DELTA: DEsigning a steaLthy trigger mechanism for analog hardware Trojans and its detection Analysis

Nishant Gupta, Mohil Sandip Desai, Mark Wijtvliet, Shubham Rai, Akash Kumar
Chair of Processor Design, Technische Universität Dresden, Dresden, Germany
shubham.rai@tu-dresden.de

ABSTRACT

This paper presents a stealthy triggering mechanism that reduces the dependencies of analog hardware Trojans on the frequent toggling of the software-controlled rare nets. The trigger to activate the Trojan is generated by using a glitch generation circuit and a clock signal, which increases the selectivity and feasibility of the trigger signal. The proposed trigger is able to evade the state-of-the-art *run-time detection* (R2D2) and *Built-In Acceleration Structure* (BIAS) schemes. Furthermore, the simulation results show that the proposed trigger circuit incurs a minimal overhead in side-channel footprints in terms of area (29 transistors), delay (less than $1ps$ in the clock cycle), and power (1μW).

ACM Reference Format:
Nishant Gupta, Mohil Sandip Desai, Mark Wijtvliet, Shubham Rai, Akash Kumar. 2022. DELTA: DEsigning a steaLthy trigger mechanism for analog hardware Trojans and its detection Analysis. In *Proceedings of the 59th ACM/IEEE Design Automation Conference (DAC) (DAC '22), July 10–14, 2022, San Francisco, CA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3489517.3530666

1 INTRODUCTION

Defence applications require sophisticated hardware Trojans which are hard to detect and disable [12]. Hardware Trojans consist of two parts: 1) Trigger 2) Payload. The most critical component of the two is the trigger because it determines the degree of stealthiness of the attack while the payload is just the after-effect [4]. Trojans can be broadly classified into two categories, digital and analog. Digital domain Trojans can be inserted at almost all the stages in the IC design flow like RTL [22], gate-level [8], and through malicious CAD tools [15]. In contrast, insertion of analog Trojans in a digital IC is mainly feasible during the back-end design or fabrication since they cannot be defined at RTL or gate-level stages. Fabrication stage attacks vary from inserting additional gates in the layout to modifying circuit parameters like dimensions, and dopant concentration [20].

Several analog Trojan designs have already been proposed in literature. Amongst them, the high-frequency analog Trojan proposed in [20] has been one of the notable and stealthy design. This is because of its small area footprint and highly selective trigger conditions in the analog domain. It leverages the use of *rare-active* nets by controlling them through software-controlled processes. It uses the charge-sharing concept of capacitance to gradually build up a charge and deploy the attack if it reaches a certain threshold. In [3], authors have tried different arrangements for charge-sharing capacitances as an extension to the A2 attack. Another fabrication stage software-controlled analog Trojan has been demonstrated in [1] using an E-fuse-based pre-trigger. The Trojan in [21] uses gate-leakage or reverse current of a diode to charge a large capacitance generated by the Miller effect to increase the trigger time beyond testing duration. [10] deploys charge building on a Trojan capacitor which triggers by repeated write operation of some data pattern to the same address to leak the cache data. Moreover, cross-talk can also be used maliciously to generate trigger signals for charge sharing-based analog Trojans [11]. While most of these Trojans could evade majority of the detection schemes of the digital domain and have low side-channel footprints, analog detection schemes of *R2D2* and *BIAS* have been effective in countering and detecting these Trojans.

This paper proposes a stealthy analog trigger, *DELTA* that has minimal overhead in terms of area, power, and delay and is robust against analog Trojan detection schemes, especially *R2D2* [7] and *BIAS* [4]. This design is the most stealthy form of analog trigger available to date.

This main contributions of this paper are as follows:

- The DELTA trigger that is inserted at the fabrication stage, does not depend upon the selection of rare-active nets. It can be implemented using any available arbitrary net, as long as it does not have a prolonged state of active high signal (which may trigger the Trojan) during its routine functioning.
- It exploits the usage of clock distribution network and glitch generator for its toggling input, which means no need for a high-frequency input signal at the software-controlled net. Hence, it can evade state-of-the-art detection schemes.
- We carry out an overhead analysis in terms of delay, power, and area consumption by our trigger. Further, we determine the range of their working conditions followed by a comprehensive evaluation using analog detection schemes.

