Improving Transition Probability for Detecting Hardware Trojan Using Weighted Random Patterns

Kshirod Chandra Mohapatra(✉), M. Priyatharishini, and M. Nirmala Devi

Department of Electronics and Communication Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India
kshirod_00066@yahoo.co.in,
{m_priyatharishini,m_nirmala}@cb.amrita.edu

Abstract. Computer system security has related to security of software or the information processed. The underlying hardware used for information processing has considered to as trusted. The emerging attacks from Hardware Trojans (HTs) violate this root of trust. The attacks are in the form of malicious modification of electronic hardware at different stages; possess major security concern in the electronic industries. An adversary can mount HT in a net of the circuit, which has low transition probability. In this paper, the improvement of the transition probability by using test points and weighted random patterns is proposed. The improvement in the transition probability can accelerate the detection of HTs. This paper implements weighted random number generator techniques to improve the transition probability. This technique is evaluated on ISCAS 85' benchmark circuit using PYTHON and SYNOPSYS TETRAMAX tool.

Keywords: Transition probability · Hardware Trojan · Weighted random patterns

1 Introduction

The security of computer systems has related to the protection of the software or the information processed. The essential hardware needed for processing the information seems to be trusted. The emerging attacks from the Hardware Trojans (HTs) contravene this root of trust. Increasing dependence on untrusted 3rd party Intellectual properties, CAD tools and decreasing control on the design or fabrication steps of Integrated circuits results in a malicious modification in ICs.

HTs are activated in aberrant conditions as discussed in the paper [1], and the conventional tests are not effectual for the detection of HTs. The furtive nature of HTs implies, most of the time, they are integrated into nets that has less transition probability, or less SCOPE measurements. The inputs to these HTs are fed from the nets (wires) with low transition probability (tp), which in turn influence power and delay analysis. Several techniques for HT have been suggested in the last decade. The techniques are characterized into two types (1) Side-channel analysis [2–6], (2) Logic testing [7–10].

N. Kumar and R. Venkatesha Prasad (Eds.): UBICNET 2019, LNICST 276, pp. 207–217, 2019.
https://doi.org/10.1007/978-3-030-20615-4_16

In the side-channel analysis, the parameters like power, delay, current (both transient and leakage) are analyzed and techniques for Trojan detection are discussed in the paper [2]. In [3] the path delay model is used to detect the HTs by using the separate clock (shadow clock). In [2] uses the power (transient) to detect the Trojans as side channel analysis. Here they conclude that the resolution of the detection of HTs measured directly on the activity rate of HTs and inversely on the activity of the circuit. The method given in [4] based on reordering the scan cells to decrease the activity of the circuit. It helps in magnifying power consumes by HTs, hence total circuit power (transient) is improved, as a result, side channel power consumption improved. In paper [5], another side-channel power consumption technique that uses nonlinear detection method and it shows the differences in nonlinear curves of consumed power between the inflected circuit and reference circuits. In this [5] paper for observation, projection and analysis of different nonlinear functions are proposed. In order to improve the side channel signal analysis, an algorithm is given by the [6]. Here the idea is to decrease the effect of the aberrant points due to the noise the superimposed curve of a signal by a smooth filtering algorithm.

Logical testing is additional way to detect HT. In this method, test patterns are provided to activate hidden HTs. In [7], generated test patterns used to excite the aberrant logic at the internal nodes for multiple times. This statistical approach maximizes the probability of interpolate HT's being triggered and detected by logic testing. In [8] N vectors form a set, and this set has given to the circuit under analysis(CUA) and design circuit, if the CUA and design have same probability signature, implies CUA isn't inflected. In [8], HTs detection based on a method called activation sequence generation. This uses an activation sequence, which activates the HT circuit and this activation effect will propagate to the memory elements or PO's. At PO's the effects can be observe by the designer.

