Special Session: Security Verification of Microelectronic Systems with Integrated AI Accelerators: Scope, Practice, and Challenges

Kazi Mejbaul Islam, Tambiara Tabassum, Dipal Halder, Sandip Ray
Department of ECE, University of Florida at Gainesville, Gainesville, FL 32611. USA.
kmejbaulislam@ufl.edu, tambiaratabassum@ufl.edu, dipal.halder@ufl.edu, sandip@ece.ufl.edu

2025 IEEE 43rd VLSI Test Symposium (VTS) | 979-8-3315-2144-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/VTS65138.2025.11022911

***Abstract*—The rapid advancement of artificial intelligence (AI) has resulted in creation of a vast array of accelerator hardware, including GPUs, TPUs, and FPGAs, alongside the latest ASICs, to efficiently train and deploy AI models. However, AI accelerators are vulnerable to security threats that can compromise sensitive information, such as model architecture and jeopardize operational integrity, leading to unreliable applications. While significant efforts have been made for security verification of AI accelerators, existing techniques have inherent limitations and struggle to keep pace with evolving attack strategies. In this paper, we present a comprehensive analysis of the evolving field of security verification methodologies for AI accelerators, critically examining their effectiveness and identifying key limitations. Furthermore, we explore emerging trends in the field and outline potential research directions that could be pursued to enhance the security verification of AI accelerators.**

***Index Terms*—Verification, security**

I. Introduction

Recent years have seen an explosive proliferation of Artificial Intelligence (AI) application in a variety of areas, including computer vision, speech processing, robotics, industrial automation, and natural language processing [1]–[3] ushering in a new era of AI revolutions. AI computations have been traditionally deployed in the cloud, exploiting the availability of high-performance servers to enable computation-intensive learning tasks. However, there has been increasing demands for edge intelligence in diverse Internet-of-Things (IoT) applications of high societal impact, including smart surveillance, healthcare, and autonomous vehicles. Consequently, AI Hardware is being increasingly embedded into accelerators that are integrated into edge devices. AI accelerators are poised to become even more critical to embedded computing in the future, as privacy and connectivity concerns push more and more AI applications from the cloud towards edge devices [4], [5]. On the other hand, AI accelerators are well-known to be susceptible to a variety of subversions, attacks, and misuse. Attacks on AI hardware, already demonstrated in vast body of past research, has been large and varied, ranging from physical and side-channel attacks on one hand to data poisoning, evasion, and model inversion on the other [6]. Unsurprisingly, in a recent survey by Deloitte, 23% of executives cited security risks as their primary concern in AI and 32% of surveyed companies say that they have experienced a cybersecurity breach related to AI initiatives in the past two years [7]. Attacks on AI hardware when deployed on mission-critical applications can have catastrophic consequences on our national security and societal safety. ***As we increasingly depend on AI for critical decision-making in such applications, it is crucial to ensure that these systems cannot be subverted or misused when deployed under real-life conditions.***

In spite of the criticality of the challenge, we have not found a comprehensive unified treatise on the scope and spectrum of security verification of microelectronic systems induced by integrating AI accelerators. While there has been significant research in this area, these research results are sprinkled across different research articles often devoted to specific applications such as vision, speech, automotives etc. as well as a plethora of journals and conference proceedings devoted to different aspects of cybersecurity, AI, and hardware designs. This makes it daunting for a security researcher getting initiated into this area to get a sense of the scope of the problem, what has been done so far to address it, and the limitations of current practice that can be targeted with new research.

In this paper, we provide a broad "lay of the land" in security of microelectronics systems with integrated AI accelerators. The emphasis of the paper is on recent research, current practice, and limitations. In particular, we summarize the existing verification methodologies, their scope and shortcomings. In addition, we address the gaps that need to be filled to stay in tune with this rapidly evolving field. We also discuss future research direction in this field which will help the community to develop more secured microelectronic systems.

