ATLANTIS PRESS

Advances in Intelligent Systems Research, volume 159

International Conference on Mathematics, Modelling, Simulation and Algorithms (MMSA 2018)

A Hardware Trojan Detection Technique Based on Rapid Trigger and Reducing Resource Consumption

Shenghua Yang[1,*], Lei Li[2], Jianhao Hu[3], Wanting Zhou[4] and Xueying Li[5]

[1]Space Integrated Circuit Laboratory of UESTC, Chengdu 611731, China
[2]Space Integrated Circuit Laboratory of UESTC, Chengdu 611731, China
[3]National Kay Lab. on wireless communications, Chengdu 611731, China
[4]Space Integrated Circuit Laboratory of UESTC, Chengdu 611731, China
[5]Space Integrated Circuit Laboratory of UESTC, Chengdu 611731, China
[*]Corresponding author

***Abstract*—The insidious of Hardware Trojans (HT) makes it hardly to active in the Integrated Circuit(IC). We designed a special structure, been named AND-AND or OR-OR, which is inserted in the critical points that HT is liable to insert by analyze the characteristics of the IC. Then the state turnover rate of these notes will be greatly improved, and the activate time of HT that driven by these notes will be reduced. The implementation results of UART benchmarks prove that this method can significantly increase the activity of Hardware Trojans and improve the detection efficiency. Compared with the previous method such as inserting Dummy flip-flop or MFTD structure, not only will our method reduce the number of input ports and the consumption of the area, but it will easier to control.**

Keywords—AND-AND; OR-OR; Trojan detection; transition probability; quickly activate

I. INTRODUCTION

Designers and users are gradually losing their control over integrated circuit design and manufacturing because of globalization. Malicious tampering with the original design is possible at any stages of IC design and manufacturing[1]. Such tampering can have devastating effects, such as information disclosure, function change, circuit damage and so on. Hardware Trojan will bring serious security risk in finance, national defense, medical treatment and the other key domains. The seriousness of these problems also makes the detection and prevention of Hardware Trojans more and more important.

However, because of the insidious of Trojans, detection of Hardware Trojans is a very challenging issue. In order to make the Trojan circuits more difficult to be detected, attackers tend to insert it into those circuits with low activity nodes[2]. In this way, unless under some specific or rarely happen circumstances, Trojans are not triggered in most cases. Then, the probability that Trojans were detected would be greatly reduced. From users' point of view, for purpose of avoiding the severely impact of hardware Trojans, they wish the circuits they used are no hardware Trojans, and no backdoors.

II. THEORY ANALYSIS OF FLIP PROBABILITY

The hardware Trojan has the characteristics of high concealment and low activation rate. Conventional testing will not work well at this time. When calculating the flip probability of a node, it can be modeled by geometrical distribution[3]. The geometric distribution is a discrete distribution and its probability function can be expressed as

$$P(\mathrm{n}) = P^*(1-P)^n (n=0,1...) \tag{1}$$

This function indicate that the node will turnover in the (n+1) clock cycle. The (P_0, P_1) is used to represent the probability of 0 or 1 for a node. Furthermore, the process of probability calculation follow three principles[4]. For a single logic gate, we can acquire its flip probability by analyzing its function. Taking two-input XOR gate for example, its input ports are a and b, the output is c, only a and b are reversed, the value of c will be 1. We suppose $(P_0\ P_1) = (1/2\ 1/2)$ is the value of transition probability for a and b, so the transition probability that c is equal to 0 is also 1/2. In a similar way, the output flip probability of other gates can be calculated. Figure I shows the flip probability of all nodes in a simple three-level combination circuit. It's also the trigger circuit of hardware Trojan that used in RS232-T100. We can see that the probability of c as 1 is 1/262144. As the circuit becomes larger and the number of level increasing, some nodes with lower flip probability may also exist.

