



PDF Download
3489517.3530545.pdf
12 February 2026
Total Citations: 3
Total Downloads: 301

 Latest updates: <https://dl.acm.org/doi/10.1145/3489517.3530545>

RESEARCH-ARTICLE

VIPR-PCB: a machine learning based golden-free PCB assurance framework

ARITRA BHATTACHARYAY, University of Florida, Gainesville, FL, United States

PRABUDDHA CHAKRABORTY, University of Florida, Gainesville, FL, United States

JONATHAN CRUZ, University of Florida, Gainesville, FL, United States

SWARUP K BHUNIA, University of Florida, Gainesville, FL, United States

Open Access Support provided by:

University of Florida

Published: 10 July 2022

[Citation in BibTeX format](#)

DAC '22: 59th ACM/IEEE Design
Automation Conference
July 10 - 14, 2022
California, San Francisco

Conference Sponsors:
[SIGDA](#)

VIPR-PCB: A Machine Learning based Golden-Free PCB Assurance Framework

Aritra Bhattacharyay, Prabuddha Chakraborty, Jonathan Cruz, and Swarup Bhunia

Department of Electrical and Computer Engineering, University of Florida

Gainesville, Florida, USA

{abhattacharyay,p.chakraborty,jonc205}@ufl.edu,swarup@ece.ufl.edu

ABSTRACT

Printed circuit boards (PCBs) form an integral part of the electronics life cycle by providing mechanical support and electrical connections to microchips and discrete electronic components. PCBs follow a similar life cycle as microchips and are vulnerable to similar assurance issues. Malicious design alterations, i.e., hardware Trojan attacks, have emerged as a major threat to PCB assurance. Board-level Trojans are extremely challenging to detect due to (1) the lack of golden or reference models in most use cases, (2) potentially unbounded attack space, and (3) the growing complexity of commercial PCB designs. Existing PCB inspection techniques (e.g., optical and electrical) do not scale to large volume and are expensive, time-consuming, and often not reliable in covering diverse Trojan space. To address these issues, in this paper, we present VIPR-PCB, a board-level Trojan detection framework that employs a machine learning (ML) model to learn Trojan signatures in functional and structural space and uses a trained model to discover Trojans in suspect PCB designs with high fidelity. Using extensive evaluation with 10 open-source PCB designs and a wide variety of Trojan instances, we demonstrate that VIPR-PCB can achieve over 98% accuracy and is even capable of detecting Trojans in partially-recovered PCB designs.

1 INTRODUCTION

Printed Circuit Boards (PCBs) play an important role in the electronics life cycle. They, similar to microchips, follow a horizontal business model and a globally distributed supply chain. This supply chain increasingly involves untrusted people, components, and entities in the PCB life cycle – from PCB design house to foundry to component and system supply chain – that compromises the trustworthiness of PCBs and electronic systems manufactured with them [9]. Recent reports highlight major issues in PCB assurance. In particular, vulnerabilities in the PCB supply chain have garnered much attention since the publication of the “Big Hack” [1] article. This widely-circulated article claimed that a tiny, stealthy microchip was found in server motherboards of nearly 30 US companies. The microchip was able to control the PCB and leak sensitive data to an unauthorized receiver.

Fig. 1 illustrates the modern PCB life cycle. It starts from the design house that creates a PCB schematic and layout; then moves

to a foundry that fabricates the bare PCBs and mounts the components on one or more of its layers; and then goes to the testing facility for production testing. The manufactured PCBs that pass testing then flow through a supply chain (typically untrusted) and then are acquired by the system integrator. Finally, the PCBs are integrated into electronic systems, along with other components (e.g., input/output devices, energy sources, physical enclosure, etc.) and again pass through a supply chain (often untrusted) to various field applications. During this long and distributed life cycle, PCBs are vulnerable to malicious alterations at various stages. Fig. 1 pinpoints four major stages where PCB designs can be altered (i.e., Trojans implanted). These stages include an untrusted design facility (e.g., with a rogue designer as an adversary), an untrusted foundry, an untrusted supply chain for PCB, and finally untrusted supply chain of integrated systems.

