Hardware Trojan Detection in Soft Error Tolerant Macro Synchronous Micro Asynchronous (MSMA) Pipeline

Faiq Khalid Lodhi [1], *Syed Rafay Hasan* [2], *Osman Hasan* [1] *and Falah Awwad* [3]
[1] Sch. of Elect. Engg. & Comp. Sc., National University of Sciences and Technology (NUST), Islamabad, Pakistan
[2] Department of Electrical and Computer Engineering, Tennessee Technological University, Cookeville, TN, USA
[3] College of Engineering, United Arab Emirates University, Al-Ain, UAE
{faiq.khalid, osman.hasan}@seecs.edu.pk, shasan@tntech.edu, f_awwad@uaeu.ac.ae

Abstract—**Glitches due to soft errors have become a major concern in circuits designed in ultra-deep sub-micron technologies. Most of the soft error mitigation techniques require redundancy and are power hungry. Recently, low power quasi delay insensitive (QDI) null conventional logic based asynchronous circuits have been proposed, but these circuits work for pure asynchronous designs only. This paper extends the low-power soft-error-tolerant asynchronous technique for conventional synchronous circuits. The main idea is to accommodate asynchronous standard cells within the synchronous pipeline, and thus giving rise to a macro synchronous micro asynchronous (MSMA) pipeline. An important application of this design is found in detecting the hardware Trojans. The state-of-the-art signature based hardware Trojan detection is implemented using the clock referencing signals for timing signatures. However, an intruder can intrude into clock distribution network itself and may lead to many false positive or even false negative cases. Asynchronous handshake signals, on the other hand, provide event trigger nature to the digital system, and hence the timing analysis is unique to the data path itself alone, without getting affected by the clock distribution network. This paper provides a proof of concept soft error tolerant MSMA design. Time delay based signature without using clock distribution network is obtained to detect hardware Trojan insertion in MSMA.**

I. Introduction

Because of the shrinkage in modern deep submicron (DSM) technology, the semiconductor devices are becoming increasingly vulnerable to the soft errors (SE) [1]. These errors can lead to malfunctioning in digital systems. Different techniques have been devised to mitigate soft errors, which can be broadly categorized into redundancy based technique (e.g. triple modular redundancy (TMR)) or error detection and correction (EDAC) codes [2]. These conventional techniques are to mitigate soft errors in sequential designs and memory elements only. In modern DSM technologies, combinational circuits are also subject to soft error propagation [2]. Vulnerability against soft errors in combinational circuits and lack of soft error protection for combinational logic poses a major challenge to future high performance mission critical computing [3]. Due to the inherent error detection capability of quasi delay insensitive (QDI) asynchronous circuits, researchers explored them for the prevention and detection of soft errors. But most of the asynchronous techniques resolve SE from the communication interface perspective, hence they cannot be used readily as an alternative to conventional combinational circuits. Recently, a technique to detect and correct the soft errors by using the dual rail property of the threshold gates in Null Conventional Logic (NCL) pipelines, is proposed in [4] and [5]. In this work, we leveraged upon these techniques and developed a soft error tolerant NCL pipeline by using the soft error tolerant threshold gates proposed by Mosaffan et al [6]. Our proposed solution uses asynchronous (NCL based threshold gates) at the micro level to implement combinational logic, and maintain the synchronicity at the macro level. Therefore, we named it as macro synchronous micro asynchronous (MSMA).

An important application of the proposed design is found in detecting the hardware Trojans. With the globalization of integrated-circuit chip design-process, the chances of malicious hardware design intrusion, known as hardware Trojan, have grown tremendously [7, 8, 9]. Hardware Trojans can lead to many mischievous activities, including leaking confidential information, changes in the timing characteristics of the circuits, malfunctioning, denial of service, counterfeiting and the list goes on [7, 10, 11]. Researchers developed various techniques to detect the hardware Trojans. This includes micro-architecture modification to improve triggering of the potential Trojan payload during test in [12] [13]. The main idea proposed in [14] is to scramble the inputs to reduce the possibility of triggering the Trojan payload. A drawback of such techniques is the enormous requirement of observable nodes in a multimillion gate chip. To overcome this issue, timing, and power signature based techniques are proposed [15-21]. These state-of-the-art signature based hardware Trojan detection is implemented using the clock referencing signals for timing signatures. However, an intruder can intrude into clock distribution network itself and the data path timing signature may become obscured. Because, in such detection methods it remains unknown that whether the data path is intruded or the clock network, therefore it leads to many false positive and negative detection scenarios. Such obscurity is especially harmful for the Trojan detection techniques that need to isolate the intruded module. In this paper, we utilize the proposed MSMA design and exploited the asynchronous handshake signals of the digital system to provide timing signature without a requiring clock distribution network. This makes the timing signatures unique to the data path alone.

