Survey of Recent Developments for Hardware Trojan Detection

Ayush Jain, Ziqi Zhou and Ujjwal Guin
Dept. of Electrical and Computer Engineering, Auburn University
Emails: {ayush.jain, ziqi.zhou, ujjwal.guin}@auburn.edu

***Abstract*—The outsourcing of the design and manufacturing of Integrated Circuits (ICs) poses a severe threat to our critical infrastructures as an adversary can exploit them by bypassing the security features by activating a hardware Trojan. These malicious modifications in the design introduced at an untrusted fabrication site can virtually leak any secret information from a secure system to an adversary. This paper discusses all three different hardware Trojan models, such as combinational, sequential, and analog Trojans. We provide a survey of the recent advancements in Trojan detection techniques classified based on their applicability to different Trojans types. We describe a practical approach recently developed using the characterization of Electro-Optical Frequency Mapping (EOFM) images of the chip to detect a hardware Trojan by identifying malicious state elements. This survey also presents open problems with Trojan detection and suggests future research directions in hardware Trojan detection.**

***Index Terms*—Hardware Trojan, Tampering.**

I. Introduction

Over the past few decades, the semiconductor industry has witnessed prodigious advances in designing and manufacturing integrated circuits (ICs). However, the massive cost for building and maintaining a fabrication unit or foundry [1] has propelled the system-on-chip (SoC) design house to outsource their production. The design, fabrication, assembly, and tests are performed by different entities located offshore. This globalized semiconductor supply chain comes with more security threats than ever before due to the inclusion of various untrusted entities. One such threat is the insertion of hardware Trojans inserted at an untrusted IC production site, which is one of the leading concerns for the industry, government, and academic research [2]–[5]. In general, a hardware Trojan is a malicious alteration or inclusion of additional malicious circuitry to the original design to modify its functionality so that an adversary can gain control of the system or leak protected critical information. Such modification are demonstrated to expose the security-critical information from the hardware implementation of cryptographic devices [6], [7], IP designs [8], [9], through wireless channels [10], system failure [5], and many other malicious activities [11].

An abstract representation of the steps involved in the development of an SoC component is shown in Figure 1. The design can be maliciously modified with a hardware Trojan at any stage of IC development till assembly and packaging [5]. However, researchers have mainly focused and studied the Trojans injected either at the design or fabrication phase,

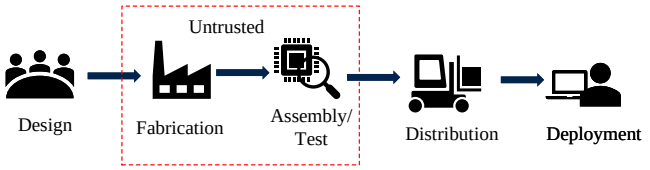

Figure 1. Modern IC supply chain, where malicious entities capable of implanting a hardware Trojan are represented as untrusted (red).

especially at an untrusted manufacturing site. Moreover, since the manufacturing and production tests are performed at the foundry, they can access the test patterns to design stealthy Trojans that do not get activated during manufacturing tests. Note that this paper mainly focuses on Trojan insertion at an untrusted foundry and their detection processes. However, a Trojan can be placed by an untrusted third-party IP (3PIP) vendors, and its detection is beyond the scope of this paper.

The defense techniques against the threat of hardware Trojans emerging from an untrusted foundry can be classified into two categories: detection and prevention of Trojans. The detection methods can be grouped into two different categories, such as logic testing [12]–[19], and side-channel analysis [20]–[29]. On the other hand, prevention methods can be categorized as design-for-trust measures [30], [31], and split manufacturing [32], [33]. This paper mainly focuses on prominent detection techniques and provides the survey based on detecting different types of hardware Trojans.

This survey aims to familiarize the research community with the accomplishments of recent works towards the modeling and detection of hardware Trojans. The contributions of this survey are:

- It updates the community on the recent research on detecting hardware Trojans as significant research has been performed over the years.
- We discuss the different Trojan designs studied by the researchers so far. Additionally, we identify the similarity between the characteristics and functionality of different types of Trojans. The modeling of Trojans can help us to evaluate the effectiveness of various detection techniques.
- We believe this survey will provide novel directions toward hardware Trojan detection using image processing based approaches.

