Protection Against Physical Attacks Through Self-Destructive Polymorphic Latch

Andrew Cannon[1], Tasnuva Farheen[1], Sourav Roy[1], Shahin Tajik[2], and Domenic Forte[1]
[1]Department of Electrical and Computer Engineering, University of Florida
[2]Department of Electrical and Computer Engineering, Worcester Polytechnic Institute

***Abstract*—On-chip assets, such as cryptographic keys, intermediate cipher computations, obfuscation keys, and hardware security primitive outputs, are usually stored in volatile memories, e.g., registers and SRAMs. Such volatile memories could be read out using active physical attacks, such laser-assisted side-channels. One way to protect assets stored in volatile memories can be the employment of sensors that detect active physical attacks and trigger complete zeroization of sensitive data. However, hundreds or thousands of clock cycles are often needed to accomplish this. Further, the sensing and self-destruction mechanisms are decoupled from the sensitive circuitry and can be disabled separately by an adversary. Moreover, defensive actions (e.g., zeroization) may be disabled by bringing the CPU/SoC into an inoperable condition, while registers may still hold their data, making them susceptible. This paper proposes a self-destructive latch to protect sensitive data from active side-channel attacks, which require supply voltage manipulations.Our proposed latch senses supply voltage interference required during such attacks, and reacts instantaneously by entering a forbidden data state, erasing its stored data. The design uses a NULL convention logic (NCL)-based polymorphic NOR/NAND gate, which changes its functionality with supply voltage. Our results show that the latch is stable across temperature and process variation reacting to attacks with 91% confidence. Even for the 9% where data is not destroyed, in 3.33% of cases data flips its state which makes reliable extraction difficult for an attacker. The polymorphic latch is straightforward to implement due to its NCL implementation and the voltage for the self-destructive behavior is easily altered by resizing only two transistors. Further, this self-destructive behavior extends to registers which are built out of latches.**

***Index Terms*—hardware security, active side-channel attacks, voltage modulation, polymorphic latch, self-destructive countermeasure, polymorphism.**

I. Introduction

In today's digital age, where vast amounts of sensitive information are processed and stored, ensuring the security and protection of data is of paramount importance. One crucial aspect of data security revolves around the volatile memory of computing systems. The security of caches, latches, flip-flops, and/or registers can be compromised by attackers, who gain access to them in a hostile environment and launch physical attacks. Similarly, keys used for protecting hardware intellectual property (IP), such as in logic locking [1], are also susceptible to such attacks [2].

An attacker with physical access to an integrated circuit (IC) can utilize active side-channel attacks such as laser-based probing and fault injection to gain full access to a device's sensitive contents. One common feature of such active attacks is the disturbance of certain physical parameters such as supply voltage, temperature, current, etc. Supply voltage can be lowered to a brownout level at which circuit-based countermeasures are disabled but memory contents in volatile memory elements are preserved. Such successful attacks have been demonstrated to read out the encryption keys stored in SRAM in an FPGA circuit [3], [4], and are equally applicable to ASICs. In addition, supply voltage can be modulated at a controlled frequency and laser-probing can be used to extract static state of gate nodes or memory elements [5], [6]. Supply voltage manipulation such as voltage glitching can be used to inject faults at a strategic timing to divulge secret keys at low-cost and with little technical skills [7], [8]. Such active physical side-channel attacks pose a significant threat and with the growing use of advanced techniques in hardware attacks, new methods are needed to protect sensitive on-chip data.

Several countermeasures have been developed recently to detect the environmental manipulation (e.g., altering supply voltage and/ system clock) involved in these active side-channel attacks. For instance, sensors were proposed to detect on-chip voltage modulation and clock freeze, and then trigger the complete destruction of the IC substrate or zeroization of all sensitive data [9], [10], [11], [12]. However, these sensors are separate from the volatile storage elements under attack and therefore could be disabled or isolated from the response mechanisms. Most notably, zeroization can be disabled by lowering the CPU/SoC supply voltage to the brownout level where on-chip registers still maintain their contents and thus are susceptible to data exfiltration [13], [14]. Sensors and zeroization mechanisms can also be disabled by physically editing circuits using focused ion beam (FIB) systems [15], [16] before probing. Nanopyramid structures have been proposed in the metal layers of an IC to scramble optical signals required by laser-based attacks [17], but this technique requires unconventional fabrication steps. Therefore, a comprehensive, CMOS-compatible, and reliable sense-and-destroy solution remains to be seen.

In this paper, we design a novel self-destructive, polymorphic latch to protect sensitive data from active side-channel attacks that rely on supply voltage manipulation. Our approach requires no extra circuitry or fabrication steps and possesses acceptable power, performance, and area (PPA) overhead. On top of that, it can instantaneously and locally zeroize data upon attacks. In addition to the polymorphic latch, we also propose a supplementary polymorphic buffer/always-off gate, which is used to lock the clock at 0 when supply voltage is lowered below a brownout threshold. Note that since registers are built out of latches, the proposed self-destructive behavior should extend to them as well.

Contributions. Our main contributions in this paper are summarized as follows:

- We propose a self-destructive polymorphic latch to protect sensitive data from physical attacks. This latch obfuscates sensitive bits by entering a "forbidden" data state when an the attack's environmental conditions are fulfilled. In this paper, our latch responds to voltage manipulation, and thus we also elaborate on several attacks that can be prevented under that scenario.
- We apply the state of the art in polymorphic circuit design methodology (NCL-based) to create a supply voltage-controlled NOR/NAND gate. We show that voltage by which the gate's function changes from NOR to NAND is easily tunable through transistor sizing to fit different attack scenarios.
- We design a polymorphic buffer/always-off gate, which freezes the clock in low state during a drop in supply voltage.

- We generate post-layout power, performance, and area of the polymorphic latch and compare to them to a standard NOR-based latch in the same technology node.
- We use simulations to verify the reliability of the polymorphic latch across process variation and temperature. Worst-case performance characteristics are also measured.

The rest of the paper is organized as follows. In Section II, we provide background on attack vectors, existing countermeasures, and polymorphic circuits. In Section III, we describe our threat model. In Section IV, we introduce the polymorphic latch concept and its implementation details. In Section V, we discuss the simulation results and PPA characterization of the latch. In Section VI, we compare existing zeroization approaches with our proposed approach. Finally, in Section VII, we draw conclusions and discuss future research directions.

II. Background

A. Attack Vectors

Active physical attacks such as laser-based probing and fault injection requires physical access to the device under attack. To carry out such attacks attacker has to actively manipulate some parameters such as system clock or supply voltage. In order to show the effectiveness of our approach, we highlight three active physical attacks described below.

