A Clock Sweeping Technique for Detecting Hardware Trojans Impacting Circuits Delay

Kan Xiao, Xuehui Zhang, and Mohammad Tehranipoor
University of Connecticut

Editor's notes:
Clock sweeping can be used to generate signatures for the purpose of detecting hardware Trojans. With the help of simulations and FPGA results, this article demonstrates the effectiveness of their proposed clock-sweeping technique under process variations, even for Trojans as small as a few gates.

—Dakshi Agrawal, IBM

■ **To lower the** cost of integrated circuit (IC) design and fabrication, the supply chain of the semiconductor industry has been distributed around the world. As the complexity of ICs increases, more and more highly specialized companies have become involved in the IC fabrication process to enhance efficiency and improve manufacturability; thus, providing attackers with more opportunities to make malicious inclusions and alterations. This is a serious problem for security-critical applications, such as military, transportation, and financial systems. Some Trojans can be inserted into a design if any untrusted tools or IP blocks are used. Other Trojans can be implanted by modifying the layout of the design during GDSII development and fabrication. Trojans in ICs may cause malfunctions, lower the reliability of the ICs, leak confidential information to adversaries, or even destroy the system under specifically designed conditions [1], [2]. Detecting these malicious inclusions and alterations is extremely difficult, because of the following characteristics of Trojans. First, Trojans are small compared to the designs they have altered, which makes attributes of Trojan-inserted ICs almost the same as those of Trojan-free ICs. Second, Trojans can be kept dormant during most of their operation, and be activated under very specific conditions. Third, a Trojan's behavior is unknown. Thus, it would be challenging to devise a Trojan detection technique that can target all types of Trojans.

Several Trojan detection approaches have been proposed in recent years. In general, they can be divided into two categories: full Trojan activation methods and side-channel analysis methods [1]. The first approach [3] tries to activate Trojans by applying test vectors and comparing the responses with the correct results. Because of the numerous logical states in a circuit, it is practically impossible to enumerate all states in a real design. Additionally, some Trojans may transmit information with an antenna, or modify the specification instead of changing the function of the original circuit [4]. Full Trojan activation methods may fail to detect these kinds of Trojans.

Side-channel signal analysis has been developed to detect hardware Trojans by measuring circuit parameters, such as power (transient and leakage current) and delay. The methods developed in [5]–[7]

Digital Object Identifier 10.1109/MDAT.2013.2249555
Date of publication: 07 March 2013; date of current version: 29 May 2013.

use transient power as a side-channel signal to detect Trojans, but a partial activation (i.e., generating signal transitions in a Trojan's component without fully activating the Trojan) of Trojans is still required. Moreover, the transitions for partial activation may be very difficult to generate on some hard-to-activate nets in the circuit. The authors in [8] and [9] measure path delays to detect changes caused by Trojans. Although effective, only critical paths in [8] are measured, limiting detection of Trojans inserted on noncritical paths. The authors in [9] use one additional shadow register and one comparator to measure each path delay in the circuit.

Compared to full Trojan activation and other side-channel signal techniques, a delay-based technique has a unique benefit because it does not need to activate the Trojan either partially or fully. Moreover, each path delay is relatively independent, so it is less affected by other paths of the chip, and a Trojan can potentially contribute more to a path delay change than total circuit power. Existing delay-based Trojan detection methods face the following challenges: 1) to ensure a maximum detection coverage of Trojans that can potentially be placed and distributed on various paths besides critical paths, and 2) the measurement of paths delay at a low cost. Because there is a huge number of paths in a design, any additional hardware for path delay measurement will increase the area and silicon cost significantly. Taking into account these issues in this paper, we propose a "clock sweeping" technique to obtain path delay information without any additional hardware. Transition delay fault (TDF) and path delay fault (PDF) patterns are used to obtain high coverage on the nodes of critical and noncritical paths. Once the data has been collected by clock sweeping, we generate a series of delay signatures for ICs, and then analyze whether ICs contain Trojans. Because transitions are easier generated on short paths as demonstrated in [7], the Trojans on the short paths can be detected more efficiently by power-based Trojan detection techniques [5], [6]. This is evident by the fact that nodes on short paths have generally higher controllability and observability [7].

