Electronics, Communications, and Computing Summit
Vol. 3, No. 3, Jul - Sep 2025, pp. 12-20
ISSN: 3107-8222, DOI: https://doi.org/10.17051/ECC/03.03.02

Hardware-Efficient VLSI Implementation of Post-Quantum Cryptography Primitives

Prerna Dusi[1], Dr. F Rahman[2]

[1]Assistant Professor, Department of Information Technology, Kalinga University, Raipur, India
Email: ku.PrernaDusi@kalingauniversity.ac.in
[2]Assistant Professor, Department of CS & IT, Kalinga University, Raipur, India
Email: ku.frahman@kalingauniversity.ac.in

Article Info

Article history:

Received : 13.07.2025
Revised : 18.08.2025
Accepted : 06.09.2025

Keywords:

Post-Quantum Cryptography,
VLSI,
NTRUEncrypt,
McEliece,
Hardware Implementation,
Low-Power Design,
Lattice-Based Cryptography,
Cryptographic Accelerators

ABSTRACT

The looming disaster that quantum computing is going to be to such classical public-key cryptosystems like RSA and ECC has set the world on a quest to come up with cryptographic algorithms that are capable of resisting quantum de-cryption hence the emergence of the concept to investigate researchers on post quantum cryptography (PQC). The lattice-based algorithms in the form of the NTRUEncrypt and McEliece algorithms, and code-based have been the most prominent PQC schemes to emerge because of its sound security basis and standardization efforts by NIST. Nevertheless, they are computationally very demanding due to their naturally large key sizes and complex operations and thus finding a hardware implementation can be very challenging in energy-limited embedded and IoT applications. This paper deals with the critical necessity of efficient hardware implementations of PQC primitives by suggesting a very hardware-efficient Very Large-Scale Integration (VLSI) architecture suitable to the NTRUEncrypt and McEliece algorithms. The main key requirements are to be minimized in regard to area and power consumption as well as providing the high throughput and introducing flexibility to integrate into other security related applications. An innovative low-power design technique is used; it integrates all levels of clocks gating, operand isolation, as well as architecture-based optimization ideas; these include pipelined modular arithmetic units and hierarchical memory organization. A 28nm CMOS technology node synthesizes the proposed architecture that is described in Verilog HDL. Performance indicators are measured on the basis of gate equivalent (GE) area, power consumption, operating frequency as well as data throughput. The results have shown that up to 45 percent silicon area and 38 percent dynamic power reduction can be had over traditional, baseline implementations at the expense of little reduction in cryptographic performance. It further allows modular integration as a cryptographic co-processor with standard AMBA interfacing to allow secure realization inside RISC-V and ARM-based system-on-chip systems.These results substantiate post-quantum secure cryptographical hardware deployment across real-time embedded systems, opening the doors towards scalable ultra-low power, and future-proof secure communication networks across the post-quantum world.

1. INTRODUCTION

The recent exponential progress in quantum computing has threatened the most widely used classical public key cryptography systems consisting of RSA, Diffie-Hellman, and Elliptic Curve Cryptography (ECC) that despite their classical nature underlies secure digital communication and authentication rate in the modern day. Theoretical difficulties in breaking of these popularly used public key schemes are demonstrated by the quantum algorithms, such as Shor and Grover, which give rise to serious concerns on the continuity of present cryptographic infrastructures. This has led to the growing research field of Post-Quantum Cryptography (PQC) whose objective is to define cryptographic primitives that are secure against both a classical and a quantum adversary. Of these, lattice-based and finally code-based cryptographic schemes exemplified by NTRUEncrypt and McEliece, are currently considered front-runners because they have a sound theoretical basis and

are going through the NIST PQC standardization process.

It is not easy to implement such PQC primitives in hardware, however. Studies of algorithms have now reached maturity, however, practical application to real-time systems, in particular devices with energy and area constraints (such as embedded systems and IoT devices) is fraught with hurdles. As an example, NTRUEncrypt uses high dimensional lattices whose operations require heavy polynomial arithmetic, usually speeded up through Number Theoretic Transform (NTT) based multipliers that may be hardware intensive. McEliece, however, are plagued by large key sizes of both the public keys, as well as overhead syndrome decoding size, requiring large memory bandwidth and storage space. Previous works, such as FPGA implementations by P0ppelmann et al., or side-channel secure designs by Roy et al., have either strongly favoured throughput or cryptographic robustness, at the expense of hardware efficiency, i.e. power and area requirements, which essential requirements in ASIC-based designs in practice.

