Stealthy Logic Misuse for Power Analysis Attacks in Multi-Tenant FPGAs

Dennis R. E. Gnad *, Vincent Meyers *, Nguyen Minh Dang *,
Falk Schellenberg †, Amir Moradi †, and Mehdi B. Tahoori *
*Institute of Computer Engineering, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
†Horst Görtz Institute for IT Security, Ruhr-Universität Bochum, Germany
†{falk.schellenberg, amir.moradi}@rub.de
*{dennis.gnad, mehdi.tahoori}@kit.edu, {vincent.meyers, nguyen.dang}@student.kit.edu

***Abstract*—FPGAs have been used in the cloud since several years, for workloads such as machine learning, database processes and security tasks. As for other cloud services, a highly desired feature is virtualization in which multiple tenants share a single FPGA to increase utilization and by that efficiency. By solely using standard FPGA logic in the untrusted tenant, on-chip logic sensors have recently been proposed, allowing remote power analysis side-channel and covert channel attacks on the victim tenant. However, such sensors are implemented by unusual circuit constructions, such as ring oscillators or delay lines, which might be easily detected by bitstream and/or netlist checking. In this paper we show that such structural checking methods are not universal solutions as the attacks can make use of "benign-looking" circuits. We demonstrate this by showing a successful Correlation Power Analysis attack on the Advanced Encryption Standard.**

I. Introduction

Recently, power analysis attacks have been lifted from pure physical attacks requiring access to the device to threats that endanger remotely through software or firmware. This has been shown both in Field Programmable Gate Arrays (FPGAs) as well as Systems on Chip (SoCs) [1, 2]. As FPGAs are getting more widespread adoption in the cloud from companies such as Amazon, Alibaba, and Telekom, a highly desired feature is virtualization and sharing FPGAs between multiple tenants [3–5].

One of the obstacles for virtualizing FPGAs among multiple tenants are powerful side-channel attacks that can be performed by realizing voltage sensors with standard FPGA logic [1, 2, 5, 6], as well as fault attacks working in a similar way [7, 8]. These circuits have unusual properties that are not found in ordinary digital circuits, such as feeding a clock as a data signal [1, 5, 6], or using combinational loops [2, 7]. Consequently, various attempts of checking bitstreams emerged, which analyze FPGA bitstreams and check the resulting netlist for malicious patterns, before allowing them to be loaded into the device and perform fault or side-channel attacks [9, 10].

In this paper, we show how benign logic in existing bitstreams or netlists can be misused as a voltage sensor. Our results show that such sensors are indeed potent enough to be used for standard power analysis attacks. Notably, this way, the sensors are entirely stealthy to any feasible bitstream-checking attempts [9, 10]. This is because the circuit is not altered and still performs its intended meaningful task when not exploited by the attacker. By applying specific data patterns to critical path endpoints at elevated clock rates, they become sensitive to voltage fluctuations. Using post-processing, their results can be used as another type of improvised voltage sensor. Except for their potential timing violations with a secondary clock, measuring voltage in this manner is thus entirely stealthy.

In short, this paper makes the following contributions:

- Misusing existing logic of a normal FPGA design, such that voltage estimates can be measured without the need of previously-used specialized circuits. That makes it harder to detect.
- Post-processing data from path endpoints to estimate voltage fluctuations for performing a successful power analysis attack.

Adversary Model: The adversary model in this paper follows what has been proposed for multi-tenant and cloud FPGAs [1, 2, 5, 7, 9]. The FPGA is split in separated regions that are logically isolated, and an adversary (malicious user) tries to perform power analysis side-channel attacks on a victim user through the common Power Distribution Network (PDN).

II. Background and Related Work

Power analysis side-channel attacks are now becoming possible from one component to another component of a system, because most integrated circuits are supplied by a common PDN for the entire chip or board. Because these electrical connections between attacker and victim exist, voltage fluctuations that are caused by the victim can principally be observed by the attacker that is connected through the same PDN. In FPGAs, it has been shown that such attacks are indeed feasible by using the existing FPGA primitives to create specialized circuits capable of indirect voltage sensing [1, 2, 11], through with sufficient power side-channel information is leaked.

