Hardware Trojan Detection

Ghalia Alluhaib, Hanan Aldissi, Rasha Alqarni, Shoroq Banafee, Wafaa Nagro, and Asia Aljahdali

University of Jeddah, College of Computer Science and Engineering, Saudi Arabia

Summary
In this study we present, overview structure of Integrated Circuit (IC), Trojans design and taxonomy that gives a primary step in higher understanding existing and probable threats and the most common techniques for Trojan detection against Hardware Trojan threats. As well countermeasures for hardware Trojan insertion to verify the trustworthiness of the manufactured ICs. Recently security of Integrated Circuits is exposed to hardware Trojans that emerged as a serious security threat, which are malicious alteration to the original circuit either during design or fabrication time. An attacker can easily add Hardware Trojan into Integrated Circuits. Since hardware Trojans are tiny and invisible, their detection is hard. Probably cause disaster effects (Denial of Service, sensitive information leakage from inside a chip-e.g., the key in a cryptographic chip, during field operation. etc.). Especially for those used in susceptible applications such as military or medical. Based on previous researches in Hardware Trojan papers, we conclude the importance of insurance the trustiness of Integrated Circuit (i.e.-protected against Hardware Trojans), where different methods have been proposed to void Hardware Trojans such as Optical Inspection, Side-Channel Analysis (SCA), Run Time Detection Techniques and Logic Test Techniques. The main result of this paper is to exhibit the mostly techniques used for detecting the Hardware Trojans to overcome from spread out the infected integrated circuit into market and reduce the rate of loose and disclosure of critical information.

Key words:
Hardware trojan, integrated circuit, threats, side channel analysis.

1. Introduction

A hardware Trojan (HT) is a new type of hardware attack, that causes changes to the intended function of ICs or force them to perform additional malicious functions. They are generated by an attacker and are extraordinarily difficult to observe. Hardware Trojans try to bypass or disable the security fence of a system in order to destroy the system or leak secret information and cryptographic keys to the attacker. Trojans can be hidden in the electronic components of ICs, field programmable gate arrays (FPGA), system-on-chips (SoC), application-specific integrated circuits (ASIC), and third-party intellectual property (3PIP). Multiple hardware Trojans have been designed and their effects have been realized. Concerns about hardware Trojans have been expressed widely, and it is thought that more advanced hardware Trojans will be developed in the future. Subsequently, recognition of Hardware Trojan threats and countermeasures have been achieved globally. The rest of this paper is organized as follows: In Section 1, we present an overview of the integrated circuit. In Section 2, we present Trojan design and taxonomy. In Section 3, we present hardware Trojan detection techniques. In Section 4, we present the threat that hardware Trojans pose. In Section 5, we present case studies of hardware Trojans. Finally, in Section 6, we conclude this paper with countermeasures to hardware Trojan attacks.

2. The Integrated Circuit (IC)

The integrated circuit (IC) consists of microscopic arrays of electronic circuits and other components, such as capacitors, resistors, diodes, and transistors, on the surface of a silicon chip, all working together to perform a particular function or a series of functions. The term “integrated” is used because all of an IC’s components, circuits, and substrate materials are manufactured from a single piece of silicon. The individual ICs are used as the building blocks of the digital electronic circuitry. We can use the terms “semiconductor” or “chip” when referring to an IC. ICs vary in complication level, from simple logic modules to complex microcomputers that contain very large number of circuits and components. ICs are exposed to many threats, the most current threats to the security properties of ICs are:

- Threats to authenticity: the ability to copy an IC’s IP for exploitation and particularly, counterfeiting, by untrusted IC fabricators.
- Threats to data confidentiality: Reverse engineering to extract intellectual property (IP) or discover sensitive data, such as cryptographic keys, contained in on-chip memory.
- Threats to integrity and trustworthiness: Tampering to sabotage IC operation or insert malicious functions, such as Trojan attacks, the focus of our research [1].

Manuscript received March 5, 2020
Manuscript revised March 20, 2020

3. Trojan Design and Taxonomy

Trojan insertion in the structure and function of a chip in many different forms. We are abstracting different categories according to the architecture to physical, activation, and action category, as shown in Figure 1.