II. Security Challenges in AI Accelerators

AI Accelerators can be subjected to a plethora of security attacks, that can target their confidentiality, integrity, and availability. Table 1 provides a sampling of some known attacks. Note that each attack category includes a variety of *attack methods* depending on the deployment as well as the

TABLE I
ATTACK CLASSIFICATION AND MECHANISMS

Attack Class	Attack Mechanism	Description
Data Manipulation	Evasion	Modifying input data to fool the model
	Poisoning	Tampering with training data to influence the model's learning process
Model Manipulation	Model Stealing	Repeated querying of an AI model to extract enough information for cloning or replication
	Model Inversion	Reverse-engineering the model's decision process through repeated queries to reconstruct inputs like private training data
Side Channel	EM Radiation	Exploiting electromagnetic emissions to gain information about the processing tasks or data
	Power Analysis	Monitoring the power consumption to deduce information about ongoing operations
	Timing	Exploiting the time to perform computations to infer details about the data being processed or the algorithms being used
Physical Attack	Fault Injection	Creating system instability or corruption by introducing glitches in computation through voltage or clock manipulation, laser attacks, etc.
	EM Interference	Using electromagnetic waves to interfere with hardware operations, which can cause data corruption, system crashes, or unwanted behavior
	Sensor Manipulation	Manipulating the input data that hardware sensors capture (e.g., cameras, microphones, or environmental sensors) to fool the AI system into making incorrect decisions

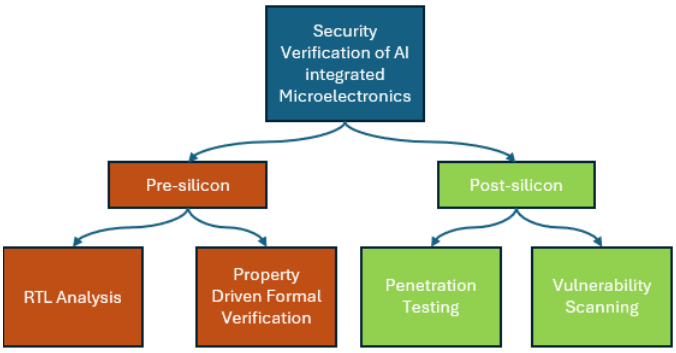

Fig. 1. Current Security Verification Landscape for AI Accelerators

level of access the attacker has on the accelerator. For instance, fault attacks can be mounted by an attacker by manipulating voltage, skewing clocks, or introducing electromagnetic interference or malicious RF signals [8]. Furthermore, deployed AI systems include the AI functionality integrated deep within a platform containing many other hardware and software components, each of which can also be attacked under diverse field condition resulting in subversion of AI functionality. Correspondingly, while many attacks have been analyzed and mitigations proposed, each such mitigation is a point fix against a previously known attacks and can be easily subverted by a new attack mechanism. Consequently, security of AI hardware has devolved into a cat-and-mouse game, with a litany of diverse attack mechanisms being proposed on one hand and increasingly sophisticated defense mechanisms on the other, with little systematic guarantee on defense against potential future attacks possibly identified after deployment.

III. CURRENT SECURITY VERIFICATION LANDSCAPE

The spectrum of security verification technologies developed in recent research has been large and varied. For this paper, we decompose the verification technology into *pre-silicon* and *post-silicon* components. Fig, 1 provides a rough taxonomy of the activities involved, During pre-silicon, formal verification, simulation-based testing, and hardware-assisted verification aim to uncover design vulnerabilities before the hardware is fabricated. Post-silicon verification targets fabricated (pre-production) silicon and includes techniques like side-channel analysis, in-system testing, penetration testing and vulnerability scanning to help uncover vulnerabilities that may only appear in physical implementations.

A. Pre-silicon Verification Portfolio

Pre-silicon security verification primarily relies on RTL level analysis. Both static verification techniques (code analysis, property driven formal verification) and dynamic verification techniques such as fuzzing are widely used to assess security of the design. When successful, these techniques have the advantage of early vulnerability detection, possibly obviating a costly silicon spin. However, the techniques are limited by scalability and generally cannot target multiple attack combinations. For microelectronic systems, side channel attacks and fault attacks have also been explored in pre-silicon verification. In side channel attack, the attacker probes any signal from the device such as power, electromagnetic interference to extract meaningful information. Woudenberg *et al.* [9] proposed SCATE, a pre-silicon side channel vulnerability detection framework leveraging power estimation and simulation using design automation tools. EDA tool's estimated power profile analysis informs the designer about potential information leakage. In addition, they have used Formal Property Verification (FPV) for the assessment of fault injection countermeasures in the design. But the effect of fault attack on the design may vary on the fault attack mechanism. Wang *et al.* [10] have developed SoFI, a formal method based property driven fault injection vulnerability assessment system. SoFI assesses fault attack vulnerability based on the security

