INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD)
© 2026 IJNRD | Volume 11, Issue 3, March 2026 | ISSN: 2456-4184 | IJNRD.ORG

Threat Modeling and Security Requirements for Dynamic Reconfigurable UAV Flight Control Systems

[1] Kalakurasa Rakesh, [2] Moturi Satyanarayana,
[1] Associate Professor, [2] Professor,
[1] ECE Department,
[1] MVGR College of Engineering, Vizianagaram, India

Abstract : Unmanned Aerial Vehicles (UAVs) have evolved into autonomous cyber–physical systems operating in adversarial and safety-critical environments. While recent research has emphasized hardware-assisted security and reconfigurable System-on-Chip (SoC) architectures, a systematic problem formulation addressing security threats in dynamically reconfigurable UAV flight control systems remains largely unexplored. This paper presents a structured threat modelling framework tailored for UAV flight controllers implemented on reconfigurable SoC platforms. We identify adversarial capabilities, characterize attack surfaces across cyber, physical, and reconfiguration layers, and derive security requirements specific to dynamic partial reconfiguration. The proposed formulation establishes a foundational problem statement for adaptive, hardware-rooted security architectures in next-generation UAV systems

IndexTerms **- UAV Security, Threat Modeling, Flight Control Systems, Reconfigurable SoC, FPGA-SoC, Hardware Security, Dynamic Partial Reconfiguration**

I.INTRODUCTION

The FPGA market now offers a variety of reconfigurable devices and systems thanks to developments in integration technology. There is fierce rivalry among FPGA manufacturers as a result of the devices' ongoing expansion. Shortening the time to market for the product or device and winning over the competitors has led to shorter design time and the requirement for rapid prototyping. The third-party IP core usage in design always boosts the design cycle, leading to reduced time to market. Furthermore, third-party IP cores from various vendors could also potentially carry a Trojan embedded in the design, disrupt the intended operation of the system, and diminish the trust level of integrated circuit being designed. Malicious code in these chips can lead to unintentional leak of private information in turn allowing unauthorized access. In order to sell the product in the market, the end users must have a guarantee that the designs are hardware Trojan-free and exhibited acceptable behavior. Hardware Trojan is referred to an extraneous logic coupled to the original design of the system in the logic design phase or after fabrication. Due to the high cost of facility maintenance and the ongoing need for improvements, fewer fabrication units are operating globally. A lot of manufacturers depend on these facilities owned by third parties for their manufacturing because the majority of design units lack IC fabrication facility. As system density levels rise, maintenance costs rise and time to market decreases and increased competition from manufacturers, the designers rely on third-party fabrication units for their designs to get fabricated. It is a challenging aspect to narrow down the existence of hardware bugs in form of Trojans inserted inside a design, as these remain hidden deep in the logic designed for the system functionality. There are different approaches proposed for revealing the source and type of hardware Trojans. By analysing the behavior of golden chips, estimating hardware defects by estimating thermal maps, electromagnetic leakage, or any operational changes will determine whether a Trojan is present in a circuit. Most of the methods compare the operational characteristics of the affected chip designed with a golden-chip known as a Trojan free chip for the output sequence [18]. As the complexity of the design requirements increases, these golden chips become increasingly rare and expensive to use for all fault identification.

II.BACKGROUND AND MOTIVATION

2.1 Evolution of UAV Flight Control Architectures

The original UAVs' flight control systems were based solely around one microcontroller, which executed a limited number of predetermined flight controls and did not connect to any other system; however, the current UAV configurations contain multiple types of processing units from different manufacturers (PCs, RTs, GPUS, RPL);
therefore, UAVs have benefited from the use of advanced functionalities like autonomous navigation, onboard sensory feedback, and adaptive mission planning; however, this has increased the overall complexity of UAV hardware designs, making them less able to support FP-SoC designs with time-critical feedback processes implemented in hardware but higher-level functions implemented as software, using dynamic partial reconfiguration where necessary to reconfigure/change access, control functions, fault recovery modules and/or security accelerator modules during operation without having to turn off all parts of the UAV (e.g., kill switch).

