A LAYOUT-AWARE APPROACH FOR IMPROVING LOCALIZED SWITCHING TO DETECT HARDWARE TROJANS IN INTEGRATED CIRCUITS

*Hassan Salmani, Mohammad Tehranipoor**

ECE Department
University of Connecticut
{salmani_h,tehrani}@engr.uconn.edu

Jim Plusquellic

ECE Department
University of New Mexico
jimp@ece.unm.edu

ABSTRACT

Malicious activities and alterations to integrated circuits have raised serious concerns to government agencies and the semiconductor industry. The added functionality, known as hardware Trojan, poses major detection and isolation challenges. In this paper, we present a method to localize design switching to any specific region independent from test patterns. The new architecture allows activating any target region and keeping others quiet which reduces total circuit switching activity. This helps magnify the Trojan's contribution to the total circuit transient power by increasing Trojan-to-circuit switching activity (TCA) and power consumption. The proposed method is aimed at improving the efficiency of power-based side-channel signal analysis techniques for detecting hardware Trojans. Our simulation results demonstrate the efficiency of the method in significantly increasing TCA.

Index Terms— Hardware Security, Trojan Detection, Trojan-to-Circuit Activity, Scan-cell Reordering

1. INTRODUCTION

Design and fabrication of Integrated Circuits (ICs) are becoming increasingly vulnerable to malicious activities and alterations with globalization. These vulnerabilities have raised serious concerns regarding possible threats to many critical applications. Responsible for design or fabrication, the third party can maliciously change the design before fabrication by inserting extra logic, called a hardware Trojan. Hardware Trojans aim to fail the design in the field or transmit secret data to the adversary [1][2]. Therefore, confidence in imported products is a serious concern especially in mission-critical applications.

Having the knowledge of IC fabrication and testing, an adversary can design a Trojan that cannot be activated and detected with traditional functional and structural tests. However, side-channel signal analysis techniques over transient power have proven to be highly effective in extracting information about the internal operation of designs [3][4]. In a power-based side-channel signal analysis, it is possible to extract the Trojan signal by monitoring power pads/ports even in the presence of various types of noise, including measurement noise, ambient noise, and other random signal variations that manifest themselves during the circuit operation.

Although approaches such as circuit delay analysis, full activation of Trojans, randomization-based probabilistic analysis, and extracting design characteristics are presented for hardware Trojan detection [5][6][7][8][9][10][11], in this paper, we focus on the methods that use transient power signals and perform switching activity analysis to target hardware Trojans in integrated circuits [3][12][13][14][16]. The methods that analyze circuit transient power depend strongly on the effectiveness of patterns to magnify Trojan contribution to circuit power consumption.

Authors in [12] present a method to generate power fingerprint of genuine ICs considering various types of noise in the circuit. Random patterns are applied to IC-Under-Authentication (IUA) to generate a measurable difference between the power profiles of the genuine IC and IUA. However, its effectiveness is limited when targeting small Trojans. The proposed method in [13] is based on analyzing local transient current from power ports on the target chip. To alleviate the impact of process variations during measurement, a calibration step is performed for each IUA before actual measurement. Trojan-inserted designs are distinguished using outlier analysis. However, the capability of the method is not evaluated for small Trojan circuits in large designs. A multiple supply transient current integration method to detect hardware Trojans in IUA is presented in [14]. The current is measured locally from various power pads or controlled collapse chip connections (C4s) on the die. Random patterns are applied to increase the switching in the circuit in a test-per-clock fashion [17].

In [15], a circuit is divided into regions consisting of a number of flip-flops and other components in a specific radius from the flip-flops. Randomly generated patterns are grouped based on generating high switching activity in a region while keeping the rest of circuit at low activity. The method is not layout-aware and does not consider the distribution of switching activity across the layout of circuit in practice. In [16], the authors present a sustained vector technique. A vector is applied to the circuit, and for several clock cycles (up to 25) primary inputs are kept intact. State bits (flip-flops) supply transitions in the circuit, and after some clock cycles it is expected activities converge to a specific portion

* THIS WORK WAS SUPPORTED IN PART BY NATIONAL SCIENCE FOUNDATION GRANTS CNS-0716535 AND CNS-0844995.

of the circuit. Applying the next vector would target another portion of the circuit. Authors in [11] proposed a technique to eliminate the rare triggering conditions in circuits by adding dummy scan flip-flops to improve the controllability of the low transition probability nodes. Finally, authors in [18] present a taxonomy for Trojans and discuss the issues related to hardware Trojan detection and isolation.

