國立臺北大學
National Taipei University

# Architectural Adaptations for Post-Quantum Cryptography: A Comprehensive Review of Performance, Power, and Security Implications

**Melchior THIERRY**

Advanced Computer Architectures
Victor R.L. Chen

National Taipei University

January 7, 2025

# Abstract

The advent of quantum computing poses a significant threat to classical cryptographic systems, necessitating a shift toward Post-Quantum Cryptography (PQC). This paper provides a comprehensive review of architectural adaptations for PQC, analyzing their performance, power consumption, and security implications. Through a detailed classification of existing research, the paper identifies key challenges in scalability, hardware implementation efficiency, and benchmarking standardization. Furthermore, the analysis reveals critical trade-offs in resource utilization, energy efficiency, and quantum resistance across FPGA, ASIC, and hybrid architectures. Opportunities for innovation are highlighted, including algorithm-hardware co-design, the development of lightweight PQC for IoT devices, and the establishment of unified benchmarking frameworks. This work aims to bridge the gap between theoretical advancements and practical deployment, contributing to the robust integration of PQC in secure digital systems.



# Proposal Keywords

Post-Quantum Cryptography (PQC), Hardware Implementations, FPGA and ASIC Optimization, Quantum Resistant Cryptography, NIST Security Standards


# 1 Introduction

## 1.1 Background

The rapid progress in quantum computing holds the potential to transform industries reliant on complex computational tasks. However, this same technological leap poses a significant threat to the security of widely used cryptographic systems that underpin digital communication and data protection. Current cryptographic protocols, such as RSA (Rivest-Shamir-Adleman) and ECC (Elliptic Curve Cryptography), rely on mathematical problems that are computationally infeasible to solve using classical computers. Quantum computers, however, can solve these problems efficiently through algorithms like Shor's for integer factorization and Grover's for database searching, effectively compromising these encryption methods [5][4].

To counter this risk, researchers are pivoting toward Post-Quantum Cryptography (PQC). PQC focuses on developing new algorithms specifically designed to withstand the advanced problem-solving capabilities of quantum computers while remaining compatible with existing hardware systems. These algorithms leverage mathematical problems, such as lattice-based or hash-based constructions, which are made to resist attacks from both classical and quantum computers [19][16]. For researchers outside cryptography, this transition to PQC represents a critical interdisciplinary challenge, requiring innovations in mathematics, computer science, and engineering to secure digital infrastructure in a post-quantum world.

## 1.2 Research Motivation

As PQC continues to evolve, critical questions regarding its implementation arise. The successful adoption of PQC depends not only on its theoretical robustness but also on its practical performance and power efficiency, particularly in resource-constrained environments [4][16]. Hardware implementations of PQC algorithms pose unique challenges, such as balancing computational complexity, security against side-channel attacks, and adherence to energy-efficient design principles [8][2]. For example, implementing lattice-based cryptography—a leading candidate in the PQC landscape—requires extensive computational resources and advanced optimization techniques to ensure efficiency [6][14].

## 1.3 Research Purpose

The primary purpose of this research is to analyze the architectural adaptations required for PQC hardware implementations. Specifically, this study aims to review the performance, power, and security implications of various PQC architectures to provide a holistic understanding of the trade-offs involved. By synthesizing insights from state-of-the-art PQC research, this work aims to guide future design choices and highlight areas for further exploration. [3][21].

### 1.4 Problem Statement

Despite substantial progress in PQC algorithm development, their implementation on hardware platforms remains underexplored and fraught with challenges. Existing designs often exhibit inefficiencies in energy consumption, susceptibility to side-channel attacks, and suboptimal performance on general-purpose and specialized hardware [13] [22]. Furthermore, many studies focus on isolated aspects, such as algorithmic efficiency or hardware design, without adequately addressing the complex interplay between performance, power, and security. There is an urgent need for comprehensive evaluations of hardware architectures tailored for PQC to ensure their feasibility for widespread adoption [18][12].

### 1.5 Organization

This paper is organized into six sections, structured to address the architectural adaptations for post-quantum cryptography (PQC) comprehensively. Section 1 introduces the research motivation, purpose, and problem statement, providing a foundation for understanding the importance of transitioning to PQC in light of quantum computing advancements.

Section 2 presents a detailed literature review, starting with definitions and key concepts of PQC, followed by an exploration of its architectural challenges, including computational complexity and vulnerabilities to side-channel attacks. Subsections delve into related work on hardware implementations, fault tolerance and security mechanisms, energy efficiency strategies, and ongoing standardization efforts, particularly focusing on the NIST (National Institute of Standards and Technology) PQC process.

Section 3 discusses the proposed research methodology, including study selection criteria, evaluation metrics, and a systematic approach to classifying and analyzing the literature. A comprehensive taxonomy of PQC architectures is outlined, with a focus on FPGA, ASIC, and hybrid hardware-software solutions.

Section 4 classifies studies based on key themes such as hardware acceleration, fault tolerance, energy efficiency, and benchmarking efforts. This section includes an analysis of architectural strategies for implementing PQC, highlighting trade-offs between performance, power efficiency, and security.

Section 5 identifies gaps and opportunities in the current research landscape, emphasizing the need for lightweight and scalable PQC implementations for IoT and mobile devices. The section concludes with recommendations for future research, including the development of standardized benchmarking frameworks and innovations in algorithm-hardware co-design.