978-1-4799-4132-2/14/$31.00 ©2014 IEEE

A comprehensive set of simulation results against various test vectors shows that a viable timing signature for a periodical signal, without meddling with the clock signal, can be achieved.

The rest of the paper is organized as follows: Section II provides the detail explanation of proposed technique. In Section III, we describe the hardware Trojan detection in MSMA pipeline. Section IV presents and discusses the experimental results of the proposed technique. Finally, Section V concludes the paper.

II. Soft Error Tolerant Macro Synchronous Micro Asynchronous (MSMA) Pipeline

This section describes our proposed soft error tolerant MSMA technique, which is based on the NCL pipeline [22]. A traditional NCL pipeline architecture is primarily composed of three main blocks: NCL based registers, combinational logic block and completion detection scheme as shown in Figure 1. In this protocol, two stages interact through request and acknowledgement signals. In Fig. 1, the request and acknowledgement are based on Ki and Ko signals. This pipeline works in two different states: DATA state and NULL state. DATA state represents the DATA0 (0, 1) and DATA1 (1, 0) while NULL represents the NULL (0, 0).

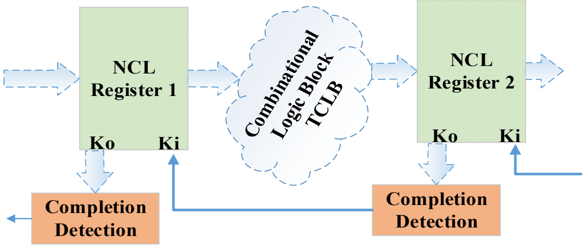

Figure 1. Single Stage Asynchrnous (NCL) Pipeline [4]

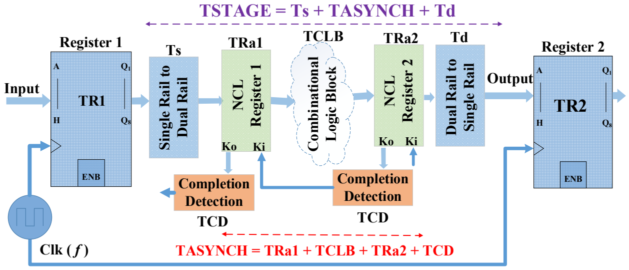

Figure 2. Macro Synchronous Micro Asynchronous Pipeline, where TR1, TR2, TRa1, TRa2, , TCD, Ts, Td represent delay in Synchronous registers 1 & 2, Asynchronous registers 1 & 2, Combinational Logic Block, Completion Detection, Single to dual rail and Dual to single rail block, respectively [22]

Our proposed MSMA is a hybrid form of the synchronous and asynchronous pipelines. In this design, between every synchronous pipeline stage, there is NCL based pipeline to replace the combinational logic block. We utilized the low power soft error tolerant threshold gates proposed in [6] to implement the NCL pipeline, consequently making our MSMA pipeline soft error tolerant. The implementation of the MSMA pipeline, as shown in Fig. 2, can be divided into two main blocks: Asynchronous and Synchronous. The asynchronous block includes the NCL pipeline, which consists of three main blocks: Dual rail Registers, Combinational Block and Completion Detection. Since NCL is based on dual rail encoding therefore all these blocks are implemented using dual rail encoding. The synchronous block consists of synchronous registers at both ends of the MSMA pipeline. The detailed hardware implementations of each block can be found in [22].

III. Hardware Trojan Detection in MSMA Pipeline

The combinational unit of MSMA is an asynchronous circuit, i.e., based on event driven handshake signals. We took the advantage of this inherent characteristic of NCL and measured one complete handshake cycle to obtain a unique timing signature.

