© 2020 IEEE. This is the author's version of the work. It is posted here for personal use. Not for redistribution. The definitive Version of Record is published in IEEE TETC, DOI 10.1109/TETC.2020.3039375

Security Promises and Vulnerabilities in Emerging Reconfigurable Nanotechnology-Based Circuits

Shubham Rai, *Graduate Student Member, IEEE,* Satwik Patnaik, *Member, IEEE,* Ansh Rupani, Johann Knechtel, *Member, IEEE,* Ozgur Sinanoglu, *Senior Member, IEEE,* and Akash Kumar, *Senior Member, IEEE*

Abstract—Reconfigurable field-effect transistors (RFETs) based on emerging nanotechnologies allow switching between p-type and n-type behavior at runtime upon applying different bias potentials. While prior works have focused on particular security schemes using RFETs, here we first revisit the underlying security promises, and further showcase specific circuit vulnerabilities which can lead to adversarial scenarios. More specifically, first, we explore how transistor-level reconfigurability can be leveraged for logic locking and split manufacturing in the pretext of RFET-based modeling of the ITC-99 benchmarks. We find that with only 30% reconfigurable logic gates, we can induce a 100% output error rate (OER) and 31% Hamming distance (HD) on split manufacturing schemes. Second, arguably more disruptive, we explore how the very reconfigurability can be exploited to induce either short-circuit currents or open-circuit configurations, essentially destroying the reliability as well as electrical or functional characteristics of the chip. We apply detailed circuit evaluation and fault modeling toward this end. The novelty and severity of such disruptive scenarios lie in the fact that they can be readily realized in an actual on-field RFET-based chip, either as an adversarial or a fail-safe measure.

Index Terms—Reconfigurable FETs (RFETs), Silicon Nanowire RFETs (SiNW RFETs), Hardware Security

1 Introduction

For many decades, the complementary metal-oxide-semiconductor (CMOS) technology has been the status quo to drive electronic circuits. Further, with the globalization of the supply chain for electronic circuits, giving security guarantees with CMOS-based circuits often comes with huge area and performance overheads [1]. Due to this exponential increase in the cost-to-functionality ratio posed by CMOS technology scaling, researchers have been looking at emerging nanotechnologies, among others, to either substitute or at least complement CMOS technology [2]. A prospective solution has been seen in newer nanotechnologies which exhibit reconfigurability at runtime, thereby offering "more functionality per computational unit" [3].

- *Shubham Rai, Ansh Rupani, and Akash Kumar are with the Chair for Processor Design, Center For Advancing Electronics Dresden, Technische Universität Dresden, 01169 Dresden, Germany (e-mail: shubham.rai@tu-dresden.de; akash.kumar@tu-dresden.de).*
- *Satwik Patnaik was with the Department of Electrical and Computer Engineering, Tandon School of Engineering, New York University, Brooklyn, NY 11201, USA. He is currently with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843 USA (e-mail: satwik.patnaik@tamu.edu).*
- *Johann Knechtel and Ozgur Sinanoglu are with the Division of Engineering, New York University Abu Dhabi, Saadiyat Island, 129188, UAE (e-mail: johann@nyu.edu; ozgursin@nyu.edu).*
- *Corresponding authors: Shubham Rai (shubham.rai@tu-dresden.de) and Satwik Patnaik (satwik.patnaik@tamu.edu).*

The work presented in this article is supported by the German Research Foundation (DFG) funded project SecuReFET (Project Number: 439891087). The work of Satwik Patnaik was supported by the Global Ph.D. Fellowship at New York University/New York University Abu Dhabi. A part of this work was carried out on the High Performance Computing resources at New York University Abu Dhabi.

Reconfigurable field-effect transistors (RFETs) based on materials like carbon [4], graphene [5], or silicon [6], [7] enable reconfigurability by allowing to switch between p-type and n-type functionality on application of different bias potentials. While various works like [3] have shown efficient circuit-level implementations using RFET technologies, their unique features have also been advocated in the field of hardware security [8], [9], [10], [11].

On the one hand, runtime-reconfiguration enables the design of polymorphic logic gates, which, unlike regular CMOS gates, can perform more than one logic function. In the context of hardware security, polymorphic gates are promising for schemes like camouflaging or logic locking, as shown in [8], [12], [13], [14]. RFETs-based standard cells have uniform physical layouts, enabling a security-enforcing designer to obfuscate the circuitry from malicious entities. In turn, this hinders the theft of chip intellectual property (IP) by such adversaries. In this work, we leverage RFET-based circuits in the context of split manufacturing for the first time. Independently, we also leverage RFETs for logic locking. The unique nature of runtime-reconfigurability offers security features for RFET-based circuits at very low area and power overheads [3].

On the other hand, the very same runtime-reconfigurability holds quite severe implications for hardware security, which, to the best of our knowledge, has not been discussed in literature yet. The electrical nature of the logic gates based on CMOS demands a pull-up and a pull-down network for stable outputs. The same concept holds for RFET-based logic gates where even though transistors can be reconfigured at runtime, they still

maintain separate pull-up and pull-down networks [3]. For RFETs, since they can be configured to operate in either p-type or n-type behavior, a "*misconfiguration*" can disrupt the individual pull-up/pull-down network. In other words, RFETs can be misconfigured, even at runtime, representing the root-cause of impeding security vulnerabilities.

1.1 Motivation

Ambipolar conduction, a natural phenomenon observed below 45nm, is the primary reason behind the inherent device-level reconfiguration offered by RFETs. The current drive through RFETs in both p- and n-type configuration is almost identical and, hence, individual or a group of transistors are configured to enable separate pull-up networks (PUNs) and pull-down networks (PDNs), which decides the actual circuit functionality [3]. However, this very same functional polymorphism can lead to intentional or non-intentional misconfigurations of individual transistors disrupting the PUN and PDN, hindering the circuit functionality.

Hence, circuits employing RFETs for their inherent polymorphic behavior are vulnerable to such misconfiguration. This security vulnerability represents application opportunities for both "bane" and "boon" from the perspective of hardware security. On the one hand, an adversary could incorporate a related hardware Trojan to disrupt the underlying functionality. On the other hand, a security-enforcing designer could incorporate a fail-safe measure or "kill-switch," which can destroy the data/chip upon request [15].

The present work evaluates RFETs as a technology with various security measures. We believe that security should not be an afterthought once the technology matures, but it should be evaluated from various present (and probable future) threat scenarios already when the devices are under research and development. In this work, we study for the first time such implications for RFETs, and we leverage circuit-level simulations and fault modeling to do so.

1.2 Our Contributions

The primary contributions of this work are as follows:

- Security Promises
  1) Given their inherent reconfigurability, RFETs are promising devices for large-scale locking, which we study in terms of resilience against Boolean satisfiability-based (SAT-based) attacks.
  2) For the first time, we leverage RFETs in the context of split manufacturing. That is, we propose to route the signals for configuring RFETs above the split layer.
- Security Vulnerabilities
  1) We demonstrate how PUNs and PDNs can be disrupted in RFET-based circuits by misconfiguring one or more transistors in the circuit. By means of circuit simulations of an inverter, we carry out a detailed analysis on *short circuit* and *open circuit* scenarios.
  2) Through circuit-level simulations, we also show how such misconfigurations have significant detrimental impact on current and voltage for both combinational circuit paths and sequential elements like flip-flops, leading to partial or complete derailment of the circuit functionality.

Our study on logic locking shows that the seminal SAT-based attack [16] incurs *time-out* failure for RFET-based circuits when 30–50% of gates are locked. Similarly, analytical experiments conducted for split manufacturing scenarios demonstrate that 100% output error rate (OER) and up to 49% Hamming distance (HD) can be imposed while executing related attacks.

To demonstrate the impact of the new-found security vulnerabilities, for a representative setup considering silicon nanowire RFETs (SiNW RFETs) [17], in case of the short-circuit configuration, we observe a **surge of** 10^5 **times** for the short-circuit current compared to the normal operating current. This can potentially have a significant impact on the overall functioning of the circuit. Such a massive surge of current for an extended period can be exploited for adversarial, detrimental circuit behaviour. Similarly, the voltage impact of an open-circuit configuration leads to the derailment of combinational paths and flip-flops. By simulating the effect of such a misconfiguration using fault modeling, our benchmark-level evaluation over *MCNC*, *EPFL* [18] and *ITC-99* benchmarks reveals 100% OER and average HD of around 36.5%, 40.02%, and 6.3%, respectively, for these benchmark suites. The security concepts explored in this work are applicable to RFETs in general, i.e., they are technology-agnostic.

Organization: The rest of the paper is organized as follows. Section 2 serves to review different emerging reconfigurable nanotechnologies and background about hardware security in general. Section 3 explores various security promises offered by RFET-based circuits. More specifically, we discuss logic locking and split manufacturing in the context of RFET-based circuits. Section 4 demonstrates how the aforementioned security vulnerabilities can be exploited in any RFET-based circuit. We study short-circuit and open-circuit scenarios for RFET-based inverters and then explore such scenarios in combinational and sequential circuits. We also discuss the resulting reliability concerns caused by the abnormal current increase resulting from these configurations. This is followed by Section 5, which provides an analytical investigation for both security promises and vulnerabilities. In Section 6, we conclude our work and discuss how security vulnerabilities can be exploited as future work.

2 Background

2.1 Reconfigurable Nanotechnology Transistors

Various emerging nanotechnologies based on materials like silicon [6], [7], germanium [20], carbon [4], graphene [5], and even 2D materials like WSe_2 [21] bring about the feasibility of designing runtime-reconfigurable transistors. These transistors are ambipolar and can be programmed at runtime to perform the functionality of either a p-type or an n-type transistor. Depending upon their device geometry, they can be broadly classified into two main categories – 1D and 2D devices. 1D devices come predominantly in geometries like nanowires, nanoribbons, etc. Materials like silicon and germanium are used to realize such RFETs. The related transistors have two gate types: the *program gate* (PG) and the *control gate* (CG). The CG is analogous to the gate input of a conventional MOSFET, as it facilitates the

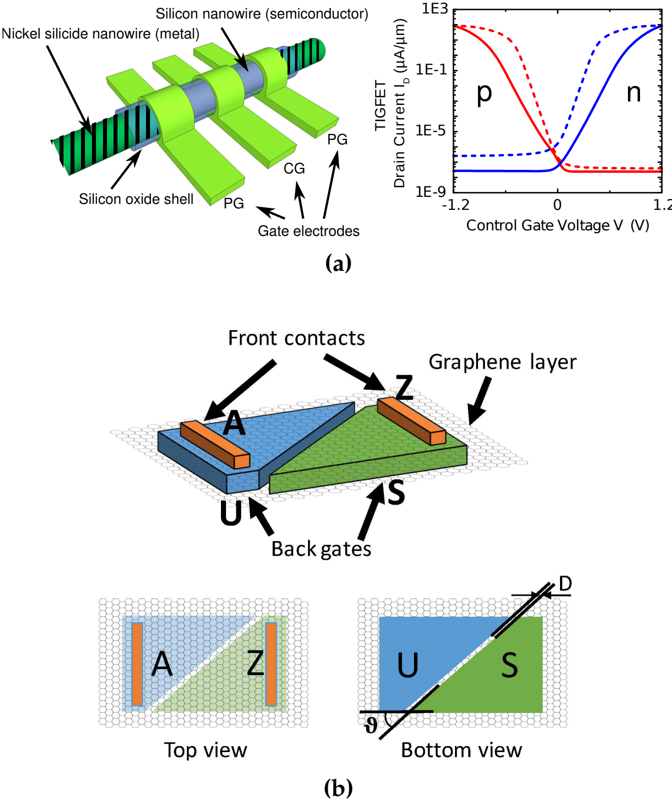

Fig. 1: (a) Silicon Nanowire (SiNW) Reconfigurable field-effect transistor (RFET) [3]. The sub-figure on the left illustrates the device buildup. The sub-figure on the right illustrates the electrical symmetry in both p-type and n-type behavior. (b) Graphene p-n junction RFET [19]. Two metal-to-graphene contacts, A and Z, serve as signal input and output, respectively. A thick oxide layer isolates the two back gates, S and U. Voltage potentials applied at S and U work as "control knob" to impact the device ambipolarity.

