ISBN: 978-1-6654-4502-3

Part Number: CFP21DTW-USB

2021 IEEE East-West Design & Test Symposium (EWDTS) Proceedings

Batumi, Georgia, September 10 – 13, 2021

Proceedings of 2021 IEEE East-West Design & Test Symposium (EWDTS)

IEEE Catalog Numbers:
XPLORE COMPLIANT: CFP21DTW-ART
ISBN: 978-1-6654-4503-0

USB: CFP21DTW-USB
ISBN: 978-1-6654-4502-3

Additional copies of this publication are available from:
Curran Associates, Inc
57 Morehouse Lane
Red Hook, NY 12571 USA
Phone: (845) 758-0400
Fax: (845) 758-2633
E-mail: curran@proceedings.com

CONTENTS

Applying Incompletely Specified Boolean Functions for Patch Circuit Generation

A. Matrosova
National Research Tomsk State University
Tomsk, Russia
mau11@yandex.ru

V. Provkin
National Research Tomsk State University
Tomsk, Russia
prowkan@mail.ru

***Abstract*— We consider combination circuit *C* and some its nodes that faults are detected on the last stages of circuit fabrication. Besides, injections of Trojan Circuits (TCs) in certain circuit *C* lines may be also detected. (It is assumed that TC payload output is inserted into a line of a combinational circuit). In both cases circuit *C* improper functions are changed for incompletely specified Boolean functions correlated with the corresponding fault nodes or the circuit lines of correct circuit C. That is why masking of circuit faults and TCs injections may be reduced to implementation of a patch circuit that realizes the correct system of incompletely specified Boolean functions, in particular, depending on internal circuit C variables. The correct system is calculated with using SAT solver. Patch circuit inputs are connected with internal nodes of circuit C that precede both fault nodes and the lines. Patch circuit outputs are connected with nodes that are fed either by fault nodes or by circuit lines in which Trojan circuits are inserted. Experimental results are executed on ISCAS test circuits. The results show that patch circuits as a rule are simpler than duplication subcircuit with fault node as outputs and preceding nodes as inputs.**

Keywords—combinational circuits, incompletely specified Boolean functions, patch (masking) circuits, Trojan Circuits

I. INTRODUCTION

In this paper we consider only one of the problems of ECO technologies [1, 2], namely, problem of forming patch circuit for masking fault nodes of set *V*. Correct information on circuit *C* behavior is represented by its structural description.

Commonly forming a patch circuit is based on results of circuit *C* simulation on some sub-set of input Boolean vectors [1, 2]. The last means that it is possible to guarantee the correct behavior of circuit *C* only on simulated sub-set of vectors. We suggest to derive a patch circuit using incompletely specified Boolean functions for nodes from set *V*. In this case, circuit *C* malfunction of node *v* from *V* may be detected on some input Boolean vectors (at least on one of them) during simulation. In the frame of our approach it is enough only to reveal that a malfunction on the considering node takes place. Main distinction of our method from ones suggested in [1, 2], is as follows. Our method guarantees correct behavior of circuit *C* on all input Boolean vectors with using the obtained patch circuit. It is because of knowledge of incompletely specified Boolean functions in terms of internal nodes corresponding to the considered internal nodes of correct circuit C. In this paper we show that ECO technologies may be also applied for masking of Trojan Circuit injections.

We calculate on-set and off-set of incompletely specified Boolean function of node *v* from *V* representing them by two SoPs. These SoPs depend on internal variables of circuit *C* from set *U*. It is necessary to get these SoPs for each node *v* from *V*. Irredundant SOP, realizing incompletely specified Boolean function of node *v* in terms of preceding internal nodes, is derived with using ESPRESSO algorithm, which is available both as standalone utility and part of ABC system. Obtained SOP is applied in ABC to derive patch circuit.

The rest of the paper is organized as follows. In section II the problem statement is considered. Section III contains detailed description of the proposed algorithm of incompletely specified function computation. In Section IV the approach to deriving of patch circuit is described. In Section V experimental results are discussed. Section VI concludes the paper.

