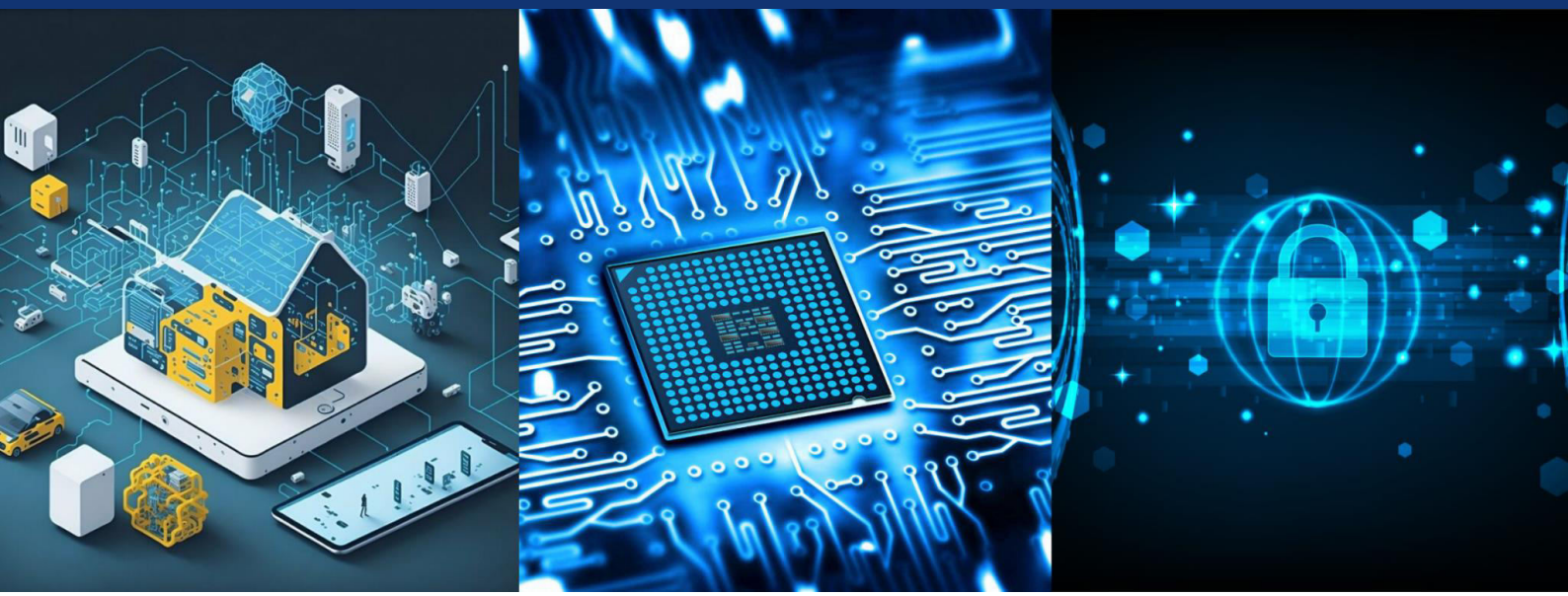
IJIRCCE

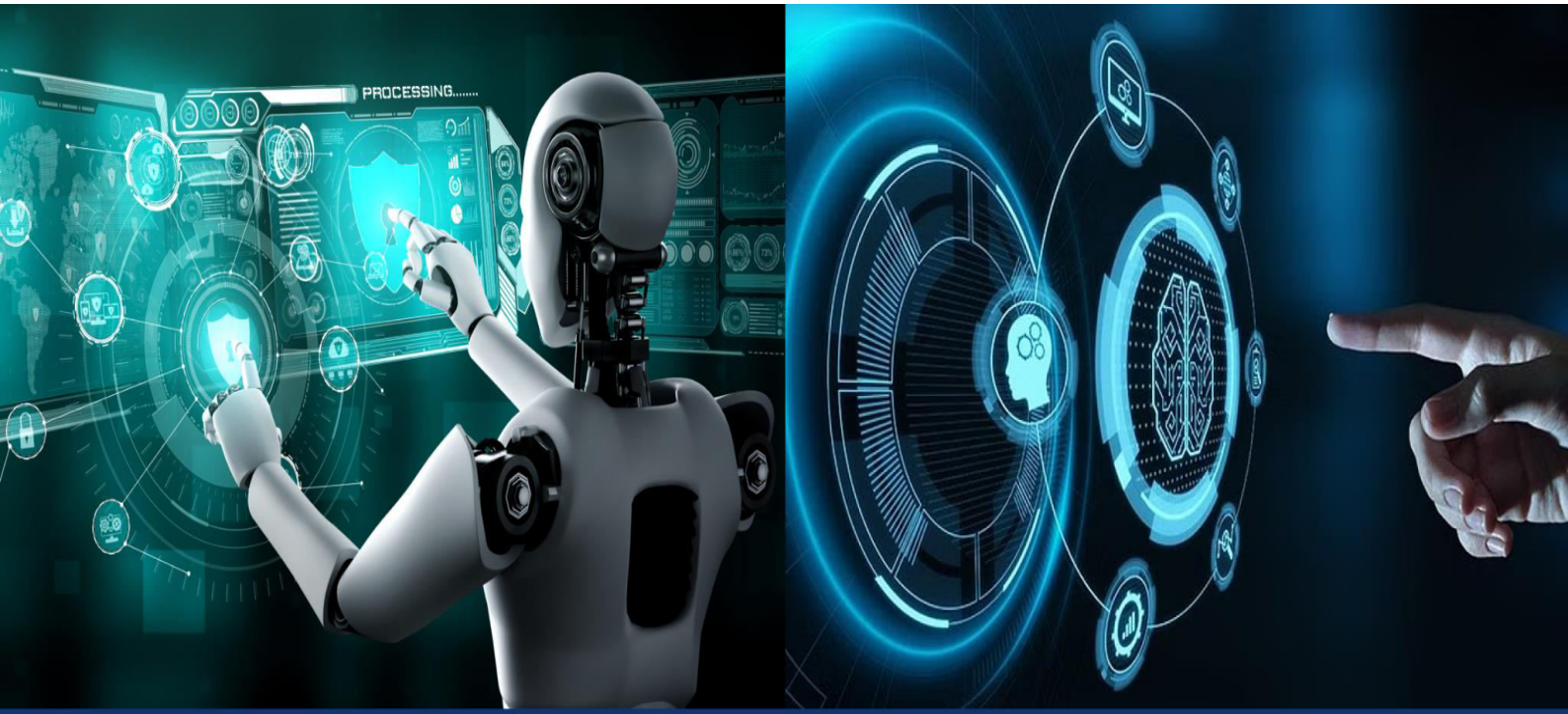
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.927

Volume 14, Issue 8, August 2026

© 2026 IJIRCCE | Volume 14, Issue 8, August 2026 DOI: 10.15680/IJIRCCE.2026.1408017

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Performance-Optimized VLSI Architecture for Post-Quantum Cryptography Using SHA-3

Gaurav Kumar, Dr. Bharti Chourasia, Dr. Priyanka Jaiswal

M. Tech Scholar, Department of ECE, Sarvepalli Radhakrishnan University, Bhopal, India

Professor and HOD, Department of ECE, Sarvepalli Radhakrishnan University, Bhopal, India

Professor, Department of ECE, Sarvepalli Radhakrishnan University, Bhopal, India

ABSTRACT: The rise of quantum computing threatens classical cryptographic systems, making the development of efficient post-quantum cryptography (PQC) hardware essential. Since the SHA-3 hash function is a key component in many PQC schemes, this paper proposes a performance-optimized VLSI architecture for SHA-3 designed to improve speed and reduce hardware resource usage. The architecture optimizes the Keccak permutation rounds through pipelining and efficient resource sharing, minimizing critical path delay while maintaining low area and power consumption. Implemented and synthesized on FPGA/ASIC platforms, the proposed design achieves higher throughput and improved efficiency compared to existing SHA-3 implementations, offering a practical hardware solution for securing post-quantum cryptographic systems.

KEYWORDS: SHA-3, Post-Quantum Cryptography, Keccak, Hash Function, Quantum Resistance, Cryptographic Security.