1) Thermal Laser Stimulation (TLS): TLS is based on the translation of heat to current in a transistor [3], [4]. First, the laser is directed to a transistor and heats it up, changing the resistance of the active device. If the transistor is carrying some short-circuit current (i.e., is in on-state), then the current will change due to the change in temperature. Otherwise, it will not. A sense amplifier at an output terminal is used to detect variation in current, if it exists. By performing this analysis on each transistor in a volatile memory structure, an attacker can get an idea of the bias states of each individual transistor. This allows an attacker to get a bit-by-bit readout of the memory state, as demonstrated by Lohrke et al [3]. Before TLS can be executed, two conditions must be fulfilled [18]:

- The clock must be frozen to preserve the data in the volatile memory cells. This keeps the sequential elements from changing state.
- TLS is also often accompanied by *lowering of supply voltage* to make the chip enter into a "brownout" state and disable CPU-based defense measures [13]. However, the voltage is kept high enough for the data in volatile memory elements to be retained (typically $\frac{V_{DD}}{2}$).

2) Laser Logic State Imaging (LLSI): LLSI is a fault analysis technique used to detect faults by analyzing the static signals of nodes or registers in a chip [5]. In the hands of an attacker, this technique provides unlimited probes to obtain static signals of any nodes or registers and thus extract secret assets [6]. Near infrared (NIR) with wavelength above the silicon bandgap is transparent to silicon substrate. In LLSI, a laser with such wavelength is used to scan the region of interest of the chip. The incident light is partially absorbed and partially reflected. The intensity of the reflected light depends on the state of the node or register whether it is in 'ON' or 'OFF' state. The reflected light carrying secret information is fed to a detector containing a spectrum analyzer which is set at a known modulation frequency and from the 2D image generated by the spectrum analyzer it is clear to see the data carried by any node, gate, or memory element in the chip divulging secret information.

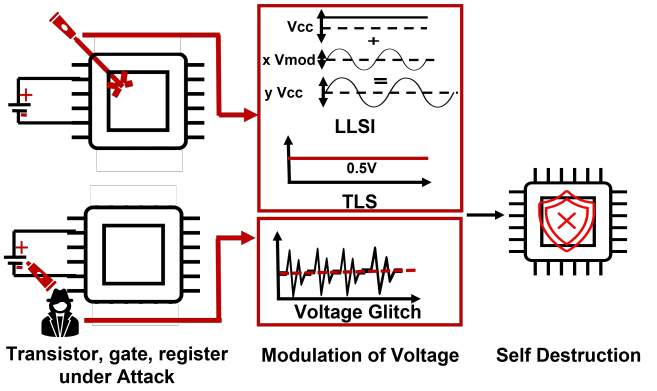


Fig. 1: (left) Laser assisted side-channel (LLSI and TLS) and fault injection attacks against chip via the control of the specific supply voltage signal; (right) Desired self destruction of sensitive on-chip data as a defense.

The conditions that need to be fulfilled to carry out LLSI attack are as follows:

- Like TLS, the clock is frozen at an instance where secret information is available. This keeps the sequential elements from changing state.
- Unlike TLS, *supply voltage has to be modulated at a known frequency* such that the chip is still operational and none of the gates or registers lose their original states.

3) Voltage Fault Injection (VFI) Attack: Optical attacks such as TLS and LLSI requires access to specialized fault analysis machinery and moderate technical skills. On the other hand, noninvasive attacks such as voltage fault injection or voltage glitching are low cost and involve little to no technical skill. The attack complexity may be further reduced by using a software managed voltage fault injection setup [7]. Even trusted execution environments (TEEs) built for security have been compromised using fault injection attacks [8]. A voltage glitch attack setup requires a circuitry for trigger of the voltage glitch which can be very inexpensive. A software based framework can be designed to enable the trigger at appropriate time to inject fault and the result can be analyzed using an appropriately designed software to extract asset by analyzing circuit behavior under fault. Such framework can be reused without any technical knowledge making such attack extremely powerful.

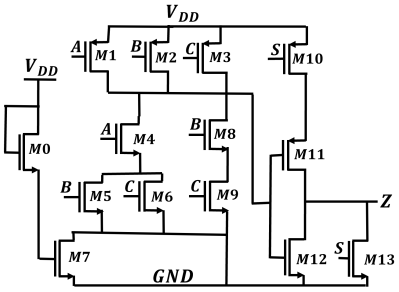
The steps of the VFI attack are as follows:

- Design of a fault injection setup. In case of voltage fault injection it is done either by using typical transistor based setup or by using arbitrary waveform generator. The setup introduces sudden temporary voltage drop from logic '1' to logic '0' which can be termed as voltage glitch.
- Construct a mathematical model and corresponding I/O and timing controller circuitry for triggering voltage faults.
- Extract assets based on fault behavior of the circuit using appropriate algorithm based on the target, power supply, and components in use.

Figure 1 shows the attack vectors of these attacks.

B. Countermeasures Against Physical Attacks

Nanopyramid structures fabricated in the metal layers of a device have been used to interfere with optical elements of laser-based attacks [17]. These structures are able to scatter the incident laser so that it cannot accurately attack an isolated transistor. This results



A	B	Z(NAND)	Z(NOR)
0	0	1	1
0	1	1	0
1	0	1	0
1	1	0	0

Fig. 2: Polymorphic threshold gate, with gating NMOS transistors M0 and M7 [21]. NAND is considered a Boolean subset of NOR, as it outputs a logic 0 under stricter logical conditions (yellow) than NOR.

in lower attack accuracy and less reliable data. This approach is incorporated during metal 1 layer fabrication, but can result in lower device reliability due to higher metal complexity and susceptibiltiy to electromigration [19].

Another work utilizes a network of ring oscillators (ROs) from a physical uncloable function (PUF) and monitors for the change in ring oscillator frequency due laser-based probing attempts [20]. Although this sensor is able to detect physical attacks, it comes with high area and power overhead from the ROs. It also suffers from false positives due to voltage and temperature variations. Another previous work utilized clock freeze and voltage modulation sensors to detect the conditions of an LLSI attack [10], [11]. Under this approach, CMOS-compatible analog circuits were implemented to raise flags. However, this solution is costly in terms of circuit area. Additionally, having sensors separate from volatile memory elements introduces the possibility for an attacker to isolate and disable the zeroization mechanisms, while keeping the volatile memory elements intact and available for exploitation.