II. PROBLEM STATEMENT

A combinational circuit *C* is considered. A set *V* of internal nodes is given the circuit behavior on which differs from the correct one. The correct behavior of circuit *C* is represented by its structural description. Different behavior of nodes from set *V* may be generated either by logical faults of circuit *C* gates outputs of which are nodes from set *V* or by Trojan Circuits (TCs) injected into lines that run to nodes from set *V*. In the last case the nodes are inputs of circuit *C* gates. It is necessary to provide the correct circuit behavior by masking faults of these nodes in the frame of ECO technologies. We suppose that patch circuit inputs are connected with internal nodes of circuit *C* corresponding to variables from set *U*. Outputs of the patch circuit are linked with nodes that are fed either by nodes from *V* or by lines. It is desirable to get masking circuit as simple as possible and simplify calculations of system of incompletely specified Boolean functions for set *V* of internal nodes. Remind that these functions depend on intenal variables of set *U*. We don't discuss here how to determine set *U* when set *V* is given.

Incompletely specified function in terms of internal variables of set *U* can be computed through mapping incompletely specified function in terms of primary input variables by using operations on ROBDDs [3]. But for some circuits sizes of ROBDDs are very huge: the number of ROBDD internal nodes is exponential function of circuit inputs number. It means that in this case ROBDDs cannot be derived for circuits with several tens or hundreds inputs. Computation of the incompletely specified function in terms of primary

978-1-6654-4503-0/21/$31.00 ©2021 IEEE

inputs followed by the above mentioned mapping is feasible with using SAT solvers, but run time of this computation is often inappropriate long. In this paper we propose algorithm of calculation of incompletely specified function in terms of internal variables without prior computation of incompletely specified function in terms of primary input variables. We assume that a patch circuit has not so many inputs (up to twenty).

III. Computation of Incompletely Specified Function in Terms of Internal Nodes

Let M_0, M_1 be off-set and on-set of incompletely specified Boolean function and M_* is don't care area.

$B=\{0,1\}$ is Boolean set, $B^k=\underbrace{B\times B\times\ldots\times B}_{k\ times}$ is set of all Boolean vectors of k length.

The suggested algorithm consists of three steps:

1. Forming set W of all reachable Boolean vectors on internal variables u_1, …, u_k from set U in circuit C for node v from V.
2. Splitting set W in two sets W_0 and W_1.
3. Extracting sub-sets M_0 and M_1 from sets W_0 and W_1, accordingly.

Further detailed description of these steps is considered.

A. Forming a set of all reachable Boolean vectors

Note that not each vector from B^k is reachable on internal nodes corresponding to set U of circuit C. Further we mark variable and corresponding node by the same symbol for simplicity. Let $\alpha=(\alpha_1,\alpha_2,\ldots,\alpha_k)\in B^k$ be Boolean vector that reachability is necessary to check. This vector is reachable when there is a root of the following system of Boolean equations:

$$\begin{cases} f_{u_1}(x_1,x_2,\ldots,x_n)=\alpha_1 \\ f_{u_2}(x_1,x_2,\ldots,x_n)=\alpha_2 \\ \quad\ldots \\ f_{u_k}(x_1,x_2,\ldots,x_n)=\alpha_k \end{cases} \quad (1)$$

Here f_{u_i} is the function implementing by sub-circuit $C_{u_i}(x_1,x_2,\ldots,x_n)$ with output node u_i and inputs x_1, x_2, …, x_n (primary inputs of circuit C). In order to solve system (1) first Tseitin transformation is applied to each sub-circuit $C_{u_i}(x_1,x_2,\ldots,x_n)$, i = 1, 2, …, k. Let formula $C_{u_i}(x_1,x_2,\ldots,x_n,u_i)$ be CNF for sub-circuit $C_{u_i}(x_1,x_2,\ldots,x_n)$ (internal variables of this formula are omitted.

Then we form CNF for system (1) as a whole. $C(x_1,x_2,\ldots,x_n,u_1,u_2,\ldots,u_k)=C_{u_1}(x_1,x_2,\ldots,x_n,u_1)\wedge$

$\wedge C_{u_2}(x_1,x_2,\ldots,x_n,u_2)\wedge\ldots\wedge C_{u_k}(x_1,x_2,\ldots,x_n,u_k)$

multiplying sub-circuits CNFs $C_{u_i}(x_1,x_2,\ldots,x_n,u_i)$. After that we substitute values $\alpha_1, \alpha_2, \ldots, \alpha_k$ instead of variables u_1, u_2, …, u_k, accordingly. Then we solve SAT problem for obtained formula. If CNF is satisfiable, there is a root of system (1), and vector $\alpha=(\alpha_1,\alpha_2,\ldots,\alpha_k)$ is reachable. Otherwise, it is unreachable. Doing that operation for all vectors of set B^k, we form a set W of all reachable vectors on internal nodes u_1, u_2, …, u_k.

