DOI: 10.21917/ijme.2023.0281

SECURED HARDWARE PLATFORM FOR FPGA AI MODEL PROTECTION

S. Aramuthakannan[1], S. Lokesh[2], R. Kumar[3] and P. Gajendran[4]

[1,4]*Department of Mathematics, PSG Institute of Technology and Applied Research, India*
[2]*Department of Computer Science and Engineering, PSG Institute of Technology and Applied Research, India*
[3]*Department of Computer Science and Engineering, Sri Ramakrishna Institute of Technology, India*

Abstract

The widespread adoption of Field-Programmable Gate Arrays (FPGAs) in deploying Artificial Intelligence (AI) models has ushered in a new era of computational efficiency. However, the vulnerabilities associated with these platforms have raised concerns regarding the protection of sensitive AI models from malicious attacks. This study addresses the pressing need for a secured hardware platform to safeguard FPGA-based AI models, employing Deep Neural Networks (DNNs) as a robust defense mechanism. As FPGAs become integral to AI model deployment, the risk of unauthorized access and tampering increases. Existing security measures often fall short in providing comprehensive protection, leaving AI models vulnerable to exploitation. This research aims to bridge this gap by developing a novel hardware platform that integrates DNNs to fortify the security of FPGA-based AI models While previous studies have explored FPGA-based AI models and security measures independently, a significant research gap exists in the integration of DNNs specifically tailored for protecting these models. This study fills this void by proposing a holistic solution that combines the adaptability of FPGAs with the robustness of DNNs to create a secure and resilient hardware platform. The research employs a two-fold methodology, starting with the design and implementation of a secure FPGA architecture. Subsequently, DNNs are integrated into the hardware platform to detect and respond to potential security threats. The model is trained on diverse datasets to ensure adaptability to various AI applications. Preliminary results showcase a significant enhancement in the security posture of FPGA-based AI models. The integrated DNNs effectively identify and mitigate potential threats, providing a robust layer of defense against unauthorized access and tampering.

Keywords:

FPGA, AI Model Protection, Deep Neural Networks, Hardware Security, Threat Detection

1. INTRODUCTION

In recent years, the integration of Field-Programmable Gate Arrays (FPGAs) in deploying Artificial Intelligence (AI) models has demonstrated unparalleled computational efficiency, making them a cornerstone in various applications [1]. However, this widespread adoption has brought forth a new set of challenges, primarily centered around the security of FPGA-based AI models [2]. The inherent flexibility of FPGAs makes them susceptible to unauthorized access and tampering, necessitating innovative solutions to fortify their security [3].

The utilization of FPGAs in AI introduces unique challenges that demand immediate attention. Traditional security measures often prove inadequate in the face of dynamic threats, leaving AI models vulnerable to exploitation [4]. Additionally, the rapid evolution of AI technologies further complicates the development of robust security mechanisms for FPGA-based implementations.

The core problem addressed by this research is the need for a comprehensive and adaptable security solution tailored specifically for FPGA-based AI models. Current approaches fall short in providing a holistic defense, prompting the exploration of a novel hardware platform that integrates Deep Neural Networks (DNNs) to enhance the security posture of these systems.

The primary objectives of this research are twofold. Firstly, to design and implement a secure FPGA architecture capable of withstanding a spectrum of potential security threats. Secondly, to integrate DNNs into the hardware platform, leveraging their inherent ability to learn and adapt, thereby creating an intelligent defense mechanism against unauthorized access and tampering.

This research introduces a novel approach by combining the flexibility of FPGAs with the learning capabilities of DNNs to address the security challenges posed by FPGA-based AI models. The integration of DNNs into the hardware platform represents a pioneering step towards achieving a comprehensive and adaptive security solution. The contributions of this study extend beyond traditional FPGA security measures, presenting a paradigm shift in safeguarding AI models deployed on these platforms.

2. RELATED WORKS

Previous research has explored various security measures for FPGAs, focusing on encryption and access control. However, the application of these measures to AI contexts, especially considering the dynamic nature of neural networks, remains a research gap that this study seeks to address [6].

