2019 32nd International Conference on VLSI Design and 2019 18th International Conference on Embedded Systems (VLSID)

Hardware Trojan Detection by Stimulating Transitions in Rare Nets

Tapobrata Dhar*, Surajit Kumar Roy† and Chandan Giri‡
Indian Institute of Engineering Science and Technology, Shibpur, India.
Email: {tapobrata.dhar91*, suraroy†}@gmail.com, chandan@it.iiests.ac.in‡

***Abstract*—Outsourcing in the various stages of Integrated Circuit (IC) manufacturing process leaves the ICs vulnerable to inclusion of malicious circuitry called Hardware Trojan Horses (HTH). Hardware Trojans are often inserted in nets in IC which have low chance of having transitions in signals thus making them covert. This paper presents an efficient technique to increase the probability of transitions by using tri-state buffers paired with scan registers. Simulation results show the increase in number of transitions of nets in benchmark circuits.**

Keywords: Design for Testing, Hardware Trojan, Transition Probability

INTRODUCTION

IC manufacturing process usually depends on outsourcing because of the economically viability. However, this can give an adversary chance to make clandestine modifications by inserting extra circuitry called Hardware Trojans in them which has the potential to cause disruption in their functionality [1], [2]. Hardware Trojans are inserted in the IC's nets which have low number of transitions in signals making them covert. Design for Security (DFS) methods of adding extra hardware components in the IC can be used for increasing the transition probability (TP) of the signals within it which can reveal the presence of any malicious circuit through parametric analysis or direct triggering of the payload [3], [4].

In [4], dummy scan flip flops are inserted to help detect Hardware Trojans by increasing the TP in the circuits upto a decided threshold. The work in [3] improved upon this technique by using 2-to-1 MUXs and signals with weighted probability (WSP). However, Shekarian et al. [5] circumvented these design for testing (DFT) techniques by exploiting nets with TP close to threshold.

Thus, to overcome this neutralisation measure, this paper introduces a technique in which weighted signals are supplied in various parts of the IC by tri-state buffers to increase the TP in the nets to their maximum possible value and thus detect presence of Hardware Trojans using minimal DFT insertions.

This paper is structured as follows: Section 2 introduces the proposed method for detection of hardware trojans. Section 3 includes the simulation results of the experiments conducted. Section 4 ends the paper with concluding statements.

PROPOSED METHOD

The proposed algorithm tries to maximise the TP of all the nets in a circuit. TP in a net is maximum only when $SP_i = 0.5$, where SP is the signal probability of net i. The transition probability i^{th} net can be evaluated as $TP_i = SP_i \times (1 - SP_i)$.

Test Architecture

The TP in a particular section of the circuit is improved by using tri-state buffers which is coupled with a scan register. The tri-state buffers are introduced in the input of the gate of the candidate net with the lowest transition probability. Its input is stitched with the output of a flip-flop of the register. WSP values are provided in the inputs of the flip-flops by a linear feedback shift register (LFSR) paired with AND/OR gates as used in [3].

Application of Weighted Signal Probability

When an insertion point in the IC is established, consulting the list of possible WSP values, the overall mean and variance value of TP for every WSP in the IC is evaluated in an exhaustive manner. The WSP value for which the overall mean yields the maximum value and the overall variance yields the minimum value is chosen to be applied to the candidate net.

Algorithm to Select Insertion Points

Algorithm 1: Proposed Algorithm

```
Input  : Net − List
Output : Net − List
1  IterationCount ← 0;
2  LD ← EvalLogicalDepth(Net − List);
3  SP ← EvalSignalProbability(Net − List);
4  TP ← EvalTransitionProbability(SP);
5  I_MINTP ← GetInputNet(Net − List, TP);
6  Health ← HealthEval(LD, TP, IterationCount, I_MINTP);
7  TargetNet ← SelectMinimum(Health, Net − List);
8  Prob ← EvalProbability(WSP, TargetNet, Net − List);
9  InsertBuffer(Net − List, Prob, TargetNet);
10 Net − List ← UpdateNetList(Net − List, Prob);
11 if Further improvements possible then
12 |   IterationCount ← IterationCount +1;
13 |   Go To 3 ;
14 end
```

The inputs of the gates with the lowest transition probability are stored in the I_MINTP array. The $Health$ function described in Algorithm 1 comprises of the parameters of logical depth of nets, the transition probability of the output net and the current iteration of the algorithm. The algorithm is stopped when the mean and variance values of transition probability stop improving upon further insertions.