To avoid easy detection, an adversary can design the Trojans to have little impact on circuit power. Developing a pattern generation strategy to localize switching and reduce the background noise (i.e. circuit noise) is an extremely challenging task. In this work, we propose a design technique that helps localize switching in the circuit without any requirement to any specific pattern set. Regional activation to limit transitions to a target region of the circuit while keeping other regions quiet would be an effective way to increase the ratio of Trojan-to-circuit power consumption. The methods proposed in [12][13][14][16] can be significantly enhanced when combined with our switching activity localization technique.

1.1. Contributions and Paper Organization

In general, scan chains provide increased controllability for the circuit-under-test [17]. It has been demonstrated that there is high correlation between switching activities in the internal nodes of a circuit and the transitions taking place in scan cells [19]. The Trojan contribution to circuit power consumption can be minimal considering the circuits' high switching activity during functional, test, or authentication modes. All the previously proposed power-based signal analysis methods to detect Trojans lack an efficient localizing transition generation strategy. In this paper, a new scan-cell reordering method is proposed to localize switching activity in an IUA. The proposed method is layout-aware and can effectively restrict switching activity within a target region.

Simulation results show that by partially activating a circuit (i.e. the target region experiences switching while other regions are kept quiet), it is possible to considerably increase the ratio of Trojan-to-circuit activity (TCA). Our method is able to increase TCA by significantly reducing circuit switching activity (also called background noise). Note that in this work we do not assume to have any knowledge of the size, type and location of Trojans in a circuit. Also, note that the adversary will not be able to evade the authentication process by using a test enable signal in the Trojan circuit since the test enable signal must switch between test mode and functional mode. Therefore, our method is able to deal with such attempts and effectively target Trojans even if the Trojan is only active during functional mode.

The paper is organized as follows. The layout-aware scan-cell reordering method is presented in Section 2. Simulation results are presented in Section 3. Finally, Section 4 presents the concluding remarks.

2. SCAN CELL REORDERING

In general, transitions in a circuit are mainly caused by transitions on primary inputs and scan flip-flops. In large designs, the primary inputs' ability to cause switching is restricted to the first levels of the circuit. However, the scan architecture allows access to internal cells of the circuits. Furthermore, the total power consumption of the circuit under test (CUT) is highly correlated with the total number of transitions in the scan cells during scan-based testing [19][20]. During scan insertion, scan cells are grouped into a number of scan chains. They can be grouped based on different criteria.

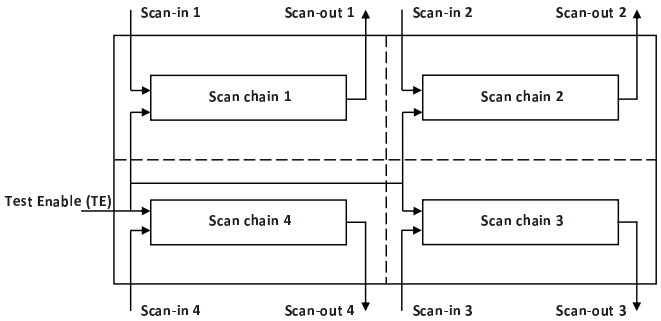

Fig. 1. Layout-aware scan-cell reordering concept.

Scan-cell reordering techniques have already been proposed to reduce power during scan test [21][22], enhance delay fault coverage [23][24], or minimize scan paths [25][26]. In this work, we also develop a scan cell reordering method but for improving Trojan detection using power-based signal analysis. In general, scan cells are scattered across the layout, and many gates can be activated at the same time across the layout during IC authentication. Reordering of scan cells based on their geometric positions can significantly restrict switching activity into a specific region.

Figure 1 shows the basic concept of layout-aware scan-cell reordering. Assume that the design with four scan chains is divided into four regions. The method forms the scan chains such that scan cells placed in each selected region are connected to each other. It is to ensure that the scan chains have the same length, but that is not a requirement. The technique enables magnifying the Trojan's impact by increasing the Trojan-to-circuit power consumption ratio, by maximizing switching in the target region (e.g. the region containing scan chain 4) while minimizing switching in all the other regions (1, 2, and 3).