Background

Trojan impact on path delay

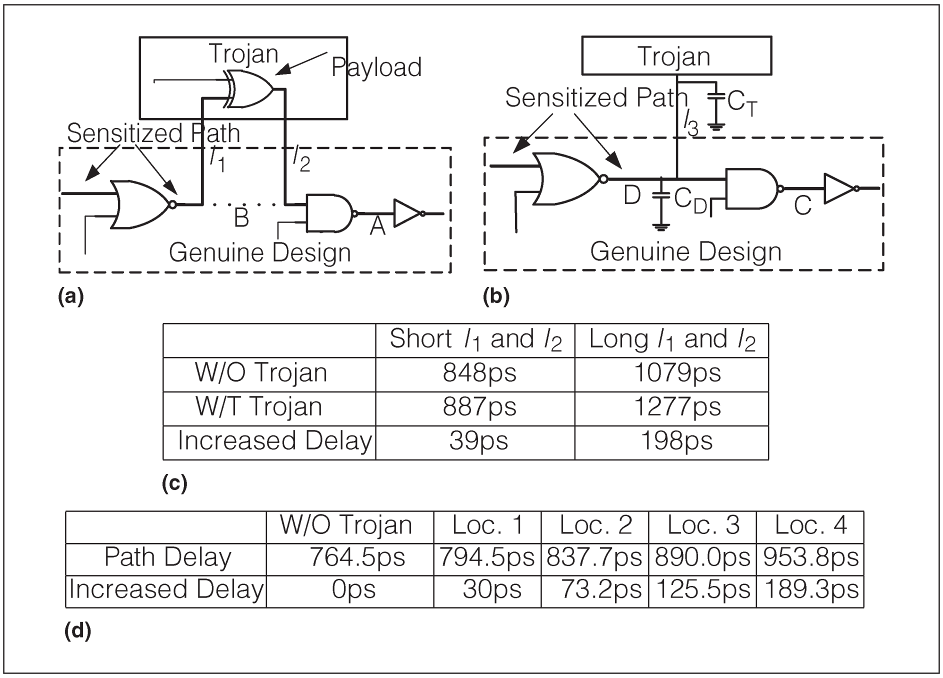
We can expect that intelligent adversaries will try to maintain the original design layout and insert Trojans into unused spaces of the layout to keep the Trojan hidden. In order to simplify this problem, we consider the nodes (outputs of gates) in the genuine IC instead of the paths for the analysis in this section. These nodes might be affected by either a trigger or a payload from a Trojan [8]. Thus, we consider three types of Trojans depending on how they are activated and their action to the functional circuit: Trojans with only payloads (TP), Trojans with only triggers (TT), and Trojans with triggers and payloads (TTP) [1].

For any Trojan trying to change the function of design, a payload gate has to be inserted at a node. An example of a TP is shown in Figure 1a. The sensitized path in the genuine design (the bold line) passes through node *B*. The additional delay consists of the propagation delay of the payload and the delay from the two wires' capacitances (l_1 and l_2). For any internally activated Trojan, the triggered parts will introduce additional interconnections which will cause unavoidable increased capacitance on the node. Besides TP and TT, TTP would have the cumulative effect of TP and TT.

In order to show the effect of TP and TT on paths delay, we performed simulations in 90 nm technology. We inserted two payload gates (minimum-sized NAND) at two positions. One is physically very close to the node (short l_1 and l_2 as in Figure 1a) and the other is remote from the node (longer l_1 and l_2). In Figure 1c, delay of path going through node *B* is measured, and the results show that a TP has increased the path delay significantly, more so for Trojans with long interconnections.

Next, we place a Trojan gate (minimum-sized NAND) at four different locations, with one input connecting to the node *D* on the sensitized path. The first location is very close to node *D* (l_3 is short as in Figure 1b), with locations 2, 3, and 4 being successively further away from node *D*. The delay of sensitized path is measured with and without Trojans for the different locations. This data is shown in Figure 1d.

The extra delay caused by TT is shown in the third row of Figure 1d. Although the increased delay is still relatively small at the location 1 (shown as Loc. 1 in Figure 1d), the TT effects at locations 2 through 4 are comparable to the effects of the payload shown in Figure 1c. Thus, as long as the standard cells in standard design style ASICs are well planned and tightly packed, the TT effect could be



	Short l_1 and l_2	Long l_1 and l_2
W/O Trojan	848ps	1079ps
W/T Trojan	887ps	1277ps
Increased Delay	39ps	198ps

	W/O Trojan	Loc. 1	Loc. 2	Loc. 3	Loc. 4
Path Delay	764.5ps	794.5ps	837.7ps	890.0ps	953.8ps
Increased Delay	0ps	30ps	73.2ps	125.5ps	189.3ps

Figure 1. (a) An example of TP. (b) An example of TT. (c) A path's delay without and with TP with short and long l_1 and l_2. (d) A path delay without and with TT at four different locations.

very obvious. Additionally, if a Trojan is activated and its payload becomes nontransparent, the required transition cannot be generated at payload gate. This faulty function indicates Trojan's existence and makes it easier to detect. *Trojans without triggers and payloads would not be detected using this technique since they most likely use an antenna to receive triggers or leak information*. They can be more effectively detected by power-based Trojan detection approaches [5], [6] because they tend to use more gates and consume more power than TP, TT, or TTP.

Clock sweeping

When using the TDF or PDF test vectors for Trojan detection, only Trojans that increase a path's delay by more than its available slack can be detected. This is unlikely to happen because an adversary can design Trojans that are hard to be detected using these patterns, by avoiding critical paths and ensuring that the delay induced by Trojan is smaller than the slack. We develop a clock sweeping technique to target shorter paths affected by Trojans without any design or silicon overhead. Clock sweeping involves applying a pattern at different clock frequencies, from a lower speed to higher speeds, which is a common practice in industry used for speed binning of parts. Some paths sensitized by the pattern which are longer than the current clock period start to fail when the clock speed increases. The obtained start-to-fail clock frequency can indicate the delays of the paths sensitized by the patterns.