2.2 Security Challenges Introduced by Reconfiguration

Most existing UAV security frameworks assume static hardware and predictable execution environments [6], [12]. Dynamic reconfiguration violates these assumptions by allowing hardware functionality to change at runtime. While this capability can be leveraged for adaptive defense, it also introduces challenges such as:

- Trust establishment for dynamically loaded hardware modules
- Timing uncertainty during reconfiguration
- Increased attack surface at the configuration interface

These challenges motivate the need for a reconfiguration-aware threat model.

III.SYSTEM MODEL AND ASSUMPTIONS

3.1 UAV Flight Control System Model

The system under consideration consists of multiple tightly coupled subsystems:
• Sensor subsystem: IMU, GNSS, magnetometer, barometer
• State estimation: Sensor fusion and filtering
• Control computation: Attitude, position, and trajectory control
• Actuation: Motor and servo control
• Communication: Telemetry and command channels
• Security and monitoring: Hardware security primitives and runtime monitors
The platform is assumed to be implemented on an FPGA-SoC, with clearly defined static and reconfigurable regions. Control loops operate at deterministic frequencies (typically hundreds of Hz to 1 kHz), making timing predictability a critical requirement [17].

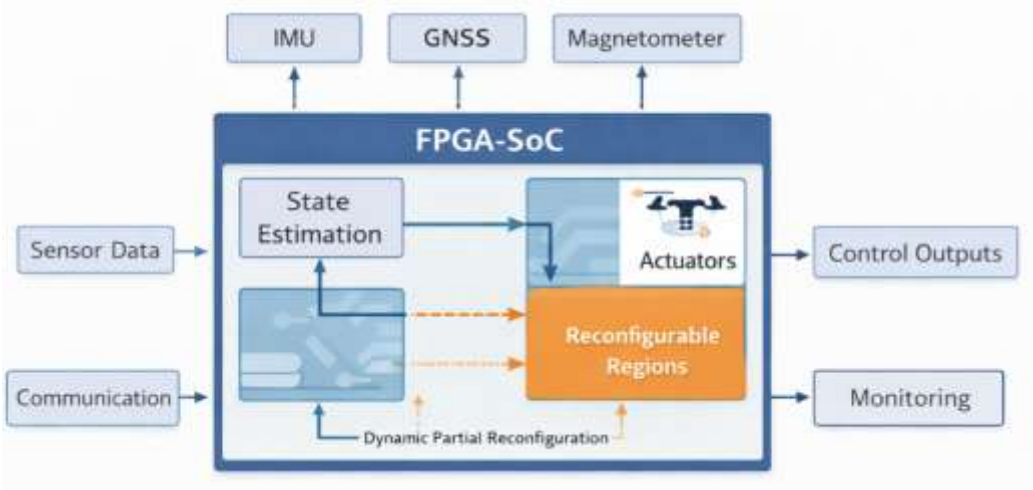

Figure 1. System model of a UAV flight control platform implemented on a reconfigurable SoC.

The figure illustrates the integration of sensing, control computation, communication, and reconfigurable hardware regions under hard real-time constraints. Dynamic partial reconfiguration enables adaptive functionality while preserving continuous flight control operation [4], [13], [17].

3.2 Adversary Model

The adversary model extends beyond conventional cyber attackers to include reconfiguration-aware adversaries. The attacker may:

- Exploit wireless interfaces to inject malicious commands
- Manipulate sensor inputs to mislead control algorithms
- Tamper with firmware or configuration files
- Physically access the device to perform side-channel or fault injection attacks
- Target dynamic partial reconfiguration mechanisms

This comprehensive adversary model reflects realistic threat scenarios for UAVs deployed in untrusted or hostile environments [7], [21].