The rest of the paper is organized as follows: the classification of hardware Trojans based on their design is presented in Section II. The existing detection techniques and their

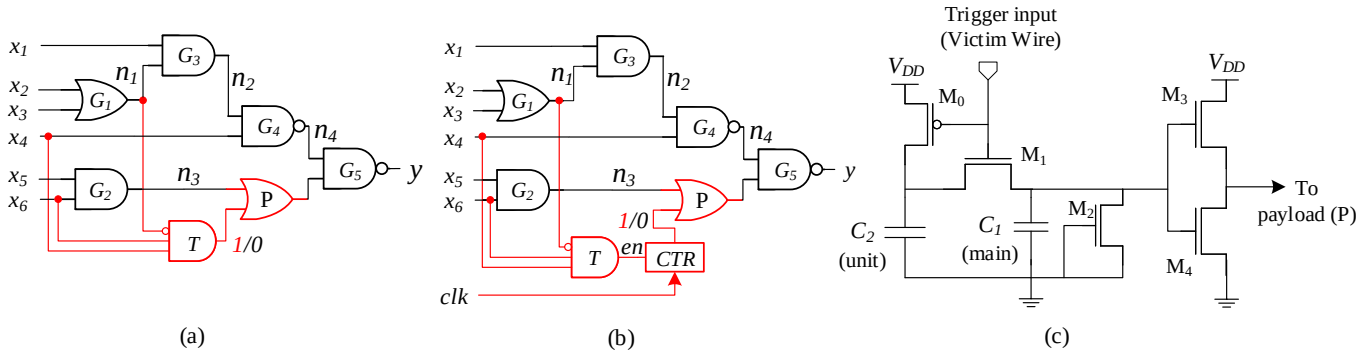

Figure 2. Design of different types of hardware Trojans, (a) *Type-3* combinational Trojan [15], (b) Type-3 Sequential Trojan [8] and (c) A2 analog Trojan [34].

effectiveness over different types of Trojans are presented in Section III. Finally, we conclude this survey by mentioning future directions in Section IV.

II. Hardware Trojan Design

Hardware Trojans can be identified as the intentional malicious modification of the original design without the knowledge of the SoC house. The taxonomy of Trojans can be broadly classified based on their designs. This includes combinational Trojan (*i.e.,* the addition of combinational logic gates), sequential Trojan (*i.e.,* the addition of state or memory elements), and analog Trojan (*i.e.,* utilization or addition of analog techniques and characteristics of the design). In this section, we provide the details for different types of Trojans.

A. Combinational Trojans

A combinational hardware Trojan comprises of a trigger that is taken from the primary inputs and/or internal nodes of a circuit and a payload that can be activated once the trigger is asserted [15]. Any Trojan design can be described based on the *p*-trigger inputs as *Type-p* Trojan. The simplest form of a Trigger can be designed from an AND gate with *p*-inputs. Any other combinational logic can also serve the purpose of trigger, which produces logic 1 upon activation. The foremost important property of a Trojan is to remain quiet during manufacturing and production tests (*e.g.,* stuck-at fault tests, and delay tests). In other words, the circuit should not come across any condition that activates the Trojan during scan-based structural or functional tests, which can lead to its detection. Generally, low probability switching nodes are selected as the trigger inputs for a combinational Trojan. Over the years, researchers have studied various combinational Trojan design, and the detailed modeling can be found in [8], [15], [35]–[37]. Figure 2.(a) represents the original circuit implanted with the combinational hardware Trojan along with the payload (P) OR gate. Upon activation, the output of the trigger (*i.e.,* AND gate) becomes logic 1; else it is always 0. Such combinational Trojans delivers the payload in the original netlist and manifests its effects once a unique specification condition is satisfied.