The novel approach suggested in this paper improves upon previous methods by *integrating sense and response (zeroization) into the volatile memory element itself, eliminating the possibility for an attacker to disable the countermeasure*. Additionally, this technique is CMOS-compatible, has very little PPA overhead, and does not sacrifice device reliability. This is accomplished by using circuit polymorphism.

C. Design of Polymorphic Gates

Polymorphic gates are logic circuits which change their functionality in response to factors such as voltage, temperature, and light. They were first introduced by NASA's Jet Propulsion Laboratory in 2001 [22]. Since this original design, polymorphic logic has been implemented in CMOS processes to engineer high-performance gates with multiple functions [23]. These original polymorphic gates are often designed by using genetic algorithms to size transistors so that the output behavior of the gate changes with a design variable, but this is very challenging, time consuming, and difficult to port to different technology nodes in practice.

The polymorphic NOR/NAND gate used in our proposed approach utilizes more recent techniques demonstrated in [21]. In this publication, researchers developed supply voltage-controlled polymorphic circuits for use with asynchronous null-convention logic (NCL) circuits. The design process is summarized as follows:

- Select two logic functions, where the low-voltage function is a Boolean subset of the high-voltage function. For example, NAND is considered a Boolean subset of NOR because it has similar, but more strict, conditions to output a logic 0 than NOR. This idea is illustrated by the truth table in Fig. 2.
- The high-voltage function is the less specific Boolean equation. In the case of our later design this is the NOR functionality. The low-voltage function is the more specific Boolean equation, which is the NAND functionality in our design.
- Construct a logic gate with the pull-down network of the high voltage function connected to the pull-up network of the low voltage function. The pull-down network transistors are sized to be five times larger than the pull-up network.
- Add two "gating" NMOS transistors – one in threshold drop configuration;, that is, it drives the gate of the other which gates the connection between the pull-down network and ground. These transistors are sized to select the voltage at which the gate exhibits polymorphic behavior, i.e, changes from NOR to NAND and vice versa.
- Include an output buffer stage, with sleep transistors, for use with null-convention logic.

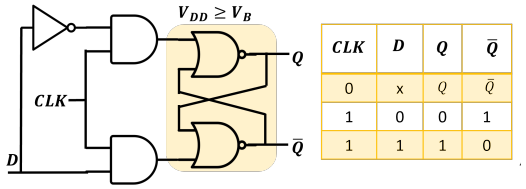
An example polymorphic threshold gate from [21], shown in Figure 2, demonstrates such a polymorphic structure. In the example, the pull-up network is that of a threshold three-of-three gate, whereas the pull-down network is that of a threshold two-of-three gate. The transistors M0 and M7 are the threshold drop and gating NMOS transistors, respectively. At high voltages, the gating NMOS transistors connect the dominant pull-down network to ground, allowing the high-voltage function to dominate. At low voltages, the gating transistors are turned off, and the pull-down network is disconnected from ground. The pull-up network is then able to drive the output functionality.

III. Threat Model

An adversary can get access to devices or chips after they are deployed and probe volatile memory at run-time or inject fault at specific region of interest (ROI) to divulge secrets. The location of the ROI can be obtained in various manners depending on the entity carrying out the attack. In case of active physical attacks, the likely adversary is the end user. An adversary can obtain the placement of cryptographic cores or secret assets on chip either by reverse engineering or inside information from rogue employee or foundry. Depending on the nature of active physical attacks, an adversary needs access to different kind of capabilities. For example, to conduct semi-invasive or invasive laser-based probing attacks, an adversary needs access to sample preparation machine, coherent or incoherent NIR laser, scanning stage, detector system, spectrum analyzer, oscilloscope, etc. In case of a flip-chip device, the chip backside is exposed and readily accessible and thus a sample preparation machine is not needed. For other packaging types, attacker may need to depackage and thin the silicon substrate to a few micrometers from the active layer so that NIR laser can penetrate. To carry out noninvasive active attacks, an adversary also needs access to a setup which can compromise the supply voltage or clock with carefully injected glitches at proper timing to divulge secret information from analyzing the circuit behavior under fault.

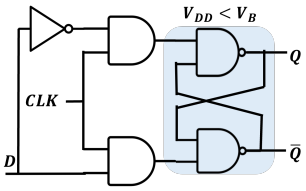
Another necessary aspect of active physical attacks is that an adversary needs to actively control some parameters such as system clock or the supply voltage. An adversary can manipulate the system clock using the external clock pin and manipulate the supply voltage by bypassing the on-chip LDO. These assumptions hold under TLS, LLSI, VFI, and other similar attacks.

When it comes to cost of active physical attacks, invasive or semi-invasive attacks are more accurate but requires expensive machinery. However, such machinery can be rented at hourly rate and such



CLK	D	Q	$\bar{Q}$
0	x	Q	$\bar{Q}$
1	0	0	1
1	1	1	0

(a)



CLK	D	Q	$\bar{Q}$
0	x	1	1
1	0	0	1
1	1	1	0

(b)

Fig. 3: Polymorphic NOR/NAND D-latch and truth tables for (a) normal operation and (b) voltage brownout condition. When V_{DD} is lowered below a brownout voltage, V_B, the polymorphic gates change from NOR behavior (yellow) to NAND (blue) and the latch enters the forbidden state (red).

attacks can be carried out by individual adversary at an affordable cost. Noninvasive physical attacks are much cheaper to carry out and may not require extensive technical knowledge. Some adversaries may perform minor invasive attacks before a semi-invasive or non-invasive attack as well. For example, a FIB can be used to disable on-chip countermeasures, such as sensors or tamper response mechanisms, thereby enabling semi-invasive attacks to proceed without detection/response. Active physical attacks may require minutes to hours to carry out depending on the level of automation employed by the adversary. Considering all these, it is necessary to have a comprehensive and reliable sense-and-response solution that defends against these attacks.

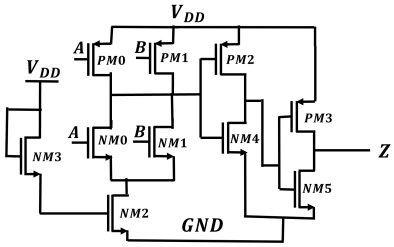
IV. Proposed Polymorphic Latch

In this paper, we design a self-destructive polymorphic latch to protect sensitive data from physical attacks that instantaneously erases data based on the latch's voltage condition.