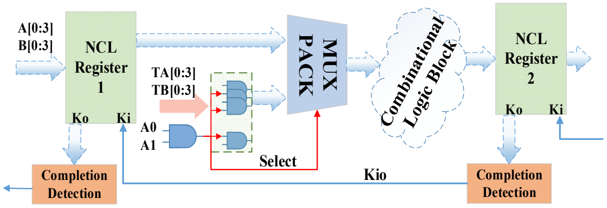

Figure 3. Test vector insertion to obtain timing signature in MSMA

Fig. 3 shows the implementation of test vector insertion to generate timing signature. The test vectors TA and TB are 2 bit dual rail signals. The two LSBs of TA [0:3] are used to select whether the circuit is in test mode or in normal operational mode. If both A0 and A1 are '1' then the design works in the test mode, otherwise it remains in the normal operating condition. Since asynchronous block of MSMA is implemented by using the NULL Conventional Logic, therefore the delay of this pipeline must include the delay of NULL and DATA. For each set of value of test vector, MSMA has a unique DATA delay, leading to distinctive timing signature.

Timing Based Signature Characteristics:

We have used the timing based signature to detect the hardware Trojans. The propagation delays of register1, MUX, combinational block, register 2 and completion detection blocks are assumed as T_{R1}, T_{MUX} ,T_{CLB}, T_{R2} and T_{CD}, respectively, as shown in Fig. 2. Equation (1) provides the latency of the NCL handshake signals (T_{ASYNCH}), which is the duration from the availability of data at register 1 to the assertion of the acknowledgement signal through completion detection unit.

$$T_{ASYNCH} = T_{R1} + T_{MUX} + T_{CLB} + T_{R2} + T_{CD} \qquad (1)$$

NCL pipeline consists of two states of data communication: data and null states. These two states have different latencies, and we call them as data latency (T_{DATA}) and null latency (T_{NULL}), respectively. The total latency of the pipeline should be the sum of T_{DATA} and T_{NULL}. So the total latency of this pipeline can be written as:

$$T_{TOTAL} = T_{DATA} + T_{NULL} \qquad (2)$$

Due to cyclic nature of handshake signals within NCL based MSMA pipeline, the periodic signals can be obtained. For example, one of the handshake signals is periodically repeated during simulation results, shown in Fig. 4, which is further explained later.

It is observed that this total delay (T_{TOTAL}) is unique for each set of the test vector. In the next section we used the Monte Carlo statistical methods to obtain timing signature due to mismatch and process variation.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

This section analyzes the electrical and Monte Carlo simulation results showing the time delay variation in the proposed MSMA using the 130nm CMOS technology. Transistor sizing in all the normal and threshold gates have been optimized using the logical effort characteristics. The feedback inverter is sized on the basis of gate capacitances of the feedback nMOS and pMOS transistors. In the asynchronous block of MSMA, all low power soft error tolerant threshold gates have been implemented using the method given in [6]. Initially, in order to prove the functional correctness of the proposed MSMA, two-bit dual rail adder is designed using these NCL threshold gates. It has been stated earlier that the Ki0 Signal of the MSMA pipeline, shown in Fig. 3 as a feedback from NCL register 2 to NCL register 1, represents the total latency of the pipeline. Fig. 4 represents the delay of Ki0 signal, the point M in the Fig. 4 represents the time at which NCL register 1 sends the request for the DATA and the point O is the instant at which NCL register 1 requests for the NULL. So the time difference between points M and O represents the DATA delay. Similarly, in Fig. 4 the time difference between points O and N represent the NULL delay. As in Equation (2) it is mentioned that the sum of both delays is equal to the total delay of the pipeline stage, which is illustrated as the time difference between point M and N in Fig. 4.