Existing work on PCB assurance under diminishing trust in its life cycle have primarily focused on investigating the attack vectors [9] [1] [15]. In particular, prior work in this field has identified diverse and powerful Trojan attacks in PCBs with variety of triggers and payloads and analyzed the potential of an attack to be carried out at various parts of the life cycle. Trojan detection in a PCB design is a very challenging verification task. Even with access to a golden PCB design, the task is still daunting as most of the PCBs today are very complex and have multiple layers where Trojans of countless forms can be implanted. While chip-level Trojan detection techniques have been widely investigated [11], they suffer from the following major deficiencies in terms of extending to PCB designs: (1) chip-level techniques cannot be employed at board level, specifically because of the huge diversity in PCB components (as opposed to standard cell-based designs in ASIC or FPGA); (2) board-level inspection techniques (e.g., optical or electrical inspection) do not scale to large volume, and are expensive, time-consuming, and often not reliable in covering diverse Trojan space; and (3) they often rely on the availability of golden or reference designs, which may not be available for PCBs.

In this paper, we present, Verification of IP Trust in PCB (VIPR-PCB), a machine learning (ML) based golden-free PCB Trojan detection framework. VIPR-PCB uses supervised ML to learn board-level Trojan signatures from known attack datasets and then uses this trained ML model to discover Trojan instances in a suspect PCB under test. The proposed framework is scalable, flexible (capable of adapting to evolving Trojan attack space), and can be applied at various stages of the PCB life cycle. Fig. 1 shows the three points in the PCB life-cycle where the VIPR-PCB framework can be used for Trojan detection in PCB. First, it can be used by a design house after a manufactured (optionally tested) PCB returns to the design facility and before it goes to a PCB supply chain. Second, it can be used by a system integrator to verify the trustworthiness of acquired third-party PCBs. Finally, before field application, a user can

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

DAC '22, July 10–14, 2022, San Francisco, CA, USA

© 2022 Association for Computing Machinery.

ACM ISBN 978-1-4503-9142-9/22/07...\$15.00

<https://doi.org/10.1145/3489517.3530545>

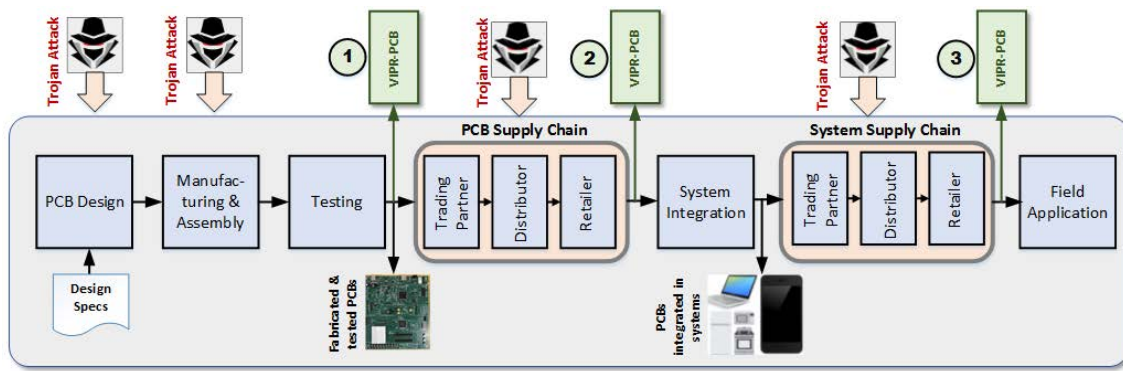


Figure 1: Long and distributed life cycle of a PCB with four major stages, which are vulnerable to Trojan insertion. The proposed PCB Trojan detection framework (VIPR-PCB) can be used at three different points in PCB life-cycle. It assumes availability of full or partial PCB design information (e.g. through reverse engineering) at these points.

use VIPR-PCB to verify the trustworthiness of PCBs integrated into a working system. Note that VIPR-PCB requires a PCB design information in terms of its structure and/or function – either complete or partial – to perform an ML-based trust verification. We assume that extensive prior work on PCB reverse engineering (RE) [4] [14] and commercial practices (e.g., using X-Ray Tomography and computer vision) can be employed to obtain the structural/functional design information from a PCB for performing the proposed trust verification process.

In particular, the paper makes the following major contributions:

- It identifies the Trojan attack space in the PCB life cycle and pinpoints the deficiencies of existing work in detecting PCB-level Trojan attacks that come with diverse trigger/payload characteristics.
- It presents an automatic PCB level Trojan detection framework, VIPR-PCB, that can identify unknown Trojan instances in a PCB design. The framework is based on training a supervised ML-based classifier with a set of structural (sub-graph based) and functional (activity-based) features. It integrates an automatic Trojan insertion tool in a PCB design [12] to generate a large population of random Trojans of various types. The Trojan insertion tool makes the Trojan detection framework dynamic and flexible, making it capable of incorporating new classes of Trojans.
- The framework is capable of detecting Trojan instances on complete or partial PCB designs. The partial designs reflect practical use cases where a PCB design is not fully recovered, e.g., due to the limitation of the RE process.
- It presents a complete tool flow for PCB-level Trojan detection interfacing with commercial PCB design tools. By evaluating 10 open-source PCB designs, it presents an extensive analysis of the VIPR-PCB framework and provides a quantitative analysis of its effectiveness under various use cases and assumptions. It shows that the ML model can classify individual components or the entire PCB as malicious with very high accuracy.