There are methods, which uses both logic testing and side-channel analysis to identify HTs such as [9, 10]. In [9], Principal component analysis (PCA) algorithm is the data processing algorithm used for the detection of Trojans in the circuit. In this the characteristics of the reference circuit and inflected circuit are analyzing by PCA algorithm and corresponding projections are plotted on 3D graph, which gives enough information about the reference circuit and inflected circuit. In [10], demonstrates a new side-channel test generation mechanism. This presents the concept of multiple number of excitations of rare switching (MERS). MERS can expressively improve sensitivity of HTs detection. This approach statistically increases switching activity in an unrecognized HT and amplify the HT effect in the large process variation conditions. In [11], HT detection was based on the improving the transition probability of the internal nodes by weighted input probability. In this method, they found the internal node has less controllability & observability and improved the transition probability using weighted input probability.

As we know that, HTs activated in aberrant conditions, less coverage detection provided by random test pattern and generation and deterministic test pattern generation by ATPG tools. However better results can be obtained for the probability of

activation of HTs by increasing tp by means of extra hardware such as DSFF or test points. In [12], insertion of 2:1 mux as the test point is proposed. This paper based on the tp of fan-out cone gates extremely depends on the input probability of applied to logical-gate inputs. Candidate nets are the nets with low tp than user defined threshold tp. For each candidate nets, a 2:1 MUX inserted in its input, which has less transition probability. Again, to optimize the number of insertion of 2:1 MUX in the candidate nets, weighted random patterns (WRP) are applied. In paper [13], gate-level characterization (GLC) approached is used for detection HTs. They have calculated side channel parameter such as power for each test vector applied to inflected circuit.

In [14, 15] given the idea about the weight set calculation for the weighted random generator (WRG). In the paper [14], the basic idea is minimize the variance such that number of test patterns required is less for test the circuit. Paper [15] is based on the fact that, test pattern which have less sampling probability than sampling probability of test pattern from LFSR are deleted and new test set is created.

In this paper, we have compared the different weight set generation scheme from [14, 15] and obtained the corresponding results for more efficient weights set generation algorithm for improving the tp of low probable nets. After successfully generating the weight sets, low tp nets are identified and from these suitable candidates net is observed for insertion of 2:1 mux. After insertion of 2:1 mux and application of weight set, it is observed that there is significant improvement in low tp net. Improved tp of each net for the circuit is an effective way to enhance HTs detection in two ways as follows. (1) As the tp of the nets will increased, results more transitions in the nets, hence the side channel signal analysis of HTs will improved. (2) it allows to fully activated the HTs and observes faulty outputs at POs of circuit.

The rest of the paper organized as follows. In Sect. 2, proposed methodology is discussed, in Sect. 3 comparison of test set based on [14, 15], simulation results are shown in Sect. 4 and Sect. 5 concludes this paper.

2 Proposed Methodology

The flow chart for the proposed methodology shown in Fig. 1. The aim of the proposed methodology is to identify the low probable nets and from these low probable nets the suitable candidate net for insertion of 2:1 multiplexer as test points are determined. In the proposed methodology, generation of weighed random numbers and application of these patterns for obtaining the optimal number of test points for improving transition probability.

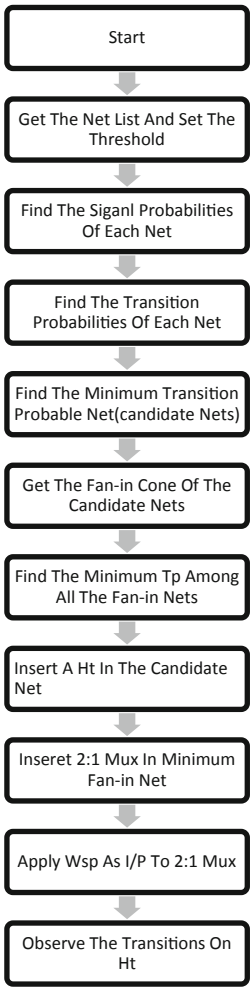

Fig. 1. Flow chart for proposed method

2.1 Generating the Circuit's Netlist

For our proposed method, circuit is designed using hardware description language. There are several hardware description language like VHDL, VERILOG, and SYSTEM C. Here circuit is designed using Verilog hardware description language. The netlist for the circuit obtained using SYNOPSYS TETRAMAX tool. For identifying the low tp nets, threshold for tp is provided by the user. In [13], general method is discussed for setting the threshold tp. After setting the threshold for tp, circuit's netlist is analyzed for finding out the low tp nets.