2 DELTA TRIGGER COMPONENTS

This section presents the design and implementation details of our trigger. The proposed DELTA trigger uses easily accessible components of any circuit, such as the clock distribution network and an arbitrary net, which helps in its easy implementation. It comprises of two main components: a glitch generator and a charge detector. Using these two components, it enables charge sharing through capacitances to activate the Trojan [20].

2.1 Glitch Generator

A glitch generating hardware circuit [16], as shown in Fig. 1, is used to generate a glitch on every positive edge of the clock which acts as a toggling input for the DELTA trigger. The input to the

DAC '22, July 10–14, 2022, San Francisco, CA, USA
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9142-9/22/07…$15.00
https://doi.org/10.1145/3489517.3530666

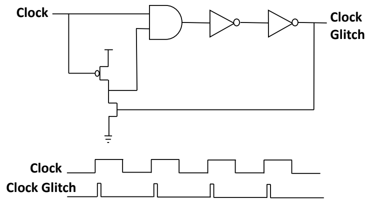
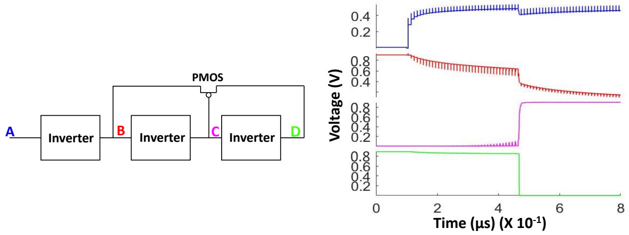


Figure 1: Glitch generator: The clock glitches act as toggling input to build the charge for the trigger circuit.



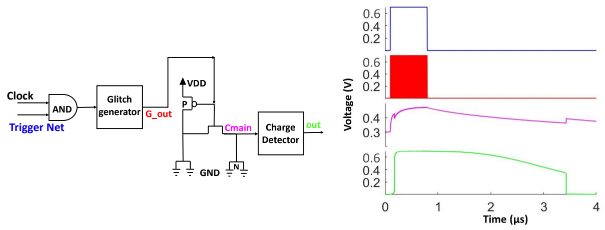
(a) Charge Detector **(b) Transient analysis**

Figure 2: (a) The inverters can be skewed up or down to fine-tune the triggering time and detection threshold. (b) The spikes on node A and B are the result of high frequency clock glitches used as toggling signal to build the charge for the trigger circuit.

glitch generator circuit needs to be conditionally gated, otherwise the trigger input starts toggling as soon as clock is supplied to the circuit. This gating is done by an additional AND gate at the input of the glitch generator. The other input of the AND gate can be controlled by any software-controlled net (it does not need to be a rare-toggling net). This is unlike all the conventional Trojans which use the high frequency toggling of the software-controlled net to activate the Trojan. Instead of frequent toggling, the software controlled, rarely activated, any arbitrary net must be kept at a constant value (logic high, if gating with an AND gate) for a specific duration to activate the trigger. To evade detection during verification, the duration for which a logic high signal is applied should be greater than the maximum duration for which the logic is typically activated by any benchmark or regular program. During this time, clock glitches are generated in order to build the charge on Cmain and assert the charge detector output.

2.2 Charge Detector

The charge detector is the part of the DELTA trigger which detects the charge build-up in the capacitance Cmain and accordingly change the output voltage to initiate the attack of the targeted payload. The authors in [20] proposed charge detection using a Schmitt trigger or a skewed inverter. The skewed inverter does not give a sharp change in the output voltage and has a significant duration of meta-stable output as the voltage gradually builds on Cmain. A Schmitt trigger is commonly used in analog and mixed-signal circuits [17]. Therefore, inserting a Schmitt trigger specifically for the Trojan in a digital IC is a non-trivial task. Moreover, it may not camouflage well in the layout and be observable during optical inspection.



(a) DELTA Trigger - HP **(b) Transient analysis**

Figure 3: The trigger is reset by gate leakage of NMOS N and gives rise to significant retention time.

Therefore, in this paper a novel charge detector is proposed which uses an inverter pair cross-coupled via a pass transistor, as shown in Fig. 2a. Cross-coupled inverters are traditionally used in SRAMs, latches and flip flops, hence they are common in digital circuits. This means that the detector will also camouflage well. Cross-coupling ensures that, unless the input signal is significantly high, the charge detector output would not change. Consequently, it gives a sharp voltage change at the output of the charge detector, as can be seen in the Fig. 2b. Node C or D (as shown in Fig. 2a) can be used to trigger the payload based on active high or low requirement respectively.