For example, assume that the six paths in Figure 2a can be sensitized by test patterns. The clock period is swept from f_0 to f_5 and the sweep step size is Δt, as shown in Figure 2b. As an example, path B-D is able to propagate correct values at frequency f_0 to f_3 (pass), and will produce wrong logic values at frequency f_4 (fail). Thus, its start-to-fail clock frequency is f_4, which denotes the length of path B-D is between the frequency f_3 and f_4. When the clock is swept from low frequencies to high frequencies, paths will fail sequentially, with longer paths failing before shorter paths.

Suppose that a Trojan load is added to a path as shown in Figure 2a; the additional capacitive load will result in a small extra delay on paths A-E and B-E, which may push the arrow to the right and even fail path A-E at f_2 and path B-E at f_3. In this case, the change of start-to-fail frequency could be detected by clock sweeping. Finally, the maximum frequency applied to the circuit depends on the design characteristics, path-delay distribution in the design, and the on-chip or off-chip frequency generator's limit. Note that in today's designs, most of the paths are long or critical; this is due to the aggressive pipelining to increase circuit performance [12].

Trojan detection methodology

Signature generation procedure

Both the nodes on short and long paths have been taken into considerations in our proposed procedure which is shown in Figure 3. The following steps are performed to generate signatures for all ICs.

Test Pattern Generation: Similar to the TDF model, here we assume that TP or TT affects a node in the circuit, making the corresponding gate slow to fall or slow to rise. Thus, the TDF patterns are used in our procedure. One advantage of the TDF model is that the number of faults is linear relative to the number of nodes; thus, we can use a portion of all paths to sensitize all nodes in the circuit, and the cost of measuring paths delay could be decreased significantly. Another advantage of using the TDF pattern set is that it is widely used in industry; the ATPG processes are mature and many proposed techniques can be used to increase its fault coverage [10].

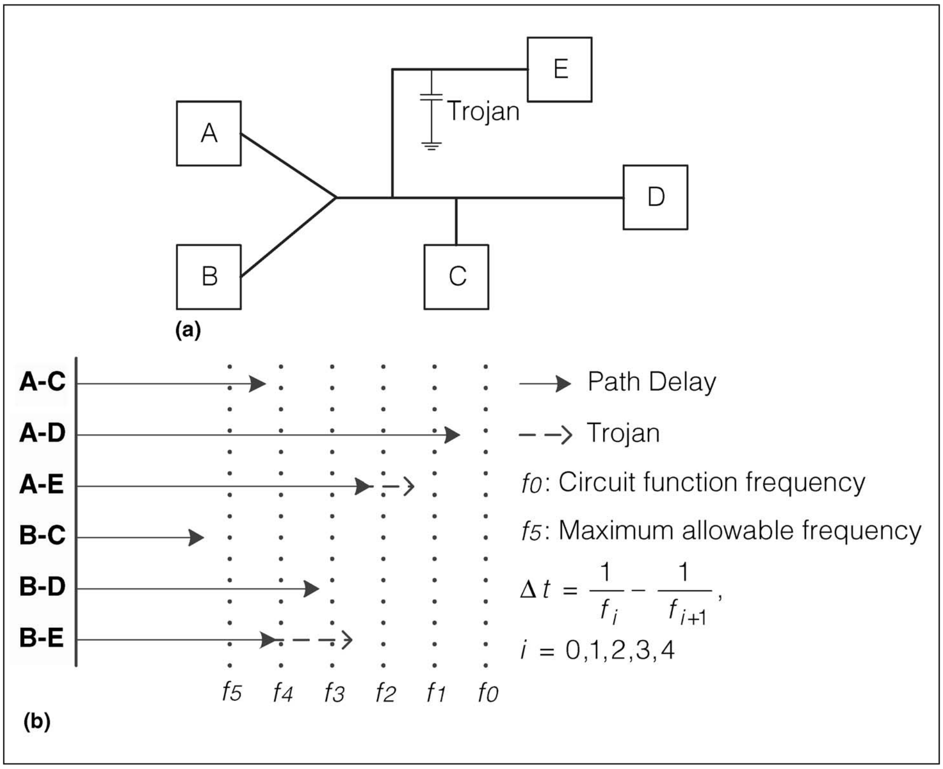

A
B
C
D
E
Trojan
(a)
A-C
A-D
A-E
B-C
B-D
B-E
f5 f4 f3 f2 f1 f0
Path Delay
Trojan
f0: Circuit function frequency
f5: Maximum allowable frequency
$\Delta t = \frac{1}{f_i} - \frac{1}{f_{i+1}}$,
$i = 0,1,2,3,4$
(b)

Figure 2. (a) An example circuit. (b) Clock sweeping.

