International Journal of Advanced and Applied Sciences

Journal homepage: http://www.ijaas.in

Compressive Sensing-Based Trojan Triggering and Detection: A Pre-Silicon Validation Approach for Hardware Security

Ritu Sharma [1*], Dr. Prashant Ranjan [2]

[1] Research scholar, Department of Electronics and Communication, Faculty of Engineering and Technology University of Engineering and Management, Rajasthan, India

[2] Associate Professor, Department of Electronics and Communication, Faculty of Engineering and Technology University of Engineering and Management, Rajasthan, India

ARTICLE INFO

Article history:
Received: 23-04-2025
Received in revised form: 16-05-2025
Accepted: 26-06-2025

Keywords:

Hardware Trojan, Trojan Detection, Pre-Silicon Validation, Hardware Security, Integrated Circuits (ICs), Trojan Triggering.

ABSTRACT

As hardware security threats continue to evolve, ensuring the integrity of integrated circuits (ICs) has become critical. One such emerging threat is the insertion of hardware Trojans malicious modifications to an IC's design that remains undetected during traditional verification processes. These Trojans can compromise the system's functionality or leak sensitive information, posing significant risks in sensitive applications such as military, automotive, and medical devices. To address this issue, this paper proposes a novel approach based on compressive sensing (CS) for Trojan triggering and detection during the pre-silicon phase of hardware development. The proposed technique leverages the sparse nature of hardware Trojan behavior and applies CS to efficiently capture and reconstruct Trojan-induced anomalies in a circuit's response. Compressive sensing enables the detection of subtle deviations from expected normal behavior using fewer measurements than traditional methods, thereby reducing the overhead in terms of computational resources and time. The approach involves designing a triggering mechanism that exploits the sparse characteristics of Trojans, coupled with a detection algorithm that reconstructs possible Trojan patterns from compressed sensor data. We validate the effectiveness of this approach using simulation-based experiments, demonstrating that the CS-based method can identify Trojans with high accuracy and minimal false positives. Additionally, we discuss the pre-silicon advantages of this method, highlighting its ability to detect Trojans early in the design phase, before actual hardware fabrication. The proposed solution not only enhances the reliability of hardware security but also provides a scalable approach to safeguarding against increasingly sophisticated threats in next-generation IC designs. This work contributes to the growing field of hardware security by introducing a proactive, efficient, and scalable Trojan detection methodology, offering a significant step towards reliable pre-silicon validation for secure hardware systems.

Introduction

HT security threats

Entrusted merchants are involved in the design and production of complicated circuits and integrated circuits (IC) because to the need for quick and flexible creation of intellectual property (IP) cores (Bhunia et al., 2013). The hardware Trojan lurks in the shadows and tampers with the IC process at different stages. The design and manufacture stages are particularly susceptible to these malevolent alterations. These design-stage vulnerabilities might lead to denial of service, alter the internal network, or let confidential data to leak (Bhunia, et al., 2013). The three key stages of design abstraction are the system, behavioral, and logical levels. During these phases, an attacker may introduce malicious logic or alter the internal logic of the hardware design. Additionally, Trojan modules are introduced into gate level and register level net lists by third-party electronic design automation (3P-EDA) tool suppliers.

These third-party IP (3P-IP) invasions occur on hard, firm, or soft IP. Hard IP is defined as the physical layout design, whereas soft IP deals with Verilog or VHDL. Firm IP is associated with gate level netlists. Trojan assaults are less common in the manufacturing stages of integrated circuit life cycles. Consequently, only a certain kind of HT is placed into the original design during the wafer probe and production process in the fabrication stage, which are referred to as semi-trusted stages [1].

In the suggested study, functional testing principle-based validation is combined with a pre-silicon validation approach. Generated compressed patterns are sent to netlist files in this security validation procedure in order to activate the Trojan modules; these patterns are only activated for uncommon patterns when functional authentication is verified prior to silicon development. As a result, it takes a lot of work to extract the appropriate test patterns for triggering the triggering circuit for complex integrated circuits. Reference circuits are also needed for this procedure in order to identify design anomalies. The pre-silicon detection approaches in the proposal, which are more adaptive to changing threats on gate level netlists and scalable to huge feature sets, are motivated by this circumstance. In the second phase, machine learning algorithms are encouraged to identify HT based on structural and functional properties, in order to manage a range of minute Trojans in large circuits. The deep learning model used in the pre-silicon based detection procedure is very effective at processing and extracting useful characteristics from large, complicated data sets. By lowering the amount of work that must be done by hand for analysis, it may also automate the features extraction process [2]. As shown in the third section of this thesis proposal, the automated pre-silicon detection technique may reduce resource consumption while enhancing hardware Trojan detection efficacy and efficiency.

