DOI: http://dx.doi.org/10.26483/ijarcs.v16i6.7386

Volume 16, No. 6, November-December 2025

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

A SURVEY OF HARDWARE TROJAN DETECTION APPROACHES IN FPGA AND ASIC SYSTEMS

Himanshu Barhaiya
Department of Computer Science and Engineering
Lakshmi Narain College of Technology
Bhopal

***Abstract*—**Hardware security has become highly threatened by the high rate of semiconductor design and production globalization. The hardware Trojans have become a major danger that can negatively affect confidentiality, integrity, and availability of electronic systems due to the large number of ICs that are currently being produced in insecure facilities. These destructive changes can be introduced in any of the design stages or manufacturing processes and usually they spend many years being in dormancy before activation. This paper provides a wide overview of hardware Trojan detection methods in FPGA and ASIC systems. The paper discusses different detection methods, such as side-channel testing, optimization-based testing, machine learning-assisted testing and electromagnetic (EM)-based testing. It also addresses the weaknesses specific to FPGA and ASIC design and architecture, recent research trends and challenges, including the process variation, scalability and the complexity of the design of modern SoCs. The survey gives a complete account of the methodologies of detection, and the need to have secure design flows, trusted manufacture, and superior AI-based solutions defines the future hardware security solutions.**

***Keywords*—**Hardware Trojans, FPGA Security, ASIC Security, Machine Learning, Hardware Vulnerabilities, Trojan Detection.

I. INTRODUCTION

Integrated Circuit (IC) design and production processes are becoming more and more globalized, and hardware Trojans have become a big menace to both production firms and customers. These trojans are a big menace because of large usage of integrated circuits (ICs) in mission-critical applications. Owing to the fact that, the complexity of ICs and the sophistication of their production have increased exponentially, it is no longer possible to go without relying on unreliable foundries and design tools [1]. A hardware trojan is any malicious code that generates abnormal behavior and is incorporated into an IC. Hardware trojan can infect a wide variety of types of integrated circuits (ICs): microprocessors, microcontrollers, FPGAs, ASICs, and network and digital signal processors. Hardware Trojan is among the most visible threats in the number of possible hardware threats because most organizations are outsourcing the production, testing and assembly of the chip. This has rendered hardware security and outsourcing of some of the chip component processes to be very tricky [2]. Hardware Trojan is referred to as the modification of the circuitry of the chip [3]. One can inject the sleeping Trojan at the manufacturing stage with ion beams that might interfere with the functioning functionality of the chip.

FPGAs are versatile devices suitable to various applications in addition to high-performance computing (HPC) and prototyping, which is why they are more suitable than more traditional logic devices such as application-specific integrated circuits (ASICs). Physical systems applications that interface with high-speed restrictions usually require a real-time operation. The FPGA technology has been widely used due to its low cost, high level of integration, flexibility and high level of modification [4]. The FPGAs excel when it comes to high-performance and real-time data processing and hardware customization, although the high-end microcontrollers are excellent in most situations. Using FPGAs has clear and appealing advantages over the microcontrollers. FPGAs are a promising new technology because of their high clock rates, low internal delays, fast operational speeds, plenty of internal RAM resources, and ability to manage complicated peripheral circuits in a detailed and precise fashion.

A custom-designed microchip that is specifically designed for a single, specific purpose or task is known as an Application-Specific Integrated Circuit (ASIC). In comparison to general-purpose processors, ASICs provide superior performance and efficiency. One key distinction between ASICs and FPGAs is that the former permanently imprints circuitry onto silicon, while the latter allows for the connecting of reconfigurable logic blocks [5][6]. This improves the area, power, and latency performance of ASIC. However, the FPGA provides greater adaptability and requires less time to implement. Investigating the similarities and differences between ASIC and FPGA performance design metrics is a worthwhile endeavor. Computer components Methods for detecting Trojans (HTs) have largely concentrated on design verification throughout development, particularly during phases such as layout study, simulation, and side channel analysis. Some examples of these techniques include examining side-channel signals like power usage and electromagnetic emissions, conducting thorough simulations to find inconsistencies, or comparing the actual circuit layout to the design requirements [7]. Machine learning and artificial intelligence have recently become effective tools for HT detection [8][9]. It uses algorithms to detect Trojans, even in more complex attacks, with detailed patterns. To determine if a circuit is Trojan-free or infected, ML models can be trained on various datasets. Processing in real time allows for constant surveillance and quick reaction to threats.

A. Structure of the Paper

The paper is organized as follows: Section II discusses the characteristics and types of hardware Trojans, Section III explores the threat landscape in FPGA and ASIC systems, Section IV sheds light on new hardware security trends and difficulties, Section V provides a comprehensive literature analysis of current detection methods, and Section VI

summarises the study's important findings and suggests avenues for further research.

