REDCAP: Reconfigurable RFET-based Circuits Against Power Side-Channel Attacks

Nima Kavand*, Armin Darjani*, Giulio Galderisi†, Jens Trommer†, Thomas Mikolajick†‡, Akash Kumar*
*Chair of Processor Design, TU Dresden, Germany, {nima.kavand, armin.darjani, akash.kumar}@tu-dresden.de
†NaMLab gGmbH, Germany, {giulio.galderisi, jens.trommer, thomas.mikolajick}@namlab.com
‡Chair of Nanoelectronics, TU Dresden, Germany

***Abstract*—Power attacks are effective side-channel attacks (SCAs) that exploit weaknesses in the physical implementation of a cryptographic circuit to extract its secret information like encryption key. In recent years, emerging technologies have unlocked new possibilities in designing effective SCA countermeasures with less overhead. Reconfigurable Field-Effect Transistors (RFETs) are a type of beyond-CMOS technology that can be configured at run-time to act as an NFET or PFET transistor and provide two or more independent gates. These features make RFETs potent candidates for implementing hardware security techniques like logic locking and SCA countermeasures. In this paper, we propose REDCAP, a method to add randomness to the power traces of a circuit, employing compact reconfigurable RFET-based gates to make the design resilient against power SCAs. First, we explain the construction and control of reconfigurable blocks with isofunctional configurations inside the circuit. Then, we provide an algorithm to efficiently compose the reconfigurable blocks with other circuit parts to minimize the overhead and enable designers to determine the granularity of the reconfiguration. To evaluate our approach, we performed a Correlation Power Attack (CPA) on the S-box of the Piccolo and PRESENT, two lightweight cryptographic circuits, and the results show that REDCAP can highly enhance the resilience of the circuit against power SCAs.**

***Index Terms*—Hardware security, Side-channel attack, RFET, Correlation Power Analysis**

I. Introduction

Today, with the rapid increase in the usage of the Internet of Things (IoT) and portable computing devices, preserving data security becomes more critical. For this reason, cryptography algorithms (like AES, PRESENT, and Piccolo) are widely used to encrypt sensitive data using a secret key and protect it from unauthorized access. Although these algorithms are considered mathematically unbreakable, their hardware implementation might be vulnerable to side-channel attacks (SCAs). SCAs indicate a group of attacks that aim to extract confidential information like the secret encryption key by analyzing the data-dependent physical behavior of the circuit, like power consumption, delay, or electromagnetic radiation.

Power attacks are among the most effective SCAs that utilize the dependency between input data and dynamic or static power consumption of the circuit. Power SCAs are non-invasive physical attacks that usually need proximity to the victim circuit. These attacks are mainly done in two steps: Gathering power traces from the device and analyzing the traces using different methods to extract the secret information. Simple power analysis (SPA), differential power analysis (DPA) and correlation power analysis (CPA) are the most well-known classical power analysis methods.

This work was financially supported by the German Federal Ministry of Education and Research BMBF under the framework of VE-CirroStrato.

In SPA, the attacker tries to find the required information directly from the power traces, like the famous example of extracting the RSA secret key from an unprotected circuit [1]. DPA divides the power traces into two groups based on a target intermediate value (derived from a hypothetical key) and calculates the difference between the means of these two groups. Only a correctly guessed key (subkey) leads to a high difference between the two means. CPA uses a power model like the Hamming weight of the target intermediate value to increase the efficacy of DPA. The hypothetical key, which results in the highest correlation between the modeled power and real power traces, is considered the correct key. DPA and CPA are more powerful than SPA and can break many unprotected cryptographic circuits.

As the root of SCA vulnerability is the dependency between input data and power consumption of the circuit, all the proposed countermeasures aim to reduce this link. These countermeasures can generally be categorized as "masking" or "hiding" techniques. Masking is an algorithmic level method that seeks to randomize the input by a random mask to make the power consumption depend on the variable hidden mask. In the hiding method, the designer wants to reduce the signal-to-noise ratio (SNR) to conceal the correlation between power and data [2]. This can be realized by equalizing the power consumption for different input combinations (signal reduction) or adding randomness to the physical behavior of the circuit, like using a randomized clock (noise addition).