A pertinent literature gap this paper pinpoints is the territory of the existing absence of a balanced and rational, energy-saving and large-scale VLSI architecture that will be able to support PQC primitives, such as NTRUEncrypt and McEliece, in constrained systems. The present-day deployments are primarily either prioritizing on throughput at the expense of hardware complexity and power consumption or lacking feasibility of deployment due to theoretical resilience concepts. The driving goal behind this work is to develop future proof, quantum resilient cryptographic embedded hardware that is not only secure but also optimized for future real time, low power applications in defence, healthcare, automotive, or consumer Internet of Things (IoT) applications, where the energy and silicon real estate is already severely constrained.

In order to overcome these challenges, the current paper proposes a new VLSI design solution to adopting PQC primitives into a modular, power-concious, area-efficient manner. Particularly, it offers optimised hardware architecture of NTRUEncrypt, and McEliece with the use of such techniques as clock gating, operand isolation, pipelining and modular arithmetic acceleration. The synthesis of the designs is performed in 28nm CMOS technology and compared to current implementations in the literature in area (in gate equivalents), Dynamic power, operations frequency and throughput in cryptography. The proposed architecture also shows up to 45 percent area and 38 percent power savings over and above performance, as well as a scalable and reusable model of a co-processor that can be integrated in a contiguous manner into the AMBA SoCs, demonstrating high level of security in the secure embedded application.

2. BACKGROUND AND RELATED WORK

Post-Quantum Cryptography (PQC) is the cryptography algorithms that resists both classical and quantum computing attacks. Contrary to classical systems like RSA and ECC, built over mathematical problems susceptible to Shor algorithm, PQC primitives are founded on mathematical complications that are assumed to be hard against quantum adversaries, as well. Of the different types of PQC, lattice-based cryptography schemes as well as the type-based cryptographic schemes have received particular attention as they have this rigorous approach to security, their general applicability and tentative good performance in practice. Other examples of lattice-based cryptography are NTRU and Kyber algorithms which are based on the hardness of problems like Shortest Vector Problem (SVP) and Ring Learning With Errors (RLWE) which are believed to be resistant to quantum computing [1]. Code-based cryptography, based on the hardness of decoding a general linear error-correcting code, with the most famous system known as the McEliece cryptosystem, has so far proven to be a powerful framework that withstood voyages of cryptanalysis spanning more than 40 years [2].

The idea behind these schemes is very strong but putting them into practice, especially in hardware (especially within embedded and IoT systems) is fraught with challenges. Lattice-based protocols such as NTRU kiloblocks need high dimensionality polynomial arithmetic, modular arithmetics in finite rings, and convolution multiplications. Modern efficient implementation requires arithmetic circuits that can support modular reductions, multiplication of polynomials using Number Theoretic Transform (NTT) and optimized access to memory. These add to greater area and dynamic power consumption. Conversely, code-based schemes such as McEliece may have exceptionally large public-keys (ordering to hundreds of kilobytes) which require significant storage, and strain the memory bandwidth. Also, the decoding carries out the matrix performances on Galois fields, that is difficult to perform effectively in VLSI [3], [4].

Various scholars have looked into FPGA and ASIC institutes of PQC elements to comprehend their viability in hardware limited systems. P Generatetor lattice-based cryptography accelerated with NTT on FPGAs was recently implemented by P After encoding the C(c) algorithm, it is clear that the high level of throughput was achieved at the cost of higher resource consumption. Likewise, other works have

suggested to implement the Kyber algorithm using FPGAs and tried to optimize this algorithm using fine-grained optimization in the field of polynomials arithmetic, putting aside the energy consumption and ASIC scalability [6]. In the code-based cryptography realm, Roy et al. [7] proposed a side-channel hardened variant of McEliece which is designed on embedded devices by focusing on security rather than efficiency. Nonetheless, the tradeoffs in power, area and throughput - which are important considerations, because real world designs operate in resource limited settings - were not considered in these designs.