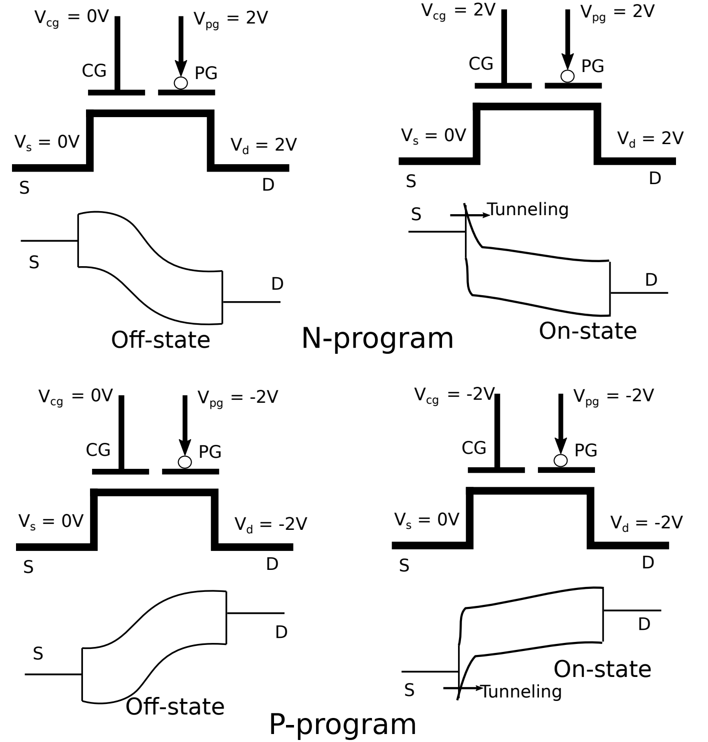

Fig. 2: Conceptual representation of a working principle for a nanowire-based RFET. One can notice how the bands move on application of potential at PG and CG. On-state is $|V_{CG}| = 2V$ and $|V_{CG}| = 0V$ for n and p-type operation, respectively [25].

creation of the channel. The PG controls the type of charge carriers (electrons or holes) in the channel. By applying different bias potentials at the PG, the electrical property of the device switches between p-type and n-type functionality. The current-voltage curve of these transistors is fully symmetrical for p-type and n-type functionality, as shown in Fig. 1a.

Devices made of alternative channel materials, such as graphene [5] or other transition metal dicalchogenide (TMD) materials like $MoTe_2$ [22], WSe_2 [21] belong to a class of 2D reconfigurable nanotechnologies that have been shown to exhibit ambipolarity. This ambipolarity in case of graphene p-n junctions (Fig. 1b) is due to the use of coplanar split gates [19], which are similar to different types of gate terminals (PG and CG) as present in SiNW RFETs. Their extreme thinness offers superior electrostatic control and they are conducive for low-power applications.

Nanowire RFETs can eliminate the employment of multilevel stacked transistors as is the case in the CMOS paradigm, thereby doing away with the individual load offered by the capacitance (e.g., gate-to-source capacitance) from individual transistors cascaded in series [23]. Such functionality has been exploited in works like [24] to design a wired-AND transistor containing multiple independent logical inputs requiring a single supply voltage. Some specific examples of RFET-based logic gates with 3 inputs are shown in Fig. 3.

2.1.1 Working Principle of a Nanowire-based RFET

We take the example of a nanowire-based (Si or Ge based) RFET [6], [20] to explain its working principle. As indicated before, an RFET is a multi-gate structure containing one PG and one or more than one CG. The junction contacts at the source and drain are Schottky contacts [25]. In the off-state, the current is shut-off, due to the barrier induced by the opposing potential at the PG and the CG. In the on-state of the n-type (or the p-type), the control gate enables electron-tunneling (hole-tunneling) through the Schottky junctions by bending down the silicon bands as shown in Fig. 2. For gate-all-around (GAA)-based RFETs as proposed in [7], [23], the potential at the CG and the all-around PGs bend the silicon band in a similar way, which allows tunneling currents based on majority carriers through the Schottky junctions. Further details regarding the physics of such reconfigurable devices can be found in [23], [26].

At the logical abstraction level, an RFET is a programmable device that can be tuned to specific electrical behavior depending upon PG, source, and drain potential. This is shown in TABLE 1. With the default configuration shown in TABLE 1, the device is *ON* in n-FET (p-FET) configuration when CG is at DD (GND) and vice-versa. This runtime-reconfigurability, in turn, leads to functional flexibility at the logic-gate level, where a single logic gate exhibits more than one functionality [3].

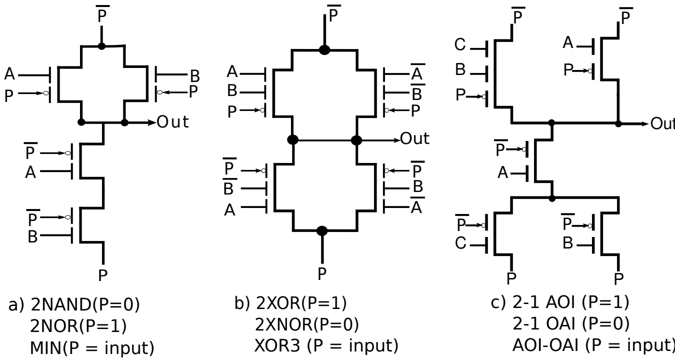

Fig. 3: Reconfigurable logic gates from SiNW RFETs [3].

TABLE 1: Programming RFETs [25]

Functionality	Potential at PG V_{pg}	Potential at Source V_S	Potential at Drain V_D
n-FET	DD	Low	High
p-FET	GND	High	Low

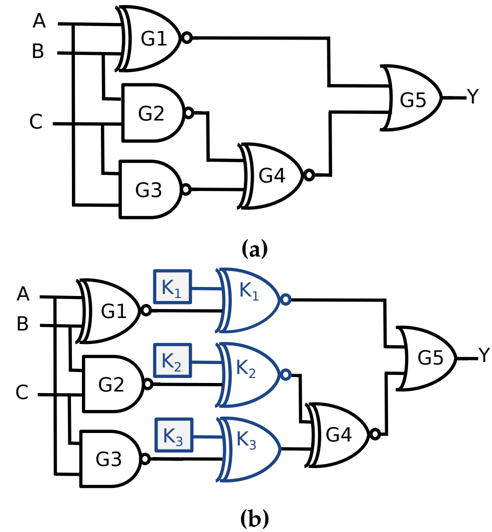

Fig. 4: Illustration of logic locking. (a) Original circuit. (b) Circuit locked with three XOR/XNOR key-gates labelled as K_1, K_2, and K_3, respectively for traditional CMOS technology. The key-bits K_1, K_2, and K_3 are driven from an on-chip *tamper-proof* memory.

2.1.2 Feasibility Aspects for RFETs

Emerging nanotechnologies often come with their share of apprehension in terms of commercial adoption. We list down following points which are relevant considering wide-scale commercial adoption of RFETs:

1) Reconfigurability is a logical abstraction of electrical symmetry or ambipolarity, a common phenomenon observed below 45 nm. There are various materials like silicon, germanium from which RFETs can be realized. These materials are also used in CMOS fabrication. Hence RFETs made of these materials can readily be adopted [27].
2) The stacked nanowire or nanosheet geometry is a successor to *FinFET* geometry that is promoted to be used at lower technology nodes [28]. Hence, from a geometry point of view as well, RFETs follow similar CMOS integration process.
3) A commercial benefit that can be advocated for silicon- or germanium-based nanowire RFETs is that their fabrication process is entirely compatible with CMOS technology [26], [29]. Additionally, these nanowire-based RFETs are dopant-free technologies and hence do not require high-temperature process steps.

2.2 Hardware Security

In this section, we review the security promises of logic locking and split manufacturing along with their related threat models.

2.2.1 Logic locking

Logic locking has emerged as a viable and promising solution for protecting the design IP throughout the IC supply chain [30]. The IP is protected by the insertion of dedicated locks that are operated by a secret key. Therefore, a locked circuit has additional inputs, which are referred to as key-inputs; these key-inputs are driven by an on-chip tamper-proof memory. The additional logic gates inserted to realize these locks are known as key-gates. Traditionally, these locks have been realized by adding XOR/XNOR gates, AND/OR gates, or look-up tables (LUTs). A logic locked IC functions correctly only when the correct key is applied, in the event of a wrong key being fed to the circuit, the IC becomes non-functional. After implementing a given locking scheme, the design house sends the chip to a foundry for fabrication, potentially *untrustworthy*. Once the chip has been fabricated and tested (but before deployment), the locked IC is activated by loading the secret key onto the chip's dedicated, tamper-proof memory by some *trustworthy* entity. For in-depth coverage of logic locking and associated attacks, interested readers are referred to [30].

Figure 4a shows an original (unprotected) circuit and Fig. 4b shows its locked version in traditional CMOS through three XOR/XNOR key-gates. One of the inputs of each key-gate is driven by a wire from the original design, while the other input, referred to as key-input is driven by a key-bit stored in a *tamper-proof* memory.

In general, a threat model quantifies the attackers' capabilities and available resources for launching attacks. The threat model for logic locking enumerated next bears consonance with the academic community's assumptions.

