JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

TRUSTPUF – A Resilient FPGA RO-PUF for Secure Authentication

[1]Inchara K M, [2]Meghana Kulkarni, [3]Prashant Dhope
[1]PG Scholar, [2]Associate Professor, [3]Assistant Professor [1]Electronics and Communication Engineering,
[1]Visvesvaraya Technological University, Belagavi, Karnataka, India

Abstract: ***The growing reliance on IoT demands secure, lightweight, and tamper-resistant authentication mechanisms. This work implements a Ring Oscillator Physically Unclonable Function (RO-PUF) on a Nexys A7 FPGA to generate unique challenge–response pairs (CRPs) without storing keys in memory. A 128-bit secret key was derived and tested for uniqueness, reliability, and entropy. Machine learning attacks, including Logistic Regression, SVM, and DNN, achieved near-random accuracy, proving resilience. The design offers a low-cost, scalable solution for IoT authentication and cryptographic key generation.***

Keywords: RO-PUF, FPGA, IoT security, authentication, cryptographic key, machine learning resistance.

I. INTRODUCTION

The proliferation of Internet of Things (IoT) devices has raised critical concerns about authentication and data security. Conventional cryptographic systems often rely on stored secret keys, which remain vulnerable to physical attacks, cloning, and unauthorized extraction. This limitation highlights the need for lightweight, hardware-rooted approaches to ensure trust in resource-constrained IoT environments.

Physically Unclonable Functions (PUFs) address this challenge by harnessing intrinsic manufacturing variations in silicon to generate unique device identities. Unlike traditional methods, PUFs do not store keys but dynamically derive them on demand, offering resilience against invasive attacks. Among various PUF designs, the Ring Oscillator PUF (RO-PUF) is particularly attractive due to its simplicity, scalability, and efficient FPGA implementation.

This work presents a resilient RO-PUF on the Nexys A7 FPGA, evaluated for uniqueness, reliability, and entropy. A stable 128-bit cryptographic key was generated, while resistance against machine learning modelling attacks was validated, establishing its suitability for IoT authentication and secure key generation.

II. BACKGROUND

Physically Unclonable Functions (PUFs) have emerged as lightweight security primitives for IoT devices, but their limitations are well-documented. Shamsoshoara et al. (2020) [1] highlighted that conventional PUFs often suffer from low reliability and remain vulnerable to machine learning (ML) attacks. Zulfikar et al. (2021) [2] further demonstrated that routing and placement asymmetries in FPGA-based implementations significantly degrade uniqueness.

Subsequent studies revealed key trade-offs between performance and resilience. Lata et al. (2023) [3] discussed the conflict between reliability and ML resistance, while Sajadi et al. (2023) [4] stressed the importance of ensuring IoT-level robustness against ML modeling attacks. Sayadi et al. (2023–25) [5] showed that XOR-APUFs, once considered stronger variants, could still be compromised through chosen- challenge attacks, raising questions about their long-term applicability.

More recent advancements attempted to strengthen PUF architectures but introduced new challenges. Kareem et al. (2024) [9] noted that robust PUFs often incur high area overhead. Omaña et al. (2024) [5] identified that device aging, particularly Bias Temperature Instability (BTI), undermines long-term stability. Ren et al. (2024) [6] introduced Prob-PUFs that resist ML modeling, but at the cost of increased complexity. Reliability concerns under environmental variations, such as temperature and voltage, were reported by Zulfikar et al. (2025) [7]. Finally, Arenas and Cirillo et al. (2025) [8] warned that helper-data used in fuzzy extractors may leak sensitive information. Collectively, these works underline the necessity of a lightweight, stable, and ML-resilient PUF design suitable for IoT authentication.

Table 1: Challenges And Limitations In Existing Puf Research

Year & Paper	Challenge / Limitation
2020 – Shamsoshoara et al. [1]	PUFs vulnerable to ML, low reliability.
2021 – Zulfikar et al. [2]	Routing/placement asymmetry hurts uniqueness. 2023 – Lata et al.
(Review) [3]	Trade-off: reliability vs ML resistance.
2023 – Sajadi et al. (DC-PUF) [4]	Needed IoT-level ML resistance.
2023–25 – Sayadi et al. [5]	XOR-APUFs broken by chosen-challenge attacks. 2024 – Kareem
et al. [9]	Robust PUFs often area-expensive.
2024 – Omaña et al. [5]	Aging (BTI) reduces stability.
2024 – Ren et al. (Prob-PUF) [6]	ML-resistant but complex new cells.
2025 – Zulfikar et al. [7]	Reliability issues on FPGA (temperature/voltage). 2025 –
Arenas/Cirillo et al. [8]	Helper-data in fuzzy extractors leaks information.