II. Characteristics of Hardware Trojans

Trojan circuitry can purposefully destroy circuitry, disclose secret keys to adversaries, transfer unencrypted data to insecure channels, or wrongly execute intended functions [10]. A hardware Trojan gives an attacker an unfair advantage by changing the chip's input-output features. The scan approach is used to examine the device, which is implemented using flip-flops and combinational logic, and is assumed to be sequential. At any point in the production or supply chain, a hardware trojan attack can insert malicious circuitry into an integrated circuit (IC) [11]. Until they are activated, these trojans do nothing. Once they do, they can change features, leak important information, or even entirely disable the device. Table I compares and contrasts two types of hardware Trojan threats: threats and Trojans.

A. Emerging Hardware Trojan Threats

Hardware A new and serious danger to the security of integrated circuits (ICs) is the prevalence of trojan attacks, as seen in Fig. 1. Malicious alterations to an integrated circuit (IC) during its design or manufacture in an unreliable design house or foundry, including unreliable individuals, design instruments, or components, are the subject of these attacks [12]. Such alterations can cause an IC to behave in an undesirable way or open hidden pathways for sensitive data to be released. The goal of an adversary's Trojan horse program is to make it undetectable during standard post-production testing, but to spring into action during peak field usage times.

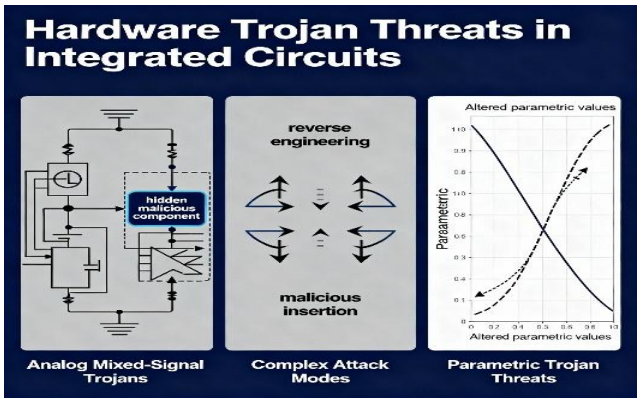

Fig. 1. Hardware Trojan Attacks and Threats

1) Analog and Mixed-Signal Trojans

Analog Trojans are malicious modifications to a circuit, either in the design phase or in the fabrication phase of the chip. These Trojans are designed to be activated under specific analog stimuli, such as voltage, temperature or process variations, and remain dormant until triggered [13][14]. The two main parts of a Trojan are the payload and the trigger. The trigger can be activated using software or an external pin.

2) Complex Attack Mode

Hardware Trojan attacks can be amplified when numerous individuals or groups work together maliciously throughout the design, production, and deployment phases. An instance where this may happen is when bad actors gain access to the encryption key through a hidden side channel made possible by Trojan hardware that has been implanted.

3) Parametric Trojan Threat

Side-channel analysis based on power and timing can reveal parametric effects on the circuit, such as changes in the power consumption profile and delays. One of the prerequisites for these methods is comparing the altered chip to a "golden chip" [15]. Therefore, rising process variability in integrated circuits frequently impact the efficacy of these methods. Since integrated circuits (ICs) can have millions of paths, timing-based SCA may not be feasible in that case.

B. Hardware Trojan

Hardware Trojan (HT) is a malignant modification of the circuitry of an integrated circuit. The sole distinguishing characteristics of a hardware Trojan are its physical appearance and behavior. The Trojan executes the entire activity upon being triggered, which is referred to as the payload of an HT. In general, Trojans endeavor to disable or circumvent the security barrier of a system, for example, by leaking confidential information through radio emission. The complete chip or its components could also be disabled, damaged, or destroyed by HTs, as illustrated in Fig. 2.

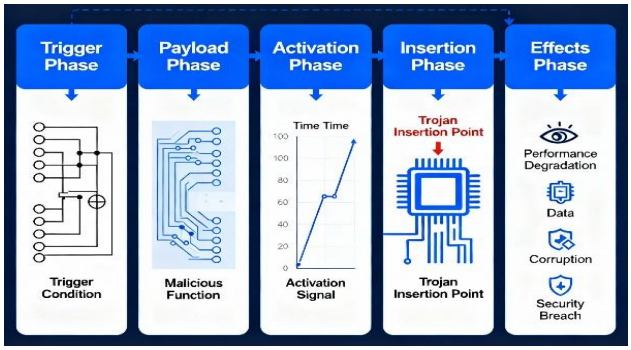

Fig. 2. Hardware Trojan System

1) Trojan Model

Hardware Trojans, a rising concern in the realm of electronic design automation (EDA) security, represent malicious modifications or inclusions within the hardware [16]. These manipulations are often subtle and sophisticated, making them difficult to detect but potentially devastating in their impact. The payload and the trigger are the two fundamental components of each hardware Trojan. Frequently, the Trojan is activated by a condition that is latent and inconspicuous.

2) Trojan Taxonomy

Hardware Trojans can be classified according to numerous critical attributes [17], as illustrated in Fig. 3.