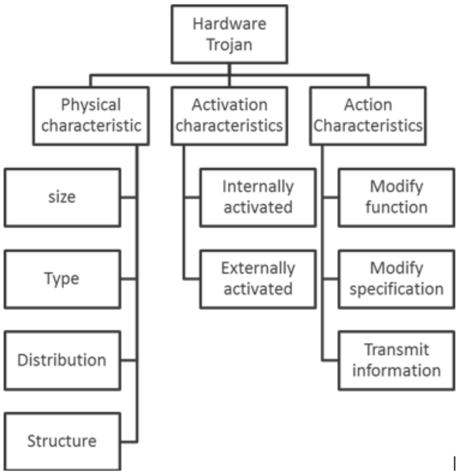

Fig. 1 Trojan Taxonomy

3.1 Physical characteristics

This category represents several hardware aspects of Trojans. The Type of the Trojan can be divided into functional or parametric. The first type (Functional Trojans) appear through add/delete transistors or gates in design of the original. The second type (Parametric Trojans) revealed by modification of wires and logic that effect the reliability of the chip. The Size of a hardware Trojan is another physical characteristic that the attacker has to consider. Size, in this case, refers to the number of elements that have been added, deleted or compromised. During the activation, the size of an HD can be significant factor; the activation of a larger Trojan has a lower probability than a smaller Trojan. The Distribution of a Trojan describes the layout of the Trojan components within the chip. An example of loose distribution is when the attacker distributes a large hardware Trojan that consists of many components placed where they can execute their payload according to determined function. A tight distribution would be when a small hardware Trojan with a few localized components is occupying only a small part of the layout. Structure is important as well. Trojans can be easy detected if the adversary is forced to insert Trojan through reconstruct the chip's layout, where changes can happen in the chip's physical dimensions that affect the delay and power characteristics of the chip [2–4].

3.2. Activation characteristics

Activation characteristics refer to the standards that cause Trojans to be active in their disruptive functions. Trojan activation characteristics have two main classifications: Internally activated and Externally activated. There are two categories of internally activated Trojans: "Always on" and "Condition-based". "Always on" means the Trojan is active and at any time can damage the function of the chip. This subclass covers Trojans that are executed by adjusting the geometry of the chip so that some nodes or paths are more susceptible to failure. The adversary may embed the Trojans on rarely exercised nodes. "Condition-based" means Trojans are inactive until the attacker identifies a specific condition or cause. The externally activated category has triggered Trojans externally. They will usually consist of malicious logic inside the IC utilizes by using an external sensor, such as a radio antenna. Then the attacker communicates through the compromised element, allowing them to start the Trojan. The activation condition may be based on the output of a sensor monitoring temperature, voltage, or any external condition like electromagnetic interference or humidity [2,4].

3.3. Action Characteristics

Action characteristics describe the effects of a Trojan on chip design and determine the type of destruction introduced by the Trojan. There are three main classes of action characteristics. Modify function: in this class, the Trojan changes the original function of the chip by adding, removing or bypassing existing logic to cause a failure in operations or add extraneous logic. Modify specification: in this class, the Trojan make changes in some of the parametric properties chip, for Example delay when we reduce the quantity of existing wire. Transmit information: in this class, the Trojan doesn't make a change in the operation of the device; instead, it transmits important information to an opponent [2,3].

4. Hardware Trojan Detection Techniques

Detection of hardware Trojans is much more important than detection of software Trojans, because hardware Trojans cannot be removed once inserted. Detection is used to prevent spread of the infected circuit into the market. Different techniques can be implemented to detect or prevent Trojans, according to the level of trust in each phase of IC design. Detection can be divided into destructive and nondestructive testing, depending on the type of mediation applied to the device. Destructive testing includes techniques such as optical inspection, which necessitates the active removal of layers of the chip. Nondestructive techniques can be classified as testing or run-time monitoring techniques. Testing methods can be classified as logic testing or side-channel analysis, as shown in Figure 2.