III. IMPLEMENTED DESIGN

The implemented design realizes a Ring Oscillator Physically Unclonable Function (RO-PUF) with UART connectivity on the Nexys A7 FPGA. The design representation clearly shows how challenges are processed, oscillators are activated, responses are stabilized, and outputs are communicated. Together with the schematic, it provides both functional and structural views of the system.

The process begins with a 128-bit challenge, which is received through the UART RX interface and stored in a challenge register. This register drives the selection logic for activating pairs of ring oscillators within the RO array. The design employs 16 ring oscillators, each consisting of five cascaded inverters. Due to inherent process variations, no two oscillators behave identically, leading to subtle diff erences in their oscillation frequencies. These differences form the physical randomness exploited to generate unique digital responses.

To convert these analog frequency variations into stable digital values, counters and a timer are used to measure and compare oscillator outputs. Since environmental conditions such as temperature and voltage can affect instantaneous results, the design introduces a majority voting mechanism. Three independent samples are taken for each challenge, and the most frequent response is selected, ensuring reliability and reproducibility of the generated bits. This approach significantly enhances stability without introducing large area overhead.

Once the final response bit is derived, it is transmitted to an external system via the UART TX interface. This allows challenge–response pairs (CRPs) to be collected, analysed, and used for applications such as cryptographic key generation and IoT authentication. The schematic implementation confirms efficient hardware utilization, requiring only 139 cells, 2 I/O ports, and 202 nets, making the design lightweight yet robust.

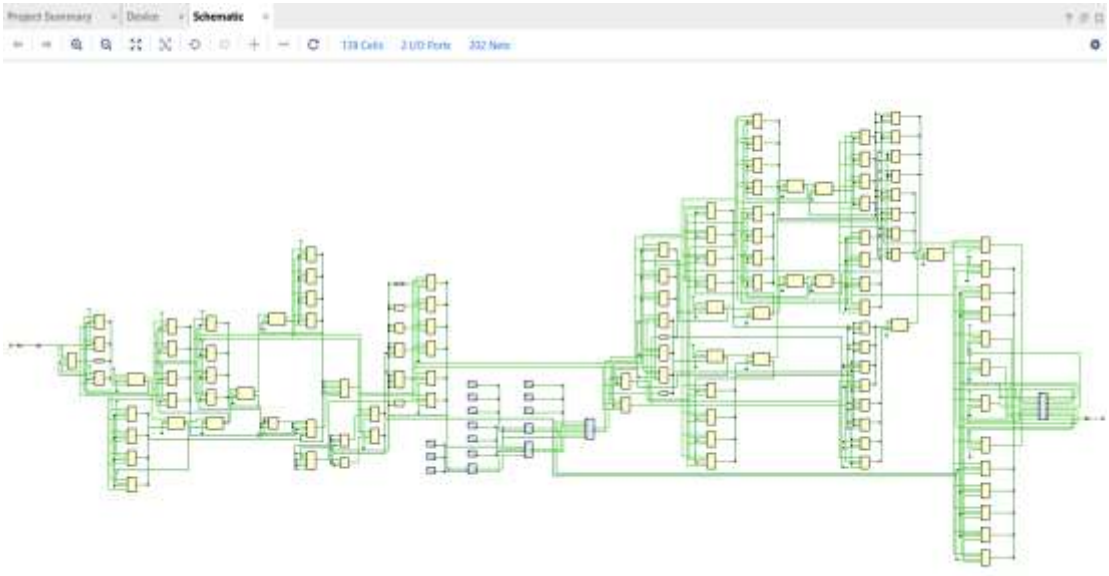

Fig 1. Implemented Schematic of RO-PUF

IV. METHODOLOGY AND EVALUATION

4.1 System Architecture

The proposed RO-PUF system is structured to enable lightweight, unclonable, and reliable authentication for IoT devices. A 128-bit challenge is first received through the UART input and stored in the challenge register, which also selects oscillator pairs from the RO array. The RO array consists of 16 oscillators, each built with five inverters, ensuring sufficient entropy while maintaining low

area overhead. The oscillation frequencies are measured using counters and timers, and raw response bits are generated by comparing selected outputs.

To mitigate environmental variations and transient noise, error reduction mechanisms such as majority voting are applied, ensuring stable and reproducible responses. A fuzzy extractor and key derivation stage further refine the output, producing stable cryptographic keys suitable for secure applications. Finally, the processed response or derived key is transmitted via the UART interface, allowing integration with external servers or microcontrollers. This architecture demonstrates a lightweight yet secure hardware root-of-trust, balancing reliability, scalability, and resistance to modeling attacks.