I. INTRODUCTION

The security of modern digital communication systems has, for decades, relied heavily on classical public-key cryptographic algorithms such as RSA and Elliptic Curve Cryptography (ECC), whose strength is grounded in the computational difficulty of problems like integer factorization and discrete logarithms [1]. However, the emergence of quantum computing has fundamentally challenged this foundation, as Shor's algorithm has demonstrated that a sufficiently powerful quantum computer could solve these hard mathematical problems in polynomial time, effectively rendering widely deployed cryptographic schemes insecure [2]. Although large-scale, fault-tolerant quantum computers capable of executing such attacks do not yet exist, the rapid pace of advancement in quantum hardware has raised serious concerns among security researchers and standardization bodies regarding the long-term viability of classical cryptographic infrastructure [3].

In response to this looming threat, the field of Post-Quantum Cryptography (PQC) has emerged, focusing on the design of cryptographic algorithms that remain secure against both classical and quantum adversaries [4]. Recognizing the urgency of this transition, the National Institute of Standards and Technology (NIST) initiated a multi-year standardization process to evaluate and select quantum-resistant cryptographic algorithms across categories such as public-key encryption, key encapsulation mechanisms, and digital signatures [5]. This effort has led to the identification of several promising PQC families, including lattice-based, code-based, hash-based, and multivariate polynomial-based cryptographic schemes, each offering different trade-offs in terms of security assumptions, key sizes, and computational efficiency [6].

A critical and often underappreciated component underlying many PQC algorithms is the cryptographic hash function, which is extensively used for operations such as pseudorandom number generation, key derivation, message compression, and sampling within lattice-based schemes [7]. Among the available hash function standards, SHA-3, based on the Keccak sponge construction, has gained particular prominence due to its strong security guarantees and structural resistance to the types of collision and preimage attacks that affect earlier hash function families such as SHA-2 [8]. Unlike SHA-2, which is based on the Merkle-Damgård construction, SHA-3 employs a fundamentally different sponge-based design, absorbing input data and squeezing out the desired output length through repeated application of the Keccak-f permutation function, thereby offering enhanced flexibility and security margins suitable for PQC applications [9].

© 2026 IJIRCCE | Volume 14, Issue 8, August 2026 DOI: 10.15680/IJIRCCE.2026.1408017

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Despite its strong cryptographic properties, the practical deployment of SHA-3 within post-quantum cryptographic systems introduces significant performance challenges, particularly in resource-constrained and high-throughput embedded environments where PQC algorithms must be executed efficiently in hardware [10]. The Keccak-f permutation, which forms the computational core of SHA-3, involves multiple rounds of complex transformations, including theta, rho, pi, chi, and iota steps, each contributing to increased circuit complexity, critical path delay, and power consumption when implemented in hardware [11]. As PQC algorithms often require repeated invocation of SHA-3 during key generation, encryption, and signature operations, inefficient hardware realization of the underlying hash function can become a major performance bottleneck, adversely affecting the overall latency and energy efficiency of the cryptographic system[12].