properties and fault injection technique's specification. It is notable that if the attacker works with entirely new fault injection method, SoFI may not be able to assess the vulnerability against that appropriately. Additionally, hardware Trojan is a serious threat for security critical applications which can cause data leak, denial of service or corrupted operations at runtime. To detect hardware trojan in a design, Yasaei *et al.* [11] proposed graph neural network based hardware Trojan detection at RTL level.

For processing units, another concerning security threat is timing attack where the attacker uses timing information to guess secret information about the system or computation. Computation units continuously keep loading and storing data back and forth so timing attacks are difficult to eradicate without knowing the leakage paths. Borkar *et al.* [12] proposed a white box fuzzing to detect timing vulnerabilities in processors at RTL level. Leveraging simulation, their tool is able to detect timing vulnerabilities automatically.

Recent work from the authors' group has explored pre-silicon evaluation of AI accelerators for vulnerability against stealthy integrity attacks, specifically targeting multi-tenant computations. The idea is to systematically induce vulnerabilities in the accelerator based on an attack model, and identify ways in which the attack can corrupt AI computation or data before the AI model is integrated into a specific accelerator. The attacks are *stealthy* in the sense that the adversary model enables the AI model to function after the attack but produce targeted misclassification. The attacks explored include RowHammer attack [13] and fault attacks [14]. However, these works are preliminary and only focus on the inherent characteristics of the model itself rather than the idiosyncrasies induced by their implementation on specific accelerators.

B. Post-silicon Verification Portfolio

Post-silicon validation targets pre-production silicon for security validation. This enables running tests at real clock speed, but is constrained by limited observability. Post-silicon validation also provides an avenue for detecting attacks through non-functional channels (*e.g.*, side channels and fault injection). For instane, recent research has shown that clock perturbation is enough to create glitch and fault in a system resulting in misprediction or crash in the AI implementation. In such cases, after validating the device against fault injection vulnerability, the designer also needs to consider how the application deployed on the device behaves under fault. To observe the effect of fault injection attacks on AI models deployed on NVIDIA GPUs, NVBitFl [15] is a popular framework. It helps the application designer to observe how the model behaves under bit flips, which is one of the expected outcomes of fault injection.

The industrial practice in post-silicon security validation entails *peneration testing*, where the idea is to "hack" the system to identify security vulnerabilities. Penetration testing for AI systems can detect vulnerabilities such as bootloader attack, microprobing, timing-attack etc. Unfortunately, penetration testing is limited by its strong reliance on human expertise, particularly against new (0-day) threats.

C. Industry Practice

Emerging AI startups and leading cloud computing companies are developing advanced AI chips such as GroqChip, NVIDIA H100 GPU, Ambarella CV52S, Atlazo AZ-N1, AWS Trainium, and Google TPU v4. These innovations are enhancing speed and efficiency for AI acceleration. NVIDIA's H100 GPU brings confidential computing to the forefront, ensuring workloads run securely within a Trusted Execution Environment (TEE), anchored by an on-die hardware root of trust. This setup helps ensure that sensitive computations remain isolated and protected from unauthorized access or manipulation. The H100 is built to defend against a range of threats, including software-based exploits, physical attacks, cryptographic breaches, rollback attempts, and replay attacks. Its hardware-enforced security mechanisms helps ensure computations remain tamper-proof, even in hostile environments. To reinforce this security, the GPU employs attestation mechanisms that verify its integrity and isolation. These hardware-backed assurances provide cryptographic proof that workloads are running in a trusted and uncompromised state. Additionally, NVIDIA has collaborated with Intel to offer comprehensive attestation services for H100 GPUs to further bolster their security posture [16]. NVIDIA's Blackwell architecture also leverages confidential computing to achieve nearly the same performance as unencrypted modes when running large LLMs [17]. Ambarella CV52S AI chip features enhanced secure boot capabilities with ARM TrustZone® [18]. While publicly available information on the security verification of Ambarella CV52S is limited, various verification methodologies have been explored concerning the security properties of the TrustZone TEE [19]. IBM Telum is the first server-class chip with a dedicated on-chip AI accelerator, designed specifically for IBM Z and LinuxONE systems [20]. Its design prioritizes zero downtime by embedding on-chip error detection and correction mechanisms that prevent faulty computations from being written back to memory instead of placing error-checking logic in the critical path. Security is enforced through Telum's TEE, which ensures encrypted memory access and isolates workloads from system administrators. To further bolster confidentiality, the system enforces firmware-based access controls and automatically clears temporary data.