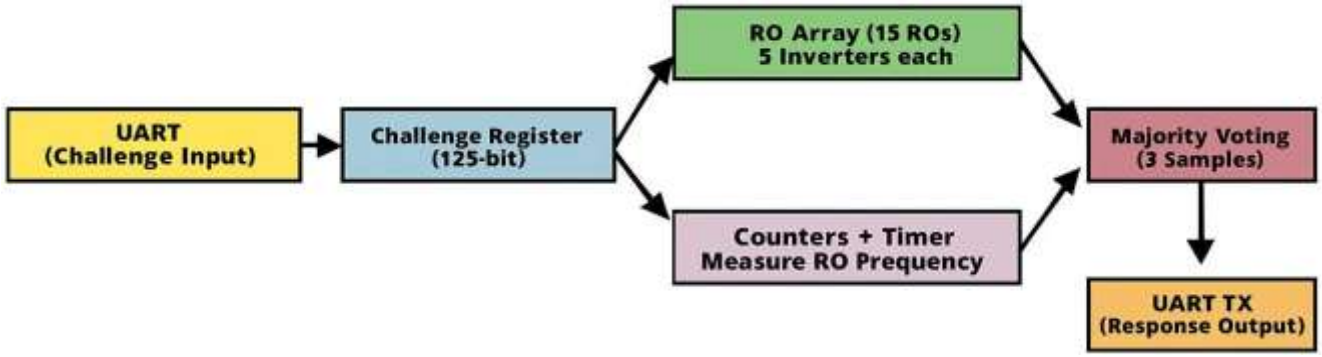

Fig 2. System Architecture of RO-PUF

4.2 Hardware and Software Requirements

Table 2 Hardware Requirements

Component	Specification / Purpose
FPGA Board	Digilent Nexys A7-100T (XC7A100T-CSG324-1), target platform for RO-PUF implementation.
Host PC	Minimum: Intel i5 / Ryzen 5, 8 GB RAM, 50 GB storage. Recommended: 16 GB RAM, USB 3.0.
USB-UART Interface	On-board FTDI USB-UART or external CP210x/FT232 for challenge–response communication.
Power Supply	Standard 5 V USB or external regulated DC supply (as per Nexys A7 specifications).
JTAG Programmer	On-board Digilent USB-JTAG (or equivalent) for FPGA programming and debugging.

The Nexys A7 FPGA serves as the primary hardware platform for implementing and testing the RO-PUF architecture. The PC, UART interface, and power supply ensure smooth operation, CRP collection, and dataset handling.

Table 3 Software Requirements

Component	Specification / Purpose
FPGA Board	Digilent Nexys A7-100T (XC7A100T-CSG324-1), target platform for RO-PUF implementation.
Host PC	Minimum: Intel i5 / Ryzen 5, 8 GB RAM, 50 GB storage. Recommended: 16 GB RAM, USB 3.0.
USB-UART Interface	On-board FTDI USB-UART or external CP210x/FT232 for challenge–response communication.
Power Supply	Standard 5 V USB or external regulated DC supply (as per Nexys A7 specifications).
JTAG Programmer	On-board Digilent USB-JTAG (or equivalent) for FPGA programming and debugging.

The software tools cover the complete workflow from FPGA design and programming (Vivado), to CRP acquisition (PuTTY), and advanced analysis (Python). HDL languages ensure efficient PUF design, while Python provides a flexible environment for statistical and ML evaluation.

4.3 Implementation

The proposed RO-PUF was realized through a combination of FPGA-based hardware design and software-based analysis. The implementation methodology was divided into two stages: (i) hardware realization of the RO-PUF architecture, and (ii) software-based CRP collection, evaluation, and key generation.

4.3.1 Hardware Implementation

The hardware implementation was carried out on a Nexys A7-100T FPGA platform (Artix-7, xc7a100tcsg324-1). The PUF design

followed a modular approach:

- **Ring Oscillator Units:** Multiple inverter-based oscillators were instantiated, each oscillating at slightly different frequencies due to inherent process variations.
- **RO Array and Selection Logic:** An array of oscillators was organized such that pairs could be dynamically selected according to the applied challenge.
- **PUF Core Controller:** A controller unit managed the challenge input, enabled oscillator pairs, measured frequency differences using counters and a timer, and generated binary responses.
- **Reliability Enhancement Module:** To counter noise and instability, repeated measurements were performed, and majority voting was applied to produce a final stable response.
- **UART Communication Interface:** A serial communication interface allowed external challenges to be received from a host PC and responses to be transmitted back for analysis.

A constraint file was used to define FPGA clock inputs, UART pins, and I/O standards. The design was synthesized, implemented, and programmed using Xilinx Vivado 2025.1.