Several works have proposed different circuit-level equalization solutions, such as using sense amplifier based logic (SABL) or dynamic current mode logic (DCML), to make the power independent of input data. Although effective, these methods lead to significant increases in the power consumption and complexity of the circuit, which hinder us from employing them in power or area budget-limited systems. Besides, in some cases, ideal power equalization is not possible, and the circuit might still leak some information under high-accuracy DPAs [3]. For this reason, many researchers have presented randomization methods. Randomization can be achieved through different techniques, such as inserting random delay in the circuit [4] or reconfiguring the hardware with diverse functionally equal structurally different logic components [5]. Since in these techniques, reconfigurability of underlying hardware is a highly desired feature, most of the randomization solutions have been proposed only for Field Programmable Gate Arrays (FPGAs) [5]–[7]. For example, [5] suggests an FPGA architecture containing static and dynamic parts. The control unit is implemented in the static part, and the dynamic part is

979-8-3503-4859-0/DATE24/© 2024 EDAA

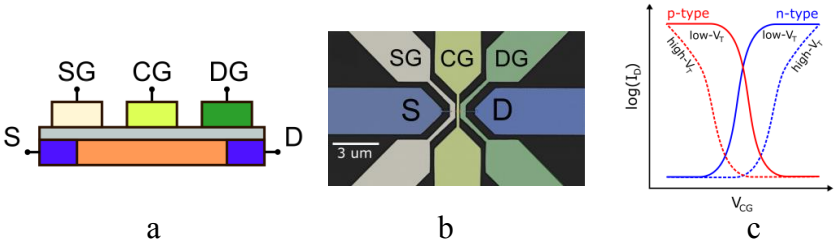

Fig. 1. a) Schematic cross section of three-gated reconfigurable field effect transistor. b) false colored SEM image of the three-gated RFET represented in a). c) general diagram of the transfer characteristic of a three-gated RFET.

dedicated to the AES circuit netlist. Periodically, the whole AES part is replaced randomly with one of the other AES implementation variants by dynamic partial reconfiguration (DPR) to randomize the power profile. In [6], authors also utilize DPR in FPGA, wherein they reconfigure only the SBox parts instead of the entire circuit.

As mentioned, providing such randomization methods is suited to FPGA, and implementing them on Application-Specific Integrated Circuits (ASICs) is costly. Since reconfigurability is not an inherent feature of ASICs, we must implement different variants of isomorphic functions and switch between them at run-time, which imposes an immense power and area overhead on the circuit. Although [3] proposed an energy-aware method based on a combination of power-management techniques and randomized clock gating to avoid higher power consumption, it still suffers from large area overhead due to the isomorphic function units and extra required registers. Besides, the range of power profiles is narrow in this approach [8].

Fortunately, emerging technologies with reconfigurability or polymorphism features like spintronic devices or Reconfigurable Field-Effect Transistors (RFETs) allow us to implement reconfigurable subcircuits inside an ASIC. In PARC [8], authors proposed a randomization method inspired by [3], using STT-MTJ based polymorphic gates. Although promising, employing these polymorphic gates still burdens significant power and area overhead. Furthermore, spintronics has a much higher delay than CMOS-based logic (especially in the write operation). For these reasons, the PARC algorithm has limited options in the circuit to insert polymorphic gates.

In contrast, RFETs, with features like transistor-level reconfigurability, multi-independent gate support, and low standby power, offer compact and low-power reconfigurable gates [9]–[11] with less delay than spintronics. Moreover, RFET follows a CMOS-compatible manufacturing process and can be highly integrated with CMOS transistors. These features make the RFET suitable for implementing different hardware security methodologies. Several works have shown the benefits of RFET in IP protection techniques like logic locking and layout camouflaging or implementing SCA countermeasures [12]. However, all the prior research in RFET-based power SCA-resilient circuits has focused on the power equalization method.

In this paper, we propose REDCAP, a method to provide power SCA resiliency for the circuit by randomizing the circuit power consumption utilizing reconfigurable RFET-based gates. First, we introduce the required reconfigurable gates and present how to create and employ reconfigurable blocks with constant functions to build a circuit with variable power traces. Then, we offer an algorithm to compose the reconfigurable blocks with other circuit parts to reduce power and area overheads. Besides, this composing technique enables the designer to combine two or more reconfigurable blocks into one larger block and determine the granularity of reconfiguration. Finally, we evaluate the REDCAP by performing a CPA attack on the S-box of the Piccolo [13] and PRESENT [14] cryptographic circuits. To test the viability of our approach in different technologies, we have done our experiments using both 14nm Germanium nanowire (GeNW) [15] and 10nm Silicon nanowire (SiNW) [16] RFET models. To the best of our knowledge, this is the first work that analyzes the capability of RFET in adding random noise to the circuit.

The rest of this paper is organized as follows. Section II provides a brief background about RFETs. Section III explains our proposed method, REDCAP, for protecting the circuit against power SCAs. The evaluation results are presented in Section IV, and finally, Section V concludes the paper.