3 TRIGGER DESIGN FOR ENABLING ANALOG TROJANS

For demonstration, our design is carried out in Cadence Virtuoso using *Predictive Technology Model* (PTM) at 16 nm technology node. The proposed trigger has two versions: High Performance (HP) and Low Performance (LP). The HP 16 nm library introduces a higher leakage current in the MOSFETs, while the LP 16 nm library has a comparatively lower leakage. The leakage factor leads to change in the design element of the respective triggers.

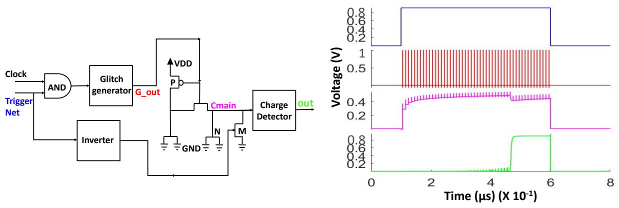
3.1 DELTA Trigger - High Performance

Fig. 3a shows the proposed circuit for a 16 nm HP trigger design. The HP circuit has a larger leakage current due to the low threshold voltage of the technology node. Hence, PMOS P needs to be adjusted to a sufficiently small size so that the pull-up leakage is slightly smaller than the gate leakage through NMOS N. It can be seen from Fig. 3b that the initial voltage on Cmain is not 0 V. Cmain sets to a non-zero value based on pull-up and pull-down leakage currents. It is difficult to balance them as the size of NMOS N also decides the value of Cmain. It can also be seen that, during the reset phase Cmain discharges below a threshold via leakage current, causing the detector output to fall to 0 V. There is a small dip in the Cmain voltage when the detector output goes high as well as a small rise in the Cmain voltage when the detector output goes low.[1]

3.2 DELTA Trigger - Low Performance

Fig. 4a shows the proposed circuit for a 16 nm LP trigger design. In this case a separate charge leakage path is required to drain

[1] This phenomena was observed due to sudden inflow and outflow of current spike at detector input. Inadequate convergence of the device model is suspected for such behavior. However, this particular phenomena has no impact on the functioning of the trigger circuit.



(a) DELTA Trigger - LP

(b) Transient analysis

Figure 4: Clock glitches are generated when the trigger net is high and the detector output resets as soon as the trigger net goes low. Pre-pulses (in the green trace) occur before the detector output goes high as there is a sudden charge transfer to Cmain during every glitch. As the voltage level nears the detection threshold, it tries to pull the detector output high. However, the pull-down leakage and cross-coupling of the detector prevents logic assertion unless the threshold is reached and hence, the detector output goes high after small spikes.

the charge build-up in Cmain because of the comparatively high threshold voltage, which leads to lower gate leakage. If this separate path is not introduced, then if the Trojan is triggered, the output will remain high for a very long duration of time. Additionally, there is a possibility of charge build-up in Cmain (since leakage current would be insufficient to reset the trigger) because of the false trigger.

For the LP trigger circuit, instead of grounding the gate of NMOS M (as done in [20]), it is connected to the trigger net that is used for controlling the clock gating of the glitch generator through an inverter. As a result, when the trigger net is low, the NMOS M gate input is high and Cmain remains discharged. As soon as the trigger net goes high to generate the glitches, NMOS M turns off, enabling the charge build-up on Cmain. As the trigger net goes low again, Cmain is discharged instantaneously through the NMOS M rather than discharging it slowly by leakage current. Consequently, the detector output goes low immediately, as shown in Fig. 4b. This is beneficial to prevent false triggers which might hold the trigger net high for a small duration (smaller than the triggering time) and build unnecessary charge on Cmain. Moreover, by doing so we also eliminate the need for retention time. We can keep the trigger net high until the payload is executed. We balance the pull-up leakage and pull-down leakage currents for the LP trigger circuit so that leakage has no impact on our trigger and it is reset only by de-asserting the trigger signal.