There is a body of work on utilizing Deep Learning techniques for enhancing hardware security. While these studies have shown promise in other domains, their application to FPGA-based AI models is limited. This research aims to build upon these findings and tailor them to the specific challenges posed by FPGAs in AI applications [7].

Existing literature highlights the importance of adaptive defense mechanisms in the context of cybersecurity. This research aligns with these findings by incorporating Deep Neural Networks into the FPGA architecture, allowing for real-time learning and adaptation to emerging threats [8].

Several studies have identified vulnerabilities specific to FPGA-based AI models, including side-channel attacks and unauthorized reprogramming. This research synthesizes these findings to develop a comprehensive security solution that mitigates these vulnerabilities and ensures the integrity of deployed AI models [9].

While there is a growing interest in deploying Deep Neural Networks for cybersecurity, their integration into FPGA-based systems is relatively unexplored. This study contributes by demonstrating the feasibility and effectiveness of embedding DNNs directly into the FPGA hardware for robust threat detection and prevention [10].

By building upon these related works, this research aims to provide a holistic and innovative solution that addresses the security challenges unique to FPGA-based AI models, ultimately contributing to the advancement of secure and reliable AI deployments in diverse applications [11].

3. METHODS

The proposed method is a multifaceted approach aimed at fortifying the security of FPGA-based AI models through the integration of Deep Neural Networks (DNNs) into the hardware architecture. The method comprises two main components: the design and implementation of a secure FPGA architecture and the integration of DNNs for real-time threat detection and response.

The first step involves the development of a secure FPGA architecture. This includes the implementation of encryption mechanisms to protect the integrity of the AI model and access control measures to prevent unauthorized tampering. Additionally, the architecture is designed to withstand common security threats such as side-channel attacks and unauthorized reprogramming. By establishing a robust foundation, the secure FPGA architecture serves as the initial line of defense against potential security breaches.

The second component focuses on enhancing the security posture through the integration of DNNs. These neural networks are trained to recognize patterns indicative of security threats, creating an intelligent and adaptive defense mechanism. The DNNs operate directly within the FPGA hardware, allowing for real-time analysis of incoming data and immediate response to potential security incidents. This integration enables the system to learn and adapt to emerging threats, providing a dynamic layer of defense that goes beyond static security measures.

The DNNs are trained on diverse datasets encompassing normal and potentially malicious patterns. This training process enables the neural networks to differentiate between legitimate and suspicious activities. The adaptive nature of DNNs ensures that the system continues to evolve and refine its threat detection capabilities over time, staying resilient against evolving security threats. During operation, the integrated DNNs continuously analyze the behavior of the FPGA-based AI model and incoming data. Any deviation from normal patterns triggers an immediate response, which may include isolating the affected portion of the system, alerting administrators, or implementing corrective measures. This real-time threat detection and response mechanism significantly reduces the window of vulnerability, enhancing the overall security of the FPGA-based AI deployment.

3.1 SECURE FPGA ARCHITECTURE USING DNN

The Secure FPGA architecture using DNN involves integrating a DNN directly into the architecture of a FPGA to enhance the security of AI models deployed on these platforms. This integration aims to create a robust defense mechanism against potential threats and attacks. The approach is the development of a secure FPGA architecture. This entails implementing encryption techniques to protect the confidentiality and integrity of the AI model. Access control measures are also put in place to regulate and authenticate interactions with the FPGA. These security measures are crucial for preventing unauthorized access, tampering, or extraction of sensitive information from the FPGA. A DNN is embedded directly into the FPGA architecture. The DNN serves as an intelligent layer responsible for real-time analysis of the system's behavior. It operates alongside the conventional FPGA components, continuously monitoring data flow and the execution of AI models.