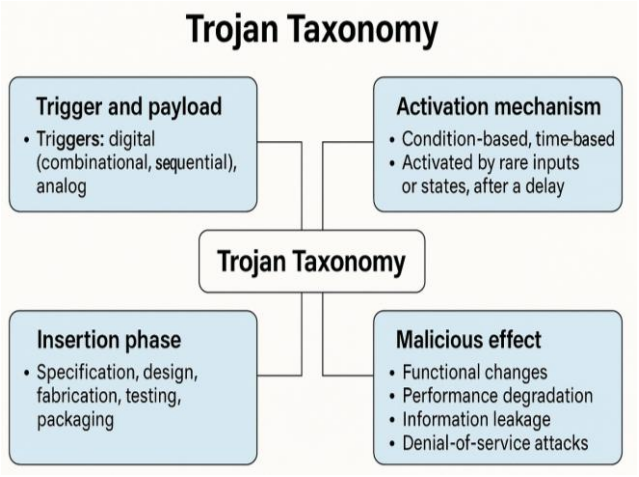

Fig. 3. Trojan Taxonomy

- **Trigger and payload:** Triggers may be digital or analog, with digital Trojans further divided into combinational types that target logic blocks and sequential types that

manipulate registers or control signals. Payloads can disrupt outputs, alter control flow, or leak information.

- **Activation mechanism:** Trojans may be condition-based, activated by rare inputs or states, or time-based, triggered after a specific delay.
- **Insertion phase:** They can be introduced during specification, design, fabrication, testing, or packaging, each posing unique challenges; fabrication-time Trojans are especially stealthy.
- **Malicious effect:** Their impact may include functional changes, performance degradation, information leakage, or denial-of-service attacks, making them a serious threat to system trustworthiness.

3) Threat Model

A hardware Trojan can be introduced at any point during the lifecycle of an integrated circuit. This paper assumes that the adversary has the capacity to modify RTL code or a netlist file in order to establish information leakage channels and access the hardware design [18]. Conversely, let us presume that the adversary is aware of the hardware design's functionalities and design details. Consequently, they can acquire the desired information by activating the Trojan and observing the Trojan's output.

TABLE I. COMPARATIVE OVERVIEW OF EMERGING HARDWARE TROJAN THREATS AND HARDWARE TROJANS

Aspect	Definition	Threat Nature	Types / Variants	Activation / Trigger	Impact
General Overview	Attacks against integrated circuits (ICs) in unsecure settings during design and fabrication.	Stealthy; may evade post-manufacturing tests and become apparent during field operation.	Analog, Digital, Parametric, Complex attack mode.	Can be triggered by specific conditions, stimuli, or adversary manipulation.	Causes undesired behavior, covert channels, or sensitive info leakage.
Analog & Mixed-Signal Trojans	Trojans designed for analog circuits; activated under specific stimuli.	Dormant until triggered; stealthy in nature.	Analog/Mixed-Signal Trojans	Triggered by voltage, temperature, or process variations.	Alters IC behavior, leaks sensitive data, or compromises functionality.
Digital Trojans	Trojans affecting digital logic or sequential circuits.	Difficult to detect; subtle triggers can bypass conventional testing.	Combinational and sequential digital Trojans	Condition-based, time-based, or rare-event triggers.	Disrupts outputs, control flow, or leaks critical information.
Complex Attack Mode	Collusion between multiple parties during design/fabrication.	Coordinated attacks increase potency and stealthiness.	Multi-party attack mode	Activated via covert channels or collaborative triggers.	Can reveal encryption keys or critical information to attackers.
Parametric Trojans	Modifications affecting IC delay, power, or timing characteristics.	Hard to detect due to process variations and circuit complexity.	Parametric Trojans	Detected via side-channel analysis comparing with golden chip.	Causes functional changes, performance degradation, or info leakage.
Lifecycle & Threat Model	Trojans can be inserted at any stage: design, fabrication, or deployment.	Adversaries exploit design knowledge to remain undetected.	Insertion during specification, design, fabrication, testing, or packaging.	Triggered through RTL/netlist manipulation, rare states, or environmental conditions.	Severe impact: functional disruption, info leakage, denial-of-service, or hardware damage.

III. THREAT LANDSCAPE IN FPGA AND ASIC SYSTEMS

Potential threats of the FPGA and ASIC systems include a variety of security threats caused by the complex design and manufacturing procedures. These are hardware Trojans, side-channel attacks, reverse engineering and IP piracy, which might lead to the compromising of integrity, confidentiality and functionality of the devices [19]. Trusted third-party IP cores, fabrication factory and design tools pose serious threats of malicious modification and data leakage. Also, there are physical tampering and fault injection attacks, which are even more dangerous to system reliability. With the growing adoption of FPGAs and ASICs in the critical infrastructures and defense systems, knowledge and mitigation of the vulnerabilities are crucial to create trust, resilience, and secure deployment of hardware.