As an example, Figure 2 shows the organization of scan chains in a small ISCAS'89 benchmark s838, where 32 scan cells are grouped into four scan chains using the Synopsys Design Compiler [27]. The figure shows that scan chains are scattered across the layout, and the entire design is subjected to dispersed transitions using each of scan chains. Our proposed procedure groups scan cells based on their final physical location in the layout. Due to the lack of placement information at the front-end phase during scan chain insertion, it is not possible to group scan cells and arrange scan chains based on geometric information. Therefore, we perform layout-aware scan-cell reordering after placement and before routing.

The reordering procedure obtains placement information of scan cells and re-stitches scan chains. Although the basic idea is applicable to any design environment, here, the procedure is

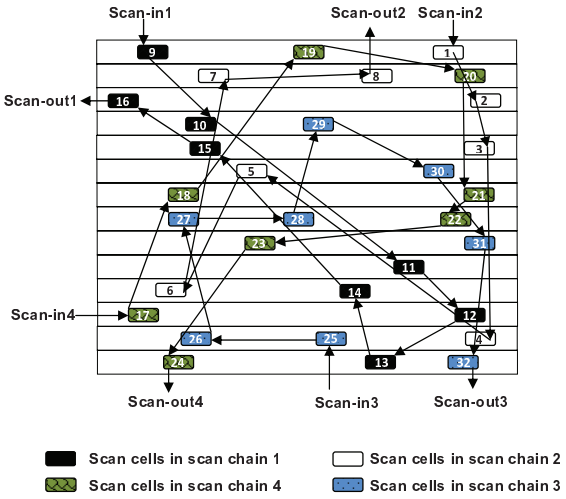

Fig. 2. Traditional scan chain the organization in the s383 benchmark.

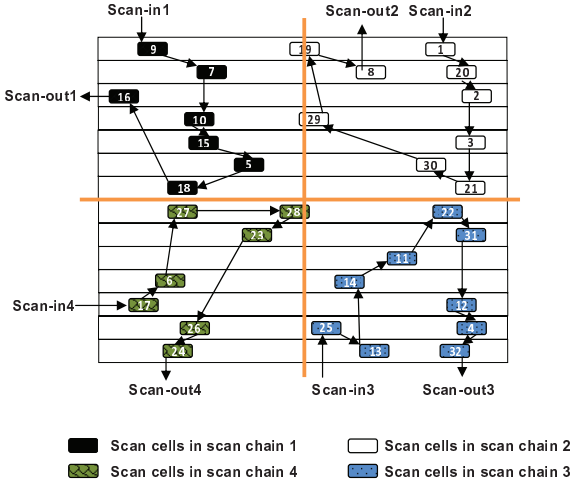

Fig. 3. Scan chain organization in the s383 benchmark using our layout-aware scan-cell reordering.

implemented using Synopsys Astro [27]. First, the placement information of scan cells is extracted. Then, the available connections between scan cells are removed. In the following, scan cells are connected to each other based on the physical information and the number of regions (N). Finally, the netlist is updated with re-stitched cells to be considered for routing. The entire procedure has been automated in our simulation flow.

The number of regions, N, determines the size of the regions. With large N, each region consists of a small number of components. A small region may magnify the impact of Trojan activity on the design's power profile because there is less activity in the entire design. On the other hand, small regions may decrease transition probability of the Trojan since some of the Trojan inputs may be from regions kept inactive. In contrast, large regions can increase the probability of generating a transition in a Trojan, but the Trojan impact can be lessened due to an increase of activity in the entire design.

Trojan components can be distributed over the entire layout, and their inputs may originate from several different regions. Authentication of all combinations of regional activation for designs

Region 1	Region 2
s38417: 15.7%, 15.8%, 15.2%, 15%	s38417: 7.1%, 7.2%, 7.2%, 7%
s35932: 11.6%, 11.4%, 11.9%, 11%	s35932: 8.3%, 7.5%, 8.2%, 6.8%
Region 4	**Region 3**
s38417: 67.5%, 67.7%, 69%, 68.3%	s38417: 9.7%, 9.3%, 9.6%, 9.5%
s35932: 69.7%, 71%, 69.3%, 73%	s35932: 10.3%, 9.9%, 10.4%, 9%

Fig. 4. The percentage of switching activity in each region of s38417 and s35932 after running four simulations. The results are shown as (Run1, Run2, Run3, Run4) for each benchmark.

with large N would be time-consuming. Suppose the design is divided into N regions, then there are $\left(\binom{N}{1} + \cdots + \binom{N}{N}\right)$ combinations to be considered during authentication. However, in practice, it is not necessary to examine every combination of regions. The number of gate inputs (or fan-in) is limited by the technology library. Even the adversary may not necessarily be able to design a gate with high fan-in to reduce its activity since such a gate greatly impacts the delay characteristics of the design and can easily be detected using delay-based techniques [5][6]. This fact leads to only inspect $\left(\binom{N}{1} + \cdots + \binom{N}{I_{max}}\right)$ combinations where I_{max} is the largest fan-in in the utilized technology library.

