

ARTICLE

Efficient Cryptographic Architectures for Edge AI and Real-Time Embedded Systems

Makhkamov Bakhtiyor Shukhratovich¹, Akmalbek Abdusalomov^{1,2,3,4},
Akhmedova Nodira Aminjanovna¹, Botirov Sokhibjon Rustam Ugli¹, Jasur Sevinov^{2,5},
Alpamis Kutlimuratov⁶, Boburjon Vafoev⁷, Yodgorkhon Ilkhamova⁷ and Young Im Cho^{8,*}

¹Department of Computer Systems/Physics, Tashkent University of Information Technologies Named after Muhammad Al-Khwarizmi, Tashkent, Uzbekistan

²Department of Information Processing and Control Systems, Tashkent State Technical University, Tashkent, Uzbekistan

³Department of Electronics and Instrumentation, Fergana State Technical University, Fergana, Uzbekistan

⁴Department of International Scientific Journals and Rankings, Alfraganus University, Tashkent, Uzbekistan

⁵Department of Computer Engineering, University of Tashkent for Applied Sciences, Tashkent, Uzbekistan

⁶Department of Applied Informatics, Kimyo International University in Tashkent, Tashkent, Uzbekistan

⁷Department of Digital Economy, Tashkent State University of Economics, Tashkent, Uzbekistan

⁸Department of Computer Engineering, Gachon University, Seongnam-si, Gyeonggi-Do, Republic of Korea

*Corresponding Author: Young Im Cho. Email: yicho@gachon.ac.kr

Received: 06 May 2026; Accepted: 10 August 2026; Published: 22 October 2026

ABSTRACT: This paper presents field-programmable gate array (FPGA)-based engineering optimizations of the Ascon lightweight cryptographic algorithm, targeting real-time edge AI and embedded systems. The work systematically evaluates how established design strategies, such as selective round unrolling and pipelined permutation stages, scale across heterogeneous FPGA platforms. Two architectures are explored, each optimized for different design objectives, including throughput (TP), latency, and hardware efficiency. The designs are implemented on two FPGA families, Xilinx Kintex UltraScale and the 7-Series Spartan, to evaluate scalability across both high-performance and resource-constrained platforms. Experimental results show that selective round unrolling in Architecture 2 achieves the highest TP, reaching 624.57 Mb/s on UltraScale and 377.78 Mb/s on Spartan-7, while Architecture 1 provides the lowest LUT utilization, requiring as few as 1804 LUTs. Throughput-to-area ratio analysis reveals platform-dependent efficiency, with Architecture 2 performing best on UltraScale and Architecture 1 on Spartan-7. Application mapping demonstrates suitability across smart surveillance, autonomous systems, and embedded inference nodes. The results indicate that lightweight Ascon architectures can be suitable for diverse edge environments, based on their measured TP, latency, and area characteristics.

KEYWORDS: Lightweight cryptography; FPGA architectures; edge AI systems

1 Introduction

The rapid evolution of edge computing and artificial intelligence (AI) has led to the emergence of edge AI systems, where data processing is performed locally on embedded devices such as smart cameras, autonomous platforms, and intelligent sensors. This paradigm reduces latency, minimizes bandwidth usage, and enables real-time decision-making. However, it also introduces significant security challenges, as sensitive data (images, sensor readings, and AI inference results) is processed and transmitted directly

from edge devices. Recent studies highlight that edge-enabled Internet of Things (IoT) environments are particularly vulnerable due to their distributed nature and limited computational capabilities [1,2].

To address these concerns, cryptographic mechanisms must be integrated directly into edge devices. However, conventional cryptographic solutions, originally designed for high-performance computing systems, often fail to meet the stringent resource constraints of edge environments. This has driven growing interest in lightweight and hardware-efficient cryptographic architectures capable of delivering strong security guarantees while maintaining low latency and resource consumption [3,4].

1.1 Problem Statement

Despite the critical need for secure edge processing, several challenges remain:

- **High computational complexity:** Traditional encryption algorithms such as AES [5] and public-key schemes [6] impose significant computational overhead, making them less suitable for real-time edge AI applications.
- **Resource constraints:** Edge devices operate under strict limitations in terms of logic resources and memory, restricting the deployment of conventional cryptographic solutions [2].
- **Real-time requirements:** Applications such as smart surveillance, autonomous systems, and industrial control demand low-latency and high-throughput (TP) encryption, which is difficult to achieve with software-based implementations [7].
- **Scalability challenges:** Edge environments require adaptable solutions that can scale across heterogeneous platforms, from low-end devices to high-performance edge nodes [8].

Recent research has explored hardware acceleration and field-programmable gate array (FPGA)-based solutions to address these challenges. For instance, FPGA-based and in-memory cryptographic architectures have demonstrated improvements in TP and energy efficiency for edge and IoT applications [7,8]. Similarly, emerging hardware-assisted encryption techniques have shown promise in reducing latency while maintaining strong security guarantees [9,10].

1.2 Contribution of This Work

To address the above challenges, this paper presents efficient FPGA-based hardware architectures for the Ascon lightweight authenticated encryption algorithm [11], specifically for edge AI and real-time embedded systems. These designs implement the Ascon permutation and round functions in hardware, with architectural variations targeting different performance and resource objectives. The architectural techniques employed are well-established in lightweight cryptographic hardware design. The novelty of this work lies in the application-driven, cross-platform optimization of Ascon for edge AI constraints. Our contribution is therefore positioned as a systematic engineering exploration that quantifies how standard architectural strategies perform across heterogeneous FPGA platforms and real-time edge workloads. This engineering-oriented exploration fills a gap in the existing literature by providing a systematic Throughput-to-Area ratio (TP/A) analysis and application-specific mapping for Ascon. The main contributions of this work are summarized as follows:

- **Ascon-Specific Hardware Architecture Design:** We implemented two FPGA architectures of the Ascon encryption engine, each optimized for distinct performance goals such as area efficiency, low latency, and high TP. The designs explore different levels of round unrolling, pipelining, and resource sharing for Ascon's permutation structure.

- **Cross-Platform FPGA Evaluation:** The Ascon architectures are implemented and evaluated on two distinct FPGA families, one high-performance and one resource-constrained, to demonstrate scalability, portability, and adaptability across heterogeneous edge platforms.
- **Throughput-to-Area Ratio (TP/A) Metric for Ascon:** A key contribution of this work is the use of TP/A ratio as a comparative metric for Ascon hardware implementations, enabling a balanced evaluation of performance and hardware efficiency across different architectural choices.
- **Application-Driven Mapping for Ascon in Edge AI:** The proposed Ascon architectures are mapped to real-world edge AI scenarios, including smart surveillance nodes, low-power inference devices, and autonomous embedded systems, demonstrating how each architecture aligns with specific latency, TP, and resource constraints.
- **Real-Time Performance Optimization:** Through selective round unrolling and pipelined dataflow, the architectures achieve low latency and competitive TP, indicating their potential suitability for real-time secure processing in edge environments.

The remainder of this paper is organized as follows: [Section 2](#) reviews related work in lightweight cryptography and FPGA-based security architectures. [Section 3](#) presents the system model for edge AI applications. [Section 4](#) describes the proposed architectures in detail. [Section 5](#) outlines the experimental setup, followed by performance evaluation and discussion in [Section 6](#). [Section 7](#) presents application mapping, and [Section 8](#) concludes the paper.