A. FPGA-Specific Vulnerabilities

FPGAs are very flexible and are also susceptible because of security threats because they have a reconfigurable architecture. The system can be compromised if attackers get access to the bitstream, configuration memory, or side-channel information. Third party IP cores bring on board new threats such as the presence of Trojan. FPGA deployment on clouds also exposes vulnerability to tampering and data leakage even more. The security of configuration and trusted design tools is important in protecting FPGA some important secure vulnerabilities shown in Fig. 4.

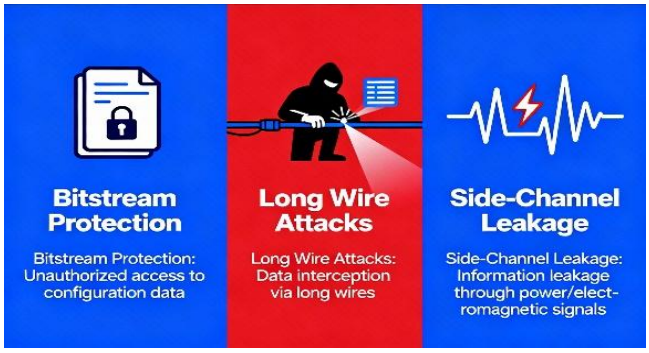

Fig. 4. Security Vulnerabilities in FGPA

1) Bitstream Protection

To protect the users' designs and their Intellectual Property from manipulation, reverse engineering, and tampering, almost all FPGA vendors have developed bitstream protection techniques such as bitstream encryption and authentication. Although these techniques have been employed for decades, providing robust bitstream protection mechanisms has been a vexing problem for the FPGA industry [20]. A large and constantly growing body of literature demonstrates that existing bitstream protection mechanisms can be defeated by side-channel and probing means, as well as implementation flaws.

2) Long Wire Attacks

The potential to significantly alter the threat level of multi-tenant FPGAs is present with the discovery of a covert

communication channel between adjacent FPGA long wires, which are also referred to as "long lines" [21][22]. Any long wire in a given channel affect the latency of both its nearby neighbor and another long wire two wires away, depending on the logic value it carries. Relative to when a logic 0 is sent, the delay on the adjoining wire (the receiver) is reduced when a logic 1 is carried on a wire (the transmitter). The direction of signal transmission in FPGA channels, the length of the wire on the FPGA, and the transmitter's signal switching rate have no bearing on this effect.

3) Malware and Intrusion Detection

Prior FPGA generations were vulnerable to side channel attacks (SCAs), which collect data through unintended channels in order to decrypt data [23][24]. The phenomenon of side-channel linkages FPGAs can be linked in ways that have not been intended; this may happen as a result of unwanted physical effects. The initial FPGA generations have applied power analysis to recreate the encryption keys by observing the change in power consumption during cryptography [25]. In the same way, layouts used in FPGA fabrics have been demonstrated to have sensitive information leakage via electromagnetic emissions.

B. ASIC-Specific Vulnerabilities

ASICs have security threats that are associated primarily with their fixed design and reliance on manufacturing chains on a global level see Fig. 5. Hardware Trojans can be placed in untrusted foundries or the circuits can be manipulated during their production [26]. The design confidentiality is also at risk as a result of reverse engineering and IP piracy. After ASICs are created, they cannot be readily reconfigured and thus mitigation after deployment is very hard. In order to prevent malicious intent, it is crucial to conduct manufacturing and design verification with trustworthy individuals.

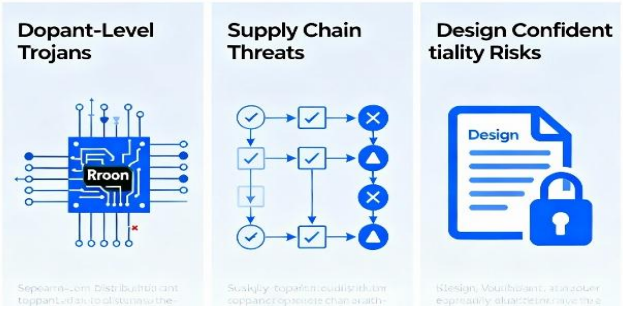

Fig. 5. Secure Vulnerabilities in ASIC

1) Dopant-Level Hardware Trojans

The fundamental concept of an efficient hardware design process Trojans can be implemented in target designs by altering the polarity of dopants applied to specific regions of the gate's active area, rather than by altering the metal or polysilicon layers themselves. These alterations mimic the method used for code obfuscation in certain commercial designs and make the target gate behave in a predictable manner.

2) Threats in the Semiconductor Supply Chain

ASICs are also founded on a complex web of supply chains throughout the world which exposes ASICs to a number of security threats [27]. A malicious modification, or design error, that compromises the integrity of the chip can be introduced by the origin of untrusted foundries, third-party IP suppliers, and design tools. Another factor of supply chain is trust to ensure that the occurrence of hardware Trojans is eliminated and ensure that devices are secure in their operations.

IV. Emerging Trends and Challenges

The hardware security is a rapidly developing sphere, and the following trends gain popularity nowadays: machine learning-based Trojan detection, advanced side-channel analysis, and runtime monitoring. In the meantime, there are still problems of Process Variation, Process Drift, design complexity and the need of scalable and cost-effective security mechanisms. These tendencies and issues are the most significant to consider in order to develop intensive safeguarding systems of the FPGA and ASIC systems of the modern world.