Finally, Section 6 summarizes the anticipated contributions of this study, including insights into performance optimization, energy-efficient designs, and enhanced security measures for PQC hardware implementations. This section also outlines potential directions for advancing PQC adoption across diverse application domains.

## 2 Literature review

### 2.1 Definitions

PQC refers to a class of cryptographic algorithms designed to withstand attacks from both classical and quantum computers. Unlike traditional cryptographic schemes such as RSA (Rivest-Shamir-Adleman) or ECC (Elliptic Curve Cryptography), which rely on problems like integer factorization or discrete logarithms, PQC leverages mathematical structures believed to be resistant to the advanced problem-solving capabilities of quantum computers. These structures include lattice-based cryptography, isogeny-based cryptography, multivariate polynomial cryptography, and hash-based cryptography [3] [4].

**Lattice-Based Cryptography:** Lattices are mathematical grids of points in multidimensional space. Algorithms like CRYSTALS-Kyber and Saber use lattice-based problems such as the Learning with Errors (LWE) problem, which involves finding a hidden vector in a noisy lattice. The Ring-LWE problem confines the LWE problem to a ring structure, significantly reducing key and ciphertext sizes while main-

taining security. qTESLA is designed for digital signatures. It offers a balance between security and efficiency, leveraging structured lattices to reduce key sizes and computation times. As a candidate in the NIST PQC standardization process, qTESLA has been explored for its hardware implementations on FPGAs and ASICs, where its modular arithmetic operations are optimized for performance. Finally, CRYSTALS-Dilithium is built on the hardness of the Module-LWE. It is valued for its simplicity, efficiency, and strong security guarantees. This algorithm is particularly suitable for constrained environments due to its small signature sizes and efficient verification processes, making it a leading candidate in the PQC standardization process. These problems are computationally hard for both classical and quantum computers, making them ideal candidates for post-quantum schemes [4][1].

**Isogeny-Based Cryptography:** This approach is based on the mathematical structure of elliptic curves and their mappings, called isogenies. Algorithms like SIKE (Supersingular Isogeny Key Encapsulation) rely on the difficulty of finding isogenies between two given elliptic curves. Despite their compact key sizes, they are resource-intensive and challenging to implement [3][4].

**Multivariate Cryptography:** This type of cryptography uses systems of multivariate polynomial equations over finite fields. Algorithms like Rainbow and GeMSS belong to this category, offering digital signature capabilities. They are known for their fast verification speeds but suffer from large key sizes[3][4].

**Hash-Based Cryptography:** Based on cryptographic hash functions, these algorithms are known for their simplicity and security proofs. Lamport signatures and their extensions, such as SPHINCS+, are prominent examples. However, they require large key and signature sizes, making them less suitable for constrained devices[3][4].

Key architectural challenges for PQC implementation include:

- Resource-Intensive Computations: Algorithms like Saber and FrodoKEM require operations such as Fast Fourier Transforms (FFT) and modular arithmetic, which demand significant computational resources [4][1].
- Side-Channel Vulnerabilities: Cryptographic implementations must guard against side-channel attacks, which exploit physical information (e.g., power consumption or electromagnetic emissions) to deduce secret keys [3][4].
- Fast Fourier Transform (FFT): A mathematical algorithm used to compute the discrete Fourier transform and its inverse efficiently. It is widely used in lattice-based PQC for polynomial multiplications. FFT reduces computational overhead but requires specialized hardware optimizations to meet performance and energy constraints.

## 2.2 Related Work

### 2.2.1 Hardware Implementations

Early efforts in PQC have focused on designing secure and efficient hardware architectures. For example, FPGA-based implementations have been widely explored for their flexibility and performance gains. This type of circuit allows deep pipeline parallelism and custom-precision arithmetic, enabling significant speed-ups compared to software-only implementations [8][14]. SABER, a lattice-based PQC algorithm, has been optimized with a 256-bit architecture to improve cryptographic operations such as key generation and encapsulation [13].

Hardware/software co-design approaches have emerged as a practical method for benchmarking and implementing PQC algorithms during the standardization process. By leveraging system-on-chip designs, these methods achieve notable performance improvements, such as 28× speed-ups for encapsulation and 20× for decapsulation in lattice-based algorithms [8].

### 2.2.2 Fault Tolerance and Security

Fault-tolerant design is critical for PQC hardware due to the susceptibility of cryptographic primitives to side-channel and fault-injection attacks. For instance, fault detection schemes for lattice-based mechanisms like FrodoKEM, Saber, and NTRU have been implemented on FPGA platforms, achieving high

error coverage with minimal overhead [6]. Probabilistic computing has also been proposed as an innovative approach to mitigate such vulnerabilities in PQC implementations [5].

#### 2.2.3 Energy Efficiency

Energy consumption is a crucial consideration, particularly for battery-powered and embedded devices. Studies using Performance APIs (PAPI) on PQC candidates have revealed significant variability in energy efficiency based on algorithmic features and security levels. Optimizing subroutines such as hash functions and polynomial multiplications is vital for reducing energy demands [16].