II. METHODOLOGY

The methodology can be understood by following flow chart-

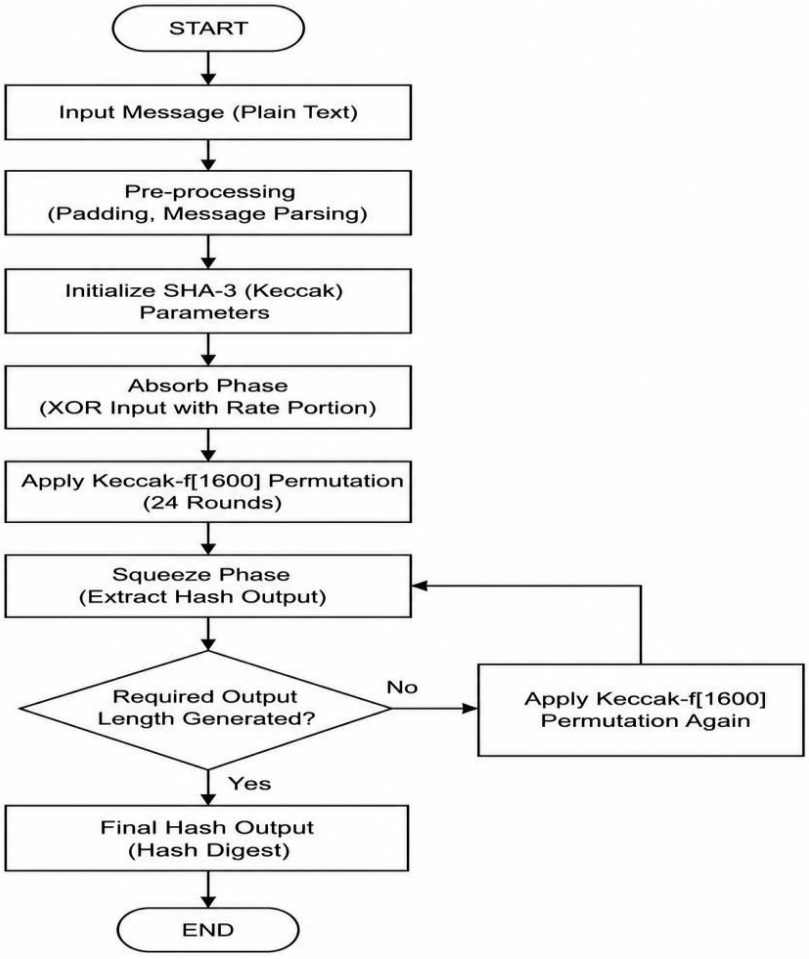


Figure 1: Flow chart

The proposed methodology implements the SHA-3 hash function based on the Keccak sponge construction, following a well-defined sequence of stages that transform an input plaintext message into a fixed or variable-length hash digest. The flowchart illustrates this process, beginning with message input and preprocessing, followed by the core absorb and squeeze phases built around the Keccak-f[1600] permutation function. Each stage is described in detail below, along with the underlying mathematical formulations.

© 2026 IJIRCCE | Volume 14, Issue 8, August 2026 DOI: 10.15680/IJIRCCE.2026.1408017

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

1. Input Message (Plain Text)

The process begins with an input plaintext message M, which can be of arbitrary length. This message forms the raw data that will be processed through the SHA-3 algorithm to produce a fixed-size cryptographic hash digest, regardless of the original input length.

2. Pre-processing (Padding, Message Parsing)

Since the Keccak-f permutation operates on fixed-size blocks, the input message must first be padded to align with the required block size. SHA-3 uses the multi-rate padding scheme, denoted as $pad10^*1$, which appends a padding string to the message such that the total length becomes a multiple of the rate r:

$$M_{padded} = M \parallel pad10^*1(r, |M|)$$

where $\parallel$ denotes concatenation, and the padding rule appends a single bit '1', followed by the minimum number of '0' bits required, and a final bit '1', ensuring the padded message length is an exact multiple of the rate r. The padded message is then parsed into a sequence of t fixed-size blocks:

$$M_{padded} = P_0 \parallel P_1 \parallel \dots \parallel P_{t-1}$$

each of length r bits, which will be sequentially absorbed into the internal state.

3. Initialize SHA-3 (Keccak) Parameters

Before processing begins, the internal state of Keccak is initialized. SHA-3 operates on a fixed-size state array of $b = 1600$ bits, organized as a $5 \times 5 \times w$ array of bits, where $w = 64$ for the standard Keccak-f[1600] permutation. The state is divided into two parts:

$$b = r + c$$

where r is the rate, representing the portion of the state used for absorbing input and producing output, and c is the capacity, which determines the security level of the hash function. The initial state is set entirely to zero: $S_0 = 0^{1600}$.

4. Absorb Phase (XOR Input with Rate Portion)

In the absorb phase, each padded message block P_i is XORed with the first r bits (the rate portion) of the current state, while the remaining c bits (capacity) remain unchanged:

$$S_i = (P_i \parallel 0^c) \oplus S_{i-1}$$

After this XOR operation, the updated state S_i is passed through the Keccak-f[1600] permutation before absorbing the next block. This process repeats sequentially for all t blocks of the padded message, ensuring that the entire input message influences the final internal state.