II. Background

Reconfigurable Field Effect Transistors (RFETs) are emerging electronic devices that exploit the nanoscale properties of Schottky junctions to enable to set and dynamically reconfigure the polarity of the device itself between n-type and p-type. Investigated since the early 2000s [17], RFETs can be realized on a large set of different materials and even exploiting different transport mechanisms: in particular, the most common implementation of such device concept is based on the individual control of carriers injection through sharp Schottky junctions formed between undoped silicon channels and nickel silicide source/drain contacts (shown in Fig. 1.a). To enable the realization of this concept, two independent gates must be placed on the source and drain side junctions so that one of them can control the injection of the desired carriers (electrons for n-type, holes for p-type) through the selected barrier, while the other one has the task of blocking the injection of the undesired carriers (holes for n-type, electrons for p-type). The use of materials well known and widely adopted in the semiconductor industry recently made possible to transfer such laboratory-scale technology to large-scale fabrication platforms such as 22 nm FDSOI [18], [19]. Nevertheless, the RFET concept has been applied to different material systems (such as SiGe, Ge, and even 2D materials) exploiting the same transport mechanism control or different ones like, for example, band to band tunnelling (BTBT) [20]. The device level reconfigurability between n-type and p-type can be enriched

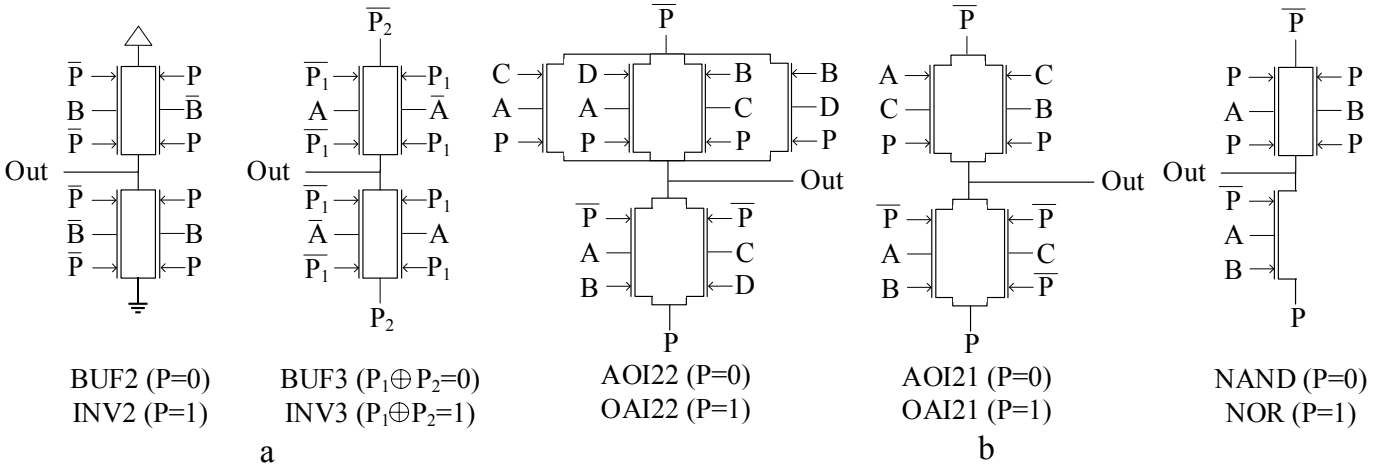

Fig. 2. Required RFET-based gates for REDCAP: a) BUF/INV gates (based on XOR gates) b) Reconfigrable Gates (RGs)

of further features when, for example, three-gated devices are taken into consideration (Fig.1.b). In this case, the additional control over a thermal barrier, raised and depressed in the middle of the semiconducting channel, allows to enable a so called low-V_T switching mode. This way, the single device can be controlled in both terms of its polarity (n-type/p-type) and switching properties (low-V_T/high-V_T), as depicted in Fig.1.c. This multiple independent gate concept [21], can be extended by adding an arbitrary number of gates thus realizing logic functionalities embedded to the devices itself, like the wired-AND demonstrated in [22]. Moreover, the unrivalled control over the injection of undesired carriers [23] leads to devices characterized by extremely low off-currents that reflects in low standby power consumption. Finally, the dependence of the amount of carriers injected through the tunneling barrier from the programming voltage applied to the barrier itself, allow to modulate the on-state current of the transistor [24], introducing like this a certain source of randomness directly at the device level.

III. Proposed Method

This section proposes REDCAP, a method to employ reconfigurable RFET-based logic gates to provide power SCA resiliency by adding randomness to the power traces of a circuit. In the following subsections, first, we present how to build reconfigurable blocks with isofunctional configurations using Reconfigurable Gates (RGs) and create a circuit with a variable power profile utilizing these blocks. Second, we propose an algorithm to compose RGs with their neighboring static or reconfigurable gates in an efficient way to minimize the number of required gates in the circuit.