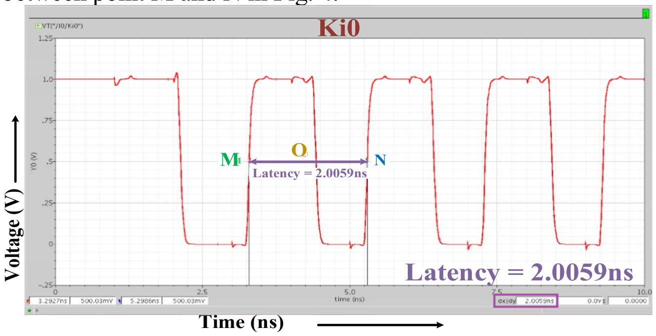

Figure 4. Total Delay of the MSMA pipeline

It has been shown in the contemporary literature that the unique timing based signatures can be obtained by exploiting the fact that process variations for each die lead to distinct delay characteristics [7]. The process variation simulations result in providing a range of possible deviations from the mean timing delay values. To incorporate these effects we also tested our design for the process and mismatch variations. Fig. 5 shows the histogram (following Gaussian distribution) of some of the important process variation parameters that we used in our Monte Carlo based analysis. For nMOS we have changed the vth0 (threshold voltage) from 223mV to 318mV (which is ± 18% from the mean), tox (thickness oxide) from 2.73n to 2.99n (± 5% from the mean) and xj (S/D junction depth) from 2.01×10^{-7} to 2.19×10^{-7} (± 5% from the mean). Similarly, for pMOS we have changed the vth0 (threshold voltage) from 227mV to 329mV (± 19% from the mean), toxp (thickness oxide physical) from 2.74n to 2.98n (5% from the mean) and xj (S/D junction depth) from 2.0×10^{-7} to 2.17×10^{-7}(± 5% from the mean).

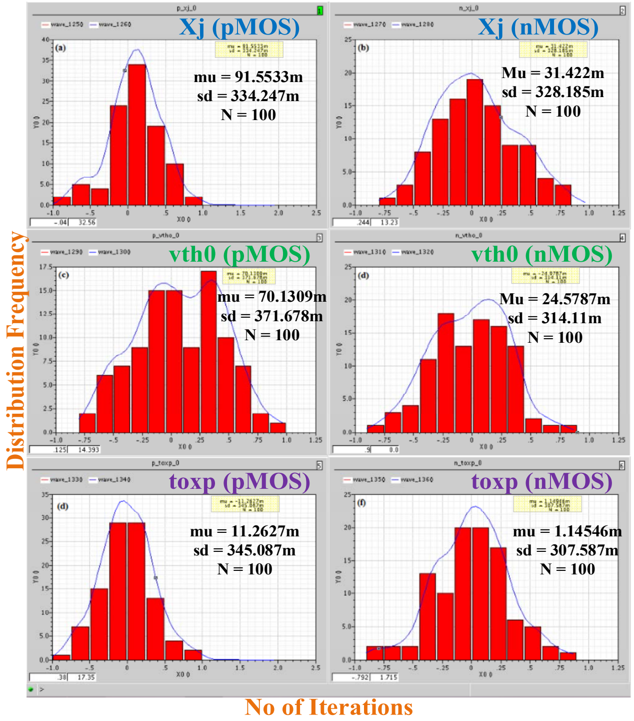

Figure 5. Distribution of Parameters (a) xj of pMOS (b) xj of nMOS (c) vth0 of pMOS (d) vth0 of nMOS (e) toxp of pMOS (f) toxp of nMOS

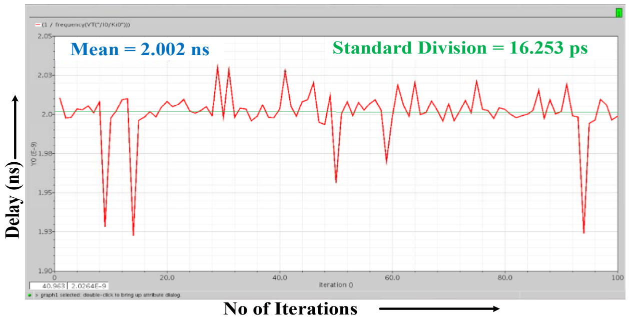

Figure 6. Variation of the delay for Worst Case (when Test Vector is 1011)

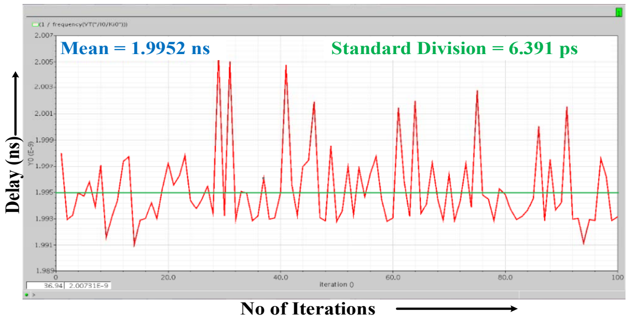

Figure 7. Variation of the delay for Best Case (when Test Vector is 0000)

Table 1 presents the effects of the process and mismatch variations on the total delay of MSMA, which can serve as time based signature of the NCL based MSMA system. In Table 1 it can be observed that the maximum and minimum values of the total delay are 2.0801ns and 1.9089ns respectively. The maximum deviation of the delay is observed when the 1011 test vector is applied at the input and similarly the minimum deviation is observed for test vector 0000. Figs. 6 and 7 shows the delay variation for the worst and best timing delays, respectively.