3.3 Triggering time of DELTA triggers

Triggering time is the duration for which the software-controlled net must remain high to activate the trigger. It depends on many design parameters, such as the transistors sizes (which determines leakage - a dominant factor at smaller technology nodes and drive current for the small duration of glitch), the size of Cmain, clock frequency and the detector design. The charge detector plays a significant role in determining the triggering time which can adjust detection threshold by permutations and combinations of skew-up and skew-down inverters (Fig. 2a). The duty-cycle provided by the glitch generator also has some impact on the triggering time. Apart from this, another important factor that places an upper limit on the triggering time is a scheduler or an operating system. The attacker should adjust the above-mentioned parameters and software trigger conditions such that the triggering time is within the processor time allotted to the attacker's program. As seen from Fig. 4b, for a particular configuration of the above-mentioned parameters, the triggering time is less than 400 ns (40 clock cycles) which is well within the limits, yet long enough to avoid accidental trigger activation.

4 DETECTION METHODS

4.1 R2D2 Detection Scheme

This is one of the very famous defense mechanism proposed to protect digital circuits. R2D2 can detect the presence of Trojans several cycles before their activation [7]. The principle behind this method is to guard a set of software-controlled rare-toggling nets because those wires have high chances to incorporate a Trojan. Before deploying the scheme in the given circuit, the *Monitoring Timing Window* (T_m) and the *Attack threshold* (A_{TH}) are required to be tuned carefully in order to eliminate any false alarm and to improve the effectiveness of the method. However, several required built-in features such as an interrupt mechanism cause overhead. There are scalability problems because of the increase in the number of rare-toggling nets in complex circuits. Additionally, difficult-to-tune parameters create an implementation issue for this mechanism [4].

4.2 Built-In Acceleration Structure Detection Scheme (BIAS)

Due to the limitations posed by the R2D2 method, a new technique was developed by [4]. The main goal of this scheme is to make the suspicious net toggle between its rare and typical value at high frequency during detection. To achieve this task, a combination of a Composite-Logic Ring Oscillator (CLRO) and a Multi-Purpose Controller (MPC) is used [4].

BIAS is the only technique available which *might* detect DELTA. Hence, for this work, BIAS-Time-Division Mode-Switching Detection Scheme (TDMS) has been implemented to complete the detection analysis for the proposed DELTA trigger.

4.3 Some Other Techniques

We list some of the other techniques which are used for detecting analog Trojans. We discuss the experimental evaluation in Section 6.

Side-channel Analysis: Trojan detection using side-channel analysis is one of the earliest forms of Trojan detection techniques. The side-channel signatures used in this technique include area, power, delay, and temperature [14], [20]. If there is a signature mismatch then the presence of hardware Trojan is confirmed.
Visual Inspection: Visual inspection [20] uses a scanning electron microscope or any other precise and reliable microscopy that can generate a complete image of the malicious IC and distinguish it from the ideal IC (which is assumed to be reliable).
Adding on-chip sensors: There are three parameters that can be monitored by deploying on-chip sensors in the circuit: signal delay [13], temperature [5], and power spikes [9]. Although all these parameters can now be measured more accurately because of the presence of precise sensors, they all are accompanied by additional hardware overhead [20].

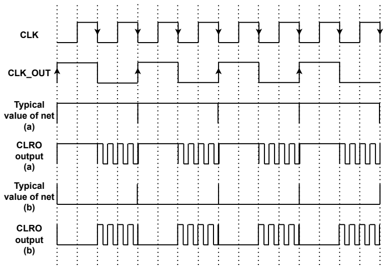


Figure 5: Timing Diagram - BIAS

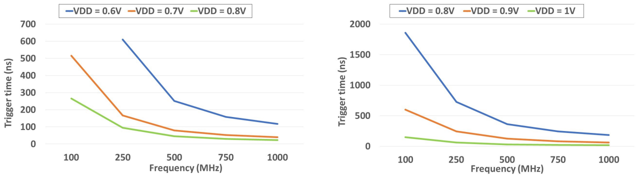
Apart from these, various other technique such as the analytical methods [6], analog IFT [2], and reverse engineering [11] are some other studied methods used for analog hardware Trojan detection. Although they sound promising, they have limited scope, are not yet very well developed, and an attacker can easily find ways to evade them, respectively [18].

5 IMPLEMENTATION OF BIAS - TDMS

In the TDMS detection scheme, the MPC is made to work in the mode where it makes the CLRO to switch between *Conditional Oscillation* (COSC) and *Normal Functioning* (NF) mode at every negative clock edge (opposite to what is stated in [4]). Conditional oscillation refers to the condition when a high-frequency oscillation of 200 MHz, 1000 MHz and 2000 MHz [4] is provided at the suspicious net one by one. On the other hand, when the CLRO is in the normal functioning mode, a routine program that is supposed to run on that functional circuit, is executed. This program will rarely change the value of the suspicious net.