A. RFET-based Logic Gates and Reconfigurable Blocks

As noted earlier, transistor-level reconfigurability, multi-independent gate support, and low standby power of RFETs allow us to design many logic gates with fewer transistors and less power consumption. This is important because it gives us more chances to insert RFET-based RGs in different circuit parts. Also, we can utilize saved power and area budgets for add-on circuitry needed for security purposes. Several static and reconfigurable RFET-based logic gates have been introduced in [11]. In this paper, we use NAND/NOR and AOI/OAI gates as RGs and BUF/INV gates (which are the same as RFET-based XOR gates presented in [25]). The schematic of these logic gates is illustrated in Fig. 2.

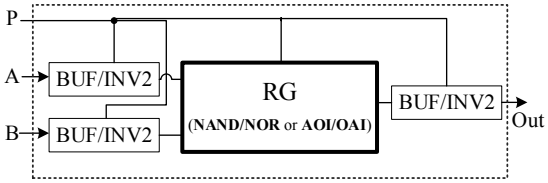

Fig. 3. A simple constant function Reconfigurable Block with one RG

To design a circuit with randomly variable power, we need reconfigurable blocks with constant logic functions in different configurations. We provide such blocks with the help of RGs and De Morgan's law. Fig. 3 shows a simple reconfigurable block with isofunctional configurations. Here, the BUF/INV gates are basically RFET-based XOR gates that can be implemented with only four transistors. A naive way that enables us to add randomness to the power traces is to replace a specific number of basic logic gates like NAND, NOR, AOI, and OAI in the circuit directly with these reconfigurable blocks. However, this method is inefficient in terms of overhead due to the redundant BUF/INV gates. We address this issue with our composing algorithm proposed in Section III-B.

The configuration of the circuit is controlled by a True Random Number Generator (TRNG), which delivers Program Signals (PSs). The RFET-based TRNG proposed in [26] can be used as this programming controller. Alternatively, a ferroelectric based RNG function as proposed in [27] can be directly integrated into the gate dielectric of the RFETs. It is worth noting that reconfiguration of RGs is fast and comparable to the delay of other input signals. Besides, the reconfiguration period is flexible, and the designer can decide how often the

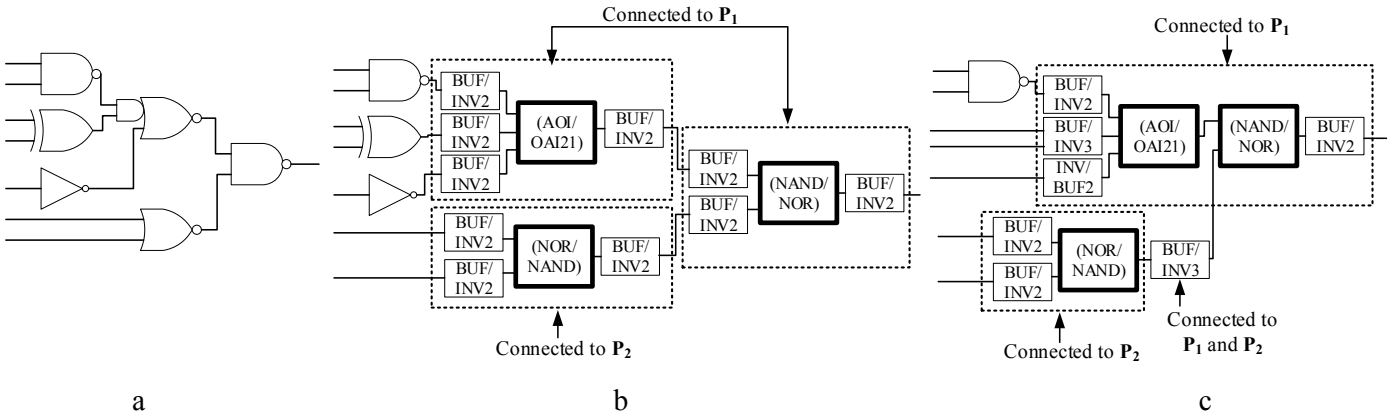

Fig. 4. a) Original circuit b) Result of RG insertion with naive method c) Result of RG insertion using composing algorithm

circuit should be reconfigured to achieve the desired security level.