1) The design house, designers, and computer-aided design (CAD) tools are considered *trustworthy*, whereas the foundry, the test facility, and the end-user(s) are all considered *untrustworthy*.
2) The attackers know the locking scheme implemented by the design house.
3) The attackers have access to the locked gate-level netlist (e.g., by reverse engineering) and can identify the key inputs, key-gates but are oblivious to the secret key.
4) The secret key which is stored in a *tamper-proof* memory cannot be tampered with.
5) The attackers possess a functional chip that can be bought from the open market. Only "black-box" usage

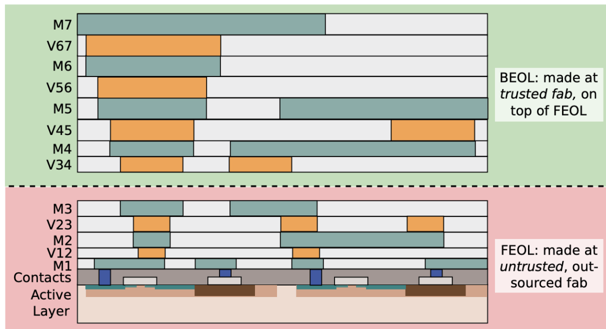

Fig. 5: Concept of classical split manufacturing, i.e., the separation of a physical layout into Front-end-of-line (FEOL) and Back-end-of-line (BEOL). The different pitches across the metal layers facilitates split manufacturing. © 2018 IEEE. Reprinted, with permission, from [31].

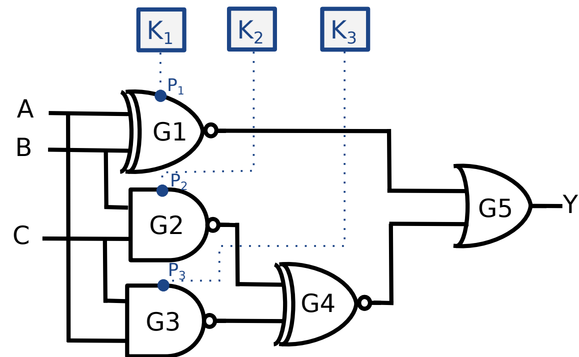

Fig. 6: Logic locking using RFETs, where the program gate (PG) acts as the key-input. Realizing logic locking in conventional CMOS necessitates insertion of additional logic gates whereas the inherent construction of RFETs facilitates non-insertion of additional logic gates. The PG signals are driven from an on-chip *tamper-proof* memory.

of the chip is permitted, i.e., an attacker can only evaluate input/output patterns.

2.2.2 Split Manufacturing

Split manufacturing helps in the protection of the design IP from untrustworthy foundries during manufacturing time [31], [32], [33]. The split manufacturing premise dictates splitting up the IC manufacturing flow, typically into the front-end-of-line (FEOL) and back-end-of-line (BEOL). An attacker in the FEOL foundry views the layout as a “sea of unconnected gates” where some of the connections are complete and some of the connections are missing/incomplete. The notion of splitting the IC manufacturing flow into FEOL and BEOL is practical for multiple reasons: (i) outsourcing the FEOL is desired, since it necessitates advanced, high-end and costly fabs, (ii) BEOL fabrication on top of the incomplete FEOL layout is significantly less complicated than FEOL fabrication, (iii) the sole difference for the supply chain is the preparation and shipping of FEOL wafers to the BEOL facility. Figure 5 illustrates the idea of classical split manufacturing where the FEOL is outsourced to an advanced, off-shore, *untrustworthy* foundry while the BEOL is fabricated at a *trustworthy* foundry. For in-depth coverage of split manufacturing and associated attacks, interested readers are referred to [34].

The most common threat model adopted for split manufacturing is summarized as follows:

1) The design house, designers, CAD tools, and end-user are *trustworthy*, while the FEOL foundry is considered *untrustworthy*. Split manufacturing dictates the existence of a BEOL foundry, with assembly and testing facilities, also considered as *trustworthy*.
2) The attackers cannot obtain a functional chip from the open market, as the end-user is trusted. This scenario exists for military applications where the chips are deployed only for specific sensitive and mission-critical applications and are not available in open markets. Also, the chip has not been fabricated before and is unavailable for reverse engineering-based attacks.
3) The objective for an attacker (located in the FEOL foundry) is to infer the missing BEOL connections from the incomplete FEOL layout. Towards this end, the attacker is aware of the underlying protection schemes (if any) and has access to the EDA tools, libraries, and other information available to a trustworthy designer/design house.

2.3 RFET-Based Hardware Security

Runtime reconfiguration in RFETs enables the design of polymorphic logic gates, which, unlike regular CMOS gates, can perform more than one fixed logic function. Such polymorphic gates are promising for security schemes like camouflaging, logic locking, physically-unclonable functions (PUFs), etc. The authors in [8], [12] laid the foundation for hardware security using RFET-based circuits. Due to their inherent virtues of uniform physical layouts and post-manufacturing reconfigurability, such logic gates enable a security-enforcing designer to carefully obfuscate the circuitry from malicious foundries, test facilities, and/or end-users. In turn, this hinders the theft of chip intellectual property (IP) by such adversaries. They also showed how a circuit layout composed with such RFET-based logic gates is difficult to reverse-engineer. They demonstrated how a single tile layout could implement either a *NAND* or *XOR* using different pin configurations. However, these studies lacked both circuit-level simulations and thorough security evaluations against state-of-the-art attacks, e.g., the seminal SAT-based attack [16]. Other hardware security schemes, such as watermarking using RFETs-based circuits, have also been proposed in [11]. Similarly, RFET-based logic gates are less prone to delay-side-channel attacks as their CMOS counterpart [35], [36].

3 Security Promises

In this section, we discuss the security promises offered by RFETs due to their transistor-level reconfigurability. Detailed benchmark-level evaluations are presented in Sec. 5.1.

3.1 RFETs for Logic Locking (Transistor-level Locking)

Figure 6 illustrates logic locking using RFETs. It should be noted that realizing logic locking in conventional CMOS necessitates the insertion of additional logic gates (e.g.,

XOR/XNOR, look-up tables (LUTs), etc.). In contrast, the inherent construction of RFETs does not require the insertion of additional gates to realize locking. This is because RFETs come with program gates (PG), which act as an in-built key, and hence this notion of leveraging RFETs for logic locking can also be viewed as *transistor-level locking*. In general, RFET-based logic locking offers lower area overheads than traditional logic-locking schemes since it does not require additional logic gates. Furthermore, since no additional logic gates are inserted, the original circuit's critical path continues to have the same number of stages as in the original circuit, thereby having no performance penalty.

With regards to area estimation, let us take an example of a circuit to be logic locked with k-bits. In the case of CMOS-based circuits, we need to add k logic gates (XOR/XNOR) for logic locking [30] (e.g., RLL/FLL). For CMOS circuits, each XOR consists of 10 transistors. Hence, for k-bit locking, the circuit will require $10 \times A_{CMOS} \times k$ number of more transistors, where A_{CMOS} is the size of a CMOS transistor. For RFET-based circuits, since they allow inherent reconfiguration, for k-bit logic locking, it will require k inverters to have P (and P') as the extra input for each reconfigurable logic gate [3]. Each inverter consists of 2 transistors. Hence, the area overhead for the same locking scheme is $2 \times A_{RFET} \times k$, where A_{RFET} is the size of an individual RFET transistor. Between the technologies, the overall area overhead is calculated as:

$$Overhead = (n_{CMOS} \times A_{CMOS} - n_{RFET} \times A_{RFET}) + (10 \times A_{CMOS} \times k - 2 \times A_{RFET} \times k) \quad (1)$$

Here, n_{CMOS} and n_{RFET} are the number of transistors in CMOS and RFET-based circuits, respectively. The first term in Eq. 1 is the difference in the area of the same circuit in two technologies while the second term is the actual overhead for ensuring k-bit locking. One can notice, that for k-bit locking, the RFET overhead is less than that of CMOS by a factor of 5. Hence, the overall area overhead depends upon two main factors— the number of transistors in a circuit and the size of an individual transistor. Due to higher functional expression, the number of transistors in case of RFET-based circuits is much less as compared to CMOS-based circuits [3], [23], i.e., $n_{RFET} < n_{CMOS}$. Since RFETs models are still evolving, with better RFET models, A_{RFET} will get closer to A_{CMOS}. For instance, for an early evaluation model of RFET, it was demonstrated in [37] that for the same circuit, the actual area for RFET-based circuits is just 17% more than that of CMOS. Bringing all these factors in consideration and when using the same technology node, an RFET-based circuit has a lower area as compared to a CMOS circuit for the same k-bit logic locking.

3.2 RFETs for Split Manufacturing

The applicability of split manufacturing in the context of RFETs is shown in Fig. 7. Here, only the PG signals must be wire lifted beyond the split layer. In contrast, all the other, regular signal nets can be routed freely following the designer's specifications and the CAD tool heuristics.

In this work, we leverage the concept of *functional polymorphism* enabled by RFETs for enhancing the security of split manufacturing. As indicated, the application of various control signals to the PG of RFETs results in different functionalities, as shown in Fig. 3. However, without knowing the control signal, one cannot readily infer the logic gate's actual functionality. Moreover, we utilize the concept of logic locking towards securing split manufacturing.

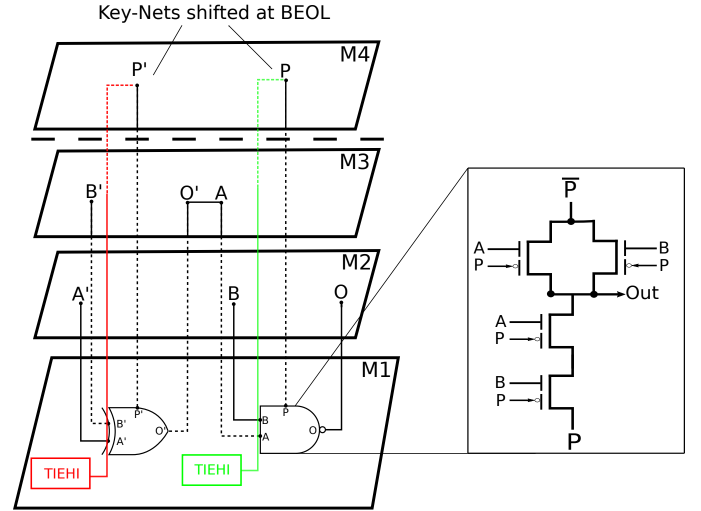

Fig. 7: Split manufacturing for RFETs; here, only the program gate (PG) signals are lifted beyond the split layer (in this example, M3) to BEOL (M4). The inset shows the internal transistor-level schematic of the underlying logic gate. These lifted PG signals will be driven by constant 0/1 signals (generated by TIELO/TIEHI cells placed in the FEOL), routed in the BEOL. The absence of placement- and routing-related hints for the key-nets makes the key indecipherable for an FEOL-centric attacker.

Both concepts taken together, polymorphism and locking, allow us to assign individual key-bits directly for any logic gate of choice, but we also require to lift the related wiring associated with the program gates above the split layer. This way, we obfuscate the actual functionality of these gates from the foundry-based adversaries. The essence of this approach is similar to a recently published work [33] where the authors inserted key-gates to lock the FEOL layout and then controlled the routing of the related key-nets through the BEOL. However, there is one crucial difference between theirs and ours—the approach in [33] relies on the insertion of additional logic gates to achieve the required security guarantees, but our scheme benefits from the inherent polymorphism of the RFETs, avoiding any additional modifications while still imposing strong security.