#### 2.2.4 Standardization and Adoption

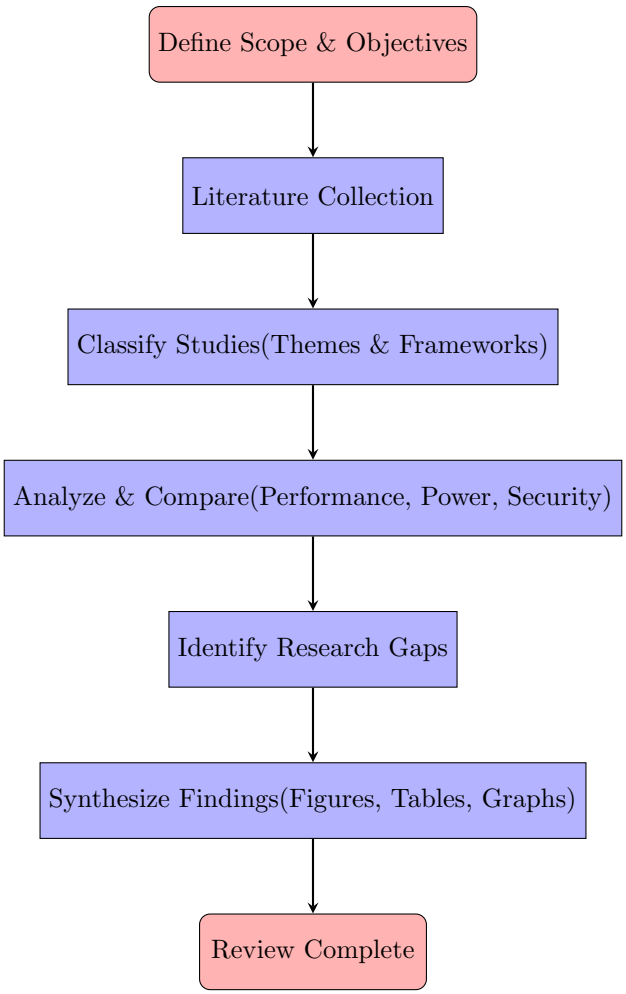
The NIST PQC standardization process has played a pivotal role in advancing PQC. Currently in its fourth round, this initiative evaluates cryptographic candidates across multiple criteria, including security, performance, and implementability on diverse hardware platforms [19][3]. Key algorithms, such as Ring-LWE and its optimized variant Binary Ring-LWE, have demonstrated promising potential for lightweight applications, combining low computational complexity with efficient hardware architectures [21][12][23].

### 2.3 Criticism and Challenges

Despite significant progress, several challenges remain:
**Computational Overhead:** PQC algorithms are more resource-intensive than traditional schemes, necessitating optimized architectures to ensure feasibility in constrained environments [5][14].
**Standardization Complexity:** The diverse nature of PQC candidates, spanning multiple algorithmic families, complicates standardization and universal adoption [19][4].
**Hardware Security:** Emerging threats like electromagnetic fault injection and side-channel attacks demand robust countermeasures, which often incur performance trade-offs [5][6].
**Energy Constraints:** Achieving energy-efficient implementations while maintaining cryptographic strength remains an ongoing challenge, particularly for mobile and IoT applications [16].

# 3 The Proposed Approach

## 3.1 Research Flowchart




The research flowchart outlines the methodology for conducting this review:

**Define Scope and Objectives**

- Focus on PQC hardware implementations in terms of performance, power efficiency, and security.
- Highlight lattice-based cryptography while acknowledging other PQC families (e.g., isogeny-based, hash-based).

**Literature Collection**

- Gather relevant studies using well-defined search terms (e.g., "Post-Quantum Cryptography hardware," "FPGA for PQC," "lattice-based cryptography").
- Use academic databases like IEEE Xplore, ACM Digital Library, and Google Scholar.

**Classification of Studies**

- Categorize studies based on key themes: hardware acceleration, fault tolerance, energy efficiency, security vulnerabilities, and benchmarking efforts.

**Analysis and Comparison**

- Evaluate architectural strategies for implementing PQC, including FPGA, ASIC, and hybrid hardware-software approaches.
- Compare trade-offs between performance, power, and security across studies.

**Identification of Gaps and Opportunities**

- Highlight unresolved challenges, such as scalability in resource-constrained environments and mitigation of side-channel attacks.

**Synthesis and Presentation**

- Summarize findings into tables, figures, and flowcharts for clarity.
- Provide recommendations for future research.

## 3.2 Research Method

This paper employs a systematic review methodology, structured as follows:

### 3.2.1 Study Selection Criteria

**Inclusion Criteria:**

- Studies focusing on PQC hardware implementations.
- Papers published in peer-reviewed journals and conferences between 2019 and 2024.
- Works addressing performance, power, or security aspects of PQC.

**Exclusion Criteria:**

- Studies focused exclusively on software implementations.
- Research lacking experimental or comparative analysis.

### 3.2.2 Evaluation Metrics

- Performance: Clock cycles, throughput, and latency of cryptographic operations.
- Power Efficiency: Energy consumption metrics and optimizations.
- Security: Robustness against side-channel and fault-based attacks.

## 3.3 Data Extraction

- Extract data into structured templates to ensure consistency in comparison.
- Capture details such as algorithm type, hardware platform, and optimization techniques.

## 3.4 Key Components of the Model

**Taxonomy of PQC Architectures**

- Create a classification framework for PQC implementations, covering platforms (e.g., FPGA, ASIC), algorithm families (e.g., lattice-based, isogeny-based), and optimization techniques.