A. Conceptual Overview

The polymorphic nature of the latch is derived from the behavior of a latch constructed with NOR gates versus that of a latch constructed with NAND gates. A normal NOR-based D-latch, shown in Figure 3(a), operates with the ability to hold data when the clock signal (CLK) is low (0). On the other hand, when the clock is high (1), it will set (reset) Q if $D = 1$ ($D = 0$). However, if the same latch is constructed with NAND gates, the latch enters a forbidden state when CLK is 0. This is illustrated by the red row in the truth table of Figure 3(b). Regardless of D, the outputs Q and $\overline{Q}$ both output a logic 1, due to the NAND gates both having a low input when $CLK = 0$. This state, with Q and $\overline{Q}$ equal, does not represent valid data and effectively destroys any previous data contained in the latch.

By constructing a latch with polymorphic NOR/NAND gates, the latch can function normally for a voltage above a brownout voltage V_b where the polymorphic gate operates as a NOR gate, but enter a forbidden state when the clock is stopped and the supply voltage is below V_b, when the polymorphic gate operates as a NAND gate. Since lower voltage or voltage modulation are prerequisites of the attacks which have been explained in Section III, our proposed latch



A	B	Z(NAND) $V_{DD} < V_B$	Z(NOR) $V_{DD} > V_B$
0	0	1	1
0	1	1	0
1	0	1	0
1	1	0	0

Fig. 4: Polymorphic NOR/NAND gate. The gate functions as NOR for V_{DD} greater than V_B and functions as NAND for V_{DD} less than V_B.

TABLE I: Transistor sizing for NAND/NOR gate.

Transistors	Width (nm)	Length (nm)
NM0, NM1	600	45
NM2, NM3	1800	45
NM4, NM5	120	45
PM0, PM1, PM2, PM3	120	45

can effectively destroy data when the supply voltage drops below the threshold V_b set by a designer. *It is also worth mentioning that as a register consists of two latches in series, it is possible to create a polymorphic self-destructive register by using two of the proposed polymorphic self-destructive latches.*

B. Multi-Threshold Null Convention Logic

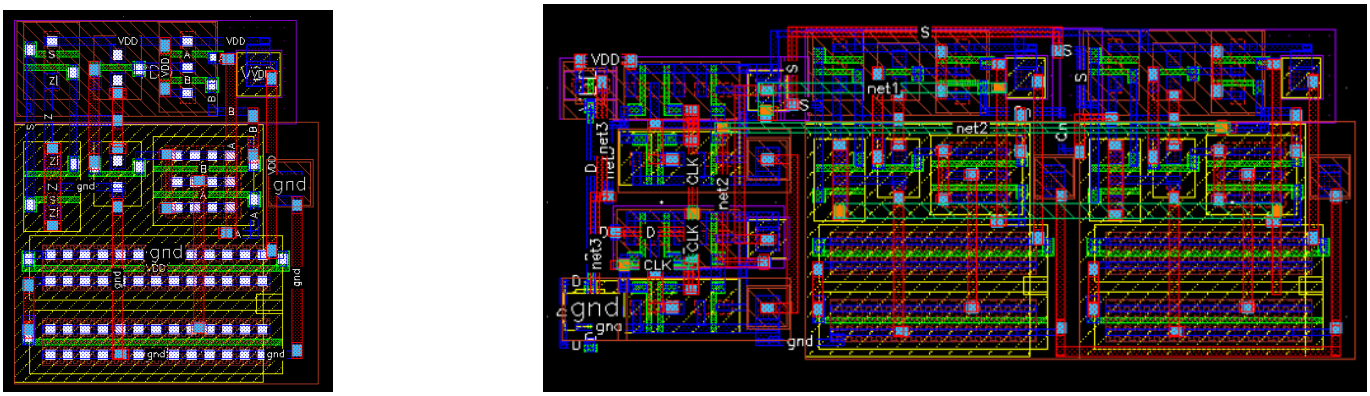
The gates used to construct the polymorphic latch are designed according to Multi-Threshold Null Convention Logic (MTNCL). MTNCL is a design methodology utilized in the implementation of asynchronous logic circuits. Built on traditional Null Convention Logic (NCL), MTNCL uses dual-rail encoding to represent valid data of 0 and 1 [24]. This preserves the principle of quasi-delay-insensitivity provided by NCL, under which a clock is not needed to synchronize data changes in a pipeline. MTNCL expands upon NCL by including sleep transistors with high threshold voltages [24]. These sleep transistors force the logic into a low-power sleep state to reduce leakage current and to separate data states.

The methodology presented in [21] highlights MTNCL's advantages for polymorphic circuit design. Implementing polymorphic circuits as asynchronous circuits allows for low area overhead and afford simpler timing analysis. *Nevertheless, note that the proposed latch will be able to protect data in both synchronous and asynchronous circuits.*

C. Polymorphic NOR/NAND Gate Layout, and Simulation

The design methodology from [21] is used to implement the polymorphic NOR/NAND gates of the latch in a general 45nm PDK. The schematic of the polymorphic NOR/NAND gate is shown in Figure 4. The pull-up network for the polymorphic gate is a transistor-for-transistor copy of the pull-up network of a 2-input NAND gate, consisting of two PMOS devices in parallel. The pull-down network is that of a 2-input NOR gate, consisting of two NMOS devices in parallel. The pull-down network transistors are sized to be five times larger (600nm) than those of the pull-up network (120nm) so that the high-voltage functionality dominates when all transistors are in on-state. Back-to-back inverters form an output buffer.

As discussed earlier, two additional NMOS devices are used to gate the pull-down network and change the device behavior with supply voltage variation. One NMOS ($NM3$) is connected in threshold



(a) (b)

Fig. 5: Layout view of (a) polymorphic NOR/NAND gate and (b) polymorphic latch. Areas are measured as 2.4µm × 2.94µm and 7.0µm × 2.9µm, respectively.

drop configuration, with its gate and drain connected to V_{DD}. The source of this transistor drives the gate of the other gating NMOS ($NM2$) to control the pull-down network's connection to ground. $NM3$ experiences a voltage drop across its channel up to its threshold voltage. Thus, when V_{DD} is near this threshold voltage, the signal driving the gate of NM2 is degraded and will not turn $NM2$ on. This causes the pull-down network to be disconnected and causes the NAND functionality to dominate. However, when V_{DD} is much higher than the threshold voltage, $NM2$ turns on and connects the pull-down network to ground. This causes the NOR functionality to dominate.