Prior to deployment, the DNN undergoes a training phase using diverse datasets that encompass normal system behavior and potential security threats. This training equips the DNN with the ability to recognize patterns associated with malicious activities. The DNN learns to distinguish between normal and anomalous behavior, enhancing its capacity for threat detection. During the operational phase, the integrated DNN monitors the FPGA-based AI model and the incoming data in real-time. Any deviation from the learned normal patterns triggers the DNN to classify the behavior as potentially malicious. This real-time threat detection is crucial for identifying and responding to security incidents as they occur, reducing the impact of potential breaches.

The adaptive nature allows it to evolve and improve its threat detection capabilities over time. As the system encounters new data and potential threats, the DNN updates its understanding of normal and malicious patterns, ensuring an adaptive defense mechanism that remains effective against emerging security challenges. By combining a secure FPGA architecture with the capabilities of a Deep Neural Network, this approach provides a comprehensive security solution.

$$C=\text{Encrypt}(P,K) \tag{1}$$

where C is the encrypted data, P is the plaintext data, and K is the encryption key.

$$\text{AG} = \text{Authenticate}(\text{UserCredentials}, \text{AccessRights}) \tag{2}$$

where AG is a binary indicator of whether access is granted based on user credentials and access rights.

Forward pass in a basic neural network layer:

$$Z=W\cdot X+B \tag{3}$$

$$A=\text{RELU}(Z) \tag{4}$$

where W is the weight matrix, X is the input data, B is the bias vector, and RELU is the activation function.

Training the DNN involves minimizing a loss function:

$$\text{Loss}=\text{ComputeLoss}(Y_a,Y_p) \tag{5}$$

where Y_a is the actual output, Y_p is the predicted output, α is the learning rate.

Optimization involves adjusting the weights and biases using backpropagation and gradient descent:

$$W_{new} = W_{old} - \alpha\partial Loss/\partial W_{old} \tag{6}$$

where:

W_{new} is the updated weight matrix,

W_{old} is the current weight matrix,

α is the learning rate,

$\partial Loss/\partial W_{old}$ is the gradient of the loss function with respect to the weights.

$$\text{AnomalyScore} = \text{Score}(X_{curr},X_n) \tag{7}$$

where X_{curr} is the current system behavior, and X_n represents normal behavior.

3.2 ADAPTIVE RESPONSE

The adaptation process is typically governed by adjusting the neural network's weights based on new information and experiences. The weights (W) of the DNN are updated using gradient descent to minimize the loss function over time. This process allows the DNN to adapt to new patterns and information. To further enhance adaptation, an adaptive learning rate (α) can be introduced. This ensures that the model adjusts the step size in the weight update process based on historical information, preventing large updates that may destabilize the system.

$$\alpha_{new} = \sigma(\alpha_{old}, Loss) \quad (8)$$

where:

α_{new} is the updated learning rate,

α_{old} is the current learning rate,

Loss represents the historical record of loss values.

The DNN can be designed for online learning, allowing it to adapt in real-time as new data becomes available. The weight update occurs incrementally for each new data point.

Algorithm 1: Secure FPGA Architecture using DNN

a) Initialize FPGA architecture with secure features such as encryption mechanisms and access control.
b) Initialize DNN parameters (weights and biases) randomly.
c) Collect diverse datasets representing normal and potentially malicious system behavior.
d) Split the dataset into training and validation sets.
e) Train the DNN on the training set using backpropagation and gradient descent:
   i) Forward pass
   ii) Compute Loss
   iii) Backward pass
f) Integrate the trained DNN into the FPGA architecture.
g) Implement encryption mechanisms for AI model protection.
h) Implement access control measures to regulate interactions with the FPGA.
i) During operation, continuously monitor incoming data and AI model behavior.
j) For each data point
   i) Perform a forward pass through the integrated DNN.
   ii) Compute an anomaly score based on the difference between predicted and expected behavior.
k) If the anomaly score exceeds a predefined threshold, trigger a threat detection response.
   i) Continuously update DNN parameters for adaptive learning:
   ii) Adjust learning rate based on historical loss values.
   iii) Incrementally update weights for each new data point.
   iv) Regularly update and retrain the DNN using new datasets to stay resilient to evolving threats.
   v) Monitor system logs for potential security incidents.
l) End
m) End

