INDIAN INSTITUTE OF ENGINEERING SCIENCE AND TECHNOLOGY

SHIBPUR , HOWRAH

Submitted to the Department of Information Technology

For the final submission of major project for the degree of B.Tech. in Information Technology

Project: Analysis of trust hub for hardware trojan detection

NAME: **SUKANYA NASKAR, SHREYASI KARAK, ARATI SHAW**

ENROLL: 510816048, 510816022, 510816046

SEM: 8 TH YEAR: 4TH

UNDER GUIDANCE OF : DR. SURAJIT ROY

ACKNOWLEDGEMENT

I thank Dr. Surajit Roy who provided insight and expertise that greatly assisted the project. His suggestions have really helped in the creation of this project.

I would also like to show gratitude towards Mr.Tapobrata Dhar (Research Scholar under Dr. Surajit Roy) for his support during the course of this project.

INDEX:

1. Introduction……………………………………….3
2. History…………………………………………..4
3. Problem statement……………………………….5
4. Definitions……………………………………....5
5. Aim of project…………………………………...10
6. Algorithm………………………………………...12
7. Additional circuits……………………………….16
8. Result…………………………………………….18
9. Conclusion……………………………………….22
10.References………………………………………23

INTRODUCTION

Privacy remains a priority in any computational systems, circuits and programs for the proper functioning and security of any organization. This basic necessity is at treat when a seemingly harmless trojan attacks the system. These attacks, in the form of malicious modifications of electronic hardware at different stages of its life cycle, pose major security concerns in the electronics industry.

In this project we are analyzing a complex circuit to find out the low transition probability point where trojan could have been inserted in the circuit. Low transition point refers to the points in the circuit where activity is less than a certain predefined threshold. Low activity in this region makes the trojan circuit difficult to detect. Thus the purpose of trojan, i.e to remain undetected in the circuit and cause harm, is satisfied. We will be placing such a circuit at this low transition probability points, such the transition probability at this points increases and trojan inclusion by attackers become difficult or trojan is detected if present in the circuit.

HISTORY

The name Trojan Horse comes from the story of Trojan War about the subterfuge that the Greeks used to enter the independent city of Troy and win the war. In the canonical version, after a fruitless 10-year siege, the Greeks constructed a huge wooden horse, and hid a select force of men inside including Odysseus. The Greeks pretended to sail away, and the Trojans pulled the horse into their city as a victory trophy. That night the Greek force crept out of the horse and opened the gates for the rest of the Greek army, which had sailed back under cover of night. The Greeks entered and destroyed the city of Troy, ending the war.

Metaphorically, a "Trojan Horse" has come to mean any trick or stratagem that causes a target to invite a foe into a securely protected bastion or place. A malicious computer program that tricks users into willingly running it and remains unidentified or seems to be harmless is also called a "Trojan horse" or simply a "Trojan".

Similar to the mythological analogy, it should have 2 intentions:

1) it should have a malicious intent; and

2) it should evade detection under conventional post-manufacturing test/validation process.

PROBLEM STATEMENT

Analysis of trust hub circuit for hardware trojan detection

DEFINITIONS:

1. TROJAN :

A Hardware Trojan (HT) is a malicious modification of the circuitry of an integrated circuit. A hardware Trojan is completely characterized by its physical representation and its behavior. The payload of an HT is the entire activity that the Trojan executes when it is triggered. In general, malicious Trojans try to bypass or disable the security fence of a system: It can leak confidential information by radio emission. HT's also could disable, derange or destroy the entire chip or components of it.

In high security governmental IT departments, hardware Trojans are a well known problem when buying hardware such as: a KVM switch, keyboards, mice, network cards, or other network equipment. These are intentional faults in IC done during design or fabrication in an untrusted design house or foundry, which involve untrusted people, design tools, or components.

Such modifications can give rise to undesired functional behavior of an IC, or provide covert channels or backdoor through which sensitive information can be leaked.