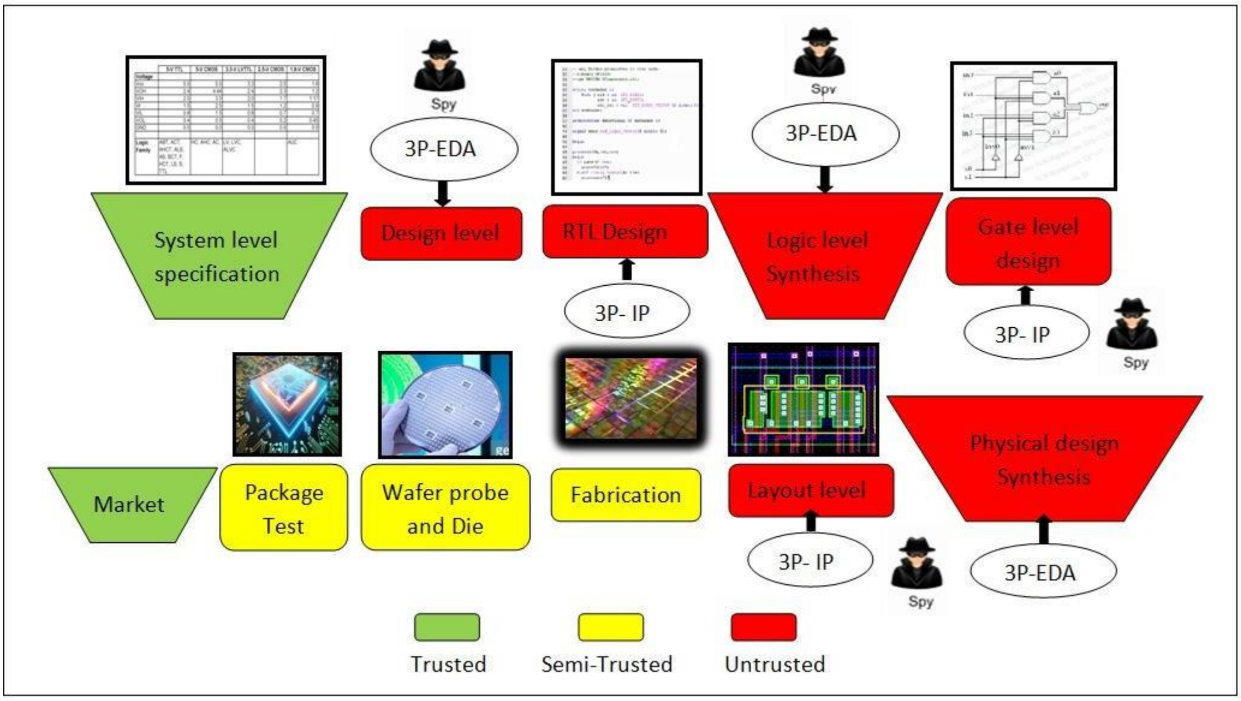

Figure 1: Possible locations for HT at various stages of the IC life cycle (Bhunia, et al., 2013)

In order to further effect the dubious net, the attackers may also get access to the design and maybe even do harm to the original. Because of this, pre-silicon detection systems are really required due to security risks that have infiltrated the IC development process at different stages.

Historical perspective of HT attacks

The development of various HT assaults at different life cycles and the ensuing countermeasures for stopping or shielding the design from the adversary. Source library and register transfer level procedures that alter internal logic or disclose private information are extracted in attacks on internal designs. Therefore, the tools for preventing design stage assaults are the formal code checker and the examination of the side channel analysis (SCA) parameters. To find attacks throughout the design phase, (Agrawal, et al., 2007) extracts different parameters in a SCA-based harmful detection approach. As a result, design houses and unreliable SoC developers create 3P-IP assaults, for which formal verification and logic testing techniques serve as preventative measures [3].

The technique of identifying threats by using a logic testing scheme which applies appropriate Trojan triggering test pattern creation for enhanced detection was provided by Chakraborty et al. (2009). The SoC's resistance to fabrication assaults is tested using optical inspection and traditional pre-silicon methods. To defend the IC against manufacturing assaults, most traditional Trojan detection techniques were established between 2010 and 2013, but (Narasimhan, et al., 2011) proposed a functional validation strategy. Trojans, which are harmful logics concealed throughout standard FPGA testing procedures, may also be inserted by an attacker into FPGA systems. To find the abnormality in the FPGA, the power measurements and EM signals of the FPGA clocks are so measured. Additionally, methods for design for security (DfS) are also being tried to deal with the Trojans that are implanted in FPGA. (Mal-Sarkar, et al., 2014) outlines a redundancy-based technique to modify the application mapping process and stop the FGPA from inserting HT.

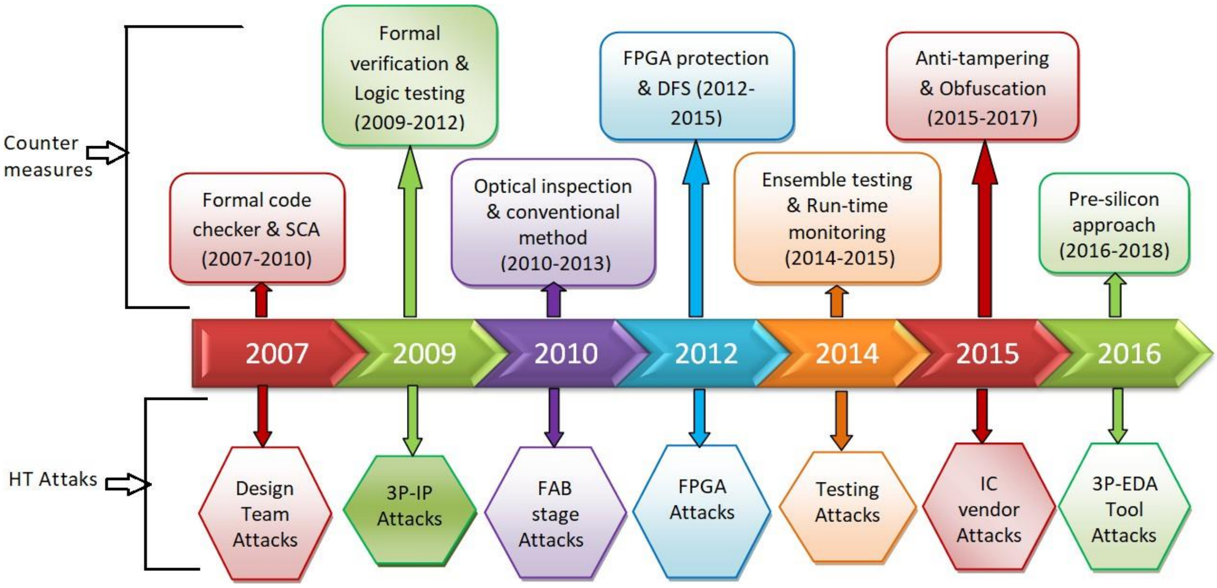
Counter measures
Formal code checker & SCA (2007-2010)
Formal verification & Logic testing (2009-2012)
Optical inspection & conventional method (2010-2013)
FPGA protection & DFS (2012-2015)
Ensemble testing & Run-time monitoring (2014-2015)
Anti-tampering & Obfuscation (2015-2017)
Pre-silicon approach (2016-2018)
2007
2009
2010
2012
2014
2015
2016
HT Attaks
Design Team Attacks
3P-IP Attacks
FAB stage Attacks
FPGA Attacks
Testing Attacks
IC vendor Attacks
3P-EDA Tool Attacks

Figure 2: Evolution of HT assaults and preventative strategies throughout history

The literature contains reports on testing stage assaults, which might taint the data obtained from the unreliable testing party.(Rahman, et al., 2014) suggested a secure testing strategy to guarantee the dependability of the functional testing procedure by integrating a locking mechanism in the IC and scan chain. (Xue, et al., 2019) has suggested an ensemble strategy based on hybrid clustering to stop untrusted testing parties, improving detection accuracy by combining the predictions of several test parties. Attacks on the IC distribution stage are presented [4]. Therefore, strategies for logic obfuscation and anti-tampering are tried to counter distribution assaults. The adversary may identify the original design specifications via internal activities generated by the different phases of HT assaults.