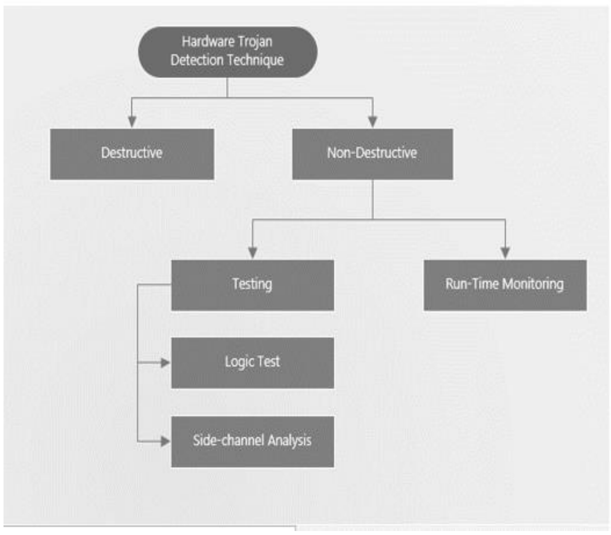

Fig. 2 Hardware Trojan Detection Technique

4.1 Optical Inspection

Optical inspection (or visual inspection) depends on reverse engineering to detect Trojans. This technique works by removing the layers of a chip’s circuitry, one by one, and comparing the layout to that of a manufactured chip. The tested chip is destroyed in the process. Examples of this technique are scanning optical microscopy (SOM), scanning electron microscopy (SEM), and picosecond imaging circuit analysis (PICA). Highly accurate and complex techniques for imaging acquisition and analysis are applied to get the die photo of the chip under test. Then comparison took a place, between the layout of the chip made by the designer and the reconstructed layout of the chip from the collected images.

Table 1: Comparison between two types of Trojan information leakage attacks.

Cyber Security Attack	**Local Physical Attack**
A malicious software in the system is the Trojan trigger.	Physically access to the hardware and trigger the Trojan.
Must remotely compromise the network and the system, then leverage from identified system weaknesses. Remote cyber security attacks are much easier than physical attacks.	Must obtain physical location of the system, just as you would for a physical attack. This is more difficult than cyber compromise.
Remote cyber security attacks are more common than local physical attacks.	Local physical attacks are less common than remote cyber security attacks.

Optical inspection is a very powerful tool, used where it is appropriate, in detecting hardware Trojans inserted during fabrication. The main obstacles for Optical Inspection technique are the expense and the time required to implement it. Because of these obstacles, optical inspection has become less favored and less accurate than other techniques [5].

4.2. Side-Channel Analysis (SCA)

One of the non-destructive techniques to detect Trojan modification in integrated circuits is side channel analysis. This method based on observing the Trojan effect in physical characteristics of a device, like dynamic power, leakage current, path delay, electromagnetic (EM) radiation, or a combination of these characteristics. Trojans affect these characteristics even when inactive, thus side-channel analysis has the advantage of not having to activate Trojans in order to detect them. This method is performed by comparing the side-channel traces from golden ICs and DUTTs (Designed Under Trojan Tests). In side-channel analysis, the designers need to deal with two main things: first, the real need of a golden model, and second, the process (PV) and environment variations that can hide the Trojan’s effects on the side channel signals. PV variations can result in alterations of circuit parameters, such as threshold voltages (Vth), channel lengths (L), and oxide thickness (Tox). For instance, Vth can fluctuate by approximately 20% of its original value in modern technologies [5]. Thus, ultra-small Trojans – sized in the order of 100 to 10,000 times smaller than the original circuit dimensions – would naturally be masked by PV. Therefore, design and test effects must be considered in order to reduce or compensate for PV effects. The side-channel technique suffers from sensitivity to error from PV and noise that can cause the infected chips to remain undetected.

Using side channel analysis for Trojan detection is limited because of two main reasons:

- The physical characteristics can be modified by factors other than the hardware Trojan.
- Some of the physical characteristics is hard to be measured.

For example, it's hard to compute the exact time in the circuit for a specific path. Also, side-channel techniques are commonly and effectively used for low complexity ICs that are not dense. Side-channel analysis is effective for large Trojan while the logic testing effective for ultra-small Trojan; however, detecting small Trojans by using side-channel is a significant challenge [5, 7].

