Preventing and Masking Trojan Circuits Triggering Out of Working Area

A. Matrosova*, E. Mitrofanov†, S. Ostanin‡, I.Kirienko
Tomsk State University
Tomsk, Russia
{*mau11, †qvaz, ‡sergeiostanin}@yandex.ru, irina.kirienko@sibmail.com

Abstract—**Inserting malicious sub-circuits that may cause a circuit failure or lead to theft of confidential information from a system containing the logical circuit demands detection of such sub-circuits followed by their masking if possible. We propose an approach to selection of nodes in sequential circuit, where insertion of Trojan Circuit (TC) is likely. The method is based on applying precise (not heuristic) probabilistic calculations of controllability and observability of combinational part internal nodes and algorithms of a transfer sequence detection for a set of internal states. Note that TC may be inserted in an internal node out of working area (don't care set) represented by State Transition Graph (STG). The out of working area states are compactly represented by Reduced Ordered Binary Decision Diagram (ROBDD) that is derived when the precise estimations are being calculated. ROBDD operations characterizing by polynomial complexity are used both for calculation of precise controllability and observability estimations and detection of transfer sequence. As the suggested approach considers facilities of inserting TCs out of the working area, they cannot be detected both under verification and testing in the working area (care set). The technique of masking TCs with rather small overhead is proposed. The experimental results on MCNC benchmarks illustrate applicability of the approach.**

Keywords—sequential circuits, controllability and observability of combinational circuit nodes, State Transition Graph (STG), Malicious circuit (Trojan Circuit), Reduced Ordered Binary Decision Diagram (ROBDD), working area.

I. INTRODUCTION

The enhanced utilization of outsourcing services for a part of VLSIs (Intellectual Property cores, reprogramming components based on FPGA and so on) to cut VLSI cost increases risk of Trojan Circuits (TCs) insertion that may destroy VLSI or cause theft and exposure of confidential information [1]. TCs usually activate on a small part of all possible inputs, therefore they are not detectable during neither verification nor testing stages of VLSI design process. TC are composed from two parts. Trojan trigger activates when the certain combination of signals appears on TC inputs. Trojan payload act as operation unit and is activated by trigger sub-circuit. The problem lies in detection of such malicious sub-circuits and, if achievable, masking their influence. It is preferable to utilize precise calculations in finding circuit nodes suitable for inserting TC.

In [2] authors consider TCs that contain no more than five logic elements. The conditions of TCs detections are determined by using heuristic estimations of nodes controllability. When TC is activated, there is no need to follow its influence to primary output. The approach allows detecting TC as soon as it is triggered.

In out paper the method of detection of suspicious nodes is based on using precisely calculated probabilistic estimations of controllability and observability of combinational part internal node. It guarantees finding all internal states that may provide triggering the node. The controllability estimations are derived out of working area of the sequential circuit. The estimations calculations are based on using structural description of the combinational part and representation of the sequential circuit behavior by State Transition Graph (STG). Calculations are executed with using operations on Reduced Ordered Binary Decision Diagrams (ROBDDs, further just BDDs). Algorithms of transfer sequence detection for a set of internal states are also based on using BDD operations. Technique of masking TCs is proposed. The experimental results on benchmarks illustrate applicability of the suggested approach and show that overhead for masking TC inserting out of working area is rather small.

In Section II techniques of precise calculation of controllability and observability estimations for combinational part nodes of a sequential circuit are briefly described. In Section III the way of calculation of precise controllability estimations for combinational part nodes out of working area is given. In Section IV methods of detecting transfer sequence for a set of internal states both without finding the sequence itself and with finding one sequence are discussed. In Section V the technique of masking TC is proposed. In Section VI the experimental results are considered.

II. PRECISE CALCULATION OF CONTROLLABILITY AND OBSERVABILITY ESTIMATIONS WITH USING STRUCTURAL COMBINATIONAL PART DESCRIPTION

1(0)-controllability of an internal node is defined as probability of delivering 1(0) value to it, observability is defined as probability of observation of changing 1(0) value of an internal node on the proper circuit output. Precise estimations is based on the usage of corresponding BDDs [3]

The reported study was partially supported by Russian Science Foundation, research project № 14-19-00218.