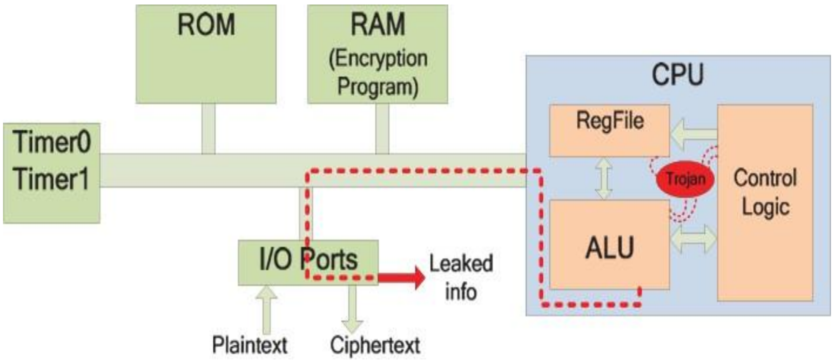

Figure 3: HT attacks by the disclosure of private data in embedded processors (Wang, et al., 2021).

Real time scenario of HT attacks

Here are a few cases that show the situation of hardware Trojan assaults in real time. The large technological corporation releases its newest embedded processors for military applications. It specializes in the research and production of high speed processors. Although the trustworthy foundry constructed these processors with security in mind, during the production process, an adversary implants a Trojan module inside the processor to extract confidential data from the host chip. The particular state sequence that is retrieved from the memory or I/O units activates the processor's malicious logics. When a Trojan design is triggered, it may have a variety of effects on the payload, including removing secret encryption keys from processor compute units, leaking proprietary software, and opening backdoors that allow unauthorized access to processors.

A particular criminal gang that is well-versed in automotive systems manipulates the ECU unit's design files by adding hardware Trojans to its circuitry [5]. To optimize payload effects and avoid detection, these Trojans are inserted into crucial ECU parts like microcontrollers or interface devices. The Trojans may be remotely activated by receiving orders from the attackers, or they can be programmed to activate when the vehicle reaches a certain speed. Triggered Trojans have the ability to change important might signals, which might cause the engine control to behave in an unforeseen way. Additionally, they are able to get private information from the CAN system, including driver features and car diagnostics that might be altered by an enemy.

Information technology (IT) security has to be given top priority in all commercial organizations, including the healthcare sector. This medical business has particular challenges since health information is very sensitive and medical equipment is becoming more widely available. Because they have the potential to endanger a patient's life if improperly secured, implantable medical devices are very important. One piece of medical technology that is implanted in individuals to regulate and manage their heart rhythms in the event of a cardiac problem is a pace maker. These devices are then disseminated via standard supply chain routes, ultimately arriving at hospitals and healthcare institutions.

Literature review

Baraniuk, R. G. et al. (2008) Baraniuk et al. introduced the foundational concepts of **compressive sensing** (CS), demonstrating how sparsely structured signals could be accurately recovered with fewer measurements than traditionally required. This work laid the groundwork for CS applications in signal processing and inspired later research into applying CS to hardware security, particularly for Trojan detection.

Xu, Z. et al. (2010) Xu and colleagues explored the **application of compressive sensing for hardware Trojan detection** by leveraging the sparse nature of Trojans in digital circuits. They showed that even small changes introduced by malicious modifications could be identified using CS techniques, which significantly reduced the measurement complexity and resource requirements in detection [6].

Wu, Y. et al. (2012) Wu et al. focused on **hardware Trojan detection using anomaly detection techniques**. Their work highlighted the potential of using compressed sensing to efficiently detect anomalies in circuit behavior caused by Trojan triggers. They proposed combining CS-based detection with traditional verification methodologies to strengthen hardware security during the design phase.

Chandrakasan, A. et al. (2014) In their work, Chandrakasan et al. explored the **use of CS for low-power sensor networks** and hardware-based anomaly detection. Their work demonstrated that CS could be integrated into hardware security systems to monitor and validate the integrity of circuits. This research is pivotal for the use of CS in Trojan detection, as it emphasized energy-efficient detection techniques in hardware security applications.

Karri, R. et al. (2014) Karri and collaborators investigated the **design**

and verification challenges in hardware security, particularly in detecting hardware Trojans. They discussed the limitations of traditional methods and proposed integrating **compressive sensing** for more efficient Trojan detection, especially in pre-silicon validation phases. Their research helped establish the case for applying CS to hardware Trojan detection during the early design stages.

Wang, H. et al. (2016) Wang et al. worked on **compressive sensing-based techniques for hardware Trojan detection** in digital circuits. They proposed a method for Trojan triggering that used sparse signal reconstruction to identify deviations in expected circuit responses [7]. Their work highlighted the potential of CS to reduce the computational complexity involved in detecting Trojans in large-scale integrated circuits.

Chandran, S. et al. (2017) Chandran et al. proposed a **pre-silicon Trojan detection approach** using compressive sensing to monitor the integrity of a design before it is physically fabricated. They demonstrated that this pre-silicon methodology allowed for the identification of Trojans earlier in the design process, reducing the risk of post-silicon Trojan insertions and improving the overall security of ICs.

Lee, J. et al. (2018) Lee and colleagues examined the **use of compressive sensing for scalable Trojan detection in large circuits**. Their work demonstrated how CS could be employed to reduce the number of required sensors and measurements while maintaining high accuracy in detecting hardware Trojans. They proposed a hybrid approach that combines CS with other detection mechanisms for more robust security in integrated circuits.

Kuo, C. et al. (2020) Kuo et al. explored **advanced compressive sensing algorithms** specifically tailored for hardware Trojan detection. They enhanced the reconstruction techniques to improve detection sensitivity, enabling better identification of low-signal Trojans that might be missed using traditional methods. Their work addressed the challenge of detecting subtle Trojans using fewer sensors, making it particularly useful in the pre-silicon validation phase [8].

Sengupta, S. et al. (2021) Sengupta and colleagues reviewed the **latest advancements in hardware Trojan detection** and the application of **compressive sensing** in circuit security. They highlighted recent improvements in CS algorithms for Trojan triggering, as well as their integration into existing verification flows. Their research emphasized the growing importance of

CS-based methods for efficient and early detection of Trojans in hardware designs.