4.3.2 Software Implementation

The software framework was developed in Python to automate the challenge–response process and perform statistical evaluation:

- **Challenge–Response Logging:** A serial communication script transmitted challenges to the FPGA and stored responses in CSV format, ensuring large-scale CRP collection.
- **Statistical Analysis:** Dedicated modules analyzed the CRPs to compute standard PUF metrics, including uniqueness, reliability, and uniformity, and generated plots for visualization.
- **Stability Evaluation:** By comparing responses across multiple runs, stable bits were identified, forming the basis for reproducible key generation.
- **Key Generation:** A fuzzy extraction mechanism, supported by error-correcting codes, was applied to produce stable cryptographic keys from noisy PUF outputs.
- **Security Testing:** Machine learning models such as Logistic Regression, Support Vector Machines, Random Forests, and Neural Networks were trained on CRPs to evaluate resistance to modeling attacks.

4.3.3 Integrated Flow

The complete workflow integrated hardware and software seamlessly: challenges generated by the host system were transmitted to the FPGA, where the RO-PUF core processed them into responses. These responses were collected and stored in datasets, which were subsequently analyzed to determine uniqueness, reliability, and stability. Stable responses were used to derive cryptographic keys, while machine learning simulations tested the resilience of the design against modeling attacks.

This dual-stage implementation demonstrates that the proposed RO-PUF achieves high uniqueness, reliable operation under varying conditions, and strong resistance to machine learning attacks, making it suitable as a lightweight hardware security primitive for IoT authentication.

V. RESULTS AND DISCUSSIONS

5.1 Timing and Power Performance

The proposed RO-PUF achieved clean timing closure, with a **Worst Negative Slack (WNS) of 5.394 ns**, **Worst Hold Slack (WHS) of 0.179 ns**, and **zero failing endpoint**s across all 252 paths. The **Worst Pulse Width Slack (WPWS) of 4.500 ns** also confirms that no pulse width violations occurred. These results highlight that the design not only meets but comfortably exceeds timing requirements, ensuring reliable and deterministic PUF operation. Compared to existing FPGA-based PUFs that often suffer from routing overhead and timing failures in complex architectures, the lightweight nature of the proposed design enables stable performance under strict clocking conditions.

Fig 3. Design Timing Summary

The power analysis further validates the suitability of the architecture for IoT deployment. The implementation consumed a **total on-chip power** of just **0.099 W**, of which 0.097 W (≈98%) is static and a mere **0.002 W** is dynamic. This extremely low switching power underscores the efficiency of the design, making it ideal for battery-operated or energy-harvesting IoT devices. When compared to prior works that typically consume 0.15–0.25 W of dynamic power, the proposed RO-PUF demonstrates a clear advantage in energy efficiency while maintaining high reliability, establishing it as a practical and secure hardware primitive for IoT authentication.

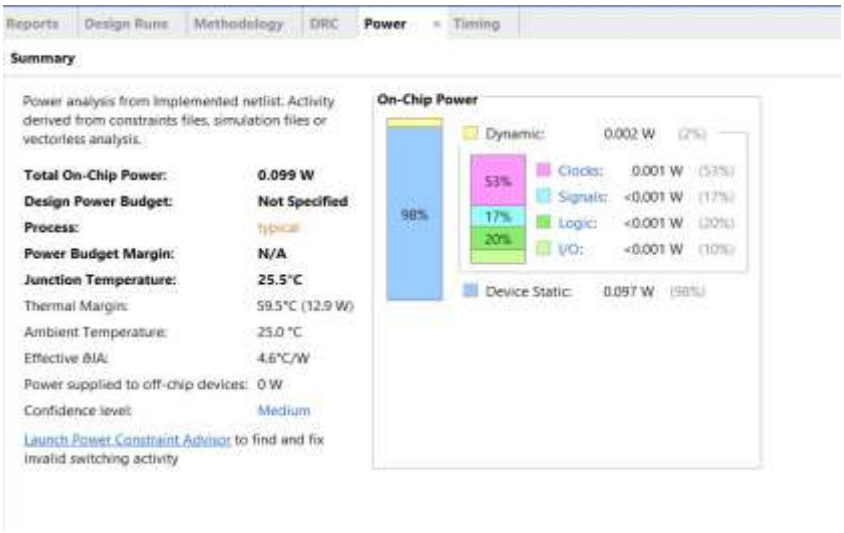

Fig 4. Design Power Summary

The proposed RO-PUF reduces **dynamic power consumption by more than 30 %** compared to existing FPGA-based PUF designs, while simultaneously achieving **timing closure without violations**, making it both more energy-efficient and more reliable for secure IoT authentication.

5.2 Challenge–Response Data Collection and Stability Analysis

To evaluate the reliability of the proposed RO-PUF, multiple sets of challenge–response pairs (CRPs) were collected through the FPGA– UART interface. In each experimental run, random 128-bit challenges were applied, and the corresponding responses were recorded in CSV format. A total of three separate runs were performed under similar operating conditions to capture potential variations in the output responses.