A. Process Variation

Physical constraints encountered during manufacture cause transistors' physical and electrical characteristics to vary, a phenomenon known as random process variation [28][29]. The detection of HTs is more difficult due to the manner in which random process variation affects the drive strength and delay of produced transistors, as the test engineer must differentiate between the delays caused by random process variation and the timing impact of an HT.

B. Process Drift

The standard cell libraries employed in the production process of a physical design house are characterized by a new technology node using SPICE models. Soon after the process achieves a consistent level, the results are disclosed to the public. A decent yield is ensured by padding the SPICE model and common cell libraries with conservative margins.

C. Secure Software Upgrades

The capacity to safely update devices while they are out in the field is essential for ensuring their security in the long run. To keep the system secure, though, these updates must be impenetrable [30]. Organizations use Trusted Platform Modules (TPMs) to provide security to these updates. Such modules enable them to patch any vulnerabilities without the fear of being further compromised.

D. Design Complexity

Increasingly intricate designs, FPGA and ASIC, millions of gates and convoluted interconnections, and it is more difficult to locate hardware Trojans and guarantee security. The bigger the design is, the better the possibility of undetected vulnerabilities, it takes more advanced methods of testing and analysis.

E. Scalability Issues

The hardware security of large-scale systems is a difficulty due to their scalability. The methods used to secure small circuits might not be viable or even affordable in high-density designs, and it is necessary to find security solutions which can be scalable without affecting performance or reliability.

V. Literature Review

This review highlights recent hardware Trojan detection methods, including coupling analysis, optimization-based test patterns, ring oscillators, TinyML-FPGA, EM-based benchmarks, current sensors, and AI attacks, aiming to improve anomaly detection, system reliability, and overall hardware security.

Haga et al. (2025) the possibility of detecting HTs implemented on an adjacent wire by focusing on the parasitic coupling between the sensing wire and its adjacent wires. Specifically, the influence of slight variations in the equivalent circuit caused by HT implementation was measured on the voltage distribution during the charge and measurement phases of the capacitance sensor under idealized conditions using an

evaluation board. As a result, changes were observed in the histogram distribution with and without HT implementation, indicating the potential for detecting HT [31].

S and Anita (2025) introduces a novel multi-objective optimization framework that targets these rare nodes to enhance Trojan detection. The proposed algorithm combines Ant Colony Optimization (ACO) and Non-Dominated Sorting Genetic Algorithm II (NSGA-II) to create the optimum test patterns of these weak spots. Trojans are placed deliberately in the rare nodes of the combinational benchmark circuits and NSGA-II is used to devise optimized tests pattern and which are optimized by the use of ACO to enhance detection capability as well as reducing the overall number of test patterns required [32].

Abe, Fujimoto and Hayashi (2024) suggests a technique based on a ring oscillator (RO), which in general can be performed by digital circuits, to measure electrical variations with the addition of HTs. In particular, the wiring of the RO installed within the IC is extruded and the fact that the propagation delay induced by the introduction of the HT on the wiring varies is observed as a change in the oscillation frequency of the RO [33].

Gurram, PK and Amsaad (2024) introduces a novel way of identifying hardware Trojan in real-time with the help of TinyML and FPGA technology. Through the use of the features of FPGA, timing and power information are acquired to identify abnormalities that may be used to point towards the presence of hardware Trojans. They use a training process that involves the use of an Isolation Forest machine learning model which is known to be very strong in detecting anomalies. The model is then converted to Verilog code and executed on the FPGA which enables the efficient and effective detection of Trojans as the program is being executed [34].

Takou et al. (2023) proposes an electromagnetic-based methodology of HT placement that generates benchmarks which are practically inaccessible with side-channel measurements. In order to reduce the effect of electrostatic stress on chip performance, this technique calculates it and incorporates the HT circuitry into clock tree networks. The technology generates demanding HT benchmarks with minimal stress current overhead, as shown experimentally on various clock tree networks [35].

Abedi et al. (2022) current-sensor is an improvement over previous power-sensing based methods; it can detect currents as low as 10s of nano-Amps. In addition, a current sensor with a programmable architecture is created to allow for detecting at broad ranges. The design is created to adhere to the digital design flow and is also illogical. To prevent analog Trojan activation in the field, use this detection approach during runtime to send out early warning signals. The proposed idea is validated using commercially available 65nm CMOS technology [36].

Pan and Mishra (2022) suggest a strong backdoor attack to expose this critical flaw in ML-based Trojan detection techniques. The suggested framework may create an AI Trojan, insert it into the ML model, and set it to activate in response to prescribed inputs. The proposed AI Trojans are able to evade even the most advanced security systems, according to the experimental outcome. The method also outperforms state-of-the-art adversarial attack-based approaches by a wide margin, and it does it quickly and cheaply [37].