4.3. Run-Time Detection Techniques

Among the non-destructive testing techniques is to monitor for hardware Trojans during run-time. Here, run-time monitoring is designed together with a physical countermeasure. It continuously monitors chip operation to detect the effects of malicious circuitry and also initiates mitigation techniques. The run-time monitoring technique detects the Trojan in the operation phase, bypasses it, and then operates the circuit safely. Chips are often equipped with self-destructive packaging that disables function or discards output once a Trojan is detected. Run-time monitoring technique used to evaluate logic and side-channel signals by embedding the structures in the original design. Thus, if a Trojan is activated after the deployment phase, the surveillance system triggers alert after generating a flag as an indication of the existence of a Trojan. This technique is not able to detect all kinds of Trojans and is somewhat expensive in the field of circuit area [8].

4.4. Logic Test Techniques

Logic testing is typically applied to a chip before shipping. This additional testing can be used to detect hidden hardware Trojans. Obviously, because the object to be detected is anonymous, the biggest problem during these cases is in outlining the proper set of test vectors. Several test patterns can be applied to an Integrated Circuit to detect any irregular action, but a Trojan with standard test patterns is very hard to trigger; thus, a typical Trojan has low activation probability. By dividing the IC logic structures into their functional behaviors prior to analysis, this method can detect hidden features. The functional behavior analysis method is used to detect parametric hardware Trojans (hardware Trojans added by modifying the structure of the circuit) and to detect functional errors. However, this method is not able to detect hardware Trojans that are inserted by adding or subtracting elements into a circuit (functional hardware Trojans). In functional behavior analysis method, researchers insert test vectors into the inputs of the electronic circuit and then analyzing the outputs. If the output is incompatible with the input, an abnormality is recognized. The biggest drawback with logic test functional behavior analysis methods is the large scale of the test environment within ICs, which makes the complete testing nearly impossible in large ICs. Jha proposed a method to defeat that limitation by using randomized testing. In this method, when different patterns are implemented in the input of a circuit, probabilistic fingerprints for that particular circuit are created within the outputs. When the same pattern is implemented in the examined circuit, the output result is examined for the probabilistic fingerprint. It is assumed that the circuit is infected by a hardware Trojan if there are differences in the outputted fingerprint. Jha's study was able to detect ten out of twelve modifications. Another proposal, by Chakraborty and et al., showed a new method to detect hardware Trojans. They propose a methodology for the statistical test generation and coverage de-termination of hardware Trojan. This logic testing method finds hidden features by identifying IC structure characteristics and is not very well known.

Skrobogatov and Woods perform studies on actual hardware instead of in the simulation environment. In their study, some hidden commands were detected via power analyzing. They also found that the hidden commands requested a bit block of data used as a key, and that some of the chip features, which were supposed to be inaccessible, became activated and programmable [6].

5. Hardware Trojan Threat

A hardware Trojan may lead to many harms to the system, such as leakage of information, denial of service (DoS) attacks, reduced reliability, and failure of devices.

5.1. Information Leakage

The leakage of information is caused by malicious modifications to the original design of the IP core. The information leakage hardware Trojan works as a backdoor to the system that leaks important and sensitive information to the attacker. The attacker can trigger the Trojan and steal information by using one of two methods: a local physical attack or a remote cyber security attack. In a local physical attack, the attacker will physically access to the hardware system and can trigger the Trojan. They can then use direct memory access (DMA) or bus monitoring attacks to gain confidential data. In this case, should apply an information protection scheme on hardware system that is strong enough to overcome and prevent the bad behaviors of attackers. In a remote cyber security attack, the attacker can use malicious software running in the system that uses a Trojan trigger to invoke the Trojan- infected service. The output from the hardware IP service will be treated as important information that a malicious software can compromise it and then send it to the remote attacker by a hidden communication channel [9].

5.2. Reduction of Reliability

Trojans can disrupt performance by purposely changing device characteristic or by changing the functional, interfacial or the characteristics such as energy and delay. For example, a Trojan might add buffers in the chip and therefore spend more power, which may exhaust the battery quickly.