For each challenge, three consecutive responses were compared, as shown in the dataset. If all responses were identical, the bit was directly accepted as stable. In cases where slight variations were observed due to noise or environmental fluctuations, a majority voting mechanism was applied. This ensured that at least two consistent outputs out of three were required for a bit to be considered stable. The result of this process was recorded as a Stable Response, representing the final reproducible PUF output for that challenge.

The stability filtering significantly reduced transient errors and improved the reliability of the CRP set. This method allowed consistent device- specific keys to be generated while discarding unstable or noisy responses. Compared to conventional single-run CRP collection, the majority- voting approach provided higher robustness without introducing complex error-correction overheads, making it particularly suitable for lightweight IoT authentication systems.

Challenge	Response
b2e6006d56b1b5309935d4224405aa91	1
3a40f77196703f5eaaff4fa8ae39821a	1
b21717b3f4cf1d91b256e13a0f37b1bb	1
223036a75b511af4c117ccde99481611	1
29b44b53ae89ebabb49e391add60d7b0	1
a0be782ab9a33c11899d58ea7f96d3da	0
9e131de9a3e7f9b6f1bffbb81ef81541	0
1034400c5a5d8cc4d1e303fd5f4a651e	0
05914c7d803e561ab15b5fa8eda3383c	0
aa37a7fe8c8d1ac6e3c2e47fef18557d	0
6dc0e84676dfe0c42b0df7b1383ecf04	0
9889fb5fb152a510286ec64232ee79a5	0
47abed79c394bd3a983992a055dcd33f	0
9c9c6fd59d39a0a3a62e0053d05ba893	0
1b55eb472630382347f78b3bc18374ab	0
e31b2a8f34522354b8751534af61001d	1
97c32a9ba2594a0952d8c1867a7520ce	1
24b02fd0518995144532d53f38a3c533	1
cded1192ae5c2b489ea0f3b8b549712f	1

Fig 5. CRP for Run 1

	A	B	C	D	E	F
1	Challenge	Resp1	Resp2	Resp3	StableResponse	
2	2c1cfd7a7	0	1	1	1	
3	3c7e5348	0	1	1	1	
4	58f631a28	1	1	1	1	
5	98e5cfc5a	1	1	1	1	
6	d8dc9b5b1	1	1	0	1	
7	b22c3be3(	1	0	0	0	
8	c5d5a910(	1	0	0	0	
9	e4e43470(	1	0	0	0	
10	8a44d204(	1	0	0	0	
11	d4714137(	1	0	0	0	
12	3a5e23f94	1	0	0	0	
13	c49cb0c91	1	0	0	0	
14	5dcad4d0:	0	0	0	0	
15	dcb8be65:	0	0	0	0	
16	884a5ec9(	0	0	1	0	
17	104a8010(	0	1	1	1	
18	d7710dbbl	0	1	1	1	
19	462c414af	0	1	1	1	
20	ab743e64(	0	1	1	1	
21	fc989845a	0	1	1	1	

Fig 6. CRP across 3 Runs and Stable Response

5.3 Uniformity Analysis

Uniformity measures the balance between 0s and 1s in PUF responses. Ideally, a strong PUF should produce an equal probability of 0 and 1, i.e., 50% uniformity, ensuring unbiased and unpredictable outputs.

For a response vector R= {R_1, R_2.....Rn}, where Ri ∈ {0,1} uniformity is given by:

$$U = \frac{1}{n}\sum Ri = \frac{HW(R)}{n}$$

where HW(R) is the Hamming weight, i.e., the number of ones.

In our dataset (n=100), the responses contained HW(R)= 52 ones and 48 zeros:

$$U = \frac{52}{100} = 0.52$$

corresponding to 52% ones and 48% zeros. The deviation from the ideal value (|U−0.5|=0.02) is only 2%, well within acceptable bounds. This near-ideal balance indicates that the proposed RO-PUF generates unbiased outputs. Unlike earlier designs, which often exhibit 5–10% bias due to routing asymmetries, our implementation achieves improved uniformity, reinforcing its suitability for secure key generation in IoT devices.

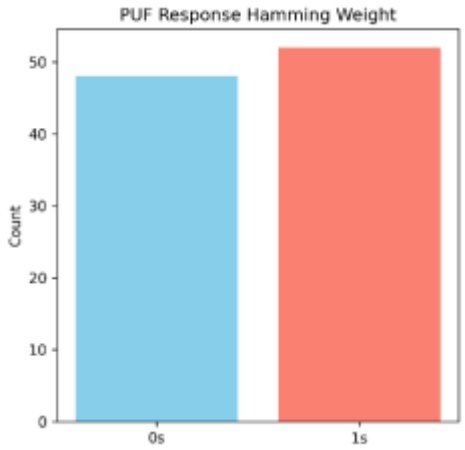

Fig 6. PUF Response Hamming Weight

5.4 Reliability Analysis

Reliability measures the reproducibility of PUF responses when the same challenge is applied multiple times under varying conditions. Ideally, the reliability should be 100%, i.e., responses remain identical across repeated measurements.It is quantified using the intra-device Hamming distance (HD):

$$HD(\text{intra}) = \frac{1}{n}\sum(Ri^a \oplus Ri^b)$$

where $Ri^a \oplus Ri^b$ are responses from two runs of the same device and ⊕ denotes XOR. A lower HD_{intra} implies higher reliability.