The rest of the paper is organized as follows: Section 2 consists of the background and motivation behind the proposed PCB assurance framework. Section 3 consists of the proposed golden-Free Trojan

detection methodology. The experimental setup and the evaluation results are presented in Section 4, followed by the conclusion and future directions in Section 5.

2 BACKGROUND AND MOTIVATION

2.1 Trojan Model

PCB Trojans are malicious modifications in a PCB design, consisting of two primary sections: trigger and payload. Fig. 2 provides a high-level illustration of a PCB Trojan. The Trojan trigger is constructed from nets tapped from various locations and is activated only under a specific condition. On being triggered, the payload alters the connection between an IC and a sensor, resulting in erratic sensor behavior. There are two types of application-independent PCB level Trojans: ASIC-based Trojans (ABT) and Discrete component-based Trojans (DBT). In an ASIC-based Trojan, an IC or a programmable circuit of a very small form factor can be used as a trigger and payload. Discrete component-based Trojans consist of transistors, resistors, operational amplifiers, diodes, and capacitors. Using the above mentioned components, a circuit is created that behaves like a logic gate, for example, AND gate, XOR gate, etc. Discrete component-based Trojans consist of simple trigger and payload conditions, but they can blend in with the PCB design.

A good PCB Trojan can hide its footprint in several ways. One way is by ensuring the Trojan consists of similar components to those in the host PCB or component camouflaging. Another method to reduce detection is selecting a trigger with a low activation probability. But there is a class of PCB Trojans that are always active/always-on. An always-on Trojan may or may not have a trigger circuit, and will remain active since its inception or upon triggering. Always-on Trojans are usually used to leak sensitive information to an unauthorized receiver. In this case, sizing and side-channel footprint reduction can further help achieve stealthiness.

2.2 Related Works

The existing hardware Trojan detection techniques have focused primarily on gate-level netlists of an ASIC. In [11], the authors propose an ML model using structural and functional features for the detection of hardware Trojans in a gate-level IP. The authors propose an ensemble ML model using 13 structural and functional

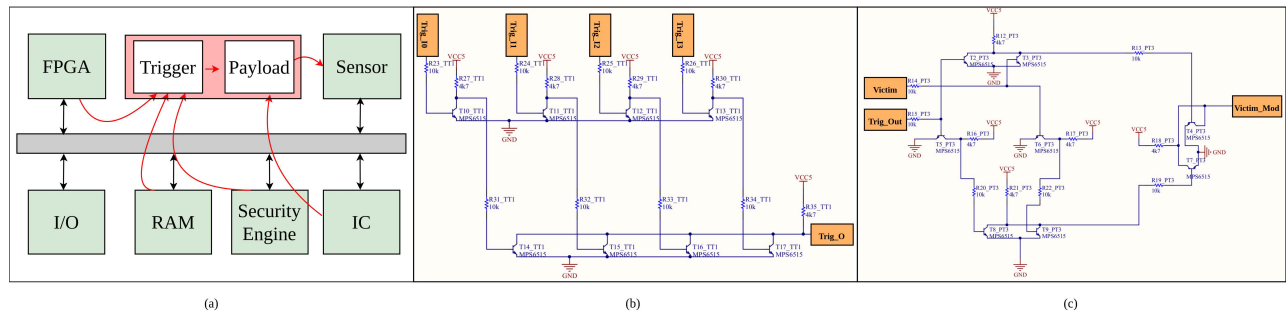


Figure 2: Example board-level Trojan: (a) represents a high-level overview of a PCB Trojan, (b) represents a NOR logic based Trigger, and (c) represents a XOR logic based Payload.

features. There have also been recent works on hardware Trojan detection, such as [10] and [17]. In [10], the authors propose a two-step neural network approach in which the framework first identifies the Trojan probability of each net in the design, and then finds the Trojan probability of the neighboring nets of each of the considered net. In [17] the authors use an ensemble classifier based on the K-means algorithm to detect and locate hardware Trojan signals. The authors of [7], [8] use neural networks on side-channel data to detect hardware Trojans in an IC.