978-1-5386-3974-0/17/$31.00 ©2017 IEEE

and operations on them. The input and output poles of a TC serve as a target for these estimations.

For precise calculation of 1(0)-controllability for internal node v [4] of combinational part C we derive BDD $R^{cont}(1)$ ($R^{cont}(0)$) using the combinational circuit which output is pole v, and inputs coincide with circuit C inputs. ($R^{cont}(0)$) is obtained from $R^{cont}(1)$ by permutation of terminal nodes.

For precise calculation of observability for internal node v [4] of combinational part C and the proper circuit output we derive first BDD $R(C_v)$ for sub-circuit C_v. The sub-circuit corresponds to the chosen output of circuit C and is retrieved from it by setting internal node v as an input of sub-circuit C_v [3]. During construction of BDD $R(C_v)$, the variable order is arranged such, that root node of BDD is marked by variable v.

Let BDD $R(C_v)$ implement function f. We derive from $R(C_v)$) BDDs $R(f^{v=0})$, $R(f^{v=1})$ which roots are child nodes of $R(C_v)$ root. These BDDs implement functions $f^{v=0}$ and $f^{v=1}$ accordingly. Multiplications $R(f^{v=0})\overline{R(f^{v=1})}$, $R(f^{v=1})\overline{R(f^{v=0})}$ are executed and results are merged being represented by BDD R^{obs}.

Getting $\overline{R(f^{v=0})}$, ($\overline{R(f^{v=1})}$) from $R(f^{v=0})$, ($R(f^{v=1})$) is equivalent to exchanging of terminal nodes of the corresponding BDDs. Note that BDD operations have a polynomial complexity.

Calculating precise controllability and observability estimations we suppose that 1 value probabilities of all input variables are equal to ½. Using BDDs $R^{cont}(1)$ and R^{obs} we calculate 1 controllability and observability random estimations for node v.

For an internal node μ of the BDD, there is a corresponding function η and $p(\eta)$ - a probability of its output being 1 value. $p(\eta)$ is calculated with using probabilities $p(\eta_\mu^{x_i=0})$, $p(\eta_\mu^{x_i=1})$ of 1 values of functions $\eta_\mu^{x_i=0}$ and $\eta_\mu^{x_i=1}$, corresponding to child nodes of node μ according to formula (node μ is marked by variable x_i): $p(\eta) = p(x_i)p(\eta_\mu^{x_i=1}) + p(\overline{x_i})p(\eta_\mu^{x_i=0})$.

Moving from 1 terminal node of the corresponding BDD with using the above mentioned formula for internal nodes we reach the BDD root. As a result random estimations of 1(0)-controllability or observability are calculated.

Consider circuit of Fig. 1. Construct BDD $R^{cont}(1)$ (Fig. 2a) and BDD $R(C_v)$ (Fig. 2b) for pole v. BDDs $R(f^{v=0})$, $R(f^{v=1})$ are represented by Fig. 2c, 2d.

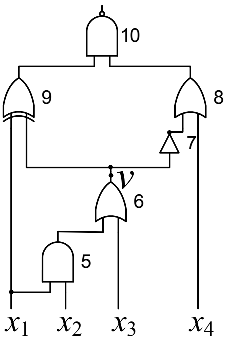

Fig. 1. The combinational circuit C and pole v.

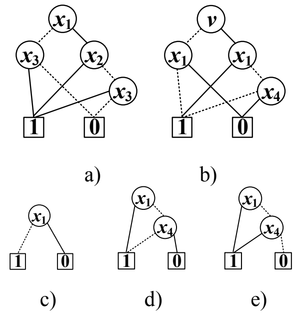

Fig. 2. a) BDD $R^{cont}(1)$; b) BDD $R(C_v)$; c) BDD $R(f^{v=0})$; d) BDD $R(f^{v=1})$; e) BDD R^{obs}.