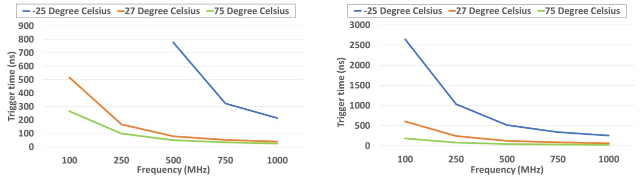
Since the BIAS architecture is deployed on the rare-toggling nodes, it leads to two cases, a) when the typical value of the suspicious net is 1, and b) when the typical value of the suspicious net is 0. As could be seen in Fig. 5, both these cases were implemented for our evaluation. Furthermore, all the primary outputs and intermediate states are then compared with their expected values during the detection process [4] to check the presence of Trojan. A special attention needs to be given to the CLRO outputs shown in the Fig. 5. To showcase the worst-case scenario, we simulated our DELTA trigger without any initial oscillation before the pattern was applied.

As evident from the DELTA trigger design, it takes the clock signal as one of its inputs. Two scenarios are possible– i) when CLK is given as the input, by the attacker ii) when a delayed and out-of-phase version of CLK, CLK_OUT is given as the input, by the attacker. As mentioned in [4], CLK_OUT is the signal provided to the functional circuit, generated by MPC using CLK as an input [4]. The frequency of CLK is taken to be 100 MHz for the implementation of this mechanism so as to have vast range of toggling frequency at our disposal during evaluation. Additionally, the stated value of CLK follows the restriction posed in [4]. Moreover, if the frequency of the CLK signal is increased (by following the mentioned restriction), then the triggering time will decrease.



(a) DELTA Trigger - HP (b) DELTA Trigger - LP

Figure 6: Triggering time VS Frequency with VDD variation



(a) DELTA Trigger - HP (b) DELTA Trigger - LP

Figure 7: Triggering time VS Frequency with Temp. variation

6 RESULTS AND DISCUSSION

6.1 Trigger time analysis of DELTA trigger

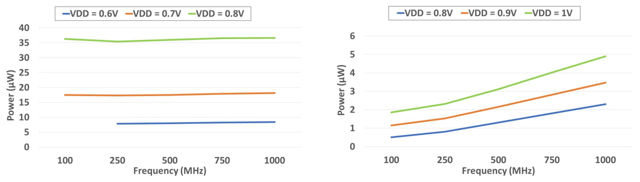
The trigger time of the HP trigger is small compared to that of the LP trigger circuit. This is due to the lower threshold voltage and hence higher leakage (Fig. 6 and 7). From Fig. 6 it can be observed that as the VDD supply increases, the trigger time decreases in both HP and LP DELTA triggers. This is because of an increased drive current through the transistors and an increased leakage current from VDD. Additionally, the trigger time reduces as the toggling frequency increases.

From Fig. 7 it can be concluded that as the temperature increases trigger time decreases in both HP and LP DELTA triggers[2] For the HP trigger, it must be ensured that pull-down leakage is only slightly larger than the pull-up leakage in order to reduce the impact of leakage variation with temperature. Moreover, the impact of leakage current is further overshadowed by the reduction in the detection threshold of the charge detector circuit with the increase in temperature. For LP trigger, the pull-up and pull-down leakage is balanced which eliminates its influence on the trigger time with temperature variation. A decrease in trigger time with an increase in temperature is solely caused by a decrease in threshold voltages, leading to a reduction in the detection threshold of the charge detector.

6.2 Area, Power and Delay Analysis

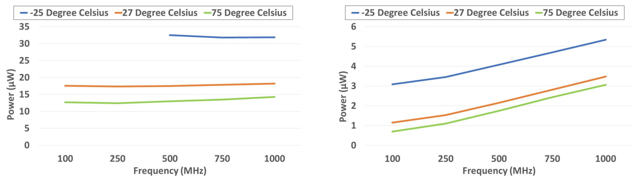
The higher trigger selectivity comes at the cost of extra transistors, thus incurring an area overhead. Table 1 gives the overhead of the DELTA triggers in terms of transistor count in compared to the conventional A2 circuit [20]. The clock-gated glitch generator is the main cause of the area overhead, contributing 18 transistors. However this can still be considered low overhead considering complexity of present-day designs. This low overhead in terms of the

[2] It is to be noted that certain data points are not present in Fig. 6, 7, 8 and 9 as the DELTA trigger circuit is not functional or has a large triggering time for those conditions.