B. Splitting a set of reachable vectors

Now it is necessary to split reachable vectors of set W in two sets: W_0 and W_1. Denote $C_v(u_1,u_2,\ldots,u_k)$ as sub-circuit of circuit C with output node v and input nodes u_1, u_2, …, u_k. Set W_0 (W_1) consists of Boolean vectors that turn into 0 (1) output of sub-circuit $C_v(u_1,u_2,\ldots,u_k)$. In order to split set W, we use logic simulation for this sub-circuit. It is necessary to execute logic simulation of all vectors of set W.

C. Extracting on-set and off-set of incompletely specified function in terms of internal variable $u_1, \ldots, u_k$ of circuit C

Step 3 of the above mentioned algorithm is reduced to extracting sub-sets M_0 and M_1 (off-set and on-set of incompletely specified Boolean function in terms of internal variables) from sets W_0 and W_1, correspondingly. Boolean vector $\alpha=(\alpha_1,\alpha_2,\ldots,\alpha_k)\in W_0\ (W_1)$ belongs to set M_0 (M_1) when changing value in node v from 0 (1) to 1 (0), under condition: $u_1=\alpha_1, u_2=\alpha_2,\ldots,u_k=\alpha_k$, alters output value of the circuit C (in case of multi-output circuit — value of at least one output). These conditions mean that vector α is a test pattern for stuck-at 1 (stuck-at 0) fault of node v. The following steps have to be done for extracting sub-sets M_0 and M_1 from sets W_0 and W_1, correspondingly.

1. Firstly, we form sub-circuit $C_{y_i}(v,x_1,x_2,\ldots,x_n,u_1,u_2,\ldots,u_k)$ with output y_i (primary output of circuit C) and inputs v, x_1, x_2, …, x_n, u_1, u_2, …, u_k (internal nodes u_1, u_2, …, u_k can be inputs of the formed circuit due to branching). Let $f_i(v,x_1,x_2,\ldots,x_n,u_1,u_2,\ldots,u_k)$ be the function implementing by circuit $C_{y_i}(v,x_1,x_2,\ldots,x_n,u_1,u_2,\ldots,u_k)$.

2. Then we form miter-circuit for output y_i (figure 1) that implements the function representing by the formula: $f_i(0,x_1,x_2,\ldots,x_n,u_1,u_2,\ldots,u_k)\oplus f_i(1,x_1,x_2,\ldots,x_n,u_1,u_2,\ldots,u_k)$ This formula is Boolean difference on variable v.

3. Next we need to substitute values α_1, α_2, …, α_k instead of variables u_1, u_2, …, u_k in the Boolean difference. Denote the result as $f_i(x_1, x_2, \ldots, x_n)$:

$$f_i(x_1,x_2,\ldots,x_n)=f_i(0,x_1,x_2,\ldots,x_n,\alpha_1,\alpha_2,\ldots,\alpha_k)\oplus$$
$$\oplus f_i(1,x_1,x_2,\ldots,x_n,\alpha_1,\alpha_2,\ldots,\alpha_k)$$

This function takes 1-value on input vectors of circuit C, for

which changing value of node v, under condition $u_1 = \alpha_1, u_2 = \alpha_2, \ldots, u_k = \alpha_k$, alters output value of i-th output of circuit C.

4. For multi-output circuit, we construct miter-circuit as shown in figure 2. Corresponding function is as follows: $f^{mit}(x_1, x_2, \ldots, x_n) = \bigvee_{i=1}^{m} f_i(x_1, x_2, \ldots, x_n)$.

It takes 1-value on such input vectors of circuit C, for which changing value of node v, under condition $u_1 = \alpha_1, u_2 = \alpha_2, \ldots, u_k = \alpha_k$, alters output value at least of one (any) output of circuit C (possibly several outputs).

5. It is also necessary to provide reachability of values $\alpha_1, \alpha_2, \ldots, \alpha_k$ on internal nodes $u_1, u_2, \ldots, u_k$. for input vectors of circuit C. It means that vectors of variables $x_1, x_2, \ldots, x_n$ have to be roots of system (1).