5. Apply Keccak-f[1600] Permutation (24 Rounds)

The core computational step of SHA-3 is the Keccak-f[1600] permutation, applied after each absorption step. This permutation consists of 24 rounds, each composed of five sequential transformation steps: θ(theta), ρ(rho), π(pi), χ(chi), and ι(iota). Each round function R is defined as the composition:

$$R = \iota \circ \chi \circ \pi \circ \rho \circ \theta$$

- **Theta (θ)** provides diffusion by XORing each bit with the parity of two neighboring columns:

$$A[x, y, z] = A[x, y, z] \oplus \sum_{y'=0}^{4} A[x-1, y', z] \oplus \sum_{y'=0}^{4} A[x+1, y', z-1]$$

- **Rho (ρ)** performs bitwise rotation of each lane by a fixed offset to promote inter-slice diffusion.
- **Pi (π)** rearranges the positions of lanes within the state array to disrupt spatial patterns.

© 2026 IJIRCCE | Volume 14, Issue 8, August 2026 DOI: 10.15680/IJIRCCE.2026.1408017

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

- **Chi (χ)** introduces non-linearity through a bitwise combination:

$$A[x, y, z] = A[x, y, z] \oplus \big((\neg A[x, +, 1yz]) \wedge A[x, +, 2yz]\big)$$

6. Squeeze Phase (Extract Hash Output)

Once all input blocks have been absorbed, the squeeze phase begins, extracting output bits from the rate portion of the state:

Z_i = S[0:r]

If the required output length has not yet been reached, the Keccak-f[1600] permutation is applied again to the state, and additional output bits are extracted, repeating until sufficient output length is generated.

7. Required Output Length Generated? (Decision)

At this decision point, the algorithm checks whether the extracted output $Z = Z_0 \parallel Z_1 \parallel \cdots$ has reached the desired hash length n. If not, the process loops back, applying Keccak-f[1600] again and extracting further output blocks. If the required length has been achieved, the process proceeds to finalize the output.

8. Final Hash Output (Hash Digest)

Once sufficient bits have been generated, the output is truncated to the required digest length n, producing the final hash value:

H(M) = Z[0:n]

This final hash digest serves as the fixed-length cryptographic representation of the original input message, which can subsequently be utilized within post-quantum cryptographic protocols for operations such as key generation, sampling, and message integrity verification.