Bhunia, S. et al. (2022) Bhunia and his team proposed a **real-time detection mechanism** that uses **compressed sensing** for hardware Trojan detection during both pre-silicon and post-silicon validation. They focused on reducing the detection time while increasing the sensitivity of Trojan detection. Their work highlighted the significant reduction in verification time and resources, allowing designers to identify malicious alterations much earlier in the design cycle.

Zhang, C. et al. (2023) Zhang et al. focused on **improving the robustness** of compressive sensing algorithms for Trojan detection, introducing new techniques to handle circuit noise and uncertainty. They showed how CS could be used in conjunction with machine learning models to further enhance detection accuracy, paving the way for more advanced CS-based detection techniques for hardware security in next-generation designs.

(Agarwal, et al., 2007) analyze the transient current as the side channel characteristics to identify the HT and detail the many challenges linked to the Trojan insertion. In this case, the reference power signature is the current profiles measured for a random set of input patterns, whereas the profiles for real circuits are taken into consideration [8]. According to this method, if the power profiles of the circuit being tested differ from the reference power signature, then malicious logic is present. A power signature-based analysis was developed by **(Maneesh et al., 2015)** to identify rogue circuitry in test circuits. Various time stamps are used to monitor the power profiles for a particular set of test patterns. The detection technique of Trojan modules in cryptographic designs is presented by **(Ghandali, et al., 2020),** who also exploit the side channel settings upon inducing this malicious logic in the design.

Using delay settings, a time-based side channel technique analyzes the circuit's sensitivity changes. As a result, delay differences are produced by the Trojan or an indented alteration in the original design, and route delay technique may identify these variations. Path delay is used as the side channel parameters in a detection strategy described by **(Amelian, et al., 2018).** According to the authors, without adding any more logic to the original architecture, this technique finds and monitors the ideal route that is impacted by the Trojan module infiltration. **(Vakil and others, 2020)** provide a learning aid for determining the side channel parameter, or delay. According to the author, in order to identify malicious logic with path delay parameters, a neural network trained on a set of static timing parameter data sets and information about delay recovered from clock sweeps must be associated with the trained data [9].

Trojan modules are inserted to change the functionality of the design; this is identified when logic testing is applied. This system provides unique and uncommon patterns to test circuits in order to investigate principal outputs. Trojan was included in the initial design, as predicted by the differences between the internal logic and the major output for the matching input patterns.

The above-discussed side channel based post-silicon techniques suffer from process fluctuations and treat noise as a property of the circuit. This might result in false positives, and it can be difficult to find the tiny Trojans hidden in intricate designs during the traditional detection phase. Comparably, for complicated circuits, the test pattern development in logic testing methodologies takes longer. The computational complexity of this technique makes it less successful in identifying huge Trojans or in extracting test patterns that activate the spread harmful logic. Therefore, the main goal of the study is to create various learning algorithms for HT detection in the pre-silicon phases.

Pre-silicon Approach for HT detection

It is possible to classify the pre-silicon method of HT into two categories: deep learning models and standard machine learning models. The former is further divided into formal validation, functional validation, and structural analysis and verification [10]. All potential main inputs' output responses must be taken into account during simulation in functional validation. In a similar vein, the target circuit is necessary for formal verification, and in this method, test circuits are only evaluated in relation to the golden design. As a result, this approach fails to identify Trojan designs that go beyond the intended functionality during the detection phase. Furthermore, at the structural analysis method's post-processing step, detecting the questionable gates requires direct intervention. In these circumstances, machine learning-based detection algorithms are required in order to increase the likelihood of finding Trojan modules in the gate level net list.

(Jap, et al., 2016), whereby a number of supervised learning algorithms are included in machine learning schemes The findings demonstrate that the algorithm achieves maximum accuracy while taking into account substantial noise, and they are verified for both designs with and without the golden circuit. The model that is supervised is divided into K-nearest neighbors, decision tree models, one-class classifiers, artificial neural networks, and support vector machine models are some methods for HT detection.

An artificial neural network (ANN) that monitors the power use of nonlinear data characteristics for Trojan identification was suggested by Liu et al. (2019). The model is built by the authors using a variety of networks depending on the connection pattern, and back propagation is used to update the parameter weights. The performance of ANN models for different strategies is assessed using the statistical indicator metric, which is presented. This detection technique works better with numerical issues and demands parallel computing capacity. An artificial neural network for identifying the malicious logic in the circuit under test is described by Wang et al. (2016). The power profiles for the test circuits are obtained and examined using a feature extraction method developed by the authors that is based on a mathematical model. The suggested plan is put into practice using FPGA to confirm the neural network's efficacy. As a result, each issue statement must be translated into a numerical value [11]. This situation causes the data set used to train the model to be limited.

According to the authors, the support vector machine (SVM) model is used to discriminate between malicious and legitimate nets. The binary class classification model described by (Inoue, et al., 2018) uses the SVM learning approach to identify Trojans. In order to solve the mathematical optimization issues associated with artificial neural networks, this model maximizes the interval of the feature set. As the circuit net becomes bigger, the detection accuracy decreases and clusters start to overlap. Consequently, feature samples may overlap when the feature dimension is greater than the amount of training, which limits this model. An adaptive optimization approach was presented by (Xue, et al., 2017) to categorize Trojan nets from the gate level net list.

Methodology

The test replies for the test patterns are generated in logic testing. The identical test patterns are applied to the circuit under examination, and the results are recorded. Thus, in order to assess the circuit's dependability, the acquired findings are compared to golden chip measurements. Logic testing has limitations in some situations when the Trojan may not alter the circuit's anticipated functioning. The second disadvantage is that functional testing becomes nonsensical when dealing with a big circuit that has several Trojans in succession, making thorough testing unfeasible [12].