Two categories of digital circuits have been used to measure voltage fluctuations for side-channel attacks Ring Oscillators (ROs) [2] and Time-to-Digital Converters (TDCs) [1, 5, 6] (c.f. Figure 1). These circuits depend on the voltage-dependent speed of transistors, where TDCs are typically faster than RO-based sensors.

As countermeasures to power analysis attacks, *hiding* and *masking* schemes have been used to reduce side-channel

978-3-9819263-5-4/DATE21/©2021 EDAA

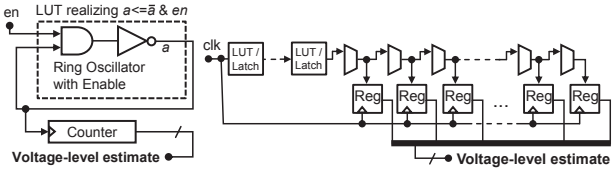


Fig. 1: Specialized circuits used as voltage fluctuation sensors for power anaysis attacks in FPGAs, taken from [9]. **Left:** Sensor based on RO; **Right:** Sensor based on a delay line.

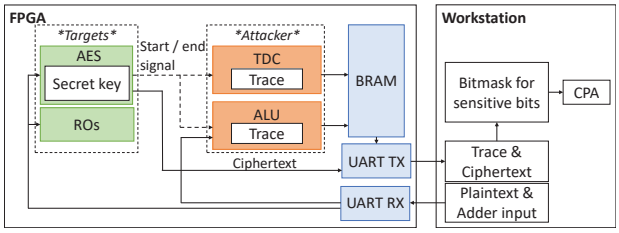


Fig. 2: Overview of the experimental setup

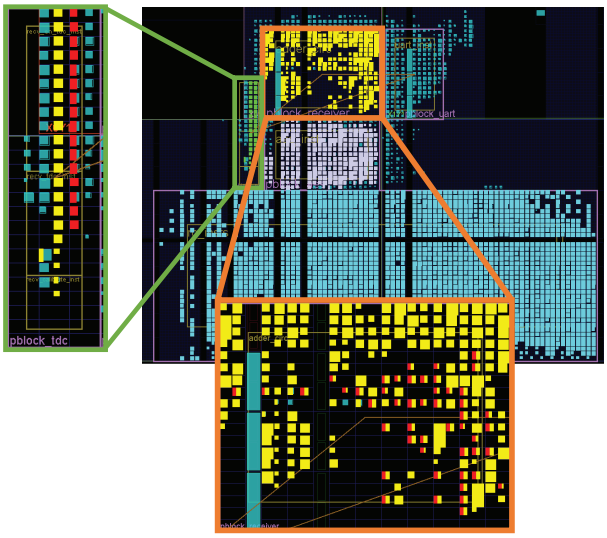


Fig. 3: View of the relevant part of the floorplan. The TDC circuit is marked green and enlarged on the left, AES is marked lilac and the ROs in light blue. The ALU is marked yellow and enlarged on the bottom. Sensitive endpoints of the TDC and ALU are marked red in their enlarged views.

leakage [12, 13].Another approach which is exclusively proposed for cloud FPGAs is bitstream checking. There, bitstreams/netlists are analyzed for malicious circuits, similarly to how software is checked for viruses [9]. Using that approach, many circuits for power analysis and also fault attacks can be detected thus far [9, 10]. These approaches search for known constructs that improvise voltage sensors. In this paper we will show that this is not sufficient.

III. Re-using Benign Circuits as Sensors

In this paper we show how normal circuits can be repurposed as TDCs, sufficient to perform Correlation Power Analysis (CPA) and extract AES keys. That makes the attack extremely stealthy over previous approaches [1, 5, 6].

By operating at timing critical conditions, any path in a circuit can become sensitive to voltage fluctuations. Under normal situations this behavior can not be exploited, but running the circuit at higher clock rates will. This effectively reduces the timing margin of the circuit, making it to produce wrong outputs when the critical path is activated by proper input patterns, and voltage fluctuations occur.

In order to observe transitions in critical path endpoints that we want to use as sensor bits, they need to be stimulated with the proper inputs to the circuit. For that, we use two clock cycles. In the first, we reset the logic to a known value, and the output in the second clock cycle is used as a measurement value. Otherwise, a bit that might not have flipped because of insufficient propagation speed would remain set and no switch from 1 to 0 – or vice versa – would be noticed. As a result, the circuit has to alternate between two modes in the consecutive clock cycles, the "reset" mode and "measure" mode. These modes are activated by proper input stimuli of the circuit.