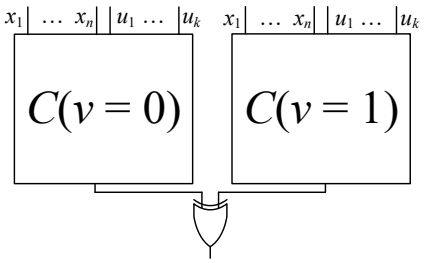

Fig. 1. Miter-circuit for one output circuit C

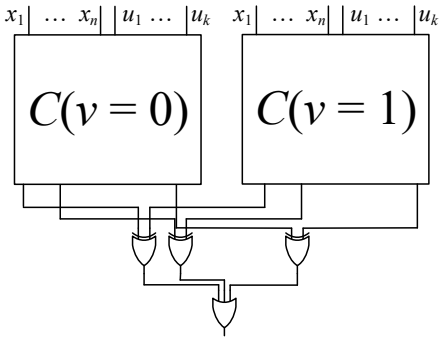

Fig. 2. Miter circuit for multy output circuit C

Thus Boolean vector $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_k) \in W_0\ (W_1)$ belongs to set M_0 (M_1) when there is a root for the system of Boolean equations:

$$\begin{cases} f^{mit}(x_1, x_2, \ldots, x_n) = 1 \\ f_{u_1}(x_1, x_2, \ldots, x_n) = \alpha_1 \\ f_{u_2}(x_1, x_2, \ldots, x_n) = \alpha_2 \\ \ldots \\ f_{u_k}(x_1, x_2, \ldots, x_n) = \alpha_k \end{cases} \quad (2)$$

If system (2) has no root, vector α belongs to set M_-. In order to solve system (2), Tseitin transformation is applied for each equation in system (2). Let $C(x_1, x_2, \ldots, x_n, d)$ be CNF of the first equation (variable d is output of miter-circuit), $C_{u_i}(x_1, x_2, \ldots, x_n, u_i)$ be CNFs for the rest equations. We form CNF

$$C^*(x_1, x_2, \ldots, x_n) = C(x_1, x_2, \ldots, x_n, d = 1) \wedge$$
$$\wedge C_{u_1}(x_1, x_2, \ldots, x_n, u_1 = \alpha_1) \wedge$$
$$\wedge C_{u_2}(x_1, x_2, \ldots, x_n, u_2 = \alpha_2) \wedge \ldots \wedge C_{u_k}(x_1, x_2, \ldots, x_n, u_k = \alpha_k)$$

and solve SAT problem for obtained formula. If CNF is satisfiable, there is input vector of variables $x_1, x_2, \ldots, x_n$, on which changing value of node v under condition $u_1 = \alpha_1, u_2 = \alpha_2, \ldots, u_k = \alpha_k$, alters output value of at least one (any) output of circuit C, and this vector delivers values $\alpha_1, \alpha_2, \ldots, \alpha_k$ to internal nodes $u_1, u_2, \ldots, u_k$. In this case $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_k) \in W_0\ (W_1)$ belongs to set M_0 (M_1). Otherwise, $\alpha \in M_-$.

IV. Patch circuit synthesis

On-set and off-set of incompletely specified Boolean function for node v in terms of internal nodes $u_1, u_2, \ldots u_k$ are represented by two sums of products (SoPs). Some algorithm of minimization has to be applied in order to get completely specified function realizing incompletely specified one. In our experiments we use ESPRESSO tool. Having generated irredundant system of SoPs implementing the incompletely specified Boolean functions we apply one of CAD tools (e. g. ABC) to derive the corresponding circuit. The patch circuit inputs are connected with internal nodes $u_1, u_2 \ldots, u_k$ of circuit C and the patch circuit outputs are connected with nodes of circuit C that are fed by fault nodes and the lines. For one fault node v we have the following picture (Fig. 3).

For set V of fault nodes it is necessary to get system of incompletely specified Boolean functions and their implementation by irredundunt system of SoPs. The obtained system then is used for deriving multy output patch circuit.