It is known, that obtaining $\overline{R(f^{v=0})}$ ($\overline{R(f^{v=1})}$) from $R(f^{v=0})$, ($R(f^{v=1})$) is reduced to permutation of terminal nodes of the corresponding BDDs. BDD R^{obs} is represented by Fig. 2e.

$p(R^{cont}(1)) = ½ \cdot (½ \cdot 1 + ½ \cdot ½) + ½ \cdot (½ \cdot 0 + ½ \cdot 1) = 0.625$,

$p(R^{obs}) = ½ \cdot 1 + ½ \cdot (½ \cdot 1 + ½ \cdot 0) = 0.75$.

Take into consideration that random estimations are obtained by using structural description of a combinational part. This part behavior contains the working area, however they do not coincide. The point is that TC may be triggered just out of working area. If we know STG description from which the sequential circuit is obtained by applying the proper encoding states, we may calculate precisely random estimations of controllability out of working area. As for precise random observability estimations they are always calculated by using structural description of a combinational part.

III. Deriving precise controllability estimations out of working area

A sequential circuit behavior can be represented by STG. In order to synthesize a sequential circuit we need to encode internal states of STG. The result of encoding is the system of incompletely specified Boolean functions. By transforming this system to completely specified Boolean functions system we increase the number of possibilities for TC inserting. The problem is that minimization of the system of completely specified Boolean functions usually makes both set-off and set-on areas larger in comparison with the system of incompletely specified Boolean functions. As a result, some of the full states (which depend on both input and state variables) fall out of the working area (it is represented by STG). Normal circuit verification and testing in the working area cannot reach these full states; therefore, they can be used for triggering TC, which will be impossible to detect in the above-mentioned way. We intend to calculate 1(0)-controllability precise estimations for internal nodes out of the working using the precise 1(0) controllability estimations derived from structural description of a combinational part of a sequential circuit.

STG is known to be a representation of Finite State Machine (FSM) behavior in which symbols of input and output alphabets are encoded. Consider an example of STG (Table I). Here, x_1, x_2, x_3 are input variables of the circuit and $y_1, \dots, y_5$ – output variables. The table has 4 columns. The first represents input cubes (ternary vectors). The second represents current states. The third column represents following states. The forth column contains output vectors.

TABLE I. STATE TRANSITION GRAPH

$x_1x_2x_3$	q	q	$y_1y_2y_3y_4y_5$
0—	1	1	0 0 0 1 0
−0−	1	1	0 0 0 1 0
1 1−	1	2	1 0 0 1 0
−−0	2	2	0 0 1 1 0
−−1	2	3	1 0 1 1 0
1 0−	3	3	0 1 0 0 0
0—	3	4	1 1 0 0 0
−1−	3	4	1 1 0 0 0
−−0	4	4	0 1 0 0 1
−−1	4	1	1 1 0 0 1

TABLE II. SYSTEM OF INCOMPLETELY SPECIFIED BOOLEAN FUNCTIONS

$x_1x_2x_3$	z_1z_2	z_1z_2	$y_1y_2y_3y_4y_5$
0−−	00	00	0 0 0 1 0
−0−	00	00	0 0 0 1 0
1 1−	00	01	1 0 0 1 0
−−0	01	01	0 0 1 1 0
−−1	01	10	1 0 1 1 0
1 0−	10	10	0 1 0 0 0
0−−	10	11	1 1 0 0 0
−1−	10	11	1 1 0 0 0
−−0	11	11	0 1 0 0 1
−−1	11	00	1 1 0 0 1

The system F of incompletely specified Boolean functions (Table II) is obtained by encoding all internal states with some code (for example with the minimum width code words). The second and the third columns of the table represent encoded internal states.

The first two columns provides the products of the Boolean system. In turn, last two columns contain functions representing next states and outputs of the sequential circuit correspondingly. Table II is used to derive gate's implementation of the combinational part C.