2 Related Work

2.1 Lightweight Cryptography for Resource-Constrained Systems

Lightweight cryptography has emerged as a key research area for addressing the security requirements of resource-constrained systems such as IoT devices, embedded platforms, and edge nodes. Unlike traditional cryptographic algorithms, lightweight ciphers are designed to minimize hardware area, power consumption, and computational complexity while maintaining acceptable security levels. A comprehensive survey in this domain highlights that modern lightweight block ciphers are typically categorized into substitution-permutation network (SPN), Feistel, and ARX-based structures, each offering different trade-offs between performance and hardware efficiency [12].

Among the most widely adopted lightweight ciphers is PRESENT [13], which is designed for ultra-constrained hardware environments and achieves significantly smaller implementation area compared to AES while maintaining adequate security. Similarly, ciphers such as GIFT [14] and RECTANGLE [15] have been proposed to further optimize hardware efficiency and reduce energy consumption in embedded systems [12]. These algorithms are particularly suitable for applications in which minimizing logic utilization is critical.

Despite these advancements, lightweight cryptographic designs often prioritize reducing area or energy consumption, sometimes at the expense of TP or latency. Recent works have also explored hybrid approaches combining lightweight symmetric cryptography with emerging techniques to improve performance and security in IoT and edge environments [16].

In parallel with these developments, Ascon has gained significant prominence as a lightweight authenticated encryption and hashing family optimized for constrained devices. Its design is based on a sponge construction with a permutation for low-area hardware that offers strong resistance against differential and linear attacks while maintaining excellent performance across ASIC, FPGA, and microcontroller platforms. Ascon's suitability for both encryption and hashing makes it particularly attractive for unified security stacks in IoT deployments. Reflecting its maturity and robustness, National Institute of Standards

and Technology (NIST) selected Ascon in 2023 as the winner of the lightweight cryptography (LWC) standardization process, leading to its formal standardization as NIST SP 800-232 [17] in 2024 for lightweight authenticated encryption. This marks the first globally recognized standard specifically targeting constrained environments, complementing ongoing efforts in lightweight block cipher design and providing a cohesive foundation for secure-by-design IoT and edge systems.

2.2 Ascon Internal Architecture

Ascon is a lightweight authenticated encryption algorithm built around a 320-bit permutation structured as a substitution-permutation network (SPN). Its internal state is organized into five 64-bit words, which undergo a sequence of round transformations consisting of three main components: (1) the addition of round constants to ensure resistance against symmetry-based attacks, (2) a non-linear substitution layer based on a 5-bit S-box applied in parallel across all state bits, and (3) a linear diffusion layer that mixes the state through word-wise rotations and XOR operations. In Ascon, the permutation uses either 12 or 6 rounds depending on the phase of the algorithm. This round-based structure, together with its bit-sliced S-box and simple linear operations, makes Ascon well suited for hardware implementations and enables designers to explore a wide range of architectural choices, from compact iterative designs to fully unrolled, deeply pipelined high-TP engines. Fig. 1 illustrates the core components of the Ascon permutation and their dataflow relationships.

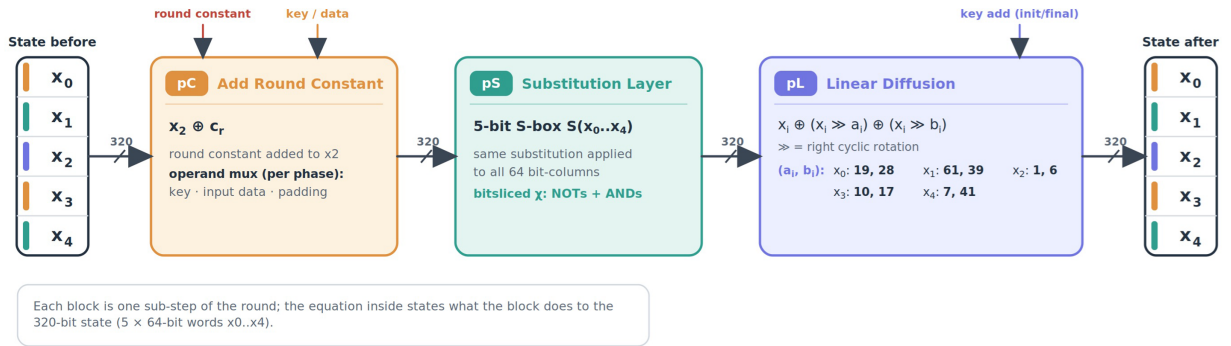


Figure 1: One round of the permutation function.

2.3 FPGA-Based Cryptographic Architectures

FPGAs have been widely used as a platform for implementing cryptographic algorithms due to their flexibility, parallelism, and suitability for hardware acceleration. FPGA-based implementations enable designers to achieve high TP and low latency by exploiting parallel processing and pipelining techniques.

Recent research has demonstrated the effectiveness of FPGA-based cryptographic accelerators for edge and embedded systems. For instance, lightweight AES implementations optimized for FPGA platforms have achieved significant improvements in TP while maintaining moderate resource utilization [18]. Similarly, scalable encryption engines implemented on modern FPGA platforms have demonstrated high TP and low-latency performance suitable for edge computing applications [8].

In addition, emerging architectures such as processing-in-memory and hybrid encryption accelerators have been proposed to enhance performance and reduce data-movement overhead in edge systems [8,19]. These approaches highlight the growing importance of hardware acceleration in enabling secure real-time processing in edge AI environments.

Beyond traditional block ciphers, FPGA-based architectures for Ascon have gained significant attention due to Ascon's suitability for constrained hardware platforms. Ascon's permutation-based design enables efficient hardware mapping, and several FPGA implementations have demonstrated that its round-based structure can be optimized through techniques such as partial or full round unrolling, deep pipelining, and resource-sharing strategies [20,21]. Fully unrolled architectures achieve very high TP but require significantly more look-up tables (LUTs) and registers, making them suitable only for high-performance edge accelerators. In contrast, iterative or resource-shared FPGA architectures significantly reduce area while maintaining competitive TP, making them ideal for IoT and battery-powered devices.

With Ascon now standardized as part of the NIST Lightweight Cryptography portfolio, FPGA-based implementations are expected to play a central role in deploying authenticated encryption and hashing capabilities across edge AI, industrial IoT, and embedded sensing platforms. These architectures provide a balanced combination of security, area efficiency, and performance, complementing existing FPGA implementations of lightweight block ciphers.

2.4 Limitations of Existing Work

Despite significant progress in lightweight cryptographic architectures, several limitations persist in the literature. A major shortcoming is the tendency toward single-objective optimization, where many designs focus primarily on either maximizing TP or minimizing hardware area, without considering a balanced trade-off between the two. This narrow focus limits their applicability in practical edge systems, where both performance and resource efficiency are critical. Furthermore, most studies rely on standard benchmarking approaches and lack application-driven evaluation, failing to relate architectural performance to real-world scenarios such as edge AI, smart surveillance, or autonomous systems. Another important limitation is the lack of cross-platform validation, as the majority of FPGA-based implementations are evaluated on a single hardware platform, making it difficult to assess their scalability and adaptability across diverse edge environments. In addition, existing works often overlook system-level constraints (interplay between latency, TP, and hardware efficiency) which are essential for deploying secure, real-time edge-computing systems. These gaps highlight the need for more comprehensive and application-oriented design methodologies in FPGA-based cryptographic research.