B. Composing Algorithm

Here, we propose a composing algorithm to minimize the overhead of the REDCAP method. In the first step, we synthesize the target circuit (e.g., AES Sbox) using a logic synthesis tool like Cadence Genus to obtain the gate-level netlist. Then, a percentage of logic gates, like NAND, NOR, AOI, and OAI, are selected for the replacement. In this work, selection is done randomly, and designing an intelligent selection algorithm remains for future work. As mentioned, replacing these logic gates with a whole reconfigurable block results in redundant BUF/INV gates and increases the overhead. Hence, we employ the proposed composition algorithm, wherein static logic gates are substituted with their corresponding RGs (rather than reconfigurable blocks), and then make decisions regarding the insertion of BUF/INV gates. The proposed composing procedure is defined in Algorithm 1. Based on this algorithm, after the placement of the RGs, we visit each RG and make a list of its successor and predecessor gates in the netlist graph. Below, we listed possible composing decisions based on the type of each neighbor gate and compared its result with the naive approach:

1) If the neighbor gate is another RG with a different PS, insert a BUF/INV3 gate, driven by the PS of both RGs, in between. (one BUF/INV3 instead of two separate BUF/INV2)
2) If the neighbor gate is another RG with the same PS, no additional gate is needed. (only a wire instead of two BUF/INV2)
3) If the neighbor gate is an INV gate, replace the INV gate with an INV/BUF2 gate. (one INV/BUF2 instead of a BUF/INV2 and the existing INV)
4) If the neighbor gate is an XOR2 (XNOR2) gate, replace that gate with a BUF/INV3 (INV/BUF3) gate. It is possible because the BUF/INV3 is basically an RFET-based XOR3 gate. (one BUF/INV3 instead of a BUF/INV2 and the existing XOR2)
5) If the neighbor gate is among other logic gate types, insert a BUF/INV2 in between. (same as naive approach)

Algorithm 1 REDCAP Composing Method

Input: $GNet$ **Gate-level netlist of the circuit in HDL (after RG insertion)**
Output: $mGNet$ **Modified gate-level netlist of the circuit in HDL**

$GN\_DAG = CreateDirectedGraph(GN)$
for g in $GN\_DAG.nodes()$ **do**
  if $g.type == RG$ **then**
    $neighbors[] = GN\_DAG.Neighbors(g)$
    $succ[] = GN\_DAG.Successors(g)$
    for n in $neighbors$ **do**
      if $n.type == RG$ **then**
        if $(n.ps\ != g.ps)\&(find(n, succ) ==$ **true**$)$ **then**
          $new\_node.type = BUF\_INV3$
          $new\_node.ps1 = g.ps$
          $new\_node.ps2 = n.ps$
        end if
      else if $n.type == INV$ **then**
        $new\_node.type = INV\_BUF2$
        $new\_node.ps = g.ps$
      else if $n.type == XOR2$ **then**
        $new\_node.type = BUF\_INV3$
        $new\_node.ps1 = g.ps$
        $new\_node.ps2 = n.in1$
      else
        $new\_node.type = BUF\_INV2$
        $new\_node.ps = g.ps$
      end if
      $GN\_DAG.InsertNode(new\_node,\ g,\ n)$
      $new\_succ[] = GN\_DAG.Successors(g)$
      $GN\_DAG.MergeSimilarBufInvs(new\_succ)$
      $mGN = NetList(GN\_DAG)$
      return mGN
    end for
  end if
end for

6) **Branch management:** If the RG has more than one fanout with the same condition, insert only one proper BUF/INV before branches.

These decision rules reduce the overheads significantly because, in many cases, extra BUF/INV gates can be merged with the existing logic gates (cases 3 and 4). Besides, we can build larger reconfigurable blocks and eliminate the need for extra BUF/INV gates in between by coalescing consecutive reconfigurable gates with an identical PS (case 2). This option gives designers the freedom to determine their required granularity of reconfiguration. Fig. 4 shows the result of the composing algorithm for a simple circuit.