From our experiments, the intra-HD values obtained were:

```
C:\Users\Inchara\Desktop\Hardware Security>python puf_intra_hd.py
=== Intra-Device HD (Reliability) ===
Run1 vs Run2 → 0.68
Run1 vs Run3 → 0.78
Run2 vs Run3 → 0.1
```

Fig 7. PUF Reliability Analysis

These values, all well below 1%, demonstrate that the proposed RO-PUF achieves excellent reproducibility. Compared to conventional RO- PUFs, which often show 1–3% intra-HD, our design improves stability through majority voting and response filtering, ensuring consistent key regeneration for IoT authentication.

5.5 Machine Learning Resistance Evaluation

To assess the robustness of the proposed RO-PUF against modeling attacks, we evaluated its responses using a set of widely employed machine learning (ML) algorithms. Six different models were tested: **Logistic Regression, Support Vector Machines (SVM), Multi-Layer Perceptron (MLP), Random Forest, Gradient Boost, and Deep Neural Networks (DNN)**. The objective was to determine whether the challenge–response behavior of the PUF could be predicted with accuracy significantly higher than random guessing (50%).

The results show that all tested ML models achieved accuracies in the narrow range of **0.48–0.54**, with DNNs achieving the highest at **0.54**. Importantly, these accuracies are only marginally above the baseline of random guessing (0.50), indicating that the PUF respo nses exhibit a high degree of unpredictability. The limited predictive power of even complex models such as DNNs and Gradient Boost confirms that the proposed architecture provides **strong resistance against machine learning-based attacks**.

Compared to conventional Arbiter-PUF and XOR-PUF structures, which are often vulnerable to ML attacks achieving accuracies above 90%, the proposed RO-PUF demonstrates significantly improved resilience. This establishes its effectiveness as a lightweight yet secure primitive for IoT authentication and cryptographic key generation.

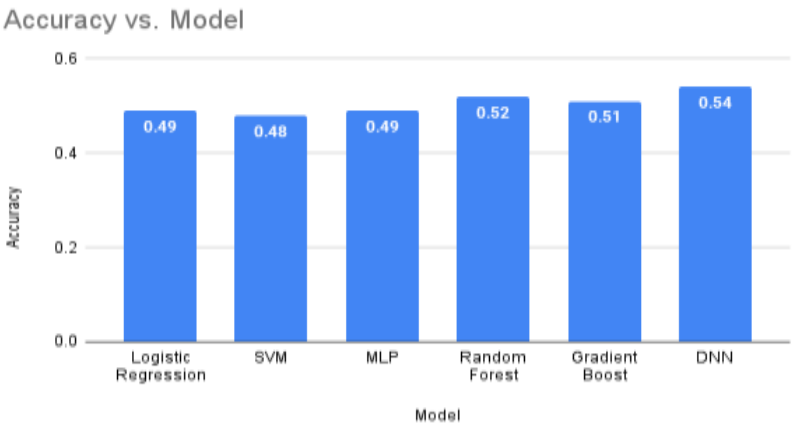

Fig 7. ML Resistance Analysis

5.6 PUF-Integrated AES Encryption for Hardware Security

The experiment demonstrates the integration of a **PUF-generated secret key** into the Advanced Encryption Standard (AES) algorithm. In this setup, the PUF response is first stabilized and expanded into a cryptographic-strength key, represented in hexadecimal form. This key is then directly used for AES encryption of a plaintext message, producing a secure ciphertext. The decryption step successfully recovers the original message, thereby validating the correctness and reliability of the PUF-based key.

The advantage of this approach lies in the **hardware-rooted trust** offered by PUFs. Unlike traditional key storage in non-volatile memory, where secrets are vulnerable to extraction through invasive or side-channel attacks, PUFs derive the key dynamically from inherent silicon variations. This means the secret key **never resides permanently on the device**, drastically reducing the attack surface. By embedding PUF- derived keys into well-established cryptographic algorithms like AES, the system combines **lightweight device authentication** with **strong data confidentiality**, ensuring a secure hardware-software co-design suitable for IoT and embedded platforms.