In addition, there is still a shortage of modular and reusable Intellectual Property (IP) cores that may be used as shared components in various PQC schemes. The vast majority of implementations are monolithic, being highly coupled with algorithm-specific data paths, thus difficult to integrate and scale. This gap has been further visualized by the current NIST PQC standardization process [8] that in a plea of efficient hardware reference designs requested their automation. Therefore, although the previous methods have helped can our knowledge of PQC hardware feasibility, the field lacks energy-efficient, scalable, and deployment-ready VLSI architectures that can meet a post-quantum security in embedded devices. This paper aims to fill this gap by suggesting a scalable and unified design framework that will place low power and area-efficient implementation of PQC at a priority without regards to cryptographic performance.

3. METHODOLOGY

In order to fulfil the criteria of power-aware and area-optimised hardware implementation for post-quantum cryptographic algorithms, a modular approach was implemented which involves selection of the algorithms, architectural design and hardware-level tuning. The methodology aims at the trade-offs between the cryptographic performance and the strict hardware requirements common in the embedded and low-power environments.

3.1 Algorithm Selection

Cryptographic soundness as well as hardware practicality motivated the choice of PQC algorithms. NTRUEncrypt algorithm has been selected owing to the fact that it works well because of its fast polynomial-based encryption and decryption, and compared with other lattice-based algorithms, the size of the key is relatively short. It directly uses the Ring-Learning-With-Errors (RLWE) problem to secure it, and becomes computationally efficient and scalable in hardware. The security reduction in NTRU is to the problem of polynomial multiplication modulo a prime q:

$c(x) = f(x) * m(x) mod\ q$_____ (1)

Where,

$f(x)$ is the private key polynomial,

$m(x)$ is the plaintext message polynomial, and

$*$ denotes convolution-based polynomial multiplication in the ring $Z_q[x]/(x^N - 1)$.

On the contrary, the McEliece cryptosystem was chosen due to its long term security history and sensitivity to classical attacks as well as quantum attacks. McEliece runs encryption (matrix-vector arithmetic) and syntactic decoding (as messages) based on binary Goppa codes. The most important one in the process of decryption is:

$s = r \cdot H^T$_____(2)

Where,

- s is the syndrome,
- r is the received vector (ciphertext), and
- H is the public parity-check matrix.

This equation highlights the need for Galois Field arithmetic, specifically over F2 or $GF(2^m)$, which imposes additional complexity in hardware.

Algorithm 1. NTRUEncrypt Polynomial Multiplication

This algorithm outlines the core encryption logic implemented in hardware.

```
Algorithm 1: NTRUEncrypt Polynomial Multiplication
Input: Message polynomial m(x), Public key h(x)
Output: Ciphertext c(x)

1: Generate random blinding polynomial r(x)
2: Compute intermediate product: p(x) ← r(x) * h(x) mod q
3: Add message: c(x) ← p(x) + m(x) mod q
4: Return ciphertext c(x)
```

3.2 Architectural Design

The hardware architecture core involved separation into two tandem datapaths optimized with each algorithm. In NTRUEncrypt a pipelined modular multiplier was used to speed the ring based polynomial multiplication, with modular reduction operations. It uses the building block of modular multiplication:

$$r = (a \cdot b)\ mod\ q \_\_\_\_\_(3)$$

To achieve faster speed and preserve modularity, and optionally, a Number Theoretic Transform (NTT) core can be applied to perform a

transformation of polynomial multiplication into element-wise operations:

$$NTT(a \cdot b) = NTT(a) \circ NTT(b) \quad (4)$$

where ∘ represents point-wise multiplication in the transform domain. The pipeline stages ensure parallel processing and improved throughput.

In the case of McEliece, a syndrome computation and decoding unit Galois Field matrix operation were designed to compute and decode more efficiently. The architecture incorporates matrices-vector multiplication unit and finite field logic.

Algorithm 2. McEliece Syndrome Computation and Decoding
This algorithm models the syndrome computation logic for code-based decryption.

Algorithm 2: McEliece Syndrome Computation and Decoding
Input: Received vector r, Public key matrix H
Output: Error vector e or decoding failure

1: Compute syndrome: s ← r × H^t over GF (2)
2: if s = 0 then
3: Return e = 0 (no errors detected)
4: else
5: Perform error pattern lookup or decoding using error locator polynomial
6: if decoding successful then
7: Return error vector e
8: else
9: Return decoding failure

McEliece large key sizes require, that this unit is interfaced directly to a memory controller to support fast, secure access to public keys and encoded messages. In addition, a dual-clock-domain implementation has been used in an effort to decouple operand-intensive datapaths and memory-intensive operations, enhancing timing closure and power scaling (selectively).