IV. Research Gap and Future Direction

Underexplored Emerging AI Hardware Architectures: Security verification research has mostly been focusing on popular accelerators, especially GPUs. There has not been much attention in terms of security for emerging devices like Tensor Processing Units (TPUs) and Deep Learning Processing Units (DPUs). Along with them, there are some ASICs which are specifically designed to train and deploy large language models such as Groq's Language Processing Units (LPUs), Sambanova System's SN40L but these are not yet explored from a security perspective. All these architectures

are different from each other and attacks may have different effect on different architecture. To use these accelerators for critical security applications, security researchers need to verify and validate them against known attacks.

Absence of Standardized Security Verification Platform: Despite rapid advancements in GPU, DPU, TPU and other AI architectures, a notable challenge is the lack of comprehensive, standardized verification platforms specifically tailored to assess thesecurity vulnerabilities of these architectures. Existing verification tools primarily emphasize performance optimization and general correctness, leaving a critical gap in hardware security evaluation [21]. This absence complicates vulnerability identification and hampers the development of robust and trustworthy hardware systems.

Targeted Exploration of Emerging Architectures and Collaborative Ecosystems: Despite the prevalence of GPUs, innovative architectures have not been fully evaluated for security. To address these, specific approaches are needed that are suitable for the specific data stream and computational patterns of these systems. Furthermore, an open-source collaborative environment will also enhance the disclosure of design information, test tools, vulnerability reports, and security benchmarks. This collaborative strategy not only extends the understanding of the possible threats but also provides recommendations that can be useful to the entire ecosystem.

Unified, Standardized, and Scalable Verification Frameworks: Developing universally accepted security benchmarks and a unified verification flow is essential for addressing the security of complex AI accelerators. This approach calls for standardized metrics and evaluation protocols that facilitate reproducible experiments and fair comparisons across diverse platforms, while integrating pre-silicon simulation, formal verification, and post-silicon testing into a validation continuum. Furthermore, as AI designs grow in scale and heterogeneity, exploiting tools based on AI or machine learning to automate verification processes will be critical.

V. Conclusion

Security verification of microelectronic system integrated with AI accelerator is a complex and evolving field that demands constant attention to keep pace with technological advancements. We have outlined some of the key limitations in current practices and identified areas where more research is needed. While there are existing methodologies that provide a foundation for security assurance, it is clear that there are still significant gaps considering the unique characteristics of AI hardwares. Advancements in technology and evolving attacks underscore the necessity of more comprehensive, scalable, and adaptable verification practices.

References

[1] S. M. Siniscalchi, T. Svendsen, and C.-H. Lee, "An artificial neural network approach to automatic speech processing," *Neurocomputing*, vol. 140, pp. 326–338, 2014.

[2] B. Alshemali and J. Kalita, "Improving the reliability of deep neural networks in nlp: A review," *Knowledge-Based Systems*, vol. 191, p. 105210, 2020.

[3] M. Kawato, Y. Uno, M. Isobe, and R. Suzuki, "Hierarchical neural network model for voluntary movement with application to robotics," *IEEE Control Systems Magazine*, vol. 8, no. 2, pp. 8–15, 1988.

[4] J. Chen and X. Ran, "Deep learning with edge computing: A review," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1655–1674, 2019.

[5] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A. Y. Zomaya, "Edge intelligence: The confluence of edge computing and artificial intelligence," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7457–7469, 2020.