If the Trojan designers use these nodes with low flip probability as hardware Trojan trigger condition, according to statistical principles, the probability of Trojan been triggered is $P = \prod_{i=1}^{R} P_i \ (P_i << 1)$. As a result, hardware Trojans will be more difficult to activate in routine functional verification tests. Based on this, a method that can controllably change the flip probability of those low activate nodes is proposed. By increasing the activity of these nodes, the hardware Trojan can be quickly activated and then be detected.

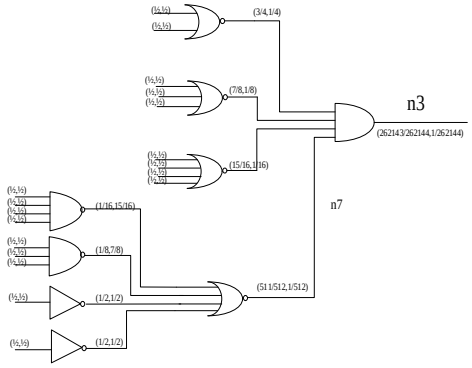


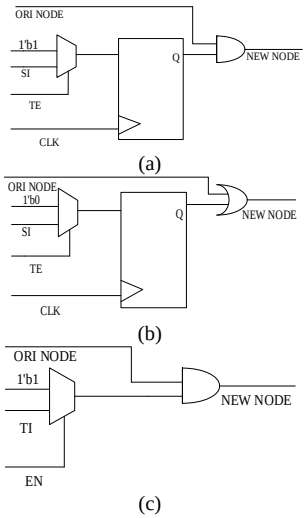
FIGURE I. RS232-T100 TROJAN TRIGGER CIRCUIT

III. AND-AND AND OR-OR STRUCTURES

From the above, conclusion can be draw that in a circuit, the probabilities that certain nodes are 0 or 1 will be greatly different, so the flip probability of such nodes will be very small. In this paper, we devise a new structure to improve the flip probability of such nodes.

The method we used was firstly proposed by [4] and some improvements were made by [5]. The key of this method is find the circuit nodes whose turnover probability are less than a certain value in the circuit network and then insert a special circuit structure at these nodes. Document [4] inserts a virtual scan flip-flop in the circuit and The literature [5] is a structure named MFTD. When these structures were inserted in the selected nodes, the turnover probability of these nodes will be greater than before. And the Trojan trigger probability will be improved at the same time. The two methods all increase the turnover rate of some nodes without changing the function and timing of the original design. The structure proposed in this paper has all common features of the two structures above, and it's also superior to them. Such as saving more area, reducing the number of ports, and easier to control. For large-scale design, the structure proposed in this paper has more obvious advantages.

Figure II (a) and (b) are the two kinds of Dummy flip-flop structure; Figure II (c) and (d) are the two structures of MFTD.



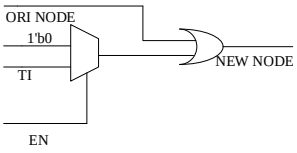


FIGURE II. (A) DUMMY FLIP-FLOP STRUCTURE(P_0<<P_1) (B) DUMMY FLIP-FLOP STRUCTURE($P_0 >> P_1$) (C) MFTD STRUCTURE ($P_0 << P_1$) (D) MFTD STRUCTURE ($P_0 >> P_1$)

From equation (1), the flip probability of a node can be expressed as $P_{\text{tra}} = P_0 \times P_1$. From this we can calculate the average flip cycle of the node is $T_{\text{tra}} = 1/P_{tra}$. The two have an inverse relationship, and the smaller the difference between them, the greater flip probability will be. Therefore, we designed AND-AND&OR-OR structure, as shown in Figure III (a), (b):