B. Sequential Trojans

Sequential Trojans deliver the payload upon the occurrence of a sequence of input patterns or after a period upon triggered. To achieve this goal, the trigger design of a sequential Trojan involves state elements along with the combinational logic [6], [8], [38]. Figure 2.(b) shows a sequential Trojan where the trigger consists of an AND gate and a counter (CTR). The trigger mechanism can be divided into two types: (i) every time the trigger condition is satisfied, *i.e.,* $en = 1$, counter increments, and (ii) once the trigger condition appears, the counter is activated, which increments with every clock after that. For the first approach, the payload is delivered only when the counter reaches its maximum count; in other words, the FSM for the counter reaches its last state. This property of sequential Trojan makes its detection even difficult, as it is highly unlikely that specific test patterns or inputs occur consecutively multiple times during the testing or normal operations of an IC.

C. Analog/RF Trojans

An adversary can also leverage the analog characteristics to design a hardware Trojan [39]. The implementation of the trigger, however, can be different for different analog/RF Trojan designs. Yang et al. proposed to use a capacitor(s) to design the trigger circuit, which is activated by accumulating the charge from the toggling of nearby victim wire that goes above a certain threshold. The voltage of the capacitor rises above the threshold when the wire frequently toggles because the charge starts accumulating on the capacitor faster than it leaks [34]. This capacitor-based trigger mechanism is shown in Figure 2.(c). A similar notion is utilized to introduce triggers that are activated after some delay or operate on a specific voltage threshold [40]. Analog Trojans are also designed using the coupling capacitor between the victim and aggressor wire in sub-micron process technologies [41] so that the low to high transition on the aggressor can adequately affect the victim wire and flip its digital value. Similarly, RF-leaking Trojans leak the information through the Trojan-induced channel without affecting the legitimate signal/channel [10], [42].

Note that an analog Trojan can also be modeled as a particular type of sequential Trojan as it requires either trigger multiple times or affects the circuit after a certain period once the Trojan is triggered. The only difference lies in the trigger design as sequential Trojan involves state elements (*e.g.,* counter), whereas analog Trojan involves discrete elements (*e.g.,* transistors and capacitors). Additionally, both

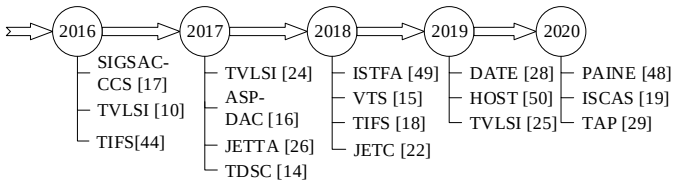

Figure 3. Timeline of recently proposed hardware Trojan detection techniques.

sequential and analog Trojans can be modeled using combinational Trojans.

III. Hardware Trojan Detection

The security of integrated circuits (IC) and Trojan detection are closely linked together to each other. Research work focused on post-silicon Trojan detection can be bifurcated into destructive and non-destructive approaches. IC reverse engineering in any form can be regarded as the prominent destructive method which delivers high-confidence results but leaves the chip unusable after processing. On the other hand, non-destructive approaches differ in the result accuracy (*i.e.,* lower-reliance); however, the chip can be used even after the analysis is performed. In this section, we discuss the recent approaches towards Trojan detection, *i.e.,* techniques developed after the survey presented in [5]. Figure 3 shows the timeline of newly proposed detection techniques along with some of the prominent solutions that gained significant interest from the research community.

A. Combinational Trojan Detection

State-of-art combinational detection techniques aim to stimulate the Trojan during post-silicon testing. Logic testing targets activate the rare nodes to trigger the Trojan through the test vectors generated by pre-silicon design analysis. It relies on monitoring the responses at primary/observable outputs (POs) to detect any mismatch [12]–[16], [43]. Amongst the many test pattern-based techniques proposed so far, Zhou et al. demonstrated how generating test patterns targeting a single net trigger (*Type-1* Trojan) can be beneficial in detecting higher-order of Trojans as well [15]. They proposed to use conditional stuck-at fault (*CSP-n*) patterns over the entire circuit to trigger all possible *Type-n* combinational Trojans. However, with the higher-order of *n*, the complexity and the number of Trojans increase exponentially. As a result, the authors restrict test generation for CSP-1 and evaluate their higher type of Trojans coverage. For CSP at any given net, any of the two stuck-at faults (*saf*), *i.e.,* either *sa0* or *sa1*, is detected with a fixed logic value or condition on one of the remaining nets in the circuit. For the same fault, the process is iterated by moving the condition over all the other nets in the circuit to generate a set of CSP for a specific net. These steps are followed for every net in the circuit to generate the complete set of CSP-1 in the entire circuit. The results presented on ISCAS'85 benchmark circuits demonstrated this approach's efficiency over N-detects and random test pattern generation in terms of detection.