Table II summarizes recent hardware Trojan detection studies, covering approaches such as parasitic coupling analysis, optimization-based test patterns, ring oscillators, TinyML-FPGA integration, EM-based benchmarks, current sensors, and AI-based attacks. Key findings, challenges, and future directions for improved detection and resilient designs are highlighted

TABLE II. COMPARATIVE ANALYSIS SUMMARY OF RECENT APPROACHES FOR HARDWARE TROJAN DETECTION

Reference	Study On	Approach	Key Findings	Challenges / Limitations	Future Directions
Haga et al. (2025)	Detecting HTs on adjacent wires	Parasitic coupling measurement using capacitance sensor and histogram analysis	Voltage distribution changes clearly observed with and without HT presence	Limited to idealized evaluation board conditions and simplified setup	Extend method to practical large-scale on-chip environments
S and Anita (2025)	Rare node-based Trojan detection	Multi-objective optimization using NSGA-II combined with Ant Colony Optimization	Optimized test patterns significantly improve detection effectiveness while reducing test count	Focused mainly on combinational benchmark circuits only, lacking broader coverage	Apply methodology to sequential circuits and industrial-scale ICs
Abe, Fujimoto & Hayashi (2024)	RO-based Trojan detection	Ring oscillator frequency variation analysis due to HT wiring insertion	Oscillation frequency shifts reliably indicate hardware Trojan presence	Requires external wiring extension and additional design overhead	Fully integrate within IC design for improved scalability
Gurram, PK & Amsaad (2024)	Real-time HT detection with TinyML	TinyML with FPGA combined with Isolation Forest anomaly detection	Achieved high-efficiency Trojan detection during active program execution	Complexity in Verilog code translation and deployment issues	Expand ML integration with other anomaly detection algorithms
Takou et al. (2023)	EM-based HT placement benchmarks	EM-induced hydrostatic stress computation within clock tree networks	Generated HT benchmarks are extremely difficult to detect via side-channels	Limited to EM stress in clock-tree structures, narrow scope	Extend framework applicability to diverse digital circuit components
Abedi et al. (2022)	Run-time analog Trojan detection	Nano-Amp current sensor featuring configurable wide sensing range	Detects analog Trojans at run-time, CMOS compatible and logic obfuscated	Sensor design complexity and limited scalability with newer nodes	Enhance sensing range and adapt to advanced nanoscale CMOS technologies
Pan & Mishra (2022)	AI-based backdoor attack on Trojan detection	Backdoor Trojan embedded inside ML detection frameworks	Achieved 100% attack success rate while bypassing state-of-art defenses	Highlights vulnerabilities in ML-based Trojan detection approaches	Develop robust defense strategies against adversarial AI Trojans

VI. CONCLUSION AND FUTURE WORK

The growing reliance on third-party design houses and global fabrication processes has made integrated circuits increasingly susceptible to hardware Trojans, posing significant threats to device reliability, performance, and information security. This study reviews key detection and mitigation techniques for

hardware Trojans in FPGA and ASIC systems, including side-channel monitoring, optimization-based test generation, ring oscillator sensing, TinyML-FPGA integration, and current-sensor detection. Each approach offers unique advantages but faces challenges such as process variations, detection scalability, and stealthy activation mechanisms. Machine learning and AI-driven solutions are transforming detection by enabling real-time, data-driven anomaly analysis. The findings emphasize the need for robust, multi-layered hardware security frameworks that combine physical, analytical, and intelligent detection methods to ensure resilience against evolving Trojan threats.

As Trojan detection systems become more complex and diverse, future research should focus on improving scalability and adaptability. Integrating AI and federated learning models can enable continuous learning from distributed data without compromising confidentiality. Strengthening collaboration between design, fabrication, and verification stages is essential for end-to-end trust in the hardware supply chain. Developing lightweight runtime detection architectures for resource-constrained FPGA and ASIC applications and creating standardized benchmarks and open datasets will be vital for fair evaluation across threat models and technology nodes.

REFERENCES

[1] S. Moein, F. Gebali, T. A. Gulliver, and A. Alkandari, "Hardware Trojan Identification and Detection," Int. J. Cryptogr. Inf. Secur., vol. 7, pp. 1–20, 2017, doi: 10.5121/ijcis.2017.7201.

[2] M. Hussain et al., "Hardware Trojan Mitigation Technique in Network-on-Chip (NoC)," Micromachines, vol. 14, no. 4, 2023, doi: 10.3390/mi14040828.

[3] U. A. Korat and A. Alimohammad, "A Reconfigurable Hardware Architecture for Principal Component Analysis," Circuits, Syst. Signal Process., vol. 38, no. 5, pp. 2097–2113, 2019, doi: 10.1007/s00034-018-0953-y.

[4] Y. Zhu and K. Hou, "Development and Implementation of an FPGA-Embedded Multimedia Remote Monitoring System for Information Technology Server Room Management," Int. J. Digit. Multimed. Broadcast., vol. 2024, no. 1, p. 4420578, 2024, doi: https://doi.org/10.1155/2024/4420578.