3 System Model

Edge AI systems operate at the intersection of embedded computing, AI, and real-time data processing. In these systems, sensitive data, such as video feeds, sensor measurements, or control signals, is acquired locally, processed on-device, and then transmitted to a central server or peer devices when necessary. This paradigm significantly reduces the latency associated with cloud-based processing and enables autonomous decision-making in latency-sensitive applications. However, it also introduces a unique set of security and performance challenges, as the data must be protected without compromising real-time processing capabilities.

To capture the end-to-end operation of such systems, we model an edge AI pipeline consisting of four primary stages: data acquisition, edge processing, encryption, and transmission as shown in Fig. 2. The first stage, data acquisition, involves sensors or cameras that generate raw data from the environment. In smart surveillance systems, this data can be high-definition video streams, whereas in autonomous embedded systems, it may consist of LiDAR readings, inertial measurements, or other multi-sensor fusion data. The second stage, edge processing, executes AI algorithms locally on the device, such as object detection, activity recognition, or anomaly detection. This stage is computationally intensive and operates under

strict latency requirements, necessitating efficient hardware implementations of neural networks or other inference models.

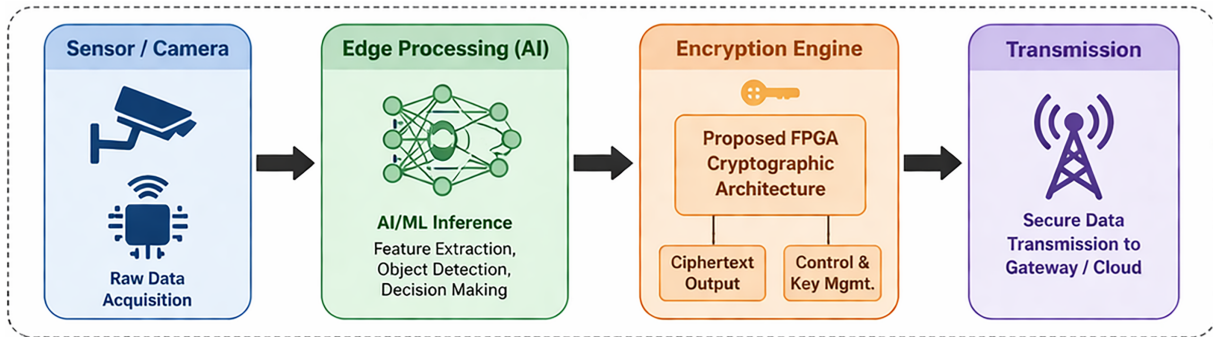


Figure 2: System model of the edge AI pipeline with hardware-based encryption.

Following edge processing, the encryption stage ensures that sensitive data, both raw sensor readings and AI-generated features, are secured before transmission or storage. Lightweight, hardware-efficient cryptographic architectures are particularly suitable at this stage because they minimize logic utilization, reduce latency, and maintain high TP. By integrating encryption directly on the edge device, the system protects against eavesdropping, tampering, and other security threats while preserving the performance benefits of local processing.

Finally, the transmission stage involves sending the encrypted data over communication channels to central servers, other edge nodes, or cloud services for storage, further processing, or collaborative decision-making. The effectiveness of the pipeline relies on the seamless integration of these stages, ensuring that the system meets the combined objectives of low latency, high TP, and robust security.

This system model is applicable across a variety of emerging use cases. In smart surveillance, high-definition video streams are processed in real-time to detect security threats, requiring fast encryption to prevent unauthorized access. Edge AI inference nodes, such as local predictive maintenance devices in industrial environments, must process and encrypt sensor data efficiently to provide timely alerts without excessive energy consumption. In autonomous embedded systems, including unmanned aerial vehicles or autonomous ground vehicles, the pipeline enables secure, real-time decision-making by protecting AI-generated decisions and sensor fusion outputs from malicious interference. By framing the system within this end-to-end pipeline, the proposed cryptographic architectures can be evaluated under realistic workloads and hardware constraints, ensuring practical relevance and scalability.

4 Proposed Architectures

Designing cryptographic engines for edge AI systems requires balancing multiple objectives: hardware efficiency, low latency, high TP, and adaptability to different edge platforms. To this end, we developed two distinct FPGA-based encryption architectures that explore different trade-offs between resource utilization and performance, enabling their deployment across a variety of real-world edge AI scenarios. These architectures apply established hardware design strategies in a structured manner to evaluate their suitability for edge AI environments. The contribution lies in engineering optimization, platform-dependent scaling, and real-time performance characterization rather than methodological novelty. Existing Ascon implementations do not provide such cross-platform, application-driven design-space exploration, and our results highlight platform-dependent trade-offs that have not been previously quantified.

4.1 Baseline Architecture: Conceptual Upper-Bound

The baseline architecture serves as a conceptual upper bound illustrating the maximum theoretical parallelism achievable in Ascon hardware design. It models a fully unrolled and deeply pipelined permutation, where all 12 rounds (or 6 in reduced phases) are instantiated as separate hardware blocks. The 320-bit state passes through these stages with pipeline registers between rounds, enabling a new state to enter the pipeline every cycle once filled.

Each round includes the complete Ascon operations: round-constant addition, a fully parallel 64-instance S-box layer, and the rotation-based diffusion layer. This eliminates iterative control logic and minimizes the per-stage critical path, representing the highest achievable TP for any Ascon FPGA architecture.

However, fully replicating all S-boxes and diffusion networks results in extremely high LUTs and routing usage. Therefore, the Baseline Architecture is not implemented or synthesized and is included only to define the upper limit of the design space and motivate the practical, resource-aware Architecture 1 and Architecture 2 presented next.

4.2 Architecture 1: Area-Efficient Design

While selective round unrolling and resource sharing are established optimization techniques, this work quantifies their impact on TP/A ratio efficiency across heterogeneous FPGA families, an aspect that has not been systematically evaluated in previous Ascon implementations.

Architecture 1 is designed to balance hardware efficiency and performance for resource-constrained edge platforms where logic utilization and power consumption are critical. Unlike the high-resource baseline, it reduces instantiated hardware while maintaining acceptable TP and latency for real-time encryption. The architecture targets low-cost FPGA deployments in edge AI, IoT, embedded surveillance, and other cyber-physical systems where area and energy efficiency are primary concerns.

Fig. 3 illustrates Architecture 1. Its key design principle is selective round unrolling: only a single Ascon permutation round is implemented in hardware and reused iteratively under the control of a lightweight finite-state machine (FSM). Reusing the same hardware resources across all twelve rounds significantly reduces LUT utilization. Although this increases the number of clock cycles per permutation, it substantially lowers area and power consumption.

The architecture follows the standard Ascon permutation flow while maximizing hardware reuse. The 320-bit state, comprising five 64-bit words, is stored in a register bank. During each iteration, the state is read, processed through the round datapath, and written back. The datapath performs round-constant addition, nonlinear substitution, and linear diffusion before updating the state registers. This iterative read–process–write mechanism enables efficient execution of all permutation rounds using a single round implementation.

The FSM controls round counting, round-constant injection, state updates, and completion signaling through a small set of control states. It also generates enable signals to activate only the required modules during each processing stage, minimizing control overhead.

The S-box layer, the most resource-intensive component of the Ascon permutation, is implemented using shared logic rather than fully parallel instances. The same S-box unit is reused across rounds, reducing LUT consumption while preserving functionality. Likewise, XOR networks, rotation units, and state-update logic are shared throughout the design, further reducing area requirements.