4. EXPERIMENTS

In experimental settings, we utilized the Xilinx Vivado Design Suite as the primary simulation tool for developing and testing the proposed Secure FPGA Architecture using DNN. The simulation environment offered a comprehensive platform for FPGA design, allowing us to implement and evaluate the secure architecture seamlessly. The experiments were conducted on a high-performance computing cluster comprising Intel Xeon processors and NVIDIA GPUs, ensuring efficient execution of the simulations and training processes. The FPGA implementation was carried out on Xilinx FPGAs, with particular emphasis on maintaining compatibility and optimizing performance for real-world deployment scenarios.

To assess the performance of our proposed method, we employed a set of well-established metrics. These included accuracy, false positive rate, and detection latency. Accuracy measured the ability of the integrated DNN to correctly identify normal and anomalous behavior. The false positive rate quantified the occurrence of false alarms, crucial for evaluating the reliability of the security system. Detection latency gauged the speed at which potential threats were identified and responded to. In our comparative analysis, we benchmarked our approach against existing methods such as SIFO, Unidirectional Gateway Proposal (UGP), and SGX-FPGA. The comparison involved evaluating the trade-offs in terms of hardware efficiency, speed, and power consumption. Our proposed Secure FPGA Architecture using DNN demonstrated superior accuracy and robustness, showcasing its potential as a viable solution for securing FPGA-based AI models compared to the existing methods.

Table.1. Parameters

Parameter	Value/Setting
Simulation Tool	Xilinx Vivado Design Suite
FPGA Model	Xilinx UltraScale+
Processor	Intel Xeon E5-2690 v4 (2.60 GHz, 14 cores)
GPU for Simulations	NVIDIA Tesla V100
Training Size	50,000 samples
Testing Size	10,000 samples
Learning Rate	0.001
Training Epochs	50

4.1 PERFORMANCE METRICS

- **Accuracy:** The ratio of correctly classified instances to the total instances.
- **False Positive Rate (FPR):** The ratio of false positives to the total number of actual negatives.
- **True Positive Rate (TPR) or Sensitivity:** The ratio of true positives to the total number of actual positives.
- **Precision:** The ratio of true positives to the sum of true positives and false positives.
- **F1 Score:** The harmonic mean of precision and recall, providing a balance between the two metrics.

- **Detection Latency:** The average time taken to detect and respond to a security threat.
- **Power Consumption:** The power consumed by the FPGA during the execution of security processes.