Note that cubes corresponding to the first and the second columns of Table II represent the working area of the sequential circuit. Form the SoP from these cubes. Derive BDD R^w from the SoP. Let R^{nw} be an inversion of R^w. Then 1(0)-controllability out of working area may be calculated using BDDs:

$$R^{cont\,nw}(1) = R^{cont}(1)R^{nw} \quad \left(R^{cont\,nw}(0) = R^{cont}(0)R^{nw}\right).$$

The set V of suspicious nodes is constructed from all the nodes 1(0)-controllability estimations of which are less than chosen threshold, but greater than 0 (for $\mathrm{R}^{contnw}(1)$ or $\mathrm{R}^{contnw}(0)$).

We may cut set V using precise calculations of node v* observability (node v* is connected with TC output). If the precise estimation of node v* observability is more than the proper threshold, we exclude corresponding node v from further consideration. Node v may be also excluded from consideration if there is no rather short transfer sequence triggering TC with input v and output v*.

IV. FINDING TRANSFER SEQUENCE FOR A SET OF INTERNAL STATES

Execute multiplication BDD $\mathrm{R}^{contnw}(1)$ or $\mathrm{R}^{contnw}(0)$ for node v and BDD R^{obs} for node v*. The multiplication result is represented by BDD R^f.

The paths from R^f root to the 1 terminal compose sets of full states of sequential circuit. To trigger TC, one have to reach any of these states.

By abstracting input variables from R^f, we obtain BDD R^{S_0} of internal states (which depend only on state variables), that upon reaching provide malicious action TC. The procedure of finding existence evidence of a transfer sequence (the length is not more preset value l) for some state from a set presented by BDD R^{S_0} is described in detail in [4]. In this algorithm we did not derive the sequence itself but only set up its existence. Exclude some nodes from V that have no transfer sequence with length not more l.

Then we may find the transfer sequence itself for each node of the obtained set V using algorithm [5]. Both algorithms are oriented to cutting calculations but the algorithm in [5] is more complicate in comparison with the algorithm represented in [4]. Applying the derived transfer sequences for set V we may detect node v in which TC is inserted. Based on the result we may mask TC attack.

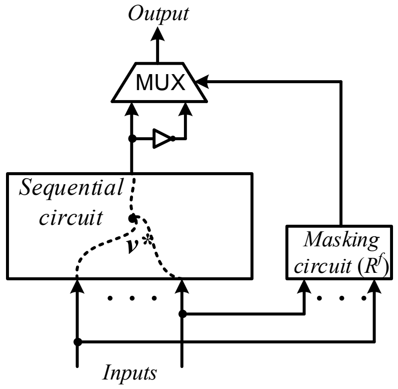

Fig. 3. Masking TC scheme inserted into out of working area.

V. TROJAN CIRCUIT MASKING

The masking sub-circuit together with MUX are placed out of a sequential circuit area (Fig. 3).

The masking sub-circuit implements function represented by BDD R^f. Connecting the proper output with MUX we keep correct behavior of a sequential circuit. Note that masking TC we don't interfere inside of sequential circuit we use only its inputs and outputs.

VI. Experimental results

We have performed experiments on MCNC [6] sequential benchmark circuits. The set of circuits has been made from corresponding FSMs (in KISS2 format) by minimum width encoding of states and using of a logic synthesis and optimization in ABC system [7].

For experiments we have limited to TCs which can be inserted into internal nodes with low controllability estimations without taking into consideration observability estimations. This approach is suited for any type of TC. When we know the type, we may use more simple BDDs R^f and consequently to cut overhead.

Experiments show that for each internal node with low controllability there exists rather short transfer sequence [4] *triggering TC. For the benchmark circuits considered the transfer sequence lengths are not more than 8 (in average 1.1).*

Calculations of controllability estimations for internal nodes of combinational part of sequential circuits and overhead estimations of masking sub-circuits out of the working area (Table IV) are executed. Names of benchmark (Circuit), number of gates (N_Gs), minimum nonzero value of controllability estimation (Min_VC), number of gates (their output poles) with low value (< 0,05) of controllability estimation (N_Gs*), part of gates with low value of controllability in percentage (%_Gs*), size of minimum masking sub-circuit for nodes with low controllability estimation (Min), size of maximum masking sub-circuit for nodes with low controllability estimation (Max), size of minimum masking sub-circuit as a percentage of initial circuit (%_Min), size of maximum masking sub-circuit as a percentage of initial circuit (%_Max) are represented in Table III.