4 Security Vulnerabilities

While the previous section focused on the security promises offered by RFETs, in the present section, we demonstrate the security vulnerabilities for RFET-based circuits. We conduct detailed circuit-level simulations here and provide further analytical studies in Sec. 5.2. It becomes interesting to note that while the same feature of *functional polymorphism* contributes to making the circuit secure, it can be exploited by an opportune attacker for circuit degradation techniques.

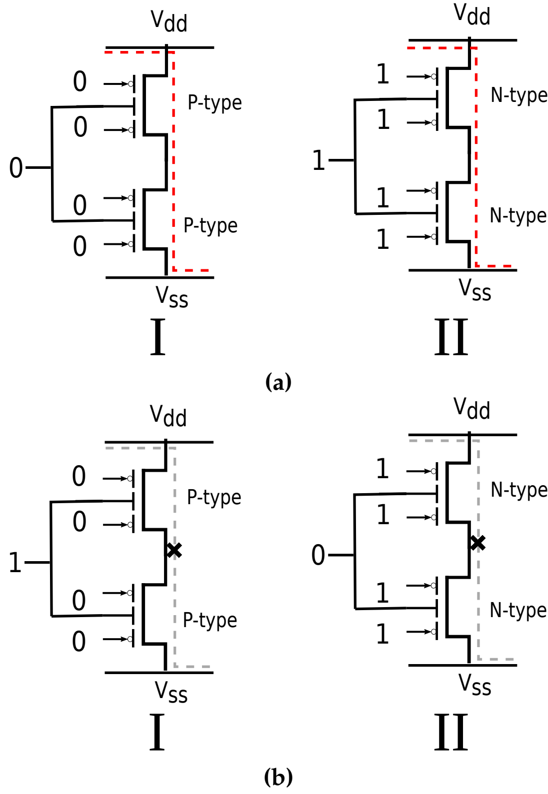

Fig. 8: Manifestations of a modified RFET inverter: (a) two short-circuit configurations, (b) two open-circuit configurations

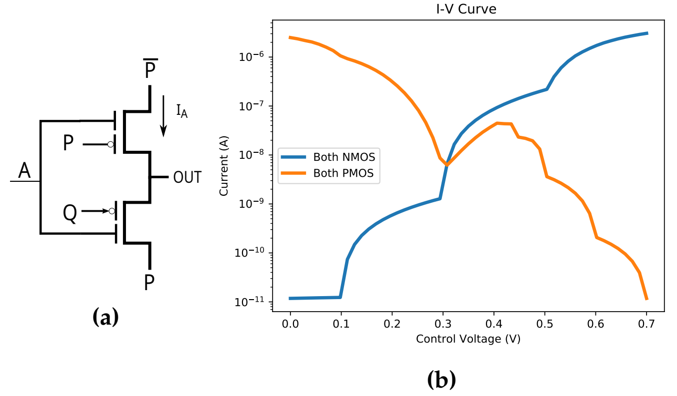

Fig. 9: a) Short-circuit operation possible in case of INVERTER by switching on both the transistor in either p-type or n-type at runtime b) Simulation results showing current reaching 10^{-7}A range for INVERTER in two configurations

4.1 Realization of Short-Circuit and Open-Circuit Scenarios in an RFET-Based Inverter

Conventionally, for CMOS-based logic gates, the PMOS and NMOS transistors realize their specific functionality in pull-up and pull-down networks, respectively. CMOS circuits require a separate pull-up and pull-down network for keeping the output either at logic high (1) or low (0). When the pull-up network is switched on (off), the output is pulled-up (pulled-down) to logic 1 (logic 0). Both networks are simultaneously on and current flows through the transistors only during the switching of the output from logic 0 to logic 1. However, in RFET-based logic gates, the boundary between PUNs and PDNs is somewhat diminished. That is, the same pull-up (pull-down) network can also work as a pull-down (pull-up) network by merely changing the configuration of all the related transistors. The drive strength for RFETs in both n-FET or p-FET configuration is identical [25], which makes this switching between pull-up and pull-down possible, to begin with. This specific property is the root cause of the security vulnerabilities discussed here.

More specifically, the potential exploit in RFET-based circuits arises from the fact that individual RFETs of a circuit can be individually programmed maliciously. RFETs can be used as p-type or n-type transistors, irrespective of their presence in the pull-up or pull-down network, as this behavior merely depends upon the potential at the PG of individual RFETs, as shown in TABLE 1. In essence, given this reconfigurability of individual RFETs, one can either switch 'on' certain transistors in the pull-up and pull-down network to induce a *short-circuit* path from V_{dd} to V_{ss} or switch 'off' both the transistors to induce an *open-circuit* between V_{dd} and V_{ss}, respectively.

This is explained by an example in Fig. 8. The figure shows a normal inverter, where particular "misconfigurations" lead to unfavorable conditions. Fig. 8a shows an inverter comprising of two RFETs where both the transistors are configured as either p-type (I) or n-type (II). This is possible when both the gate terminals, PG and CG, are assigned the same potential. In these cases, both the transistors are switched-on, leading to a conducting path between the V_{dd} and the V_{ss} nodes. We refer to this configuration as *short-circuit path*. The reverse scenario, shown in Fig. 8b, occurs when the potential at the PG and CG is configured oppositely. In these cases, both the transistors are switched-off, leading to an *open-circuit path*. The red dashed lines in Fig. 8a signifies a short-circuit path, while grey dashed lines in Fig. 8b signifies an open-circuit path.

To understand the behavior in the two conditions mentioned above, we study an RFET-based inverter (shown in Fig. 9a) and compare its normal operation to the case when it is misconfigured. We explore scenarios when a potential bias is swept across the gate terminal of an inverter. The simulations are performed using *Cadence Virtuoso* with a simple table-based RFET model used in [17].

Figure 9b shows the amount of current drawn from the inverter in which both the transistors are *On* and are either fixed as n-type or p-type (indicated by red and blue curves respectively). The red curve shows the condition in which the control gate input to the inverter is fixed at logic 1 while the program gate input of the lower transistor, Q, is changed from logic 0 to logic 1, signifying reconfiguration from p-type to n-type. Similarly, the blue curve exhibits the case when the control gate input to the inverter is fixed at logic 0. Simultaneously, the program gate input P is varied from logic 1 to logic 0, signifying reconfiguration from n-type to p-type. Both the cases can create a short-circuit or open-circuit path between V_{dd} and V_{ss}.

1) *Large Short-Circuit Currents*: One can see from Fig. 9b that the amount of current flowing through the inverter during the short circuit condition is of the order of

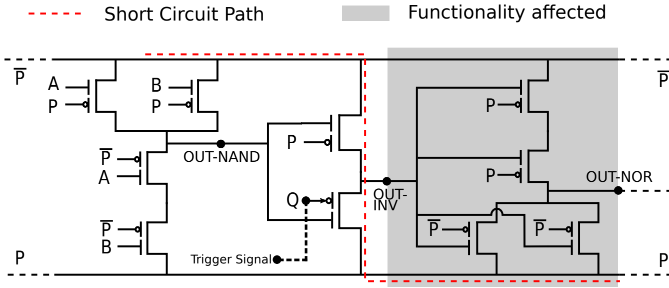

Fig. 10: A sub-circuit consisting of NAND, INVERTER and NOR, where the INVERTER is misconfigured

10^{-6}A in both the cases, which is almost 10^5 times higher than the static leakage current of 10^{-11}A under normal operating conditions for SiNW RFETs. The unique property of RFETs is that any logic gate can be similarly reconfigured at runtime, which gives rise to the possibility of creating multiple short-circuits, resulting in a large amount of current being drawn from the power source. Other unwanted effects include high power dissipation and excessive localized heating near the affected portion of the circuit. Such a scenario could lead to accelerated aging of the circuit and critical reliability issues like the thermal shutdown of the system [38]. The increase in power dissipation will eventually lead to a reduction in the battery lifetime, especially for portable devices like laptops, mobile phones, etc.

2) *Fault in Logic Values*: In the cases shown in Fig. 8a and 8b, the output of the inverter will be at an indeterminate voltage level. If this misconfigured inverter drives a gate, it can have faulty inputs and evaluate wrong logic outputs. This phenomenon can continue further in a combinational circuit leading to a chain reaction where preceding nodes can lead to indeterminate voltages at subsequent nodes. Eventually, this would lead to incorrect functioning of the overall circuit.

4.2 Circuit Evaluation on Sub-circuits

We notice the current and voltage repercussions in the case of an individual inverter. This section discusses how such a misconfiguration in an inverter (or any RFET-based logic gate) can disrupt the normal functioning of combinational or sequential sub-circuit elements.

4.2.1 Combinational Sub-circuits:

Here we evaluate the impact of the misconfigured inverter in a small circuit, as shown in Fig. 10. For simplicity, we study the effect on a small sub-circuit consisting of NAND, INV, and NOR. However, the conditions can be exploited in any combinational chain. Such behavior is triggered when the gate inputs are in a steady-state and not switching[1]. The NAND gate on the left drives the misconfigured inverter, which in turn drives the NOR gate. The program input *Q* of the inverter (as discussed in the case study above) is fixed at logic '1' (for n-type configuration) for it to work undetected as a regular inverter. When the inverter is misconfigured, the current and voltage values for all the possible configurations and inputs are listed in TABLE 2. It can be observed from the table that current values reach to orders as high as 10^{-6}A, as opposed to 10^{-12}A under regular, static operation. This million times more current can potentially damage the power source and cause thermal issues and affect the logical output of the overall circuit. The implication of the security vulnerability in the case of RFETs is somewhat similar to the issue of "latchup," which was a serious problem with CMOS integration in the early days of VLSI fabrication [38]. "Latchup" was caused due to faulty transistors in either pull-up or pull-down network, which caused current discharge from V_{dd} to V_{ss}.

The activation of such a scenario also leads to the node *OUT-INV* in Fig. 10 to be at an undefined state, which consequently affects the input to the NOR gate. It implies that the subsequent gates in the combinational path have the likelihood of going into an indeterminate state (NOR's output can affect subsequent logic gates). This can cause unpredictable bit-flips along the way, affecting the functional correctness of the circuit. TABLE 2 shows the voltage levels at various nodes of the circuit in Fig. 10 for all possibilities of inputs. Especially, it can be remarked that from the table, the *OUT-NOR* node voltages for *Both p-type* (*Both n-type*) configurations are held at logic high (low) when compared to the one under normal operation. This further strengthens the point that such a malicious modification can interfere with the functional correctness of the circuit.

It can be further observed that such a scenario is possible in other combinational chains and is independent of the logic-gate used as shown in TABLE 2. The inverter is a special case because here, we need only one input of an individual transistor (Q, as shown in Fig. 10) to be wrongly configured. Larger gates like *XOR*, etc., can also be disrupted as it depends upon the number of transistors which are wrongly configured to enable the realization of such *short-circuit* or *open-circuit* paths.