As an example, we use a ripple carry adder of an ALU. In such an adder, a simple underflow or overflow will use its critical path. That path is one of the first to have faults, if the circuit gets overclocked. Now, voltage fluctuations can be extracted from the ALU result, depending on how far the carry signal can propagate. Additionally, other paths beside the carry signal might be sensitive in a similar way, when the right input stimuli are provided.

IV. Experimental Setup

For the implementation and experiments we use the Xilinx Pynq-Z1 board. An overview of our experimental setup is shown in Figure 2. The data transmission with the FPGA is realized through UART TX and RX. Through RX, the FPGA receives the inputs to the ALU of the Attacker in the FPGA, the inputs to the AES module of the victim, or an enable signal for 8000 ROs. Through TX, the FPGA sends the AES ciphertexts, or the trace data, which is either a sequence of results from the ALU, or the traces from a TDC sensor. While ROs are used as a controlled surrogate for voltage fluctuation generation, the TDC is used as a standard way of measuring differences in voltage levels for side-channel attacks.

We show the respective floorplan in Figure 3, in which the TDC sensor and ALU are shown large with their respective path endpoints marked red. The ALU was synthesized for 50 MHz, and runs at 300 MHz overclocked, i.e. when used as sensor. The TDC sensor is set up for 100 MHz. The AES is synthesized and running at 100 MHz, and uses four parallel SBoxes per clock cycle. To evaluate whether the captured side-channel data indeed contains enough information for a successful attack we perform textbook CPA using a single bit before the final SBox computation as hypothesis.

V. Results

A. Preliminary: RO and AES influence on TDC and ALU

As a preliminary experiment, we look at the response of the output of the ALU Adder when all ROs are activated, and the ALU is overclocked at 300 MHz, and compare it against the results of the TDC-based sensor. The ROs are turned on

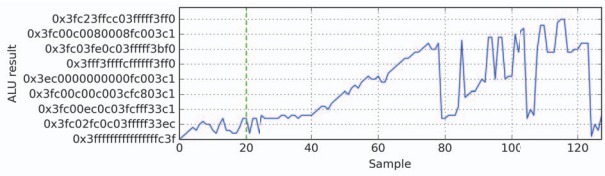


Fig. 4: Absolute value of the toggling ALU bits under influence of 8000 ROs. The ALU runs at 300 MHz where the result of every second clock cycle is shown. The dashed vertical green line indicates when the ROs get enabled.

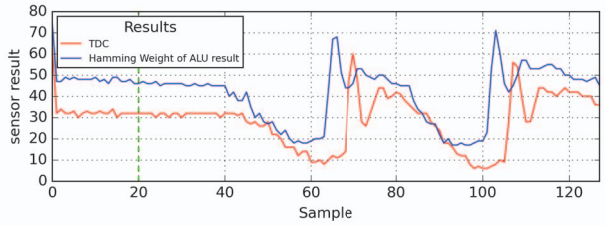


Fig. 5: Influence of 8000 ROs causing two consecutive voltage drops. Results of the TDC sampled at a frequency of 150MHz shown in red. Hamming weight of the toggling sensitive ALU bits is shown in blue. The ALU runs at 300 MHz where the result of every second clock cycle is shown. The time shift between TDC and ALU is due to additional buffer registers inside the circuit. The dashed vertical green line indicates when the ROs get enabled.

and off in a frequency of 4 MHz, where they are gradually enabled and suddenly disabled. We first look at the raw output of the ALU in Figure 4, which shows a rather random output after the ROs get enabled after around Sample 20.

We post-process this output by selecting all bits of the ALU that fluctuate, and then apply the Hamming weight. That result is compared against the output of the TDC in Figure 5. The ROs are gradually enabled from around Sample 40. From that, the TDC output (**red**) goes down from around 30 to 10 on the Y-axis (TDC), which is indicating increased transistor delays inside the TDC from a voltage drop, while for the post-processed ALU result (**blue**), a similar change is observed with minor offsets in sample or sensor result. When all ROs are disabled at around Sample 70, an overshoot occurs, reducing transistor delay that leads to outputs of 60 and 70. A similar behavior is repeated from around Samples 80 to 120. We thus assume, the ALU can be used in this mode for further experiments, such as performing CPA on the AES. However, its sensitivity to minor fluctuations is not yet known from this experiment.

