SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I6P111

Volume 11 Issue 6, 101-107, June 2024
© 2024 Seventh Sense Research Group®

Original Article

Detection and Location of Hardware Trojans Using Path Delays

Allagadda Seetharamaraju[1], Arun Raaza[2], Omprakash Laxmanappa Mandge[3], M. Meena[4]

[1,2,4]Vels Institute of Science, Technology and Advanced Studies, Tamilnadu, India.
[3]MET Institute of Computer Science, Mumbai Educational Trust, Mumbai, Maharastra, India.

[1]Corresponding Author : seetharamaraju.allagadda@gmail.com

Received: 05 April 2024 Revised: 08 May 2024 Accepted: 06 June 2024 Published: 29 June 2024

Abstract - The *IC industry transition to a decentralized and outsourced business strategy, wherein design, inclusion, production, packaging, testing, and assembly are being performed by numerous firms around the world, must have generated issues pertaining to IP rights piracy, reverse engineering threats on IC netlists as well as layouts, cloning of ICs, and detecting deceptive chips and also Hardware Trojans (HT). Threats posed by the injection of HTs have developed as a major issue for different vendors, governmental entities, and their vendors. This paper offers a new strategy for identifying HT using path-delay-dependent parametric techniques. The key advantage of this technique is that there is no requirement to compare the HT-inserted circuit to the golden reference circuit. The proposed method has been implemented in 4-bit RCA for detecting and locating the HTs.*

***Keywords* -** *IC, Hardware Trojan, RCA, Path-delay, Netlist.*

1. Introduction

With the advent of the internet and technology, we now face more significant threats of cyber assaults in our daily lives compared to ever before, and new ways of cyber threats are appearing every day. In the initial stages, cyber-attacks were classed as harmful software-based intrusions, with little attention paid to the physical level. The architectural complexities of IC chips are increasing as semiconductor technology has advanced, making it more challenging to detect a minimal quantity of malicious circuitry inside an IC chip. Simultaneously, with the advent of worldwide corporate cooperation and severe rivalry amongst IC design firms, an increasing number of design firms are outsourcing some of their conceptual design to other company engineers to save time and money. This strategy introduces several weaknesses in the IC design phase, as well as possible risks to the IC chip's security [1, 2].

One of the threats to Integrated Circuits (ICs) and their protection is the presence of hardware trojans. These trojans are malicious modifications made to the circuit design of an IC, which can cause harm to the IC. Because a Hardware Trojan does have a minimal design cost as well as a great potential for further development, it has gotten increased attention from the research community [3, 4]. So far, no strong evidence has indeed been discovered to demonstrate whether hardware trojans exist and do damage in the physical realm [5]. In 2007, the Israeli Air Force attacked a suspected partially built Syrian nuclear reactor [6]. The Syrian defence system has yet to respond to Israeli aircraft. This occurrence prompts speculation as to why Syria's missile defence system did not respond to Israeli aircraft. The radar's visibility was lost due to advanced jamming equipment, according to the description [7, 8].

Figure 1 depicts a scenario of Hardware Trojan injection at different stages of IC design. Inserting hardware Trojans may be accomplished by removing, adding, or altering the configuration of integrated circuits. Attackers introduce hardware Trojans within IC design for a variety of reasons, including maligning the organization in order to boost the reputation of a competitor company, unlawfully exploiting the security system, and leaking sensitive data from the IC chip [9].

There has been significant research conducted on novel methods for detecting hardware trojans in recent times. These novel tactics involve utilizing detection methods that rely on the distinctive characteristics of hardware Trojans. This encompasses the assessment of the underlying structure of logic circuits, the scrutiny of the logical composition of Integrated Circuits (ICs), and the analysis of side-channel signals. The research [10-13] indicates that side-channel signals consistently emit bypass signals, such as electromagnetic radiation, temperature information, power use signals, and time delay signals.

With this research's modelling, the injecting of Hardware Trojan would affect certain bypass signals within the IC design process. Observing these impacts is the basis of the detection method using side-channel signals. To showcase, [14] uses the side-channel signal detection technique to determine the existence of a Hardware Trojan by examining the input and output signals. The work described in [15] presents an approach for detecting hardware Trojans by using dynamic power usage.