Figure 3 shows the new organization of scan chains in s838 benchmark after performing our layout-aware scan-cell reordering. In this design, the scan cells are grouped into four regions, N=4, as if the circuit layout is divided by 4 based on the location of the cells. The effectiveness of scan-cell reordering in limiting switching activity in any target region is evaluated in larger ISCAS'89 benchmarks, s38417 with 1564 flip-flops and 4933 gates, and s35932 with 1728 flip-flops and 3926 gates. Using our layout-aware scan-cell reordering method, scan cells in both benchmarks are grouped into N=4 regions with each benchmark consisting of 48 scan chains. The simulation is run four times, and each run consists of three pattern sets. Each time different random patterns are applied. Each pattern set consists of 41 test vectors in s38417 and 46 test vectors in s35932. Patterns apply random '0' and '1' to scan chains covering the target region at the left bottom corner of the benchmarks' layouts while a '0' is applied to all other scan chains. To increase randomness, the circuit is always set to scan mode by keeping the test-enable (TE) input active; however, other cases such as using TE and the functional capture clock can be applied as well in case the adversary uses TE to deactivate the Trojan in test mode. The percentage of activity in each region of the benchmarks is reported in Figure 4 as (Run1, Run2, Run3, Run4). The results clearly indicate that in all four runs switching activities are mostly limited to the target region labeled 'Region 4' in Figure 4 while the other regions are kept fairly inactive in both benchmarks. Note that our detailed analysis demonstrated that the majority of transitions in the non-target regions take place in cells adjacent to the target region.

Regions are controlled by scan-chains; therefore, the number of scan chains may determine N. Given the limitation on the number of pins used for testing, when the number of regions is large, a compression-like architecture including a phase shifter can be used. Our technique has no impact on the pattern generation flow and fault coverage, and it does not pose area and pin overheads.

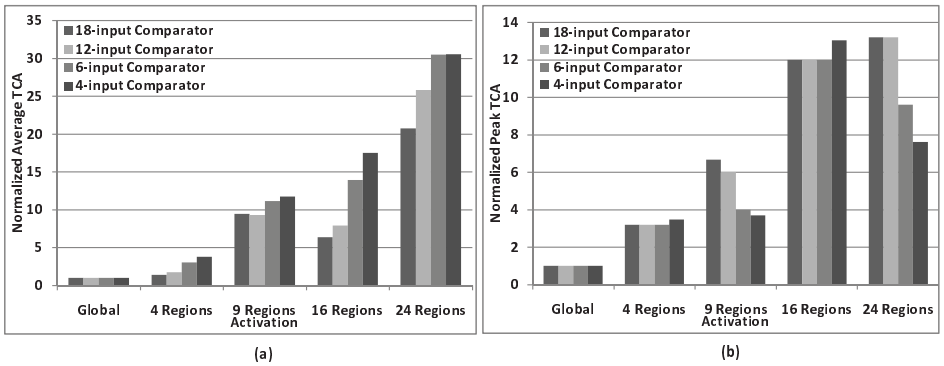

Fig. 5. Impact of localized activation on Average and Peak TCA.

3. SIMULATION RESULTS

Four layouts of the s38417 benchmark with different number of regions are generated (i.e. N =4, 9, 16, and 24.) Four combinational comparator Trojan circuits with 4 (consisting of 3 gates), 6 (consisting of 7 gates), 12 (consisting of 15 gates), and 18 (consisting of 23 gates) inputs are designed. Note that our method works as effectively for sequential Trojans, as well. The original design is synthesized using the Synopsys Design Compiler, and the layout is generated using the Synopsys Astro [27]. The scan-cell reordering procedure is applied with different Ns. After obtaining the DEF file of each new design, the circuit is updated by inserting Trojan cells into unused spaces in the circuit layout and making the required connections. Considering four Trojans and four regions, in total, 16 different designs are created. Verilog code corresponding to each design is extracted and used for simulation at the logic level [27]. The Synopsys' Verilog compiler (VCS) is used to analyze switching activity of Trojans and designs. Furthermore, the HSPICE model of Trojan-inserted designs is extracted by StarRCXT [27]. NanoSim is used to perform simulation at the circuit level [27]. Four power ports are placed at the four corners of every design during power distribution network synthesis phase, and the circuit power consumption is the superposition of current drawn from the power ports.