(a) DELTA Trigger - HP **(b) DELTA Trigger - LP**

Figure 8: Average Power VS Frequency with VDD variation



(a) DELTA Trigger - HP **(b) DELTA Trigger - LP**

Figure 9: Average Power VS Frequency with Temp. variation

number of transistors leads to correspondingly small overhead in terms of power and delay as well.

For the HP circuit the average power consumption remains constant throughout the frequency domain (Fig. 8a), while in the LP circuit the average power consumption increases with frequency (Fig. 8b). As VDD increases, average power consumption increases for both the HP and LP DELTA trigger (Fig. 8). Power dissipation in the proposed trigger mainly comprises dynamic and short circuit power dissipation in the glitch generator, short circuit dissipation in the detector and dynamic, and leakage power dissipation in the charge-sharing circuit. For the LP trigger, power dissipation is mainly a function of frequency and VDD as dynamic and short circuit power are the major sources of power dissipation. However, for the HP trigger, short circuit dissipation in the detector and leakage dissipation in the charge-sharing circuit dominate. This results in a almost constant power over frequency variation.

From Fig. 9 it can be inferred that as the temperature increases, average power decreases in both HP and LP DELTA triggers. With an increased temperature, the threshold voltage reduces and leakage current increases which should increase the power dissipation. However, the opposite trend can be observed. As there is short circuit power dissipation in the detector, the triggering time determines the duration for which the short circuit dissipation prevails in the detector circuit. As triggering time decreases with an increase in temperature, the duration of the short circuit dissipation decreases, leading to reduced average power dissipation.

For the clock signal, the percentage increase in delay from circuit without the trigger is only 2%, for all conditions (these conditions are similar to the VDD and temperature variations as shown in the Fig. 6, 7, 8, 9). The maximum delay overhead across all the conditions including HP and LP circuits is less than 1 ps which is well within the limits of PVT variation [20]. In case of software-controlled nets, delay overhead is within 7 ps across all the conditions in HP and LP circuits.

Table 1: Transistor count for the trigger circuits

Conventional A2	*Charge Detector - Inverter*	7
	Charge Detector - Schmitt trigger	11
DELTA Trigger	*HP*	29
	LP	32

6.3 Detection analysis of DELTA trigger

In this subsection we will discuss the effect of different detection methods on DELTA trigger.

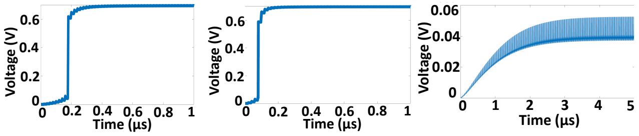
6.3.1 ***R2D2 mechanism.*** The R2D2 detection mechanism is not be able to detect the proposed DELTA trigger because of the following two reasons:

- The DELTA trigger is not limited to the rare-toggling nets, and can even be incorporated with high-toggling nets. Therefore, even if the R2D2 detection mechanism is present in the circuit, it may still not affect the DELTA trigger's detectability.
- The R2D2 mechanism detects the Trojans only if their activation requires a high toggling frequency at the input net. However, the DELTA Trigger needs a constant active high signal to activate, and thus, R2D2 can not detect it.

6.3.2 ***BIAS results when CLK is given as an input to the DELTA trigger.*** Fig. 10 shows the output of the DELTA HP trigger when the respective toggling frequencies are provided at the COSC stage of the BIAS implementation. It can be seen that when the typical value of the suspicious net is 0, the DELTA trigger can be detected in 2 out of 3 cases. It is not detectable in the case when the toggling frequency is 200 MHz, because of the low toggling frequency and prolonged duration of zero triggering activity. This gives time to the charge in Cmain capacitance to leak. In the case where the typical value of the suspicious net is 1, the DELTA HP trigger is detectable in all the three cases.