Currently, little research has been carried out in detecting malicious behavior in a PCB design. In [2] and [16], the authors propose a side-channel based anomaly detection technique using the power distribution network of a PCB. The Power Distribution Network (PDN) [18] has been used to monitor the change in the impedance profile while a PCB is running a workload. However, there are several issues with the approach. The primary issue is the proposed method cannot detect Trojans in the pre-fabrication stage of the design flow. A Trojan can be stealthy, resulting in minimal change in the impedance profile that might not be detected depending on the threshold. Another issue is the user needs to know the impedance profile of all the workloads running in a PCB. The power consumption for each PCB will be different, thus making it very difficult to detect Trojans. It will result in very high false positives and might also contain a significant process noise. Also, the impedance profile alone cannot be relied on to detect Trojans in a PCB. The authors of [5] propose a PCB authentication system by automatically generating a Bill of Materials (BoM) for a PCB using image analysis. The components in a PCB might be embedded inside the internal layers, as discussed in [6], making image analysis difficult. Moreover, the automatic BoM requires the user to have a reference golden design for comparison. Yet, even with this reference design for comparison, a PCB cannot be completely authenticated. Table 1 highlights the main differences between our proposed framework and existing PCB detection frameworks.

2.3 Motivation

The horizontal business model for PCB involving untrusted parties has resulted in the increasing need for trust verification during the different stages of this process. Fig. 1 represents the primary stages in the life cycle of modern PCB. In most cases, the end-user does not have a reference or golden PCB design. Even if a reference PCB design is available, malicious components and traces might be hidden in the internal layers of a PCB, making it difficult to detect

Table 1: Comparison with Related Work

Related Works	Golden-free detection	Works on Partial PCB	Side channel measurement	Trojan coverage
Zhu et al. [19]	No	No	Required	Limited
Rossi et al. [16]	No	No	Required	Limited
Ghosh et al. [9]	No	No	Required	Limited
Azhagan et al. [5]	No	No	Not Required	Limited
Proposed method	Yes	Yes	Not required	Very high

them through traditional optical inspection techniques. In the stages of a PCB design and manufacturing, there are possibilities of Trojan insertion starting from the schematic to the fabricated design steps. Similarly, PCBs can be inserted post-manufacturing since surface-mounted components and traces are often easily accessible for malicious alterations. It requires a PCB assurance framework that does not rely on golden models and is amenable for application at different stages. ML models have been used to detect Trojans in gate-level netlists without golden models. So, similarly, for PCB trust verification, machine learning appears to be effective. Using a Trojan insertion tool, various types of Trojans can be inserted in a large number of PCB designs. The PCB designs and Trojan inserted designs can be used to train an ML model for verifying the trustworthiness of any new PCB design.

3 TROJAN DETECTION METHODOLOGY

A golden-free Trojan detection framework is necessary for accurately detecting Trojans in a PCB design. We propose a framework that can detect Trojans in a PCB design using ML models. The overall flow of the Trojan detection framework is described in Fig. 3. Initially, a Trojan benchmarking tool is used to generate the Trojan inserted designs for each type of Trojan under consideration. We then extract the structural and structural-functional features of the Trojan inserted and benign designs. The extracted features are represented in the form of an adjacency matrix which is flattened to form a feature matrix. The data of the feature matrix is split into train and test datasets comprising of different benign and Trojan designs, having no overlap in Trojan type or base design. A machine learning classifier is trained on the training dataset, comprising of benign and Trojan inserted design features. The trained ML model is evaluated on the test dataset comprising of different benign and Trojan inserted designs.

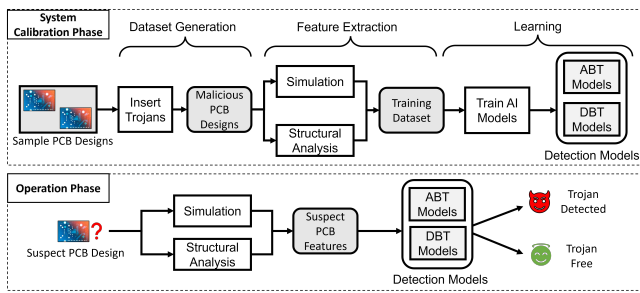


Figure 3: Overview of the proposed framework for hardware Trojan detection using machine learning.

3.1 Training Dataset Generation

The presence of Trojan-inserted PCB designs is one of the primary requirements for an ML-based Trojan detection framework. The Trojan inserted dataset needs to be sufficient and robust. Therefore, we use a dynamic and automated Trojan insertion tool that can insert Trojans in any PCB design. The tool simulates the input design using an open-source simulator to generate the activation probability of each net in the input design. Based on the resulting activation probability of the nets, the tool selects the rare nets and connects them to the Trojan trigger. The Trojan payload is connected with a randomly selected design net at a random location to increase the coverage of the Trojans across the design. The structure of the Trojan templates is also modified to provide variants.