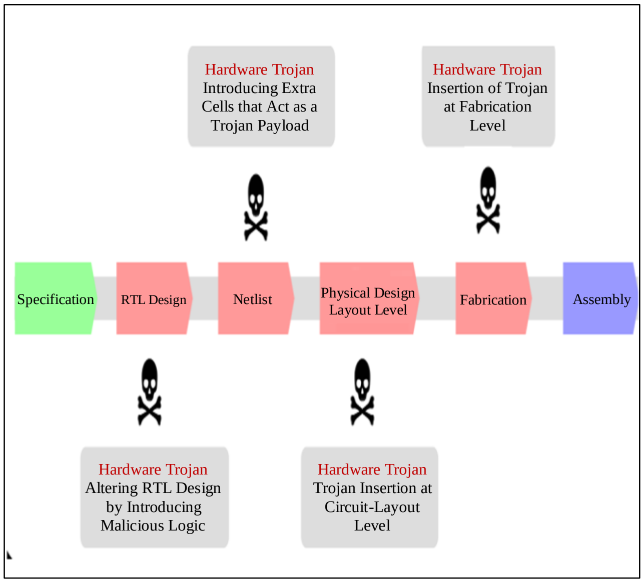

Hardware Trojan
Introducing Extra Cells that Act as a Trojan Payload
Hardware Trojan
Insertion of Trojan at Fabrication Level
Specification
RTL Design
Netlist
Physical Design Layout Level
Fabrication
Assembly
Hardware Trojan
Altering RTL Design by Introducing Malicious Logic
Hardware Trojan
Trojan Insertion at Circuit-Layout Level

Fig. 1 IC design phases during which a Hardware Trojan could be introduced [9]

It processes its power value using a single-value decomposition technique, and the viability of this approach is proved using FPGA. The main premise is that whenever the circuit is operational, it will use energy, much like the Hardware Trojan inside an IC. As a result, under the identical input test vectors, IC having Hardware Trojan as well as IC with no Hardware Trojan exhibits small changes in power usage pulses.

The power consumption of an IC having a Hardware Trojan is higher when compared to the IC without a Hardware Trojan. By observing variations in electric currents, like in [9], hardware Trojans may be found. It depends on connecting various numbers of inverters as a technique to observe the delay time of every IC line in order to ascertain if there is a hardware trojan or a defect. The danger of harming digital devices could be decreased, the budget of IC manufacturing could be reduced, and users may be protected from the risks of attackers by identifying the Hardware Trojan within IC design.

This study employs a novel path delay technique to identify and pinpoint the presence of hardware Trojans. This research presents a novel approach for detecting Hardware Trojans (HT) utilizing path-delay-dependent parametric approaches. An inherent benefit of this technique is the absence of a need to compare the HT-inserted circuit with the golden reference circuit. The suggested approach has been applied to a 4-bit Ripple Carry Adder (RCA) to identify and determine the positions of the Hardware Trojans (HTs). Section 2 analyses the background information on hardware trojans. Section 3 discusses the proposed method for the detection and location of the combinational circuitry hardware trojans. Section 4 presents an analysis of the simulation findings, which is then followed by the conclusion in Section 5.

2. Background Information on the Hardware Trojans

Hardware Trojan is indeed a harmful change of an IC chip's circuitry, which is commonly known as 'HT'. HT would be a hardware component that is hidden within another bigger item of hardware. It awakens at inconvenient times and performs something harmful, resulting in inconvenience for the user.

2.1. Architecture of Hardware Trojan

As shown in Figure 2, the structure of the Hardware Trojan has been predominantly composed of two functional components [9].

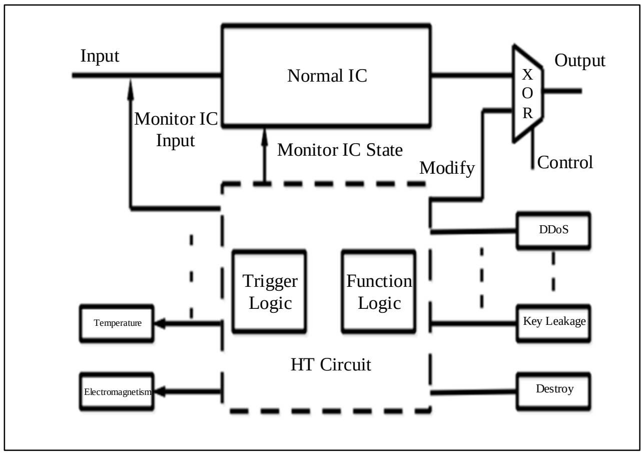


Fig. 2 The block diagram illustrating Hardware Trojan [9]

Hardware Trojan circuitry, as shown in Figure 2, consists of two components, which are discussed further.

- The trigger logic component is one of them. The trigger may initiate the Hardware Trojan's functioning logic component [16]. The criteria of Hardware Trojan assaults are classified into two forms based on whether the trigger has been needed: conditional trigger as well as unconditional trigger. The terminology "conditional trigger" describes that the Hardware Trojan circuitry consists of the triggering and functional logic units [17].

Since they may specify more stringent trigger conditions, the Hardware Trojans could elude detection by typical functional testing [18].