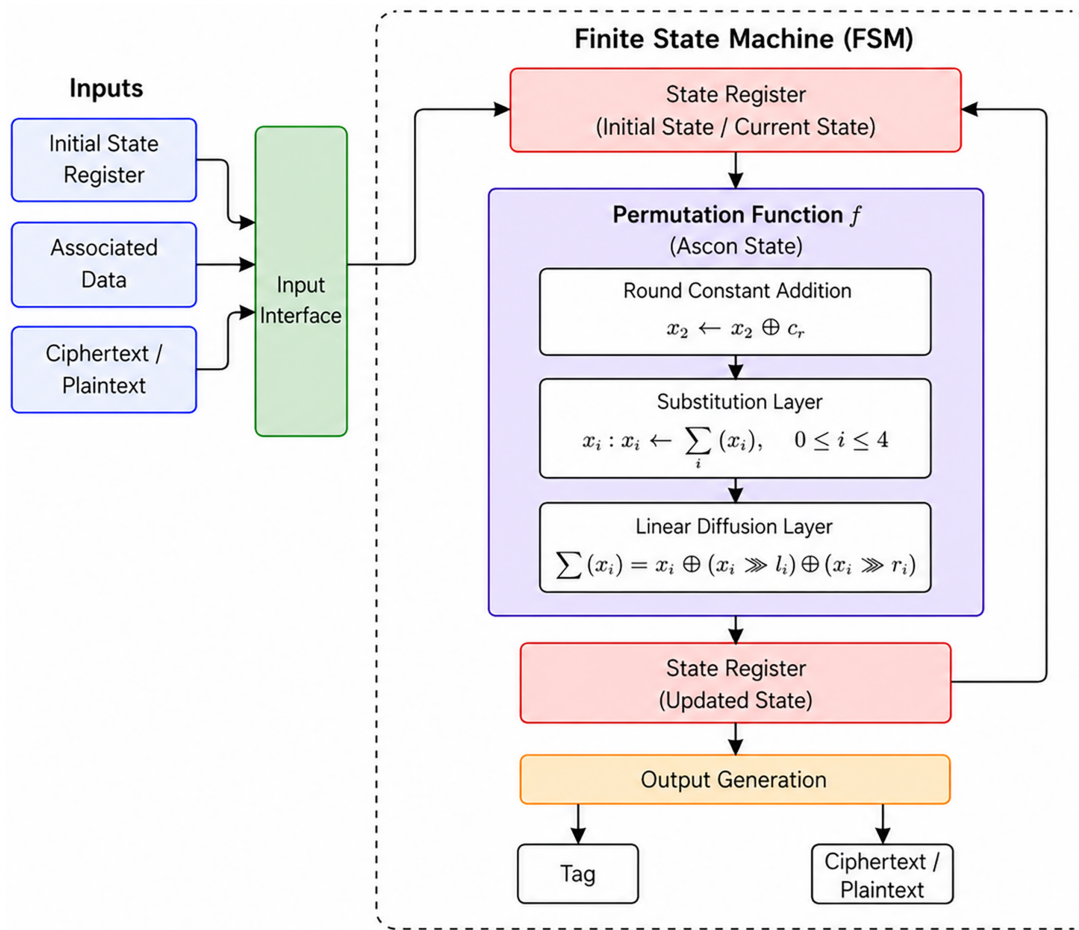


Figure 3: Area-efficient design for Ascon.

The linear diffusion layer uses compact XOR and rotation networks optimized for FPGA fabrics. Since rotations are largely implemented through routing resources, logic utilization remains low. Optimized XOR structures also reduce logic depth and routing congestion, improving timing performance.

To increase operating frequency with minimal area overhead, pipeline registers are inserted selectively between major functional blocks, particularly the substitution and diffusion layers. This limited pipelining shortens critical paths and aids timing closure without the cost of deep pipeline structures.

Although Architecture 1 provides lower TP, it achieves a significant reduction in hardware footprint. The increase in execution cycles is offset by savings in LUTs, registers, and routing resources, resulting in an improved TP/A ratio. Consequently, the architecture is well suited to battery-powered surveillance systems, industrial IoT nodes, wireless sensor networks, and embedded AI platforms where resource efficiency is prioritized over maximum TP.

In summary, Architecture 1 combines selective round unrolling, iterative processing, resource sharing, lightweight FSM control, and targeted pipelining to maintain Ascon's cryptographic functionality while substantially reducing hardware requirements. By balancing area, latency and TP, it provides an efficient FPGA-based solution for secure edge-computing applications.

4.3 Architecture 2: Low-Latency, High-Throughput Design

This architecture refines pipeline granularity and stage partitioning to achieve deterministic low latency for real-time edge workloads, an aspect that has received limited attention in existing Ascon FPGA implementations.

Architecture 2 targets real-time secure processing in edge AI and embedded systems where both low latency and high sustained TP are required. Unlike the fully unrolled baseline, which prioritizes TP, and Architecture 1, which emphasizes area efficiency through resource reuse, Architecture 2 adopts a multi-stage pipelined design that balances TP, latency, and resource utilization. The goal is to provide high-performance encryption suitable for mid-range FPGA platforms.

Fig. 4 illustrates Architecture 2. Its key design principle is pipeline-based round partitioning. Rather than fully unrolling all twelve Ascon rounds or executing them sequentially through a single round unit, the permutation is divided into multiple pipeline stages, each performing two consecutive rounds. Pipeline registers separate stages, allowing multiple data blocks to be processed simultaneously. This approach delivers high TP while requiring significantly fewer resources than a fully unrolled design.