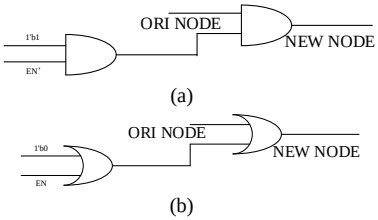


FIGURE III. (A) AND-AND STRUCTURE (B) OR-OR STRUCTURE

We assume that the probability distribution of node n is (P_0, P_1), and $P_0 >> P_1$, if

$$P_1 = \frac{K}{N}\ ,\ P_0 = 1 - P_1 = 1 - \frac{K}{N} (K \approx N, K < N)$$

then,

$$P_{\text{tra}} = P_0 \times P_1 \le \frac{(P_0 + P_1)^2}{4} \qquad (2)$$

Apparently, if $P_0 \approx P_1$, this inequality has maximum value. Then the transition probability has maximum value. Taking OR-OR structure as an example, after inserting the OR-OR structure, P_0 and P_1 are all close to 1/2. Therefore, the flip probability after inserting the OR-OR structure increases compared with the original turning probability.

When we insert OR-OR structure into figure I, we can get figure IV. The flip probability of node n3 in figure I is $P_{\text{tra}} = 262143/(262144)^2$. After inserting OR-OR structure, $P'_{\text{tra}} = 511/(512)^2$. By comparison, it can conclude that the flip probability of node n3 is about 512 times lager, then the activation time of hardware Trojan will be greatly reduced.

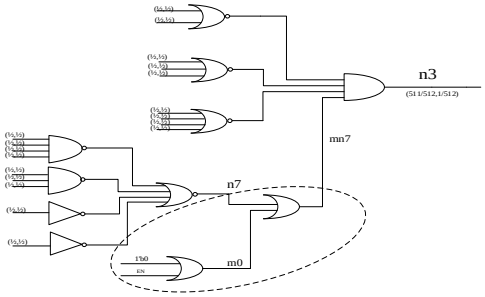


FIGURE IV. THE OR-OR STRUCTURE IS INSERTED INTORS232-T100 TROJAN TRIGGER CIRCUIT

Figure IV also shows how to insert the OR-OR structure into a circuit. For node n, if $P_0 >> P_1$, OR-OR structure is selected, the circuit works normally when EN = 0 and the inversion probability of node n increases when EN = 1; if $P_0 << P_1$, the AND-AND structure is selected, the circuit works normally when EN = 1, but when EN = 0, the flip probability of node n will increase. After analyzing node n7 we can found $P_0 >> P_1$, so OR-OR structure is selected, and connect to the circuit in dashed lines according to the figure. Similarly, if $P_1 >> P_0$, using AND-AND structure, inserting in the same way.

IV. VERIFICATION AND ANALYSIS

To validate the availability the proposed structure, We use the RS232 benchmarks provided in Trust-Hub, which described 10 kinds of Trojan prototype. We calculated the area of pure circuit and the circuit with different Trojans, and estimated the percentage of the Trojans. According to the random test, we extracted the node with a flip probability that less than 5%. For low transition nodes, comparing the sizes of P_0 and P_1. If $P_0 >> P_1$, inserting OR-OR structure; If $P_0 >> P_1$, inserting AND-AND structure.

After the synthesis of design compiler, we calculate the area ratio the Trojan account for. According to the power estimation method, we estimated the power consumption of Trojans by calculating the number of its equivalent gates in RS232-TX, the detailed data are in Table I.

TABLE I. COMPARE THE FILE LIST

Circuit Type	total area (um^2)	Area without Trojans (um^2)	Increased area (um^2)	Number of equivalent gate	Equivalent power consumption (uW)	Percentage of Trojans(%)
RS232-T100	2827.868341	2739.603540	88.264801	18	1.8	3.22
RS232-T200	3337.088310	2739.603540	597.48477	118	11.8	21.81
RS232-T300	4700.100527	2739.603540	1960.496987	385	38.5	71.56
RS232-T400	4847.774328	2739.603540	2018.170778	397	39.7	73.67
RS232-T500	4734.048509	2739.603540	1994.444969	392	39.2	72.80
RS232-T600	3007.792742	2739.603540	268.189202	53	5.3	9.79
RS232-T700	3019.674543	2739.603540	280.071003	56	5.6	10.22
RS232-T800	2785.433341	2739.603540	45.829801	10	1	1.67
RS232-T900	3070.596538	2739.603540	330.992998	65	6.5	12.08
RS232-T901	3072.293939	2739.603540	332.690399	66	6.6	12.14