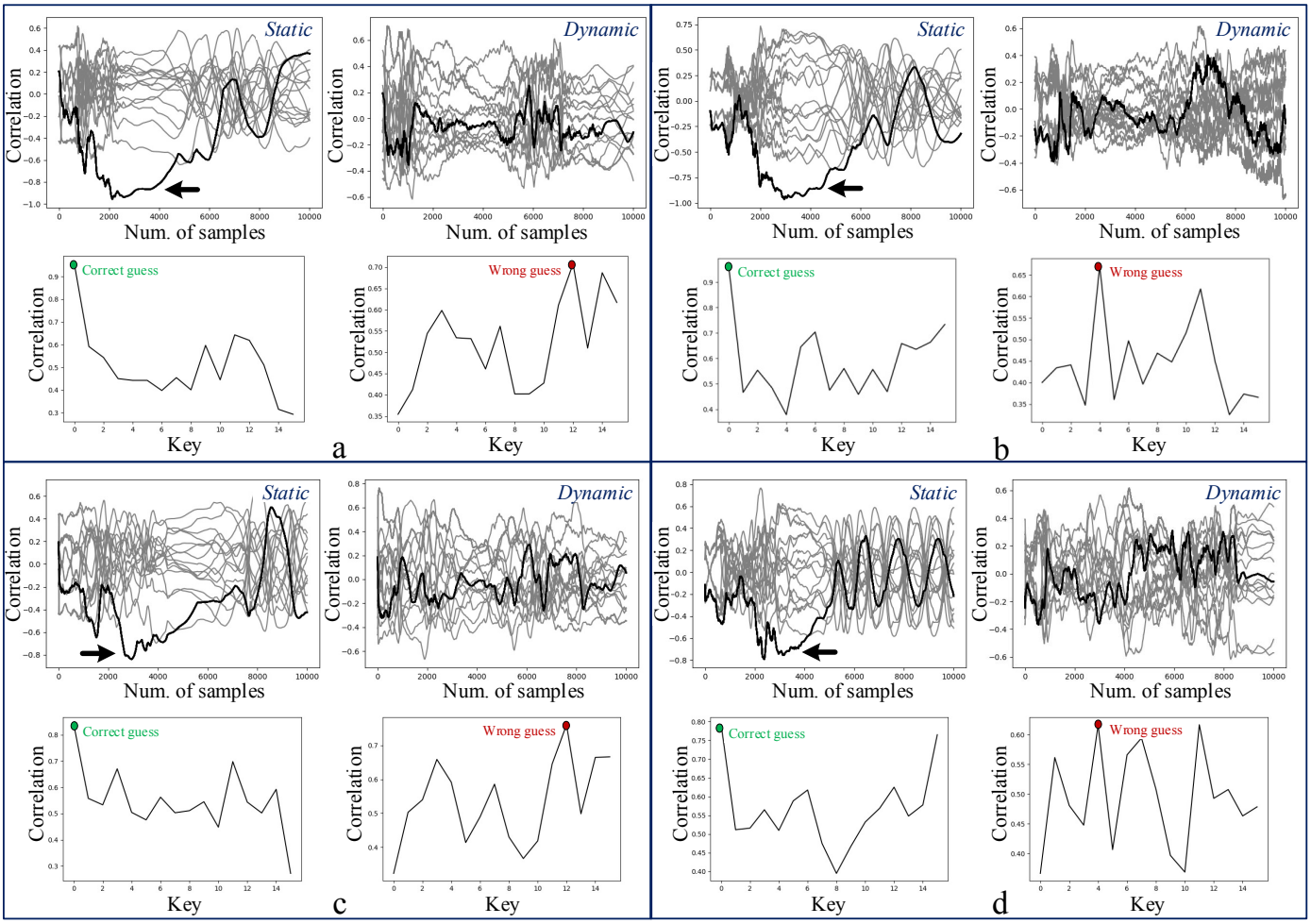

Fig. 5. Result of CPA attack on the S-box of the cryptographic circuits with static configuration and random dynamic reconfiguration. a) Piccolo with GeNW b) PRESENT with GeNW c) Piccolo with SiNW d) PRESENT with GeNW

IV. Experimental Results

In this section, we evaluate the efficacy of REDCAP in raising the power SCA resiliency of the circuit. In our experiments, first, we synthesized the design with Cadence Spectre and general standard cells like NAND, NOR, XOR, etc. Second, we modified the gate-level netlist to its reconfigurable version employing the proposed replacing and composing method. Then, we converted the resulting gate-level netlist to a SPICE netlist. Finally, We performed circuit-level SPICE simulations of the circuit using Cadence Spectre to obtain the power traces. We developed the required RFET-based standard cells in SPICE using 14nm GeNW [15] and 10nm SiNW [16], which are publicly available table-based Verilog-A models for RFET. To analyze the resiliency of the circuit in the worst-case scenario, no additional white noises are considered in our simulations.

To test the resiliency of a valid circuit safeguarded by REDCAP against a real power SCA, we performed the CPA attack on the Sbox of Piccolo and PRESENT, two well-known lightweight cryptographic circuits. The minimum number of gates that need to be replaced with RGs to ensure the required security level of the design depends on several parameters like the technology, implementation, and attack model and should be determined by the designer. Besides, the designer has the flexibility to choose the number of program bits (signals) and RGs controlled by each program signal. Here, we replaced all the NAND, NOR, AOI, and OAI gates with RGs and considered four program signals. Each program signal drives an equal number of RGs, and a new random configuration is applied for every input plaintext.