2. TRUST HUB:

It is a circuit used to study the insertion of hardware Trojan, it’s detection, side-channel attacks analysis, vulnerability analysis, etc.

3. TRIGGER:

It refers to any method or action that activates a trojan. Trojans are placed in such points in the circuit where the activity is less, now on data flow through this circuit or component(even in rare cases), the trojan activates and leaks information, or may disrupt normal functioning and give wrong results/output of the circuit.

4. PAYLOAD:

It refers to the effect of the trojan on activation. The payload could range from a mechanism that presents dummy keys, predefined by the attacker, instead of the actual cryptographic keys used for sensitive encryption or signature verification operations, to leaking the secret hardware keys via covert side channels, e.g., information leaked through a power trace.

5. SIDE CHANNEL:

Side-channel analysis approaches depend on the measurement and analysis of physical “side-channel” parameters like power signature or path delay of an IC in order to identify a structural change in the design. Unlike logic testing, these approaches do not require Trojan activation in order to detect them. Side-channel analysis (SCA), primarily based on supply current, has been extensively investigated by large number of research groups and various solutions to increase the signal-to-noise (SNR) have been proposed.

6. TRANSITION PROBABILITY:

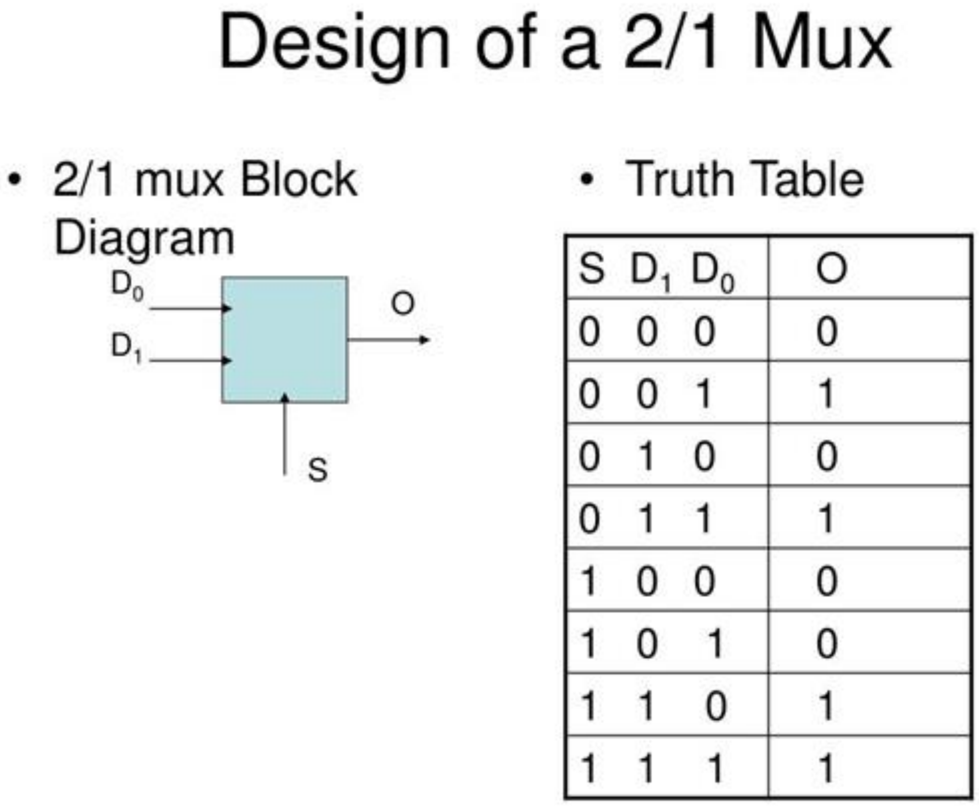
The *one-step transition probability* is the probability of transitioning from one state to another in a single step. It is product of signal output probability for zero and one for a particular gate.