The transistor sizes are provided in Table I while the layout view is shown in Figure 5(a). As the base design, we have sized the NMOS devices $NM2$ and $NM3$ to be 1.8µm in width, providing a polymorphic V_b near 800mV. The post-layout design is simulated to verify the NOR/NAND polymorphism. We carry out the simulation in Cadence Virtuoso version IC6.1.7 with 45nm process library with model library set up to `tt` (i.e., typical typical). All the transistors in the design have nominal threshold voltage V_{th}, with the exception of gating NMOS transistors $NM2$ and $NM3$, which are high threshold voltage devices.

The waveform in Figure 6 demonstrates the simulation behavior of the polymorphic gate for both supply voltages of 1.1V and 0.55V. As shown in Figure 6(a), the gate functions as a NOR gate for 1.1V operation. The output, Z, is only high if inputs A and B are both low. Otherwise, Z is low. However, at 0.55V operation in Figure 6(b), it can be seen that the gate operates as a NAND gate. In this mode, the gate output Z is only low if inputs A and B are both low. Otherwise, the gate output Z is high.

D. Polymorphic Latch Design, Layout, and Simulation

As discussed in Section IV-A, the polymorphic latch is built according to Figure 3 in a general 45nm PDK. The polymorphic NOR/NAND gates are used as the two bistable feedback gates in the design. When the supply voltage drops to the brownout voltage V_b, the functionality of these gates changes from NOR to NAND. The change in the gate functionality causes the latch to enter the forbidden state. Thus, when the conditions of the TLS attack are fulfilled, both the Q and $\overline{Q}$ outputs are raised to V_{DD}. This causes complete destruction of any previously stored data and effectively prevents any data from being read out.

The supporting inverter and AND gates of the latch are built using 45nm process with the library default transistor width, 120nm, for all transistors. The layout view of the latch is shown in Figure 5(b). Metal layers 1, 2, and 3 are utilized to provide interconnects between the constituent cells. Finally, the design is simulated to verify the destructive ability of the latch. The waveform in Figure 10 demonstrates this simulation result for both supply voltages of 1.1V and 0.55V. At 1.1V, the latch operates normally. The data is latched from D to Q when the CLK input is high. When the CLK is low, the latch preserves state. At 0.55V, the latch exhibits destructive behavior. When the CLK is low, outputs Q and $\overline{Q}$ both enter logical high state, which does not represent valid data and demonstrates the destruction of the latch's previous data state.

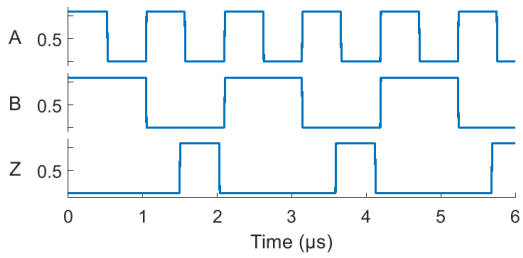
E. Polymorphic Clock Buffer/Always-off Gate

As illustrated in Figure 3(b), the polymorphic latch only enters the forbidden state and clears stored data when *CLK* input is 0. However, according to the attack model, it is possible for an attacker to freeze the system clock in 1 state, in which case the latch would still contain previously stored data. Therefore, we propose a polymorphic gate which can be included in the clock tree to force the system clock to 0 under the attack conditions. This will ensure that any instance of the proposed polymorphic latch will clear its data.

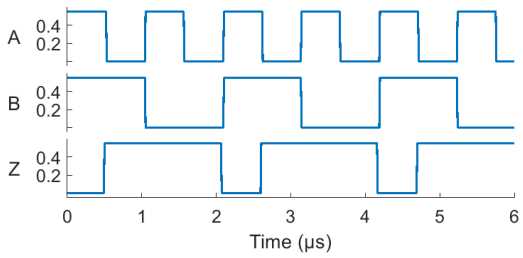
The proposed gate is a polymorphic buffer/always-off gate. When supply voltage is high, the gate acts as a buffer and passes the value of the clock signal unaltered. When supply voltage is low, the gate acts as an always-off gate, outputting a logic 0 regardless of the actual clock value. This gate is implemented as a polymorphic combination of a NOR/XNOR gate followed by an inverter stage.

The gate is designed using the same basic methodology used in designing the polymorphic NAND/NOR gate, discussed in [21]. The high-voltage function is chosen to be NOR, and so the pull-down network is that of a 1-input NOR gate. The pull-up network is that of an XNOR gate. However, to eliminate the need to invert *CLK* to drive the gate of *PM1*, keeper configuration is used. The gate of *PM1* is connected to the output, Z. This ensures that at 0.55V operation, the inverter stage is always driven by at least one path to V_{DD} and thus the output, Z, is always 0. A weak NMOS transistor is also connected between the output and ground to make sure the output pulls low as soon as the supply voltages is dropped.

The schematic representation of the polymorphic clock buffer/always-off gate is shown in Figure 7, and transistor sizings

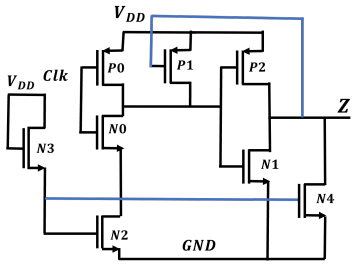


(a)



(b)

Fig. 6: Simulation results of polymorphic NOR/NAND gate behavior: (a), at 1.1V, output signal (Z) is true only for the NOR of both input signals (A, B); (b), at 0.55V, output signal (Z) is true only for the NAND of both input signals.



clk	VDD	
	1.1 V	0.55 V
0	0	0
1	1	0

Fig. 7: Polymorphic buffer/always-off gate. The gate passes *CLK* to Z unaltered at 1.1V, and outputs 0 to Z at 0.55V.

TABLE II: Transistor sizing for clock buffer/always-off gate.

Transistors	Width (nm)	Length (nm)
N0, P2	360	45
N2, N3	1800	45
N1, P0, P1	120	45
N4	180	45

are documented in Table II. The waveform in Figure 8 demonstrates the simulation behavior of the gate for a change in supply voltage from 1.1V to 0.55V. When V_{DD} is 1.1V, the clock buffer passes the value of CLK to Z. When V_{DD} is 0.55V, the gate switches functionality to an always-off gate and the output remains at logic 0.