5.3. Denial-of-Service (DoS)

Denial-of-service (DoS) hardware Trojans may causes disable, damage or modifying the settings of the device and deny the resource functions.

6. Case Study of Hardware Trojans

6.1. Information Leakage Enabled by Side Channels

Paper [10] demonstrates a class of hardware Trojans, the MOLES, which can leak secret information through side-channels. *Function:* By using the power side-channel of the IC, the attacker leaks the secret keys of a crypto core. The key is XORed with a random number created by a pseudo random number generator (PRNG), to provide an encoded signal. This encoded signal is fed into a capacitor. The energy consumption of the capacitor is related directly with the encoded signal. An attacker can measure the energy and, knowing the seed of the PRNG, can produce the secret key from the encoded signal. For a tester, these signals appear as noise, because the seed for the PRNG is not available to the tester. *Design:* The Trojan, shown in Figure 3, consists of a PRNG circuit, XOR gates, and capacitors. Each combination of an XOR gate and a capacitor is used to encode and leak a single bit of the secret key. The PRNG is implemented using a linear feedback shift register (LFSR) and only the attacker knows the seed for the LFSR. A random number is generated by the LFSR for each clock cycle. Output of the XOR is connected to the I/O pins of the IC as the I/O pins usually have the largest capacitance in an IC. *Working:* The random number is generated by a linear feedback shift register XORed with the key in every clock cycle. The capacitor can be charged or discharged, which leads to extra energy consumption based on the generated encoded values. The attacker measures this additional energy consumption and produces the leaked secret key. This Trojan can be inserted at the design phase of the chip. It does not need any triggering because this Trojan is always on, and its mission is to leak the secret key.

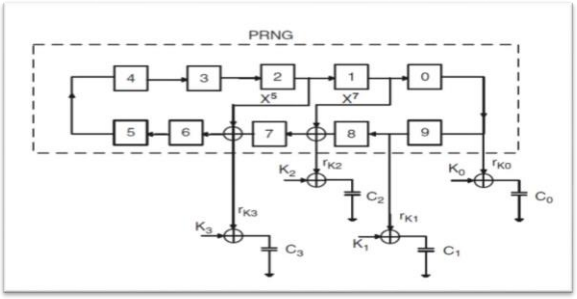

Fig. 3 Design of MOLES Trojan [10]

6.2. Information Leakage through VGA

As presented in [11], attackers use a video graphics array (VGA) display to leak secret key from the chip. The refresh rate of the VGA is changed little above or below the regular rate of refresh to set a logic “0” or logic “1”. For a normal user, in the worst case, the effects of variation in refresh rate are reflected as noise or flicker on the attached monitor, and in the best case, the variations do not cause any visually detectable effects. An attacker observes this different version in the refresh rate using an oscilloscope and produces the secret key. As in the previous case, this Trojan can be inserted at the design phase of the chip. It does not need any triggering part because this Trojan is always-on; its mission is to leak the secret key.

6.3. Denial-of-Service (DoS) Trojan

Here, once the Trojan receives the input (Practical input sequence) the clock of a chip will be freezes, resulting in a denial of service (DoS) attack on the chip in Figure 4. This Trojan consists of a sequence of XOR gates that compare the input sequence with a previous defined binary value and an OR gate. When the input of OR gate is connected to the reset input and the other input is held at logic “1” by the comparator. The output of the OR gate freezes the signal of clock at logic “1”. Once the clock signal is frozen, the chip must be reset to complete its function again or the chip will stop functioning. This

Trojan can be inserted at the design or the construction phase and can be described at gate level [9].

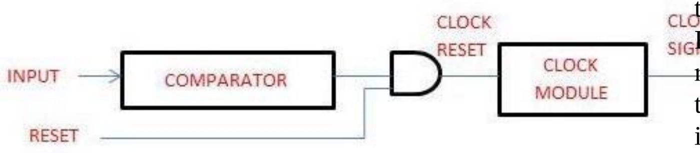

Fig. 4 Denial-of-Service by Freezing the Clock.

