A Versatile and Unified HQC Hardware Accelerator

Francesco Antognazza[1], Alessandro Barenghi[1], Gerardo Pelosi[1], Ruggero Susella[2]

[1] Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano, Milano, Italy
[2] STMicrolectronics S.r.l., Agrate Brianza, Italy

Introduction

Post-Quantum Standardization process

- Started by the National Institute of Standards and Technology (NIST) in 2016
- Protect from *harvest now, decrypt later* by attackers with quantum computers
  - All currently deployed asymmetric cryptographic algorithms are vulnerable

Motivations

- HQC [1] is a code-based Key Encapsulation Mechanism with promising performance
- Reference hardware design [2] gave a prompt assessment, although it is un-optimized

We provide an efficient HQC HW design supporting all security levels

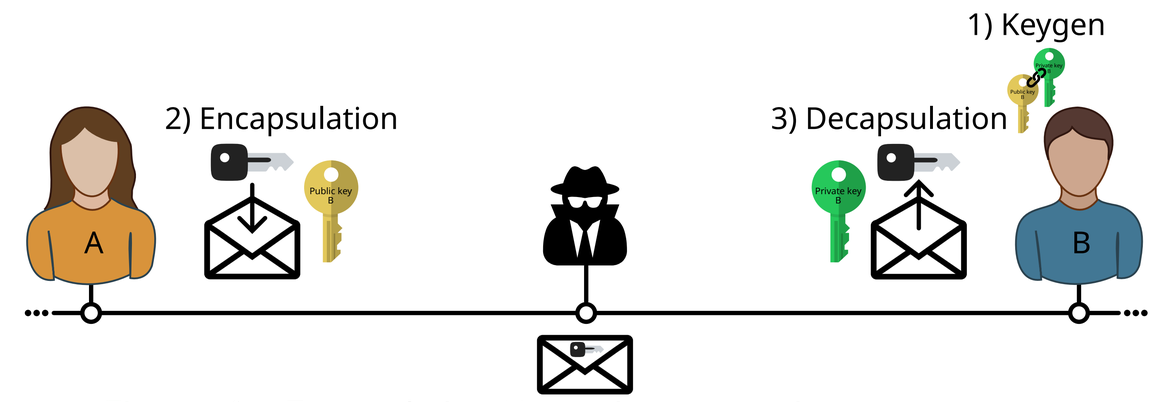


Figure: Key Encapsulation Mechanism exchanging a shared secret

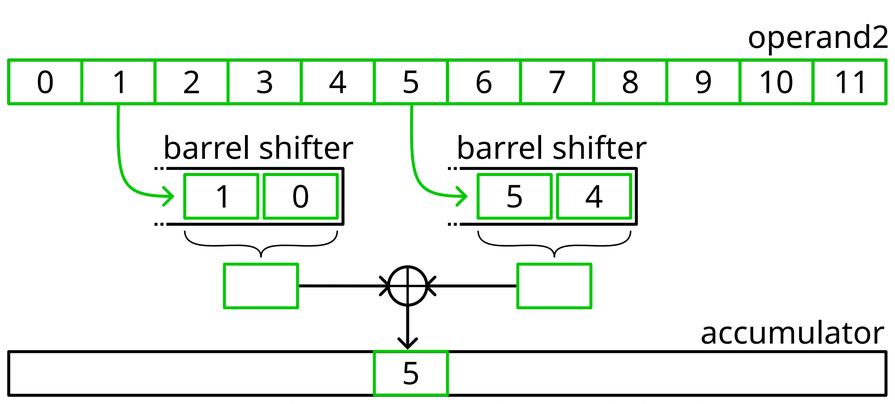


Figure: binary polynomial multiplication

1. Multiplication of binary polynomials

- `operand1` has few non-zero coefficients; `operand2` coefficients are accessed in blocks
- For each `operand2` block: shift the bits and accumulate the result
- `operand2` starting block and shift size are determined by the non-zero `operand1` bits
- Perform parallel rotations, scaling indefinitely with the number of memory read ports