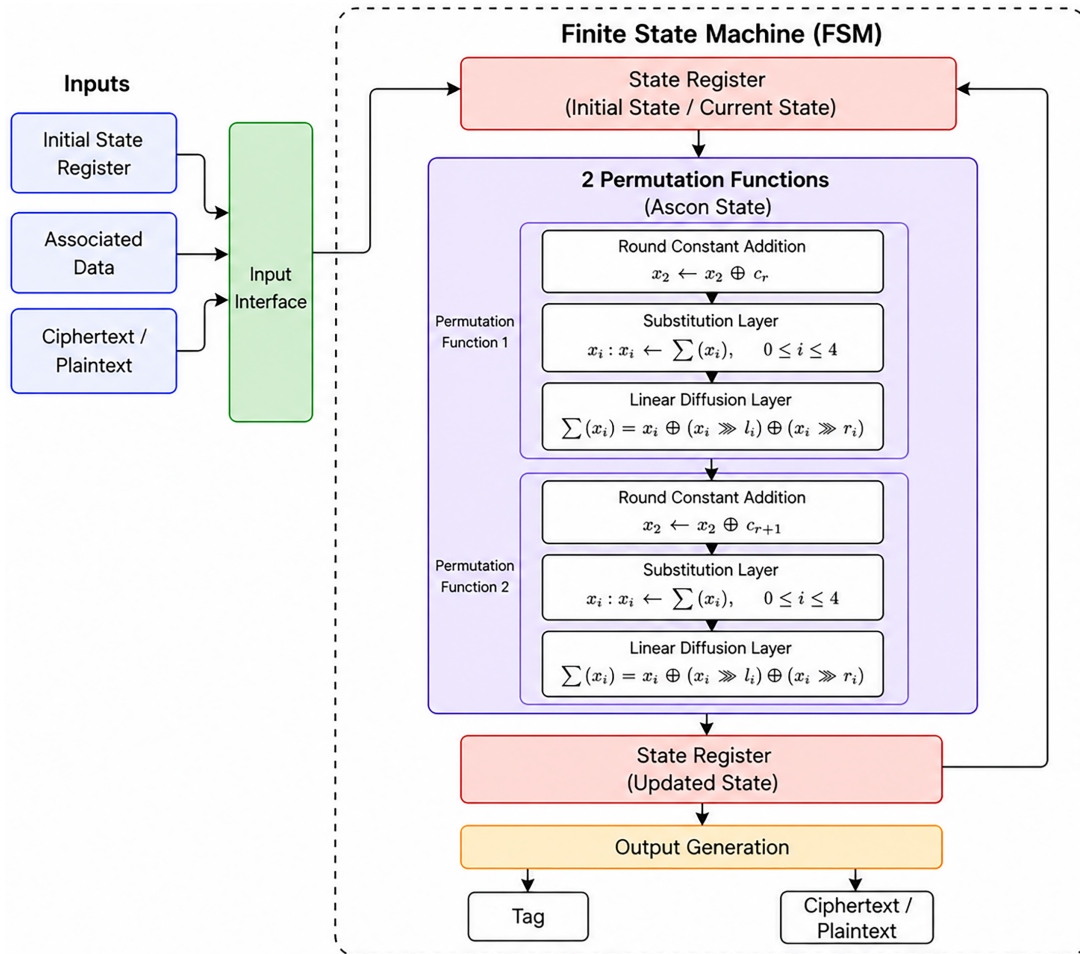


Figure 4: Low-latency, high-TP design for Ascon.

Within each stage, processing follows the standard Ascon sequence of round-constant addition, nonlinear substitution, and linear diffusion. Round constants are generated locally, while the substitution

and diffusion layers provide the required confusion and diffusion properties. After processing, the updated state is stored in a pipeline register and forwarded to the next stage.

The S-box layer uses a parallel architecture that balances performance and area efficiency. Instead of fully parallel hardware, a reduced number of S-box computation units is employed, lowering LUT consumption while maintaining sufficient TP. Resource allocation is carefully chosen to prevent the substitution layer from becoming a bottleneck.

The diffusion layer is implemented using compact XOR and rotation networks optimized for FPGA fabrics. Since rotations are largely realized through routing resources, additional logic overhead is minimal. Optimized XOR structures reduce critical-path delays and support higher operating frequencies.

Pipeline partitioning is central to the design. By limiting each stage to two rounds, combinational logic depth is reduced, critical paths are shortened, and timing closure is simplified. Pipeline registers isolate stages, enabling higher clock frequencies and providing predictable latency, which is essential for real-time applications.

A lightweight pipeline controller manages data movement, stage synchronization, round-constant generation, and valid signals. Because each stage performs a fixed set of rounds, the control logic is considerably simpler than that of iterative architectures, reducing overhead and improving timing performance.

A major advantage of Architecture 2 is continuous streaming operation. Once the pipeline is full, new data blocks can enter at regular intervals, enabling sustained encrypted output streams. This capability is particularly beneficial for edge AI applications generating continuous sensor, video, telemetry, or inference data that require immediate protection.

Compared with Architecture 1, the pipelined structure lowers latency by executing multiple rounds concurrently across different stages. At the same time, it avoids the large LUT, register, and routing overhead of the fully unrolled baseline. Consequently, Architecture 2 achieves an effective balance between TP, latency, and area, making it suitable for edge gateways, autonomous robots, industrial controllers, and embedded AI accelerators.

In summary, Architecture 2 combines strategic pipeline partitioning, moderate round unrolling, optimized S-box implementation, efficient diffusion networks, and lightweight control logic to deliver high-performance Ascon encryption without the resource costs of a fully unrolled architecture. The design provides an efficient FPGA-based solution for real-time secure processing in edge-computing and AI-enabled embedded systems.

5 Experimental Setup

To evaluate the performance and efficiency of the architectures, a comprehensive experimental setup is established using two FPGA platforms representing different classes of edge computing devices. The evaluation focuses on measuring hardware utilization, timing performance, and overall efficiency to assess the suitability of each architecture for real-time edge AI applications. Only Architecture 1 and Architecture 2 were synthesized and evaluated on the Kintex UltraScale and Spartan-7 platforms. The baseline architecture described in [Section 4.1](#) serves as a conceptual reference and was not implemented due to its excessive resource requirements.

5.1 Target FPGA Platforms

All implementations were synthesized and placed-and-routed using Xilinx Vivado 2023.1. The high-performance platform corresponds to the Xilinx Kintex UltraScale device XCKU035-1FFVA1156, while the resource-constrained platform corresponds to the Xilinx Spartan-7 device XC7S100-1FGGA676. These

platforms represent opposite ends of the FPGA design spectrum, one optimized for high performance and the other for cost-efficient, resource-constrained embedded deployment. All reported results are obtained from post-implementation timing and resource reports, ensuring accurate reflection of routing and placement effects.

5.2 Performance Metrics

The evaluation of the proposed architectures is based on a set of key performance metrics that reflect both hardware efficiency and real-time processing capability.

Area (LUTs): The primary measure of hardware utilization is the number of LUTs used by each design. LUT count provides an indication of the logic resources required for implementation and directly impacts the cost and scalability of the architecture on resource-constrained devices.

Latency: Latency is defined as the number of clock cycles required to process a single block of data. Lower latency is critical for real-time applications, particularly in edge AI systems where rapid response is required for decision-making.

Operating Frequency: The maximum achievable clock frequency (in MHz) is obtained from post-implementation timing analysis. Higher frequency enables faster data processing and contributes directly to improved TP.

Throughput (TP): TP represents the rate at which data is processed by the encryption engine, typically measured in Mb/s. It is computed based on the operating frequency, data width, and latency of the design. High TP is essential for applications involving continuous data streams, such as video processing in edge AI systems.

Throughput-to-Area Ratio (TP/A): To provide a balanced evaluation of performance and hardware efficiency, the TP/A ratio is used as a key metric. This metric captures how efficiently the architecture utilizes hardware resources to achieve high data processing rates. A higher TP/A ratio value indicates a more efficient design, particularly important for edge devices with limited resources.

5.3 Evaluation Methodology

The evaluation was conducted by implementing all proposed architectures on both FPGA platforms and extracting performance metrics from post-synthesis and post-implementation reports. The results are then analyzed to compare trade-offs between area, latency, and TP across architectures.

Special attention was given to identifying which architectures are best suited for specific edge AI scenarios. For example, designs with lower LUT utilization are evaluated for deployment on resource-constrained nodes, while high-TP designs are assessed for real-time applications such as video encryption and autonomous systems.

This methodology ensures that the proposed architectures are not only evaluated in isolation but also in the context of practical deployment requirements, providing a comprehensive understanding of their suitability for edge AI and real-time embedded systems.

The experimental setup provides a robust framework for evaluating the proposed architectures across diverse hardware environments. By combining cross-platform implementation, consistent tool flow, and comprehensive performance metrics, the study enables a detailed analysis of the design trade-offs and highlights the practical relevance of the proposed solutions for secure edge computing.