7. THRESHOLD:

A benchmark for transition probability. Values below it are considered as low transition zones, ideal for trojan injection.

8. Introduction to MUX:

MUX is a combinational circuit which have many data inputs and single output depending on control or select inputs. For N input lines, log n (base2) selection lines, or we can say that for 2^n input lines, n selection lines are required. Multiplexers are also known as **“Data n selector, parallel to serial convertor, many to one circuit, universal logic circuit”.** Multiplexers are mainly used to increase amount of the data that can be sent over the network within certain amount of time and bandwidth



Design of a 2/1 Mux

- 2/1 mux Block Diagram

- Truth Table

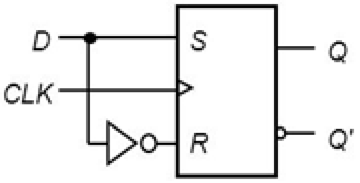
S	D_1	D_0	O
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

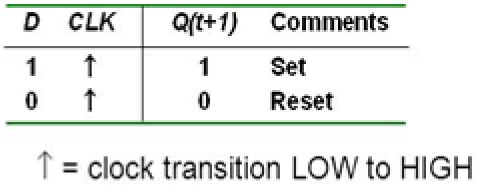
9. Introduction to Flip Flop:

Flip-flops and latches are used as data storage elements. A flip-flop is a device which stores a single bit (binary digit) of data; one of its two states represents a "one" and the other represents a "zero". Such data storage can be used for storage of state, and such a circuit is described as sequential logic in electronics. When used in a finite-state machine, the output and next state depend not only on its current input, but also on its current state (and hence, previous inputs). It can also be used for counting of pulses, and for synchronizing variably-timed input signals to some reference timing signal.

Flip-flops can be either level-triggered (asynchronous, transparent or opaque) or edge-triggered (synchronous, or clocked).

Truth table D flip flop





D	CLK	Q(t+1)	Comments
1	↑	1	Set
0	↑	0	Reset

↑ = clock transition LOW to HIGH

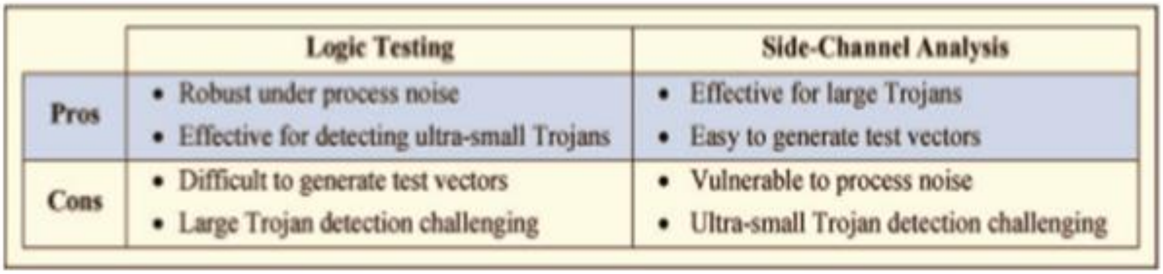
10 . TROJAN DETECTION :

The two major Trojan detection paradigms are:

- logic testing and
- side-channel analysis.

Logic testing approaches, both functional and structural, attempt to develop directed test patterns to activate unknown Trojan instances and propagating their effects to output ports. Although robust under process and measurement noise, these approaches are likely to fail to activate large Trojans consisting of large numbers of trigger inputs. An alternative approach is to measure a side-channel parameter, such as supply current or path delay, which can be affected due to unintended design modifications. However, the effective- ness of side-channel analysis is limited by large intrinsic device parameter variations in modern nanometer technologies. These detection approaches typically require a golden design or a set of golden ICs to compare the measured values in order to identify the Trojan-infected ones.

Here is a comparative study of Logic testing and side chain analysis.