An extensive database of Trojans covering most of the design space can be generated using the Trojan insertion tool. Additionally, the Trojan insertion tool makes the detection method robust by developing structural variations in the Trojan design. The tool can also support new Trojan designs which follow specific input/output rules. This feature enables us to integrate support for detecting new Trojan designs into the PCB Trojan detection framework.

3.2 Dataset Pre-processing

The benign and Trojan-inserted PCB designs are pre-processed before extracting the features. Unlike gate-level designs, PCB designs do not have a standard library which results in a high variation of component names across PCB designs. Therefore, we create a macro digest of possible PCB components for labeling and understanding the PCB. The digest consists of 10 component categories: Programmable IC, Non-Programmable IC, Resistor, Capacitor, Crystal, Memory, Inductor, Transistor, and Connector. Components that support a workload or have programmable fabric fall under the Programmable IC category. Components consisting of digital circuits, like operational amplifiers, are considered under the Non-Programmable IC category. Resistor, Capacitor, and Diode consist of analog resistors, capacitors, and diodes, respectively. Crystals and oscillators are considered to be Crystals. Filters and inductors are considered to be Inductors. Memory devices like RAMs are categorized as Memory. Discrete transistors are considered Transistors, and input/output pins are categorized as Connectors. While parsing the PCB, every component is annotated with a digest category.

3.3 Feature Extraction

Based on the pre-processed PCB design description, the framework extracts a set of functional and structural features used in the subsequent steps. Fig. 4 illustrates the process of extracting the adjacency matrix from a sample PCB design.

3.3.1 Structural Feature Extraction. The component connectivity of a PCB Trojan inherently contains patterns that can be leveraged for detection. Therefore, we extract structural features in the following manner. For each component in a PCB design, the frequency of its connection with different components is calculated. The calculated frequency is stored in an adjacency matrix consisting of the 10 categories of components mentioned previously. The adjacency matrix can be represented as a flattened vector consisting of 55 component_component connection pairs. During the flattening process, the duplicate component_component pairs are removed. This flattened vector is used as the feature vector for the ML algorithms. Next, the flattened vector is handled depending on the type of analysis. For board-level analysis (BLA), all the feature vectors extracted from all the components are summed to form a single feature vector for each PCB. BLA is used to detect whether a PCB design as a whole contains a Trojan.

In the case of the element-level analysis (ELA), the feature vector of each component is combined to form a feature matrix. ELA can be used to identify the individual components that contribute to the Trojan present in a PCB design.

3.3.2 Combined Structural and Functional Feature Extraction. Structural features alone might not be able to model Trojan behavior in a PCB. Stealthy Trojans need to have not only a low footprint but also a low activity. Therefore, considering the characteristics of a good Trojan, we incorporate functional features to capture the behavior of a Trojan in a PCB design. Specifically, we use the activation probability of nets as a functional feature. We combine the structural and functional features using a unique method. While extracting just structural features, the frequency of the connection between two components increases by a factor of 1. When structural and functional features are combined, the frequency of connection between two components is increased by a factor of the activation probability of the respective net that connects them. Thus, the connectivity features of a design are weighted by the activation probability of the respective nets. Therefore, the combined structural and functional features capture both the connectivity pattern and the functional behavior of Trojans in a PCB. We also perform board-level and element-level analysis for combined structural and functional features.

3.3.3 Partial Board Feature Extraction. In a practical scenario, it is difficult to completely reverse engineer a PCB design. Hence, it is important for a PCB Trojan detection scheme to be able to operate in such conditions. We enable this partial board recovery scenario (partial board) by partitioning the PCB into subgraphs using hmetis [13]. Using the hmetis tool, a PCB design graph is partitioned into a random number of partitions. A random subset of these subsequent partitions is selected for removal. During the creation of the adjacency matrix, the components and nets belonging to the removed partitions are not considered. This partition is applied for both benign and Trojan inserted designs.

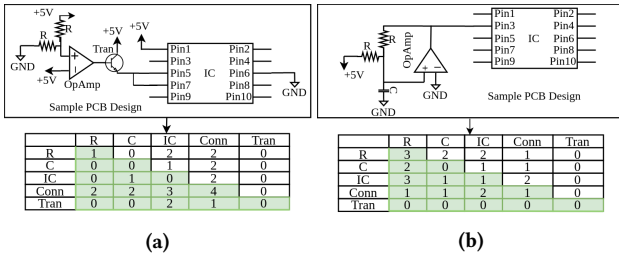


Figure 4: The feature extraction process from sample PCB designs: In (a) and (b) the sample PCB design (top) and the corresponding adjacency matrix (bottom) are shown. The part of the adjacency matrix colored in green is flattened to make the feature vector.