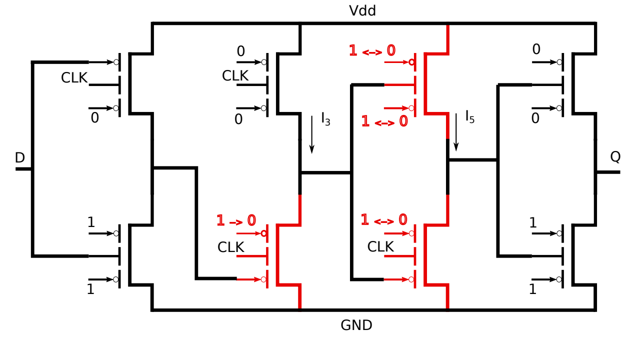

Fig. 11: True Single Phase Clock (TSPC)-DFF based on RFETs as proposed in [23]. Just by changing the polarity of the lower transistor at the second stage (Precharge stage when CLK = 0 and evaluation phase when CLK = 1).

4.2.2 Sequential Sub-circuit:

We carry evaluations over sequential components as well since most of the real circuits are sequential circuits. For this analysis, we have considered the RFET-based *True Single Phase Clock (TSPC) D-flip-flop* as proposed in [23], which

1. This is assumed considering the fact that once the scenario of *short-circuit* or *open-circuit* is activated, irrespective of the previous state, the logic gate will be in one of the conditions as mentioned in TABLE 2.

TABLE 2: Simulation results showing current drawn and voltage values for different cases of inputs and configurations

	Overall Current Drawn through INV			At Node OUT-INV			At final Output		
	Normal Operation	**Both n-type**	**Both p-type**	**Normal Operation**	**Both n-type**	**Both p-type**	**Normal Operation**	**Both n-type**	**Both p-type**
INV-INV-INV									
Output-1 = 0V	389pA	3.05uA	8.19pA	0V	31.13mV	686mV	0V	700mV	0V
Output-1 = 700mV	83pA	251pA	2.5uA	700mV	106.7mV	609mV	700mV	700mV	0V
XOR-INV-NAND									
OUT-XOR = 0V	414 pA	251.8pA	2.5uA	700mV	31.06mV	610 mV	0V	700mV	0V
OUT-XOR = 700mV	11.58pA	3.0 uA	7.17pA	0V	100mV	688mV	700mV	700mV	0V
NAND-INV-NOR									
OUT-NAND = 0V	414pA	251pA	2.5uA	700mV	31mV	609mV	0V	700mV	0V
OUT-NAND = 700mV	11.8pA	3.055uA	29.01pA	0V	106mV	688mV	700mV	700mV	0V

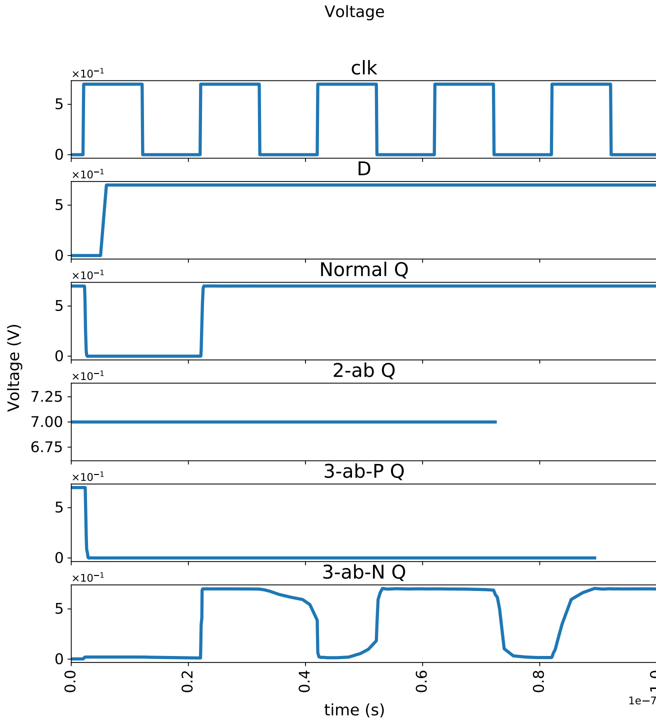

Fig. 12: Voltage values in normal and both short-circuit and open-circuit condition for TSPC-DFF. 2-ab means when the transistor in the second stage is configured incorrectly. 3-ab-N and 3-ab-P implies that both the transistors in the third stage are either configured as NMOS and PMOS respectively.

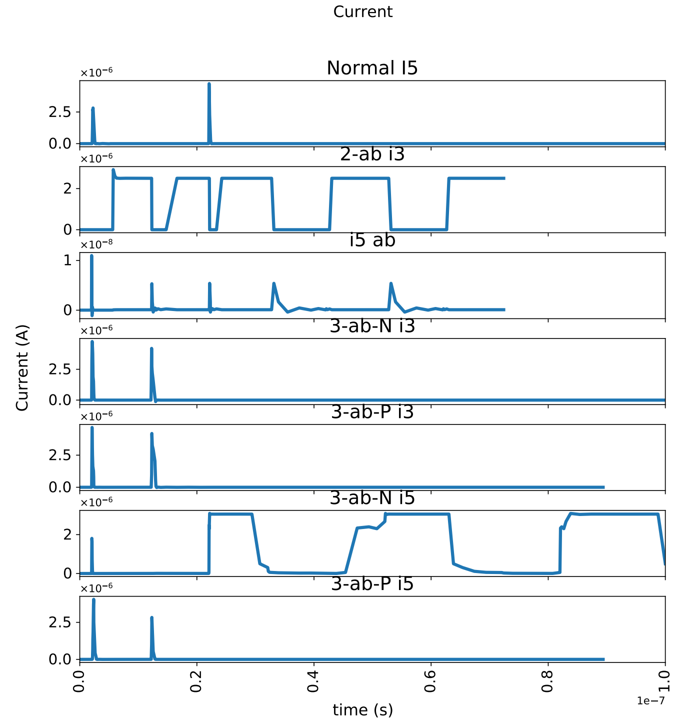

Fig. 13: Current values in normal and both short-circuit and open-circuit condition for TSPC-DFF. 2-ab means when the transistor in the second stage is configured incorrectly. 3-ab-N and 3-ab-P implies that both the transistors in the third stage are either configured as NMOS and PMOS respectively.

is based on the original CMOS-based TSPC DFF [39]. It is shown in Fig. 11. Unlike the original design [39], the proposed design in [23] contains only a single transistor in both pull-up and pull-down network[2]. The TSPC-DFF contains four stages of different inverter designs. While the first and the third-stage inverters are clock enabled inverters (low and high-edge enabled respectively), the second and the fourth inverter stages are dynamic (CLK is connected to the CG of the transistor) and static inverters, respectively.

To observe the effects of RFET-based vulnerabilities, we evaluate by introducing misconfigurations at the gate terminals of RFETs of the various stages of the TSPC-DFF. In Fig. 11, the transistors' configuration (shown in red color) in the second and third stage has been changed subsequently. When applied in isolation (the configurations are changed either in the second stage or the third stage), both conditions lead to abnormal outputs being observed at the output Q along with a large amount of current flowing through the second and third stages. We study the effects when the transistors of the second and the third stage are independently configured in a wrong manner.

Effect on voltage: Figure 12 shows voltages at the output Q in various conditions. In Fig. 12, the normal Q represents the condition when the TSPC-DFF functions normally as the output Q follows D at the next positive edge of clk. The next figure represented by *2-ab* Q (ab = abnormal case) shows

2. This is because RFETs can have multi-independent terminals on a single channel as explained in Section 2.1.

when the transistor in the second stage is misconfigured, here as p-FET. The output Q remains at logic high. Similarly, when both transistors at the third stage are either configured as p-FET or n-FET, the output Q is represented in Fig. 12 as *3-ab-P* Q or *3-ab-N* Q.

This corrupts the output Q, which is seen to be at logic 0 since the pull-up network never conducts in the fourth-stage. Similarly, when both the transistors in the third stage are configured as NMOS, we observe that output Q follows the clock as a stable pull-up ceases to exist for the fourth stage. We can see that due to the incorrect configurations, the TSPC-DFF fails to give the desired output.

Effect on current: Figure 13 shows the current through various paths of TSPC-DFF. A corresponding correlation with the voltage can be established for the currents I_3 and I_5. While the currents I_3 and I_5 in normal scenario show spikes for the second rising edge of the clock, this is highly disrupted when the second or third stage transistors are misconfigured. When the transistor in the second stage is misconfigured, there are several spikes of currents (both I_3 and I_5), which leads to useless dynamic power consumption and potential reliability concerns. Hence, such RFET vulnerabilities can be exploited for both combinational and sequential components of the circuit.

4.3 Reliability Concerns: A Consequence of Short-Circuit Scenario

We have seen that security vulnerabilities using RFETs can have a detrimental impact on both the voltage at the output stage and the current in one or multiple paths in the circuit. With devices based on emerging-nanotechnologies, where individual transistors can be configured to either a p-type or n-type behavior, current based implications can be detrimental, especially for reliability. We have seen that the current surge of the order of 10^5 - 10^6 is possible with RFET-based circuits. Such current surges in one or more of the circuit paths can induce reliability issues that can be presented in reduced quality-of-service (*QoS*). An increased current through the circuit paths for an extended time can further lead to an increase in temperature. Higher current leads to a similar mechanical stress which can cause a material degradation of the transistors. Since RFETs are similar to CMOS in terms of geometry and material, they also have dielectric separation (HfO_2) between metal contacts [26]. Such current-related issues directly impacts the dielectric and other metal-semiconductor contacts and, hence, are also detrimental in case of RFET-based circuits.

Most of the reliability issues are dependent upon current and temperature. Due to increased temperature and current, prominent reliability issues like electromigration, interconnect, and self-heating can surface, derailing the circuit's normal functionality. This can further accelerate the aging of the underlying circuit. Aging happens in circuits due to stress-related to high voltages or temperatures [38]. These can cause unwanted power dissipation and can also contribute to reducing the *mean-time-to-failure* $(MTTF)$, which is expressed as:

$$MTTF = \frac{A_{EM}}{J^n} \exp(\frac{E_{aEM}}{KT}) \quad (2)$$

where A_{EM} is a material-dependent constant, J is the current density, n is empirically determined constant with a typical value of 2 for stress-related failures, E_{aEM} is the activation energy of electromigration, K is the Boltzmann's constant, and T is the temperature. It can be seen from expression 2, that $MTTF$ is inversely proportional to the square of current density (J^2). Hence, even if we assume temperature to be constant (which is the worst-case scenario), with a current surge, the $MTTF$ is accelerated by order of 10^{10} to 10^{12}. Generally, by the basic rule of thumb, $MTTF$ is 10 years. In the case of RFET-induced current surge, the $MTTF$ reduces from 10 years to 0.3 seconds. This implies that in just 0.3 seconds, circuit's functionality is corrupted.

An important thing to note here is that the current surge's effect is more detrimental and pervasive to the circuit compared to the voltage effects. Voltage effects might be masked by other parts of the circuit and may not present itself at the circuit's output. But, in the scenario of the current surge, once triggered, this can ruin the underlying circuit even if the circuit functions according to its specifications. The silent nature of this effect is more adverse from a security point of view.