	Logic Testing	**Side-Channel Analysis**
Pros	• Robust under process noise • Effective for detecting ultra-small Trojans	• Effective for large Trojans • Easy to generate test vectors
Cons	• Difficult to generate test vectors • Large Trojan detection challenging	• Vulnerable to process noise • Ultra-small Trojan detection challenging

AIM OF THE PROJECT

1. TO CALCULATE TRANSITION PROBABILITY:

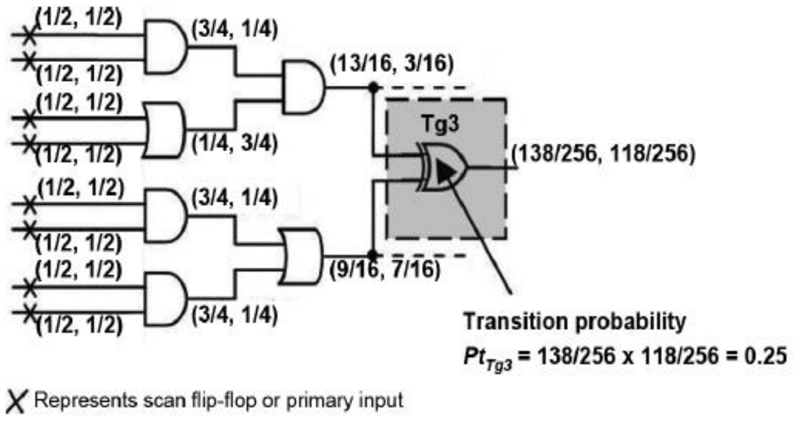
To calculate the transition probability we first consider the signal probability at the input terminal, i.e, the probability of 0 as input or 1 as input. Let the probability of 0 or 1 at input is equal ,i.e,0.5.

The probability of 0 and 1 at the output is determined by the respective truth table of the gate. Finally, the transition probability is calculated as multiplication of the probability of 0utput 0 and 1.

Eg: the signal probability of 0 and 1 are 0.5 each. Consider a AND gate , from the truth table ,the probability of 0 at output is 0.75 and that of 1 is 0.25.

So transition probability is calculated as :

For 0: 0.5*0.75=0.375 For 1: 1-0.375=0.625



2. TO FIND LOW ACTIVATION POINT:

Low activation regions are the gates where the value of transition probability are the low. We have set a threshold value as 0.15. All those gates having Tp less than 0.15 are considered as low activation points. These are the hotspots of trojan insertion. If the trojans are inserted here, then due very less frequency of data flow through these gates, the trojan remains unidentified.

Passing different test cases to check for erroneous result to detect presence of trojan will not work , since the probability of data flow through these erroneous circuit is very less, hence probability of erroneous output is also negligible when we consider some real circuit containing 50000 gates.

3. TO INSERT CIRCUIT COMPONENT THAT WILL INCREASE ACTIVITY OF THE REGION:

We aim to insert some circuit like MUX or flip-flop at these low Tp gates. This will enhance the Tp and thus the activity at these points will increase. As more data flow will be possible at these points, so the erroneous output will be easily visible. Hence, any trojan circuit at these points can be easily identified.

PROPOSED ALGORITHM

1. ALGORITHM TO CALCULATE LIST OF LOW TP POINTS :

```
INPUT : Circuit Netlist , Trigger Threshold ( $P_{th}$ )

OUTPUT : List of Low Transition Probability Nodes (L)

STEP 1 : For each gate in the circuit with  output “a” ( say ) :-

            $\text{Transition Probability}_a = (\text{signal probability of } 0)_a * (\text{signal probability of } 1)_a$

STEP 2 : For each transition probability ( $tp_i$ ) of the circuit

             If ( ( $tp_i < P_{th}$ )  :

                  Insert node i in L
```