Using two dual-port memories, the operation latency decreased by 4×

2. Concatenated Reed-Muller/Reed-Solomon decoder

- Reed-Muller code is decoded with a Maximum Likelihood approach using the fast Hadamard transform
  - Rapidly compare the 128 values with a tree of comparators to find the peak value
- Adapted a low-latency, highly optimized Reed-Solomon decoder designed for network communication [4]
  - Supporting in a single design all the three shortened RS codes defined by the HQC security levels

53× faster than the HLS-based reference code, and taking only 1/2 LUTs and 1/10 FFs

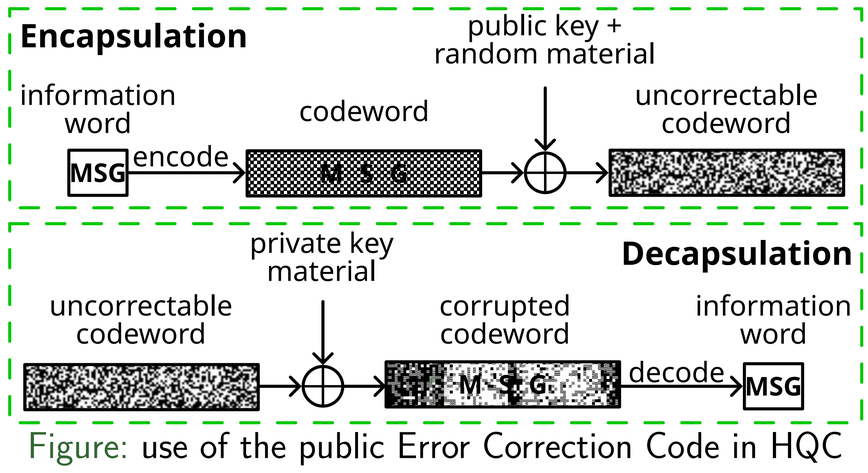


Figure: use of the public Error Correction Code in HQC

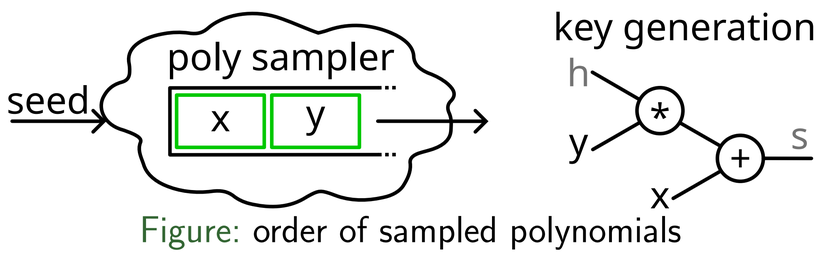


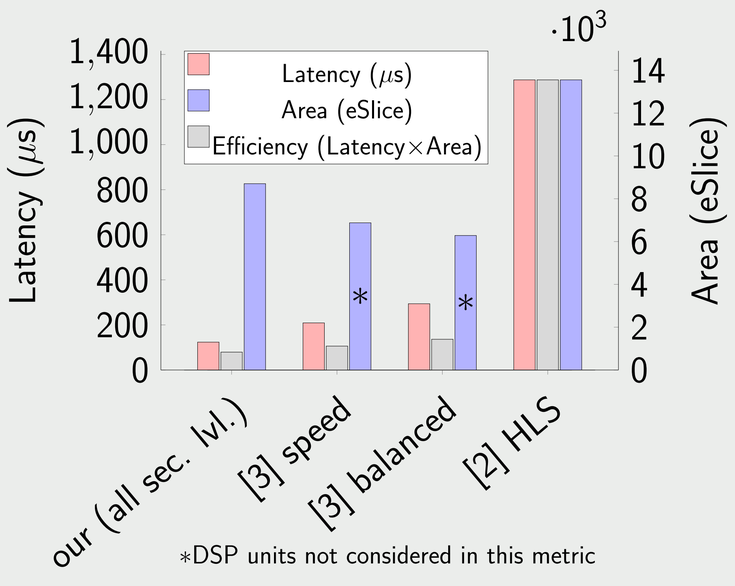
Figure: order of sampled polynomials

3. Sampling of binary polynomials

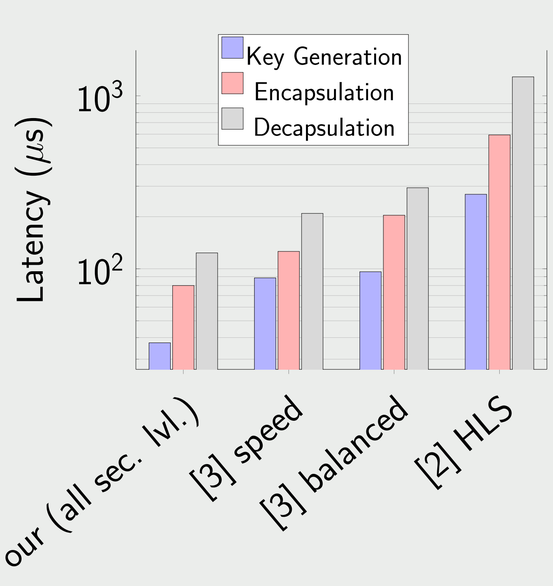
- Sample the polynomials following the data dependency given by the algorithms of primitives
- Improvement to the HQC specification, benefited by both HW and SW implementations

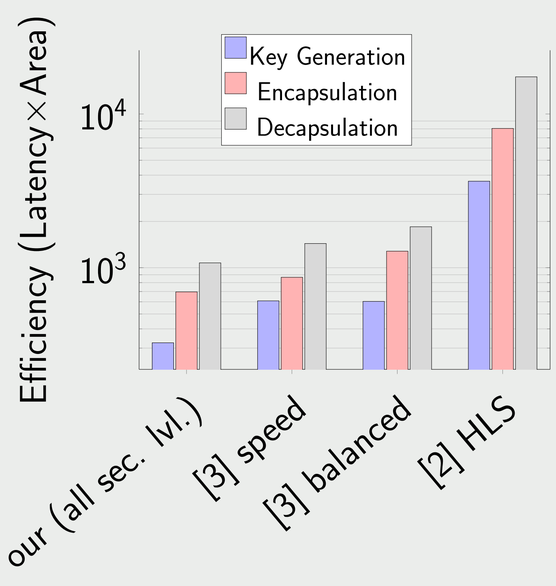
Performance gains from 13% to 32% on the entire cryptographic primitive at no cost

4. Comparison of the Key Generation, Encapsulation, and Decapsulation primitives (sec. margin of AES-128)



*DSP units not considered in this metric