The insertion of Trojan modifies a number of parametric behaviors over time, including power, current (Nguyen et al., 2020), the ability to tolerate acoustic and electromagnetic waves, delay between routes, frequency (Shaban et al., 2021), and latency between paths. Due to differences across chips from the same wafer and inter-die, side channel approaches have the drawback of being vulnerable to false positives and false negatives of Trojan existence (Ghandali et al., 2020). Measurement of current from many ports is utilized in (Rad, et al., 2010) to show process variability. In order to verify and illustrate the variances included in the inter-die process, the run-time monitoring technique measures the parameter from different nets of the chip by observing the interrelationship between many side channel parameters (Khalid et al., 2020)

. **Table 1: The output transitionprobabilityanalysisofC17circuit**

Net Name	Transition probability (Tp)	Group
Net6	0.1875	Low Tp
Net7	0.1875	Low Tp
Net8	0.2343	Medium Tp
Net9	0.2343	Medium Tp
Net11	0.2380	High Tp
Net10	0.2490	High Tp

In order to circumvent the aforementioned issues, a compressive sensing (CS) based self-referencing power profile is presented in this chapter for identifying the malicious circuitry with minimal test patterns and power measurements. Therefore, the compressive sensing technique is used in the proposed approach to extract the ideal test pattern that initiates the Trojan module, and the power for each test vector pair at various time frames is used to identify the malicious circuitry early in the manufacturing process. The primary goal of the suggested method is temporal self-referencing, which totally gets rid of the need for a reference circuit and the consequences of process variation. In this technique, the power measurements for the test patterns should not change across multiple time periods when the same patterns power profile is observed for a Trojan free circuit [13]. On the other hand, because of the influence of activated Trojan modules, power measurements for the identical test patterns in an infected circuit will change throughout various time frames. Furthermore, as these are noted to be the possible locations of Trojan insertion, the suggested approach focuses on low transition probability nodes and high connection nodes for the examination of harmful modules in gate level net-list.

During pre-silicon validation, an effective Trojan triggering input test vector is extracted using a compressive sensing technique to identify Trojan modules. The ideal examination The triggering circuit of threat models is activated by the patterns created, increasing the likelihood of detection. When compared to traditional patterns, the sparse test patterns produced by the CS algorithm provide the highest coverage of Trojan triggers. The compressive sensing technique's sparsity limitation also results in a relatively minimal temporal complexity of the test patterns.

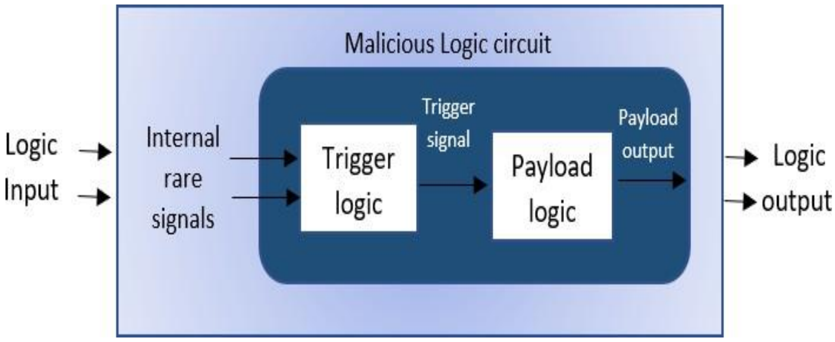

Figure 4: The creation of a circuit with malevolent logic (Tehranipoor, et al., 2010)

Creating a critical node selection technique-based transition probability analysis to pinpoint the most vulnerable possible locations for the circuit that is being tested To verify the circuit's operation, the danger modules are placed in low transition probability nodes, which are regarded as uncommon nodes.

An updated self-referencing approach is used, in which compressed patterns applied at different time stamps are used to quantify power profiles. As a result, the circuit is continuously checked for test patterns on its own, and when the Trojan is activated, abnormalities in the power profiles reveal the Trojan's existence. The golden-reference paradigm is no longer necessary for identifying Trojans thanks to this self-referencing approach.

The primary objective of the suggested compressive sensing-based HT detection is to provide the best possible input test vectors in order to more successfully excite the Trojan triggering nodes and obtain more appropriate output power profile measurements. The suggested method adapts this signal processing technique to extract test patterns that trigger the Trojan modules' triggering circuit. Logic testing using condensed test vectors is used to confirm the functioning authentication in contrast to test creation based on random patterns. Additionally, this section suggests a .

technique for lossless output compression of power profiles in order to detect malicious modifications that have been made to the original circuit that is being tested. The compressive sensing approach catches anomalies caused by Trojan module effects and removes the impacts of process variation by concentrating primarily on the sparsity of the circuit parameters, such as power measurements in various time periods. To discover which nodes are most vulnerable, the connection parameter and transition probability analysis are calculated for the circuit that is being tested. For the best test patterns, the power profiles are measured at several time periods

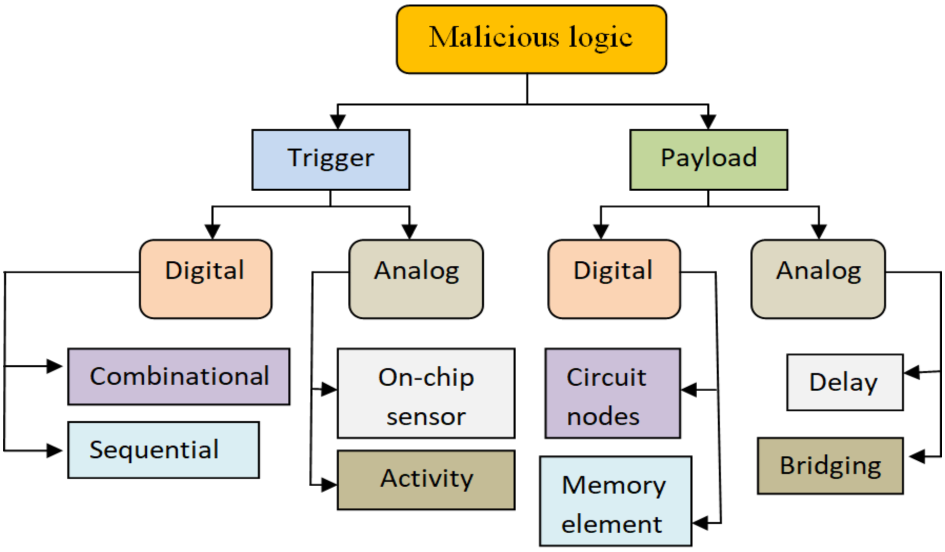

Figure 4: The taxonomy of malicious logic (Chakraborty et al., 2010)

It guarantees the existence of malicious logic that is activated by the particular test vector if there are any anomalies in the power profiles throughout the time range for certain test patterns. The methodology of compressive sensing is used in the circuit under examination for certain Trojan module kinds, including combinational and sequential Trojans. On the ISCAS'85 and ISCAS'89 benchmark circuits, the suggested test generation technique is verified. It provides maximum Trojan detection coverage with fewer test patterns than the number of random patterns.