2. ALGORITHM TO FIND INSERTION POINTS :

```
INPUT : Circuit Netlist , List of Low Transition Probability Nodes (L)

OUTPUT : List of Insertion Points (Lp)

STEP 1 : Convert the circuit to a directed graph where

            Node => gate of the circuit, incoming edges => inputs to a gate ,

            outgoing edges => outputs from a gate

STEP 2 : Transpose the graph

STEP 3 : For each node in L , run dfs and record the path

STEP 4 : Determine the common node of intersection (i) of all these paths with
minimum depth limit and insert I in Lp

[ D = a list of depth limits of I from each node in L

And   d = minimum(D) ]
```

```
STEP 5 : For each node n in L :-
            Determine nodes ($N_d$) at depth d from n
            If (at least one node from $N_d$ is present in Lp)
                Continue to check for the next n
            Else
                If ((signal probability of 0)$_n$ > ((signal probability of 1)$_n$):
                    Select one node from Nd where ((sp of 0)$_n$ < ((sp of 1)$_n$) and
                    insert in Lp
                Else If ((signal probability of 0)$_n$ < ((signal probability of 1)$_n$):
                    Select one node from Nd where ((sp of 0)$_n$ > ((sp of 1)$_n$) and
                    insert in Lp
```

EXAMPLE :

CIRCUIT :

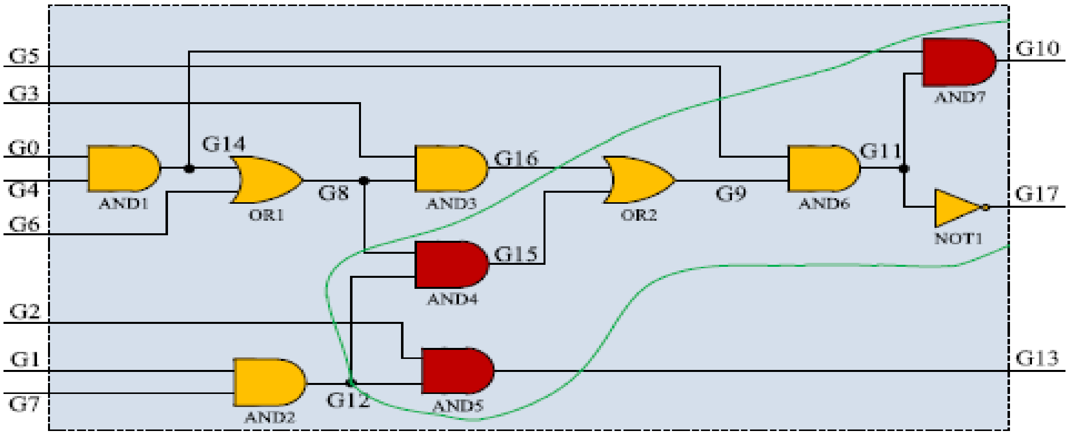


Fig. 2. Example circuit.

THRESHOLD TRANSITION PROBABILITY : 0.15

LOW TRANSITION PROBABILITY POINTS : { G10 , G13 , G15 }

INSERTION POINTS : { G12 , G14 }

TRANSITION PROBABILITY ANALYSIS APPLYING PROPOSED TECHNIQUE :

NET	ORIGINAL TRANSITION PROBABILITY (tp)	TRANSITION PROBABILITY AFTER INSERTING dsFFs (tp’)	((tp’ – tp)/ tp) %
10	0.04973500967025757	0.1875	276.9980165744863
13	0.109375	0.1875	71.42857142857143
15	0.1318359375	0.234375	77.77777777777777
11	0.1658773422241211	0.21185302734375	27.716675769684077
17	0.1658773422241211	0.25	50.713772386236236
14	0.1875	0.1875	0.0
12	0.1875	0.1875	0.0
16	0.21484375	0.234375	9.090909090909092
8	0.234375	0.1875	-20.0
9	0.24358749389648438	0.238037109375	-2.2786