Benchmark circuits and masking sub-circuits are received in ABC and they consist of 2-input logic-gates.

TABLE III. Experimental results for TC out of working area

Circuit	N_Gs	Min_VC	N_Gs*	%_Gs*	Min	Max	%_Min	%_Max
cse	219	0.000976562	136	62.1	1	11	0.5	5.0
dk14	96	0.015625	20	20.8	1	3	1.0	3.1
dk16	287	0.0078125	134	46.7	2	6	0.7	2.1
ex1	242	0.00195312	97	40.1	3	14	1.2	5.8
keyb	239	0.000976562	115	48.1	2	17	0.8	7.1
kirkman	156	0.00194741	37	23.7	2	12	1.3	7.7
sand	540	0.00012207	451	83.5	2	6	0.4	1.1
sse	139	0.00390625	82	59.0	1	4	0.7	2.9
styr	591	0.000488281	297	50.3	2	6	0.3	1.0
tbk	844	0	0	0.0	-	-	-	-
train11	77	0.015625	14	18.2	1	11	1.3	14.3

Masking of TC for out of working area requires rather small overhead (in average from 0.8% to 5.0%). The circuit “tbk” has no internal nodes with nonzero value of controllability estimation for TC because out of working area (don’t care set) is empty. TCs can’t be inserted into out of working area for this circuit.

VII. Conclusion

Possibilities of triggering TC out of working area are examined. The investigation is based on getting precise estimations of internal node controllability out of working area and precise calculating observability estimations by using structural combinational part description. The methods of getting precise estimations may be used for comparison with results of the different heuristics methods. The approach to TCs detection may be applied when they are not detectable during sequential circuit verification and testing in working area. Experiments on benchmarks show applicability of the suggested approach. The technique of masking TCs out of working area is proposed. Masking circuits overhead for chosen internal nodes of the benchmarks considered are rather small.

References

[1] R. Karri, J. Rajendran, K. Rosenfeld, and M. Tehranipoor, “Trustworthy Hardware: Identifying and Classifying Hardware Trojans,” *Computer*, vol. 43, no. 10, pp. 39–46, Oct. 2010.

[2] R. S. Chakraborty, S. Pagliarini, J. Mathew, R. S. Ranjani, and M. N. Devi, “A Flexible Online Checking Technique to Enhance Hardware Trojan Horse Detectability by Reliability Analysis,” *IEEE Trans. Emerg. Top. Comput.*, vol. PP, no. 99, pp. 1–1, 2017.

[3] A. Matrosova, S. Ostanin, and I. Kirienko, “Generating all test patterns for stuck-at faults at a gate pole and their connection with the incompletely specified Boolean function of the corresponding subcircuit,” in *2014 14th Biennial Baltic Electronic Conference (BEC)*, 2014, pp. 85–88.

[4] A. Y. Matrosova, I. E. Kirienko, V. V. Tomkov, and A. A. Miryutov, “Reliability of Physical Systems: Detection of Malicious Subcircuits (Trojan Circuits) in Sequential Circuits,” *Russ. Phys. J.*, vol. 59, no. 8, pp. 1281–1288, Dec. 2016.

[5] A. Matrosova, V. Andreeva, and A. Melnikov, “ROBDDs application for finding the shortest transfer sequence of sequential circuit or only revealing existence of this sequence without deriving the sequence itself,” in *2016 IEEE East-West Design Test Symposium (EWDTS)*, 2016, pp. 1–4.

[6] S. Yang, *Logic Synthesis and Optimization Benchmarks User Guide Version 3.0*. 1991.

[7] *ABC: A System for Sequential Synthesis and Verification*. (http://www.eecs.berkeley.edu/~alanmi/abc/).