3.4 Trojan Detection in Suspect PCB

The feature matrix generated in the previous section is processed to generate training and testing datasets. The training data is used to train an ML algorithm to distinguish between benign and Trojan inserted designs. While evaluating the accuracy among Support Vector Machine (SVM), Logistic Regression, and Random Forest (RF) ML models, the RF classifier performed the best in detecting Trojans. Hence, we use the Random Forest classifier model with our framework to detect Trojans in a PCB design.

Random Forest is an ML algorithm based on decision tree algorithms. The algorithm utilizes ensemble learning in the prediction of an outcome. Multiple classifiers are used that solves a complex problem. The algorithm is trained using bootstrap aggregating, which generates multiple training sets. Each of the generated training sets is used to train particular decision trees. The variance of each decision tree might be high, but the overall variance is reduced. The reduced variance helps in increasing the stability of the algorithm which helps in preventing over-fitting. The average outcome of multiple decision trees is considered for the final prediction.

4 EXPERIMENTS AND RESULTS

In this section, we present the evaluation of the proposed PCB-level Trojan detection framework. We procure 10 open-source KiCAD [3] golden designs (see Table 2) and use the automatic Trojan insertion tool proposed in [12] to generate a diverse Trojan inserted design database. For both BLA and ELA, we first randomly split the golden designs into two bins (train and test) based on a train-test split ratio. Next, we generate 15 variants for each design through non-malicious alterations without affecting the functionality. This leads to 150 Trojan-free golden design variants split into test/train bins. For generating the Trojan inserted designs, we insert 2 to 6 ABT/DBT Trojans in the original 10 golden designs and split them based on the same train-test split ratio. For experimenting with partial PCB designs, we first use hmetis [13] to generate a random number of partitions in the range of 8 to 20. Subsequently, a set of 1 to 8 partitions are considered hidden during the analysis.

4.1 Board-Level Analysis (BLA)

In Table 3, for DBT with complete designs, we observe > 90% Trojan/non-Trojan classification accuracy, Trojan class precision, and Trojan class recall. The same values are more than 80%, for ABT with complete designs. We observe a slight drop in detection

Table 2: Sample PCB designs for inserting Trojans

Name of Design	No. of comp.	No. of layers	No. of conn.	DC based Trojans inserted	ASIC based Trojans inserted
CIAA_ACC	569	12	787	15	10
EDU_CIAA_K60	187	2	238	15	10
CIAA_K60	467	4	407	15	10
CIAA_FSL-MINI	141	4	188	15	10
CIAA-PIC	433	4	404	15	10
EDU-INTEL	87	2	150	15	10
CIAA_PICO	67	2	144	15	10
CIAA_SAFETY	275	2	240	15	10
CIAA-Z3R0	33	2	52	15	10
Audio_Amp	49	2	78	15	10

Table 3: Trojan detection results for board-level analysis

Trojan Type	Features Used	Accuracy (%)		Trojan Class Precision (%)		Trojan Class Recall (%)	
		Complete Design	Partial Design	Complete Design	Partial Design	Complete Design	Partial Design
DBT	Struct	91.7	89.2	93.6	91.3	94.8	93.5
	Struct+Func	95.8	92.1	94.5	94.6	99.7	99.3
ABT	Struct	85.7	68.6	85.4	71.0	100	97.0
	Struct+Func	86.2	84.3	86.2	82.9	100	99.4

Table 4: Trojan detection results for element-level analysis

Trojan Type	Features Used	Accuracy (%)		Trojan Class Precision (%)		Trojan Class Recall (%)	
		Complete Design	Partial Design	Complete Design	Partial Design	Complete Design	Partial Design
DBT	Struct	97.6	94.8	94.9	92.5	96.2	92.6
	Struct+Func	98.0	95.6	95.0	93.2	95.7	93.7
ABT	Struct	95.4	89.6	76.3	90.7	87.6	86.5
	Struct+Func	96.7	92.8	95.1	92.4	91.4	88.4

efficiency for partial designs. We also observe structural (struct) and functional (func) features combined generally lead to better accuracy for both ABT and DBT. Note that the recall of our models is generally high, indicating low false negatives. ABT with only structural features appear harder to detect because they generally have a higher similarity with the host design. However, the detection efficiency increases when both structural and functional features are combined.