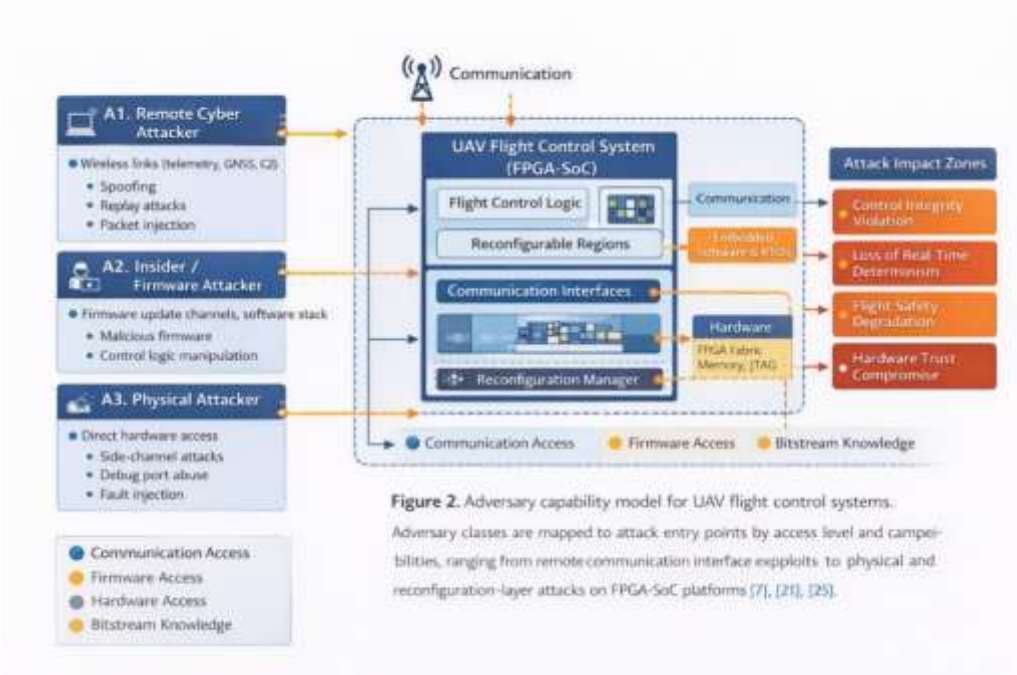

Figure 2. Adversary capability model for UAV flight control systems.
Adversary classes are mapped to attack entry points by access level and capabilities, ranging from remote communication interface exploits to physical and reconfiguration-layer attacks on FPGA-SoC platforms [7], [21], [25].

Figure 2. Adversary capability model for UAV flight control systems

The adversary is categorized based on access level and attack capabilities, ranging from remote wireless access to physical and reconfiguration-layer attacks on FPGA-SoC platforms [7], [21].

IV. Threat Model for Dynamic Reconfigurable UAV Flight Control Systems

4.1 Multi-Layer Threat Taxonomy

Threats are categorized across multiple layers, reflecting the cyber–physical nature of UAV systems:
• Communication-layer threats: GNSS spoofing, command injection, replay attacks [11], [12]
• Sensor-layer threats: IMU saturation, sensor bias injection, fault attacks [23]
• Control-layer threats: Manipulation of control laws, destabilization of feedback loops [3], [17]
• Software/Firmware threats: Unauthorized updates, malicious task insertion [19], [24]
• Reconfiguration-layer threats: Bitstream tampering, malicious partial reconfiguration, reconfiguration abuse [25]
This layered taxonomy highlights how attacks can propagate across abstraction boundaries, ultimately impacting flight safety. Unlike conventional UAV threat models, this representation explicitly incorporates the reconfiguration layer, capturing threats that arise from dynamic hardware modification and partial bitstream management [6], [11], [25].