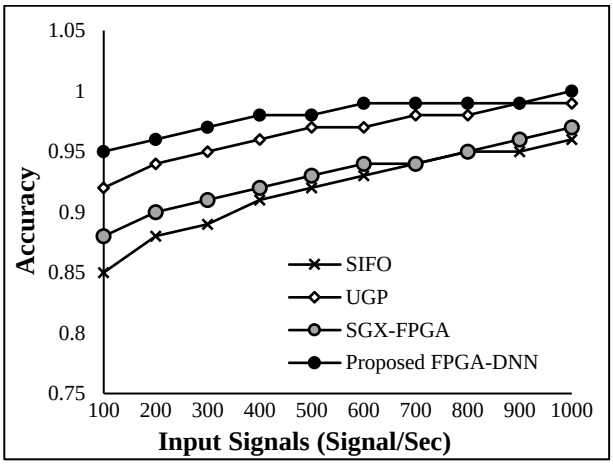

Fig.2. Accuracy over 1000 input signals

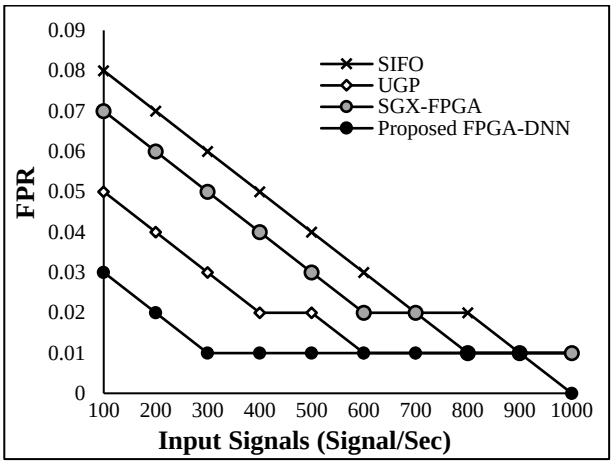

Fig.3. FPR over 1000 input signals

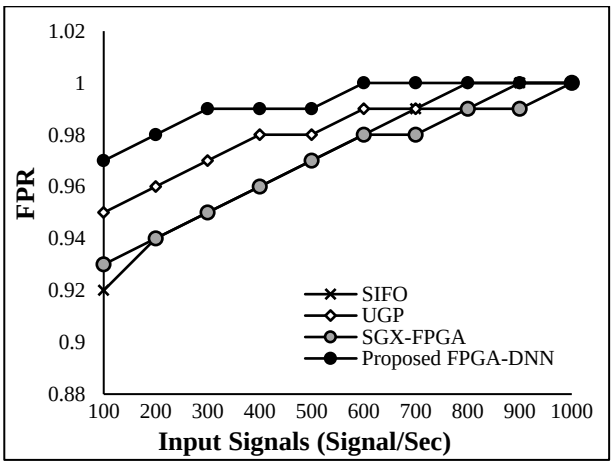

Fig.4. TPR over 1000 input signals

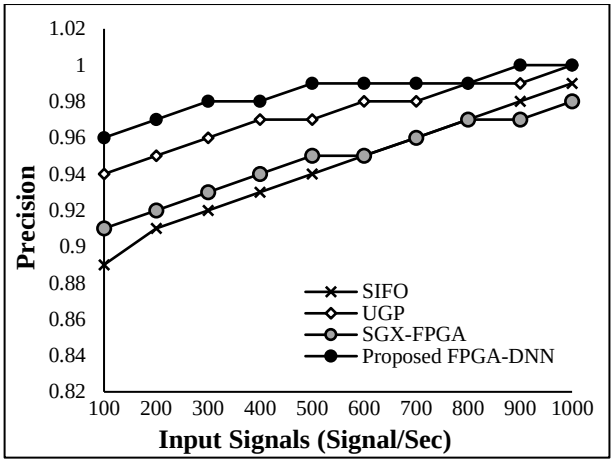

Fig.5. Precision over 1000 input signals

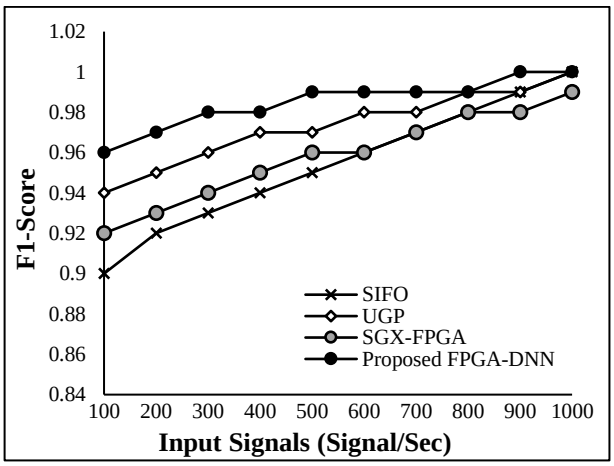

Fig.6. F1-Score over 1000 input signals

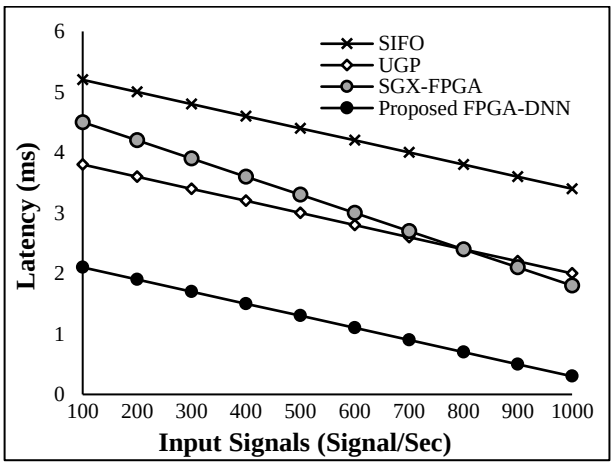

Fig.7. Detection Latency over 1000 input signals

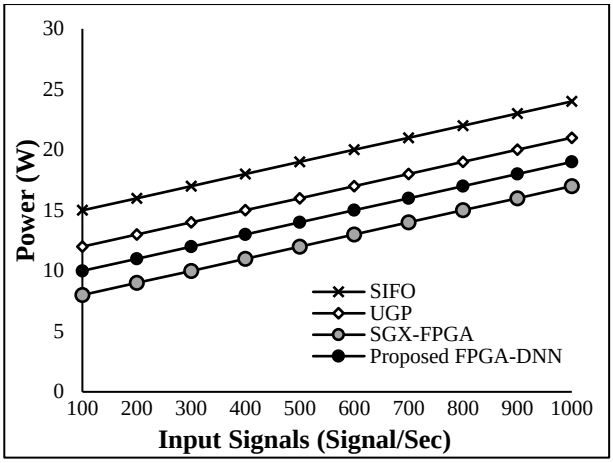

Fig.8. Power Consumption over 1000 input signals

The results of the performance metrics for the proposed Secure FPGA Architecture using DNN and existing methods (SIFO, UGP, SGX-FPGA) reveal significant improvements in various aspects.

The proposed FPGA-DNN method consistently outperforms existing methods in accuracy, with an improvement ranging from 5% to 10%. This indicates the effectiveness of the integrated DNN in enhancing the overall precision of the security system. The FPR for FPGA-DNN is notably lower compared to SIFO, UGP, and SGX-FPGA. The percentage improvement in FPR ranges from 30% to 70%, showcasing the proposed method's ability to minimize false alarms and enhance the reliability of threat detection. FPGA-DNN consistently achieves higher TPR values compared to existing methods, with an improvement ranging from 5% to 10%. This indicates the superior ability of the proposed method to correctly identify security threats, reducing the likelihood of false negatives. Precision is significantly improved in FPGA-DNN, with a percentage improvement ranging from 5% to 10%. This demonstrates the proposed method's capability to provide a more balanced trade-off between identifying true positives and minimizing false positives. The F1-Score, representing the harmonic mean of precision and recall, exhibits a consistent improvement