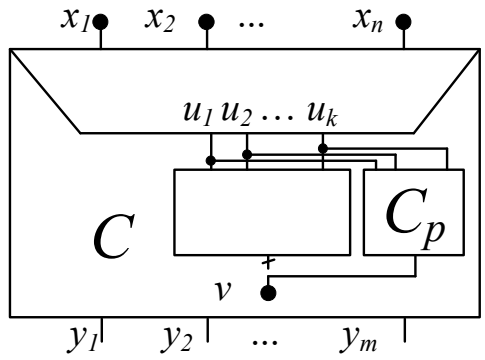

Fig. 3. Applying of patch circuit

V. Experimental results

For our experiments, we use circuits from ISCAS benchmarks, having transformed them to AIG format. In these experiments we suppose that only one node may be fault or only one Trojan Circuit may be injected.

Table 1 shows complexity of the derived patch circuits. In the column "Internal nodes" the number of internal variables u_1, u_2, …, u_k that are inputs of the patch circuit are represented. The column "Size of circuit" presents gates number of sub-circuit $C_v(u_1,u_2,\ldots,u_k)$. Output of this sub-circuit is node v and inputs are nodes u_1, u_2, … u_k. Note that duplicate of this sub-circuit could be also used as patch circuit but we want to get simpler one. In the column "Patch size" gates number of the derived patch circuit is represented. Table 2 shows comparison of run time of two algorithms: the first algorithm is based on computation of incompletely specified function in terms of primary inputs and its mapping on variables u_1, u_2, … u_k; the second algorithm is proposed in this paper. Implemented software was ran on laptop with AMD A8-5557M 2.1GHz CPU (Implementation is single-threaded).

TABLE I. Complexity of generated patch circuits

Circuit name	Primary inputs	Primary outputs	Gates count	Internal nodes	Size of circuit	Patch size
s6669	322	294	2433	15	27	20
				16	32	13
				12	18	16
				14	27	16
s9234	247	250	1888	7	9	2
s13207	700	790	2972	5	5	1
s15850	611	684	3719	5	6	2
				3	4	1

TABLE II. Comparison of run time algorithms

Circuit name	Internal nodes	Size of circuit	Mapping based algorithm run time	Proposed algorithm run time
s6669	15	27	2m 43s	10.72s
	16	32	2m 53s	20.46s
	12	18	4m 10s	1.38s
	14	27	2m 19s	5.43s
s9234	7	9	10m 38s	0.85s
s13207	5	5	5m 42s	0.82s
s15850	5	6	36m 58s	4.64s
	3	4	42m 14s	2.9s

Table 1 shows that using incompletely specified functions for internal nodes gives possibility to derive patch circuits that are better (in terms of gates count) than circuits obtained by duplication. Table 2 shows that for the same set of internal nodes the proposed algorithm runs faster in 1-2 order of magnitude in comparison with the algorithm that finds incompletely specified function in terms of primary inputs and maps it into incompletely specified function in terms of internal nodes.

Note that we may divide detected fault nodes into groups and obtain several masking sub-circuits depending on the corresponding sets of internal variables with sizes not more than twenty.

VI. Conclusion

This paper presents method of masking logic faults and Trojan Circuits injections based on computation of incompletely specified Boolean functions in terms of given set of internal nodes. The method is oriented to cut calculations when the number of internal nodes of patch circuit inputs is within two tens or so. We suppose that this restriction may be applied for real circuits, for example, by choosing the appropriate internal nodes as input ones for patch circuits for the corresponding fault nodes with sizes within twenty. Experimental results show that the suggested algorithm may be applied for circuit the corresponding ROBDDs of which are very huge. The method allows deriving patch circuits which are smaller than circuits obtained by duplication.

References

[1] A.-C. Cheng, H.-R. Jiang and J.-Y. Jou, "Resource-aware functional ECO patch generation," in Proc. DATE, 2016.

[2] A.Q. Dao, N.-Z. Lee, L.-C. Chen, M.P.-H. Lin, J.-H.R. Jiang, A. Mishchenko, and R. Brayton, "Efficient computation of ECO patch functions," in Proc. DAC, 2018.

[3] Matrosova A., Provkin V., Nikolaeva E. Masking Internal Node Faults and Trojan Circuits in Logical Circuits //Proceedings of 2019 IEEE East-West Design & Test Symposium (EWDTS), 13-16 september 2019, Batumi: IEEE, 2019. P. 416-419.

Camera-ready was prepared by Chumachenko S.
Approved for publication: 13.09.2021. Format 60x841/8.
Relative printer's sheets: . Circulation: 100 copies.
Published by SPD FL Stepanov V.V.
Ukraine, 61168, Kharkov, Ak. Pavlova st., 311