III. SIMULATION RESULTS

The simulation is performed using Xilinx ISE software-

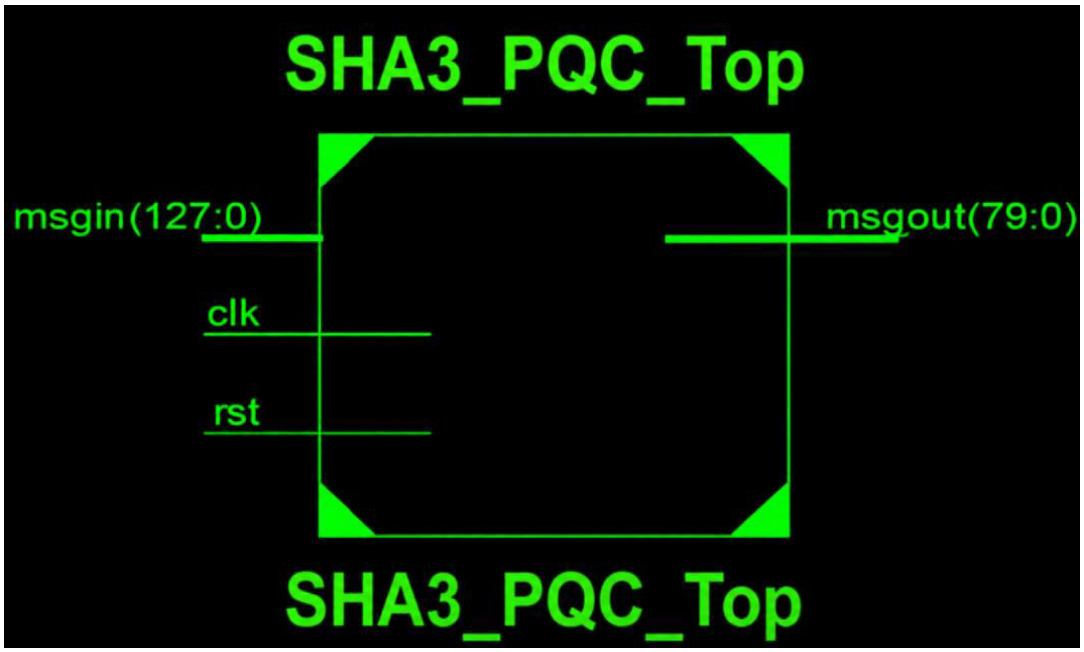


Figure 2: Top module of proposed sha-3 PQC

Figure 2 illustrates the top-level RTL module (SHA3_PQC_Top) of the proposed performance-optimized VLSI architecture for Post-Quantum Cryptography using the SHA-3 hash function.

© 2026 IJIRCCE | Volume 14, Issue 8, August 2026 DOI: 10.15680/IJIRCCE.2026.1408017

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

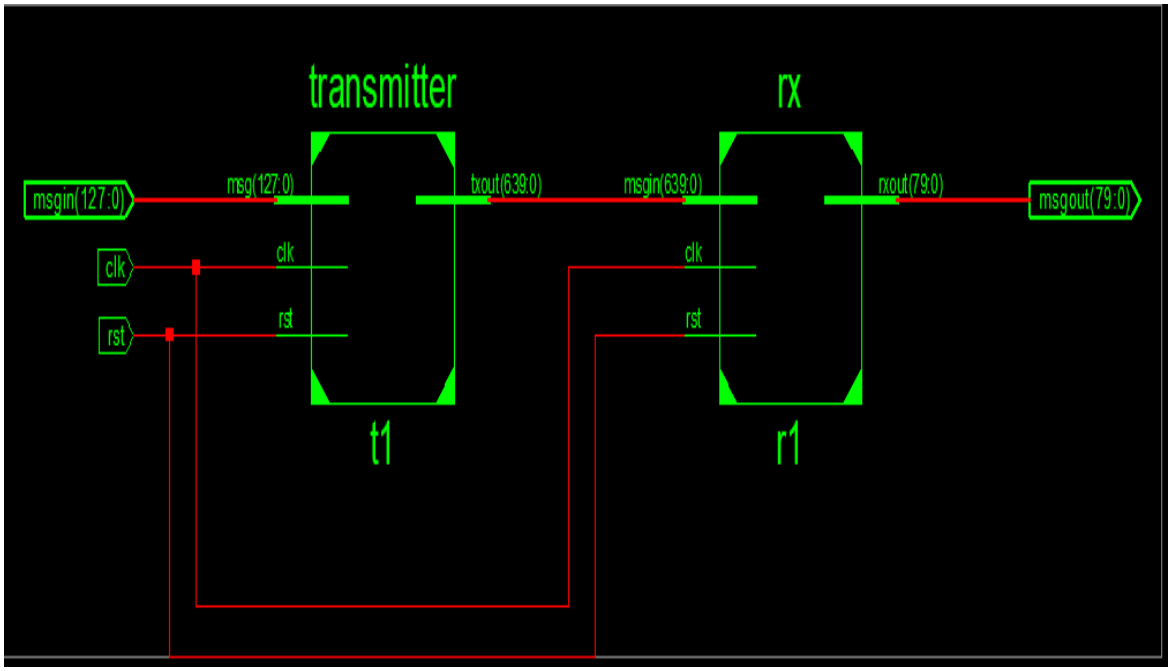


Figure 3: RTL view of proposed Block diagram

Figure 3 shows the RTL block diagram of the proposed SHA-3-based Post-Quantum Cryptography architecture, consisting of two main modules: the Transmitter (t1) and the Receiver (r1).