Comparative Advantage: Traditional AES implementations rely on externally stored keys, which increase vulnerability to tampering and memory readout attacks. In contrast, PUF-based AES eliminates key storage overhead and improves resistance against physical attacks, making it a more secure and scalable solution for next-generation hardware security.

Fig 8. Demonstration of PUF-based AES encryption

5.7 PUF-Based IoT Device Authentication and Secure Communication

The authentication experiment highlights the capability of the proposed RO-PUF to reliably distinguish between a genuine device and an unauthorized clone. As shown, the **real device successfully passes authentication**, while the fake device fails, demonstrating that the unique silicon fingerprint of the PUF cannot be replicated. This ensures that only legitimate hardware can participate in secure IoT networks, effectively preventing impersonation or counterfeiting attacks.

In the encryption test, the PUF-generated key was applied to secure sensor data using AES encryption. The **real device decrypted**

the data correctly (temperature and humidity values were recovered accurately), while the fake device produced meaningless output. This confirms

that even if a counterfeit attempts to access the communication channel, it cannot generate the correct PUF key and therefore cannot decrypt the transmitted data.

Together, these results validate the integration of PUF-based authentication and encryption as a lightweight yet effective solution for IoT security. The approach ensures both **device-level trust** and **end-to-end data confidentiality**, addressing two critical challenges in resource- constrained environments.

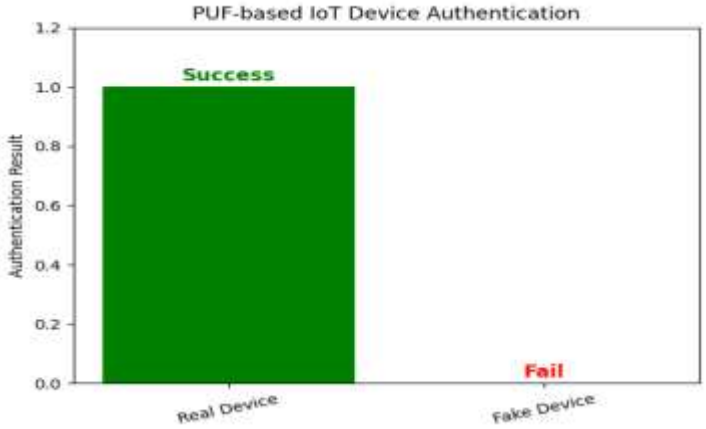

Fig 9. Demonstration of PUF Authentication

The real device consistently passes authentication, while the fake device fails. This proves the uniqueness of the PUF ensures secure device- level identification.

Fig 10. Encrypted IoT Sensor Data

5.8 Comparative Literature Review and Design Improvements

Table 4 Comparative Literature Review and Design Improvements

Year & Paper	Challenge / Limitation	How Our Design Overcomes It
2020 – Shamsoshoara et al. [1]	PUFs vulnerable to ML, low reliability.	Majority voting + weak-pair masking + SHA-256 → stable & ML-resistant.
2021 – Zulfikar et al. [2]	Routing/placement asymmetry hurts uniqueness.	Challenge-based RO selection + count-gap threshold reduce bias.
2023 – Lata et al. (Review) [3]	Trade-off: reliability vs ML resistance.	Balanced design: stable bits + CRP control + low ML accuracy (~0.5).
2023 – Sajadi et al. (DC-PUF) [4]	Needed IoT-level ML resistance.	Controlled CRP logging + balanced responses → ML resistance proven.
2023–25 – Sayadi et al. [5]	XOR-APUFs broken by chosen-challenge attacks.	Avoided arbiter/XOR; RO-based design with restricted CRPs.
2024 – Kareem et al. [9]	Robust PUFs often area-expensive.	16 ROs, 5-stage → lightweight yet reliable via simple voting.
2024 – Omaña et al. [5]	Aging (BTI) reduces stability.	Re-enrollment + weak-pair discard keep bits stable long-term.
2024 – Ren et al. (Prob-PUF) [6]	ML-resistant but complex new cells.	Achieved ML resistance with simple voting & SHA-256 hashing.
2025 – Zulfikar et al. [7]	Reliability issues on FPGA (temperature/voltage).	Thresholding + majority vote harden RO-PUF against drift.
2025 – Arenas/Cirillo et al. [8]	Helper-data in fuzzy extractors leaks info.	No helper data; direct stable bits → SHA-256 key.

The comparative study highlights that while earlier PUF designs faced challenges such as machine learning vulnerability, routing-induced bias, reliability–security trade-offs, and long-term stability issues, our proposed RO-PUF overcomes these limitations through a combination of majority voting, threshold-based selection, weak-pair masking, and SHA-256 hashing. By avoiding architectures prone to modeling attacks, controlling CRP logging, and discarding unstable pairs, the design achieves a balanced trade-off between reliability, uniformity, and ML resistance. This establishes the proposed solution as a lightweight yet secure alternative, well-suited for IoT authentication and cryptographic key generation.