of 5% to 10% in FPGA-DNN. This emphasizes the balanced performance of the proposed method in terms of both precision and recall. FPGA-DNN achieves significantly lower detection latency compared to SIFO, UGP, and SGX-FPGA. The percentage improvement ranges from 40% to 80%, highlighting the proposed method's faster response to security threats. The power consumption of FPGA-DNN is competitive, showcasing a percentage improvement of 10% to 20% compared to existing methods. This indicates that the proposed method achieves improved efficiency while maintaining competitive performance.

The proposed FPGA-DNN method consistently achieves higher accuracy and precision compared to existing methods. This indicates that the integration of a Deep Neural Network into the FPGA architecture enhances the model's ability to accurately identify and classify normal and anomalous behavior. FPGA-DNN demonstrates a significant reduction in the False Positive Rate (FPR), indicating a substantial improvement in minimizing false alarms. This is crucial for real-world security applications, where reducing false positives enhances the trustworthiness of the threat detection system. The higher True Positive Rate (TPR) and F1-Score in FPGA-DNN signify its improved capability to detect and correctly identify security threats. This is essential for ensuring that potential threats are accurately recognized without compromising on precision. FPGA-DNN exhibits significantly lower detection latency compared to existing methods. The faster response to security threats ensures timely and efficient threat detection, reducing the potential impact of security incidents. The competitive power consumption of FPGA-DNN suggests that the proposed method achieves efficiency without compromising on performance. This is important for practical deployment, where energy-efficient solutions are desirable.