4.4 Implication of the Proposed Security Vulnerability

One of the major implications of the proposed security vulnerability is the realization of *Hardware Trojans*. Given that pull-up/pull-down network in an RFET-based logic gate could be falsely configured, any gate could become a Trojan in the field. Hardware Trojans represent security concern in the outsourced IC supply chain. Trojans are malicious modifications inserted by adversaries present either in the design house or the foundry which can cause the host circuit to (i) deviate from their specified functionality, (ii) leak sensitive information, and/or (iii) become unreliable or fail at some point in time [40]. The proposed security vulnerability can thus be exploited to realize reliability Trojans which are activated either by (i) aging effects such as electromigration, or (ii) internal or external side-channel triggers. Such benign or reliability hardware Trojans have been explored before in the literature and can compromise the reliability of all or selected chips [40], [41].

An integral part of such Trojans will be to use triggering mechanisms that remain inactive during the testing phase, triggered by some external or internal trigger conditions at any given point in time. Within the context of RFET-based circuit, it is important to note that Trojan realization in this case is different from traditional Trojans, as it can evade detection during the testing phase. For example, *Iddq* testing[3] which can detect electrical faults, will not be able to identify these scenarios (*short-circuit/open-circuit*) because, in the normal case, when RFETs are not misconfigured, there will not be any such electrical faults.

The main contribution of this work, however, is the study of security vulnerability in RFETs-based circuits, not the advancement of Trojan insertion or their defence mechanisms. An actual hardware Trojan implementation and case

3. *Iddq* testing measures the supply current of a chip (or a given module) in the quiescent state (i.e., when the circuit is not switching and inputs are held at static/constant values) to detect manufacturing and/or electrical defects.

studies is scope for future work, as it would require some in-field attack and in-depth analysis against various defence schemes.

Another potential use of such security vulnerability is in developement of *kill-switch*, which is any manipulation of the chip's software or hardware that would cause the chip to shut down the intended functionally, for example, to shut down an F-35's missile-launching electronics [15]. The above scenarios of short-circuit and open-circuit represent an interesting opportunity for a security-enforcing designer. The ability to configure the RFETs at run-time to seriously disrupt the normal functioning can be highly favorable for safety-critical applications such as for military use.

In order to evade such security vulnerability, it is important to use RFETs' functional polymorphism in a controlled way. Static connections to program gate of individual transistors within a single logic gate should be avoided. Dynamic connections using inverter(s) within the gate boundary to enable both P and P′ signals can be one of the ways to get away with security vulnerability. The use of inverter(s) within the gate boundary implies that P's additional input is fed as an extra input to the enlarged standard cell. The inverter then converts the input P to deliver two signals – P and P′, which drives the RFETs accordingly to ensure that the respective pull-up and pull-down network are correctly connected. This ensures that no external signal can come into the gate to disrupt the complementary pull-up and pull-down networks. However, this approach does have a drawback of higher area overheads due to extra inverters necessary within the gate boundary.

Other measures include devising *Guard rings* [42] or *Voltage controllers* [43] to localize the vulnerability effect to a certain part of the circuit. Guard rings simply help to localize the current drawn from V_{dd} and can be placed for few RFETs together. They also help to provide a low resistance path for the current to flow without harming other parts of the circuit. Similarly, voltage controllers can be activated to cut-off certain sections of the circuits from the rest of the circuits in case of excessive current drawn from the voltage source. However, these methods are preventive orthogonal measures which help only after the attack has been realised so as to localize the effect and prevent damage to other parts of the circuits. They also have additional area and power overheads. More importantly, they have been designed primarily for signal correctness in conventional CMOS circuitry and were generally designed separately for p- and n-type transistors. However, in case of RFETs, where such boundary is blurred, these works will require additional investigation.

5 Analytical Evaluation

5.1 Investigating the Security Promises

In this section, we discuss our findings concerning security promises using RFETs. We explore the security guarantees by leveraging RFETs in logic locking and split manufacturing, respectively.

Setup for security evaluation: For the scenario of malicious end-users, we evaluate the locking approach against powerful *exact* SAT-based attacks [16] and *approximate* SAT-based attacks [44]. We assume different transistor-level configurations for RFETs, which can work as NAND/NOR, AND/OR, and XOR/XNOR, respectively. RFET-based locking is implemented as individual 2-to-1 MUXes as outlined in [30]. Since both the SAT-based attacks (exact as well as approximate) require the netlists to be in *BENCH* format, we employ custom *Python* scripts to implement the locking approach. The SAT-based attacks [16], [44] are carried out on a server with five compute nodes; each node has two 14-core Intel Broadwell processors, running at 2.4 GHz with 128 GB RAM. The time-out for the attacks ("t-o") is set to 48 hours. We implement random logic locking for our experiments; ten different sets for each benchmark is generated, ranging from 10% to 50% locking, in steps of 10%.

TABLE 3: Average runtime (in seconds) for SAT-based attacks [16] for different randomly locked configurations on selected *ITC-99* benchmarks. Time-out (t-o) is 48 hours. For each benchmark ten trials are used.

Benchmark	10% Locking	30% Locking	50% Locking
b14_C	18.43	6,183.84	t-o
b15_C	21.47	5,398.43	t-o
b17_C	312.32	21,324.76	t-o
b18_C	1,543.73	156,378.23	t-o
b19_C	4,678.79	t-o	t-o
b22_C	143.21	9,762.74	t-o

Metrics for security evaluation: We attribute the attacks' average runtime as an empirical, yet essential indicator for a design's resilience. We utilize two well-known metrics to evaluate the quality of netlists inferred by successful attack runs. The **Hamming Distance** (HD) quantifies the average bit-level mismatch between the outputs from the attacker's reconstructed netlist and the original netlist. HD reveals the degree of functional mismatch; an HD value of 50% is considered the best-case scenario [30]. The **Output Error Rate** (OER) indicates the probability for any bit per output being wrong while applying a large set of inputs to the attacker's reconstructed netlist. HD and OER are computed using *Synopsys VCS* and functional correctness of the key (dumped by the exact SAT-based attack [16]) is ascertained by *Synopsys Formality* and *Cadence LEC*.

Results for logic locking: Table 3 illustrates the average runtime (in seconds) required for SAT-based attacks [16] to decipher the locking key for randomly locked configurations on selected *ITC-99* benchmarks. We observe that as the number of locked gates is increased (50% locking), the SAT-based attack cannot decipher the locking key within 48 hours. We also examine the resilience of our large-scale locking approach against approximate key recovery attacks like *AppSAT* [44]. More specifically, regarding resilience against *AppSAT*, we executed the attack on the same locked benchmarks. While *AppSAT* ran into time-out after 48 hours, it is programmed to provide its latest, best-as-possible inference as an approximate key before terminating. We applied these keys to (approximately) unlock our netlists and then calculated the HD between this recovered and the original netlist, which gives us a quantified insight into the recovered key's fidelity. We performed the experiments only for those cases where the exact SAT-based attack [16] runs into time-out, which is 50% locking for all *ITC-99* benchmarks. As expected, we observe that the larger the locking scale, the less useful becomes the recovered key, in the sense that

the HD values approach 50% closely, where the mismatch in functional behavior is most difficult to recover.

Security promises using RFETs for split manufacturing: Here, we showcase the security promises where the unique properties of *functional reconfiguration* and *functional polymorphism* can be leveraged for protecting design IP against untrusted foundries using the concept of split manufacturing. As elucidated previously, a designer can protect the entire netlist by choosing RFETs that offer "fixed-functionality" versus RFETs which provide "variable functionality" based on the control signals provided on the program gates (PG) [3]. The conceptual difference with regards to logic locking is that, in the case of split manufacturing, the secret key shall be implemented via connections only in the BEOL with the help of TIE cells (as opposed to a tamper-proof memory). Toward that end, only the program gates of selected RFETs have to be wire-lifted above the desired split layer (see Fig. 7) and driven with constant '0'/'1' signals, which are routed through the BEOL.

Why Proximity attacks will not be successful?: The success of conventional *proximity attacks* is correlated with the type of FEOL-level hints, which can be harnessed to infer the missing BEOL connections. More specifically, the state-of-the-art network-flow attack aims to reconstruct the missing routing of a FEOL design leveraging the following hints like (i) physical proximity between connected cells, (ii) routing patterns of nets, more specifically, the direction of dangling nets in the FEOL, (iii) constraints of load capacitance for drivers, (iv) the non-formation of combinational loops, and (v) timing constraints.

None of the above hints apply to the TIE cells (which supply a fixed '0' and '1', respectively to the PG) in our construction. This is because we can eliminate physical proximity between the TIE cells and the corresponding PG signals of the RFETs by randomizing the TIE cells' placement. The randomization of TIE cells' placement does not impact the placement and/or routing of the underlying design. The hints which emanate from the routing patterns of the FEOL nets can be further obscured by lifting the whole metal segment to the BEOL, note that the usage of stacked vias can be leveraged toward this end. As we wirelift only a fixed number of PG signals, this would ensure minimal disturbance to the routing of the other regular nets in the design. Once the percentage of wires which are lifted across higher metal layers is increased, there would be an increase in congestion (due to scarcity of routing resources), which can be mitigated by increasing the die outlines. The hint of load capacitance constraints does not apply to TIE cells as they are a source for constant signals like '0' and '1'. Since any other logic does not drive TIE cells, the hint of non-formation of combinational loops is also taken care of by construction. Finally, the hint for timing constraints does not apply to TIE cells (and nets) as they provide a fixed/static path to the PG signals.

Results for split manufacturing: In general *proximity attacks* [45] are executed to ascertain the strength of any defense pertaining to split manufacturing [31], [32], [33]. However, the attack binary released in [45] cannot be ported readily to RFETs. Hence, to showcase the efficacy of RFETs for split manufacturing, we conduct a simple, yet effective experiment as follows. We assume that all the regular nets have been correctly inferred by an attacker (using some proximity attack of choice) and only the program gate (PG) signals remain to be deciphered. Since the threat model of split manufacturing dictates the non-availability of a working chip, SAT-based attacks like [16] are not directly applicable. At most, an attacker can apply "random guessing," and we imitate this by using 1 million test patterns on the netlists. Table 4 denotes the HD and OER obtained after an attack executed by attackers present in an untrusted foundry on selected *ITC-99* benchmarks as a function of the percentage of lifted program gate (PG) signals. As we can note from Table 4, the HD value increases as program gate (PG) signals are lifted to higher layers. We also observe that OER approaches the ideal value of 100% even when 10% wires are lifted to higher metal layers, thereby showcasing our approach's efficacy.

TABLE 4: Hamming Distance (HD) and Output Error Rate (OER) in percentage on selected *ITC-99* benchmarks for 1 million test patterns as a function of lifting of program gate signals.