Cruz et al. proposed to use automatic test pattern generation along with the model checking tool to increase the efficiency of the test set in partial-scan designs [36]. Researchers have also leveraged the advancement in machine learning and neural network techniques for generating test sets for logic testing. Salmani et al. developed a detection technique based on the observability and controllability values of the nets in the gate-level netlists [44]. The approach uses the distance between clusters, based on the SCOAP values of nets, to conduct unsupervised cluster analysis and feature classification. Similar approaches have also been demonstrated in [45], [46], which also relies on controllability and observability values of the nodes in the original circuit along with genetic algorithms to introduce a fitness function or perform clustering, respectively. Additionally, these methods can only judge whether there are Trojans to a certain extent. However, for large designs, it is difficult to generate test vectors for triggering a large number of Trojan choices available to the adversary [8], [15]. Moreover, sequential and analog Trojans cannot be detected by such techniques due to the difficulty of activating the Trojan.

B. Sequential Trojan Detection

The stealthiness of a Trojan lies within its capability to remain quiet under normal operation. The stealthiness of a sequential Trojan is generally higher compared to a combinational Trojan. Typically, all chips include design-for-test (DFT) architecture (*i.e.,* scan-architecture) to increase the testability [47]. The flip-flops (FFs) in a design are converted to scan FFs to convert a sequential circuit to a combinational one for test pattern generation to increase fault coverage. An adversary will not convert the FFs for triggering a sequential Trojan primarily for two reasons - (i) the sequential Trojan will become a combinational one, and (ii) the circuit will fail in the manufacturing test as the Trojan FFs are not a part of the original design. The identification of these Trojan FFs will eventually result in the detection of a sequential Trojan. This concept is represented in Figure 4, where the state elements (*i.e.,* DFFs) of a design are stitched together to form a scan-chain, while the Trojan FFs are excluded from the same. Stern et al. exploited this notion to detect Trojans using a non-destructive backside laser probing approach [48]. This approach relies on finding the location of sequential elements (FFs) for a hardware Trojan. Two different sets of Electro-Optical Frequency Mapping (EOFM) images are obtained and compared to identify the Trojan FFs.

The concept of EOFM relies on the ability of silicon to remain transparent to infrared wavelength, and therefore, the infrared laser incident on the backside of an IC passes the substrate. However, these rays get reflected from the active metal layers and provide the frequency information of the current passing through the cell, which can be analyzed under a spectrum analyzer to construct the mapping image. First, an EOFM image is obtained while running the chip at the functional clock frequency (f_{clk}), which reveals a map of all the FFs within the chip authentic and malicious. Next, the EOFM image is constructed by putting the IC in scan mode and shifting-in an altering sequence of 1's and 0's (*i.e.,* '1010...'), referred to as f_{scan_in}, to identify authentic FFs.