In this section, we use the term "*local activation*" for applying random patterns to a target region using reordered scan cells and the term "*global activation*" when applying random patterns to the design in a traditional manner (i.e. without reordered scan cells and all scan chains are activated). Simulation for each design is done by applying K =100 random vectors. A larger number of vectors may be needed for larger designs.

The Trojan's impact is evaluated by Trojan-to-circuit activity (TCA) which is defined as the ratio of the number of transitions inside Trojan to the number of transitions in the entire circuit. Figure 5 shows the impact of localization with different Ns on Trojan activity. "*Average TCA*" is defined as the ratio of the total number of transitions inside the Trojan to that of the circuit over the entire simulation time. In Figure 5(a), "*Normalized Average TCA*" is the ratio between the average TCA of local activation with specific N and the Average TCA of global activation. Figure 5(a) shows that the Normalized Average TCA increases by localizing switching activity to smaller regions (i.e. larger N) by up to 30X. Local activation significantly reduces circuit activity (i.e. background noise) and magnifies the Trojan's contribution. The results indicate that the Average TCA of global activation is less than local activation for all values of N, even though the greater number of transitions is observed inside Trojan circuits. For example, Normalized Average TCA of 18-input comparator Trojan increases by about 20X with N =24 compared with the global activation in Figure 5(a).

"*Peak TCA*" indicates the maximum value of TCA calculated per clock cycle. "*Normalized peak TCA*" is the ratio between the Peak TCA of local activation with specific N and the Peak TCA of global activation, in Figure 5(b). The Normalized Peak TCA points out cases where the Trojan's impact exceeds the impact of process variations, and the Trojan can be detected much more easily. The results show that even for small Trojans, the Normalized Peak TCA increases significantly, which can help effectively detect them in the presence of process variations. For instance, the peak TCA of 18-input comparator in the local activation with N =24 is about 13X more than that of the global activation in Figure 5(b). It is observed that Normalized Average TCA decreases from N =9 to N =16 for the Trojans' 12- and 18- input comparators. This is caused by Trojans' implementation; they are distributed among several regions, and we observe that there are fewer Trojan gates in the target regions. Therefore, the Trojans' activities relatively decreases while circuit activity increases with N =16 compared with N =9.

For the 18-input comparator Trojan, Figure 6 shows TCA per vector for N =1 (global), 9, and 24. The results for N=9 and 24 have been normalized with respect to N=1 The figure emphasizes

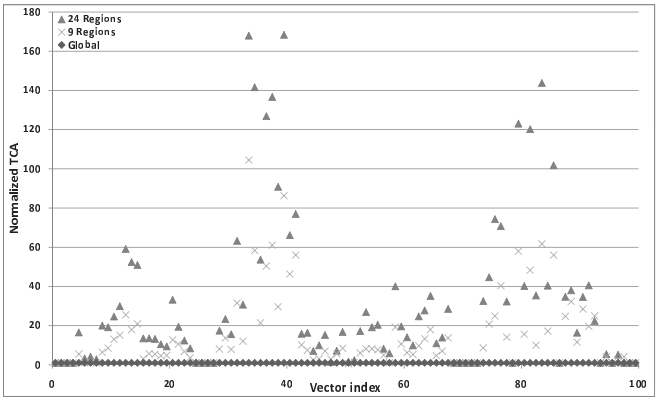

Fig. 6. TCA per vector for the 18-input comparator with N =1 (global), 9, and 24.

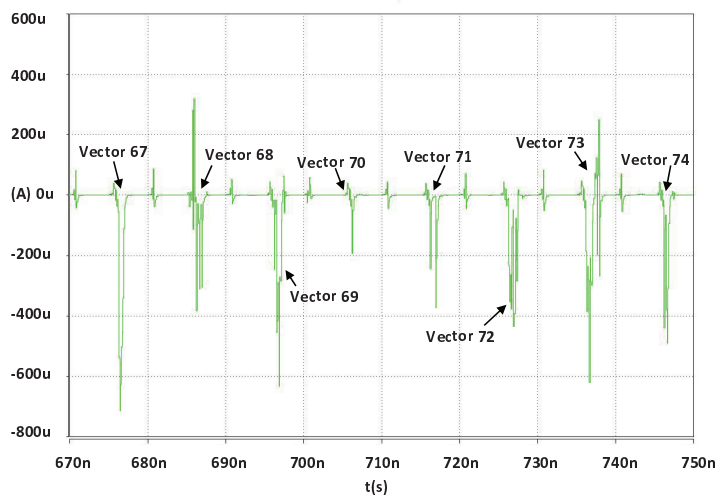

Fig. 7. Current consumption of the 18-input comparator Trojan when N =24.