Sweeping Frequencies and Step Size Selection: The range and step size of sweeping frequencies are two critical parameters involved in clock sweeping, because they determine the effectiveness of our Trojan detection technique. A smaller step size could be more sensitive to the small extra delay induced by Trojans. The range determines the range of long paths. Higher maximum frequency could fail more long paths during clock sweeping so that higher Trojan node coverage for delay-based Trojan detection can be achieved. The test time and data volume overhead will increase as a larger range or a smaller step is selected. Additionally, the clock sweeping range and step size are both dependent on the testing equipment.

Nodes Selection: Once the clock sweeping frequencies are determined, the sensitized path delays larger than the maximum frequency are considered as "long paths." Otherwise, they are called "short paths." For example, in Figure 2b, B-C is a short path and others are long paths; f_5 is the maximum application frequency. The delay of all the long paths can be obtained from their start-to-fail frequencies. Since the nodes on long paths can be authenticated in clock sweeping, all the patterns sensitizing these long paths will be kept. Patterns only sensitizing short paths are still useful in generating transitions for power-based Trojan detection [5], [6]. This step divides the nodes in a circuit into two groups-*nodes are authenticated either by the clock sweeping or by power-based detection techniques*. Moreover, removing patterns that sensitize only short paths could save a significant amount of test time during clock sweeping. In this paper, we focus on the right branch of the procedure as shown in Figure 3.

Clock Sweeping: This process is similar to the conventional TDF test. The only difference is that we need to apply each pattern that sensitizes at least one long path at different clock frequencies. The logic values captured by the scan flip-flops under the different clock frequencies are shifted out.

Signature Generation: By analyzing the pass and fail values, we find the start-to-fail frequency at each flip-flop for each pattern. As an example, Figure 4a presents the start-to-fail frequency for each pattern/flip-flop combination in ICs. Long-path patterns are able to sensitize different kinds of paths at

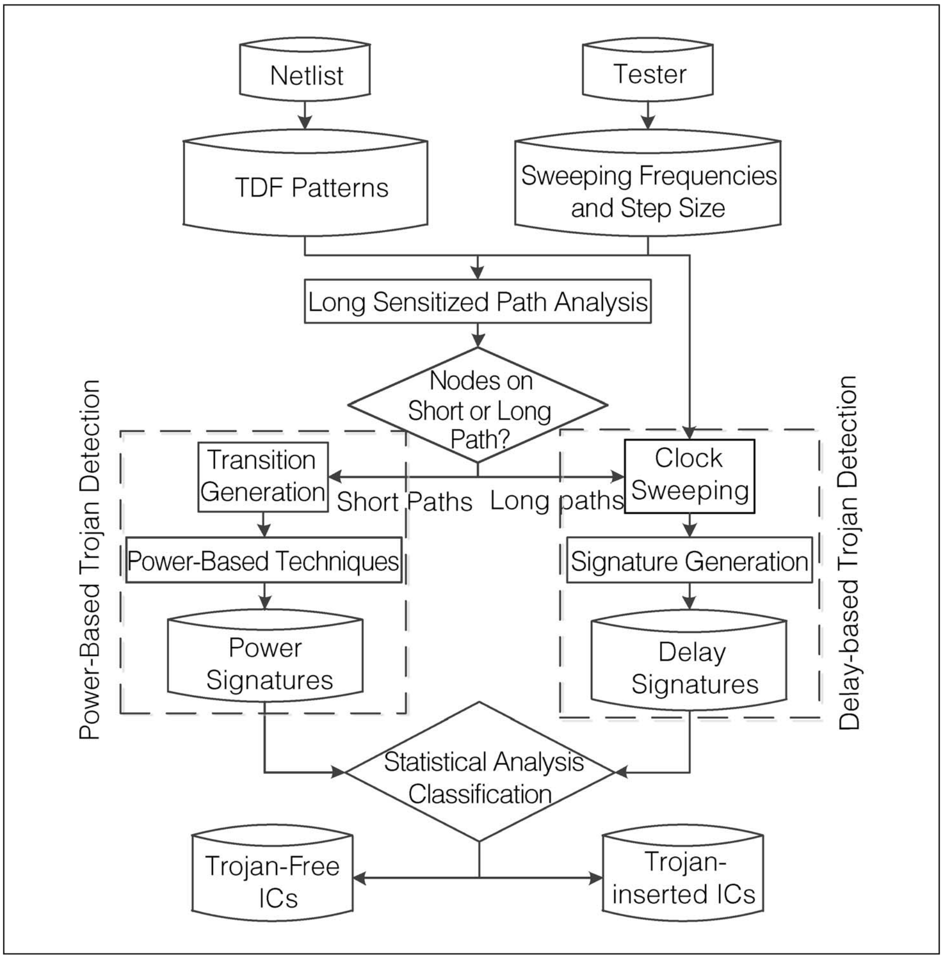


Figure 3. Our proposed signature generation procedure using clock sweeping.

different flip-flops: long, short or no path at all. For these flip-flops at which short or no path are sensitized, they always capture "passing" value during clock sweeping. These invalid pattern/flip-flop combinations need to be discarded from our signature. For example, assuming the pattern 2/flip-flop 2 (P2/FF2) is an invalid combination, the column P2/FF2 has been removed from Figure 4a. Thus, the final signature length will be shorter, as is shown in Figure 4a. Each row of the table is a signature for an IC and each column is an element of the signature. The multidimensional signature will be processed by multidimensional scaling described later.

