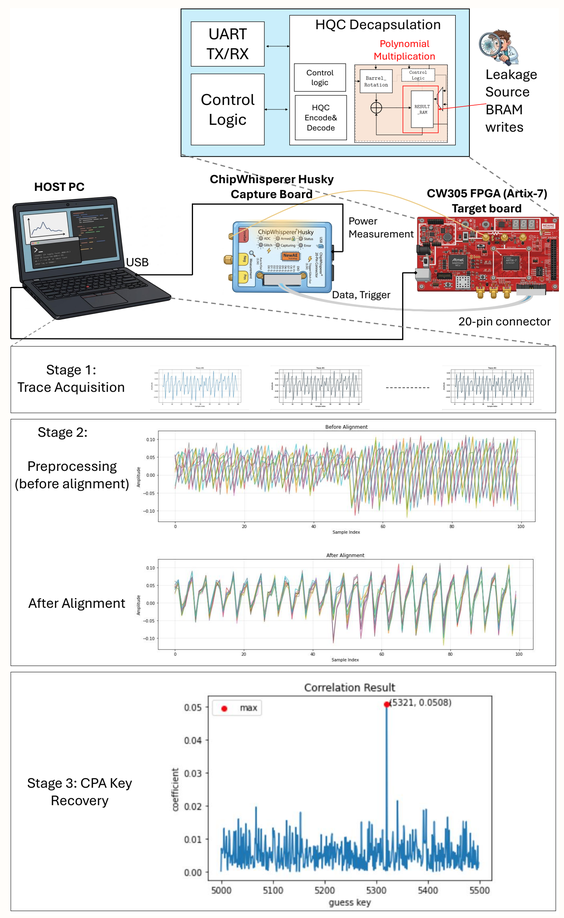
Parameter	Value
Ring degree N	17,669
Secret weight w	66
Word width	128 bits
Words per polynomial	139
Target clock	10 MHz
ADC sampling rate	200 Msps
Samples per cycle	20
Trace length	100,000 samples

4 Working of the Attack



- **XOR-accumulator**: each round computes $\mathrm{s}^{(\mathrm{i})} = \mathrm{s}^{(\mathrm{i}-1)} \oplus \mathrm{rot}(\mathrm{h}, \mathrm{y_i})$, 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 $\mathrm{y_i}$ yields the highest correlation coefficient.
- **Sequential attack procedure**: once $\mathrm{y_0}, \ldots, \mathrm{y_{i-1}}$ are recovered, the partial state $\mathrm{s}^{(\mathrm{i}-1)} = \bigoplus_{\ell<\mathrm{i}} \mathrm{rot}(\mathrm{h}, \mathrm{y}_\ell)$ is reconstructed from the known ciphertext and fed back into round i's model.

5 Experimental Setup & Attack



Stage 1 (Trace Acquisition): data is transfered 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.

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, C. Aguilar-Melchor, N. Aragon, S. Bettaieb, L. Bidoux, O. Blazy, J.-C. Deneuville, E. Persichetti, G. Zémor, J. Bos, A. Dion, J. Lacan, J.-M. Robert, P. Véron, P. L. Barreto, S. Ghosh, S. Gueron, T. Güneysu, R. Misoczki, J. Richter-Brokmann, N. Sendrier, J.-P. Tillich, and V. Vasseur, "HQC," tech. rep., National Institute of Standards and Technology, 2025. available at https://pqc-hqc.org/doc/hqc_specifications_2025_08_22.pdf.

[2] L. Beckwith, H. Zhou, J.-P. Kaps, and K. Gaj, "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, P. Barreto, S. Bettaieb, L. Bidoux, O. Blazy, J.-C. Deneuville, P. Gaborit, S. Gueron, T. Guneysu, C. Aguilar-Melchor, R. Misoczki, E. Persichetti, N. Sendrier, J.-P. Tillich, G. Zémor, V. Vasseur, S. Ghosh, and J. Richter-Brokmann, "BIKE," tech. rep., National Institute of Standards and Technology, 2022. available at https://csrc.nist.gov/Projects/post-quantum-cryptography/round-4-submissions.