that the impact of the Trojan on design's power consumption depends on both Trojan activity and circuit activity. The larger number of transitions in the Trojan does not necessarily make Trojan detection easier as circuit activity may mask the Trojan's contribution to the total circuit power. The results show that most often, in the entire simulation, the TCA of global activation, where N=1, is nearly zero. In contrast, for N =9 and 24, the Trojan contribution is significantly magnified up to 168X. The same analysis is done for N =4 and 16, and similar results are obtained. Therefore, by using power analysis based techniques [12][13][14], it is possible to more effectively detect Trojans with local activation.

Figure 7 shows the contribution (difference between Trojan-inserted and Trojan-free circuits) of the 18-input comparator Trojan when N=24 on circuit current consumption during application of vectors 67 to 74, and Table 1 presents detailed analysis of circuit activity in this interval. Table 1 also shows the behavior of global activation during this time. For example, vector 67 generates one transition in the Trojan (shown as TA) while the number of transitions in the circuit (shown as CA) is 115; therefore, the TCA of local activation would be 0.0087. The same analysis is done for global activation, and results show 12 transitions in the Trojan while 2051 transitions are generated in the circuit, hence, the TCA of global activation would be 0.0058. Comparing both TCAs indicates that global activation masks the Trojan's contribution to total switching because of high switching activity, as background noise of the circuit. The TCA of local activation is 1.6X higher than the TCA of global activation even though it generates 12 times less transitions in the Trojan. Similar results are obtained for other vectors (68-74) as well. The final row in the table shows $\frac{TCA(local)}{TCA(global)}$ where in all cases, TCA(local) is considerably larger than TCA(global); up to 27X for vector 73.

The impact of the Trojan on circuit power consumption can be seen in Figure 7. For example, vector 73 has significant impact on the current trace at time $735nsec$ compared with the other vectors, and Table 1 accordingly shows that the TCA (and accordingly $\frac{TCA(local)}{TCA(global)}$) of vector 73 is greater than other vectors. HSPICE simulation results show that the average current consumption of the Trojan (Tj. Avg. Curr. Cons.) of vector 73 is $7.8\mu A$, which is greater than other vectors. To compare the impact of the Trojan in local and global activations on circuit power consumption, in rows 7 and 13, Trojan power (current) consumption is normalized by circuit's average current consumption (Cir. Avg. Curr. Cons.) per vector. The normalized values also show the larger impact of local activation on the Trojan's power consumption.

Adversary may design Trojan circuit to be inactive during authentication time when TE signal is active. Trojan may use TE signal as a trigger input and starts working when TE signal is inactive, i.e. the circuit is in functional mode. To address this issue, TE signal must be switched on and off frequently. When TE signal is on (high), the circuit is in shift (or scan) mode and we shift one or more bits into the scan chain. To see impact of the new random pattern in the scan chain, we switch back TE signal to low. This will allow the pattern to be applied from scan flip-flops to the circuit under test and responses will go back to the scan chain.

18-input comparator is equipped with TE signal such that it is fully functional when TE signal is off. To evaluate alternating TE signal, four cases are simulated and Normalized Average and Peak TCAs are measured for N =9 and 24. In the first case, TE1(1)0(1), TE signal is switched by each clock cycle. In the second case, TE1(10)0(1), TE signal is on for ten clock cycles and then switched off for one clock cycle. In the third case, TE1(30)0(1), on state of TE signal lasts for 30 clock cycles and then switched off for one clock cycle. The final case is TE1(100)0(0) where TE signal is kept high for the entire simulation. Lengthening the high duration of TE signal would magnify TCA since circuit is subjected to less switching activity. The results, in Table 2, show that the 18-input comparator Trojan impact would be exposed by switching TE signal between on and off. The results show that the Normalized Average and Peak TCAs considerably increases (up to 12X) by using larger N and keeping TE signal high for more number of clock cycles. It should be noted that the Normalized Average and Peak TCAs are less compared with results shown in Figure 5. This is because of functional dependency among the vectors, when switching TE signal on and off.