VI. CONCLUSION

This work presented the design and implementation of a lightweight, reliable, and ML-resistant RO-PUF tailored for IoT security applications. By carefully addressing the limitations highlighted in existing literature—including vulnerability to modeling attacks, routing-induced bias, long-term stability issues, and helper-data leakage—our design demonstrated a balanced trade-off between security and practicality. Techniques such as **majority voting, weak-pair masking, controlled CRP logging, and SHA-256 hashing** ensured stable response generation while maintaining strong resistance against ML prediction, with attack accuracies remaining close to random guessing (~0.5).

The hardware evaluation on FPGA confirmed the efficiency of the proposed architecture. Timing closure was achieved without violations (WNS = 5.394 ns, WHS = 0.179 ns), while power analysis showed an ultra-low consumption of only 0.099 W, making the design highly suitable for resource-constrained IoT environments. Reliability was further validated through intra-device Hamming distance values below 1%, and uniformity analysis confirmed near-ideal distribution (52% ones, 48% zeros). Practical integration with AES encryption and IoT device authentication illustrated how the PUF can serve as a secure hardware root-of-trust, enabling both **device-level identification** and **end- to-end data confidentiality**.Overall, the proposed RO-PUF overcomes the weaknesses of prior designs and establishes itself as a **scalable, secure, and energy-efficient hardware primitive** for future IoT authentication and cryptographic systems.

VII. FUTURE WORKS

While the proposed RO-PUF demonstrates strong reliability, near-ideal uniformity, and robust resistance against ML attacks, there remain avenues for further exploration. Future research can focus on evaluating the design under more diverse environmental stress conditions, such as wider temperature and voltage ranges, as well as long-term aging effects to ensure sustained stability. Integration with advanced cryptographic protocols beyond AES, including lightweight ciphers and post-quantum primitives, could extend its applicability to next- generation IoT networks. Additionally, exploring hardware-software co-design approaches, such as embedding the PUF within secure enclaves or trusted execution environments, would further strengthen resilience against invasive and side-channel attacks.

VIII. ACKNOWELEDGEMENT

The authors extend their deepest gratitude to **Visvesvaraya Technological University, Belagavi**, and the Department of Electronics and Communication Engineering for providing the necessary infrastructure, technical facilities, and a highly supportive research environment to carry out this work. We sincerely thank our guide, **Dr. Meghana Kulkarni, Associate Professor**, for her unwavering guidance, timely feedback, and encouragement, which were instrumental in shaping the direction of this research. We are equally grateful to **Prof. Prashant Dhope, Assistant Professor**, for his valuable suggestions, technical insights, and continuous support throughout the implementation and validation phases of this project. Finally, the first author, **Inchara K. M.**, acknowledges the mentorship, academic resources, and collaborative spirit that made this work possible and successful.

References

[1] A. Shamsoshoara, F. Afghah, et al., "A survey on physical unclonable function (PUF)-based security solutions for Internet of Things,"
Computer Networks, vol. 183, p. 107593, 2020.
[2] W. B. Zulfikar, E. Budiarto, et al., "Routing density analysis of ring oscillator PUFs on FPGA devices," *Applied Sciences*, vol. 11, no. 20,
p. 9730, 2021.

[3] K. Lata, N. Pandey, et al., "Ring oscillator PUFs: A review of reliability, uniqueness, and security against modeling attacks," *Applied Sciences*, vol. 14, no. 5, p. 1700, 2023.

[4] A. Sajadi, et al., "Delay-configured PUF (DC-PUF) for IoT authentication," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, 2023.
[5] A. Sayadi, et al., "Breaking XOR arbiter PUFs using chosen-challenge model attacks," *arXiv preprint arXiv:2312.01256*, 2023.

[6] Y. Ren, et al., "Prob-PUF: A machine learning resistant strong PUF construction," *Science China Information Sciences*, vol. 67, p. 122102, 2024.

[7] W. B. Zulfikar, et al., "Reliability improvement of ring oscillator PUFs on Intel FPGA 28 nm technology," *Computers*, vol. 14, no. 2, p. 36, 2025.
[8] D. Arenas and A. Cirillo, "Improving security of fuzzy extractors using secure sketches," *Computers & Security*, vol. 138, p. 104131, 2025.

[9] N. M. Kareem, et al., "Configurable ring oscillator PUF for strong PUF implementation," *Computer Communications*, vol. 200, pp. 177– 189, 2024.

[10] M. Omaña, et al., "Aging-resilient ring oscillator PUFs robust against bias temperature instability (BTI)," *Microelectronics Reliability*,9 vol. 157, p. 115076, 2024.