- The functional logic component is the second (payload). The payload has been the executing component of Hardware Trojan assaults, and it is in charge of executing Hardware Trojan assaults [16]. Whenever the payload gets triggered, it disturbs the IC's normal functionality by modifying the IC's result. The payload may be classified as explicitly recessive based on its behavioural features and functioning procedures [19].

2.2. Classification of Hardware Trojans

Hardware Trojans are usually categorized into analog and digital types, as seen in Figures 3 and 4, respectively. Digital Hardware Trojan may modify the values of the storage module or influence the logic value in the circuit node. Analogue Hardware Trojan may modify circuit characteristics like power, noise margin, supply, delay, and so on. Again, the digital trojans, depending on circuitry type, are classified as combinational and sequential hardware trojans [20]. The absence of register circuitry in the trojan circuit distinguishes the combinational trojan. Also, the output function has been solely reliant on the input signal as shown in Figure 5. The sequential trojan circuitry has a memory; the state change is governed by the clock as well as the input signal, as shown in Figure 6 [21].

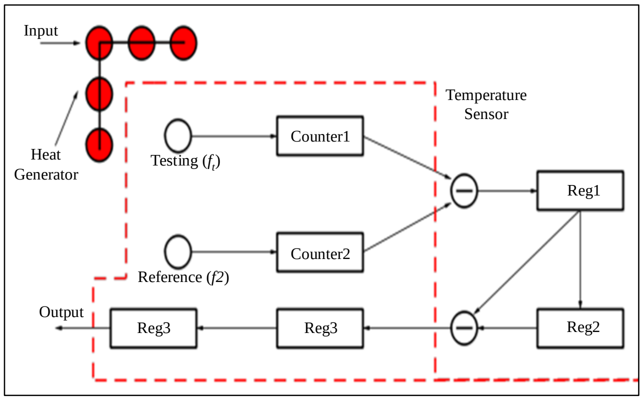


Fig. 3 Illustration of analog HT [20]

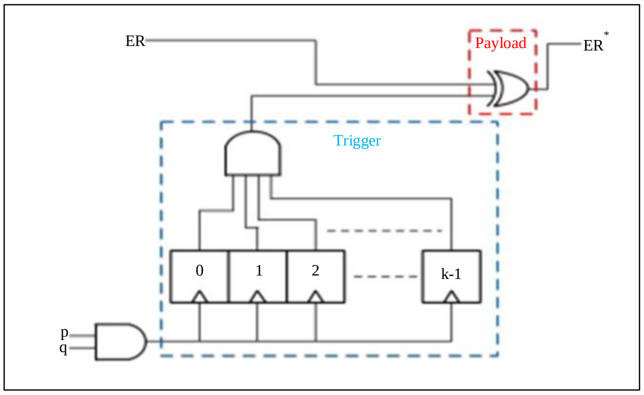


Fig. 4 Illustration of digital HT [20]

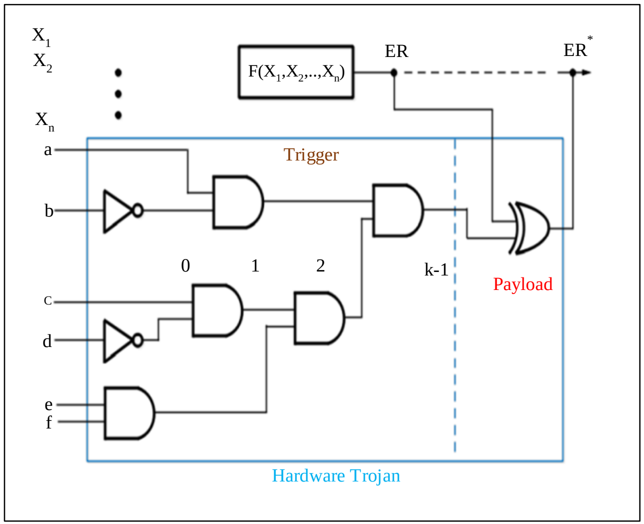


Fig. 5 Illustration of combinational HT [20]

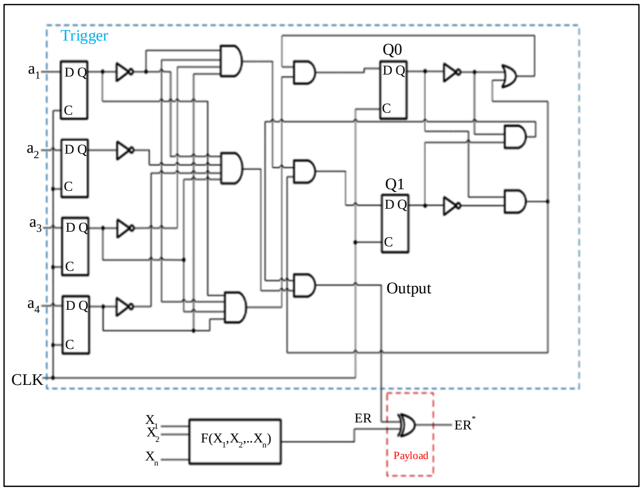


Fig. 6 Illustration of sequential HT [21]