ADDITIONAL CIRCUITS

1. HARDWARE CIRCUIT USED TO RAISE TRANSITION PROBABILITY AT LOW ACTIVITY REGIONS :

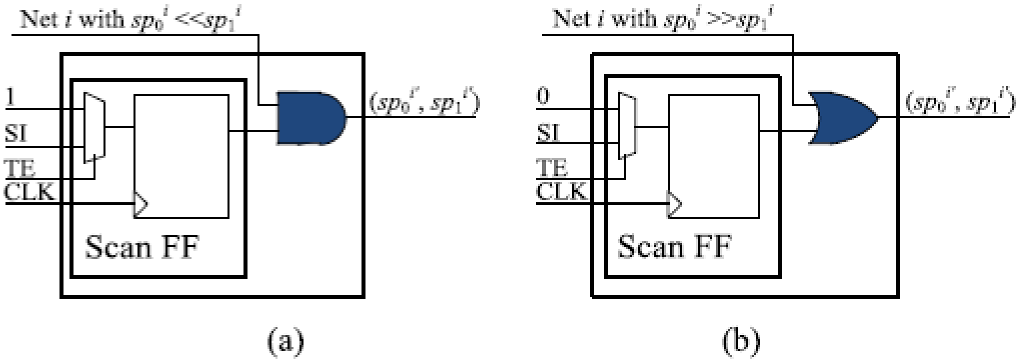


Fig. 1. dSFF structures when (a) $\mathrm{sp}_0^i << \mathrm{sp}_1^i$ and (b) $\mathrm{sp}_0^i >> \mathrm{sp}_1^i$.

The Dummy scan flip flop (dsFF) shown in Fig. 1.is inserted at the insertion points to raise the low transition probability values of the associated net. If signal probability of 0 on associated net N_i , $SP0_i$ is less than its signal probability of 1 i.e.$SP1_i$, an AND gate is connected to the output of the SFF and net N_i to increase $SP0_i$, as depicted in Fig. 1(a). Similarly, if $SP1_i$ is less than $SP0_i$, an OR gate is used to increase SP_i . When test enable (TE) is active (test mode), the output of the SFF is supplied by SI. If random test patterns are applied, the signal probabilities of 1 and 0 at the output of the SFF are 0.5. In this way, $SP1_i$ is reduced up to $0.5 \times SP1_i$ for the case shown in Fig. 1(a). Similarly, $SP0_i$ is reduced up to $0.5 \times SP0_i$ for the case shown in Fig. 1(b). Thus, after dSFF insertion, the transition probability of the fan-out nets can be improved.