6 Results and Discussion

This section presents evaluation of Ascon hardware architectures on two FPGA families: the high-performance Xilinx Kintex UltraScale and the resource-constrained Xilinx 7-Series Spartan family. The analysis examines hardware utilization, latency, operating frequency, TP, and TP/A ratio, enabling a detailed understanding of the architectural trade-offs and their suitability for diverse edge AI deployment scenarios. This cross-platform evaluation reveals scaling behaviors that are not captured in prior Ascon FPGA studies, which typically report results on a single device. By analyzing TP, latency, and TP/A across two contrasting FPGA families, our work provides new insights into how established Ascon design strategies translate to practical edge AI deployment scenarios.

6.1 Performance on the Kintex UltraScale Platform

Table 1 summarizes the results of the implementation on the Xilinx Kintex UltraScale. The implementation results demonstrate the behavior of the proposed architectures in a high-performance FPGA environment with abundant logic resources and high achievable clock frequencies. On this platform, Architecture 2 achieves the highest TP, reaching 624.57 Mb/s, which is significantly higher than the 408.21 Mb/s achieved by Architecture 1. This improvement is primarily due to the reduced latency of Architecture 2 (24 cycles compared to 42 cycles) and its selective unrolling strategy, which enables efficient data processing even at a slightly lower operating frequency of 233.30 MHz, compared to 267.89 MHz for Architecture 1. The combination of moderate parallelism and reduced critical-path complexity allows Architecture 2 to exploit the UltraScale routing fabric effectively.

Table 1: Implementation results on Kintex UltraScale XCKU035.

Design	LUTs	Latency	Frequency (MHz)	Time (ns)	TP (Mb/s)	TP/A (Mb/s per LUT)
Arch. 1	1804	42	267.89	156.78	408.21	0.226
Arch. 2	2523	24	233.30	102.47	624.57	0.247

Note: LUT = Look-Up Table; TP = Throughput; TP/A = Throughput-to-Area Ratio.

In terms of hardware utilization, Architecture 1 is the most compact design, requiring only 1804 LUTs, whereas Architecture 2 consumes 2523 LUTs. This difference reflects the additional logic required to support selective unrolling and optimized dataflow in Architecture 2. When considering the TP/A ratio, Architecture 2 again demonstrates superior efficiency, achieving a TP/A value of 0.247 compared to 0.226 for Architecture 1. This indicates that Architecture 2 delivers more TP per unit of hardware, making it the most efficient design on the UltraScale platform when both performance and area are taken into account.

Overall, the UltraScale results show that Architecture 2 provides the best balance between TP and hardware efficiency, while Architecture 1 serves as a useful reference for understanding the performance gains achieved through selective unrolling and datapath optimization.

6.2 Performance on the Spartan-7 Platform

Table 2 summarizes the results of the implementation on the Spartan-7 Platform. The results obtained on the Spartan-7 platform provide insights into how the architectures scale to a resource-constrained FPGA family. As expected, both architectures exhibit lower operating frequencies and reduced TP compared to the UltraScale platform due to the more limited routing and logic capabilities of the Spartan-7 family. Architecture 1 operates at 200.63 MHz, while Architecture 2 operates at 141.66 MHz, reflecting the tighter timing constraints of the device.

Table 2: Implementation results on Spartan-7 XC7S100-1FGGA676.

Design	LUTs	Latency	Frequency (MHz)	Time (ns)	TP (Mb/s)	TP/A (Mb/s per LUT)
Arch. 1	2027	42	200.63	209.34	305.72	0.150
Arch. 2	2865	24	141.66	169.41	377.78	0.131

Note: LUT = Look-Up Table; TP = Throughput; TP/A = Throughput-to-Area Ratio.

Despite the reduced frequency, Architecture 2 again achieves higher TP, reaching 377.78 Mb/s, compared to 305.72 Mb/s for Architecture 1. This mirrors the trend observed on the UltraScale platform: the reduced latency of Architecture 2 (24 cycles vs. 42 cycles) compensates for its lower clock frequency. In terms of area utilization, Architecture 1 remains the more compact design, requiring 2027 LUTs, while Architecture 2 consumes 2865 LUTs. This reinforces the role of Architecture 1 as a minimal-resource baseline and highlights the additional logic overhead introduced by Architecture 2's selective unrolling.

The TP/A metric on the Spartan-7 platform shows a different trend compared to the UltraScale results. Architecture 1 achieves a TP/A value of 0.150, while Architecture 2 achieves 0.131. This indicates that Architecture 1 is slightly more efficient on the Spartan-7 device, as the performance advantage of Architecture 2 is reduced on a low-frequency platform while its area overhead remains significant. These results demonstrate that both architectures scale effectively to the Spartan-7 family, but their relative advantages shift depending on the available hardware resources and achievable clock speeds.

6.3 Comparative Analysis and Trade-Offs

The combined results across both FPGA families reveal clear architectural trade-offs that are essential for selecting the most appropriate design for a given edge AI deployment scenario. Architecture 1 consistently provides the lowest area utilization on both platforms, making it well-suited for resource-constrained edge nodes where minimizing LUT usage is critical and moderate TP is acceptable. Its compact structure ensures compatibility with low-cost and low-power devices.

Architecture 2, in contrast, consistently delivers higher TP due to its reduced latency and selective unrolling strategy. It is best suited for mid-range edge devices that require higher data rates but can accommodate a moderate increase in hardware utilization. The efficiency of Architecture 2 is particularly evident on the UltraScale platform, where its TP/A value surpasses that of Architecture 1. However, on the Spartan-7 platform, the reduced operating frequency diminishes its performance advantage, making Architecture 1 slightly more efficient in terms of TP per unit area.

6.4 Comparison with Existing Implementations

To contextualize the performance of the proposed architectures within the broader Ascon hardware landscape, Table 3 compares our results against representative FPGA implementations from [22,23]. Unlike prior work, which primarily targets either high-performance FPGA accelerators or generic lightweight cryptography benchmarks, our architectures are explicitly optimized for real-time edge AI applications, including smart surveillance, autonomous embedded systems, and edge AI inference nodes.

High-TP designs such as those in [22] achieve multi-gigabit performance by fully unrolling Ascon permutation rounds. While these architectures deliver exceptional TP, their LUT usage and long critical paths make them suitable for data-center-class accelerators, not for embedded edge AI devices where silicon area is tightly constrained.

Table 3: Comparison with existing FPGA architectures.

Ref.	Platform	Area (LUTs)	Frequency (MHz)	TP (Mb/s)	TP/A (Mp/s per LUT)
[22]	Kintex-7	2958	92.59	5925.92	2.01
[22]	Artix-7	2957	55.25	3535.91	1.19
[23]	Spartan-6	2060	206.26	315.20	0.153

The Spartan-6 design in [23] represents the opposite end of the spectrum: it minimizes area but achieves only 315 Mb/s TP, with a TP/A ratio of 0.153. This design is not optimized for any specific application domain and does not consider the latency or TP requirements of real-time edge AI pipelines.

In contrast, the proposed architectures in this work are engineered to align with the operational characteristics of edge AI systems:

7 Application Mapping

To demonstrate the practical relevance of cryptographic architectures, this section maps the design characteristics to representative edge AI applications. The analysis incorporates quantitative results from both FPGA platforms, ensuring that architectural choices are grounded in measurable performance indicators such as TP, latency, LUT utilization, and TP/A ratio. This application-driven perspective highlights how each architecture aligns with system-level requirements in real-world edge deployments.

The application mapping presented in this section is analytical and is based solely on the measured hardware performance metrics of the proposed architectures. The study does not include real deployment experiments. Instead, the mapping provides a performance-driven assessment of architectural suitability for representative edge scenarios.