4.2 Element-Level Analysis (ELA)

For element-level analysis (see Table 4), the accuracy, precision, and recall values are generally high (> 90%) except for ABT with only structural features. As explained earlier, ABTs have a feature space overlap with the host design; hence detecting them is more difficult. As this is an element-level analysis, if the framework can detect most of the malicious PCB components/elements, then the undetected/misclassified malicious components can be subsequently inferred through manual inspection.

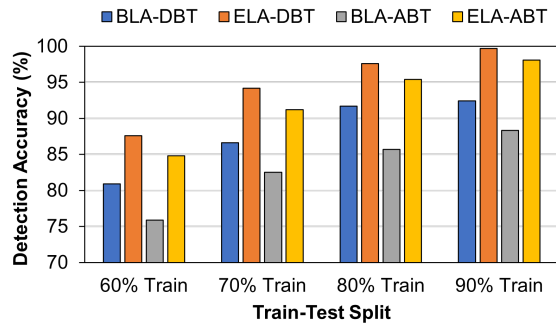


Figure 5: Overall detection accuracy across all scenarios for different training dataset sizes with structural features.

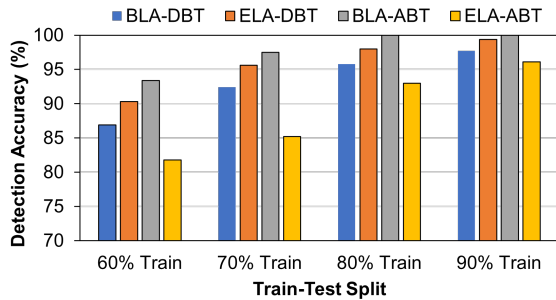


Figure 6: Overall detection accuracy across all scenarios for different training dataset sizes with both structural and functional features.

4.3 Training Data Requirement Analysis

Obtaining good quality training data for PCB Trojan detection can be challenging. Hence, in Fig. 5 and Fig. 6, we observe the effect of the training dataset size on the detection accuracy by modifying the train-test split of the entire dataset. Increasing the training dataset leads to higher detection efficacy. Even using just 60% of the entire dataset for training, we obtained $> 75\%$ accuracy for all scenarios with only structural features. We observe a similar trend in both structural and functional features.

5 CONCLUSION

We have presented VIPR-PCB, a comprehensive ML-based PCB-level Trojan detection framework that eliminates the need for golden or reference models. It integrates an automatic Trojan insertion tool for creating a training dataset capable of producing large structural and functional variants of Trojan instances for various Trojan classes. It uses a combination of structural (represented as a subgraph) and functional (computed as activation probabilities) features for training the ML models. The proposed method can not only detect whether a PCB board is malicious but also can determine the type and location of malicious components in a PCB board. We have developed a completely automated tool flow for VIPR-PCB, interfaced with commercial PCB design flow (with Ngspice for functional simulation), and evaluated it extensively with diverse PCB designs and Trojan types. We show that the proposed approach is highly effective in detecting unknown design alterations, and the false positive rate is very low ($< 5\%$). The method is scalable to large PCB designs and flexible to account for emerging Trojan types by

accounting for a new Trojan template during training dataset creation. Our future work will focus on improving the Trojan detection capability by considering additional features and more powerful ML models and evaluating emerging Trojan types.

ACKNOWLEDGEMENT

The work is funded in part by Defense Advanced Research Projects Agency (DARPA) grant FA8750-21-1-1001.