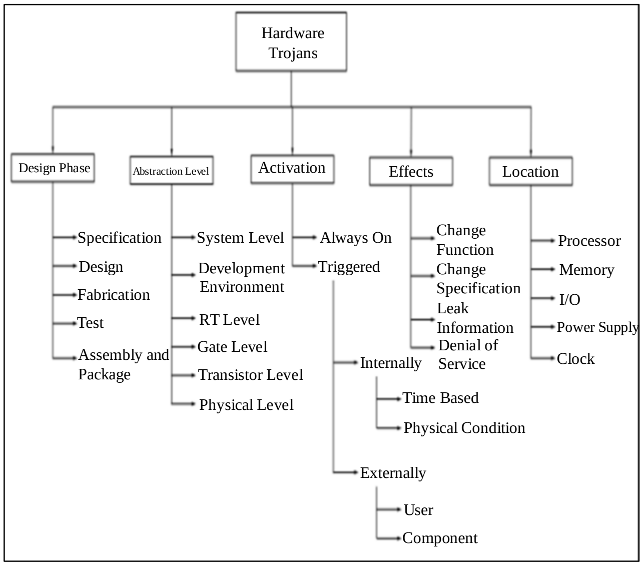


Fig. 7 The specific Hardware Trojan categorization [22]

Hardware Trojans may be completely parasitic in the CPU, memories, I/O interfaces, power source, clock grids, and other critical components [9]. The different types of hardware trojans existing at different levels of circuit design are depicted in Figure 7.

2.3. Conventional Detection Methods for Hardware Trojans

As per [9, 10, 20, 21], there are two types of hardware trojan detection techniques: Destructive techniques and Non-destructive techniques. IC splitting may be used to accomplish destructive techniques [23]. The fundamental aim is to examine high-resolution images using a scan electron microscope. The corresponding images are, therefore, merged to generate the netlist file. Non-destructive approaches are divided into two categories: logic testing as well as side-channel evaluation [24, 25]. A logic test has been employed to identify Hardware Trojan circuitry instances using the IC function test. The side-channel study analyzes the power source, route latency, and frequency spectrum of a side channel. If there is Hardware Trojan within IC, these variables may be altered.

3. Proposed Method for the Detection and Location of the Combinational Circuitry Hardware Trojans

Due to the inclusion of combinational hardware trojans, the channel containing the digital Hardware Trojan circuitry placed into the IC experiences an increment in time delay. Due to the insertion of the payload circuitry of digital Hardware Trojan, the number of gates throughout this channel has grown. Irrespective of how the Hardware Trojan has been activated, its payload circuitry simultaneously modifies the time latency status of the path within the target chip. Consequently, route delays may be exploited to identify the presence of a Hardware Trojan. The benefit entails that the presence of such Hardware Trojan may be determined without addressing its active status. In this section, a path delay-dependent Hardware Trojan detection and location technique is used to identify the presence of an HT.

This technique concentrates exclusively on the process of IC design, leaving out the manufacturing phase following tape-out. The goal of this proposed Hardware Trojan detection method is to introduce only one logical Hardware Trojan circuitry. In this technique, a latch is devised to translate any extra delay caused by Hardware Trojan circuitry. The new strategy may minimise process discrepancies and their repercussions. In other instances, designers may even eliminate the need for golden IC by employing this new technology, which allows for self-referenced detection.