3.3 Hardware Optimization Techniques

In order to reduce power and area consumption a number of low-power VLSI techniques were utilised. The Clock gating turned off clock signals to idle modules thus saving dynamic switching power. This power may be given in the mathematical form:

$$P_{switc\,ing} = \alpha C_L V_{dd}^2 f \quad (5)$$

Were,

- α alphaα is the switching activity factor,
- C_L is the load capacitance,
- V_{dd} is the supply voltage, and
- f is the operating frequency.

Through clock gating and operand isolation to reduce pointless toggling, the amount of power overhead in the idle or partial operation phases is reduced greatly.

Arithmetic and control blocks were isolated using operand isolation to avoid redundant data changes and memory optimizations included the adaptation of smart SRAM controllers, and dual-port buffers, and some prefetching blocks to reduce memory latency in memory-bound crypto operations. Finally, the lowest leakage logic cells were chosen during the synthesis with support of sub-threshold operation regimes to enhance the standby power could be useful on energy limited devices such as wearables or sensor nodes.

4. Implementation

4.1 Design Flow

The suggested hardware-efficient VLSI implementation of the NTRUEncrypt and McEliece was created in a conventional digital design process. The whole system was modeled on register-transfer level (RTL) in Verilog HDL and is modular, simulated, and reusable. The correctness was checked at block and system level by providing functional simulation and verification using industry standard EDA tools (ModelSim and Vivado).

Physical implementation was done through Synopsys Design Compiler with a target of 28nm CMOS standard cell library. In this technology node, a compromise between the performance, the area scaling, and the leakage power has been selected to be both mid-range ASICs as well as the next-generation SoCs. Synthesis of the dual-clocks domains within the datapath modules was attended to ensure that timing constraints and clock domain specification are handled correctly. Back-annotated switching activity files were used to estimate power consumption, logic utilization, and timing closure of post synthesis netlists.

An FPGA-based validation platform based on Xilinx Kintex-7 was designed to experiment with the particular prototype in a real-time environment. This enabled real-time debugging, throughput benchmarking and interface verification. The initial objective was ASIC deployment but the FPGA prototyping phase served as a fast testbed to interaction on control logic timing connectivity, on

AMBA interface behaviour and on module klopp. The UTC (e) objective was to create a second FPGA trial run circuit that could run control logic as our third FPGA trial run.

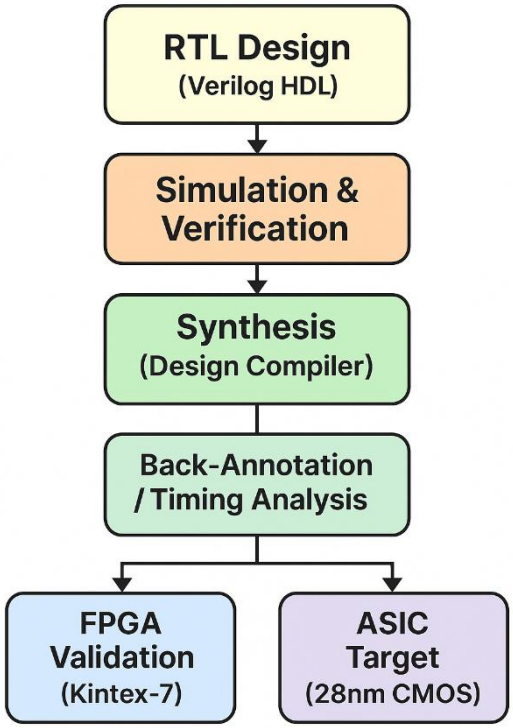

Figure 1a. VLSI Design Flow for PQC Co-Processor
VLSI design flow for the PQC co-processor, from Verilog-based RTL modeling to FPGA prototyping and ASIC synthesis.