Conclusions

A **single unified HQC HW** design compatible with:

- all the security levels (equiv. to the ones of AES-128, AES-192, AES-256)
- all the KEM primitives (key generation, encapsulation, decapsulation)

Compared to designs **compatible only with the lowest security level**:

- latency reduced from 1.56× to 2.38×
- efficiency improved from 1.24× to 1.88×

Acknowledgements

This work was carried out with partial financial support of the Italian MUR (PRIN 2022 project POst quantum Identification and eNcryption primiTives: dEsign and Realization (POINTER) ID-2022M2JLF2)

References

[1] Carlos Aguilar Melchor and et al. *HQC Documentation*. [Online]. Available: http://pqc-hqc.org/doc/hqc-specification_2023-04-30.pdf. 2023.

[2] Carlos Aguilar Melchor and et al. "Towards Automating Cryptographic Hardware Implementations: A Case Study of HQC". In: *CBCrypto 2022, Trondheim, Norway, May 29-30, 2022*. Vol. 13839. LCNS. Springer, 2022, pp. 62–76. DOI: 10.1007/978-3-031-29689-5_4.

[3] Sanjay Deshpande and et al. "Fast and Efficient Hardware Implementation of HQC". In: *IACR ePrint* (2022). URL: https://eprint.iacr.org/2022/1183.

[4] Yingquan Wu. "New Scalable Decoder Architectures for Reed-Solomon Codes". In: *IEEE Trans. Commun.* 63.8 (2015). DOI: 10.1109/TCOMM.2015.2445759.

https://www.deib.polimi.it <name>.<surname>@polimi.it