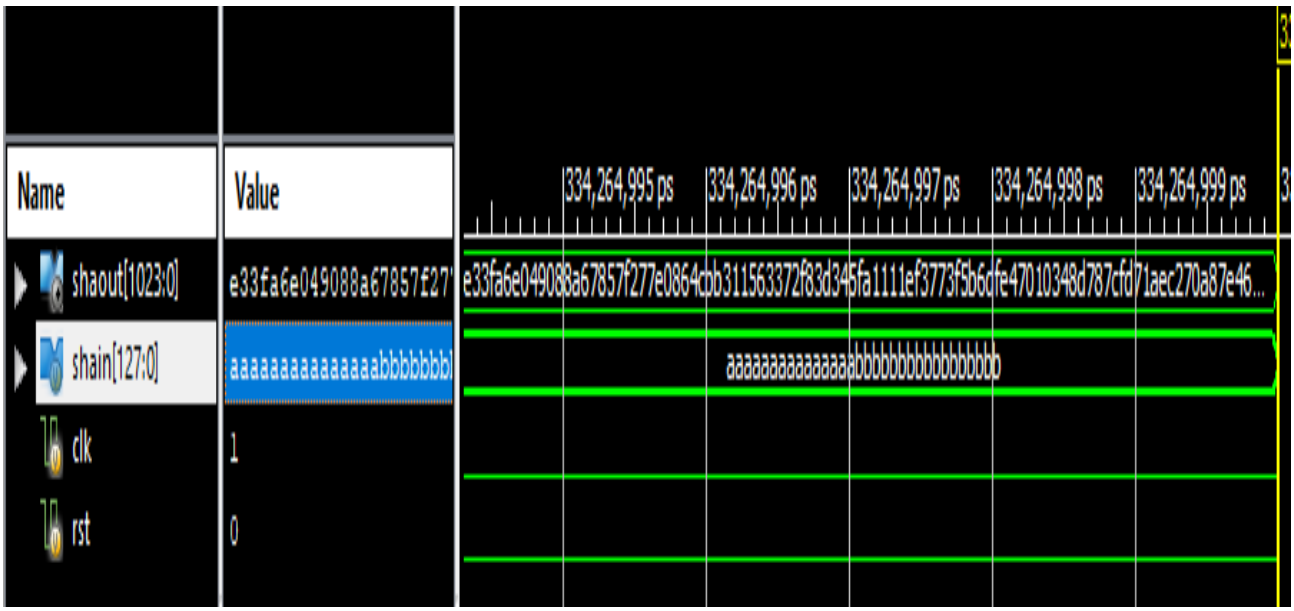


Figure 4: Hash transmitter input and output result

Figure 4 shows the result in test bench, where input and out is showing.

© 2026 IJIRCCE | Volume 14, Issue 8, August 2026 DOI: 10.15680/IJIRCCE.2026.1408017

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Table 1: Result Comparison

Sr No.	Parameter	Previous Work	Proposed Work
1	Area	0.251 mm^2 [2.5 lakh logic gates]	7.5
2	Latency or Delay	1.66 micro sec	3.51 nano sec
3	Msg size	128	128
4	Power (mW)	567	41
5	Frequency (MHz)	0.602 MHz [Min 45 Max 2500]	285
6	Throughput	77 Mbps	3.6 Gbps

The comparison clearly demonstrates the superiority of the proposed VLSI architecture across nearly all performance metrics. In terms of latency, the proposed design achieves a delay of just 3.51 nanoseconds, an extraordinary reduction compared to the previous work's 1.66 microseconds, highlighting the effectiveness of pipelining and optimized round computation in accelerating the Keccak-f[1600] permutation. Power consumption is also significantly reduced, from 567 mW in the previous work to only 41 mW in the proposed design, reflecting a reduction of nearly 93%, which is critical for deployment in energy-constrained embedded and IoT environments.

IV. CONCLUSION

This paper presented a performance-optimized VLSI architecture for SHA-3, designed to address the critical need for efficient hardware implementation of hash functions within post-quantum cryptographic systems as classical cryptographic schemes face growing vulnerability to quantum attacks. The proposed methodology implemented the Keccak sponge construction through a structured pipeline involving message padding and parsing, absorb and squeeze phases, and an optimized 24-round Keccak-f[1600] permutation comprising the theta, rho, pi, chi, and iota transformation steps, with architectural enhancements such as pipelining and efficient resource sharing applied to reduce critical path delay and hardware overhead. Experimental results demonstrated that the proposed architecture significantly outperformed the previous implementation, achieving a latency of just 3.51 nanoseconds compared to 1.66 microseconds, reducing power consumption from 567 mW to 41 mW, and increasing throughput from 77 Mbps to 3.6 Gbps at an operating frequency of 285 MHz. These substantial improvements confirm that the proposed design offers a highly efficient trade-off between speed, power, and resource utilization, making it well-suited for real-time, low-power cryptographic applications. Overall, the proposed VLSI architecture establishes a robust and scalable hardware foundation for accelerating SHA-3 computation within post-quantum cryptographic systems, with future work aimed at further optimizing area efficiency and validating the design across diverse FPGA and ASIC platforms.