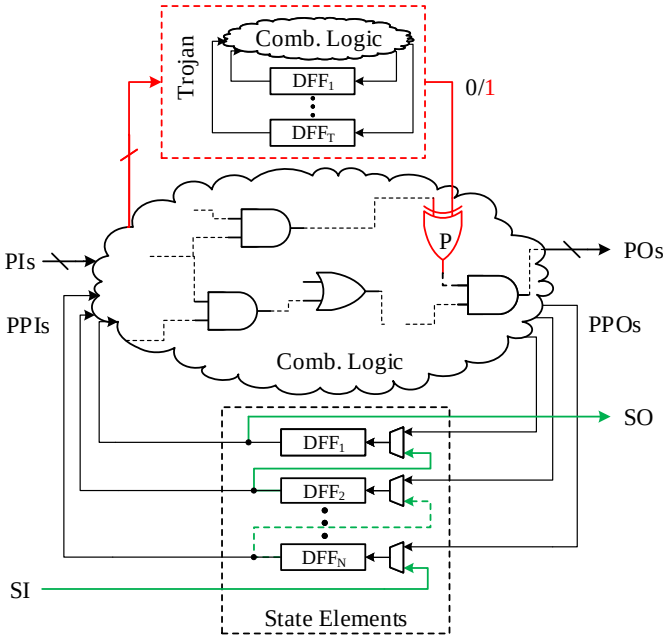

Figure 4. An abstract representation of a design tampered with a sequential Trojan (red).

Upon comparing the two sets of EOFM images, the locations of the malicious FFs are determined. Figure 5 shows the processed EOFM images from the experimentation conducted in [48] on Trust-hub benchmark circuits. The red boxes in Figure 5.(a) marks the location of all the suspected FFs, obtained under f_{clk}. After the analysis for scan input pattern (f_{scan_in}) as mentioned above, the same area helps to identify all the suspected flip-flops. The green boxes in Figure 5 show the authentic FFs, which are part of the scan chain, whereas the red ones represent the Trojan FFs. This non-destructive method does not require any prior knowledge regarding chip functionality. However, it requires some pre-processing of the map images before comparison to remove the impact of noise, which is taken care of through image processing techniques.

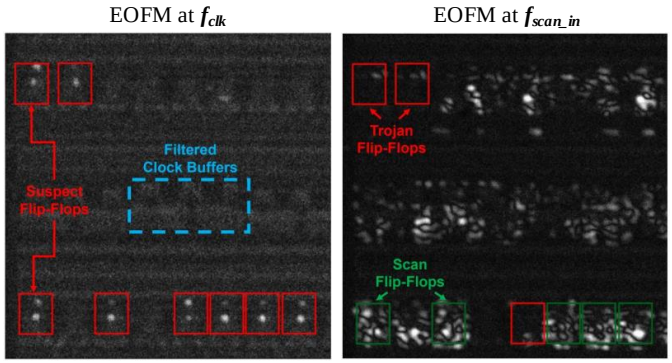

Figure 5. Processed EOFM images of the Trust-Hub Trojan benchmark with: (a) Suspect flip-flop identification. (b) Trojan flip-flop detection [48].

The other Trojan detection method includes side-channel information analysis, such as power [20], temperature [21], delay [22] and radiation [24] based techniques, which rely on the availability of Trojan-free golden circuits or simulation data. Hossain et al. [20] introduced a power-based side-channel analysis to detect the Trojan presence in the circuit. The authors divided the circuit into segments to increase the Trojan-to-circuit power consumption under the three specific methods, *i.e.,* scan segmentation methodology, Equal-Power Self-referencing (EP), and Equal-Power Neighbouring self-referencing (EPN), and it tries to improve detection sensitivity on circuits with many process variations. Temperature based thermal imaging techniques have also shown improvement towards Trojan detection [21], [23]. Tang et al. proposed the use of quiescent thermal maps and its active area shape from the GDSII file [23]. This method has been shown effective and independent of the golden circuit and process variations.

Recent research contributions showed that machine learning could also be incorporated with optical inspection techniques to detect hardware Trojans in the chip. Vashistha et al.. presented the Trojan scanner, which uses a trusted GDSII layout (golden layout) and scanning electron microscope (SEM) images to identify the malicious modifications made in the netlist during the manufacturing of a circuit [49]. A unique descriptor for each type of gate is prepared based on different features using computer vision algorithms and a machine-learning model of a golden layout and SEM images of an IC under authentication. When compared to each other, these descriptors can detect any modifications either in the form of additional gates or modified gates, which might raise suspicion for a potential hardware Trojan. A similar imaging-based technique combined with electrical testing has also been demonstrated to detect Trojans [50]. The authors proposed inserting golden gate circuits (GGC), a combination of logic gates and test infrastructure, in the unused space of the design. The GGC is first authenticated with logic tests and it is used to assist in the accuracy of the machine learning classifier for detecting any suspicious modification under backside imaging.