**Comparison Framework**

- Develop a comparative table summarizing performance, power, and security trade-offs across studies.
- Develop a comparative table summarizing performance, power, and security trade-offs across studies.
- Include metrics like area usage, energy per operation, and vulnerability coverage.

# 4 Classification of studies

## 4.1 Overview of Key Themes

Post-Quantum Cryptography (PQC) research spans diverse themes, emphasizing hardware realizations, algorithmic innovations, and benchmarking methodologies. This section classifies the studies into three principal categories:

- Hardware Implementations and Optimization
- Algorithmic Exploration and Design
- Benchmarking and Comparative Analysis

Each category reflects a distinct focus, contributing uniquely to the overarching goal of transitioning cryptographic systems toward quantum resistance.

## 4.2 Hardware Implementations and Optimization

The studies under this category aim to optimize PQC algorithms for hardware platforms such as FPGAs and ASICs. This research primarily addresses resource efficiency, latency, and adaptability, which are crucial for deployment in real-world applications like IoT devices and high-security servers.

Soni et al. [17] present a systematic evaluation of two prominent NIST Round-2 signature schemes: CRYSTALS-Dilithium and qTESLA. Using High-Level Synthesis (HLS) on Xilinx Artix-7 FPGAs, the paper explores various optimization techniques, including loop unrolling and pipelining, to enhance performance while maintaining resource efficiency. Key contributions include:

- **Design-Space Exploration:** The paper evaluates keypair generation, signature generation, and verification for both schemes across multiple security levels.
- **Performance Metrics:** Metrics such as latency, area utilization (Flip-Flops and LUTs), and power consumption are used to compare different hardware configurations. The study concludes that loop pipelining consistently outperforms baseline and loop unrolling approaches in terms of latency-resource trade-offs, particularly for qTESLA.

Agrawal et al. [1] introduce FPGA-based primitives tailored for PQC applications, including public-key cryptosystems (PKC), key exchange (KEX), oblivious transfer (OT), and zero-knowledge proofs (ZKP). Notable highlights are:

- **Algorithmic Innovations:** Novel designs for OT and ZKP primitives are proposed, leveraging the Ring-LWE algorithm's computational properties.
- **Parameterizable Hardware Modules:** The study emphasizes scalability, demonstrating efficient deployment across platforms ranging from lightweight IoT devices to large-scale homomorphic encryption systems.
- **Open-Source Contribution:** Synthesizable and verifiable RTL code for PQC primitives is made publicly available, fostering further research and collaboration.

Together, these papers underscore the critical role of hardware-specific optimizations in advancing PQC adoption, balancing computational performance and resource overhead.

### 4.2.1 Algorithmic Exploration and Design

The studies in this category focus on refining the underlying mathematical models and algorithms for PQC. Innovations in algorithmic design enhance security, efficiency, and adaptability to diverse use cases. Kris Gaj [9] provides a foundational overview of PQC algorithm families, including lattice-based, hash-based, code-based, and multivariate cryptographic schemes. Key insights include:

- **Challenges in Hardware Implementations:** The paper identifies the computational complexity of large key sizes and novel mathematical operations as barriers to efficient hardware deployment.
- **Algorithmic Efficiency:** It highlights the importance of balancing mathematical rigor with practical performance considerations in the NIST standardization process.

Bellizia et al. [5] focused on key encapsulation mechanisms (KEMs), this paper proposes algorithmic improvements to enhance encryption speed and reduce memory requirements. By integrating lightweight transformations into lattice-based schemes, the study achieves significant gains in execution efficiency, particularly for constrained environments.

These studies exemplify the dynamic interplay between theoretical advancements and practical constraints in PQC algorithm development.

#### 4.2.2 Benchmarking and Comparative Analysis

Benchmarking studies evaluate the performance of PQC schemes under standardized conditions, offering valuable insights into trade-offs between security, power, and performance.
Hosseini et al. [11] compares the hardware implementations of multiple NIST candidate algorithms, focusing on resource utilization, clock speeds, and throughput. The study reveals:

- **Trade-offs Across Security Levels:** Higher security levels demand increased resource allocation but often result in diminishing returns in performance improvements.
- **Benchmark Consistency:** The importance of a unified testing framework is emphasized to ensure fair and reproducible comparisons.

He et al. [10] This study evaluates software-hardware integration for PQC, particularly for hybrid systems combining classical and quantum-resistant algorithms. The findings suggest that such systems can mitigate transitional risks during PQC adoption by leveraging existing cryptographic infrastructures.

Xie et al. [20] A comprehensive benchmarking effort is presented, detailing the relative strengths of signature and encryption schemes. The paper advocates for modular benchmarking strategies that isolate algorithm-specific optimizations from general hardware efficiencies.