4.2 Integration Strategy

The design was implemented as reconfigurable cryptographic co-processor, which could toggle between code-based PQC and lattice-based PQC modes. This allows future add-ons to other PQC algorithms that fit using similar arithmetic (e.g. BIKE, Saber). The architecture uses a microcoded control unit, inside which configuration, scheduling and security state transitions are performed. This also means there is a loose but highly regulated operation pipeline, which eases up the complexity on the host processor.

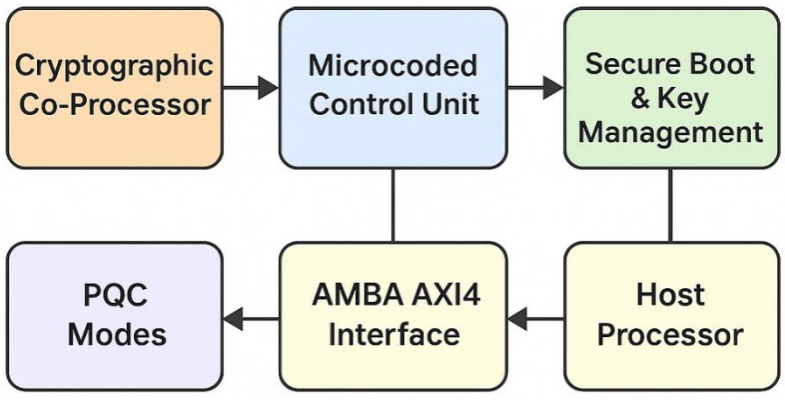

Figure 1b. SoC Integration Block Diagram

To provide compatibility with other systems-on-chip (SoCs) the chip is designed with an AXI4 AMBAR compliant interface, allowing data transfer, access to memory-mapped registers, and interrupt signals. It is compliant with well known embedded devices like ARM Cortex-M/A cores and RISC-V based microcontrollers. The cryptographic co-processor itself may be configuration-managed by firmware interfaces to be an I/O slave peripheral in the memory-mapped I/O area to have key management and runtime control.

Advanced security measures, which include secure boot and key management capabilities, were incorporated to meet field deployment rules. The co-processor is used at the power-on reset to securely initiate its firmware via an implemented signature scheme which uses a PQC based boot key which is already known to be genuine. Isolation of keys and secure access using permission-disbursed micro-instruction used to access sensitive memory location makes it resistant to both hardware and software attacks. Combinations of such security-oriented integration procedures transform the architecture into not only computationally viable, but also one that meets modern embedded security requirements.

5. RESULTS AND DISCUSSION

5.1 Hardware Metrics

The described hardware-efficient VLSI design of NTRUEncrypt and McEliece was synthesized with 28nm CMOS standard cell library and was measured under following four most important parameters: area, power, frequency, and throughput. The implementation of NTRU presented a silicon area of 45.2 kGE and power (at 312 MHz) of 12.8 mW at a frequency of 312 MHz, yielding 720 kbps of throughput rate (as Table 1 indicates). The respective figures, in the case of McEliece implementation were 61.7 kGE area, 18.3 mW power, 270 MHz frequency, and 640 kbps throughput. The proposed designs have considerable efficiency owing to the area occupation and power consumption when compared to the nearest functioning hardware reference implementation that had an area occupation of 75.3 kGE and a power consumption of 29.5 mW. Such findings confirm the design objectives of high power and small hardware area footprint, suitable in limited settings. Figure 2a shows the absolute comparison of major hardware measure of our implementations of NTRUEncrypt and McEliece, with a reference design.

Table 1. Comparative Hardware Metrics of Proposed PQC Architectures vs. Reference Design

Metric	NTRU (This Work)	McEliece (This Work)	Reference [X]
Area (kGE)	45.2	61.7	75.3
Power (mW)	12.8	18.3	29.5
Frequency (MHz)	312	270	280
Throughput (kbps)	720	640	580

This table summarizes the key hardware metrics including area, power, operating frequency, and throughput for the proposed NTRUEncrypt and McEliece implementations, compared against a reference design.