Node coverage analysis

The objective of our technique is to recognize load capacitance induced by Trojans and capture its impact on a path. The coverage analysis in our technique can be divided into two parts: nodes on long paths and nodes on short paths.

Clock sweeping can guarantee that all sensitized long paths will fail at a particular clock frequency. Hence, the node coverage on long paths is dependent on the TDF coverage. The TDF is widely used in VLSI testing, so there have been many approaches proposed to improve the coverage of the TDF in recent years [10]. All these techniques can be applied in our procedure to achieve maximum coverage. In the meantime, a Trojan's load capacitance can slow down both rising and falling transitions. For Trojan detection, one fault, slow-to-rise or slow-to-fall, is sufficient to indicate the existence of Trojans on that node instead of two faults as in the TDF model. Therefore, node coverage will be the ratio of all detected nodes by either slow-to-rise or slow-to-fall using TDF long-path patterns to the total number of nodes in the circuits.

Figure 4b shows the estimated node coverages at different clock frequencies in benchmark s38417 for reference. ATPG patterns which are able to reach 99. 4% TDF coverage are used to sensitize all testable paths. The clock period in Figure 4b is given in the form of the percentage of the longest critical path (CP) in the circuit.

Short paths, whose delay is smaller than the maximum frequency we can apply, will never fail during clock sweeping. It is very difficult to measure short paths due to the maximum frequency limitation of the tester and the power limit of the IC.

In general, the quietest nodes which have very few transitions are usually on the long paths in a circuit [7]. In order to verify this claim, we apply random patterns to primary inputs and scan chain, and then calculate transition probability for each gate. The gates with low transition probabilities are selected to be analyzed. We choose three quiet paths and their transition probabilities at different stages are shown in the Figure 4c. As the number of stages of the path increases (1 through 7), the transition probability goes down.

Trojan gates have a higher probability to switch when they connect to nodes on short paths, which means they tend to consume more power [7]. Power-based Trojan detection [5], [6] can effectively detect Trojans on short paths, so adversaries could put Trojans on long paths to hide them. Our clock sweeping technique is able to make up for the deficiency involved with power-based Trojan detection. Increasing the test coverage in both power-based and delay-based approaches can improve the possibility to identify infected chips.

	P1/FF1	P1/FF2	. . .	P2/FF1	P2/FF3	. . .	PP_1/FFm
IC 1	f_6	f_{14}	. . .	f_3	f_{20}	. . .	f_{10}
IC 2	f_7	f_{12}	. . .	f_4	f_{21}	. . .	f_9
IC 3	f_5	f_{11}	. . .	f_3	f_{19}	. . .	f_{10}
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
IC n	f_8	f_{14}	. . .	f_4	f_{23}	. . .	f_{12}

(a)

Clock Period	>1CP	0.9CP	0.7CP	0.5CP	0.2CP	0CP
Node Coverage	0%	48.59%	61.01%	78.87%	95.18%	100%

(b)

Stage	1	2	3	4	5	6	7
Path 1	0.5	0.5	0.13	0.00059	0.00040	0.00023	
Path 2	0.5	0.29	0.28	0.063	0.012	0.0024	0.00046
Path 3	0.5	0.25	0.11	0.015	0.000296	0.000199	0.000115

(c)

Figure 4. (a) Pattern/flip-flop (Pi/FFj) combinations with start-to-fail frequencies. (b) Node coverage on different clock frequencies. (c) Transition probabilities on three quiet paths.