[5] S. Y. Neyaz, I. Saxena, N. Alam, and S. A. Rahman, "FPGA and ASIC Implementation and Comparison of Multipliers," in 2020 International Symposium on Devices, Circuits and Systems (ISDCS), 2020, pp. 1–4. doi: 10.1109/ISDCS49393.2020.9263027.

[6] U. A. Korat, P. Yadav, and H. Shah, "An efficient hardware implementation of vector-based odd-even merge sorting," in 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), IEEE, Oct. 2017, pp. 654–657. doi: 10.1109/UEMCON.2017.8249010.

[7] R. Khamitkar and R. R. Dube, "Analyzing Hardware Trojans in FPGA-Based SoCs Using Machine Learning," J. Integr. Circuits Syst., vol. 20, no. 1, pp. 1–9, 2025, doi: 10.29292/jics.v20i1.971.

[8] R. Vishwakarma and A. Rezaei, "Uncertainty-Aware Hardware Trojan Detection Using Multimodal Deep Learning," Proc. -Design, Autom. Test Eur. DATE, 2024, doi: 10.23919/date58400.2024.10546558.

[9] R. Q. Majumder, "Machine Learning for Predictive Analytics: Trends and Future Directions," Int. J. Innov. Sci. Res. Technol., vol. 10, no. 4, pp. 3557–3564, May 2025, doi: 10.38124/ijisrt/25apr1899.

[10] Z. Zhou, U. Guin, and V. D. Agrawal, "Modeling and test generation for combinational hardware Trojans," Proc. IEEE VLSI Test Symp., vol. 2018-April, pp. 1–6, 2018, doi: 10.1109/VTS.2018.8368626.

[11] J. Dofe, W. Danesh, V. More, and A. Chaudhari, "Natural Language Processing for Hardware Security: Case of Hardware Trojan Detection in FPGAs," Cryptography, vol. 8, no. 3, 2024, doi: 10.3390/cryptography8030036.

[12] S. Bhunia, M. Hsiao, M. Banga, and S. Narasimhan, "Hardware Trojan Attacks: Threat Analysis and Countermeasures," Proc. IEEE, vol. 102, pp. 1229–1247, 2014, doi: 10.1109/JPROC.2014.2334493.

[13] J. Chaudhuri, A. Chaudhuri, and K. Chakrabarty, "LATENT: LLM-Augmented Trojan Insertion and Evaluation Framework for Analog Netlist Topologies," Proc. - 2025 IEEE Int. Conf. LLM-Aided Des. ICLAD 2025, pp. 133–140, 2025, doi: 10.1109/ICLAD65226.2025.00040.

[14] R. Patel and P. Patel, "The Role of Simulation & Engineering Software in Optimizing Mechanical System Performance," Tech. Int. J. Eng. Res., vol. 11, no. 6, pp. 991–996, 2024, doi: 10.56975/tijer.v11i6.158468.

[15] R. Kumar, P. Jovanovic, W. Burleson, and I. Polian, "Parametric trojans for fault-injection attacks on cryptographic hardware," Proc. - 2014 Work. Fault Diagnosis Toler. Cryptogr. FDTC 2014, pp. 18–28, 2014, doi: 10.1109/FDTC.2014.12.

[16] V. Rajavel, "Integrating Power-Saving Techniques into Design for Testability of Semiconductors for Power-Efficient Testing," vol. 07, no. 243, pp. 243–251, 2025, doi: 10.37547/tajet/Volume07Issue03-22.

[17] X. Wei, J. Zhang, and G. Luo, "Rethinking IC Layout Vulnerability: Simulation-Based Hardware Trojan Threat Assessment with High Fidelity," in Proceedings - IEEE Symposium on Security and Privacy, 2024, pp. 3789–3804. doi: 10.1109/SP54263.2024.00160.

[18] D. Li et al., "Hardware Trojan Detection Using Effective Property-Checking Method," Electronics, vol. 11, no. 17, 2022, doi: 10.3390/electronics11172649.

[19] S. Narang and A. Gogineni, "Zero-Trust Security in Intrusion Detection Networks: An AI-Powered Threat Detection in Cloud Environment," Int. J. Sci. Res. Mod. Technol., vol. 4, no. 5, pp. 60–70, Jun. 2025, doi: 10.38124/ijsrmt.v4i5.542.

[20] E. Puschner, M. Ender, S. Becker, and C. Paar, "Patching FPGAs: The Security Implications of Bitstream Modifications," in ASHES 2024 - Proceedings of the 2024 Workshop on Attacks and Solutions in Hardware Security, Co-Located with: CCS 2024, 2024, pp. 89–99. doi: 10.1145/3689939.3695779.

[21] C. Ramesh et al., "FPGA Side Channel Attacks without Physical Access," Proc. - 26th IEEE Int. Symp. Field-Programmable Cust. Comput. Mach. FCCM 2018, pp. 45–52, 2018, doi: 10.1109/FCCM.2018.00016.