Benchmark	10% Lifting		30% Lifting		50% Lifting	
	HD	OER	HD	OER	HD	OER
b14_C	16	100	34	100	47	100
b15_C	22	100	38	100	49	100
b17_C	19	100	34	100	43	100
b18_C	11	100	24	100	41	100
b19_C	12	100	23	100	39	100
b22_C	21	100	34	100	47	100
Average	**17**	**100**	**31**	**100**	**44**	**100**

5.2 Investigating the Security Vulnerabilities

We carry out a benchmark-based evaluation of the impact of the voltage effects of the security vulnerability on the functional correctness. We explain the experimentation process and present a detailed analysis of the results obtained.

Experimental Setup: As we have seen that with the creation of a *short-circuit* or an *open-circuit* path (see Fig. 8a, 8b, respectively), various nodes in a circuit can have indeterministic and random voltage. These nodes are termed as "misconfigured nodes." To model such indeterministic voltages in our experiments, we have generated random *stuck-at-1* and *stuck-at-0* faults at these misconfigured nodes in the gate-level netlist for *MCNC*, *EPFL* [18] and *ITC-99* benchmark circuits. We use the fault analysis tool *HOPE* to analyze these misconfigured nodes of a circuit.

Each benchmark is tested with 10,000 input patterns and the number of test patterns for which the outputs change is noted. To carry out our analysis, we have iterated from 1 upto 50 randomly selected nodes (logic gates) in the netlist for *MCNC*, *EPFL* and *ITC-99* benchmarks. The three benchmark suites are chosen as they represent the different sizes of electronic circuits. Similarly, the type of fault inserted at each node is also chosen randomly.

Figure 14 shows OER and HD for 10,000 input test patterns for *MCNC*, *EPFL* and *ITC-99* benchmarks. The results presented represent an average of the respective metrics over all 50 iterations. A more in-depth analysis of the benchmark level evaluation is provided below.

Results: The fault created as a result of misconfiguring various nodes in the circuit is said to create the maximum

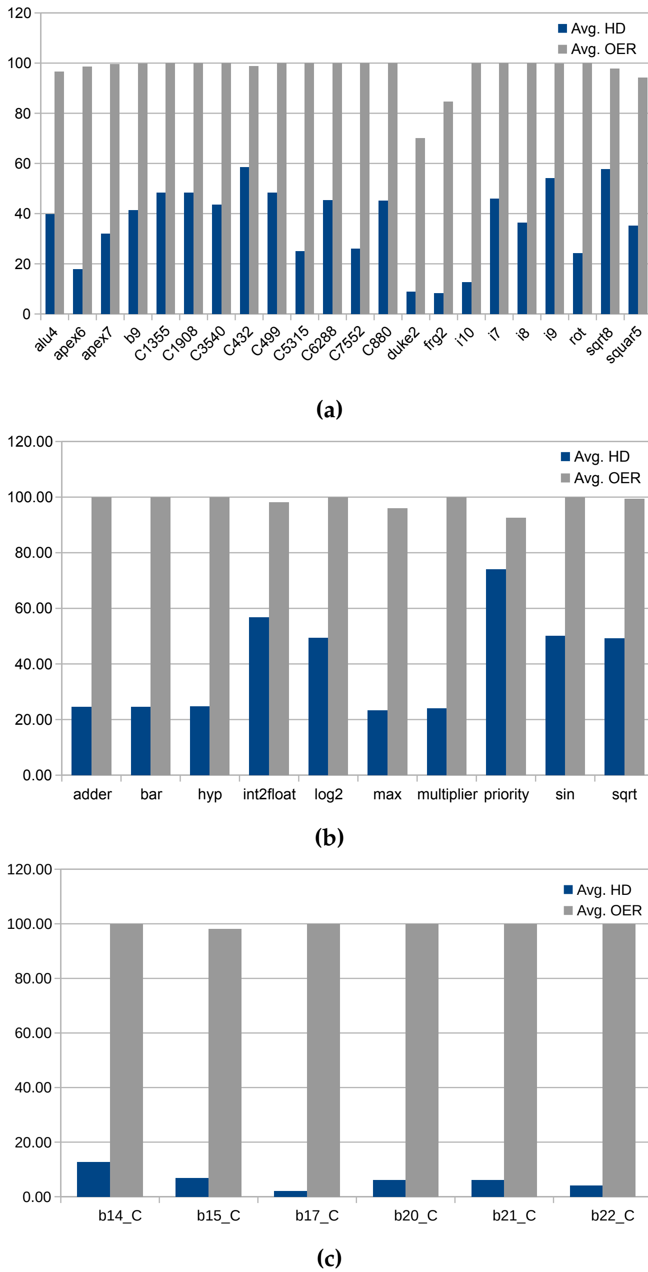

Fig. 14: Evaluation of OER and HD to study the impact of voltage effects on the proposed security vulnerability (a) Average Hamming distance and Output error rate for *MCNC* benchmarks (b) Average Hamming distance and Output error rate for *EPFL* benchmarks (c) Average Hamming distance and Output error rate for *ITC-99* benchmarks.

impact on the fidelity of the circuit's output response when exactly half of the output ports are affected, i.e., the HD is 50%. This is clearly observable in the average trend of HD from *MCNC* to *EPFL* to *ITC-99* benchmarks. Overall, the average HD for the respective benchmark suites is 36.5%, 40.02%, and 6.34%. For *EPFL* benchmarks presented in Fig. 14b, the HD numbers are less than those in the case of *MCNC* benchmarks. Among these, though the HD for *priority* is around 80%, it is equivalent to be around 20% if we converted the inverted faulty output as discussed above. Similarly, for the *ITC-99* benchmarks as shown in Fig. 14c, the average HD for the benchmarks is decreased as compared to the both *MCNC* and *EPFL* benchmarks. As far as the OER is considered, for all the three benchmark suites, i.e. *MCNC*, *EPFL* and *ITC-99* benchmarks, it mostly stabilizes at 99.99%, even with few misconfigurations.

We notice that, as the size of the benchmarks increase, the HD stabilizes at lower values. However, an observation here is that there are exceptions, and the variations in the output bits suggest that *fault-masking* plays an important role. *Fault-masking* implies that the impact of a given fault in one part of the circuit may be masked due to another fault occurring at some other part of the circuit. The second reason can be attributed to the fact that, as the faults are inserted randomly, the distribution may not be uniform across the size of the benchmarks and hence can end up affecting only a small fraction of the total number of output ports. These are the reasons why *MCNC* benchmarks, especially *duke2*, a relatively small benchmark, demonstrates lower HD. On the other hand, *frg2* and *i10*, which are reasonably large, show lower HD due to masking effects.

6 Conclusion

This work has highlighted the security promises and potential vulnerabilities in circuits based on emerging reconfigurable nanotechnologies.

Design-for-security schemes such as logic locking and split manufacturing have been evaluated for RFETs-based circuits. Using benchmark-level evaluations, these schemes established that transistor-level reconfigurability can provide effective security solutions as 100% OER and 31% HD is achieved for split manufacturing over *ITC-99* benchmarks.

We further demonstrated how circuit-level vulnerabilities could be exploited for RFETs using the very same transistor-level reconfigurability. We showed how such vulnerabilities occur due to faults or misconfigurations of individual transistors; using circuit-level simulations, *short-circuit* or *open-circuit* scenarios have been demonstrated. Such scenarios impact the current and the voltage levels, and they manifest in sequential as well as combinational circuits. Its severity can be gauged from the fact that such misconfigurations can occur in an actual RFET-based circuit.

This work aims to open up interesting future research directions regarding prospects pertaining to hardware security (in terms of both security promises and security vulnerabilities), which can accelerate the commercial integration of RFETs in electronic circuits. The present work lays the basic foundation on the security vulnerabilities in RFET-based circuits. The electrical repercussions caused by such a scenario are far more severe as it can present itself in the form of meta-stability, change of critical paths and higher dynamic and static power dissipation. Such scenarios are like double-edged swords and can be applied for both *kill-switch* or *hardware Trojans*.

Acknowledgments

The authors would like to thank Dr. Jens Trommer, NaMLab gGmbH, Dresden Germany for his valuable inputs.

REFERENCES

[1] S. Patnaik, M. Ashraf, O. Sinanoglu, and J. Knechtel, "Obfuscating the interconnects: Low-cost and resilient full-chip layout camouflaging," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 12, pp. 4466–4481, 2020.

[2] M. T. Bohr and I. A. Young, "CMOS scaling trends and beyond," *IEEE Micro*, vol. 37, no. 6, pp. 20–29, 2017.

[3] S. Rai, J. Trommer, M. Raitza, T. Mikolajick, W. M. Weber, and A. Kumar, "Designing efficient circuits based on runtime-reconfigurable field-effect transistors," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 3, pp. 560–572, March 2019.

[4] Y.-M. Lin, J. Appenzeller, J. Knoch, and P. Avouris, "High-performance carbon nanotube field-effect transistor with tunable polarities," *IEEE Transactions on Nanotechnology*, vol. 4, no. 5, pp. 481–489, 2005.

[5] S. Tanachutiwat, J. U. Lee, W. Wang, and C. Y. Sung, "Reconfigurable multi-function logic based on graphene p-n junctions," in *Design Automation Conference*, 2010, pp. 883–888.

[6] A. Heinzig, S. Slesazeck, F. Kreupl, T. Mikolajick, and W. M. Weber, "Reconfigurable silicon nanowire transistors," *Nano letters*, vol. 12, no. 1, pp. 119–124, 2012.

[7] M. De Marchi, D. Sacchetto, S. Frache, J. Zhang, P.-E. Gaillardon, Y. Leblebici *et al.*, "Polarity control in double-gate, gate-all-around vertically stacked silicon nanowire FETs," in *IEEE International Electron Devices Meeting*, 2012, pp. 8–4.

[8] Y. Bi, K. Shamsi, J.-S. Yuan, P.-E. Gaillardon, G. D. Micheli, X. Yin *et al.*, "Emerging technology-based design of primitives for hardware security," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 13, no. 1, pp. 1–19, 2016.

[9] S. Rai, M. Raitza, and A. Kumar, "Technology mapping flow for emerging reconfigurable silicon nanowire transistors," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2018, pp. 767–772.

[10] S. Rai, S. Srinivasa, P. Cadareanu, X. Yin, X. S. Hu, P.-E. Gaillardon *et al.*, "Emerging reconfigurable nanotechnologies: Can they support future electronics?" in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2018, pp. 1–8.

[11] S. Rai, A. Rupani, P. Nath, and A. Kumar, "Hardware watermarking using polymorphic inverter designs based on reconfigurable nanotechnologies," in *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, 2019, pp. 663–669.

[12] A. Chen, X. S. Hu, Y. Jin, M. Niemier, and X. Yin, "Using emerging technologies for hardware security beyond PUFs," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2016, pp. 1544–1549.

[13] A. Rupani, S. Rai, and A. Kumar, "Exploiting emerging reconfigurable technologies for secure devices," in *22nd Euromicro Conference on Digital System Design (DSD)*, 2019, pp. 668–671.

[14] S. Patnaik, N. Rangarajan, J. Knechtel, O. Sinanoglu, and S. Rakheja, "Spin-orbit torque devices for hardware security: From deterministic to probabilistic regime," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2019.