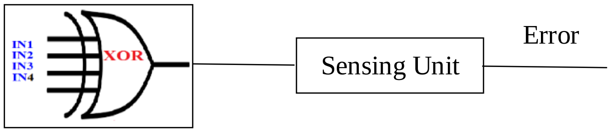


Fig. 8 Block diagram of the proposed detector

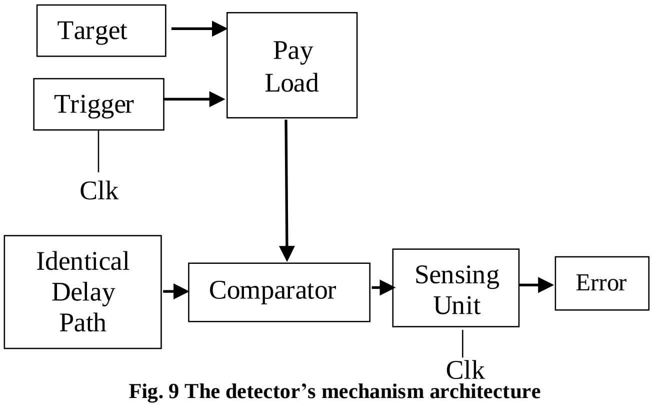


Fig. 9 The detector's mechanism architecture

As seen in Figure 8, this multi-path detector includes two components: a comparator as well as a sensing component. The comparator would be a multi-path XOR gate, while the sensing circuit relies upon SR-Latch based on NAND gates. The path that contains Hardware Trojan circuitry will lead the comparator to generate a pulse if the pathways of ICs that have a similar structure pass through it. The sensing module will detect this pulse and send an error signal to alert the user that the IC may contain a Hardware Trojan risk. Figure 9 depicts the proposed detection method configuration. In this situation, the Hardware Trojan circuitry reverses the target IC's results, like "0" to "1" / "1" to "0". Furthermore, the Hardware Trojan's payload consists of merely an XOR logic circuit.

The path delay of the output with no HT inserted in the IC is $\partial = d_{comp} + d_{sensing\ circuit}$.

This equation says that the target IC's output delay is equivalent to the original IC's delay. There are not any additional logic gates within the target IC circuitry because there is not any Hardware Trojan circuitry with in target IC.

The path delay of the output with HT inserted in the IC is $\partial = d_{comp} + d_{sensing\ circuit} + d_{payload} + d_{trigger.}$

This investigation demonstrates that the presence of a hardware trojan circuit in the targeted circuit leads to an increase in the output delay of the target IC, regardless of whether its triggering mechanism is enabled or not.

3.1. Sensing Unit

SR NAND-based Latch accepts CLK input and then evaluates the reference path with the insertion path of the Hardware Trojan. To demonstrate the insertion signal of Hardware Trojan by generating an obvious pulse, an XOR logic gate with an error signal has been connected to Q & CLK. Figure 10 depicts the improved sensing device.

3.2. Experimental Verification of the Presented Detector

The framework of the test to validate the proposed detector is shown in Figure 11. Hardware The Trojan circuitry has been arbitrarily introduced into one of the units/modules of the targeted circuit, which features a 4-bit ripple Carry adder circuit, as shown in Figure 11.