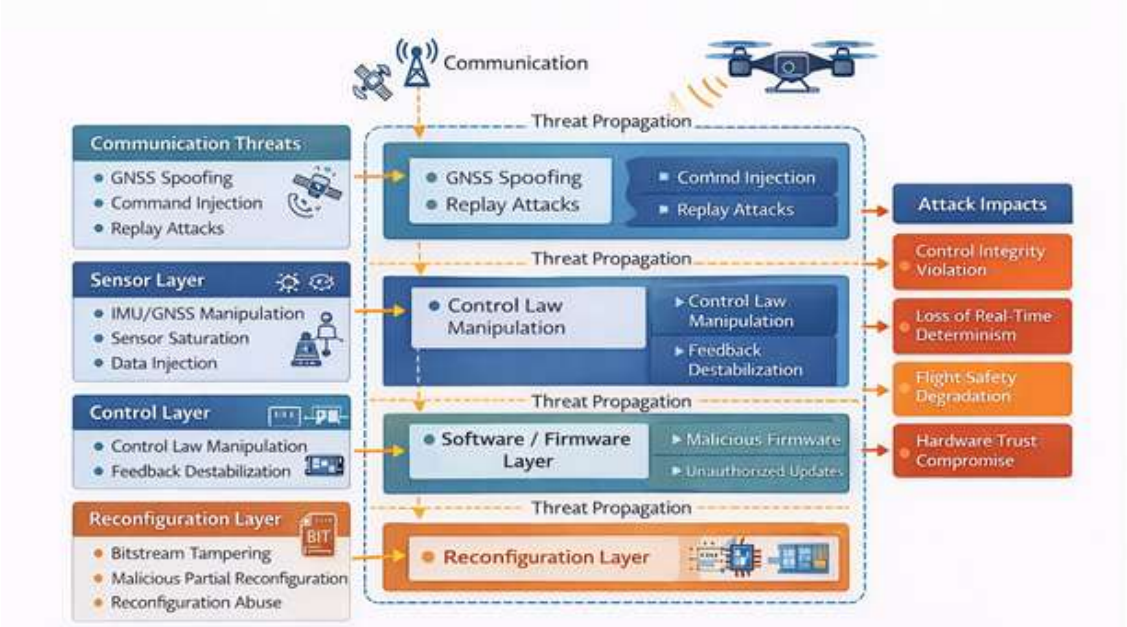

FIGURE 3. MULTI-LAYER THREAT MODEL FOR RECONFIGURABLE UAV FLIGHT CONTROL SYSTEMS.

4.2 Reconfiguration-Specific Threats

Dynamic partial reconfiguration introduces unique threats not present in static systems. An attacker may:

- Inject malicious partial bitstreams that modify control or security logic
- Trigger frequent reconfiguration to induce denial-of-service
- Exploit reconfiguration timing windows to disrupt real-time execution
- Bypass isolation boundaries between reconfigurable regions

Such attacks are particularly dangerous because they may remain undetected while operating below the software layer [13], [25].