F. Adjusting Self-destruction Threshold Per Attack

A key feature of the proposed countermeasure is its adaptability to various attack models. It would be desirable for the voltage V_b at which the polymorphic NOR/NAND gate changes state to be controllable by a designer in order to be useful in different applications. For example, in case of TLS attack attacker needs to set the supply voltage to a reduced voltage close to brownout voltage. In case of 45nm technology node with a supply voltage of 1.1V, the brownout voltage is about 550mV. In case of LLSI attack, the attacker can modulate the supply voltage at an amplitude of about 400mV peak to peak, but it can be as high as 700mv peak to peak without disturbing the normal functionality. Thus, the threshold should be set to between 900mV and 1V. In case of voltage glitch attack, voltage may drop to logic 0 temporarily for a span of about 200ns. The voltage drop is substantial and if the response time is faster than 200ns, any threshold voltage for polymorphism is suitable to detect such glitch.

The polymorphism of the NOR/NAND gate can be controlled by modifying the sizing of transistors $NM2$ and $NM3$, the two gating NMOS devices in the pull-down network. The polymorphic threshold

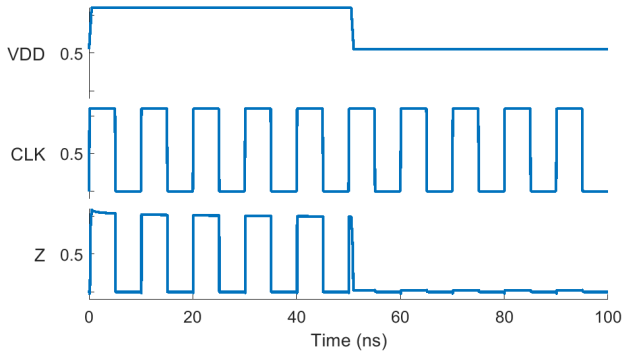


Fig. 8: Simulation results of polymorphic clock buffer. The gate outputs the value of the clock when V_{DD} is 1.1V, but outputs logic 0 when V_{DD} is 0.55V.

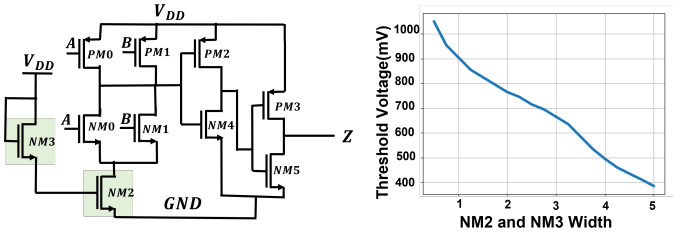


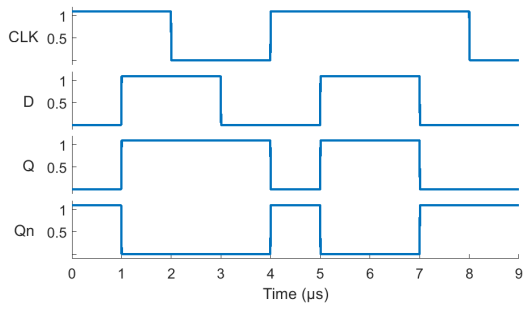
Fig. 9: Polymorphic threshold voltages for various sizes of NM2 and NM3 pull-down transistors.

of the device is inversely proportional to the size of $NM2$ and $NM3$, which should be equally sized. Various $NM2$ and $NM3$ sizes are plotted along with the respective polymorphic threshold voltages in Figure 9. It can be seen that the decrease in polymorphic voltage is approximately linear with increase in transistor size. From Figure 9, it is evident that $NM2$ and $NM3$ width of about 3.5 μm is suitable for TLS attack countermeasure where a width of about 0.5 μm is suitable as LLSI countermeasure.

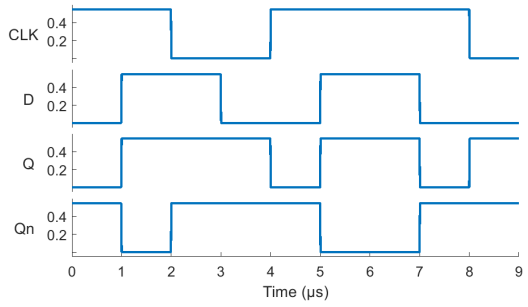
V. Results and Discussion

A. Characterization Procedure

First, we perform parasitic extraction using Cadence Quantus Extraction version 22.1.0-p089. Parasitic extraction is run on all device nets to extract resistances and capacitances, with coupling enabled. Additionally, MOS diffusion resistances are extracted to accurately simulate transistor behavior. After extraction, μs-scale simulation using Cadence ADE is run to separately verify proper



(a)



(b)

Fig. 10: Simulation of polymorphic latch behavior: (a) at 1.1V, D is latched to Q and the inverse to $\overline{Q}$ (Qn) when CLK is high; (b) at 0.55V, Q and $\overline{Q}$ (Qn) both enter logical high state when CLK is 0. This is invalid data and shows the latch entering forbidden state.

digital operation of both the NOR/NAND gate and the latch at both supply voltages of 0.55V and 1.1V.

Next, Cadence Virtuoso ADE simulation environment is again used to measure performance and power of the device. Performance is measured using transient simulation in combination with parametric analysis to characterize device timings. Power is measured using a combination of DC and transient simulation for various device states. This allows all nominal device timings and power measurements to be observed. The area of the latch is also determined from the final layout view.

Cadence Virtuoso ADE XL simulation environment is then used to perform Monte Carlo analysis for device reliability. Process variations and transistor mismatch are included to determine the consistency of the latch's operation at both 1.1V and 0.55V operation. Finally, a parametric temperature sweep is performed in Cadence Virtuoso ADE to characterize the reliability and performance of the latch in the face of temperature fluctuations. This allows worst-case delay of the device to be determined.

B. Power, Performance and Area (PPA) Overhead

Parasitic extraction results allow the latch to be simulated with all net and device RC delays. Power, performance, and area (PPA) results are then obtained using ADE simulation tool in Cadence Virtuoso. These results are compared to that of a standard NOR-based latch as shown in Table III.

CLK2Q Delay: The clock-to-Q and clock-to-$\overline{Q}$ timings are obtained by analyzing the propagation delay from a rising clock edge to valid data being latched to the output.

TABLE III: Power, performance, and area (PPA) comparison.

Parameters	NOR Latch	Polymorphic Latch
Area	7.8 µm^2	20.3 µm^2
$CLK2Q$ delay (rise/fall)	170 ps / 95 ps	320 ps / 178 ps
$CLK2\overline{Q}$ delay (rise/fall)	177 ps / 90 ps	298 ps / 180 ps
$D2Q$ delay (rise/fall)	171 ps / 117 ps	321 ps / 205 ps
$D2\overline{Q}$ delay (rise/fall)	200 ps / 95 ps	326 ps / 186 ps
Setup Time	79 ps	155 ps
Hold Time	4 ps	23 ps
Minimum Static Power	32 pW	82 pW
Average Power	904 nW	25 uW
Peak Power	73 uW	141 uW

D2Q Delay: The D-to-Q and D-to-$\overline{Q}$ timings are similarly obtained by measuring the time for data to be latched to the output when the clock is held high and the D input changes.