4. CONCLUSIONS

This paper presented a new layout-aware scan-cell reordering method aiming at limiting switching activity to a specific region to improve Trojan detection. The results showed that switching in most of the non-target regions can be reduced significantly. The impact of the region's size was evaluated, and the results indicated that smaller regions can more effectively magnify Trojan activity in comparison with global activation. Meanwhile, the method can be used for localizing hardware Trojans. In future work, we will address process variations. In addition, the effectiveness of the proposed technique in dealing with sequential Trojans will be evaluated. We will design and fabricate a circuit while the technique will be applied in physical design, and capability of the technique will be evaluated in practice.

Table 1. The 18-input comparator Trojan-inserted circuit activity analysis.

	Vector index	67	68	69	70	71	72	73	74
Local	TA	1	4	6	1	2	6	8	6
	CA	115	139	135	107	141	105	133	139
	TCA	0.0087	0.028	0.045	0.0093	0.014	0.057	0.060	0.043
	Tj. Avg. Curr. Cons. (μA)	7	4.4	6.2	0.8	2.4	6.2	7.8	4.2
	Cir. Avg. Curr. Cons. (μA)	1060	1080	1100	1060	1100	1120	1100	1120
	Current Ratio (Tj./Cir.)	0.0066	0.0040	0.0056	7e-04	0.0021	0.0055	0.0070	0.0037
Global	TA	12	8	5	10	10	6	4	4
	CA	2051	1998	1889	1763	1667	1804	1788	1773
	TCA	0.0058	0.0040	0.0026	0.0056	0.0059	0.0033	0.0022	0.0022
	Tj. Avg. Curr. Cons. (μA)	16.6	94	82	7	5.2	26	7.8	8.8
	Cir. Avg. Curr. Cons. (μA)	4660	4760	4800	4620	4740	5200	4540	4860
	Current Ratio (Tj./Cir.)	0.0035	0.0019	0.0017	0.0015	0.0010	0.0050	0.0017	0.0018
TCA(Local)/TCA(Global)		1.5	7	17	1.6	2.3	17	27	19

Table 2. Test enable signal alteration analysis for 18-input Trojan.

	Normalized Average TCA		Normalized Peak TCA	
	9 Regions	24 Regions	9 Regions	24 Regions
TE1(1)0(1)	1.29X	1.57X	1.46X	2.68X
TE1(10)0(1)	2.4X	2.6X	2.31X	2.75X
TE1(30)0(1)	3.59X	4.26X	5.90X	5.68X
TE1(100)0(0)	6.19X	11.90X	5.55X	16.66X

5. REFERENCES

[1] *http://www.acq.osd.mil/dsb/reports/2005-02-HPMS_Report_Final.pdf*

[2] S. Adee "The Hunt for the Kill Switch," *http://www.spectrum.ieee.org/print/6171*

[3] M. Tehranipoor and F. Koushanfar, "A Survey of Hardware Trojan Taxonomy and Detection," *IEEE Design and Test of Computers,* pp. 10-25, January-February 2010.

[4] P. C. Kocher, J. Jaffe and B. Jun, "Differential Power Analysis," in *Proc. of the CRYPTO*, vol. 1666 of Lecture Notes in Computer Science, pp. 388-397

[5] Y. Jin and Y. Makris, "Hardware Trojan Detection using Path Delay Fingerprint," in *Proc. of the IEEE International Workshop on Hardware-Oriented Security and Trust(HOST2008)*, pp. 51-57, 2008.

[6] D. Rai and J. Lach, "Performance of Delay-based Trojan Detection Techniques under Parameter Variations," in *Proc. of the IEEE International Workshop on Hardware-Oriented Security and Trust(HOST2009)*, pp. 58-65, 2009.

[7] S. Jha and S. K. Jha, "Randomization Based Probabilistic Approach to Detect Trojan Circuits," in *Proc. of the IEEE High Assurance Systems Engineering Symposium(HASE08)*, pp. 117-124, 2008.

[8] M. Potkonjak, A. Nahapetian, M. Nelson and T. Massey, "Hardware Trojan Horse Detection using Gate-Level Characterization," in *Proc. of the IEEE International Design Automation Conference (DAC09)*, 2009.

[9] Y. Alkabani and F. Koushanfar, "Consistency-Based Characterization for IC Trojan Detection," in *Proc. of the International Conference on Computer-Aided Design (ICCAD09)*, 2009.