The correlation of actual and modeled power consumption over time and the highest correlation for each assumed key are shown in Fig. 5. The results show that CPA retrieves the correct key from the circuits with static configuration in both 10nm SiNW and 14nm GeNW RFET technologies. Also, the maximum correlation coefficient for the correct key is high, and it is distinguishable from other wrong keys. However, the attack cannot break the security of the circuit with dynamic reconfiguration with the equal number of power traces.

Although the effect of randomization methods can be reduced by averaging over a large number of traces, it has been shown that the required number of traces increases rapidly with the noise level [3]. Therefore, randomization methods like REDCAP are considered effective countermeasures.

To clearly show the effect of the REDCAP on the power

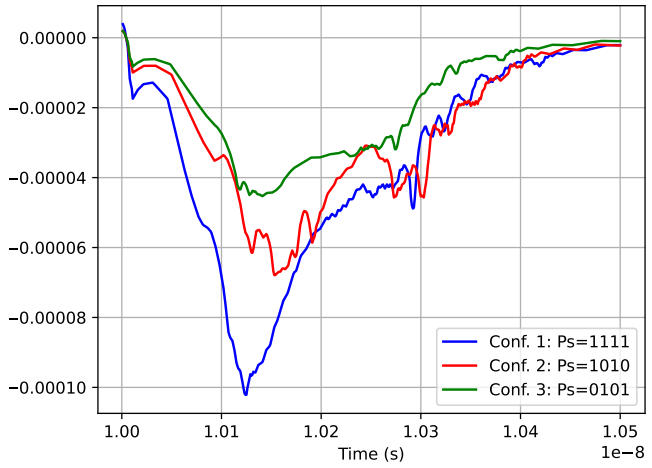

Fig. 6. Difference between power traces of piccolo S-box in different configurations for an identical input transition (input: $0000 \rightarrow 0110$)

traces, we illustrated the supply current of Piccolo's Sbox circuit with three different configurations for the identical input transition in Fig. 6. This graph demonstrates that randomly altering the configuration of the circuit impacts its power behavior considerably.

V. Conclusion

In this paper, we proposed REDCAP, a method to increase the power SCA resilience of circuits employing reconfigurable RFET-based gates. This method explains how to build and control reconfigurable blocks with isofunctional configurations to randomize the circuit power behavior. Moreover, an algorithm is developed to efficiently compose the reconfigurable blocks with other circuit parts to minimize overhead. This algorithm also allows the designer to determine the granularity of reconfigurable blocks. We evaluated our method with two different RFET technologies, and the results show that REDCAP can effectively provide the required randomness level to enhance the protection of the circuit against power SCAs.

References

[1] R. Novak, "Spa-based adaptive chosen-ciphertext attack on rsa implementation," in *International Workshop on Public Key Cryptography*. Springer, 2002.

[2] M. Ouladj and S. Guilley, *Side-Channel Analysis of Embedded Systems*, 2021.

[3] L. Benini, A. Macii, E. Macii, E. Omerbegovic, F. Pro, and M. Poncino, "Energy-aware design techniques for differential power analysis protection," in *Proceedings of the 40th Annual Design Automation Conference*, 2003.

[4] Y. Lu, M. P. O'Neill, and J. V. McCanny, "Fpga implementation and analysis of random delay insertion countermeasure against dpa," in *2008 International Conference on Field-Programmable Technology*. IEEE, 2008.

[5] B. Hettwer, J. Petersen, S. Gehrer, H. Neumann, and T. Güneysu, "Securing cryptographic circuits by exploiting implementation diversity and partial reconfiguration on fpgas," in *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2019, pp. 260–263.

[6] I. Bow, N. Bete, F. Saqib, W. Che, C. Patel, R. Robucci, C. Chan, and J. Plusquellic, "Side-channel power resistance for encryption algorithms using implementation diversity," *Cryptography*, vol. 4, no. 2, p. 13, 2020.

[7] M. M. Ahmadi, L. Alrahis, O. Sinanoglu, and M. Shafique, "Shapeshifter: Protecting fpgas from side-channel attacks with isofunctional heterogeneous modules," in *2023 IEEE 29th International Symposium on On-Line Testing and Robust System Design (IOLTS)*. IEEE, 2023, pp. 1–7.

[8] A. Roohi and R. F. DeMara, "Parc: A novel design methodology for power analysis resilient circuits using spintronics," *IEEE Transactions on Nanotechnology*, vol. 18, pp. 885–889, 2019.

[9] N. Kavand, A. Darjani, S. Rai, and A. Kumar, "Design of energy-efficient rfet-based exact and approximate 4: 2 compressors and multipliers," *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2023.