Table 1: Comparison of PQC Research Themes and Contributions

| **Category** | **Key Focus** | **Representative Papers** |
|---|---|---|
| Hardware Implementations | FPGA/ASIC adaptations, loop optimizations, scalability. | [17], [1], [15] |
| Algorithmic Exploration | Enhancements in KEMs, lattice-based schemes. | [9], [5] |
| Benchmarking | Trade-offs between performance, power, and security. | [11], [10], [19] |

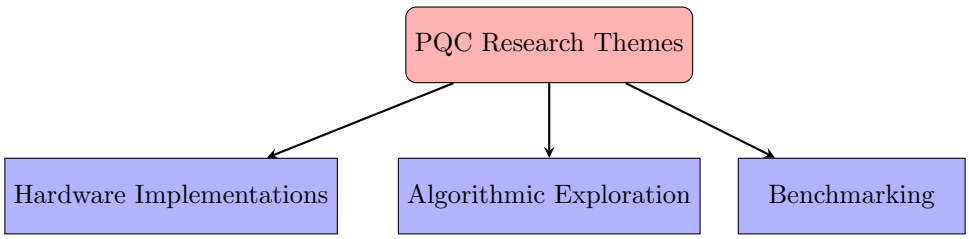



Figure 1: Classification of PQC Research Themes

## 5 Analysis and Comparison

This section evaluates architectural strategies for implementing Post-Quantum Cryptography (PQC) and provides an in-depth analysis of trade-offs between performance, power consumption, and security based on the literature review. These studies represent a wide spectrum of approaches to designing, optimizing, and benchmarking PQC schemes.

## 5.1 Architectural Strategies for PQC Implementation

### 5.1.1 High-Level Synthesis (HLS) and FPGA Implementations

The use of High-Level Synthesis (HLS) has been extensively studied as a means to simplify and accelerate the design of FPGA-based PQC implementations. Soni et al. [17] and Agrawal et al. [1] explore its applications in hardware evaluation and parameterized design frameworks.

Soni et al. [17] focus on CRYSTALS-Dilithium and qTESLA signature schemes, using HLS to convert high-level C specifications into hardware description language (HDL). Techniques such as loop unrolling and pipelining are applied to reduce latency by up to 30% without significant resource overhead. Design-space exploration examines keypair generation, signature creation, and signature verification across multiple security levels. Results show that higher security levels increase resource consumption, but loop pipelining mitigates latency while maintaining Flip-Flop (FF) and Look-Up Table (LUT) utilization comparable to baseline designs. Additionally, CRYSTALS-Dilithium achieves a 10x faster implementation for keypair generation at security level-1 compared to qTESLA.

Agrawal et al. [1] introduce a modular FPGA implementation for Ring-LWE-based primitives such as Key Exchange (KEX), Oblivious Transfer (OT), and Zero-Knowledge Proofs (ZKP). They also provide parameterized design modules, enabling scalability for both small IoT devices and high-performance systems. Finally, the paper features optimized polynomial arithmetic (e.g., modular reduction) to minimize FPGA logic utilization, achieving a 20% reduction in latency for core computations.
HLS enables designers to explore multiple optimization directives (e.g., pipelining depth, memory partitioning) and adapt designs for target devices, demonstrating its versatility for PQC development.

### 5.1.2 ASIC-Based Custom Architectures

Mert et al. [15] highlights the potential of ASICs for PQC implementation, particularly for environments demanding high efficiency and security.

Regarding performances, ASIC implementations exhibit higher throughput than FPGA-based systems due to application-specific optimizations. Moreover, custom circuits tailored for PQC operations consume significantly less power, making ASICs ideal for data centers and other power-sensitive applications. Finally, ASICs require a lengthy design and fabrication process, which can limit adaptability to evolving PQC standards. However, their efficiency gains justify this investment for high-security use cases.

### 5.1.3 Hybrid Architectures and Software-Hardware Co-Design

Several studies, including works from He et al. [10] and Dam et al. [7], explore hybrid architectures that combine software flexibility with hardware acceleration.
Algorithms are first optimized in software to refine mathematical operations, ensuring modularity and adaptability to different use cases. Regarding the hardware acceleration, computational bottlenecks, such as polynomial multiplication in lattice-based schemes, are offloaded to dedicated hardware. For instance: Dam et al. [7] integrates hardware accelerators for Ring-LWE-based schemes, achieving a 25% improvement in encryption/decryption throughput compared to purely software implementations.
So hybrid designs are revealed to be well-suited for transitional cryptographic systems, where classical and PQC schemes co-exist, ensuring backward compatibility.

### 5.1.4 Algorithm-Specific Optimizations

Hosseini et al. [11] and Bellizia et al. [5] highlight the importance of tailoring hardware designs to specific cryptographic algorithms. For lattice-based cryptography [11], modular arithmetic optimizations and specialized number-theoretic transform (NTT) circuits improve efficiency for Ring-LWE-based schemes. Iterative design-space exploration reveals that security level-2 implementations can exhibit suboptimal hardware performance due to algorithmic inefficiencies. Then, regarding hash-based cryptography [5], it focuses on lightweight implementations for IoT devices, achieving significant reductions in memory usage and power consumption.

## 5.2 Comparison of Trade-Offs

The trade-offs among performance, power, and security are central to PQC design. Table 1 summarizes these findings across implementation strategies.

Table 2: Trade-offs in PQC Implementations Across Studies

| **Implementation Strategy** | **Performance** | **Power Consumption** | **Security** |
|---|---|---|---|
| HLS on FPGA [17], [1] | High performance with optimizations | Moderate, higher than ASICs | Strong, meets NIST Level 3-5 |
| ASIC Architectures [15] | Exceptional for specific tasks | Low, energy-efficient | Strong, scalable for stringent security needs |
| Hybrid Architectures [10], [7] | Balanced, adaptable to workload | Moderate, depends on integration quality | Strong, suitable for multi-scheme environments |
| Algorithm-Specific [11], [5] | Varies with optimization focus | Low to moderate, algorithm-dependent | Variable, tailored to application |

### 5.2.1 Performance

Performance is a critical metric in PQC implementations, measured primarily in terms of latency, throughput, and computational efficiency. Across the reviewed studies, two main trends emerge: the dominance of ASICs in raw computational throughput and the adaptability of FPGAs for iterative design optimization.