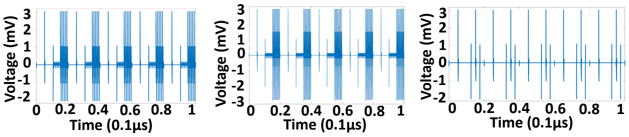
However, the DELTA LP trigger is undetectable in any of the cases, as can be seen from Fig. 11. This figure shows the outputs of the circuit when the typical value of the net is 1, however we obtained the similar results when the typical value is 0. This is because of the addition of the NMOS M and the inverter circuit (Fig. 4a). The CLRO output is applied at the software-controlled rare-toggling net, which is one of the inputs to the DELTA trigger. Therefore, as soon as the trigger net goes down, it activates the extra drain path through NMOS M, which removes all the charge from Cmain capacitance within femtoseconds. Thus making the DELTA LP trigger undetectable.

6.3.3 ***BIAS results when CLK_OUT is given as an input to the DELTA trigger.*** Both LP and HP type DELTA triggers are undetectable when CLK_OUT is given as the clock input to the trigger. When the typical value of the rare-toggling net is 0, no glitches are produced to activate the Trojan because of the opposite logic signal input to the AND gate at the trigger input. Furthermore, when the typical value of the rare-toggling net is 1, irrespective of the toggling frequency of CLRO in the COSC stage, the glitch generator will produce glitches only once in the NF stage in both LP and HP cases, and both the designs are undetectable as can be observed from trigger output in the Fig. 12; however, the explanation for both the cases is different. For the LP design, the activation of drain path through NMOS M (Fig. 4a), whenever an active low signal is



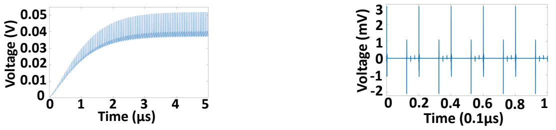
(a) 1000 MHz **(b) 2000 MHz** **(c) 200 MHz**

Figure 10: DELTA trigger (HP) output at different toggling frequencies when the typical value of the triggering net is 0 (CLK is given as one of the inputs to the DELTA trigger).



(a) 1000 MHz **(b) 2000 MHz** **(c) 200 MHz**

Figure 11: DELTA trigger (LP) output at different toggling frequencies when the typical value of the triggering net is 1 (CLK is given as one of the inputs to the DELTA trigger).



(a) DELTA trigger - HP **(b) DELTA trigger - LP**

Figure 12: DELTA trigger output when CLK_OUT is given as one of the inputs and the typical value of triggering net is 1.

provided at the trigger net, causes the leakage of charge build-up in Cmain capacitance. However, in the HP design, prolonged duration of zero triggering activity causes the Cmain capacitance charge to leak before the following glitching event.

6.3.4 ***Other detection techniques.*** As can be inferred from Section 6.2, the DELTA design only uses a few transistors compared to the millions in any processor. Moreover, the designed trigger occupies less than 0.01% of the chip area [19, 20], displays high triggering selectivity and low off-state power dissipation (particularly LP Trigger). Additionally, adding the DELTA trigger in the region of high activity and density can help evade side-channel analysis-based detection schemes, visual inspection technique, and the method of adding on-chip sensors.

6.4 Improvement on DELTA trigger (HP)

To make the DELTA trigger (HP) undetectable, the circuit can be modified by adding the same drain path (using an inverter and an NMOS M; Fig. 4a) as was done in the LP circuit. However, this comes at the expense of a few extra transistors.

7 CONCLUSIONS

In this paper, the DELTA trigger is proposed, which has been implemented on a lower technology node (16 nm). The proposed design no longer depends on a high-frequency triggering of rarely activated nets and incurs minimal side-channel overheads. Additionally, it can evade all the state-of-the-art analog detection schemes. The area and power overhead (at 500 MHz) of the HP circuit is 29 transistors and 17 μW respectively, whereas the LP circuit has the overhead of 32 transistors and 2 μW respectively. Under all the conditions, the percentage increment in the delay overhead is just 2% from the circuit without the trigger. The proposed design has also given leverage to introduce a Trojan at any of the available nets to initiate an attack, making its detection difficult using the available detection schemes.

ACKNOWLEDGMENTS

This work was partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Founda- tion)—Project Number 439891087—SecuReFET and the German Federal Ministry for Education and Research (BMBF) under the framework of VE-CirroStrato.