2. HARDWARE TROJAN CIRCUITS USED FOR EXPERIMENTAL PURPOSE:

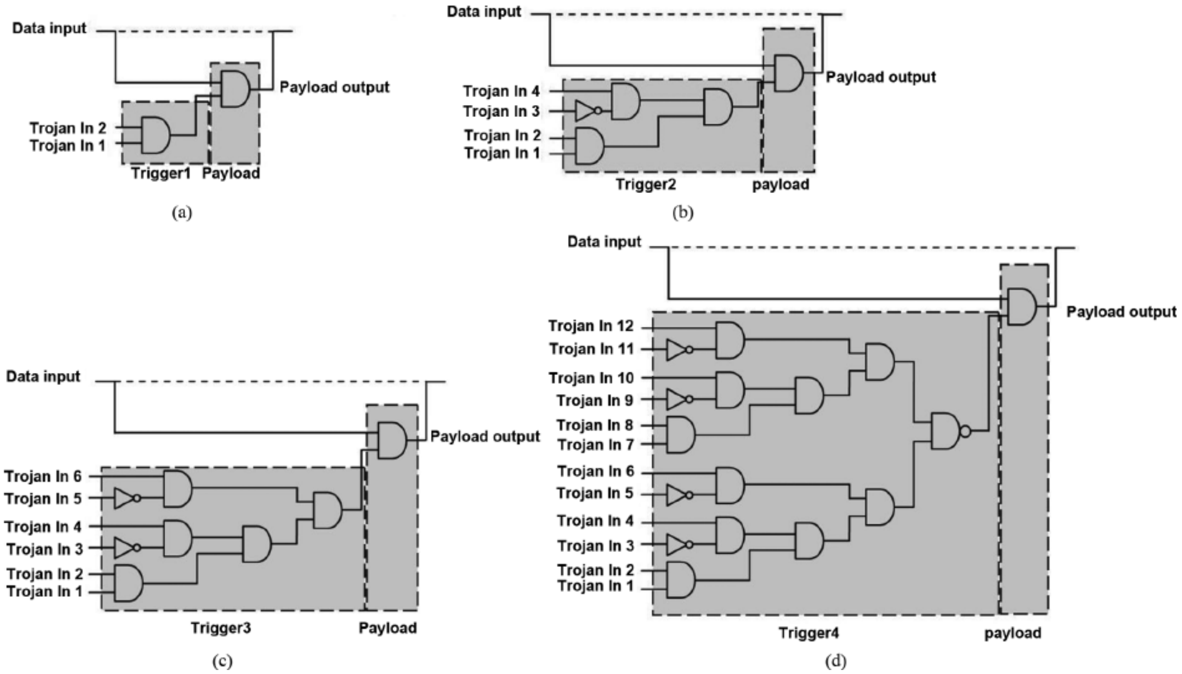


Fig 2: Trojan circuits – (a) Trojan 1 (b) Trojan 2 (c) Trojan 3 (d) Trojan 4

A Trojan circuit has two parts – Trigger and Payload.
Payload inputs come from Trigger output and data input which is part of the original circuit. The task of the comparators is to detect rare combinations of Trojan inputs based on their probabilities of 0 and 1. Selection of payload gate is done based on the dominant values of corresponding Trojan outputs. The dashed lines in the above figures represent the connection in the original circuit which is assumed to be connected through Trojan’s Payload by the attacker.

RESULTS

BENCHMARK CIRCUITS USED FOR THE PROJECT :

The proposed method has been implemented upon *s18580* , *s35932 , s38417 and s38584* circuits to check it's efficiency. We have shown the detailed analysis results on trojan activation and detection for s38417 circuit in Table 2 and 3. And for all the benchmark circuits we have estimated number of low tp points , insertion points , LINS, gate overhead and runtime in Table1 and their POC and RAC values in Table 4.

EXPERIMENTATION RESULTS :

Table 1. ***TIME COMPLEXITY AND CIRCUIT OVERHEADS AFTER APPLYING THE PROPOSED METHODOLOGY USING OPTIMUM THRESHOLD PROBABILITY***

BENCHMARK	**S15850**	**S38417**	**S35932**	**S38584**
OPTIMUM P_{TH}	**0.06**	**0.025**	**0.07**	**0.037**
NO OF LT NETS	1276	783	2289	1080
NO OF INSERTION POINTS	329	128	512	153
LINS (%)	387	611	447	705
GATE OVERHEAD (%)	6.1	3.78	7.9	4.43
RUN TIME (SEC)	1432	685	1768	857

- No of LT Nets : The total number of low transition probability nets for the threshold probability considered for that circuit.
- No of insertion points : The total number of insertion points where dsFF s will be inserted.

- LINS : The ratio of *L*T to *Ins*ertion number which defines ability to maximize transition of more LT nets by using less insertion points

  LINS % = (No of LT nets) * 100 / (no of Insertion Points)
- Gate Overhead : ((Number of gates after adding dsFFs) – (Number of gates before adding dsFFs)) * 100 / (Number of gates before adding dsFFs)
- Run Time : Time taken in seconds by the algorithm to execute

TROJAN ACTIVATION AND DETECTION ANALYSIS :