7. Countermeasures

As the manufacture process becomes untrusted, IC vendors are faced with two challenges: protecting the ICs from hardware Trojans and verifying the trustworthiness of the manufactured ICs. In this section, we describe the design methods that prevent Trojan. These methods can be classified into three categories: built-in self-authentication (BISA), design obfuscation scheme and logical encryption.

7.1. Built-in self-authentication (BISA)

In practice, after completing placement and routing, all unused spaces on a circuit will be filled with filler or decap cells with no functionality. The most hidden way for intelligent attackers to insert Trojans into a circuit is to replace filler cells, because deleting these nonfunctional cells has little impact on the electrical parameters. It is important to hide these filler cells to prevent an attacker from identifying and replacing them with Trojan cells. Built-in self-authentication (BISA) prevents the insertion of Trojan gates in a circuit. The principal idea is to fill all unused spaces with functional standard cells (SCs), called BISA, rather than nonfunctional filler cells. BISA cells are connected to each other to build a connective circuit that is independent from the original one. The BISA architecture can be used to test the functionality of its inserted cells: if any of the inserted cells are modified or replaced, the BISA test procedure will be able to detect it and prevent harm. If the adversary attempts to insert a Trojan by modifying or removing any cell in a BISA circuit, can be easily prevent it by the designer using a structural test [12].

7.2. Design obfuscation scheme

Obfuscation is a method that transforms a design into one that is functionally equal to the original, but which makes it much harder for an adversary to gain complete understanding of the internal logic. In this section, we describe a technique that prevent Trojan by obfuscates the state transition function and adds an obfuscated mode on top of the original functionality (called normal mode). Figure 5 shows the obfuscated functionality and the normal functionality after the original design's state transition function is obfuscated. The obfuscation method is realized by an alteration of the state transition transformation function, enabling circuit operation in two different modes: (a) the obfuscated mode when function of circuit is different from the normal functionality, and (b) the normal mode, when behavior is similar to its original version. By default, the IC is in obfuscated mode and the key (a sequence of specific input) allows switching from obfuscated mode to normal mode [13].

Figure 5 shows the obfuscation functionality and normal functionality after the state transition function of the original design is obfuscated. The mode control is performed by applying an initialization key sequence on initialization. As shown in the figure, the transition K3 is the only way the design can enter a normal mode from the obfuscated mode. Then, only one input pattern is able to guide the circuit into its normal mode. Without knowing this key sequence, attackers cannot get into the normal mode by randomly choosing input patterns. As a result of the obfuscation method, the inserted Trojans become more detectable or decrease in their ability activating in the obfuscated mode [9].

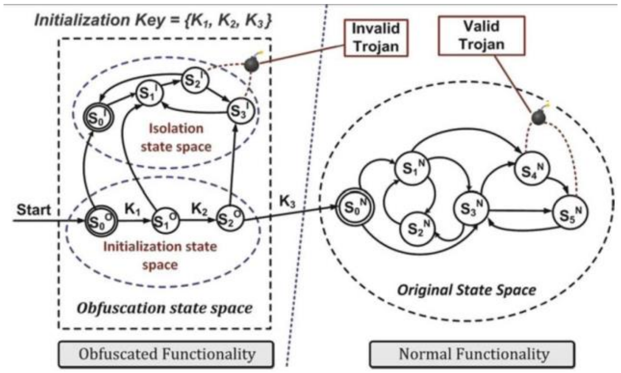

Fig. 5 Trojan Prevention by Design Obfuscation [9].

7.3. Logical Encryption

The logical encryption approach is presented in [14]. This technique allows for only authorized users to access and use the circuits, this is useful for protecting the ICs from masked theft and illegal overproduction. The functionality of a design is hidden, and an additional key is important for the proper operation of the circuit. The circuit results the correct outputs only if the valid key is uses. This is called “logic encryption.” The target is to protect ICs from masked theft and unauthorized access. When the wrong key is used, the logic gates hide the functionality of the design. This technique consists of randomly inserting XOR/XNOR gates into the design. An external key is added to the circuit so that the circuit operates correctly only if the correct key value is provided.