7.1 Smart Surveillance Systems (Edge AI Cameras)

Smart surveillance systems process continuous high-resolution video streams, often ranging from 50–200 Mb/s depending on the resolution and compression settings. These systems require encryption engines capable of sustaining high data rates without introducing additional latency into the video analytics pipeline.

The results on the Kintex UltraScale platform show that Architecture 2 achieves a TP of 624.57 Mb/s with a latency of 24 cycles and an operating frequency of 233.30 MHz. This TP comfortably exceeds the bandwidth requirements of typical 1080p and even 4K surveillance streams, ensuring that encryption does not become a bottleneck. In contrast, Architecture 1 achieves 408.21 Mb/s, which remains sufficient for many surveillance workloads but provides less headroom for multi-stream or high-frame-rate systems.

Although Architecture 2 consumes more LUTs (2523 compared to 1804 for Architecture 1), the performance benefits justify this overhead in high-performance surveillance nodes. Its TP/A value of 0.247 on UltraScale further demonstrates its efficiency in delivering high TP per unit area. Therefore, Architecture 2 is the most suitable choice for smart surveillance systems, where real-time responsiveness and high data-rate encryption are essential.

7.2 Edge AI Inference Nodes

Edge AI inference nodes typically handle lower-bandwidth sensor data streams, often in the range of 1–20 Mb/s, and operate under strict hardware and power constraints. In such systems, the encryption engine must minimize resource usage to coexist with AI inference modules on the same FPGA.

Across both platforms, Architecture 1 consistently demonstrates the lowest LUT utilization, requiring only 1804 LUTs on UltraScale and 2027 LUTs on Spartan-7. Its TP of 408.21 Mb/s on UltraScale and 305.72 Mb/s on Spartan-7 far exceeds the bandwidth requirements of typical inference workloads, providing ample performance margin while maintaining a compact hardware footprint.

The TP/A ratio further reinforces this conclusion: on the Spartan-7 platform, Architecture 1 achieves a TP/A of 0.150, outperforming Architecture 2's 0.131. This indicates that Architecture 1 delivers more TP per unit area in resource-constrained environments. As a result, Architecture 1 is the preferred choice for embedded inference nodes, where minimizing hardware footprint and power consumption is more important than achieving maximum TP.

7.3 Autonomous Embedded Systems

Autonomous embedded systems, such as UAVs, autonomous vehicles, and robotic platforms, rely on real-time processing of multi-sensor data. These systems often generate data rates between 100–400 Mb/s when combining camera feeds, LiDAR scans, and inertial measurements. Encryption must therefore support high TP while maintaining low latency to avoid interfering with time-critical control loops.

Architecture 2 again demonstrates strong suitability for this domain. On the UltraScale platform, its TP of 624.57 Mb/s and latency of 24 cycles ensure that it can handle multi-sensor fusion workloads without introducing delays. Even on the Spartan-7 platform, Architecture 2 achieves 377.78 Mb/s, which is sufficient for many mid-range autonomous systems. Although it requires more LUTs (2865 on Spartan-7), the performance gains justify the additional resource usage in systems where safety and responsiveness are paramount.

Architecture 1, while more area-efficient, provides lower TP and higher latency, making it less suitable for high-frequency sensor pipelines. Therefore, Architecture 2 is the most appropriate choice for autonomous embedded systems that demand real-time secure communication and can accommodate moderate hardware utilization.

7.4 Discussion and Insights

The application mapping demonstrates that the suitability of each architecture depends strongly on the performance and resource requirements of the target system. Architecture 1 excels in scenarios where hardware resources are limited, offering the lowest LUT utilization across both FPGA platforms while still providing TP well above the needs of typical sensor-driven applications. In contrast, Architecture 2 consistently delivers higher TP and lower latency, making it ideal for performance-critical applications such as smart surveillance and autonomous systems.

The results also reveal that architectural efficiency is platform-dependent. Architecture 2 achieves the highest TP/A on the UltraScale device, while Architecture 1 is more efficient on the Spartan-7 platform. This highlights the importance of selecting architectures based not only on application requirements but also on the capabilities of the underlying hardware.

Overall, the proposed architectures provide a flexible design space that enables system designers to balance performance, hardware area, and latency according to the requirements of diverse edge AI applications.

8 Conclusion and Future Work

This paper presented efficient FPGA-based cryptographic architectures for edge AI and real-time embedded systems, addressing the challenges of latency, TP, and hardware resource constraints. By exploring multiple architectural designs, this work demonstrated that it is possible to achieve a balanced trade-off between performance and hardware efficiency, enabling secure data processing across a wide range of edge computing environments.

The architectures were implemented and evaluated on two FPGA platforms representing both high-performance and resource-constrained devices. The results showed that Architecture 2 achieves the highest TP on both platforms, making it suitable for data-intensive and real-time applications such as smart surveillance and autonomous systems. In contrast, Architecture 1 provides the most compact and area-efficient solution, making it ideal for resource-constrained edge nodes and embedded inference systems. The use of TP/A metric enabled a comprehensive evaluation of design efficiency, revealing platform-dependent trade-offs that guide architecture selection.

A key contribution of this work is the demonstration of adaptive deployment, where different cryptographic architectures can be selected based on application-specific requirements. This flexibility is particularly important in heterogeneous edge computing ecosystems, where devices vary widely in computational capabilities, power budgets, and performance demands. By mapping architectural designs to real-world applications, the paper establishes a clear connection between hardware-level optimizations and system-level requirements.

The results confirm that FPGA-based cryptographic implementations provide a practical and scalable solution for integrating security into edge AI systems without compromising real-time performance. This makes the proposed architectures highly relevant for emerging applications in smart surveillance, industrial automation, and autonomous embedded systems.

Future work will focus on power and energy optimization, which is critical for battery-powered edge devices. Incorporating power-aware design techniques and evaluating energy efficiency will provide a more comprehensive understanding of system performance. Additionally, transitioning from FPGA prototypes to ASIC implementations will be explored to enable large-scale deployment in commercial edge devices. The integration of post-quantum cryptographic algorithms represents another important research direction, particularly in the context of future-proof security for edge systems.

Another important direction is to extend the proposed FPGA architectures by integrating them into complete edge AI platforms and evaluating them under representative real-time application workloads, including encrypted video analytics, sensor-fusion pipelines, and end-to-end embedded AI deployments. Such system-level validation will provide additional insight into the interaction between the proposed cryptographic architectures, AI accelerators, and communication subsystems under realistic operating conditions, thereby complementing the architectural evaluation presented in this work.

Acknowledgement: The authors would like to thank their respective institutions for providing the facilities, computational resources, and research environment that supported this research.