Statistical analysis method

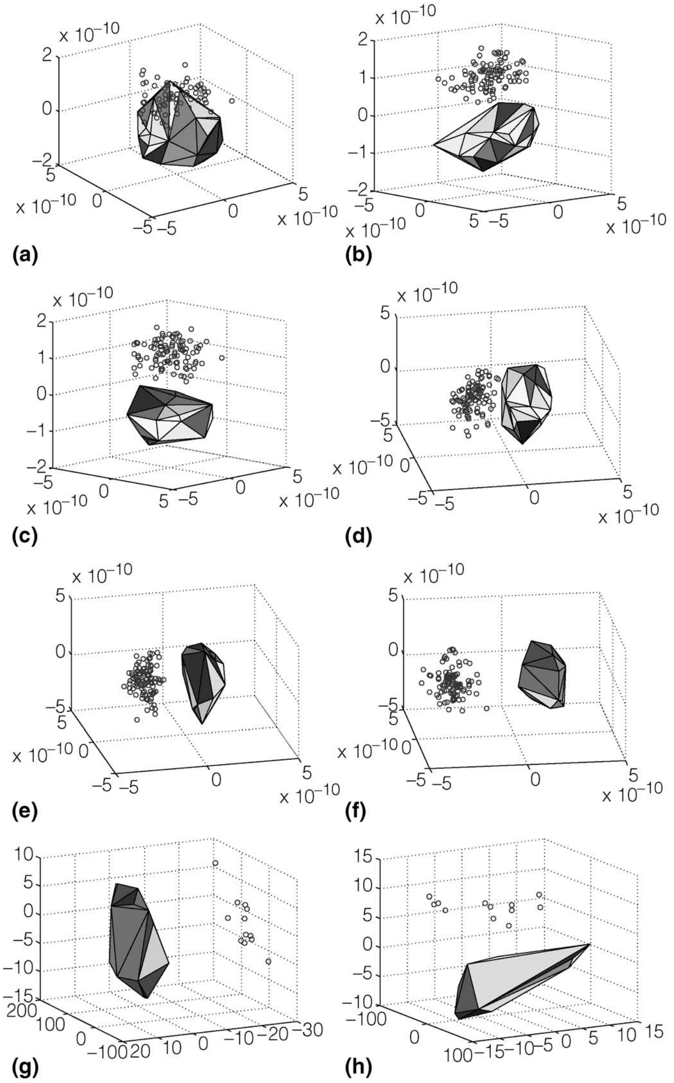
Trojan detection is extremely difficult due to process and environmental variations, especially when the Trojan is small and has very short wire connections. In order to detect Trojans, we will use a statistical analysis method to separate Trojan-free ICs and Trojan-inserted ICs.

Multidimensional Scaling (MDS) is a method for visualizing dissimilarity in data. The typical goal of MDS is to create a configuration of points in one, two, or three dimensions, whose interpoint distances represent the original dissimilarities. In a similar manner, [8] was first to use PCA/Convex for hardware Trojan detection. The different forms of MDS use different criteria to quantify dissimilarity, such as metric and nonmetric multidimensional scaling [11]. The MDS we used in our method maps the original high dimensional space to a lower dimensional space, at the same time attempts to preserve the pairwise distance and finally isolate the dissimilar chips which may carry Trojans. As described above, the signature of a chip is composed of a set of path delays. One element in a signature is considered as a dimension. Their Euclidean distances in high-dimensional space depend on the effects of both process variations and Trojans. Since outlying points in high dimensions contain Trojans and their corresponding points in low dimension are still outliers, we can use outlier points in low dimensions to predict statistical likelihood of Trojan presence. For our technique, the x-dimensional signatures for Trojan-free ICs will be mapped to a three-dimensional space by MDS, and then a convex hull will be constructed. If the signature of an IC under authentication is located outside of the convex hull, this IC is considered suspicious and may contain Trojans.

Results and analysis

Simulation results

In order to demonstrate the effectiveness of our proposed technique, the simulations were performed on an implementation of the ISCAS'89 benchmark s38417 using a 90-nm technology library. After synthesis, the s38417 benchmark circuit has 1564 flip-flops and 4046 logic gates. The layout was completed with the Synopsys physical design tool IC Compiler. The Trojan gates were inserted and routed in unused spaces in the layout by using IC Compiler. The impact of process variations on threshold voltage (V_{th}), oxide thickness (T_{ox}) and channel length (L) have been taken into considerations as



Step Size	10ps	20ps	40ps	60ps	80ps	100ps
Trojan 2	100%	65%	55%	38%	14%	9%
Trojan 3	100%	100%	100%	100%	42%	11%
Trojan 4	100%	100%	100%	100%	100%	100%
Trojan 6	100%	100%	100%	100%	100%	100%

(i)

Figure 5. Outlier analysis using MDS for Trojans 1–6 using simulation (a)–(f) and Trojans 7–8 on FPGAs (g) and (h), and step sizes analysis (i). (a) Trojan 1. (b) Trojan 2. (c) Trojan 3. (d) Trojan 4. (e) Trojan 5. (f) Trojan 6. (g) Trojan 7. (h) Trojan 8.

well. Both the interdie and intradie process variations for each of the three previously mentioned parameters are 5% in our simulations. 300 Monte Carlo Simulations, including 200 simulations for Trojan-free chips and 100 for Trojan-inserted chips, were performed at a temperature of 25 °C for each Trojan using HSPICE.

For our delay-based Trojan detection technique, we do not concern ourselves with Trojan's type or how many gates the Trojan has. Instead, we focus on how many triggers and payloads the Trojan will bring to the original design. We have inserted a large number of Trojans in this design among which we selected six Trojans to present results in details. Every Trojan is composed of a few minimum-sized gates, and these gates are placed in the nearest available unused space to keep the wire connections as short as possible. This will make the Trojan harder to detect. These six Trojans were constructed as follows: Trojan 1 has one payload and one trigger (TTP type). Trojan 2 has the same structure as Trojan 1, but is inserted at a different node. Trojan 3 has two payloads with very short connections (TP type). Trojan 4 has four triggers and no payload (TT type). Trojan 5 has three triggers and two payloads (TTP type). Trojan 6 has six triggers and four payloads (TTP type).