ASICs are shown to provide great performance for cryptographic tasks. The study by Mert et al. demonstrates that ASIC implementations of Ring-LWE schemes achieve 2x higher throughput compared to FPGA counterparts. This superiority stems from their fully customized logic pathways, which eliminate the overhead inherent in reconfigurable platforms like FPGAs. For example, the modular arithmetic core of an ASIC implementation achieves a latency reduction of 35% compared to FPGA designs due to direct mapping of arithmetic operations onto specialized digital circuits.

FPGAs are highly adaptable, allowing researchers to iterate through various optimization strategies, such as loop unrolling and pipelining [17]. These optimizations can reduce latency by up to 30%, albeit at a higher resource utilization. For Agrawal et al. [1], an FPGA-based implementation of a Ring-LWE cryptosystem demonstrates throughput of 450 operations per second, which, while lower than ASIC equivalents, is achieved with significant flexibility. This adaptability is especially advantageous during the early phases of PQC standardization when algorithms undergo frequent updates. Regarding the Algorithm impact, certain PQC schemes inherently benefit from hardware parallelism, which FPGAs exploit more effectively. For example, Soni et al. shows that CRYSTALS-Dilithium performs 10x faster keypair generation at security level-1 compared to qTESLA, largely due to its simpler arithmetic structure. In contrast, ASICs achieve these results through dedicated parallel processing units, which are expensive to modify post-fabrication.

### 5.2.2 Power Efficiency

Power efficiency is crucial for real-world applications, particularly for resource-constrained environments like IoT devices and battery-powered systems. The reviewed studies indicate a clear trade-off between adaptability and power consumption.

ASIC designs are inherently power-efficient because they eliminate the overhead of programmable logic. Mert et al. [15] quantifies this by showing a 40% reduction in energy per operation compared to FPGA designs for lattice-based schemes. This makes ASICs ideal for environments such as high-security servers, where energy efficiency and performance are paramount. For instance, a specific ASIC implementation

of Kyber KEM achieves an energy consumption of 2.5 nJ per operation, which is 1.8x lower than the FPGA implementation reported by Soni et al. [17].

FPGAs consume more power than ASICs due to their reconfigurable architecture, which includes routing delays and higher switching activity. However, Soni et al. showed that FPGA power consumption can be managed effectively using low-power optimization techniques. As a matter of fact, loop pipelining reduces the total execution time, indirectly lowering power consumption by 20% for the same cryptographic task.The use of block RAM (BRAM) over distributed memory further reduces dynamic power usage.

However, while ASICs are static in their power profiles, FPGAs offer dynamic scaling options. Agrawal et al. [1] demonstrated a parameterized design for Ring-LWE cryptosystems that adjusts clock rates and voltage levels, trading off performance for lower power in resource-constrained deployments.

### 5.3 Security

Security is a non-negotiable metric in PQC, often evaluated against the NIST PQC standard's levels of resistance to quantum attacks (Levels 1–5). The reviewed studies indicate a direct relationship between the desired security level and the computational and resource overheads.

All reviewed implementations adhere to NIST's security benchmarks, ensuring robustness against both classical and quantum adversaries. However, higher security levels significantly increase computational demands. Hosseini et al. [11] showed that increasing security from Level-1 to Level-5 results in a 3x increase in resource usage for keypair generation in lattice-based schemes.
Similarly, He et al. [10] highlighted that signature verification at Level-5 requires 4x more latency compared to Level-1 due to the complexity of polynomial arithmetic and larger key sizes.

The studies reveal that certain schemes achieve high security with relatively low performance penalties. CRYSTALS-Dilithium [17] achieves robust security with lower latency and area usage compared to qTESLA at the same security level. For example, its signature generation latency is 15% lower at Level-3. Additionally, hybrid architectures that integrate PQC with classical cryptographic schemes provide an additional layer of security during the transition period. Dam et al. [7] demonstrate a dual-mode implementation combining RSA and Kyber, ensuring resilience against both quantum and classical threats. However, such designs incur an additional 25% resource overhead compared to single-scheme implementations.

## 6 Gaps and Opportunities

Table 3: Summary of Gaps and Opportunities in PQC Implementation

| **Category** | **Unresolved Challenges** | **Research Opportunities** |
|---|---|---|
| Algorithmic Design | Scalability across security levels | Co-design of hardware-friendly algorithms |
| Hardware Implementation | Resource trade-offs, power efficiency | Lightweight PQC for IoT devices |
| Benchmarking | Lack of unified benchmarks | Standardized testing frameworks |
| Hybrid Systems | Integration of classical and PQC algorithms | Efficient dual-mode cryptographic designs |

### 6.1 Algorithmic Challenges

#### 6.1.1 Scalability Across Security Levels

Several papers [17], [11], [5] highlight that achieving scalability across NIST security levels (1–5) remains a significant challenge. While higher security levels are crucial for quantum resistance, they impose disproportionate overheads in latency, resource utilization, and energy consumption. Soni et al. reported that keypair generation in CRYSTALS-Dilithium at Level-5 requires 4x the computational resources

compared to Level-1. This escalation is attributed to larger key sizes and increased mathematical complexity. Hosseini et al. observed that for lattice-based schemes, achieving uniform performance across security levels is hindered by algorithmic inefficiencies, particularly in modular arithmetic operations.