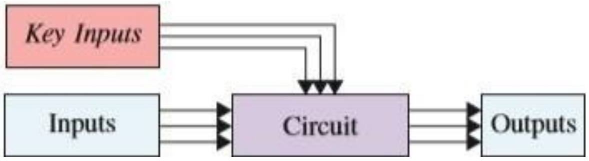

Fig. 6 Overview of Logic Encryption [15]

As shown in Figure 6, additional logic, called key inputs, is introduced to the IC and is connected to a set of newly introduced inputs and some parts of the original IC. The key inputs are connected to a tamper-proof memory and the modified IC produces the correct output only if the key inputs are set correctly. The design loads the tamper-proof memory with the correct key value, activating the IC. The activated IC is then marketed to end-use [15].

8. Conclusion

Hardware Trojan detection techniques are becoming more complex in order to enhance and improve detection rates. This makes it hard to compare the effectiveness of the different approaches. In this paper, we reviewed several techniques used to detect hardware Trojans. For each of the considered techniques, we highlighted their operating principles. We can advise of using a combination of side-channel analysis and logic testing to provide better coverage and to improve our Trojan detection methods we need to improve sensitive to power and delay. Then, we reviewed several hardware Trojan threats and showed their potential damage on the system.

References

[1] Goertzel, Karen Mercedes, and B. A. Hamilton. "Integrated circuit security threats and hardware assurance countermeasures." CrossTalk 26.6 (2013): 33-38.

[2] Karri, Ramesh, Jeyavijayan Rajendran, and Kurt Rosenfeld. "Trojan taxonomy." In Introduction to Hardware Security and Trust, pp. 325-338. Springer, New York, NY, 2012.4.

[3] Tehranipoor, Mohammad, and Farinaz Koushanfar. "A survey of hardware trojan taxonomy and detection." IEEE design & test of computers 27, no. 1 (2010): 10-25.

[4] Kutzner, Sebastian, Axel Y. Poschmann, and Marc Stöttinger. "Hardware trojan design and detection: a practical evaluation." In Proceedings of the Workshop on Embedded Systems Security, p. 1. ACM, 2013.

[5] Bhasin, Shivam, and Francesco Regazzoni. "A survey on hardware trojan detection techniques." 2015 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, 2015.

[6] Sharifi, Ehsan, et al. "Performance analysis of Hardware Trojan detection methods." International Journal of Open Information Technologies 3.5 (2015).

[7] S. Bhasin, F. Regazzoni, "A survey on hardware trojan detection techniques", 2015 IEEE International Symposium on Circuits and Systems (ISCAS), 2015.

[8] L. Lin, M. Kasper, T. Güneysu, C. Paar, W. Burleson, "Trojan Side-Channels: Lightweight Hardware Trojans through Side-Channel Engineering" in CHES, Lausanne, Switzerland:Computer Science, vol. 5747, pp. 382-395, September 2009.

[9] Karri, Ramesh, Jeyavijayan Rajendran, and Kurt Rosenfeld. "Trojan taxonomy." Introduction to Hardware Security and Trust. Springer, New York, NY, 2012. 325-338.

[10] Lin, Lang, Wayne Burleson, and Christof Paar. "MOLES: malicious off-chip leakage enabled by side-channels." Proceedings of the 2009 international conference on computer-aided design. ACM, 2009.

[11] Stefan, Deian, Christopher Mitchell, and Christian Garcia Almenar. "Trojan attacks for compromising cryptographic security in fpga encryption systems."

[12] Xiao, Kan, Domenic Forte, and Mohammed Tehranipoor. "A novel built-in self-authentication technique to prevent inserting hardware trojans." IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 33.12 (2014): 1778-1791.

[13] Bhunia, Swarup, et al. "Hardware Trojan attacks: threat analysis and countermeasures." Proceedings of the IEEE 102.8 (2014): 1229-1247.

[14] Dupuis, Sophie, et al. "A novel hardware logic encryption technique for thwarting illegal overproduction and hardware trojans." 2014 IEEE 20th International On-Line Testing Symposium (IOLTS). IEEE, 2014.

[15] Subramanyan, Pramod, Sayak Ray, and Sharad Malik. "Evaluating the security of logic encryption algorithms." 2015 IEEE International Symposium on Hardware Oriented Security and Trust (HOST). IEEE, 2015.