REFERENCES

- [1] 2018. The Big Hack: How China used a Tiny Chip to Infiltrate U.S. Companies. <https://shorturl.at/lvzAF> Bloomberg Businessweek.
- [2] 2020. PDNPulse By: Xiaolong Guo and Yier Jin. <https://cadforassurance.org/tools/pcb-trust-verification/pdnpulse/> PDN.
- [3] 2021. KiCad EDA: A Cross Platform and Open Source Electronics Design Automation Suite. <https://www.kicad.org/> KiCad.
- [4] Navid Asadizanjani, Mark Tehranipoor, and Domenic Forte. 2017. PCB Reverse Engineering Using Nondestructive X-ray Tomography and Advanced Image Processing. *IEEE Transactions on Components, Packaging and Manufacturing Technology* 7, 2 (2017), 292–299. <https://doi.org/10.1109/TCPMT.2016.2642824>
- [5] Mukhil Azhagan, Dhvani Mehta, Hangwei Lu, Sudarshan Agrawal, P Chawla, M Tehranipoor, D Woodard, and N Asadi. 2019. A Review on Automatic Bill of Material Generation and Visual Inspection on PCBs. In *ISTFA 2019: Conference Proceedings from the 45th International Symposium for Testing and Failure Analysis, ISTFA*. 256–265.
- [6] Bob Baddeley. 2021. The Components Are INSIDE The Circuit Board. <https://hackaday.com/2017/09/11/the-components-are-inside-the-circuit-board/>
- [7] Sina Faezi, Rozhin Yasaei, and Mohammad Abdullah Al Faruque. 2021. HTnet: Transfer Learning for Golden Chip-Free Hardware Trojan Detection. In *2021 Design, Automation Test in Europe Conference Exhibition (DATE)*. 1484–1489. <https://doi.org/10.23919/DATE51398.2021.9474076>
- [8] Sina Faezi, Rozhin Yasaei, Anomadarshi Barua, and Mohammad Abdullah Al Faruque. 2021. Brain-Inspired Golden Chip Free Hardware Trojan Detection. *IEEE Transactions on Information Forensics and Security* 16 (2021), 2697–2708.
- [9] Swaroop Ghosh, Abhishek Basak, and Swarup Bhunia. 2015. How Secure Are Printed Circuit Boards Against Trojan Attacks? *IEEE Design Test* 32, 2 (2015), 7–16. <https://doi.org/10.1109/MDAT.2014.2347918>
- [10] Kento Hasegawa, Tomotaka Inoue, and Nozomu Togawa. 2021. A Two-Stage Hardware Trojan Detection Method Considering the Trojan Probability of Neighbor Nets. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences* adpub (2021). <https://doi.org/10.1587/transfun.2020KEP0005>
- [11] Tamzidul Hoque, Jonathan Cruz, Prabuddha Chakraborty, and Swarup Bhunia. 2018. Hardware IP Trust Validation: Learn (the Untrustworthy), and Verify. In *2018 IEEE Int. Test Conf. (ITC)*. IEEE, 1–10.
- [12] Tamzidul Hoque, Shuo Yang, Aritra Bhattacharyay, Jonathan Cruz, and Swarup Bhunia. 2020. An Automated Framework for Board-level Trojan Benchmarking. *CoRR* abs/2003.12632 (2020). arXiv:2003.12632 <https://arxiv.org/abs/2003.12632>
- [13] George Karypis. 1998. hMETIS 1.5: A hypergraph partitioning package. <http://www.cs.umn.edu/~metis> (1998).
- [14] Stephan Kleber, Henrik Ferdinand Nölscher, and Frank Kargl. 2017. Automated PCB Reverse Engineering. In *11th USENIX Workshop on Offensive Technologies (WOOT 17)*. USENIX Association, Vancouver, BC. <https://www.usenix.org/conference/woot17/workshop-program/presentation/kleber>
- [15] Dhvani Mehta, Hangwei Lu, Olivia P. Paradis, Mukhil Azhagan M. S., M. Tanjidur Rahman, Yousef Iskander, Praveen Chawla, Damon L. Woodard, Mark Tehranipoor, and Navid Asadizanjani. 2020. The Big Hack Explained: Detection and Prevention of PCB Supply Chain Implants. *J. Emerg. Technol. Comput. Syst.* 16, 4, Article 42 (aug 2020), 25 pages. <https://doi.org/10.1145/3401980>
- [16] Gor Piliposyan, Saqib Khursheed, and Daniele Rossi. 2020. Hardware Trojan Detection on a PCB Through Differential Power Monitoring. *IEEE Transactions on Emerging Topics in Computing* (2020), 1–1. <https://doi.org/10.1109/TETC.2020.3035521>
- [17] M. Tebyanian, A. Mokhtarpour, and A. Shafieinejad. 2021. SC-COTD: Hardware Trojan Detection Based on Sequential/Combinational Testability Features using Ensemble Classifier. *Journal of Electronic Testing* (2021). <https://doi.org/10.1007/s10836-021-05960-2>
- [18] Huifeng Zhu, Xiaolong Guo, Yier Jin, and Xuan Zhang. 2020. PowerScout: A Security-Oriented Power Delivery Network Modeling Framework for Cross-Domain Side-Channel Analysis. In *2020 Asian Hardware Oriented Security and Trust Symposium (AsianHOST)*. 1–6. <https://doi.org/10.1109/AsianHOST51057.2020.9358263>
- [19] Huifeng Zhu, Xiaolong Guo, Yier Jin, and Xuan Zhang. 2021. PCBench: Benchmarking of Board-Level Hardware Attacks and Trojans. In *2021 26th Asia and South Pacific Design Automation Conference (ASP-DAC)*. IEEE, 396–401.