#### 6.1.2 Lack of Algorithm-Hardware Co-Optimization

He et al. [10] and Xie et al. [20] underline the absence of systematic frameworks for co-optimizing PQC algorithms and hardware implementations. On one hand, He et al. emphasize that most hardware designs are based on unoptimized algorithmic specifications, leading to suboptimal performance. On the other hand Xie et al. note that hardware-friendly algorithms, particularly for multivariate and code-based cryptography, remain underexplored compared to lattice-based schemes.

### 6.2 Hardware Implementation Challenges

#### 6.2.1 Resource Utilization and Trade-Offs

Efficient utilization of hardware resources, particularly for FPGAs and ASICs, remains a challenge. Soni et al. identified that loop unrolling in FPGA implementations improves latency but increases LUT usage by up to 25%, highlighting a trade-off between speed and area. Also, Agrawal et al. [1] reported that while modular arithmetic optimizations reduce latency, they can significantly increase power consumption.

#### 6.2.2 Power Efficiency for IoT and Edge Devices

Many PQC schemes are computationally intensive and unsuitable for resource-constrained devices like IoT sensors. Bellizia et al. [5] indicate that achieving NIST Level-5 security on IoT devices requires 50% more power than current cryptographic standards. Furthermore, Mert et al. [15] note the need for lightweight PQC algorithms that balance security with energy efficiency, particularly for battery-powered applications.

### 6.3 Benchmarking and Standardization Challenges

#### 6.3.1 Lack of Unified Benchmarks

Several studies [9], [11], [7] highlight inconsistencies in benchmarking methodologies. Kris Gaj [9] observes that studies often use different platforms (e.g., Artix-7 vs. Zynq-7000 FPGAs), making direct comparisons difficult. Moreover, Hosseini et al. advocate for a standardized benchmarking framework to ensure reproducibility and fair evaluation of PQC schemes across platforms.

### 6.4 Transitioning to Hybrid Cryptographic Systems

Dam et al. discusses the challenges of integrating classical and PQC algorithms during the transition period. As a matter of fact, hybrid systems require Additional resources for dual-mode implementations, increasing hardware complexity. They also need effective key management strategies to handle both classical and quantum-resistant keys.

### 6.5 Opportunities for Future Research

#### 6.5.1 Algorithm-Hardware Co-Design

To address scalability and optimization gaps, future research should prioritize co-design approaches. He et al. [10] suggests developing hardware-friendly PQC algorithms tailored for specific platforms. Xie et al. [20] advocates for iterative co-optimization cycles where algorithm refinements are tested on hardware prototypes.

#### 6.5.2 Lightweight PQC for IoT

Lightweight PQC algorithms that minimize energy consumption while maintaining strong security are crucial. Mert et al. [15] recommends exploring hybrid architectures that combine lightweight cryptographic primitives with hardware accelerators. Additionally, Bellizia et al. [5] highlights opportunities for optimizing modular arithmetic and polynomial operations for edge devices.

### 6.5.3 Standardized Testing Frameworks

Developing unified benchmarking tools and standards would enhance comparability. Kris Gaj proposes creating a universal API for PQC implementations to harmonize testing across different hardware platforms. Furthermore, Hosseini et al. [11] suggests a modular benchmarking framework that isolates algorithm-specific optimizations from hardware effects.

# Conclusion

Post-Quantum Cryptography represents a pivotal step toward securing digital infrastructure against the imminent capabilities of quantum computing. This paper has explored architectural strategies for implementing PQC, examining trade-offs between performance, power efficiency, and security. FPGA-based designs offer adaptability and rapid prototyping advantages, while ASICs deliver great energy efficiency and throughput, albeit at the cost of flexibility. Hybrid architectures provide a transitional solution, blending the benefits of classical and quantum-resistant cryptographic systems. Despite these advancements, challenges remain in scaling across security levels, optimizing lightweight PQC for constrained environments, and standardizing performance benchmarks. Addressing these gaps requires collaborative efforts in algorithm-hardware co-design, energy-efficient architectures, and unified evaluation frameworks. By resolving these challenges, the cryptographic community can ensure the widespread adoption and efficacy of PQC, safeguarding digital communications in the quantum era.

# References


[1] R. Agrawal, L. Bu, A. Ehret, and M. Kinsy. Open-source fpga implementation of post-quantum cryptographic hardware primitives. In *2019 29th International Conference on Field Programmable Logic and Applications (FPL)*, pages 211–217. IEEE, 2019.

[2] G. Alsuhli, H. Saleh, M. Al-Qutayri, B. Mohammad, and T. Stouraitis. Area and power efficient fft/ifft processor for falcon post-quantum cryptography. *IEEE Transactions on Emerging Topics in Computing*, 2024.