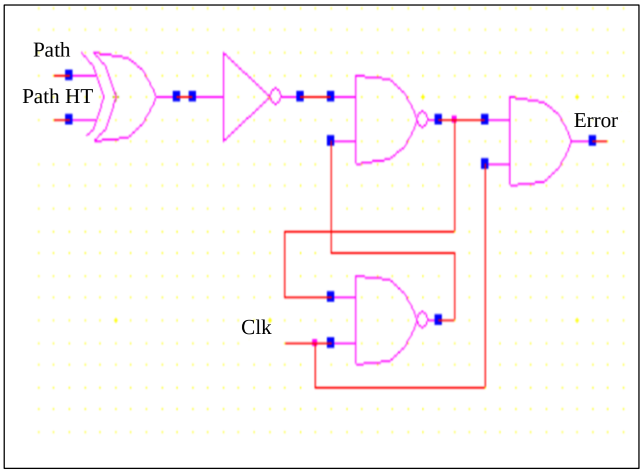


Fig. 10 Structure of the SR latch-based sensing unit

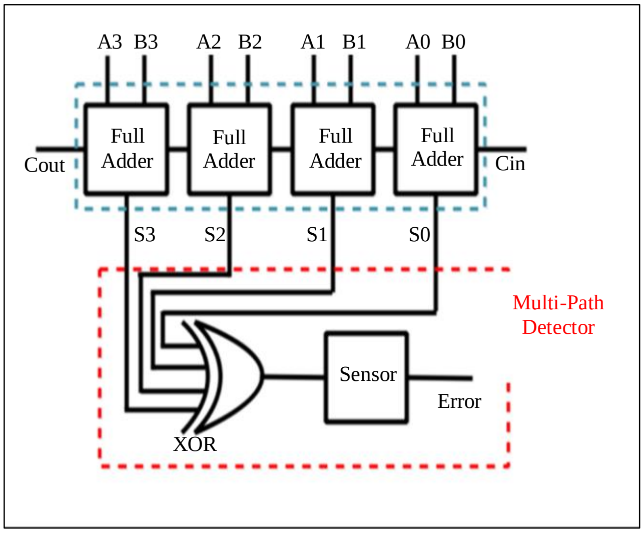


Fig. 11 Block diagram of multi-path detector

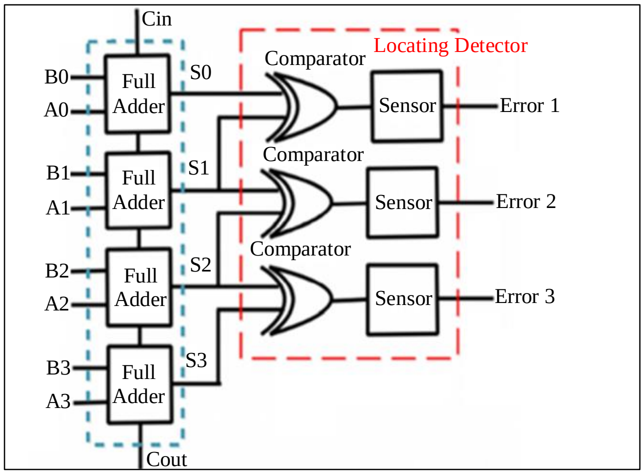


Fig. 12 Experimentation to verify the locating detector

The framework for experimental verification to validate the locating detector is illustrated in Figure 12. The output of the SR latch-based sensor is modified from “0” to “1” when any of the 1-bit adder units are arbitrarily inserted using a Hardware Trojan circuitry. Since the injected full adder unit contains an additional XOR logic gate from the Hardware Trojan circuit’s payload, overall route latency is longer than that of other conventional full adder units. The configuration of the locating detector provides the comparator with two input signals corresponding to the delays of the two neighbouring design units. Nevertheless, the comparator could communicate the output to the sensors, which would be able to identify the Hardware Trojan insertion depending on the sensor’s output if any of the system elements include Hardware Trojan circuitry.

When the 1-bit adder unit comprising the input signals A0 and B0 comprises hardware trojan circuitry, the sensor Error1 detects its relevant path S0, signalling the outcome “1,” whereas the other full adder units do not have HT circuitry; subsequently, the outcomes of sensor Error 2 and Error 3 remain “0”. The outcome of finding the detector becomes “100” as a result of this. Path S0 in the circuit having a 1-bit adder component is carrying a Hardware Trojan circuitry is indeed the location of a Hardware Trojan circuitry. Similarly, when the 1-bit adder unit comprising the input signals A1 and B1 comprises hardware trojan circuitry, the sensor Error1, Error 2 detects its relevant path S1, signalling the outcome “11,” whereas the other full adder units do not have HT circuitry; subsequently, the outcomes of the detector becomes “110”. When the 1-bit adder unit comprising the input signals A2 and B2, comprise hardware trojan circuitry, the sensor Error 2 and Error 3 detects its relevant path S2, signalling the outcome “11.” In contrast, the other full adder units do not have HT circuitry. Subsequently, the outcomes of sensor Error 2 and Error 3 remain “011”.

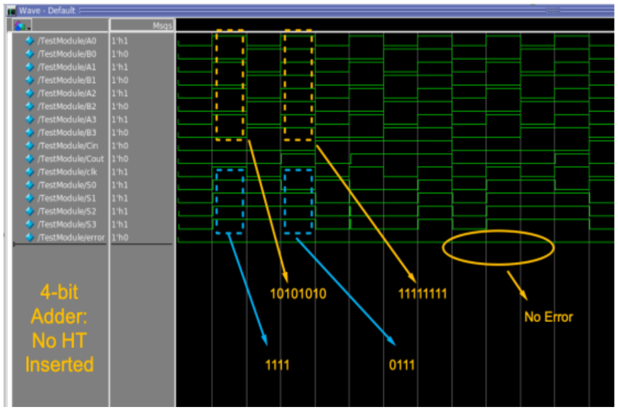
When the 1-bit adder unit comprising the input signals A3 and B3 comprises Hardware Trojan circuitry, the sensor Error3 detects its relevant path S3, signalling the outcome “1,” whereas the other full adder units do not have HT circuitry; subsequently, the outcome of finding the detector becomes “001”.

4. Simulation Results

The 4-bit RCA without Hardware Trojan and with Hardware Trojan has been modelled in Verilog HDL and has been simulated using Xilinx Vivado. The same designs of 4-RCA have been implemented on Spartan 6 (6slx45tfgg484-3) FPGA. The Hardware Trojan has been injected in the least significant 1-bit adder. With the simulation the following parameters in Table 1 have been determined for Ripple Carry Adder (RCA) with and without HT. The performance of the 4-bit RCA with Hardware Trojan has suffered a lot when compared with RCA without HT.