TABLE I. EFFECT OF PROCESS AND MISMATCH VARIATION ON DELAY

Effect of Process and Mismatch Variation on total delay						
Test Vector		***Delay***				***Deviation***
Single Rail	***Dual Rail***	***Max.*** *(ns)*	***Min*** *(ns)*	***SD.*** *(ps)*	***Mean*** *(ns)*	***Max.*** *(ps)*
0000	***01010101***	***2.0059***	***1.9851***	***6.391***	***1.9962***	***20.8***
0001	*01010110*	*2.0498*	*1.9823*	*8.493*	*2.0019*	*67.5*
0010	*01011001*	*2.0313*	*1.9879*	*6.578*	*2.00182*	*43.4*
0011	*01011010*	*2.0118*	*1.9493*	*7.923*	*2.00591*	*62.5*
0100	*01100101*	*2.0148*	*1.932*	*11.2534*	*2.00758*	*82.8*
0101	*01100110*	*2.0221*	*1.9657*	*7.743*	*2.0042*	*56.4*
0110	*01101001*	*2.0042*	*1.9089*	*13.623*	*2.00789*	*95.3*
0111	*01101010*	*2.0163*	*1.961*	*7.653*	*2.00361*	*55.3*
1000	*10010101*	*2.038*	*1.9525*	*11.502*	*2.0092*	*85.5*
1001	*10010110*	*2.019*	*1.9425*	*9.524*	*2.005*	*76.5*
1010	*10011001*	*2.0166*	*1.9357*	*10.623*	*2.0012*	*80.9*
1011	***10011010***	***2.035***	***1.9225***	***16.253***	***2.0021***	***107.5***
1100	*10100101*	*2.0441*	*1.972*	*9.663*	*2.0042*	*72.1*
1101	*10100110*	*2.053*	*1.963*	*13.592*	*2.0034*	*90*
1110	*10101001*	*2.066*	*1.9753*	*13.894*	*2.0185*	*90.7*
1111	*10101010*	*2.0801*	*1.9821*	*14.253*	*2.0376*	*98*

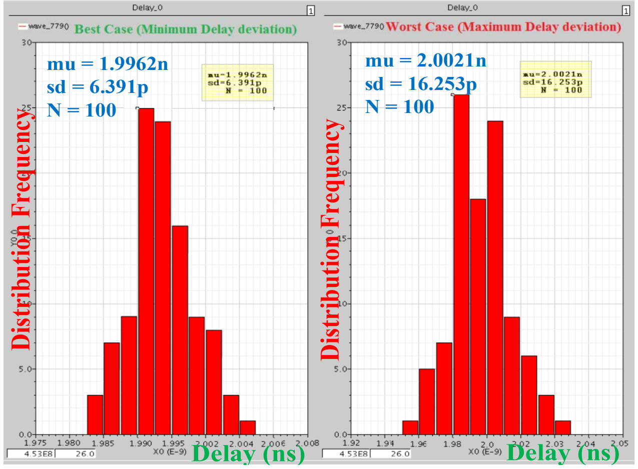

Figure 8. Distribution of Worst and Best Case Delays

Fig. 8 shows comparisons of the worst and the best case delay distribution. It can be observed that this histogram of timing delay signature is following the Gaussian distribution for each test vector combination.

V. CONCLUSION

This paper proposes an NCL based soft error tolerant pipeline that can be introduced in conventional synchronous designs leading to Macro Synchronous Micro Asynchronous (MSMA) designs. We leveraged this technique to obtain unique time delay signatures using the latency of handshake signals in NCL pipeline. We provided a proof of concept test scenario and resulting time delay signature using Monte-Carlo simulations.

REFERENCES

[1] Mukherjee, Shubu. Architecture design for soft errors. Morgan Kaufmann, 2011. ISBN: 978-0-12-369529-1.

[2] P. Shivakumar, M. Kistler, S. W. Keckler, D. Burger and L. Alyisi, "Modeling the Effect of Technology Trends on the Soft Error Rate of Combinational Logic," in the Proceedings of the IDS, 2002 pp. 1-6.

[3] W. Kuang, P. Zhao, J. S. Yuan, and R. F. DeMara, "Design of Asynchronous Circuits for High Soft Error Tolerance in Deep Submicrometer CMOS Circuits," IEEE Trans. On VLSI, vol. 18. 2010, pp. 410-422.