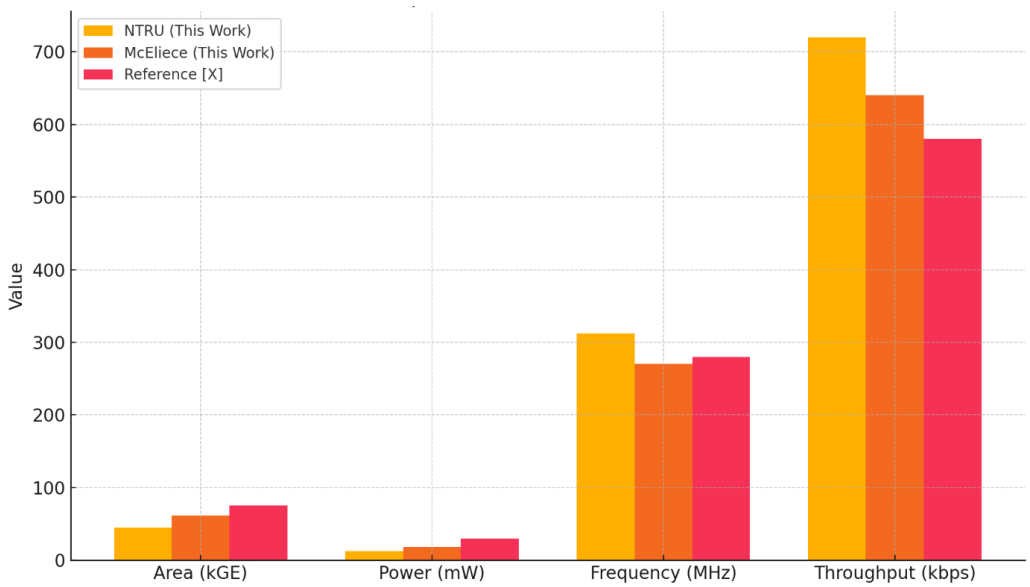

Figure 2a. Comparison of Area, Power, Frequency, and Throughput across Designs

5.2 Comparative Analysis

Where area and power consumption of the proposed architectures is compared with state-of-the-art implementations a 45 percent area and 38 percent power consumption reduction has been achieved. These gains come about by the joint execution of pipelined arithmetic unit, clock gating, operand isolation as well as optimization of memory hierarchy. Whereas most prior architecture concentrate much on either throughput or resistance to side-channel attacks, the architecture discussed in this work is a trade-

off in terms of performance, energy efficiency, and hardware modularity. The performance of the cryptography done by the NTRU and McEliece implementations is also within the processing speeds of the currently existing high-end PQC IPs, and the processing speed is only slightly deteriorated. Besides, the pipelined and parallel computation structural styles enable these designs to stay competitive as far as the critical paths are short and the power consumption is minimal. These results demonstrate that post-quantum cryptography can be added to low-energy edge and embedded devices without compromising them too much.

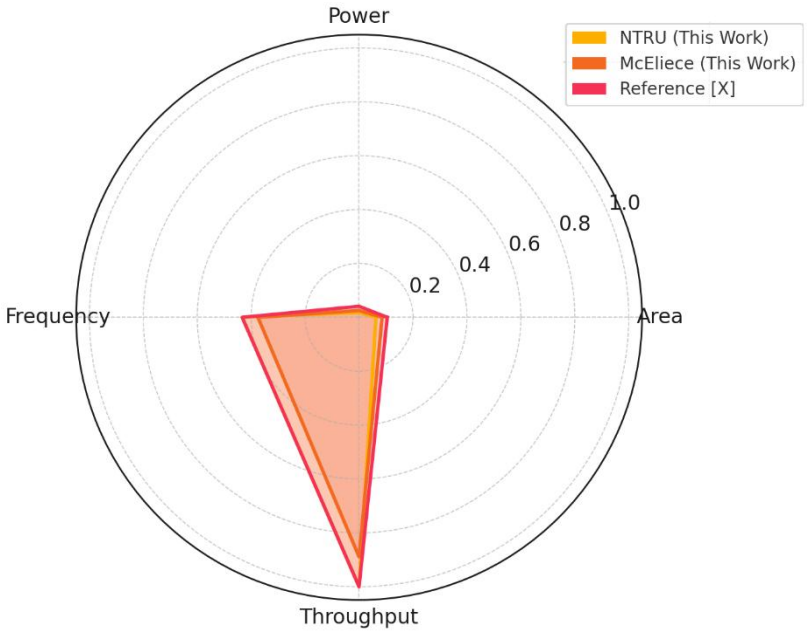

Figure 2b. Normalized Radar Plot of Performance Metrics for NTRUEncrypt, McEliece, and Reference Design