TABLE II. LOW FLIP RATE NODE

Low-flipping nodes in RS232-T100		Low-flipping nodes after inserting OR-OR		Low-flipping nodes after inserting Dummy flip-flop		Low-flipping nodes after inserting MFTD structure	
Node name	Flip times	Node name	Flip times	Node name	Flip times	Node name	Flip times
uart/u_rec/n7	5	uart/u_rec/n7	5	uart/u_rec/n7	5	uart/u_rec/n7	5
uart/u_rec/n3	0	uart/u_rec/n3	600	uart/u_rec/n3	600	uart/u_rec/n3	600
		uart/u_rec/enn	0	uart/u_rec/enn	0	uart/u_rec/enn	0
		uart/u_rec/m0	0	uart/u_rec/m0	0	uart/u_rec/m0	0
		uart/u_rec/mn7	0	uart/u_rec/mn7	0	uart/u_rec/mn7	0
				uart/u_rec/SI	0	uart/u_rec/ti	0
				uart/u_rec/SI1	0	uart/u_rec/ti1	0

In the circuits, the Trojan trigger signals of RS232-T100 and RS232-T800 are consist of internal signals with low transition probability, and the output will be changed if Trojans are activated. RS232-T200, RS232-T300 and RS232-500 are triggered by a counter inserted into the circuits. When the counter is full, Trojans will be triggered. RS232-T400 will be triggered if the circuit send and receive a same data at the same time. RS232-T600, RS232-T700, RS232-T900 and RS232-T901 all use state machine as Trojan trigger condition. When the sequence previously set was detected, Trojans are triggered. These benchmarks are used for our research. And to acquire the low flip nodes, a special algorithm is used. According to value of $(P_0,\ P_1)$, we determine which structure will be inserted at those nodes.

Using RS232-T100 as an example, We send 200 8bit random numbers to the RS232-T100 and find out the nodes whose transition times are less than 5. They are node uart/u_rec/n7 and uart/u_rec/n3. Our analysis results show n3 is driven by n7. So we inserted the Dummy flip-flop, MFTD structure and our OR-OR structure at node n7 respectively. Through the VCS simulation, Low flip nodes are listed in Table II.

In Table II, compared with the original circuit of the RS232-T100, the extra nodes are introduced after inserting different structures. Comparing these three methods, we can see that all roads lead to Rome. This means the flip rate of the circuit node n3 is increased. As shown in Table III, we can see

the area increased only inserting the structures into one node of the circuits. Our structure takes the least area. From all of above, we can conclude the method proposed in this paper use least resources and least ports , and is easier to control because of fewer ports and no timing logic. It's obvious that as the number of low transition nodes increase, the superiority of our structure will be even more pronounced.

TABLE III. INCREASE AREA COMPARISON FOR ONE NODE

method	Total area	Increased area
Dummy flip-flop	2844.84	5.875%
MFTD structure	2703.96	0.632%
AND_OR structure	2692.08	0.189%
Original circuit	2686.98	/

Fig. V is a simulation diagram of VCS without increasing the n3 node rollover probability. Figure VI is a simulation diagram of VCS with OR-OR structure inserted at n7 node to increase the flip rate of n3. By comparison, it can be seen the flip rate of n3 is improved, and the Trojan driven by n3 is triggered. As a result, the final output of signal readyH is changed.