[4] W. Kuang, P. Zhao, J. S. Yuan, and R. F. DeMara, "Design of Asynchronous Circuits for High Soft Error Tolerance in Deep Submicrometer CMOS Circuits," IEEE Trans. On VLSI, vol. 18. 2010, pp. 410-422.

[5] F. K. Lodhi, S. R. Hasan, O. Hasan, and F. Awwad "Modified Null Convention Logic Pipeline to Detect Soft Errors in Both Null and Data Phase " in IEEE International Midwest Symposium on Circuits and Systems (MWCAS 2012), pp 402 - 405.

[6] M. Mosaffan, F. Jafari, S. Mohammadi, "Designing Robust Threshold Gates against Soft Errors," Microelectronics Journal, vol 42, Issue 11, 2011, pp. 1276-1289.

[7] M. Tehranipoor and F. Koushanfar, "A Survey of Hardware Trojan Taxonomy and Detection", IEEE Design and Test of Computer, Volume: 27, Issue: 1, 2010, pp. 10 - 25.

[8] G Di Natale, S Dupuis and B. Rouzeyre, "Is Side-Channel Analysis really reliable for detecting Hardware Trojans?", Conference on Design of Circuits and Integrated Systems (DCIS), 2012 , pp. 238-242.

[9] X. Zhang, K. Xiao, M. Tehranipoor, J. Rajendran, and R. Karri, "A study on the effectivness of trojan detection techniques using a red team blue team approach," VLSI Test Symposium, 2013, pp. 1-3.

[10] F. Koushanfar and A. Mirhoseini, "A unified framework for multimodal submodular integrated circuits Trojan detection," IEEE Trans. On Information Forensics and Security, vol. 6, no. 1, 2011, pp. 162–174.

[11] R. Karri, J. Rajendran, K. Rosenfeld, and M. Tehranipoor, "Trustworthy hardware: Identifying and classifying hardware trojans," IEEE Computer, vol. 43, no. 10, 2010, pp. 39–46.

[12] M. Banga and M. S. Hsiao, "A region based approach for the identification of hardware trojans", Proc. IEEE Int. Workshop Hardware-Oriented Security and Trust (HOST), 2008, pp.40-47.

[13] M. Banga and M. S. Hsiao, "A novel sustained vector technique for the detection of hardware Trojans", International Conference on VLSI Design, 2009, pp. 327-332

[14] A. Waksman and S. Sethumadhavan, "Silencing hardware backdoors," In: Proceedings of the IEEE symposium on security and privacy, Computer Society; 2011, pp. 49–63.

[15] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar,"Trojan detection using IC fingerprinting," IEEE Symposium on Security and Privacy, 2007, pp. 296–310.

[16] X. Wang, H. Salmani, M. Tehranipoor, and J. Plusquellic, "Hardware Trojan detection and isolation using current integration and localized current analysis," Proc. Int. Symp. Fault Defect Tolerance VLSI Syst. (DFT), 2008, pp. 87–95.

[17] J. Li and J. Lach , "At-Speed Delay Characterization for IC Authentication & Trojan Horse Detection", Proc. IEEE Int. Workshop Hardware-Oriented Security and Trust (HOST), 2008, pp. 8 -14.

[18] Y. Jin and Y. Makris "Hardware Trojan Detection Using Path Delay Fingerprint", Proc. IEEE Int. Hardware-Oriented Security and Trust (HOST), 2008, pp.51 -57.

[19] S. Narasimhan, D. Du, R. S. Chakraborty and S. Paul, FG Wolff, CA Papachristou , "Hardware Trojan Detection by Multiple-Parameter Side-Channel Analysis", IEEE Transactions on Computers, Volume 62 Issue 11, 2013, pp. 2183-2195.

[20] K. Xiao, X. Zhang and M. Tehranipoor, "A Clock Sweeping Technique for Detecting Hardware Trojans Impacting Circuits Delay", IEEE Design & Test, Volume: 30 , Issue: 2 , 2013, pp. 26-34.

[21] A. Nejat, S. M. H. Shekarian and M. S. Zamani, "A study on the efficiency of hardware Trojan detection based on path-delay fingerprinting," Microprocessors and Microsystems, Volume: 38, Issue: 3, 2014, pp. 246-252.

[22] F. K. Lodhi, S. R. Hasan, O. Hasan and F. Awwad, "Low Power Soft Error Tolerant Macro Synchronous Micro Asynchronous (MSMA)Pipeline," IEEE Computer Society Annual Symposium on VLSI, 2014, pp. 1-6.