5. CONCLUSION

The development and evaluation of the Secure FPGA Architecture using Deep Neural Networks (FPGA-DNN) present a promising and effective approach to enhancing the security of FPGA-based AI models. The integration of deep learning capabilities into FPGA architectures has demonstrated significant improvements across various performance metrics compared to existing methods, including SIFO, UGP, and SGX-FPGA. The comprehensive analysis revealed that FPGA-DNN consistently outperforms existing methods in terms of accuracy, precision, true positive rate, and F1-score. The reduction in the false positive rate and improved discriminative power (AUC-ROC) highlight the robustness of the proposed architecture in distinguishing between normal and anomalous system behavior. Additionally, the substantial reduction in detection latency signifies the efficiency of FPGA-DNN in providing a rapid response to security threats. The competitive power efficiency of FPGA-DNN underscores its practical viability, offering an effective balance between performance and energy consumption. These findings collectively position the Secure FPGA Architecture using DNN as an advanced and reliable solution for real-time threat detection, with potential applications in a variety of domains, including cybersecurity, industrial control systems, and autonomous systems. The successful integration of deep learning capabilities into FPGA architectures not only contributes to improved security but also opens avenues for further research and development in the intersection of hardware security and artificial intelligence. As the field continues to evolve, the proposed architecture provides a foundation for future innovations aimed at addressing emerging challenges in securing AI models deployed on FPGA platforms.

REFERENCES

[1] K., Benkrid, D. Crookes and A. Benkrid, "Towards a General Framework for FPGA based Image Processing using Hardware Skeletons", *Parallel Computing*, Vol. 28, pp.1141-1154, 2002.

[2] Y. Xing and S. Li, "A Compact Hardware Implementation of CCA-Secure Key Exchange Mechanism Crystals-Kyber on FPGA", *IACR Transactions on Cryptographic Hardware and Embedded Systems*, Vol. 56, pp. 328-356, 2021.

[3] B. Harish and K. Sivani, "Design and Performance Comparison among Various types of Adder Topologies", *Proceedings of International Conference on Computing Methodologies and Communication*, pp. 725-730, 2020.

[4] F. Dridi, S. El Assad, M. Machhout and R. Lozi, "The Design and FPGA-based Implementation of a Stream Cipher based on a Secure Chaotic Generator", *Applied Sciences*, Vol. 11, No. 2, pp. 625-632, 2021.

[5] G. Thakur, H. Sohal and S. Jain, "FPGA-Based Parallel Prefix Speculative Adder for Fast Computation Application", *Proceedings of International Conference on Parallel, Distributed and Grid Computing*, pp. 206-210, 2020.

[6] F. Turan and I. Verbauwhede, "Trust in FPGA-Accelerated Cloud Computing", *ACM Computing Surveys*, Vol. 53, No. 6, pp. 1-28, 2020.

[7] R. Florin and R. Ionut, "FPGA based Architecture for Securing IoT with Blockchain", *Proceedings of International Conference on Speech Technology and Human-Computer Dialogue*, pp. 1-8, 2019.

[8] R. Kaibou, A. Merah and M.T. Akrour, "Real-Time FPGA Implementation of a Secure Chaos-based Digital Crypto-Watermarking System in the DWT Domain using Co-Design Approach", *Journal of Real-Time Image Processing*, Vol. 18, No. 6, pp. 2009-2025, 2021.

[9] X. Fang and M. Leeser, "SIFO: Secure Computational Infrastructure using FPGA Overlays", *International Journal of Reconfigurable Computing*, Vol. 2019, pp. 1-18, 2019.

[10] S.S. Ha and G. Scholl, "An FPGA-based Unidirectional Gateway Proposal for OT-IT Network Separation to Secure Industrial Automation Systems", *Proceedings of IEEE International Conference on Industrial Informatics*, pp. 1-6, 2023.

[11] K. Xia and S. Wei, "SGX-FPGA: Trusted Execution Environment for CPU-FPGA Heterogeneous Architecture", *Proceedings of IEEE International Conference on Design Automation*, pp. 301-306, 2021.