The maximum frequency, functional frequency and step size in our simulations are 1.5 GHz, 700 MHz, and 10 ps, respectively. By using the methodology described in Figure 3, 300 signatures for 200 Trojan-free ICs and 100 Trojan-inserted ICs are generated. After MDS processing described in Section III-D, the outlying results for Trojans 1 through 6 are shown in Figure 5a–f. The convex hull is formed using signatures of 200 Trojan-free ICs. These points are placed much closer to each other than the rest of hollow dots obtained from Trojan-inserted ICs. According to the distance between outliers and convex hull, two classes of ICs are separated. While Trojan 1's detection rate is 64% (64 out of 100 Trojan-free ICs are detected), from Trojan 2 to Trojan 6, their detection rates are all 100%. The Trojan 1 is the worst case for Trojan with payload and trigger because it has one minimum sized NAND gate which is used as a payload and only one sensitized path passes through this payload. More triggers, payloads and sensitized paths passing through Trojan nodes make detection from Trojan 2 to 6 easier. There might be some limitations associated with the MDS algorithm since it treats the signatures from the ICs as linear. As a part of our future work, we might apply different statistical classification algorithms (such as Diffusion Map and

Support Vector Machine) with the aim of further improving the detection rate. However, in our analysis (particularly for Trojans 2 to 6), MDS seems to easily classify all the chips shown in Figure 5b–f. Note that we have randomly inserted Trojans on a very large number of locations in the circuit and was able to observe similar results.

Trojan size and location analysis

Generally, larger Trojans might have more triggers and payloads, so they bring a larger impact to the original circuit. From Trojan 1 to Trojan 6, as the size of Trojans increases, these triggers and payloads affect larger number of nodes in the circuit. In other words, the larger Trojan size means that more paths, sensitized by the TDF patterns, will be impacted by the Trojan. Although Trojans' impacts may be masked by process variations, a larger number of sensitized paths always lead to higher detection possibility. Additionally, larger size Trojan most likely results in longer interconnection for triggers and payloads due to the limited unused space nearby for Trojan insertion. The extra delay induced by the larger Trojan will then increase. This can be seen in Figure 5; as we move from (a) to (f), the points also move away from the convex hull, i.e., the distance between Trojan-inserted ICs and Trojan-free ICs becomes larger. Thus, as the size of the Trojan increases, it becomes easier to separate Trojan-inserted and Trojan-free ICs by using the proposed technique.

In addition to Trojan size, the Trojan's location also has a significant influence on the results. Scan flip-flops can be considered pseudo-primary inputs and outputs. Sensitized paths spread out like a cone from scan flip-flops' outputs and converge at scan flip-flop's inputs. The nodes closer to a scan flip-flop will have more of a chance to influence these sensitized paths. Thus, these Trojans are easier to be detected. In our simulations, although Trojans 1 and 2 have same size, Trojan 2 is closer to scan flip-flops and Trojan 1 is farther away. In Figure 5, hollow dots in (b) are farther from convex hull than that in (a), which means Trojan 2 is easier to be separated than Trojan 1. Thus, we have obtained 100% detection rate for Trojan 2, while Trojan 1 only has a 64% detection rate.

Clock-sweeping step size analysis

For clock sweeping, path delay gained from simulation needs to be translated to their nearest achievable clock frequency according to the clock step size. The range and step size selection are described earlier. From the results shown in Figure 5i, we can clearly see that the detection rate reduces as the step size increases. The impact of step size becomes less for larger Trojans, because it has more triggers or payloads on sensitized paths and introduces larger extra delay.

FPGA implementation

The benchmark s9234 was implemented on 90-nm Xilinx Spartan-3E FPGAs with 145 scan flip-flops and 571 TDF patterns. Considering the limitations of the DCM, 23 different clock frequencies, sweeping from 5 ns (maximum clock frequency) to 9.4 ns (functional clock frequency) with a step size (Δt) of 200 ps, are generated by a Digital Clock Management (DCM) in the FPGA. To reduce measurement noise, each frequency was measured three times. Since the step size, limited by accuracy of the clock crystal and DCM, is larger than the predicted impact of any trigger, we only focused on TPs in the FPGA implementation.