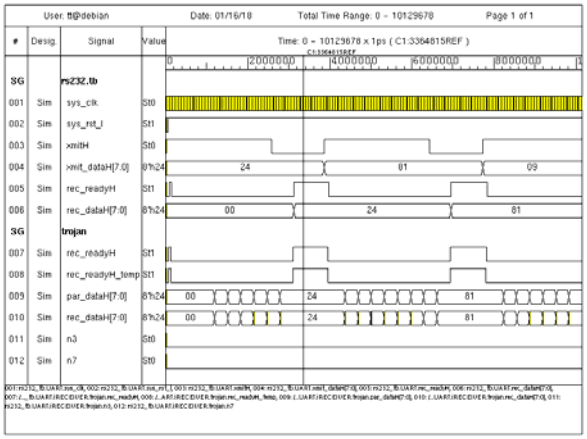


FIGURE V. SIMULATION DIAGRAM OF VCS WITHOUT INCREASING THE N3 NODE ROLLOVER PROBABILITY

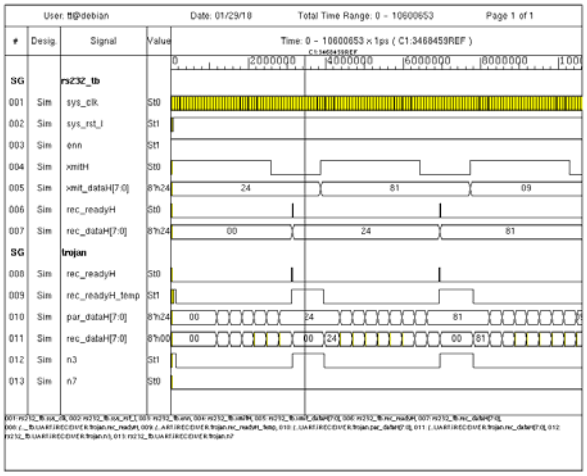


FIGURE VI. SIMULATION DIAGRAM OF VCS WITH OR-OR STRUCTURE INSERTED AT N7 NODE

V. CONCLUSIONS

In this thesis, we researched the low flip nodes in the circuits, and designed AND-AND&OR-OR structure based on this to improve the flip probability of these nodes, and demonstrated our design in RS232 benchmarks. The results show that this structure can indeed improve the flip probability of low transition nodes. Meanwhile, the activation time of Trojans will be greatly reduced. Compared with the previous proposed method like Dummy flip-flop and MFTD, the design we proposed uses less hardware resources, simpler structure, fewer ports and is easier to control. Our structure is designed for detecting Trojans with low trigger probability, for other kinds of Trojans, it may be invalid.

ACKNOWLEDGMENT

We appreciate the previous research by related researchers and this work is part of the project "The thirteenth Five-Year Plan pre-research---Credibility model and detection technology of Trojans".

REFERENCES

[1] Villasenor J D. Ensuring hasrware cybersecurity[M]. Center for Technology Innovation at Brookings,2011

[2] Chakraborty R S, Narasimhan S. Bhunia S, Hardware trojan: threats and emerging solutions [C] Proc of IEEE International High Level Design Validation and Test Workshop, 2009; 166-171

[3] D.D. Wackerly, W. Mendenhall III, and R.L. Scheaffer. Mathematical Statis tics with Application[M]. 7th edition, Thomson Learning, 2008

[4] Salmani H, Tehranipoor M, Plusquellic J. New design strategy for improving hardware Trojan detection and reducing Trojan activation time[C] IEEE International Symposium on Hardware-Oriented Sesurity and Trust (HOST). Francisco: IEEE, 2009:66-73

[5] Zhao Yiqiang,Feng Zizhu,Shi Yafeng ,etc. Approach for improving hardware Trojan detection by reducing Trojan activation time[J]. Journal of Huazhong University of Science and Technology(Natural Science Edition), 2014(6):85-89.

[6] Salmani H, Tehranipoor M, Plusquellie J. Anovel technique for improving hardware Trojan detection and reducing Trojan activation time [J]. IEEE Trans on Very Large Scale Integration Systems, 2012, 20(1): 112-125

[7] https://www.trust-hub.org/benchmarks.php