2380-6923/19/$31.00 ©2019 IEEE
DOI 10.1109/VLSID.2019.00124

TABLE I
RESULTS OBTAINED BY APPLYING THE PROPOSED ALGORITHM ON S386, S5378, S9234A, S13207A AND S38417 CIRCUITS

Benchmark Circuit	No. of Nodes	No. of Insertions	Node Outputs with TP $\in [0, 0.025]$		Node Outputs with TP $\in (0.175, 0.2]$		Node Outputs with TP $\in (0.2, 0.225]$		Node Outputs with TP $\in (0.225, 0.25]$		Area Overhead
			Before Insertion	After Insertion	Before Insertion	After Insertion	Before Insertion	After Insertion	Before Insertion	After Insertion	
s386	172	28	27	2	27	35	4	4	33	33	11.89%
s5378	2993	36	605	550	228	275	52	52	1352	1352	1.56%
s9234a	5844	41	348	277	884	905	274	298	3277	3285	0.857%
s13207a	8651	65	728	511	1740	1744	359	437	5006	5123	0.968%
s38417	47658	1732	1214	556	7178	7344	2294	2423	29972	30037	4.55%

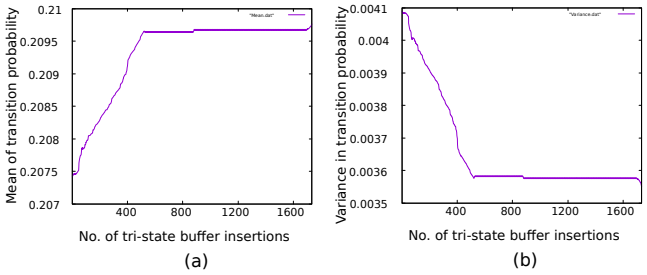

Fig. 1. (a) Mean TP value of overall circuit per insertion of tri-state buffers in s38417, and (b) Variance in TP value per insertion of tri-state buffers in s38417.

EXPERIMENTAL RESULTS

The proposed algorithm is verified using simulations on ISCAS'89 benchmark circuits as shown in Table I. The signal probabilities of all the primary inputs have been initialised to 0.5. STM 65 nm standard cell library maps the circuit during its synthesis and is used to determine the area overhead of this design for security measure.

The algorithm has been applied to s386, s5378, s9234a, s13207a and s38417 ISCAS'89 benchmark circuits. The number of required insertion points to achieve a higher mean value in transition probability is seen to be directly proportional to the size of the circuits.

Table I showcases the increase in transition probability in the nets after adding tri-state buffers. Nodes in the table consist of primary inputs and gates of the circuit. The area overhead in Table I is calculated as the percentage of transistors added for test purposes to the total number of transistors in the circuit.

The proposed method uses tri-state buffer unlike dSFFs in [4] and 2-to-1 MUX in [3] since dSFFs or 2-to-1 MUX uses 6 transistors per insertion in the circuit, whereas insertion of tri-state buffer uses 4 transistors thus lowering the spatial impact per DFT insertion.

Insertion of the tri-state buffers in the circuit increases the mean value and decreases the variance in transition probability throughout the circuit. Fig. 1 (a) shows the progressive increase in overall mean transition probability value in s38417 circuit per tri-state buffer insertion. Fig. 1 (b) showcases the overall variance value of transition probability in the entire circuit and how each insertion of tri-state buffer influences it.

However, it is to be noted that the benchmark circuits show a number of nets with low transition probabilities even after tri-state buffer addition. These nets should be considered as a critical part of the resultant circuit and should be monitored as they are vulnerable to being compromised by an adversary. Further analysis and improvement is required regarding these critical nets which remains as a part of our future work.

CONCLUSION

Increasing transition probability of nets in ICs helps reveal any inserted Hardware Trojans present through side-channel analysis or even direct triggering. In the proposed work, the transition probabilities are increased to their maximum possible values by inserting tri-sate buffers with weighted signal probability into specific regions of the circuit determined by heuristic analysis. The resultant circuit shows a significant improvement in transition probability in the nets of the ICs thus aiding in Hardware Trojan detection.

REFERENCES

[1] M. Rostami, F. Koushanfar, and R. Karri, "A primer on hardware security: Models, methods, and metrics," *Proceedings of the IEEE*, vol. 102, pp. 1283–1295, Aug 2014.

[2] M. Tehranipoor and F. Koushanfar, "A survey of hardware trojan taxonomy and detection," *IEEE Design Test of Computers*, vol. 27, pp. 10–25, Jan 2010.

[3] B. Zhou, W. Zhang, S. Thambipillai, J. T. K. Jin, V. Chaturvedi, and T. Luo, "Cost-efficient acceleration of hardware trojan detection through fan-out cone analysis and weighted random pattern technique," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, pp. 792–805, May 2016.

[4] H. Salmani, M. Tehranipoor, and J. Plusquellic, "A novel technique for improving hardware trojan detection and reducing trojan activation time," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, pp. 112–125, Jan 2012.

[5] S. M. H. Shekarian, M. S. Zamani, and S. Alami, "Neutralizing a design-for-hardware-trust technique," in *The 17th CSI International Symposium on Computer Architecture Digital Systems (CADS 2013)*, pp. 73–78, Oct 2013.