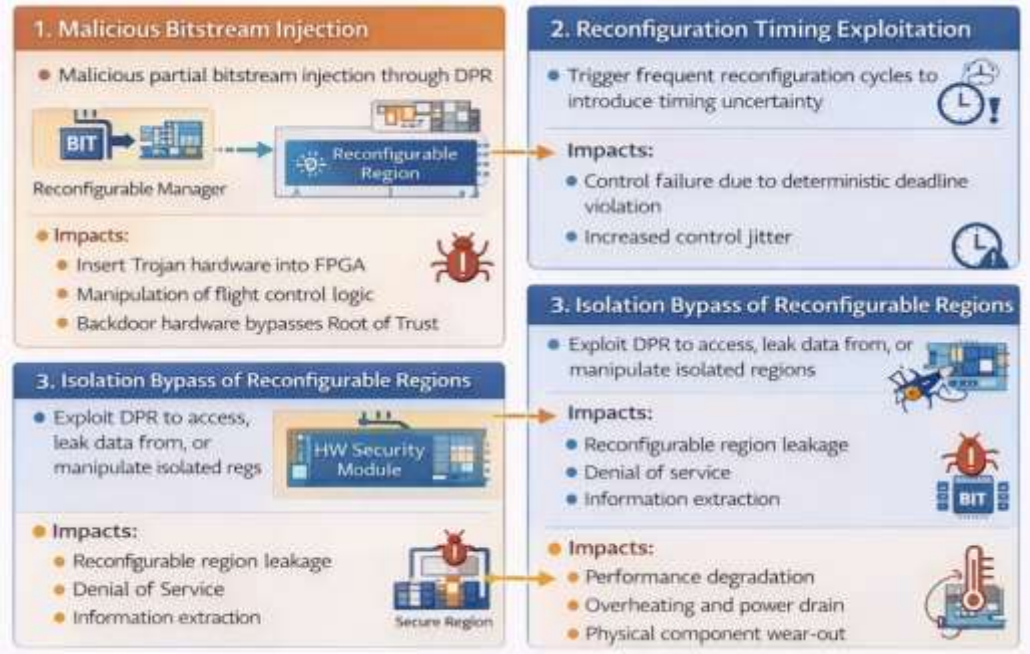

Figure 4. Attack scenarios targeting dynamic partial reconfiguration in UAV flight control systems.

The figure illustrates how adversaries may exploit reconfiguration interfaces and timing windows to disrupt control execution or compromise trusted hardware regions [13], [25].

`V. Problem Formulation

5.1 Core Research Problem

The central question addressed in this paper is: how do UAV flight control systems provide safe and secure

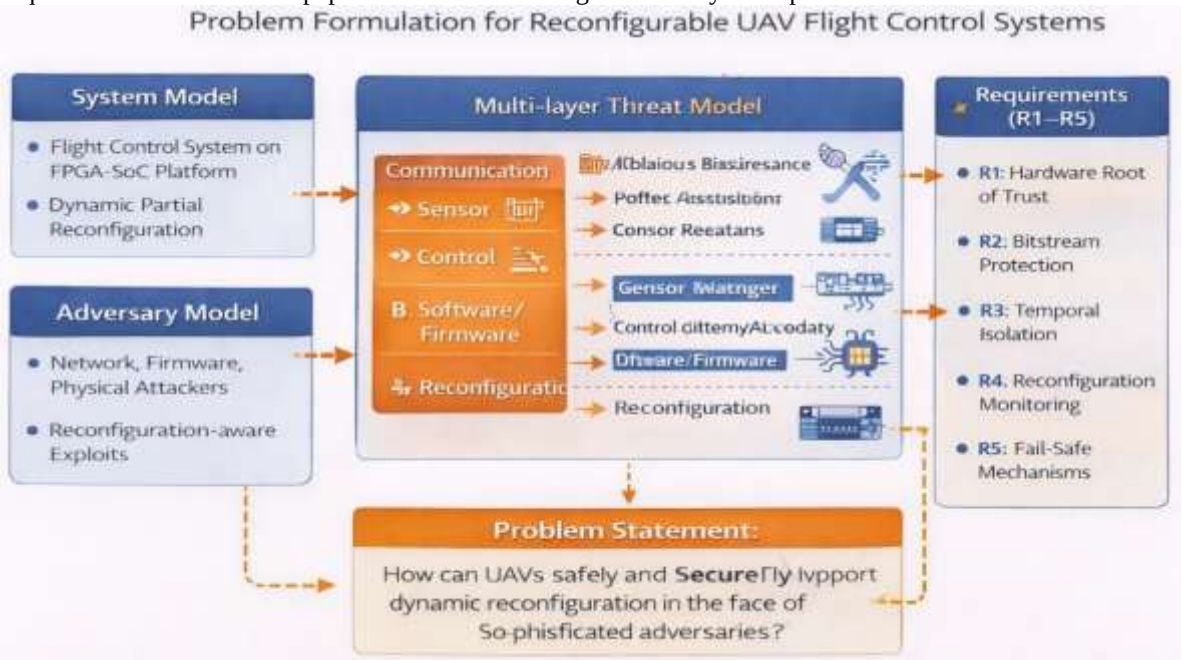

Figure 5. Core problem formulation for secure dynamic reconfigurable UAV flight control systems.

The figure highlights the fundamental trade-off between adaptive security, real-time determinism, and flight safety that motivates the need for hardware-assisted and formally constrained reconfigurable architectures [14], [17], [18].