[10] S. Rai, M. Raitza, S. S. Sahoo, and A. Kumar, "Discern: Distilling standard-cells for emerging reconfigurable nanotechnologies," in *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2020.

[11] 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*, no. 3, 2018.

[12] N. Kavand, A. Darjani, S. Rai, and A. Kumar, "Securing hardware through reconfigurable nano-structures," in *Proceedings of the 41st IEEE/ACM International Conference on Computer-Aided Design*, 2022, pp. 1–7.

[13] K. Shibutani, T. Isobe, H. Hiwatari, A. Mitsuda, T. Akishita, and T. Shirai, "Piccolo: an ultra-lightweight blockcipher," in *Cryptographic Hardware and Embedded Systems–CHES 2011: 13th International Workshop, Nara, Japan, September 28–October 1, 2011. Proceedings 13*. Springer, 2011, pp. 342–357.

[14] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. Robshaw, Y. Seurin, and C. Vikkelsoe, "Present: An ultra-lightweight block cipher," in *Cryptographic Hardware and Embedded Systems-CHES 2007: 9th International Workshop, Vienna, Austria, September 10-13, 2007. Proceedings 9*. Springer, 2007, pp. 450–466.

[15] J. N. Quijada, T. Baldauf, S. Rai, A. Heinzig, A. Kumar, W. M. Weber, T. Mikolajick, and J. Trommer, "Germanium nanowire reconfigurable transistor model for predictive technology evaluation," *IEEE transactions on nanotechnology*, 2022.

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

[17] T. Mikolajick, G. Galderisi, M. Simon, S. Rai, A. Kumar, A. Heinzig, W. Weber, and J. Trommer, "20 years of reconfigurable field-effect transistors: From concepts to future applications," *Solid-State Electronics*, 2021.

[18] V. Sessi, M. Simon, S. Slesazeck, M. Drescher, H. Mulaosmanovic, K. Li, R. Binder, S. Waidmann, A. Zeun, A.-S. Pawlik *et al.*, "S2–2 back-bias reconfigurable field effect transistor: a flexible add-on functionality for 22 nm fdsoi," in *2021 Silicon Nanoelectronics Workshop (SNW)*. IEEE, 2021.

[19] M. Simon, H. Mulaosmanovic, V. Sessi, M. Drescher, N. Bhattacharjee, S. Slesazeck, M. Wiatr, T. Mikolajick, and J. Trommer, "Three-to-one analog signal modulation with a single back-bias-controlled reconfigurable transistor," *Nature communications*, no. 1, 2022.

[20] T. Mikolajick, G. Galderisi, S. Rai, M. Simon, R. Böckle, M. Sistani, C. Cakirlar, N. Bhattacharjee, T. Mauersberger, A. Heinzig *et al.*, "Reconfigurable field effect transistors: A technology enablers perspective," *Solid-State Electronics*, 2022.

[21] J. Zhang, M. De Marchi, D. Sacchetto, P.-E. Gaillardon, Y. Leblebici, and G. De Micheli, "Polarity-controllable silicon nanowire transistors with dual threshold voltages," *IEEE Transactions on Electron Devices*, no. 11, 2014.

[22] M. Simon, J. Trommer, B. Liang, D. Fischer, T. Baldauf, M. Khan, A. Heinzig, M. Knaut, Y. Georgiev, A. Erbe *et al.*, "A wired-and transistor: Polarity controllable fet with multiple inputs," in *2018 76th Device Research Conference (DRC)*. IEEE, 2018.

[23] A. Heinzig, S. Slesazeck, F. Kreupl, T. Mikolajick, and W. M. Weber, "Reconfigurable silicon nanowire transistors," *Nano letters*, no. 1, 2012.

[24] G. Galderisi, C. Beyer, T. Mikolajick, and J. Trommer, "Insights into the temperature-dependent switching behavior of three-gated reconfigurable field-effect transistors," *physica status solidi (a)*, 2023.

[25] J. Romero-González and P.-E. Gaillardon, "An efficient adder architecture with three-independent-gate field-effect transistors," in *2018 IEEE International Conference on Rebooting Computing (ICRC)*. IEEE, 2018.

[26] S. Rai, N. Gupta, A. Bhattacharjee, A. Rupani, M. Raitza, J. Trommer, T. Mikolajick, and A. Kumar, "End-true: Emerging nanotechnology-based double-throughput true random number generator," in *IFIP/IEEE International Conference on Very Large Scale Integration-System on a Chip*. Springer, 2021.

[27] H. Mulaosmanovic, T. Mikolajick, and S. Slesazeck, "Random number generation based on ferroelectric switching," *IEEE Electron Device Letters*, no. 1, 2017.