IV. Conclusion

Detection and avoidance of hardware Trojans have gained considerable attention over the last decade. The research community has made significant improvements and contributions in this direction. However, due to the vast number of possible Trojans and their small footprint, we still lack efficient and accurate methods for detecting combinational and analog Trojans. Logic test based detection techniques fail to trigger or detect these Trojans. However, optical imaging-based techniques are establishing as the new and compelling direction toward detecting sequential hardware Trojans.

Acknowledgment

This work was supported in parts by the National Science Foundation (NSF) under grant CNS-1755733 and Air Force Research Laboratory (AFRL) under grant AF-FA8650-19-1-1707. Any opinions, findings, and conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF and AFRL.

REFERENCES

[1] Age Yeh, "Trends in the global IC design service market," DIGITIMES Research, 2012.

[2] S. Adee, "The hunt for the kill switch," *IEEE SpEctrum*, vol. 45, no. 5, pp. 34–39, 2008.

[3] S. Bhunia, M. S. Hsiao, M. Banga, and S. Narasimhan, "Hardware Trojan attacks: Threat analysis and countermeasures," *Proceedings of the IEEE*, pp. 1229–1247, 2014.

[4] R. Karri, J. Rajendran, K. Rosenfeld, and M. Tehranipoor, "Trustworthy hardware: Identifying and classifying hardware trojans," *Computer*, pp. 39–46, 2010.

[5] K. Xiao, D. Forte, Y. Jin, R. Karri, S. Bhunia, and M. Tehranipoor, "Hardware trojans: Lessons learned after one decade of research," *Trans. on Design Automation of Electronic Systems (TODAES)*, pp. 1–23, 2016.

[6] A. Jain and U. Guin, "A Novel Tampering Attack on AES Cores with Hardware Trojans," in *Int. Test Conf. Asia (ITC-Asia)*, 2020, pp. 77–82.

[7] S. Bhasin, J.-L. Danger, S. Guilley, X. T. Ngo, and L. Sauvage, "Hardware Trojan horses in cryptographic IP cores," in *IEEE Workshop on Fault Diagnosis and Tolerance in Cryptography*, 2013, pp. 15–29.

[8] A. Jain, Z. Zhou, and U. Guin, "TAAL: Tampering Attack on Any Key-based Logic Locked Circuits," *arXiv preprint arXiv:1909.07426*, 2019.

[9] S. Bhunia and M. Tehranipoor, *The Hardware Trojan War*. Springer, 2018.

[10] Y. Liu, Y. Jin, A. Nosratinia, and Y. Makris, "Silicon demonstration of hardware Trojan design and detection in wireless cryptographic ICs," *Trans. on Very Large Scale Integration (VLSI) Systems*, 2016.

[11] M. Tehranipoor and C. Wang, *Introduction to hardware security and trust*. Springer Science & Business Media, 2011.

[12] J. Cruz, F. Farahmandi, A. Ahmed, and P. Mishra, "Hardware Trojan detection using ATPG and model checking," in *Int. conf. on VLSI design and Int. conf. on embedded systems*, 2018, pp. 91–96.

[13] M. Fyrbiak, S. Wallat, P. Swierczynski, M. Hoffmann, S. Hoppach, M. Wilhelm, T. Weidlich, R. Tessier, and C. Paar, "HAL—The missing piece of the puzzle for hardware reverse engineering, Trojan detection and insertion," *Trans. on Dependable and Secure Computing*, 2018.

[14] S. K. Haider, C. Jin, M. Ahmad, D. Shila, O. Khan, and M. van Dijk, "Advancing the State-of-the-Art in Hardware Trojans Detection," *IEEE Transactions on Dependable and Secure Computing*, 2017.

[15] Z. Zhou, U. Guin, and V. D. Agrawal, "Modeling and test generation for combinational hardware Trojans," in *VLSI Test Symposium*, 2018.