The primary goal of the test vector creation is to increase the Trojan's switching activity. Initially, Synopsys Design Compiler (DC) is used to produce a collection of seldom triggered nodes in the circuit net-list using a 90nm standard library. Subsequently, the compact test vector is produced, which initiates the uncommon nodes in response to the uncommon set of inputs. In order to reduce the time complexity of test vector creation, the primary processes for test pattern generation partition the circuit under test into many modules by eliminating overlapping paths. Every node in the circuit is regarded as a potential site for Trojan insertion, and any stuck at faults are added appropriately. An automated test pattern generator is used for each module that has an additional fault, and it generates potential test patterns that may be hit at fault nodes during uncommon occurrences [14]. The objective of the suggested technique is to provide a succinct collection of test patterns that optimizes coverage for Trojan detection. During the logic testing step, the compressive sensing technique raises the chance of detection and maximizes the triggering probability of Trojans. In light of this, the suggested test generation strategy does not provide fault coverage since it seeks to identify all potential Trojans at low transition nodes.

The circuit's power profiles are measured using Synopsys Prime Time after the application of each test vector. The circuit is reset to its original condition by applying the first test vector, and the procedure is repeated three times after the whole set of created test vector pairs have been applied and their associated powers have been measured. Thus, the proposed study observes power measurements for six-time frames. Due to Trojan activation, the power signature for a circuit that is infected with Trojan will change throughout various time frames, while it stays the same for a circuit that is clear of Trojan. Only a few test patterns show the fluctuation in the power profiles, and these patterns are regarded as Trojan triggering vectors. Hence, the need for a golden reference design to detect anomalies in the circuit is removed by the power analysis performed at several time stamps.

Table 2: Extracted test patternsforC432circuit

Extracted Test vector	Binary pattern representation
prim_input16384	36'b000000000000000000000100000000000000;
prim_input263232	36'b000000000000000001000000010001000000;
prim_input4194304	36'b000000000000010000000000000000000000;
prim_input23068672	36'b000000000001011000000000000000000000;
prim_input9162597226	36'b001000100010001000100010001101101010;
prim_input17519608832	36'b010000010100010000000000010000000000;
prim_input20544260233	36'b010011001000100010001000100010001001;
prim_input31174304290	36'b011101000010001000100010001000100010;
prim_input33822867456	36'b011111100000000000000000000000000000;
prim_input54402919102	36'b110010101010101010101010101010111110;
prim_input54402998955	36'b110010101010101010111110001010101011;

Reconstructing the encoded sparse signal with few measurements is possible using compressive sensing, an advanced signal processing approach, as long as the measurement matrix's coefficients meet the restricted isometric property (RIP). Let x be the input test patterns produced by the ATPG tool for the circuit that is being tested, and let N be the test vector's length. In every area, the compressive sensing technique requires a sparse input signal. By adding the basis matrix (Ψ) to the input, one may obtain the sparsity of the signal by converting it from one domain where it is accessible to another where it is scarce. An alternate representation of the input test vector x in base form is given by Eqn. (4.1), $x = \Psi.\ g$~(4.1)

where g is a N × 1 column vector that represents the coefficients of input patterns in the ψ domain, which only contains k elements as non-zero entries, and ψ is a N × N basis matrix. In such case, the input vector x is regarded as k sparse and compressible only in the case where k's length is much less than that of the input vector N [15].

Result

The automated test pattern generation (ATPG) tool uses the gate level net-list as its input file. In every domain, the test vectors supplied as input to the compressive sensing method must be sparse. Consequently, after compressive sensing (CS), the N number of test patterns produced by the traditional test pattern tool are further compressed into M number of test patterns (M\\N). A basic test set that detects the existence of Trojans and enhances test coverage for Trojans in comparison to original test patterns is the result of the test pattern generating procedure. Leakage power, which is regarded as a side channel characteristic, is measured for the circuit under test using the compressive sensing approach that has been suggested. This process is used to derive the appropriate test vectors for activating the Trojan module. Thus, a key factor in locating the Trojan in the original architecture is the best test pattern that was recovered for the side channel analysis. Furthermore, the input test vectors created by the ATPG tool are randomly generated and do not need activation of the low transition probability nets, which are potential Trojan insertion sites.

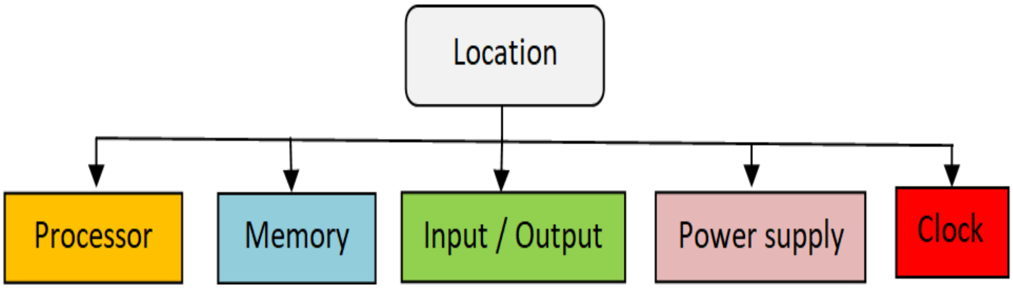

Figure 5: Hardware vulnerabilities in certain places Trojan

Therefore, in order to identify the best collection of test patterns that may activate the uncommon nodes of interest, the compressive sensing technique is included. The compressive sensing algorithm will extract the sparse value from the input test set and construct the most appropriate test patterns when it receives the circuit under test input patterns. Examining an example ISCAS'85 benchmark circuit C432, the ATPG tool produces 63 input test patterns for every 36 bits, which the suggested compressive sensing technique reduces to 23 input patterns for every 36 bits. test vectors created for the C432 circuit using the suggested technique. It can be seen that the compressive sensing system's sparse nature recovers optimum test vectors, improving Trojan detection with a smaller test set. The majority of the suspicious sites are found to be triggered by the sparse test vectors produced by the suggested approach; as a result, these minimum test sets are regarded as important vectors for determining the power profiles for the detection procedure.

Node identification for hardware Trojan insertion

The Trojan module is inserted using a node identification method based on transition likelihood. It is more likely that internal nodes with low transition probabilities (Tp) will be inserted. The reason for this is because the Trojan placed at these low Tp nodes would very rarely activate because of their extremely low switching activity.