[22] A. R. Bilipelli, "AI-Driven Intrusion Detection Systems for Large- Scale Cybersecurity Networks Data Analysis : A Comparative Study," TIJER – Int. Res. J., vol. 11, no. 12, pp. 922–928, 2024.

[23] A. Duncan, F. Rahman, A. Lukefahr, F. Farahmandi, and M. Tehranipoor, "FPGA bitstream security: A day in the life," in Proceedings - International Test Conference, 2019, pp. 1–10. doi: 10.1109/ITC44170.2019.9000145.

[24] S. S. S. Neeli, "Critical Cybersecurity Strategies for Database Protection Against Cyber Attacks," J. Artif. Intell. Mach. Learn. Data Sci., vol. 1, no. 1, pp. 2102–2106, Nov. 2022, doi: 10.51219/JAIMLD/sethu-sesha-synam-neeli/461.

[25] V. Shah, "Managing Security and Privacy in Cloud Frameworks: A Risk with Compliance Perspective for Enterprises," Int. J. Curr. Eng. Technol., vol. 12, no. 06, pp. 1–13, 2022, doi: 10.14741/ijcet/v.12.6.16.

[26] G. Modalavalasa and S. Pillai, "Exploring Azure Security Center : A Review of Challenges and Opportunities in Cloud Security," ESP J. Eng. Technol. Adv., vol. 2, no. 2, pp. 176–182, 2022, doi: 10.56472/25832646/JETA-V2I2P120.

[27] V. Rajavel, "Optimizing Semiconductor Testing: Leveraging Stuck-At Fault Models for Efficient Fault Coverage," Int. J. Latest Eng. Manag. Res., vol. 10, no. 2, pp. 69–76, Mar. 2025, doi: 10.56581/IJLEMR.10.02.69-76.

[28] K. I. Gubbi et al., "Hardware Trojan Detection Using Machine Learning: A Tutorial," ACM Trans. Embed. Comput. Syst., vol. 22, no. 3, 2023, doi: 10.1145/3579823.

[29] V. Prajapati, "Advances in Software Development Life Cycle Models: Trends and Innovations for Modern Applications," J. Glob. Res. Electron. Commun., vol. 1, no. 4, pp. 1–6, 2025.

[30] S. P. Kalava, "AI-Powered Development: How Artificial Intelligence is Shaping Software Productivity," J. Artif. Intell. Cloud Comput., vol. 3, no. 2, pp. 1–4, Apr. 2024, doi: 10.47363/JAICC/2024(3)E148.

[31] R. Haga, S. Kaji, D. Fujimoto, and Y. Hayashi, "Detection of Hardware Trojans Using a Capacitance Sensor Focused on Parasitic Coupling Between Wires," in 2025 23rd IEEE Interregional NEWCAS Conference (NEWCAS), 2025, pp. 104–107. doi: 10.1109/NewCAS64648.2025.11107121.

[32] A. S and J. P. Anita, "Rare Node Targeted Trojan Detection Using Test Pattern Optimization With NSGA-II and ACO," in 2025 IEEE 5th International Conference on VLSI Systems, Architecture, Technology and Applications (VLSI SATA), 2025, pp. 1–6. doi: 10.1109/VLSISATA65374.2025.11070041.

[33] K. Abe, D. Fujimoto, and Y. Hayashi, "Fundamental Study on Detecting Hardware Trojans in Printed Circuit Boards Using Ring Oscillators," in 2024 14th International Workshop on the Electromagnetic Compatibility of Integrated Circuits (EMC Compo), 2024, pp. 1–4. doi: 10.1109/EMCCompo61192.2024.10742048.

[34] M. R. Gurram, M. K. PK, and F. Amsaad, "Isolation Forest Based TinyML for Detecting Hardware Trojans on FPGA in Real Time," in 2024 IEEE Physical Assurance and Inspection of Electronics (PAINE), 2024, pp. 1–5. doi: 10.1109/PAINE62042.2024.10792760.

[35] A. Takou, O. Axelou, G. Floros, N. Evmorfopoulos, and G. Stamoulis, "An Optimal Methodology for EM-Based Hardware Trojan Placement on Clock Tree Networks," in 2023 IEEE 66th International Midwest Symposium on Circuits and Systems (MWSCAS), 2023, pp. 25–29. doi: 10.1109/MWSCAS57524.2023.10405928.

[36] M. Abedi, T. Yang, Y. Fei, and A. Shrivastava, "High-Precision Nano-Amp Current Sensor and Obfuscation based Analog Trojan Detection Circuit," in 2022 IEEE International Symposium on Circuits and Systems (ISCAS), 2022, pp. 3324–3328. doi: 10.1109/ISCAS48785.2022.9937796.

[37] Z. Pan and P. Mishra, "Design of AI Trojans for Evading Machine Learning-based Detection of Hardware Trojans," in 2022 Design, Automation & Test in Europe Conference & Exhibition (DATE), 2022, pp. 682–687. doi: 10.23919/DATE54114.2022.9774654.