2.2 Determining the Transition Probability for Each Gate

Calculation of transition probability (tp) for each net is done by using Shannon's decomposition theorem. The detailed tp for each gate is given in the below Table 1. Table 1 shows some basic formula to calculate the tp for each basic gate. The basic gates are AND, OR, NOR & NAND. For example consider the AND gate shown in Fig. 2. IT has two primary inputs namely A and B. "Prob0 -> 1" is the transition probability when there is a transition from 0 to 1 and "Prob1 -> 0" is transition probability from 1 to 0. "PA" and "PB" are primary input signal probability. "Probout = 0" and "Probout = 1" are the primary output probabilities being 0 and 1 respectively.

Table 1. Computation of transition probability of basic gates

	$Prob_{0\rightarrow1} = Prob_{out=0} \times Prob_{out=1}$
AND	$(1 - Prob_A Prob_B) \times Prob_A Prob_B$
OR	$(1 - Prob_A)(1 - Prob_B) \times (1 - (1 - Prob_A)(1 - Prob_B))$
NAND	$Prob_A Prob_B \times (1 - Prob_A Prob_B)$
NOR	$(1 - (1 - Prob_A)(1 - Prob_B)) \times (1 - Prob_A)(1 - Prob_B)$
XOR	$(1 - (P_A + P_B - 2P_A P_B)) \times (P_A + P_B - 2P_A P_B)$

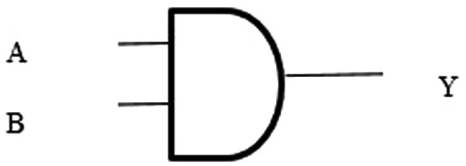

Fig. 2. AND gate

2.3 Identifying Suitable Insertion Point

The Transition probability (tp) of all the nets for the circuit under test are calculated and the low tp nets are extracted from the list. For finding the suitable test points, fan-in cone of the low tp nets is taken. Among all the fan-in cone nets the lowest tp net is considered for test point insertion of 2:1 mux. The below example shown in Fig. 3 described the procedure of finding out the suitable test points. In the below figure two AND gates are shown, the output of the second AND gate has low tp net Y'. By the fan-in cone analysis, one of the inputs of the first AND gate has the lowest tp among all the fan-in nets and this net is suitable for insertion of 2:1 mux as the test point.

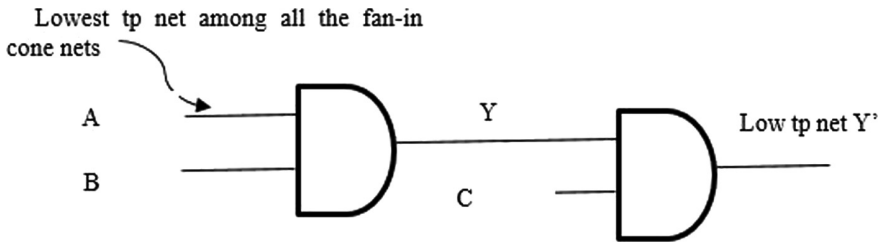

Fig. 3. Low tp example

2.4 Application of Weighted Random Patterns

After identifying test points in the circuit, insertion of 2:1 mux is done on every test points. To reduce the number of 2:1 mux insertion, weighted random patterns are applied as test inputs. Weighted random number patterns are normal random patterns with each of the bit in the test pattern has different weights for being 1. The detailed analysis of weighted random patterns are given Sect. 3.

3 Weighted Random Pattern

Random patterns are generated using LFSR and are used to detect faults that are present in circuits during the testing process. Weighted random patterns are very significant in finding out the stuck at faults, which are un–identified by the random patterns generated by the LFSR. These faults are as called random pattern resistance faults [13]. To detect random resistance faults, weighted random patterns are used. Generation of weighted random patterns are classified into two types (1) Depends on circuit topology and (2) Encode the deterministic test set into weight sets [14, 15]. In proposed method, second one is used.