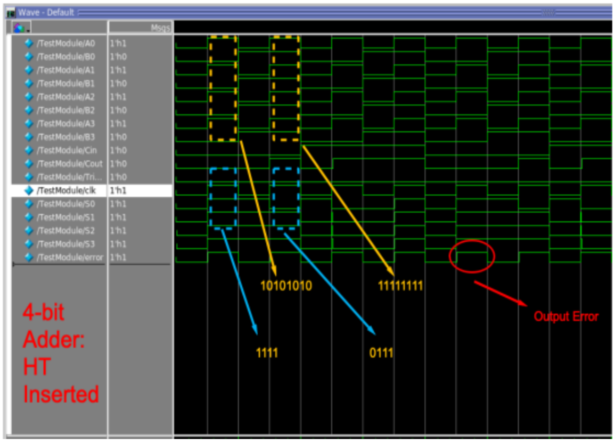
Table 1. Performance comparison of 4-bit RCA with and without HT

	Delay (ns)	**Number of LUTs**	**Power Usage**
RCA without Hardware Trojan	6.494	6 out of 27288	0.0361W
RCA with Hardware Trojan	6.893	7 out of 27288	0.0368W



```
Wave - Default
                        Msgs
/TestModule/A0          1'h1
/TestModule/B0          1'h0
/TestModule/A1          1'h1
/TestModule/B1          1'h0
/TestModule/A2          1'h1
/TestModule/B2          1'h0
/TestModule/A3          1'h1
/TestModule/B3          1'h0
/TestModule/Cin         1'h0
/TestModule/Cout        1'h0
/TestModule/clk         1'h1
/TestModule/S0          1'h1
/TestModule/S1          1'h1
/TestModule/S2          1'h1
/TestModule/S3          1'h1
/TestModule/error       1'h0
```

Fig. 13 Simulation of the original 4-bit full adder without HT



```
Wave - Default
                        Msgs
/TestModule/A0          1'h1
/TestModule/B0          1'h0
/TestModule/A1          1'h1
/TestModule/B1          1'h0
/TestModule/A2          1'h1
/TestModule/B2          1'h0
/TestModule/A3          1'h1
/TestModule/B3          1'h0
/TestModule/Cin         1'h0
/TestModule/Cout        1'h0
/TestModule/Tri...      1'h0
/TestModule/clk         1'h1
/TestModule/S0          1'h1
/TestModule/S1          1'h1
/TestModule/S2          1'h1
/TestModule/S3          1'h1
/TestModule/error       1'h1
```

Fig. 14 Simulation of the 4-bit full adder with HT

When analysing the experiment, the multi-path detector and locator may detect and also locate the HT circuitry without taking the active status of the HT into account. It also eliminates the need for a golden design model to perform the comparisons, saving up on the IC design process. However, the detector merely provides a caution about the Hardware Trojan insertion. This might examine the detection and location of introducing Hardware Trojan circuitry within an IC prototype design.

Figure 13 depicts the existing RCA's output simulation, whereas Figure 14 depicts the targeted RCA's output simulation. The yellow arrows represent the RCA's inputs, while the blue arrows represent the RCA's outputs. The simulation results demonstrate that the targeted RCA produces the very same output outcomes as the existing RCA.

Nonetheless, the detector detects an error (red arrow) within the targeted RCA, indicating that it has Hardware Trojan circuitry. Thus, the path would increase the delay from the hardwAre Trojan payload whenever one of the adder elements is inserted with a Hardware Trojan circuit. As such, the detector might ascertain the path delay.

5. Conclusion

This paper proposes a unique Hardware Trojan detection technique for mitigating attacks involving adding Hardware Trojan circuitry into IC. This HT detection method is dependent on a multi-path detection and location mechanism. By using this detection method with a unique Hardware Trojan detection methodology, a higher efficient detection of Hardware Trojan can be achieved while also safeguarding the security of IC design.

As a result, this identifying method may effectively prevent individuals from inserting Hardware Trojan circuits into the other finished units. This could prevent attackers from doing damaging activities in the background during the IC design phase. To demonstrate this technique, a 4-bit RCA is designed and implemented to detect the HTs.

References

[1] Thomas Lengaue, *Combinatorial Algorithms for Integrated Circuit Layout*, 1st ed., Springer Science & Business Media, 1990. [CrossRef] [Google Scholar] [Publisher Link]

[2] Se-Hwa Wu, *The Dynamic Cooperation between Government and Enterprise: The Development of Taiwan's Integrated Circuit Industry*, 1st ed., Routledge, 1992. [Google Scholar] [Publisher Link]

[3] Cherry Bhargava, and Gaurav Mani Khanal, *IC Fabrication Technology*, 1st ed., River Publishers, pp. 1-22, 2020. [Google Scholar] [Publisher Link]