The potential of achieving a logic low (0) or logic high (1) value of the test circuit is observed in a flowchart to calculate the overall output probabilities of a particular gate. A net with a high output probability value indicates that there is a greater chance of that net obtaining a certain logic value.

Table 3: Clustering of nets of C432 circuit based on output transition probability value

Features	**Low Tp**	**Medium Tp**	**High Tp**
Range	0≤ β/2	β/2 ≤β	β ≤1
Value	0 – 0.0853	0.0854–0.1706	0.1706–1
No. of nets	38	34	88
% of nets	23.75	21.25	55

When comparing testability measures to the analysis of suspicious nets in the provided circuits, the transition probability metric is recommended since it does not rise linearly from the main input to the primary output. When the probability value of the internal nodes is lower than that of the principal output net, the internal net is deemed suspicious and may be a potential entry point for Trojan modules. The output probability of logic 0 (P0) and logic 1 (P1), which are stated as in Eqn.4.4, are calculated to get the output transition probability of each net.

The product of $OutputTransitionPrbilits = P0 \times P1$ (4.4)

The number of inputs to the gates and the kind of gate may be used to compute the output transition probability. Figure 4.4 shows a flowchart of the transition probability for various logic functions. Combinational and 4-bit counter sequential modules are examples of hardware Trojans that are intended to activate under unusual circumstances and alter the performance of the selected benchmark circuits. For the purpose of inserting the developed Hardware Trojan (HT) modules into the original design, nodes with low Transition Probability (Tp) and high connection are chosen, and the extracted test patterns are used to validate the suggested methodology. To identify the uncommon net in the circuit, the transition probability method is used to produce the probability of each net for the circuit that is being tested. In order to determine the low Tp, medium Tp, and high Tp nets of which the low Tp nets originated from the Trojan insertion procedure the transition probability algorithm will rank the nets in increasing order. To place each net in its own Tp group, a transition probability boundary (β) is taken into account. This threshold is found by averaging the transition probability values of each net in the circuit. The low Tp value range is defined as 0 to (β/2), the medium Tp value range as (β/2) to β, and the high Tp value range as β to 1. Consequently, Tp ranges are established based on the border value of the probability of all nets. The technique is used to extract the transition probability, which will assess the likelihood of each net in the circuit.

For this circuit, β = 0.1706 is the average transition probability value for all nets. Therefore, the clustering transition probability varies from 0 to 0.0853 for low Tp, 0.0854 to 0.1706 for medium Tp, and 0.1707 to 1 for high Tp nets, the algorithm clusters the 160 nets of C432 circuits overall into several Tp groups. Furthermore, the computation of the transition probability value for the circuit being tested has simplified the process of locating Trojan sites by about 23.75%. Of the 160 nets, 38 nets for the C435 circuit are extracted as low Tp nets, meaning they are more likely to contain Trojan insertions.

Logic testing

In order to perform logic testing, all potential test patterns derived from the compressive sensing method are compared between the values of all the nodes in the golden circuit and those of the Trojan-infected circuit. A comparison of the logic values for a golden circuit and a Trojan-infected system may be found in Table 3. The existence of a Trojan module is indicated by variations in the internal nodes and the principal output when a combinational form of Trojan is installed into the low transition probability node.

Test circuits are subjected to metrics like true positive rate (TPR) and probability of detection (PD) in order to verify the effectiveness of compressive sensing in detecting Hardware Trojans. When classifying Trojan nets as either harmful or normal, the true positive (TR) value indicates the proportion of Trojan nets classified as malicious, while the false negative (FN) value displays the proportion of Trojan nets classified as normal.

Power analysis

The test vector sequences are applied to the circuit being tested using the test set that was prepared. Every test vector's dynamic power is measured using the Synopsys prime time tool ©. A four-bit counter was used as the hardware Trojan in this investigation, and its effects will only be seen during a certain clock cycle. For complicated circuits, there is a significant range in the observed power profile due to process variation, and this variation varies depending on the chip architecture. By measuring power profiles for sparse patterns, the proposed compressive sensing-based test patterns approach improves the identification of Trojan module intruded at the low transition probability nets for different circuits. Thus, in order to do away with the need for a golden model for the detection process, the power profiles are assessed by applying the best test patterns at various time periods.

Thus, the term "self-referencing approach" refers to this method of locating the harmful logic. The creation of the power profile for the ideal test pattern derived from the suggested compressive sensing technique is shown in Figure 4.5. At a given time stamp, the observed side channel parameter will change for the Trojan-infected IC, whereas it stays constant for the Trojan-free IC. The noise threshold is determined by averaging the power measurements for each cycle, and the design is anticipated to be Trojan-infected if the fluctuation exceeds the specified threshold. Golden reference circuits are not needed for Trojan identification using this side channel approach technique. The power used by the Trojan-infected circuits varies with that of the golden circuit to provide the best triggering patterns.

Compressive sensing- based detection

While minimizing the amount of input/output bandwidth required, testing quality is maintained. Instead of being created at random, the test vectors for the circuit under test are selected based on uncommon occurrences and unusual triggering of input combinations. Time complexity is decreased by the suggested algorithm's generation of compressed M significant input patterns, which are much less than those of the conventional ATPG patterns. N test vectors are produced using the suggested method and applied to the circuit being tested. At various time instants, the associated N power, such as xq1, xq2,...,xqN, is measured. The signal is compressed using the compressive sensing technique into yq, which is a K linear combination multiplied by the measurement matrix ʃ.

Table 5: Target nodes identification using Transition Probability values

Bench mark circuits	Minimum p values	Nodes with low Tp value	High Connectivity nodes
C17	0.1875	N6, N7	N6,N7
C432	0.0836691	N259, N262, N263, N266, N26 9, N272, N278, N281, N284, N 287, N288, N289, N290, N291, N292,	N295, N299, N300, N301, N302, N302, N308, N318.