[10] R. S. Chakraborty, F. Wolff, S. Paul, C. Papachristou and S. Bhunia, "MERO: A Statistical Approach for Hardware Trojan Detection," in *Proc. of the Workshop on Cryptographic Hardware and Embedded Systems (CHES09)*, 2009.

[11] H. Salmani, M. Tehranipoor and J. Plusquellic, "New Design Strategy for Improving Hardware Trojan Detection and Reducing Trojan Activation Time," in *Proc. of the IEEE International Workshop on Hardware-Oriented Security and Trust (HOST2009)*, pp. 66-73, 2009.

[12] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi and B. Sunar, "Trojan Detection using IC Fingerprinting," in *Proc. of the Symposium on Security and Privacy*, pp. 296-310, 2007.

[13] R. Rad, X. Wang, J. Plusquellic and M. Tehranipoor, "Taxonomy of Trojans and Methods of Detection for IC Trust," in *Proc. of the International Conference on Computer-Aided Design (ICCAD08)*, pp. 632-639, 2008.

[14] X. Wang, H. Salmani, M. Tehranipoor and J. Plusquellic, "Hardware Trojan Detection and Isolation using Current Integration and Localized Current Analysis," in *Proc. of the International Symposium on Fault and Defect Tolerance in VLSI Systems (DFT08)*, pp. 87-95, 2008.

[15] M. Banga and M. S. Hsiao, "A region based approach for the identification of hardware trojans," in *Proc. of the IEEE International Workshop on Hardware-Oriented Security and Trust (HOST2008)*, pp. 40-47, 2008.

[16] M. Banga and M. S. Hsiao "A Novel Sustained Vector Technique for the Detection of Hardware Trojans," in *Proc. of the International Conference on VLSI Design*, pp. 327-332, 2009.

[17] M. Bushnell and V. Agrawal, "Essentials of Electronics Testing," Kluwer Publishers, 2000

[18] X. Wang, M. Tehranipoor and J. Plusquellic, "Detecting Malicious Inclusions in Secure Hardware: Challenges and Solutions," in *Proc. of the IEEE International Workshop on Hardware-Oriented Security and Trust(HOST2008)*, pp. 15-19, 2008.

[19] R. Sankaralingam, R. R. Oruganti and N. A. Touba, "Static Compaction Techniques to Control Scan Vector Power Dissipation," in *Proc. of the IEEE VLSI Test Symposium (VTS00)*, pp. 35-40, 2000.

[20] S. Devadas and S. Malik, "A Survey of Optimization Techniques Targeting Low Power VLSI Circuits," in *Proc. of the Design Automation Conference(DAC95)*, pp. 242-247, 1995.

[21] N. Badereddine, P. Girard, S. Pravossoudovitch, A. Virazel, and C. Landrault, "Scan Cell Reordering for Peak Power Reduction during Scan Test Cycles," IFIP International Federation for Information Processing, ISBN 978-0-387-73660-0, pp. 267-281, 2007.

[22] Y.Z. Wu, M. C.-T. Chao, "Scan-Chain Reordering for Minimizing Scan-Shift Power Based on Non-Specified Test Cubes," in *Proc. of the IEEE VLSI Test Symposium (VTS08)*, pp. 147-154, 2008.

[23] W. Li, S. Wang, S. T. Chakradhar and S. M. Reddy, "Distance Restricted Scan Chain Reordering to Enhance Delay Fault Coverage," in *Proc. of the International Conference on VLSI Design*, pp. 471-478, 2005.

[24] N. Devtaprasanna, S. M. Reddy, A. Gunda, P. Krishnamurthy, I. Pomeranz, "Improved Delay Fault Coverage using Subsets of Flip-flops to Launch Transitions," in Proc. of the *IEEE Asian Test Symposium (ATS05)*, pp. 202-207, 2005.

[25] M. Hirech, J. Beausang, X. Gu, "A New Approach to Scan Chain Reordering using Physical Design Information," in Proc. of the *IEEE International Test Conference 1998 (ITC98)*, pp. 348, 1998

[26] L. Hsu, H. Chen, "On Optimizing Scan Testing Power and Routing Cost in Scan Chain Design," in Proc. of the *International Symposium on Quality Electronic Design (ISQED06)*, pp. 451-456, 2006.

[27] Synopsys Inc., User Manual, *http://www.synopsys.com/*