[16] F. Farahmandi, Y. Huang, and P. Mishra, "Trojan localization using symbolic algebra," in *Asia and S. Pacific Design Automation Conf.*, 2017.

[17] Y. Huang, S. Bhunia, and P. Mishra, "MERS: statistical test generation for side-channel analysis based Trojan detection," in *Proc. of ACM SIGSAC Conf. on Computer and Communications Security*, 2016.

[18] ——, "Scalable test generation for Trojan detection using side channel analysis," *Trans. on Information Forensics and Security*, 2018.

[19] S. Yu, C. Gu, W. Liu, and M. O'Neill, "A Novel Feature Extraction Strategy for Hardware Trojan Detection," in *Int. Symposium on Circuits and Systems (ISCAS)*, 2020, pp. 1–5.

[20] F. S. Hossain, M. Shintani, M. Inoue, and A. Orailoglu, "Variation-aware hardware Trojan detection through power side-channel," in *Int. Test Conf. (ITC)*, 2018, pp. 1–10.

[21] J. Zhong and J. Wang, "Thermal images based Hardware Trojan detection through differential temperature matrix," *Optik*, pp. 855–860, 2018.

[22] X. Cui, E. Koopahi, K. Wu, and R. Karri, "Hardware Trojan detection using the order of path delay," *Journal on Emerging Technologies in Computing Systems (JETC)*, pp. 1–23, 2018.

[23] Y. Tang, S. Li, L. Fang, X. Hu, and J. Chen, "Golden-chip-free hardware trojan detection through quiescent thermal maps," *Trans on Very Large Scale Integration (VLSI) Systems*, pp. 2872–2883, 2019.

[24] J. He, Y. Zhao, X. Guo, and Y. Jin, "Hardware Trojan Detection Through Chip-Free Electromagnetic Side-Channel Statistical Analysis," *Trans. Very Large Scale Integration Sys.*, pp. 2939–2948, 2017.

[25] L. N. Nguyen, C.-L. Cheng, M. Prvulovic, and A. Zajić, "Creating a backscattering side channel to enable detection of dormant hardware trojans," *Trans. on very large scale integration (VLSI) sys.*, 2019.

[26] T. Hoque, S. Narasimhan, X. Wang, S. Mal-Sarkar, and S. Bhunia, "Golden-free hardware Trojan detection with high sensitivity under process noise," *Journal of Electronic Testing*, pp. 107–124, 2017.

[27] H. Xue and S. Ren, "Self-reference-based hardware Trojan detection," *Trans. on Semiconductor Manufacturing*, pp. 2–11, 2017.

[28] Y. Lyu and P. Mishra, "Efficient test generation for Trojan detection using side channel analysis," in *Design, Automation & Test in Europe Conf. & Exhibition (DATE)*, 2019, pp. 408–413.

[29] S. Adibelli, P. Juyal, L. N. Nguyen, M. Prvulovic, and A. Zajic, "Near field backscattering based sensing for hardware trojan detection," *Trans. on Antennas and Propagation*, 2020.

[30] H. Salmani, M. Tehranipoor, and J. Plusquellic, "A Novel Technique for Improving Hardware Trojan Detection and Reducing Trojan Activation Time," *Trans. Very Large Scale Integration Sys.*, pp. 112–125, 2012.

[31] Q. Shi, K. Xiao, D. Forte, and M. M. Tehranipoor, "Obfuscated built-in self-authentication," in *Hardware Protection through Obfuscation*, 2017.

[32] K. Vaidyanathan, B. P. Das, and L. Pileggi, "Detecting Reliability Attacks During Split Fabrication Using Test-Only BEOL Stack," in *Proc. of Design Automation Conf.*, 2014, pp. 1–6.

[33] Y. Wang, P. Chen, J. Hu, and J. J. Rajendran, "The Cat and Mouse in Split Manufacturing," in *Proc. of Design Automation Conference*, 2016.

[34] K. Yang, M. Hicks, Q. Dong, T. Austin, and D. Sylvester, "A2: Analog malicious hardware," in *Symposium on security and privacy (SP)*, 2016.