		N293, N294, N299, N30 0, N301, N302, N303, N304, N 305, N306, N307	
C499	0.0585938	N378, N379, N380, N381, N382, N383, N384, N385	N289, N325, N312, N372, N378, N379.
C880	0.000061031	N507, N508, N509, N510, N511, N512, N513, N514.	N752, N753, N754, N755, N756, N760, N761, N770
1355	0.00727229	N996, N1001, N1006, N1011, N1016, N1021, N1026, N1031	N794, N 97, N800, N803, N806, N812, N815, N996

Sparse test patterns are used to extract the N power profiles, and the values of the power profiles at each of the many time windows are represented by each column vector in the x matrix. In our suggested technique, the calculated x is encoded with the Gaussian random matrix as the measurement matrix ϕ, with all the matrix components ranging from 0.5 to 1 to produce the output matrix y. This range of measurement matrix elements is selected in order to obtain the power profile's significant value from the sparse matrix. The reconstruction process's correlation coefficient will be weakened by the measurement matrix's random selection of elements that fall within the boundary range. The benchmark circuit is simulated, and the associated power measurements are calculated as self-referencing value x. Additionally, the power profiles are rebuilt using Pearson's co-relation coefficient technique from the sparse observations. By using this technique, the samples are efficiently separated from the golden model and the rebuilt power measurement samples to identify the infected chip. As a result, the abnormality of Trojans at certain time frames is identified by the extraction of power profiles using a compressive sensing method at the output.

Conclusion

The growing threat of hardware Trojans presents a significant challenge to the integrity and security of integrated circuits (ICs). Traditional detection methods often struggle with the complexity and size of modern ICs, making early detection increasingly difficult. The **compressive sensing-based Trojan triggering and detection** approach proposed in this work offers a promising solution to this challenge by leveraging the sparse nature of Trojan-induced anomalies in circuit behavior. This method allows for efficient detection with fewer measurements, thereby reducing the computational burden and making the detection process more scalable and timely. By applying compressive sensing during the **pre-silicon validation phase**, this approach enables the identification of hardware Trojans before the physical fabrication of ICs, allowing designers to mitigate risks early in the design cycle. The ability to detect even subtle Trojans that may not be visible using traditional verification techniques enhances the robustness of the overall hardware security framework. This research also underscores the potential of **compressive sensing** as a foundational tool in hardware security, facilitating both efficient anomaly detection and reducing the time and resources needed for verification. The integration of this technique with existing design and verification flows paves the way for a more secure, efficient, and scalable approach to IC validation. The proposed approach, therefore, represents a crucial step forward in safeguarding hardware systems, particularly in high-stakes applications such as defense, medical, and automotive industries, where security and trust are paramount. Overall, the compressive sensing-based Trojan detection methodology provides an innovative, effective, and practical solution for pre-silicon validation, contributing significantly to the field of **hardware security** and offering a proactive means to ensure the integrity of next-generation integrated circuits. Future research can build upon this work by exploring the integration of machine learning, more advanced sensing techniques, and further optimization for large-scale, high-performance designs.

Reference

1. Moein, S., Subramnian, J., Gulliver, T. A., Gebali, F., & El-Kharashi, M. W. (2015, December). *Classification of hardware Trojan detection techniques.* In *2015 Tenth International Conference on Computer Engineering & Systems (ICCES)* (pp. 357–362). IEEE.

2. Narasimhan, S., Du, D., Chakraborty, R. S., Paul, S., Wolff, F., Papachristou, C., & Bhunia, S. (2010, June). *Multiple-parameter side-channel analysis: A noninvasive hardware Trojan detection approach.* In *Hardware-Oriented Security and Trust (HOST), 2010 IEEE International Symposium on* (pp. 13–18). IEEE.

3. Narasimhan, S., Wang, X., Du, D., Chakraborty, R. S., & Bhunia, S. (2011, June). *TeSR: A robust temporal self-referencing approach for hardware Trojan detection.* In *Hardware-Oriented Security and Trust (HOST), 2011 IEEE International Symposium on* (pp. 71–74). IEEE.

4. Nasr, A. A., & Abdulmageed, M. Z. (2017, February). *An efficient reverse engineering hardware Trojan detector using histogram of oriented gradients. Journal of Electronic Testing, 33*(1), 93–105.

5. Nasr, A. A., & Abdulmageed, M. Z. (2016). *Automatic feature selection of hardware layout: A step toward robust hardware Trojan detection. Journal of Electronic Testing, 32*(3), 357–367.

6. Nguyen, L. N., Yilmaz, B. B., Prvulovic, M., & Zajic, A. (2020, December). *A novel golden-chip-free clustering technique using backscattering side channel for hardware Trojan detection.* In *2020 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)* (pp. 1–12). IEEE.

7. Nourian, M. A., Fazeli, M., & Hély, D. (2018). *Hardware Trojan detection using an advised genetic algorithm-based logic testing. Journal of Electronic Testing, 34*(4), 461–470.

8. Nowroz, A. N., Hu, K., Koushanfar, F., & Reda, S. (2014). *Novel techniques for high-sensitivity hardware Trojan detection using thermal and power maps. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 33*(12), 1792–1805.

9. Pavlidis, A., Faehn, E., Louërat, M. M., & Stratigopoulos, H. G. (2022, April). *Run-time hardware Trojan detection in analog and mixed-signal ICs.* In *2022 IEEE 40th VLSI Test Symposium (VTS)* (pp. 1–8). IEEE.

10. Perez, T., & Pagliarini, S. (2022). *Hardware Trojan insertion in finalized layouts: From methodology to a silicon demonstration. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.*

11. Popat, J., & Mehta, U. (2016, December). *Transition probabilistic approach for detection and diagnosis of hardware Trojan in combinational circuits.* In *2016 IEEE Annual India Conference (INDICON)* (pp. 1–6). IEEE.

12. Pan, Z., & Mishra, P. (2021,

January). *Automated test generation for hardware Trojan detection using reinforcement learning.* In *Proceedings of the 26th Asia and South Pacific Design Automation Conference* (pp. 408–413).

13. Pan, Z., Sheldon, J., & Mishra, P. (2020, November). *Test generation using reinforcement learning for delay-based side-channel analysis.* In *Proceedings of the 39th International Conference on Computer-Aided Design* (pp. 1–7).

14. Pearce, H., Surabhi, V. R., Krishnamurthy, P., Trujillo, J., Karri, R., & Khorrami, F. (2022). *Detecting hardware Trojans in PCBs using side-channel loopbacks. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 30*(7), 926–937.

15. Rahman, M. T., Forte, D., Shi, Q., Contreras, G. K., & Tehranipoor, M. (2014, October). *CSST: Preventing distribution of unlicensed and rejected ICs by untrusted foundry and assembly.* In *2014 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)* (pp. 46–51). IEEE.