[15] S. Adee, "The hunt for the kill switch," *IEEE Spectrum*, vol. 45, no. 5, pp. 34–39, 2008.

[16] P. Subramanyan, S. Ray, and S. Malik, "Evaluating the security of logic encryption algorithms," in *Proc. HOST*, 2015, pp. 137–143.

[17] G. Gore, P. Cadareanu, E. Giacomin, and P. Gaillardon, "A predictive process design kit for three-independent-gate field-effect transistors," in *IFIP/IEEE 27th International Conference on Very Large Scale Integration (VLSI-SoC)*, 2019, pp. 172–177.

[18] L. Amarú, P.-E. Gaillardon, and G. De Micheli, "The EPFL combinational benchmark suite," in *Proceedings of the 24th International Workshop on Logic & Synthesis (IWLS)*, 2015.

[19] V. Tenace, A. Calimera, E. Macii, and M. Poncino, "Logic synthesis of pass-gate logic circuits with emerging ambipolar technologies," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 2, pp. 397–410, 2020.

[20] J. Trommer, A. Heinzig, A. Heinrich, P. Jordan, M. Grube, S. Slesazeck *et al.*, "Material prospects of reconfigurable transistor (rfets)–from silicon to germanium nanowires," *MRS Online Proceedings Library Archive*, vol. 1659, pp. 225–230, 2014.

[21] G. V. Resta, S. Sutar, Y. Balaji, D. Lin, P. Raghavan, I. Radu *et al.*, "Polarity control in WSe2 double-gate transistors," *Scientific reports*, vol. 6, p. 29448, 2016.

[22] S. Nakaharai, M. Yamamoto, K. Ueno, Y.-F. Lin, S.-L. Li, and K. Tsukagoshi, "Electrostatically reversible polarity of ambipolar α-mote2 transistors," *ACS Nano*, vol. 9, no. 6, pp. 5976–5983, 2015.

[23] J. Zhang, X. Tang, P. E. Gaillardon, and G. De Micheli, "Configurable circuits featuring dual-threshold-voltage design with three-independent-gate silicon nanowire fets," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 10, pp. 2851–2861, Oct 2014.

[24] M. Simon, J. Trommer, B. Liang, D. Fischer, T. Baldauf, M. B. Khan *et al.*, "A wired-and transistor: Polarity controllable fet with multiple inputs," in *2018 76th Device Research Conference (DRC)*, June 2018, pp. 1–2.

[25] J. Trommer, A. Heinzig, T. Baldauf, S. Slesazeck, T. Mikolajick, and W. M. Weber, "Functionality-enhanced logic gate design enabled by symmetrical reconfigurable silicon nanowire transistors," *IEEE Transactions on Nanotechnology*, vol. 14, no. 4, pp. 689–698, 2015.

[26] T. Mikolajick, A. Heinzig, J. Trommer, T. Baldauf, and W. M. Weber, "The RFET–a reconfigurable nanowire transistor and its application to novel electronic circuits and systems," *Semiconductor Science and Technology*, vol. 32, no. 4, p. 043001, 2017.

[27] J. Wan, G. Bouche, A. Wei, and S. M. Koh, "Integrated circuits with nanowires and methods of manufacturing the same," Apr. 5 2016, uS Patent 9,306,019.

[28] P. Ye, T. Ernst, and M. V. Khare, "The last silicon transistor: Nanosheet devices could be the final evolutionary step for moore's law," *IEEE Spectrum*, vol. 56, no. 8, pp. 30–35, 2019.

[29] M. Simon, A. Heinzig, J. Trommer, T. Baldauf, T. Mikolajick, and W. M. Weber, "Top-down technology for reconfigurable nanowire fets with symmetric on-currents," *IEEE Transactions on Nanotechnology*, vol. 16, no. 5, pp. 812–819, 2017.

[30] A. Chakraborty, N. G. Jayasankaran, Y. Liu, J. Rajendran, O. Sinanoglu, A. Srivastava *et al.*, "Keynote: A disquisition on logic locking," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 10, pp. 1952–1972, 2020.

[31] S. Patnaik, J. Knechtel, M. Ashraf, and O. Sinanoglu, "Concerted wire lifting: Enabling secure and cost-effective split manufacturing," in *23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2018, pp. 251–258.

[32] S. Patnaik, M. Ashraf, J. Knechtel, and O. Sinanoglu, "Raise your game for split manufacturing: Restoring the true functionality through beol," in *55th Design Automation Conference (DAC)*, 2018, pp. 1–6.

[33] A. Sengupta, M. Nabeel, J. Knechtel, and O. Sinanoglu, "A new paradigm in split manufacturing: Lock the FEOL, unlock at the BEOL," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2019, pp. 414–419.

[34] J. Knechtel, S. Patnaik, and O. Sinanoglu, "Protect your chip design intellectual property: An overview," in *Proc. Int. Conf. Omni-Layer Intelligent Systems (COINS)*, 2019, pp. 211–216.

[35] E. Giacomin and P.-E. Gaillardon, "Differential power analysis mitigation technique using three-independent-gate field effect transistors," in *IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*, 2018, pp. 107–112.

[36] M. M. Sharifi, R. Rajaei, P. Cadareanu, P.-E. Gaillardon, Y. Jin, M. Niemier *et al.*, "A novel TIGFET-based DFF design for improved resilience to power side-channel attacks," in *Proceedings of the 23rd Conference on Design, Automation and Test in Europe*, 2020, pp. 1253–1258.

[37] S. Rai, A. Rupani, D. Walter, M. Raitza, A. Heinzig, T. Baldauf *et al.*, "A physical synthesis flow for early technology evaluation of silicon nanowire based reconfigurable fets," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2018, pp. 605–608.

[38] A. K. Das, "Design methodologies for reliable and energy-efficient multiprocessor system," Ph.D. dissertation, 2014.

[39] Y. Ji-Ren, I. Karlsson, and C. Svensson, "A true single-phase-clock dynamic cmos circuit technique," *IEEE Journal of Solid-State Circuits*, vol. 22, no. 5, pp. 899–901, Oct 1987.

[40] K. Xiao, D. Forte, Y. Jin, R. Karri, S. Bhunia, and M. Tehranipoor, "Hardware trojans: Lessons learned after one decade of research," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 22, no. 1, pp. 1–23, 2016.

[41] Y. Shiyanovskii, F. Wolff, A. Rajendran, C. Papachristou, D. Weyer, and W. Clay, "Process reliability based trojans through nbti and hci effects," in *2010 NASA/ESA Conference on Adaptive Hardware and Systems*. IEEE, 2010, pp. 215–222.

[42] K. Domanski, "Latch-up in finfet technologies," in *2018 IEEE International Reliability Physics Symposium (IRPS)*, March 2018, pp. 2C.4–1–2C.4–5.
[43] V. Svilan, T.-M. Chen, K. G. Koniaris, and J. B. Burr, "Method and system for latchup suppression," Aug. 31 2010, uS Patent 7,786,756.
[44] K. Shamsi, M. Li, T. Meade, Z. Zhao, D. Z. Pan, and Y. Jin, "AppSAT: Approximately deobfuscating integrated circuits," in *IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, 2017, pp. 95–100.
[45] Y. Wang, P. Chen, J. Hu, G. Li, and J. Rajendran, "The cat and mouse in split manufacturing," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 5, pp. 805–817, 2018.

Shubham Rai received the B.Engg. in electrical and electronic engineering and M.Sc. in Physics from Birla Institute of Technology and Science Pilani, India, in 2011. He is currently working towards the Ph.D. degree at Technische Universität, Dresden, Germany. His research focus is on circuit design for reconfigurable nanotechnologies and their logical applications.

Satwik Patnaik received the B.E. degree in electronics and telecommunications from the University of Pune, India, the M.Tech. degree in computer science and engineering with a specialization in VLSI design from the Indian Institute of Information Technology and Management, Gwalior, India, and the Ph.D. degree in Electrical engineering from Tandon School of Engineering, New York University, Brooklyn, NY, USA in September 2020.

He is currently a Post-Doctoral Researcher with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA. His current research interests include hardware security, trust and reliability issues for CMOS and emerging devices with particular focus on low-power VLSI Design. Dr. Patnaik received the Bronze Medal in the Graduate Category at the ACM/SIGDA Student Research Competition held at ICCAD 2018, and the Best Paper Award at the Applied Research Competition held in conjunction with Cyber Security Awareness Week, in 2017.

Ansh Rupani received B.E. in Electrical and Electronics Engineering from Birla Institute of Technology and Science, Pilani, Hyderabad Campus, India in 2018. He is currently pursuing M.S. in Distributed Systems Engineering from Technische Universität, Dresden, Germany. He is working as a student research assistant at the Chair for Processor Design at TU Dresden. His prior research is focused on EDA for emerging reconfigurable technologies and exploiting such technologies for hardware security.

Johann Knechtel received the M.Sc. degree in Information Systems Engineering (Dipl.-Ing.) and the Ph.D. degree in Computer Engineering (Dr.-Ing., summa cum laude) from TU Dresden, Germany, in 2010 and 2014, respectively.

He is a Research Scientist with New York University Abu Dhabi, United Arab Emirates. From 2015 to 2016, he was a Postdoctoral Researcher with the Masdar Institute of Science and Technology, Abu Dhabi; from 2010 to 2014, he was a Ph.D. Scholar with the DFG Graduate School "Nano- and Biotechnologies for Packaging of Electronic Systems" hosted at TU Dresden; in 2012, he was a Research Assistant with the Chinese University of Hong Kong; and in 2010, he was a Visiting Research Student with the University of Michigan at Ann Arbor, MI, USA. His research interests cover VLSI physical design automation, with particular focus on emerging technologies and hardware security. He has (co-)authored around 50 publications.

Ozgur Sinanoglu is a professor of electrical and computer engineering at New York University Abu Dhabi. He obtained his Ph.D. in Computer Science and Engineering from University of California San Diego. He has industry experience at TI, IBM and Qualcomm, and has been with NYU Abu Dhabi since 2010. During his Ph.D. he won the IBM Ph.D. fellowship award twice. He is also the recipient of the best paper awards at IEEE VLSI Test Symposium 2011 and ACM Conference on Computer and Communication Security 2013. Prof. Sinanoglus research interests include design-for-test, design-for-security and design-for-trust for VLSI circuits, where he has more than 200 conference and journal papers, and 20 issued and pending US Patents. Prof. Sinanoglu is the director of the Center for CyberSecurity at NYU Abu Dhabi. His recent research in hardware security and trust is being funded by US National Science Foundation, US Department of Defense, Semiconductor Research Corporation, Intel Corp and Mubadala Technology.

Akash Kumar (Senior Member, IEEE) is currently a Professor at Technische Universität Dresden (TUD), Germany, where he is directing the chair for Processor Design. He received the joint Ph.D. degree in electrical engineering in embedded systems from University of Technology (TUe), Eindhoven and National University of Singapore (NUS), in 2009. From 2009 to 2015, he was with the National University of Singapore, Singapore. His current research interests include design, analysis, and resource management of low-power and fault-tolerant embedded multiprocessor systems.