Setup Time is measured by determining the earliest that data can arrive relative to a falling clock edge and still be propagated to Q within 5% of nominal D-to-Q delay.

Hold Time is measured by determining how long data that arrives near the minimum setup time must be held after the falling clock edge in order to be properly latched.

Power: Power measurements are collected by running transient simulations up to 10µs in length. Minimum static power refers to the power consumption of the latch when there is no switching activity and the latch does not store any data. Average power refers to the average power consumption when the device experiences switching activity similar to that shown in Figure 10. Peak power consumption is the highest recorded power value observed during switching in simulation.

All PPA measurements are simulated under nominal conditions at 27°C and standard 1.1V operation.

C. Impacts of Temperature

A temperature sweep is performed to analyze the worst-case delay that the latch experiences during data destruction. It is critical that this countermeasure performs quickly to erase data locally before it can be extracted via a laser-based attack.

The objective of the sweep is to measure the time for the latch to enter the forbidden state when CLK is transitioned to 0 at 0.55V. This testing is simulated for temperatures between 0°C and 84°C. The worst case, or longest, delay observed is 4.68ns at 84°C. This worst case delay confirms that an attacker would not be able to fully execute a voltage glitch attack – the attack with the shortest time requirement – before the latch enters the forbidden state. Additionally, the results of the temperature sweep prove that the latch is able to reliably protect data even under changes in device temperature.

D. Effects of Process Variation

It is critical that the latch operates as expected, both in its ability to latch data at 1.1V and its ability to destroy data at 0.55V, across process variation and transistor mismatch. Thus, we analyze the effects of these inaccuracies on device performance. Our results are show in Table IV.

Monte Carlo simulation with 200 simulation points is performed using Cadence Virtuoso to determine the effect of both process variation and transistor mismatch. The simulation is first performed for nominal device conditions (1.1V and 27°C) to verify the ability of the cell to normally latch data. As shown in Table IV, the design could latch and hold data in 100% of simulation points.

TABLE IV: Reliability analysis with Monte Carlo simulation.

Monte Carlo	NOR Latch	Polymorphic Latch
1.1V	100% data retention	100% data retention
0.55V	100% data retention	91% data destruction, 3.33% data flipped

A similar simulation is also performed to verify the destructive ability of the latch across process variation and mismatch. Under 0.55V supply voltage, the CLK signal is dropped to 0 and the output of the latch is monitored for the forbidden state. As shown in Table IV, 91% of test points show the polymorphic latch entering the forbidden data state. Further, in the remaining 9%, 6 out of 18 failing test points showing the data in the latch being flipped. Hence, even though the data was not destroyed for 18 test points, the attacker will still be unable to determine which of the bits flipped and which were not flipped. *We draw the conclusion from this test that the polymorphic latch can either delete or scramble data under an overwhelming majority of device variations.*

The standard 45nm NOR-based latch is also tested under Monte Carlo simulation for its ability to maintain valid data across process variation and transistor mismatch. 200 data points are tested at each supply voltage, resulting in a 100% data retention rate. This demonstrates that a standard NOR latch is able to retain data even when the supply voltage is dropped to 0.55V and is therefore vulnerable to attacks within our threat model. Thus, the polymorphic latch presents a unique and necessary defense that a standard latch cannot provide.

VI. Related Work

Different types of active and passive physical attacks and protection mechanisms have been summarized in [25]. One such protection mechanism is on chip monitoring (OCM) circuit that can detect active physical attack by monitoring on-chip local voltage variation [9]. Another protection mechanism is local electromagnetic attack (LEMA) sensors [26], which are on-chip LC oscillators calibrated against environmental variations such as temperature and device parameter variation such as power supply. These circuits have to be distributed throughout the cryptographic core to ensure security and have a high area overhead. Also, a separate response mechanism is required after the attack is detected which can be identified and disabled by the attacker. Compared to such protection mechanisms, our self-destructive polymorphic latch comes with built-in instantaneous response mechanism that destroys the secret asset within $326ps$. Again, it is not possible for an attacker to remove the latch without disturbing the functionality of the chip.

There are a few existing works that deal directly with self-destruction [12] and memory zeroization [14] techniques when the chip is under active physical attacks. These countermeasures suffer from latency and extra area, power overhead. In [12], a large inductor needs to be designed covering the whole metal area above the cryptographic primitive to be protected. Once the attack is detected it permanently destructs data within a few nanoseconds. But the system does not reboot after attack subsides. In [14] additional circuitry is needed for zeroization of volatile memory segments bit by bit. These countermeasures have separate attack detection and response units and depend on successful communication between the two which can be targeted by the attacker. Compared to such countermeasures, our method of response is far superior as it reacts instantaneously within as quick as $326ps$ with negligible area and power overhead. The comparison is summarized in Table V.

TABLE V: Comparison between previous studies and our work in terms of reaction time, extra circuitry requirements, and power/area (PA) overhead.

Paper	Reaction Time	Extra Circuitry	PA Overhead
Impulse self-destructor [12]	Have latency	Yes	High
Memory zeroization [14]	Thousands of cycles	Yes	High
This work	$326ps$	No	Low

VII. Conclusion and Future Work

We have presented a fully-digital CMOS-compatible polymorphic latch to protect sensitive data from laser-assisted probing and fault injection attacks. The polymorphic nature of the latch means that the device requires no special analog design rules or additional fabrication steps. Compared to existing countermeasures such as nanopyramids or traditional sensor elements, this design comes with significantly lower area overhead and lower design complexity. Additionally, the polymorphic latch acts as both the sensor and the response element, which prevents an attacker from isolating defense circuitry from sensitive storage elements. This device can be customized by resizing of transistors, which allows it to be easily adapted to suit other protection applications. To complement this polymorphic latch, the same design techniques have been applied to develop a polymorphic clock buffer/always-off gate. This ensures that the conditions for the latch to lose its data are always fulfilled when the supply voltage is lost. This technique can also be applied to create a polymorphic self destructive register by using two of the proposed polymorphic self-destructive latches.

Our future work also involves tape-out of the proposed polymorphic gates and self-destructive latches along with verification of these results. We also plan to explore the possibility of polymorphic latch and register implementations in an FPGA fabric.