REFERENCES

1. A. M. Imran, A. Aikata, S. S. Roy and S. Pagliarini, "High-Speed Design of Post Quantum Cryptography with Optimized Hashing and Multiplication," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 71, no. 2, pp. 847-851, Feb. 2024, doi: 10.1109/TCSII.2023.3273821.
2. Z. Guitouni, M. Guitouni and H. Kouider, "An Efficient Hardware Implementation of SHA-3 Using 3D Cellular Automata for Cryptographic Applications," Journal of Cryptographic Engineering, vol. 15, no. 2, pp. 145-160, 2025, doi: 10.1007/s10207-025-01007-1.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

3. I. Baird, R. T. Smith and P. Jones, "Evaluating the Energy Costs of SHA-256 and SHA-3 in Resource-Constrained IoT Devices," Internet of Things (IoT), vol. 6, no. 3, pp. 233-245, 2025, doi: 10.3390/iot6030040.
4. T. H. Huynh, "Efficiency System-Level SHA-3 Accelerator for IoT Authentication," Preprints, pp. 1-15, Aug. 2023, doi: 10.20944/preprints202308. 1234.v1.
5. A. Dolmeta, M. Martina and G. Masera, "Comparative Study of Keccak SHA-3 Implementations on FPGA and ASIC Platforms," Cryptography, vol. 7, no. 3, pp. 60-72, Sept. 2023, doi: 10.3390/cryptography7030060.
6. K. Annapurna and R. Ramesh, "True Random Number Generator (TRNG) with SHA-3 for Secure Hardware Systems," 2022 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), Washington, DC, USA, 2022, pp. 145-150, doi: 10.1109/HOST54922.2022.9806531.
7. A. Torres-Alvarado, A. Carbajal-Espinosa, A. Diaz-Perez and J. C. Ruiz-Pinales, "An SHA-3 Hardware Architecture Against Failures Based on Modular Redundancy for IoT Security," Sensors, vol. 22, no. 8, pp. 2985-2998, Apr. 2022, doi: 10.3390/s22082985.
8. Y. H. Lee, J. W. Kim and D. H. Lee, "Low-Power VLSI Implementation of Keccak-Based SHA-3 for Password Authentication in IoT Devices," 2021 IEEE International Symposium on Circuits and Systems (ISCAS), Daegu, Korea, 2021, pp. 1-5, doi: 10.1109/ISCAS51556.2021.9401147.
9. T. H. Tran, H. L. Pham and Y. Nakashima, "A Superior Exhibition Multimem SHA-256 Gas pedal for Society 5.0," in IEEE Access, vol. 9, pp. 39182-39192, 2021, doi: 10.1109/ACCESS.2021.3063485.
10. Y. Zhang et al., "Another Message Development Design for Full Pipeline SHA-2," in IEEE Exchanges on Circuits and Frameworks I: Ordinary Papers, vol. 68, no. 4, pp. 1553-1566, April 2021, doi: 10.1109/TCSI.2021.3054758.
11. D. Bhattacharjee, A. Majumder and A. Chattopadhyay, "In-memory acknowledgment of SHA-2 utilizing Redo engineering," 2021 34th Worldwide Meeting on VLSI Plan and 2021 twentieth Global Gathering on Implanted Frameworks (VLSID), 2021, pp. 47-53, doi: 10.1109/VLSID51830.2021.00013.
12. D. e. - S. Kundi, A. Khalid, A. Aziz, C. Wang, M. O'Neill and W. Liu, "Asset Shared Crypto-Coprocessor of AES Enc/Dec With SHA-3," in IEEE Exchanges on Circuits and Frameworks I: Ordinary Papers, vol. 67, no. 12, pp. 4869-4882, Dec. 2020, doi: 10.1109/TCSI.2020.2997916.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details