In this experiment, two types of Trojan with payload are inserted separately in the layout by using Xilinx FPGA Editor. 44 separate FPGA boards were used, with 32 being Trojan-free and 12 being Trojan-inserted. We randomly chose 80 patterns from the 571 TDF patterns for analysis to reduce test and measurement time, so the total number of pattern/flip-flop combinations is $80 \times 145 = 11\,600$. After removing the invalid pattern/flip-flop combinations which cannot fail any path in the clock sweeping range as described above, the remaining number of pattern/flip-flop combinations is 786. These 786 pieces of delay information are used as the signature for each chip. Figure 5g and h show the scaled signature by using MDS for Trojan detection. The convex hull is drawn according to the signatures of the 32 Trojan-free FPGAs, and the 12 hollow dots represent the signatures from FPGAs with Trojans. While all hollow dots are totally separated from the convex hulls, and the detection rate is 100% for both Trojans, we note that the first Trojan is easier to separate as the distance between the hollow dots and convex hull is larger. The reason for this is that the payload gate closer to the scan flip-flops influences 79 sensitized paths, while the other Trojan only affects nine sensitized paths.

In this paper, clock sweeping is used to obtain the critical and noncritical path delay and then generate signatures for ICs for the purpose of detecting hardware Trojans. Statistical analysis methods have proved to be effective at identifying Trojan-inserted ICs in the presence of process variations, as demonstrated by both our simulation and FPGA implementation results.

References

[1] M. Tehranipoor and F. Koushanfar, ''A survey of hardware Trojan taxonomy and detection,'' *IEEE Design Test Comput.*, pp. 10–25, Jan.–Feb. 2010.

[2] R. Karri, J. Rajendran, K. Rosenfeld, and M. Tehranipoor, ''Trustworthy hardware: Identifying and classifying hardware Trojans,'' *IEEE Comput. Mag.*, vol. 43, pp. 39–462010.

[3] F. Wolff, C. Papachristou, S. Bhunia, and R. Chakraborty, ''Towards Trojan-free trusted ICs: Problem analysis and detection scheme,'' *Design, Automat. Test in Eur. (DATE)*, pp. 1362–1365, 2008.

[4] X. Wang, M. Tehranipoor, and J. Plusquellic, ''Detecting malicious inclusions in secure hardware: Challenges and solutions,'' in *Proc. IEEE Int. Symp. Hardware-Oriented Secur. Trust (HOST)*, 2008, pp. 15–19.

[5] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, ''Trojan detection using IC fingerprinting,'' in *Proc. IEEE Symp. Secur. Privacy (SP)*, 2007, pp. 296–310.

[6] R. Rad, M. Tehranipoor, and J. Plusquellic, ''A sensitivity analysis of power signal methods for detecting hardware Trojans under real process and environmental conditions. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 12, pp. 1735–1744, 2009.

[7] H. Salmani, M. Tehranipoor, and J. Plusquellic, ''New design strategy for improving hardware Trojan detection and reducing Trojan activation time,'' in *Proc. IEEE Int. Symp. on Hardware-Oriented Secur. Trust (HOST)*, 2009, pp. 66–73.

[8] Y. Jin and Y. Makris, ''Hardware Trojan detection using path delay fingerprint,'' in *Proc. IEEE Int. Workshop on Hardware-Oriented Secur. Trust (HOST)*, 2008, pp. 51–57.

[9] J. Li and J. Lach, ''At-speed delay characterization for IC authentication and Trojan horse detection,'' in *Proc. IEEE Int. Workshop on Hardware-Oriented Secur. Trust (HOST)*, 2008, pp. 8–14.

[10] G. Xu and A. Singh, ''Achieving high transition delay fault coverage with partial DTSFF scan chains,'' in *Proc. IEEE Int. Test Conf. (ITC)*, 2007, pp. 1–9.

[11] I. Borg and P. Groenen, *Modern Multidimensional Scaling, Theory and Applications*. New York: Springer-Verlag, 1997.

[12] H. Ren, Z. Wang, W. Shi, and D. Edwards, ''Critical path analysis in data-driven asynchronous pipelines,'' in *Proc. Int. Conf. on Comput. Commun. Informat. (ICCCI)*, 2012, pp. 1–9.

Kan Xiao is currently pursuing the PhD in computer engineering from the University of Connecticut, Storrs. His research interests include hardware security and trust. He has a BS in electrical engineering from Beijing University of Posts and Telecommunications, Beijing, China. He is a student member of the IEEE.

Xuehui Zhang is a PhD student in the ECE Department at the University of Connecticut's School of Engineering. Her research interests include hardware Trojan detection in integrated circuits and IP cores, and on-chip sensor design for reliability and temperature analysis. Zhang received an MS in computer science and engineering from Beihang University, Beijing, China. She is a student member of the IEEE.

Mohammad Tehranipoor is an associate professor of electrical and computer engineering at the University of Connecticut, Storrs. His research interests include hardware security, IC trust, DFT, at-speed test, CAD and test for CMOS VLSI designs and emerging nanoscale devices. He has a PhD in electrical engineering from the University of Texas at Dallas. He is a senior member of the IEEE and a member of the ACM and ACM SIGDA.

■ Direct questions and comments about this article to Mohammad Tehranipoor, ECE Department, University of Connecticut, Storrs, CT; tehrani@engr.uconn.edu.