[35] A. Jain, U. Guin, M. T. Rahman, N. Asadizanjani, D. Duvalsaint, and R. S. Blanton, "Special Session: Novel Attacks on Logic-Locking," in *VLSI Test Symposium (VTS)*, 2020, pp. 1–10.

[36] J. Cruz, Y. Huang, P. Mishra, and S. Bhunia, "An automated configurable Trojan insertion framework for dynamic trust benchmarks," in *Design, Automation & Test in Europe Conf. & Exhibition*, 2018, pp. 1598–1603.

[37] B. Shakya, T. He, H. Salmani, D. Forte, S. Bhunia, and M. Tehranipoor, "Benchmarking of hardware trojans and maliciously affected circuits," *Journal of Hardware and Systems Security*, pp. 85–102, 2017.

[38] X. Wang, S. Narasimhan, A. Krishna, T. Mal-Sarkar, and S. Bhunia, "Sequential hardware trojan: Side-channel aware design and placement," in *Int. Conf. on Computer Design (ICCD)*, 2011, pp. 297–300.

[39] S. Ghandali, G. T. Becker, D. Holcomb, and C. Paar, "A design methodology for stealthy parametric trojans and its application to bug attacks," in *Int. Conf. on Cryptographic Hardware and Embedded Systems*, 2016, pp. 625–647.

[40] K. Nagarajan, M. N. I. Khan, and S. Ghosh, "ENTT: A family of emerging NVM-based trojan triggers," in *International Symposium on Hardware Oriented Security and Trust (HOST)*, 2019, pp. 51–60.

[41] C. Kison, O. M. Awad, M. Fyrbiak, and C. Paar, "Security Implications of Intentional Capacitive Crosstalk," *Trans. on Information Forensics and Security*, 2019.

[42] K. S. Subramani, A. Antonopoulos, A. A. Abotabl, A. Nosratinia, and Y. Makris, "Ace: adaptive channel estimation for detecting analog/RF trojans in WLAN transceivers," in *Proc. of Int. Conf. on Computer-Aided Design*, 2017, pp. 722–727.

[43] A. Bazzazi, M. T. M. Shalmani, and A. M. A. Hemmatyar, "Hardware Trojan detection based on logical testing," *Journal of Electronic Testing*, pp. 381–395, 2017.

[44] H. Salmani, "COTD: Reference-free hardware trojan detection and recovery based on controllability and observability in gate-level netlist," *Trans. on Information Forensics and Security*, pp. 338–350, 2016.

[45] M. Nourian, M. Fazeli, and D. Hély, "Hardware Trojan detection using an advised genetic algorithm based logic testing," *Journal of Electronic Testing*, pp. 461–470, 2018.

[46] X. Xie, Y. Sun, H. Chen, and Y. Ding, "Hardware Trojans classification based on controllability and observability in gate-level netlist," *IEICE Electronics Express*, pp. 20 170 682–20 170 682, 2017.

[47] M. Bushnell and V. Agrawal, *Essentials of electronic testing for digital, memory and mixed-signal VLSI circuits*. Springer Science & Business Media, 2004, vol. 17.

[48] A. Stern, D. Mehta, S. Tajik, U. Guin, F. Farahmandi, and M. Tehranipoor, "SPARTA: Laser Probing Approach for Sequential Trojan Detection in COTS Integrated Circuits," *IEEE Int. Conf. on Physical Assurance and Inspection of Electronics (PAINE)*, 2020.

[49] N. Vashistha, H. Lu, Q. Shi, M. T. Rahman, H. Shen, D. L. Woodard, N. Asadizanjani, and M. Tehranipoor, "Trojan scanner: Detecting hardware trojans with rapid SEM imaging combined with image processing and machine learning," in *Proc. of the Int. Symposium for Testing and Failure Analysis (ISTFA)*, 2018, p. 256.

[50] Q. Shi, N. Vashistha, H. Lu, H.-T. Shen, B. Tehranipoor, D. L. Woodard, and N. Asadizanjani, "Golden Gates: A New Hybrid Approach for Rapid Hardware Trojan Detection using Testing and Imaging." in *HOST*, 2019, pp. 61–71.