3.1 Introduction to Weighted Random Number Generator

To test the pattern resistant faults WRPs are proposed. In WRPs, the probability of being logic 1 at each input is biased differently. A WRG is made up with a LFSR and some combinational circuits in order to bias the input probability. Weight calculation logic is used to bias the probability of occurrence of logic value 1 at inputs.

The procedure to generate the WRP is categorized into two types in terms of weight calculation sources; they are (1) structural analysis and (2) deterministic test pattern set. The advantage in (1), it can used in an ATPG process and time taken for generation of weight set is less than the (2). However, sufficient fault content is not guaranteed. The (2) can provide significant test coverage with considerable test length.

3.2 Weight Random Pattern Number Generator Algorithm

The Algorithm 1 is based on identifying the suitable test subset from the deterministic test pattern and this algorithm is repeated until desired fault content is achieved.

Algorithm 1.

Input: Deterministic Test Pattern,
Output: New weight random number patterns

1: Get the deterministic test pattern from any test pattern generator.
2: Create subset from deterministic test set by calculating the conflicts.
3: Calculate weights, Average, sampling probability, variance.
4: Modify the generated weight set by E_i.

Following notations are used to explain the course of the weight set calculation algorithm.

Let a deterministic test pattern 't_j' and a test pattern set be T = {t_1, t2 … tl}, 'l' is the test length. The i^{th} bit of a deterministic test set be t_j denoted as t[i], and the weight of bit position are denoted as w_i, and is calculated by the Eq. (2).

An ATPG system generates the deterministic test patterns. Then the pattern sets are grouped into different subsets according to the number of conflicts between test patterns. A conflict between 2 test patterns, tj and t_k is defined as $\Delta(tj, tk)$ and given by Eq. (1)

$$\Delta(tj, tk) = \sum_{i=0}^{m} \delta(tj[i], tk[k]) \tag{1}$$

The weights of deterministic test patterns are calculated by (2)

$$wi = \frac{|\{tj \in T | tj[i] = 1\}|}{|\{tj \in T | tj[i] \neq X\}|} \tag{2}$$

The sampling probability '*Pj*' of a deterministic test pattern with a weight set is delineate as the probability that the test pattern occurs through the WRP generation cycle and given by the Eq. (3)

$$Pj = \prod_{i=1, tj[i] \neq X}^{m} \{(wi \times tj[i]) + (1 - wi) \times (1 - tj[i])\} \tag{3}$$

Average of the sampling probability is given by (4)

$$A = \frac{\sum_{j=1}^{l} pj}{l} \tag{4}$$

Variance of the sampling probabilities is given by (5)

$$V = \frac{\sum_{j=0}^{l} (Pj - A)^2}{l} \tag{5}$$

To reduce the variance, the effect of modifying the weight of the i^{th} bit is evaluated E_i and given by (6)

$$Ej = \sum_{i=1,tj[i]\neq X}^{m} \{(A - PJ) \times tj[i]) + (Pj - A) \times (1 - tj[i])\} \tag{6}$$

Algorithm 2.

Input: Deterministic Test Pattern,
Output: New weight random number patterns

1: Get the deterministic test pattern from any test pattern generator.
2: Calculate the sampling probabilities of the deterministic test pattern Sp.
3: Calculate the sampling probabilities test patterns from LFSR Splfsr.
4: Remove test pattern from the subset which are Sp<Splfsr
5: Calculate Sp ,weights ,average ,and variance.
6: Modify the reduced subset by E_i

In the Algorithm 2 candidate list include the test patterns which have more sampling probability than the test patterns from LFSR. The idea is that WRP have better sampling probabilities than the LFSR. Weight set and sampling probability are calculated from the candidate list. Weight set calculated form the candidates list are modified in order to reduce the variance of the sampling probability. After the modification the rounding-off the weight set is performed.

The SP of a test pattern with LFSR can is calculated by Eq. (3) with all weights set to 0.5.