Funding Statement: This research was funded by the Ministry of Trade, Industry and Energy (MOTIE), Republic of Korea, through the Korea Institute for Advancement of Technology, under the project Development of an International Standardization and Sustainability Integration Framework for AI Industry Internalization and Global Competitiveness Enhancement (Grant No. RS-2025-07372968). The funding was awarded to Young Im Cho.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Makhkamov Bakhtiyor Shukhratovich, Akmalbek Abdusalomov and Young Im Cho; methodology, Makhkamov Bakhtiyor Shukhratovich, Akmalbek Abdusalomov and Jasur Sevinov; software, Makhkamov Bakhtiyor Shukhratovich, Botirov Sokhibjon Rustam Ugli and Jasur Sevinov; validation, Makhkamov Bakhtiyor Shukhratovich, Akmalbek Abdusalomov, Akhmedova Nodira Aminjanovna and Young Im Cho; formal analysis, Makhkamov Bakhtiyor Shukhratovich, Akmalbek Abdusalomov and Alpamis Kutlimuratov; investigation, Makhkamov Bakhtiyor Shukhratovich, Akhmedova Nodira Aminjanovna, Botirov Sokhibjon Rustam Ugli and Jasur Sevinov; resources, Akmalbek Abdusalomov, Boburjon Vafoev and Yodgorkhon Ilkhamova; data curation, Makhkamov Bakhtiyor Shukhratovich, Botirov Sokhibjon Rustam Ugli and Jasur Sevinov; writing—original draft preparation, Makhkamov Bakhtiyor Shukhratovich and Akmalbek Abdusalomov; writing—review and editing, Akmalbek Abdusalomov, Alpamis Kutlimuratov, Boburjon Vafoev, Yodgorkhon Ilkhamova and Young Im Cho; visualization, Makhkamov Bakhtiyor Shukhratovich and Jasur Sevinov; supervision, Akmalbek Abdusalomov and Young Im Cho; project administration, Akmalbek Abdusalomov; funding acquisition, Young Im Cho. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: Data available on request from the authors. The data that support the findings of this study are available from the Corresponding Author, Young Im Cho, upon reasonable request.

Ethics Approval: Not applicable. This study did not involve human participants, animals, or identifiable personal data. Therefore, ethical approval and informed consent were not required.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Boubaker S, Alsubaei FS, Said Y, Ahmed HE. Lightweight cryptography for connected vehicles communication security on edge devices. *Electronics*. 2023;12(19):4090. doi:10.3390/electronics12194090.
2. Rasheed AM, Kumar RMS. Efficient lightweight cryptographic solutions for enhancing data security in healthcare systems based on IoT. *Front Comput Sci*. 2025;7:1522184. doi:10.3389/fcomp.2025.1522184.
3. Rajput GS, Thakur R, Tiwari R. VLSI implementation of lightweight cryptography technique for FPGA-IOT application. *Mater Today Proc*. 2023;2020:176. doi:10.1016/j.matpr.2023.03.486.
4. Singh S, Sharma PK, Moon SY, Park JH. Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions. *J Ambient Intell Humaniz Comput*. 2024;15(2):1625–42. doi:10.1007/s12652-017-0494-4.
5. Abdullah AM. Advanced encryption standard (AES) algorithm to encrypt and decrypt data. *Cryptogr Netw Secur*. 2017;16(1):11. doi:10.1007/springerreference_73130.
6. Hellman ME. An overview of public key cryptography. *IEEE Commun Mag*. 2002;40(5):42–9. doi:10.1109/mcom.2002.1006971.
7. Ajmi H, Zayer F, Belgacem H. In-memory computing architecture for efficient hardware security. In: *Proceedings of the 2024 IEEE 7th International Conference on Advanced Technologies, Signal and Image Processing (ATSIP)*; 2024 Jul 11–13; Sousse, Tunisia. p. 71–6. doi:10.1109/atsip62566.2024.10638850.
8. Karakchi R, Stahle-Smith R, Chinnasami N, Yu T. Toward a lightweight, scalable, and parallel secure encryption engine. *arXiv:2506.15070*. 2025.
9. Cířik P, Ricci S, Dobířš P, Hajný J, Malina L, Havlín J. Quantum-resistant hardware-accelerated IoT traffic encryptor. *Internet Things*. 2025;31(5):101554. doi:10.1016/j.iot.2025.101554.
10. Min K, Jeon S, Lee JJ, Lee W. TT-SEAL: TTD-aware selective encryption for adversarially-robust and low-latency edge AI. *arXiv:2602.22238*. 2026.
11. Dobraunig C, Eichlseder M, Mendel F, Schläffer M. Ascon v1.2: lightweight authenticated encryption and hashing. *J Cryptol*. 2021;34(3):33. doi:10.1007/s00145-021-09398-9.
12. Zhong Y, Gu J. Lightweight block ciphers for resource-constrained environments: a comprehensive survey. *Future Gener Comput Syst*. 2024;157(9):288–302. doi:10.1016/j.future.2024.03.054.

13. Bogdanov A, Knudsen LR, Leander G, Paar C, Poschmann A, Robshaw MJB, et al. PRESENT: an ultra-lightweight block cipher. In: *Cryptographic Hardware and Embedded Systems—CHES 2007*. Berlin/Heidelberg, Germany: Springer; 2007. p. 450–66. doi:10.1007/978-3-540-74735-2_31.
14. Banik S, Pandey SK, Peyrin T, Sasaki Y, Sim SM, Todo Y. GIFT: a small present: towards reaching the limit of lightweight encryption. In: *Cryptographic Hardware and Embedded Systems—CHES 2017*. Cham, Switzerland: Springer; 2017. p. 321–45. doi:10.1007/978-3-319-66787-4_16.
15. Zhang W, Bao Z, Lin D, Rijmen V, Yang B, Verbauwhede I. RECTANGLE: a bit-slice lightweight block cipher suitable for multiple platforms. *Sci China Inf Sci*. 2015;58(12):1–15. doi:10.1007/s11432-015-5459-7.
16. Goyal TK, Sahula V, Kumawat D. Energy efficient lightweight cryptography algorithms for IoT devices. *IETE J Res*. 2022;68(3):1722–35. doi:10.1080/03772063.2019.1670103.
17. National Institute of Standards and Technology (NIST). NIST special publication 800-232: lightweight cryptography standard: Ascon for authenticated encryption. Gaithersburg, MD, USA: NIST. 2024 [cited 2026 Jan 1]. Available from: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-232.pdf>.
18. Ahmed S, Ahmad N, Ali Shah N, Mustafa Abro GE, Wijayanto A, Hirsi A, et al. Lightweight AES design for IoT applications: optimizations in FPGA and ASIC with DFA countermeasure strategies. *IEEE Access*. 2025;13:22489–509. doi:10.1109/access.2025.3533611.
19. Zhao Y, Yu X, Sun Y, Kuang H, Han J. DNA-HHE: dual-mode near-network accelerator for hybrid homomorphic encryption on the edge. *arXiv:2512.18589*. 2025.
20. Gross H, Wenger E, Dobraunig C, Ehrenhöfer C. Ascon hardware implementations and side-channel evaluation. *Microprocess Microsyst*. 2017;52:470–9. doi:10.1016/j.micpro.2016.10.006.
21. Khan S, Inayat K, Bin Muslim F, Ali Shah Y, Atif Ur Rehman M, Khalid A, et al. Securing the IoT ecosystem: ASIC-based hardware realization of Ascon lightweight cipher. *Int J Inf Secur*. 2024;23(6):3653–64. doi:10.1007/s10207-024-00904-1.
22. Ahmet M. High-performance FPGA implementations of lightweight ASCON-128 and ASCON-128a with enhanced throughput-to-area efficiency. In: *Proceedings of the 2024 17th International Conference on Information Security and Cryptology (ISCTürkiye)*; 2024 Oct 16–17; Ankara, Türkiye. p. 1–7. doi:10.1109/isctrkiye64784.2024.10779273.
23. Khan S, Lee WK, Hwang SO. Scalable and efficient hardware architectures for authenticated encryption in IoT applications. *IEEE Internet Things J*. 2021;8(14):11260–75. doi:10.1109/jiot.2021.3052184.