5.2 Security Requirements for Reconfigurable UAV SoCs

From the threat analysis, the following security requirements are derived:

- **R1:** Establish a hardware root of trust for all reconfiguration operations [14], [21]
- **R2:** Ensure authentication, integrity, and confidentiality of partial bitstreams [13], [25]
- **R3:** Guarantee temporal isolation between control execution and reconfiguration [17]
- **R4:** Provide runtime monitoring and anomaly detection for reconfiguration behavior [10], [18]
- **R5:** Support fail-safe and recovery mechanisms during reconfiguration failures

These requirements form a bridge between threat modeling and architecture design.

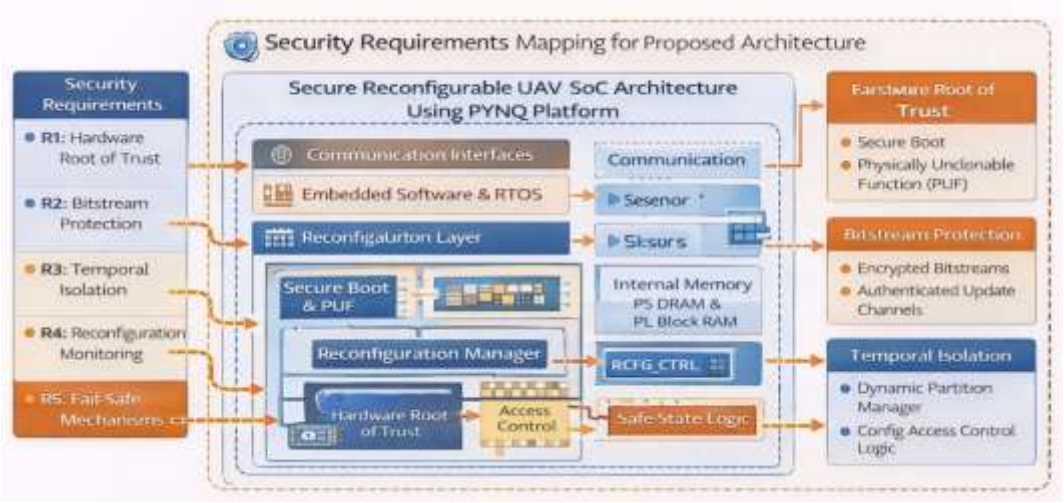

Figure 6. Mapping of identified threats to derived security requirements for reconfigurable UAV SoCs

This figure demonstrates how the proposed threat model directly informs hardware-rooted security requirements, forming a foundation for architecture design and evaluation [10], [21], [25].

VI. Research Implications and Open Challenges

The analysis reveals several open research challenges:

- Formal verification of dynamically reconfigurable control logic
- Quantifying the impact of reconfiguration latency on control stability
- Coordinating security adaptation with certification requirements
- Designing lightweight monitoring mechanisms suitable for resource-constrained UAVs

Addressing these challenges is essential for translating reconfigurable security concepts into deployable UAV systems.

VII. Conclusion

This paper presented an in-depth problem formulation and threat model for dynamic reconfigurable UAV flight control systems. By explicitly incorporating reconfiguration-layer threats and deriving security requirements, the work provides a rigorous foundation for future research on adaptive, hardware-assisted UAV security. The insights presented here motivate the development of formally constrained, real-time-aware reconfigurable architectures capable of sustaining secure and safe UAV operation in adversarial environments.