3.3 Algorithm for Detection of HTS

The following Algorithm 3 identifying the suitable nets to which 2:1 MUX is inserted.

Algorithm 3.

Input: circuit net list, transition probability threshold (Ttp),
Output: Signal probabilities of candidate nets

1: Compute the signal probabilities S.
2: Compute transition probabilities Tp.
3: Find the minimum transition probability M_{minTp}.
4: Get the fan-in cone Fin.
5: Choose the minimum Tp (from Fin) T_g.
6: Insert HT into MminTp net.
7: Insert 2:1 MUX into Tg.
8: Apply WSP (Weighted Signal Probabilities) test input to the 2:1 MUX.
9: Observe transition on HT.

As we know that Trojans are connected to the low probable nets, so in the proposed work, inputs are provided from the low probable nets to the inserted HT into circuit. After the insertion of 2:1 MUX in the target net, the tp of the candidate nets are improved, therefore we can see improvement in the activation rate in HT.

4 Experimental Results

In this paper, comparison of two different algorithms for generating weight random number is presented and the proposed method is validated using ISCAS benchmark circuits. Table 2 shows the transition probabilities without weighted random patterns. It is observed that the nets N10 and N11 have less tp as compared to other nets in the circuit.

Table 2. Transition Probability without WRP

Nets	Signal prob.1	Signal prob.0	Transition prob.
N10	0.75	0.25	0.1875
N11	0.75	0.25	0.1875
N16	0.625	0.375	0.234
N19	0.625	0.375	0.234
N22	0.53125	0.46828	0.249
N23	0.609375	0.3906	0.238

Table 3 shows the improvement in the transition probability from Algorithm-1. The proposed method in Sect. 2 is applied for the circuit under test and the results are projected in Table 3. It is observed that the nets N10 and N11 which has low tp as shown in Table 1 is improved to the proposed method. The tp of net N11 is improved by 23.1% but N16 tp is drooped by 46.4%. To improve tp of net N16 another test point is inserted in lowest fan-in cone of N16.

Table 3. Transition Probability with WRP one

Nets	Signal prob.1	Signal prob.0	Transition prob.
N10	0.775	0.225	0.174
N11	0.5176	0.4284	0.2448
N16	0.854	0.146	0.125
N19	0.733	0.264	0.1957
N22	0.33815	0.66185	0.224
N23	0.374	0.626	0.2341

Table 4 shows the improvement in the transition probability from Algorithm-2. Here all tp of low tp nets are improved. The tp net N10 and N11 is improved by 10% and 14% respectively. Further improvements can achieved for circuits that are more complex.

Table 4. Transition Probability with WRP two

Nets	Signal prob.1	Signal prob.0	Transition prob.
N10	0.7035	0.2965	0.2085
N11	0.6821	0.3179	0.2168
N16	0.6105	0.3895	0.2377
N19	0.7442	0.2558	0.1903
N22	0.5705	0.4295	0.245
N23	0.5456	0.4544	0.2479

5 Conclusion

In this paper, the two algorithms for generation of weighted random patterns have been successfully implemented. These algorithms are implemented in PYTHON and the deterministic test patterns were obtained from SYNOPSYS TETRAMAX ATPG tool. The experiment is performed on ISCAS-85 C17 circuit. The comparison between the two algorithms is shown in Tables 3 and 4 respectively. When the WRP are applied to the input nets of C17 circuit, it is observed that there is a significant increase in the transition probability as compared to Table 2. Algorithm 2 shows better improvement in transition probability when compared to Algorithm 1. Detection of hardware Trojan will be a part of this paper in future.