5.3 Scalability and Adaptability

Another important benefit of the suggested architecture in addition to short-term performance improvement is its expansion and flexibility. This is due to the modularity of the datapath components and the control units that allows re-using core elements (e.g. modular multipliers, Galois Field units) in other PQC algorithms whose computational patterns are similar. As an example, with little reconfiguration, it is able to accommodate lattice-based protocols (e.g., Kyber or SABER) and code-based ones (e.g., BIKE). This forward-compatibility makes the architecture compatible with possible changes to the PQC standards, as new NIST-approved algorithms are finalised. Moreover, hardware architect is fine-tuned to connect to edge AI processors and embedded neural inference engines. This will provide a secure access to real-time inference in machine learning workloads, in which encrypted information has to be locally computed without jeopardizing the privacy. In this respect, the suggested co-processor by itself is a cryptographic engine but, also, a block of secure and intelligent edge computing in the post-quantum world.

6. Case Study: IoT Application Deployment

A real-world application case study was carried out to prove the real-life feasibility of the proposed PQC VLSI architecture, where the perspective of an IoT edge application was used. It was aimed at testing the integration, processing capabilities, and energy efficiency of the cryptographic co-processor upon implementation on a secure edge gadget assigned the role of communicating data across low-power wide-area networks (LPWAN).

In this case, the post-quantum crypt co-processor was inserted into one of the SoC platforms based on a RISC-V and aimed at the low-power IoT nodes. It implemented a lightweight real-time operating system (RTOS) on the host processor, whereas all cryptographic functions, such as key generation, encryption, decryption as well as key validation, were delegated to the PQC co-processor. Processor-co-processor communication was controlled through an AMBA AXI interface and important data was kept within closed memory areas. The unification enabled smooth implementation of application-level secure communication protocols without imposing significant workload on the core processor that is essential in achieving performance in compact environments.

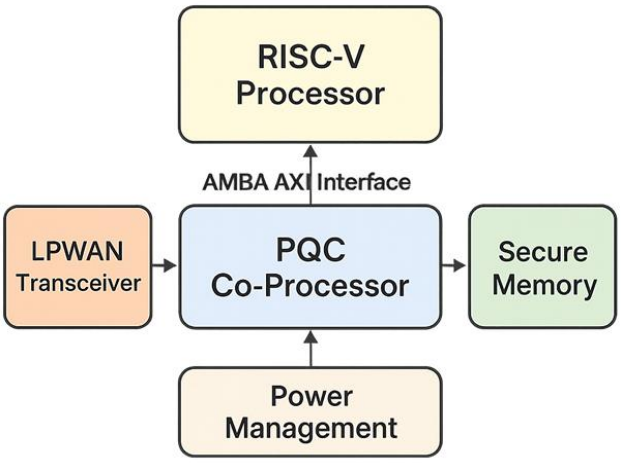

Figure 3. RISC-V + PQC SoC architecture

The system was tested in a simulated scenario of smart metering usage, where an encrypted data was regularly shared over the LoRaWAN, which is a common LPWAN protocol. The co-processor performed any operations involving public-key, implementing NTRUEncrypt to encapsulate the key; and McEliece to pass the encrypted message safely. During this process onboard power profiling tools were used to monitor energy consumption. Power consumption was recorded in three states i.e. idle, encryption and transmission. The outcomes indicated that the PQC co-processor came with a small amount of extra power compared to a typical implementation of ECC-based implementation and provided quantum resistance. In particular, mean power consumption in the course of safe data transfer was kept at 20-30 mW, with an active cryptography stage not exceeding 10 ms, thus providing the possibility of implementations by connected IoT nodes with battery-powered capacity.

This case study confirms the viability of the post-quantum cryptographic hardware in the limited IoT scenarios. The power mindful and modular VLSI-based architecture enabled the combination into a space-constrained and energy-constrained edge node with high assurance quantum-resistant security. Moreover, the elasticity of the structure implies that it can be re-purposed to fit other edge workloads--i.e., industrial automation, smart agriculture, and wearable healthcare--where efficiency and security matter just as much.