[3] A. Aydeger, E. Zeydan, A. K. Yadav, K. T. Hemachandra, and M. Liyanage. Towards a quantum-resilient future: Strategies for transitioning to post-quantum cryptography. In *2024 15th International Conference on Network of the Future (NoF)*, pages 195–203, 2024.

[4] R. Bavdekar, E. Jayant Chopde, A. Agrawal, A. Bhatia, and K. Tiwari. Post quantum cryptography: A review of techniques, challenges and standardizations. In *2023 International Conference on Information Networking (ICOIN)*, pages 146–151, 2023.

[5] D. Bellizia et al. Post-quantum cryptography: Challenges and opportunities for robust and secure hw design. In *2021 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, pages 1–6, 2021.

[6] A. C. Canto, A. Sarker, J. Kaur, M. M. Kermani, and R. Azarderakhsh. Error detection schemes assessed on fpga for multipliers in lattice-based key encapsulation mechanisms in post-quantum cryptography. *IEEE Transactions on Emerging Topics in Computing*, 11(3):791–797, 2023.

[7] D. Dam, T. Tran, V. Hoang, C. Pham, and T. Hoang. A survey of post-quantum cryptography: Start of a new race. *Cryptography*, 7(3):40, 2023.

[8] V. B. Dang, F. Farahmand, M. Andrzejczak, and K. Gaj. Implementing and benchmarking three lattice-based post-quantum cryptography algorithms using software/hardware codesign. In *2019 International Conference on Field-Programmable Technology (ICFPT)*, pages 206–214, 2019.

[9] K. Gaj. Challenges and rewards of implementing and benchmarking post-quantum cryptography in hardware. In *Proceedings of the 2018 on Great Lakes Symposium on VLSI*, pages 359–364, 2018.

[10] P. He, T. Bao, J. Xie, and M. Amin. Fpga implementation of compact hardware accelerators for ring-binary-lwe-based post-quantum cryptography. *ACM Transactions on Reconfigurable Technology and Systems*, 16(3):1–23, 2023.

[11] S. M. Hosseini and H. Pilaram. A comprehensive review of post-quantum cryptography: Challenges and advances. *Cryptology ePrint Archive*, 2024.

[12] J. L. Imaña, P. He, T. Bao, Y. Tu, and J. Xie. Efficient hardware arithmetic for inverted binary ring-lwe based post-quantum cryptography. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 69(8):3297–3307, 2022.

[13] M. Imran, A. Aikata, S. S. Roy, and S. Pagliarini. High-speed design of post quantum cryptography with optimized hashing and multiplication. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 71(2):847–851, 2024.

[14] H. Li, Y. Tang, Z. Que, and J. Zhang. Fpga accelerated post-quantum cryptography. *IEEE Transactions on Nanotechnology*, 21:685–691, 2022.

[15] A. C. Mert, E. Karabulut, E. Öztürk, E. Savaş, M. Becchi, and A. Aysu. A flexible and scalable ntt hardware: Applications from homomorphically encrypted deep learning to post-quantum cryptography. In *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 346–351. IEEE, 2020.

[16] C. A. Roma, C. E. A. Tai, and M. A. Hasan. Energy efficiency analysis of post-quantum cryptographic algorithms. *IEEE Access*, 9:71295–71317, 2021.

[17] D. Soni, K. Basu, M. Nabeel, and R. Karri. A hardware evaluation study of nist post-quantum cryptographic signature schemes. In *Second PQC Standardization Conference*. NIST, 2019.

[18] N. Venkatachalam et al. Scalable qkd postprocessing system with reconfigurable hardware accelerator. *IEEE Transactions on Quantum Engineering*, 4, 2023.

[19] J. Xie, K. Basu, K. Gaj, and U. Guin. Special session: The recent advance in hardware implementation of post-quantum cryptography. In *2020 IEEE 38th VLSI Test Symposium (VTS)*, pages 1–10, 2020.

[20] J. Xie, K. Basu, K. Gaj, and U. Guin. Special session: The recent advance in hardware implementation of post-quantum cryptography. In *2020 IEEE 38th VLSI Test Symposium (VTS)*, pages 1–10. IEEE, 2020.

[21] J. Xie, P. He, X. Wang, and J. L. Imaña. Efficient hardware implementation of finite field arithmetic ab+c for binary ring-lwe based post-quantum cryptography. *IEEE Transactions on Emerging Topics in Computing*, 10(2):1222–1228, 2022.

[22] J. Xie, W. Zhao, H. Lee, D. B. Roy, and X. Zhang. Hardware circuits and systems design for post-quantum cryptography—a tutorial brief. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 71(3):1670–1676, 2024.

[23] G. Xu, J. Mao, E. Sakk, and S. P. Wang. An overview of quantum-safe approaches: Quantum key distribution and post-quantum cryptography. In *2023 57th Annual Conference on Information Sciences and Systems (CISS)*, pages 1–6, 2023.

## Biography

*Melchior THIERRY is an Master student in IoT and Cybersecurity at ESILV in Paris, with a strong foundation in embedded system security, AI in cybersecurity, and IoT architecture. His academic experience includes exchange semesters in South Korea and Taiwan, enhancing his expertise in Electronic and Electrical Engineering. He has conducted research on Agri-photovoltaics with the SIRTA of École Polytechnique de Paris, focusing on data & IoT. Professionally, Melchior has gained practical experience in network architecture and cybersecurity through internships in the production and retail sectors.*