[6] M. Isakov, V. Gadepally, K. M. Gettings, and M. A. Kinsy, "Survey of attacks and defenses on edge-deployed neural networks," in *2019 IEEE High Performance Extreme Computing Conference (HPEC)*, 2019, pp. 1–8.

[7] Deloitte, "Accelerators program," https://www2.deloitte.com/us/en/pages/about-deloitte/articles/executive-accelerators-development-program.html.

[8] C. H. Kim and J.-J. Quisquater, "Faults, injection methods, and fault attacks," *IEEE Design & Test of Computers*, vol. 24, no. 6, pp. 544–545, 2007.

[9] J. van Woudenberg, P. Grossmann, A. L. Varna, J. Friel, D. Dinu, R. Lindsay, and S. J. Brown, "Invited: Pre-silicon side channel and fault analysis," in *2023 60th ACM/IEEE Design Automation Conference (DAC)*, 2023, pp. 1–4.

[10] H. Wang, H. Li, F. Rahman, M. M. Tehranipoor, and F. Farahmandi, "Sofi: Security property-driven vulnerability assessments of ics against fault-injection attacks," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 3, pp. 452–465, 2022.

[11] R. Yasaei, S.-Y. Yu, and M. A. Al Faruque, "Gnn4tj: Graph neural networks for hardware trojan detection at register transfer level," in *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2021, pp. 1504–1509.

[12] P. Borkar, C. Chen, M. Rostami, N. Singh, R. Kande, A.-R. Sadeghi, C. Rebeiro, and J. J. Rajendran, "Whisperfuzz: white-box fuzzing for detecting and locating timing vulnerabilities in processors," in *Proceedings of the 33rd USENIX Conference on Security Symposium*, ser. SEC '24. USA: USENIX Association, 2024.

[13] X. Chen, M. Merugu, J. Zhang, and S. Ray, "Aroma: Evaluating deep learning systems for stealthy integrity attacks on multi-tenant accelerators," *J. Emerg. Technol. Comput. Syst.*, vol. 19, no. 2, Mar. 2023. [Online]. Available: https://doi.org/10.1145/3579033

[14] X. Chen, D. Halder, K. M. Islam, and S. Ray, "Work-in-progress: Towards evaluating cnns against integrity attacks on multi-tenant computation," in *Proceedings of the International Conference on Compilers, Architecture, and Synthesis for Embedded Systems*, ser. CASES '23 Companion. New York, NY, USA: Association for Computing Machinery, 2024, p. 7–8. [Online]. Available: https://doi.org/10.1145/3607889.3609091

[15] O. Villa, M. Stephenson, D. Nellans, and S. W. Keckler, "Nvbit: A dynamic binary instrumentation framework for nvidia gpus," in *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, ser. MICRO '52. New York, NY, USA: Association for Computing Machinery, 2019, p. 372–383. [Online]. Available: https://doi.org/10.1145/3352460.3358307

[16] N. C. Intel, "Intel, nvidia collaborate to deliver confidential ai solutions that strengthen ai security, privacy," *https://community.intel.com/t5/Blogs/Products-and-Solutions/Security/Intel-Nvidia-Collaborate-to-Deliver-Confidential-AI-Solutions/post/1500066*.

[17] "Nvidia blackwell architecture," https://www.nvidia.com/en-us/data-center/technologies/blackwell-architecture/.

[18] "Arm trustzone, building a secure system using trustzone technology," https://developer.arm.com/documentation/100690/0200/ARM-TrustZone-technology.

[19] H. Sun and H. Lei, "A design and verification methodology for a trustzone trusted execution environment," *IEEE Access*, vol. 8, pp. 33 870–33 883, 2020.

[20] C. Lichtenau, A. Buyuktosunoglu, R. Bertran, P. Figuli, C. Jacobi, N. Papandreou, H. Pozidis, A. Saporito, A. Sica, and E. Tzortzatos, "Ai accelerator on ibm telum processor: Industrial product," in *Proceedings of the 49th Annual International Symposium on Computer Architecture*, 2022, pp. 1012–1028.

[21] "Security risks of ai hardware for personal and edge computing devices," https://www.nccgroup.com/us/research-blog/public-report-security-risks-of-ai-hardware-for-personal-and-edge-computing-devices/.