REFERENCES

[1] Md Mahbub Alam, Adib Nahiyan, Mehdi Sadi, Domenic Forte, and Mark Tehranipoor. 2020. Soft-HaT: Software-Based Silicon Reprogramming for Hardware Trojan Implementation. *ACM TODAES* (2020).
[2] Mohammad-Mahdi Bidmeshki, Angelos Antonopoulos, and Yiorgos Makris. 2017. Information flow tracking in analog/mixed-signal designs through proof-carrying hardware IP. In *DATE, 2017*.
[3] Mohammad Mahdi Bidmeshki, Kiruba Sankaran Subramani, and Yiorgos Makris. 2019. Revisiting Capacitor-Based Trojan Design. In *ICCD*.
[4] Ding Deng, Yaohua Wang, and Yang Guo. 2020. Novel Design Strategy Toward A2 Trojan Detection Based on Built-In Acceleration Structure. *IEEE TCAD* (2020).
[5] Domenic Forte, Chongxi Bao, and Ankur Srivastava. 2013. Temperature tracking: An innovative run-time approach for hardware Trojan detection. In *2013 IEEE/ACM ICCAD*.
[6] Xiaolong Guo, Huifeng Zhu, Yier Jin, and Xuan Zhang. 2019. When Capacitors Attack: Formal Method Driven Design and Detection of Charge-Domain Trojans. In *2019 DATE*.
[7] Yumin Hou, Hu He, Kaveh Shamsi, Yier Jin, Dong Wu, and Huaqiang Wu. 2019. On-Chip Analog Trojan Detection Framework for Microprocessor Trustworthiness. *IEEE TCAD* (2019).
[8] Kai Huang, Yun He, and Xiaowen Jiang. 2019. Holistic hardware Trojan design of trigger and payload at gate level with rare switching signals eliminated. *IEICE Electronics Express* (2019).
[9] S. Kelly, X. Zhang, M. Tehranipoor, et al. 2015. *Detecting Hardware Trojans using On-chip Sensors in an ASIC Design*. J Electron Test 31.
[10] Mohammad Nasim Imtaiz Khan, Asmit De, and Swaroop Ghosh. 2020. Cache-Out: Leaking Cache Memory Using Hardware Trojan. *IEEE TVLSI* (2020).
[11] Christian Kison, Omar Mohamed Awad, Marc Fyrbiak, and Christof Paar. 2019. Security Implications of Intentional Capacitive Crosstalk. *IEEE TIFS* (2019).
[12] He Li, Qiang Liu, and Jiliang Zhang. 2016. A survey of hardware Trojan threat and defense. *Integration* (2016).
[13] Jie Li and John Lach. 2008. At-speed delay characterization for IC authentication and Trojan Horse detection. In *2008 IEEE HOST*.
[14] Yangdi Lyu and Prabhat Mishra. 2021. MaxSense. *ACM TODAES* (2021).
[15] Christian Pilato, Kanad Basu, Francesco Regazzoni, and Ramesh Karri. 2019. Black-Hat High-Level Synthesis: Myth or Reality? *IEEE TVLSI* (2019).
[16] Jan M Rabaey, Anantha P Chandrakasan, and Borivoje Nikolić. 2003. *Digital integrated circuits: a design perspective*. Pearson edu. Upper Saddle River, NJ.
[17] Ashfaqur Rahman, Paul D Shepherd, Shaila A Bhuyan, Shamim Ahmed, Sai K Akula, Landon Caley, H Alan Mantooth, Jia Di, A Matthew Francis, and James A Holmes. 2015. A family of CMOS analog and mixed signal circuits in SiC for high temperature electronics. In *2015 IEEE Aerospace Conference*. IEEE.
[18] Kiruba Subramani, Georgios Volanis, Mohammad-Mahdi Bidmeshki, Angelos Antonopoulos, and Yiorgos Makris. 2019. Trusted and Secure Design of Analog/RF ICs: Recent Developments. In *2019 IEEE 25th IOLTS*.
[19] Shmuel Wimer and Israel Koren. 2013. Design flow for flip-flop grouping in data-driven clock gating. *IEEE TVLSI* (2013).
[20] Kaiyuan Yang, Matthew Hicks, Qing Dong, Todd Austin, and Dennis Sylvester. 2016. A2: Analog Malicious Hardware. In *2016 IEEE SSP*.
[21] Tiancheng Yang, Ankit Mittal, Yunsi Fei, and Aatmesh Shrivastava. 2021. Large Delay Analog Trojans: A Silent Fabrication-Time Attack Exploiting Analog Modalities. *IEEE TVLSI* (2021).
[22] Jie Zhang and Qiang Xu. 2013. On hardware Trojan design and implementation at register-transfer level. In *2013 IEEE HOST*.