[4] Swarup Bhunia, and Mark M. Tehranipoor, *The Hardware Trojan War, Attacks, Myths, and Defenses*, 1st ed., Springer Cham, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[5] Vivek Venugopalan, and Cameron D. Patterson, "Surveying the Hardware Trojan Threat Landscape for the Internet-of-Things," *Journal of Hardware and Systems Security*, vol. 2, pp. 131-141, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[6] Travis Boraten, and Avinash Kodi, "Mitigation of Hardware Trojan Based Denial-of-Service Attack for Secure NOCs," *Journal of Parallel and Distributed Computing*, vol. 111, pp. 24-38, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[7] Anshuman Tripathi, “The Economics of Hardware Trojans: An Expert’s Opinion,” *Journal of Information Technology Case and Application Research*, vol. 22, no. 3, pp. 159-174, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[8] William D. Toronto, “Fake News and Kill-Switches: The US Government’s Fight to Respond to and Prevent Fake News,” *Air Force Law Review*, no. 79, 2018. [Google Scholar] [Publisher Link]

[9] Konstantinos G. Liakos et al., “Conventional and Machine Learning Approaches as Countermeasures against Hardware Trojan Attacks,” *Microprocessors and Microsystems*, vol. 79, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[10] Ashkan Vakil et al., “Learning Assisted Side Channel Delay Analysis for Hardware Trojan Detection,” *2020 21st International Symposium on Quality Electronic Design (ISQED)*, Santa Clara, USA, pp. 40-45, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[11] Samaneh Ghandali et al., “Side-Channel Hardware Trojan for Provably-Secure SCA-Protected Implementations,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 6, pp. 1435-1448, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Yidong Yuan et al., “Process Variation-Resistant Golden-Free Hardware Trojan Detection through a Power Side Channel,” *Security and Communication Networks*, vol. 2021, pp. 1-15, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Jiaji He et al., “Runtime Trust Evaluation and Hardware Trojan Detection Using On-Chip EM Sensors,” *2020 57th ACM/IEEE Design Automation Conference (DAC)*, San Francisco, USA, pp. 1-6, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] Kento Hasegawa, Masao Yanagisawa, and Nozomu Togawa, “Trojan-Feature Extraction at Gate-Level Netlists and Its Application to Hardware-Trojan Detection Using Random Forest Classifier,” *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, Baltimore, USA, pp. 1-4, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[15] Krishnendu Guha et al., “Dynamic Power-Aware Scheduling of Real-Time Tasks for FPGA-Based Cyber Physical Systems against Power Draining Hardware Trojan Attacks,” *The Journal of Supercomputing*, vol. 76, pp. 8972-9009, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[16] Joseph Clements, and Yingjie Lao, “Hardware Trojan Design on Neural Networks,” *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, Sapporo, Japan, pp. 1-5, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[17] Wei Hu et al., “Leveraging Unspecified Functionality in Obfuscated Hardware for Trojan and Fault Attacks,” *2019 Asian Hardware Oriented Security and Trust Symposium (AsianHOST)*, Xi’an, China, pp. 1-6, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[18] Lei Zhang et al., “A Hardware Trojan Detection Method Based on the Electromagnetic Leakage,” *China Communications*, vol. 16, no. 12, pp. 100-110, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[19] Samaneh Ghandali, Daniel Holcomb, and Christof Paar, “Temperature-Based Hardware Trojan for Ring-Oscillator-Based TRNGs,” *arXiv*, pp. 1-7, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[20] Ajat Subhra Chakraborty, Seetharam Narasimhan, and Swarup Bhunia, “Hardware Trojan: Threats and Emerging Solutions,” *2019 IEEE International High Level Design Validation and Test Workshop*, San Francisco, USA, pp. 166-171, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[21] Tamzidul Hoque et al., “Golden-Free Hardware Trojan Detection with High Sensitivity under Process Noise,” *Journal of Electronic Testing*, vol. 33, pp. 107-124, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[22] Mingfu Xue et al., “Ten Years of Hardware Trojans: A Survey from the Attacker’s Perspective,” *IET Computers & Digital Techniques*, vol. 14, no. 6, pp. 231-246, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[23] Yajun Yang et al., “How Secure is Split Manufacturing in Preventing Hardware Trojan,” *ACM Transactions on Design Automation of Electronic Systems*, vol. 25, no. 2, pp. 1-23, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[24] Zhixin Pan, and Prabhat Mishra, “Automated Test Generation for Hardware Trojan Detection Using Reinforcement Learning,” *2021 26th Asia and South Pacific Design Automation Conference (ASP-DAC)*, Tokyo, Japan, pp. 408-413, 2021. [Google Scholar] [Publisher Link]

[25] Yangdi Lyu, and Prabhat Mishra, “Efficient Test Generation for Trojan Detection Using Side Channel Analysis,” *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, Florence, Italy, pp. 408-413, 2019. [CrossRef] [Google Scholar] [Publisher Link]