References

[1] R. Austin, Unmanned Aircraft Systems: Uavs Design, Development And Deployment, 2nd Ed. Chichester, U.K.: Wiley, 2010.
[2] R. Beard And T. Mclain, Small Unmanned Aircraft: Theory And Practice. Princeton, Nj, Usa: Princeton Univ. Press, 2012.
[3] K. Hartmann And C. Steup, "The Vulnerability Of Uavs To Cyber Attacks—An Approach To The Risk Assessment," In Proc. Ieee Int. Conf. Unmanned Aircraft Systems (Icuas), Orlando, Fl, Usa, 2013, Pp. 1–5.
[4] Y. Shoukry, P. Martin, P. Tabuada, And M. Srivastava, "Non-Invasive Spoofing Attacks For Anti-Lock Braking Systems," In Proc. Ches, 2013, Pp. 55–72.
[5] M. Psiaki And T. Humphreys, "Gnss Spoofing And Detection," Proc. Ieee, Vol. 104, No. 6, Pp. 1258–1270, Jun. 2016.
[6] H. Kopetz, Real-Time Systems: Design Principles For Distributed Embedded Applications, 2nd Ed. New York, Ny, Usa: Springer, 2011.
[7] A. Wasicek, E. A. Lee, And S. Seshia, "Challenges In The Design Of Cyber-Physical Systems," Proc. Ieee, Vol. 100, No. 1, Pp. 28–40, Jan. 2012.

[8] S. Trimberger, "Three Ages Of Fpgas: A Retrospective On The First Thirty Years Of Fpga Technology," *Proc. Ieee*, Vol. 103, No. 3, Pp. 318–331, Mar. 2015.
[9] S. Mittal, "A Survey Of Fpga-Based Accelerators," *Acm Comput. Surveys*, Vol. 49, No. 2, Pp. 1–35, Jun. 2016.
[10] Xilinx Inc., *Zynq-7000 Soc Technical Reference Manual*, San Jose, Ca, Usa, 2020.
[11] Xilinx Inc., *Partial Reconfiguration User Guide*, Ug702, San Jose, Ca, Usa, 2021.
[12] K. Vipin And S. A. Fahmy, "A Survey Of Partial Reconfiguration In Fpgas," *Acm Comput. Surveys*, Vol. 51, No. 4, Pp. 1–39, Aug. 2018.
[13] Nist, *Guide To Hardware Roots Of Trust*, Nistir 8320, 2018.
[14] G. E. Suh And S. Devadas, "Physical Unclonable Functions For Device Authentication," In *Proc. Dac*, 2007, Pp. 9–14.
[15] R. Maes, *Physically Unclonable Functions*. Berlin, Germany: Springer, 2013.
[16] A. Hodjat And I. Verbauwhede, "Area-Throughput Trade-Offs For Fully Pipelined Aes Processors," *Ieee Trans. Computers*, Vol. 55, No. 4, Pp. 366–372, Apr. 2006.
[17] M. Tehranipoor And F. Koushanfar, "A Survey Of Hardware Trojan Taxonomy And Detection," *Ieee Design & Test*, Vol. 27, No. 1, Pp. 10–25, Jan.–Feb. 2010.
[18] S. Mangard, E. Oswald, And T. Popp, *Power Analysis Attacks*. New York, Ny, Usa: Springer, 2007.
[19] S. Skorobogatov, "Semi-Invasive Attacks," Univ. Of Cambridge, Tech. Rep. Ucam-Cl-Tr-630, 2005.
[20] M. Tunstall, S. Mukhopadhyay, And S. Ali, *Differential Fault Analysis Of Cryptographic Algorithms*. Cham, Switzerland: Springer, 2017.
[21] M. Peiffer, A. Dehon, And S. Devadas, "Towards Practical Secure Fpga Reconfiguration," In *Proc. Ieee Fccm*, Napa, Ca, Usa, 2014, Pp. 129–136.
[22] J. Rajendran, O. Sinanoglu, And R. Karri, "Is Split Manufacturing Secure?" *Ieee Design & Test*, Vol. 30, No. 2, Pp. 84–90, Apr. 2013.
[23] A. Perrig, J. Stankovic, And D. Wagner, "Security In Wireless Sensor Networks," *Commun. Acm*, Vol. 47, No. 6, Pp. 53–57, Jun. 2004.
[24] P. Kocher, J. Jaffe, And B. Jun, "Differential Power Analysis," In *Proc. Crypto*, 1999, Pp. 388–397.
[25] S. Seshia, D. Sadigh, And S. Sastry, "Toward Verified Artificial Intelligence," *Commun. Acm*, Vol. 61, No. 7, Pp. 56–67, Jul. 2018.