Table 2. TROJAN ACTIVATION AND DETECTION ANALYSIS BEFORE APPLYING THE PROPOSED METHODOLOGY

(s38417) P_{th} = 0.025	TOTAL CIRCUIT ACTIVITY	NO OF TRANSITION ON LT SET	NO OF TRANSITION ON TROJAN INPUTS	NO OF TRANSITION INSIDE TROJAN	NO OF TRANSITION AT TROJAN OUTPUT	TROJAN ACTIVITY	POC	TCA
Trojan 1	486215	206	4	NA	0	0	0	0.0E+00
Trojan 2	486224	214	16	11	0	11	0	2.26E-05
Trojan 3	486238	230	35	25	0	25	0	5.14E-05
Trojan 4	486265	257	89	51	0	51	0	1.04E-04

Table 3. TROJAN ACTIVATION AND DETECTION ANALYSIS AFTER APPLYING THE PROPOSED METHODOLOGY

(s38417) P_{th} = 0.025	**TOTAL CIRCUIT ACTIVITY**	**NO OF TRANSITION ON LT SET**	**NO OF TRANSITION ON TROJAN INPUTS**	**NO OF TRANSITION INSIDE TROJAN**	**NO OF TRANSITION AT TROJAN OUTPUT**	**TROJAN ACTIVITY**	**POC**	**TCA**
Trojan 1	215846	713	69	NA	10	10	5	4.63E-05
Trojan 2	215886	758	143	72	2	74	0	3.42E-04
Trojan 3	215917	795	184	100	0	100	0	4.63E-04
Trojan 4	216020	893	357	204	0	204	0	9.44E-04

- Total Circuit Activity: Total number of transitions occurring inside the circuit.
- No of transition on LT set : Total Number of transitions in low transition set.
- No of transition at trojan output : The number of Transitions at the Trojan's output represents the number of times that the Trigger output can change from dominant to non-dominant value and vice versa.
- Trojan Activity : Trojan Activity is defined as the sum of the transitions inside and at the output of Trojan circuit.
- POC : Payload Output Change or POC indicate the number of Trojan full activations. POC=0 means none of the trojans is fully activated.
- TCA : Trojan to Circuit Activity or TCA =
  Trojan Activity / Total Circuit Activity

***Table 4.* RESULTS OF THE PROPOSED METHODOLOGY (POC & TCA) USING OPTIMUM THRESHOLD PROBABILITY**

	PAYLOAD OUTPUT CHANGE (POC)				**TROJAN TO CIRCUIT ACTIVITY (TCA)**			
	S18580	**S38417**	**S35932**	**S38584**	**S18580**	**S38417**	**S35932**	**S38584**
Trojan 1	24	5	38	17	0.5E-04	4.63E-05	1.64E-04	2.30E-04
Trojan 2	11	0	16	8	1.5E-04	3.42E-04	2.66E-04	3.92E-04
Trojan 3	3	0	10	0	2.83E-04	4.63E-04	4.74E-04	5.41E-04
Trojan 4	0	0	0	0	4.9E-04	9.44E-04	5.32E-04	7.14E-04

In Table 4 , we have studied the POC and TCA values of the four benchmark circuits using optimum threshold transition probability. POC =0 indicates that none of the trojans is fully activated.

CONCLUSION

We have observed that the number of nets and number of insertion points is minimum for s38417, hence the time complexity taken in traversal along its optimal path is also the minimum.

The proposed methodology is suitable to detect Trojan in hardware circuits. Using the additional hardware (dsFF) we can alter the transition probability of weak links (insertion point) of the circuit and therefore we can resist the malicious attack.

In this way, we can decrease the vulnerability of the integrated circuit chips.

REFERENCES

- Bhunia, Hsiao, Banga, Narasimhan: Hardware Trojan Attacks: Threat Analysis and counter meausures

- Francq,Frick: Introduction to Hardware trojan Detection methods

- Salmani, Tehranipoor,Plussquillic: A Novel technique for improving Hardware Trojan Detection Reducing trojan Detection time

- Zhou, Zhang, Thambipillai, Teo Kian Jin, Chaturvedi, Luo : Cost-efficient Acceleration of Hardware Trojan Detection Through Fan-Out Cone Analysis and Weighted Random Pattern Technique