References

1. Bhunia, S., et al. Hardware Trojan attacks: threat analysis and countermeasures. Proc. IEEE **102**(8), 1229–1247 (2014). Author, F., Author, S.: Title of a proceedings paper. In: Editor, F., Editor, S. (eds.) CONFERENCE 2016, LNCS, vol. 9999, pp. 1–13. Springer, Heidelberg (2016)
2. Rad, R., Plusquellic, J., Tehranipoor, M.: Sensitivity analysis to hardware Trojans using power supply transient signals. In: Proceedings of IEEE International Workshop on Hardware-Oriented Security Trust (HOST), Anaheim, CA, USA, pp. 3–7 (2008)
3. Cha, B., Gupta, S.K.: Trojan detection via delay measurements: a new approach to select paths and vectors to maximize effectiveness and minimize cost. In: Proceedings of IEEE Design, Automation and Test in Europe Conference Exhibit. (DATE), Grenoble, France, pp. 1265–1270 (2013)
4. Zhou, E., Li, S., Zhao, Z., Ni, L.: Nonlinear analysis for hardware Trojan detection. In: 2015 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), Ningbo, pp. 1–4 (2015). https://doi.org/10.1109/icspcc.2015.7338921
5. Zhang, Z., Li, L., Tang, T., Wei, Z.: Side channel analysis of hardware Trojan based on smooth filtering algorithm. In: 2015 8th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, pp. 192–195 (2015). https://doi.org/10.1109/iscid.2015.253
6. Chakraborty, R.S., Wolff, F., Paul, S., Papachristou, C., Bhunia, S.: *MERO*: a statistical approach for hardware Trojan detection. In: Clavier, C., Gaj, K. (eds.) CHES 2009. LNCS, vol. 5747, pp. 396–410. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04138-9_28

7. Jha, S., Jha, S.K.: Randomization based probabilistic approach to detect Trojan circuits. In: 2008 11th IEEE High Assurance Systems Engineering Symposium, Nanjing, pp. 117–124 (2008). https://doi.org/10.1109/hase.2008.37
8. Yoshimura, M., Bouyashiki, T., Hosokawa, T.: A hardware Trojan circuit detection method using activation sequence generations. In: 2017 IEEE 22nd Pacific Rim International Symposium on Dependable Computing (PRDC), Christchurch (2017). https://doi.org/10.1109/prdc.2017.40
9. He, C., Hou, B., Wang, L., En, Y., Xie, S.: A novel hardware Trojan detection method based on side-channel analysis and PCA algorithm. In: 2014 10th International Conference on Reliability, Maintainability and Safety (ICRMS), Guangzhou (2014). https://doi.org/10.1109/icrms.2014.710736221-222
10. Huang, Y., Bhunia, S., Mishra, P.: Scalable test generation for Trojan detection using side channel analysis. IEEE Trans. Inf. Forensics Secur. **13**(11), 2746–2760 (2018). https://doi.org/10.1109/tifs.2018.2833059. 1043–1046
11. Devi, N.M., Jacob, I.S., Ranjani, S.R., Jayakumar, M.: Detection of malicious circuitry using transition probability based node reduction technique. http://dx.doi.org/10.12928/telkomnika.v16i2.6812
12. Zhou, B., Zhang, W., Thambipillai, S., Jin, J.T.K., Chaturvedi, V., Luo, T.: Cost-efficient acceleration of hardware Trojan detection through fan-out cone analysis and weighted random pattern technique. IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. **35**(5), 792–805 (2016). https://doi.org/10.1109/tcad.2015.2460551
13. Karunakaran, D.K., Mohankumar, N.: Malicious combinational Hardware Trojan detection by gate level characterization in 90 nm technology. In: Fifth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Hefei, pp. 1–7 (2014). https://doi.org/10.1109/icccnt.2014.6963036
14. Lee, H., Kang, S.: A new weight set generation algorithm for weighted random pattern generation. In: Proceedings 1999 IEEE International Conference on Computer Design: VLSI in Computers and Processors (Cat. No. 99CB37040), Austin, TX, USA, pp. 160–165 (1999). https://doi.org/10.1109/iccd.1999.808421
15. Kirn, H.-S., Lee, J.K., Kang, S.: A new multiple weight set calculation algorithm. In: Proceedings International Test Conference 2001 (Cat. No. 01CH37260), Baltimore, MD, USA, pp. 878–884 (2001). https://doi.org/10.1109/test.2001.966710
16. Yeap, G.: Practical Low Power Digital VLSI Design. Springer, Heidelberg (1998). https://doi.org/10.1007/978-1-4615-6065-4