VIII. Acknowledgements

This effort was sponsored in part by NSF under grant numbers 2117349, 2150122, and 2150123. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government.

References

[1] M. Yasin, J. J. Rajendran, and O. Sinanoglu, *Trustworthy hardware design: Combinational logic locking techniques*. Springer, 2020.

[2] M. T. Rahman, S. Tajik, M. S. Rahman, M. Tehranipoor, and N. Asadizanjani, "The key is left under the mat: On the inappropriate security assumption of logic locking schemes," in *2020 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. IEEE, 2020, pp. 262–272.

[3] H. Lohrke, S. Tajik, T. Krachenfels, C. Boit, and J.-P. Seifert, "Key extraction using thermal laser stimulation: A case study on xilinx ultrascale fpgas," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 573–595, 2018.

[4] T. Krachenfels, T. Kiyan, S. Tajik, and J.-P. Seifert, "Automatic extraction of secrets from the transistor jungle using laser-assisted side-channel attacks." in *USENIX Security Symposium*, 2021, pp. 627–644.

[5] B. Niu, G. M. E. Khoo, Y.-C. S. Chen, F. Chapman, D. Bockelman, and T. Tong, "Laser logic state imaging (llsi)," in *Proceedings from the 40th International Symposium for Testing and Failure Analysis (ISTFA 2014)*, 2014, p. 65.

[6] T. Krachenfels, F. Ganji, A. Moradi, S. Tajik, and J.-P. Seifert, "Real-world snapshots vs. theory: Questioning the t-probing security model," in *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2021, pp. 1955–1971.

[7] C. Bozzato, R. Focardi, and F. Palmarini, "Shaping the glitch: Optimizing voltage fault injection attacks," *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, vol. 2019, pp. 199–224, 2019.

[8] R. Buhren, H. N. Jacob, T. Krachenfels, and J.-P. Seifert, "One glitch to rule them all: Fault injection attacks against amd's secure encrypted virtualization," *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 2021.

[9] M. Nagata, T. Miki, and N. Miura, "On-chip physical attack protection circuits for hardware security," in *2019 IEEE Custom Integrated Circuits Conference (CICC)*. IEEE, 2019, pp. 1–6.

[10] S. Roy, T. Farheen, S. Tajik, and D. Forte, "Self-timed sensors for detecting static optical side channel attacks," in *2022 23rd International Symposium on Quality Electronic Design (ISQED)*. IEEE, 2022, pp. 1–6.

[11] T. Farheen, S. Roy, S. Tajik, and D. Forte, "A twofold clock and voltage-based detection method for laser logic state imaging attack," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 1, pp. 65–78, 2022.

[12] S. Tada, Y. Yamashita, K. Matsuda, M. Nagata, K. Sakiyama, and N. Miura, "Design and concept proof of an inductive impulse self-destructor in sense-and-react countermeasure against physical attacks," *Japanese Journal of Applied Physics*, vol. 60, no. SB, p. SBBL01, 2021.

[13] D. Nedospasov, J.-P. Seifert, C. Helfmeier, and C. Boit, "Invasive puf analysis," in *2013 Workshop on Fault Diagnosis and Tolerance in Cryptography*. IEEE, 2013, pp. 30–38.

[14] A. Srivastava and P. Ghosh, "An efficient memory zeroization technique under side-channel attacks," in *2019 32nd International Conference on VLSI Design and 2019 18th International Conference on Embedded Systems (VLSID)*. IEEE, 2019, pp. 76–81.

[15] C. Helfmeier, D. Nedospasov, C. Tarnovsky, J. S. Krissler, C. Boit, and J.-P. Seifert, "Breaking and entering through the silicon," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, 2013, pp. 733–744.

[16] H. Wang, D. Forte, M. M. Tehranipoor, and Q. Shi, "Probing attacks on integrated circuits: Challenges and research opportunities," *IEEE Design & Test*, vol. 34, no. 5, pp. 63–71, 2017.

[17] H. Shen, N. Asadizanjani, M. Tehranipoor, and D. Forte, "Nanopyramid: An optical scrambler against backside probing attacks," in *ISTFA 2018: Proceedings from the 44th International Symposium for Testing and Failure Analysis*. ASM International, 2018, p. 280.

[18] H. Lohrke, "Laser-based attacks on secure integrated circuits," Ph.D. dissertation, Technische Universität Berlin, 2019.

[19] M. A. Korhonen, P. Bo/Rgesen, K.-N. Tu, and C.-Y. Li, "Stress evolution due to electromigration in confined metal lines," *Journal of Applied Physics*, vol. 73, no. 8, pp. 3790–3799, 1993.

[20] S. Tajik, J. Fietkau, H. Lohrke, J.-P. Seifert, and C. Boit, "Pufmon: Security monitoring of fpgas using physically unclonable functions," *2017 IEEE 23rd International Symposium on On-Line Testing and Robust System Design (IOLTS)*, pp. 186–191, 2017.

[21] C. Bernard, W. Bryant, R. Becker, and J. Di, "Design of asynchronous polymorphic logic gates for hardware security," in *2021 IEEE High Performance Extreme Computing Conference (HPEC)*. IEEE, 2021, pp. 1–5.

[22] A. Stoica, R. Zebulum, and D. Keymeulen, "Polymorphic electronics," in *Evolvable Systems: From Biology to Hardware: 4th International Conference, ICES 2001 Tokyo, Japan, October 3–5, 2001 Proceedings 4*. Springer, 2001, pp. 291–302.

[23] J. Nevoral, R. Ruzicka, and V. Simek, "Cmos gates with second function," in *2018 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*. IEEE, 2018, pp. 82–87.

[24] L. Zhou, R. Parameswaran, F. Parsan, S. Smith, and J. Di, "Multi-threshold null convention logic (mtncl): An ultra-low power asynchronous circuit design methodology," in *Journal of Low Power Electronics and Applications*. MDPI, 2015, pp. 81–100.

[25] M. Nagata, T. Miki, and N. Miura, "Physical attack protection techniques for ic chip level hardware security," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 30, pp. 5–14, 2021.

[26] N. Miura, D. Fujimoto, D. Tanaka, Y.-i. Hayashi, N. Homma, T. Aoki, and M. Nagata, "A local em-analysis attack resistant cryptographic engine with fully-digital oscillator-based tamper-access sensor," in *2014 symposium on VLSI circuits digest of technical papers*. IEEE, 2014, pp. 1–2.