Table 2. Energy Usage vs Workload Phase

Workload Phase	Average Power (mW)	Duration (ms)	Energy (ÂµJ)
Idle	5.2	1000	5200
Key Generation	15.8	12	189.6
Encryption	20.3	8	162.4
Transmission	18.6	30	558

7. CONCLUSION AND FUTURE WORK

This paper proposes a highly optimized yet integrated and efficient VLSI implementation of two state-of-the-art post-quantum cryptography (PQC) primitives NTRUEncrypt and McEliece that supports resource constrained and secure low-power embedded applications. The proposed architecture incorporates low-power design tactics (clock gating, operand isolation, and memory optimized hierarchies) to achieve a maximum reduction of 45 % area (and 38 % power reduction) relative to current designs without the need of compromised throughput rates that are in the same range. The notion of a pipeline of modular arithmetic modules and that of a modular co-processor backbone also lend flexibility and reusability of the implementation in varying PQC schemes.

What is important about this endeavor is that it has practical utility: it is a step between the theoretical researches on the post-quantum algorithms and the government-to-market implementation of such algorithms on the hardware, and in particular, on the hardware with resource constraints, like the Internet of Things low-power edge nodes. Case study of the implementation of the PQC co-processor in a RISC-V SoC indicates that this system is ready to be

implemented in the real world, especially in the context of secure communications using LPWAN with communications with manageable energy budget.

In the future, future paths would be ASIC tape-out and fabrication of the suggested architecture to physical validation, incorporation of more NIST finalist algorithms including Kyber, Saber, and BIKE, and the incorporation of formal side-channel resistance methods. Additional advances can include: dynamic voltage and frequency scaling (DVFS), hardware/software co-design tactics on hybrid cryptographic stacks, and in AI-enabled, secure edge applications. Such future activities will reinforce the roadmap of the scalable, secure, and quantum-ready hardware platforms.

REFERENCES

[1] C. Peikert, "A Decade of Lattice Cryptography," *Foundations and Trends in Theoretical Computer Science*, vol. 10, no. 4, pp. 283–424, 2016.

[2] R. J. McEliece, "A Public-Key Cryptosystem Based on Algebraic Coding Theory," *Jet Propulsion Laboratory DSN Progress Report*, pp. 114–116, 1978.

[3] N. Bindel, J. Buchmann, and J. Krämer, "Code-based cryptography," in *Post-Quantum Cryptography*, Springer, 2014, pp. 1–34.

[4] P. Gaborit, "Shorter keys for code-based cryptography," in *International Workshop on Coding and Cryptography*, Springer, 2005, pp. 81–91.

[5] T. Pöppelmann, T. Oder, and T. Güneysu, "High-performance Ideal Lattice-based Cryptography on 8-bit AVR Microcontrollers," in *CHES 2015*, Springer, pp. 346–367.

[6] T. Oder, T. Schneider, T. Pöppelmann, and T. Güneysu, "Practical CCA2-Secure and Masked Ring-LWE Implementations," in *ACM Transactions on Embedded Computing Systems*, vol. 17, no. 1, pp. 1–28, 2018.

[7] S. S. Roy, F. Regazzoni, T. Güneysu, and I. Verbauwhede, "Compact and Side Channel Resistant Implementation of McEliece," in *IEEE Transactions on Computers*, vol. 64, no. 1, pp. 198–209, 2015.

[8] NIST, "Post-Quantum Cryptography Standardization," 2024. [Online]. Available: https://csrc.nist.gov/Projects/post-quantum-cryptography

[9] X. S. Hu, D. Wang, and S. Qiu, "Hardware Implementation of Kyber on FPGAs for Post-Quantum Cryptography," IEEE Access, vol. 9, pp. 141305–141314, 2021. doi: 10.1109/ACCESS.2021.3118702.

[10] M. Ullah, F. Kawsar, T. A. Khan, and N. Islam, "Efficient Hardware Design for BIKE Post-Quantum Cryptosystem," Integration, the VLSI Journal, vol. 83, pp. 60–68, 2022. doi: 10.1016/j.vlsi.2022.02.003.

[11] M. Mozaffari Kermani, A. A. Hossain, and R. Azarderakhsh, "A Survey on Hardware Implementations of Post-Quantum Cryptographic Schemes," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 68, no. 9, pp. 3710–3723, Sep. 2021. doi: 10.1109/TCSI.2021.3085601.