Furthermore not all endpoints in the ALU are relevant for measuring differences in voltage levels because they are retaining their values during the experiments. When activating ROs, 79 of the 192 bits are sensitive to voltage fluctuations. On the other hand, when running the AES module in a similar way, only 40 bits toggle, where 39 of them are a subset of those affected by the ROs.

With this information, we can reduce the ALU results to the bits of interest, shown in detail in Figure 6. The sensitivity of each endpoint can be expressed by its variance. Bits with a higher variance toggle more often and therefore carry more information about the activity on the FPGA. In our

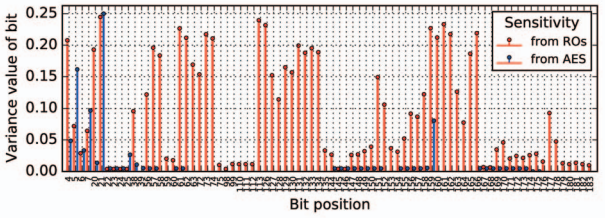


Fig. 6: Variance of each sensitive bit of the ALU under voltage fluctuations from 8000 ROs and AES respectively.

implementation Bit 21 of the ALU has the highest variance.

B. Correlation Power Analysis on the AES

Since the preliminary experiment proves the ALU sensitive to voltage fluctuations from ROs, we proceed to use them to perform a power analysis attack on AES, and compare it to results from measurements with a TDC sensor.

In Figure 7a we show a baseline of the CPA progress results achievable when using the TDC. Because of its linear behavior, just a few hundred traces are needed to clearly distinguish the correct secret key byte (**red**) from all incorrect ones (**gray**).

Since the ALU has almost similar performance in replicating the RO measurements of the TDC, we compare how well the ALU can be used to perform CPA, shown in Figure 7b. In this regard, it does not perform as fast as the TDC to recover the key, but still recovers the correct secret key byte (**red**) with about 150k traces.

C. Correlation Power Analysis with Single Bits

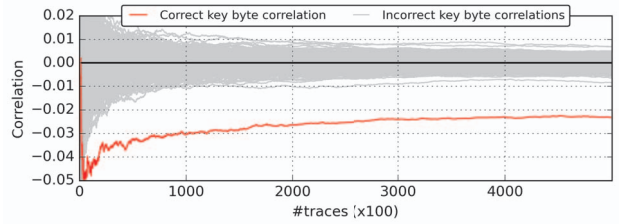
This attack is much more potent if even a single critical path is sufficient as a sensor. Thus, here we analyze if even a single bit or path endpoint can be used for CPA, further reducing the preconditions for the attack. For choosing that single path, we use the bit with the highest variance. For the TDC we use the highest variant bit 32 close to the idle value, and for the ALU, bit 21 (c.f. Figure 6). Please note that this analysis is entirely offline and easily repeated with another device.

For a TDC sensor, using all bits versus only one bit does not make a noticeable difference in key recovery effort, which again just needs a few hundred clock cycles, as shown in Figure 7c. We also just need to increase from about 150k to 200k traces for one case when a single endpoint inside the overclocked ALU is used, as shown in Figure 7d, which proves that even a single critical path can lead to a security breach. For an alternate bit (bit 6) of the ALU, also just about 150k traces were needed.

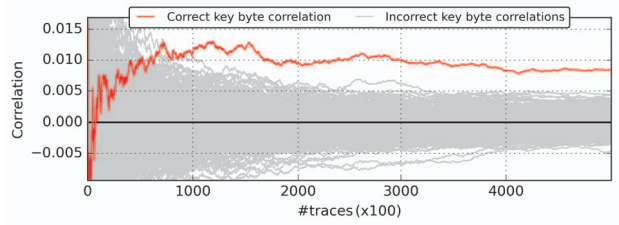
VI. Discussion

The presented results prove that arbitrary and benign digital circuits, such as an ALU, can be misused to sense voltage fluctuations. By that, power analysis side-channel attacks on cryptographic modules can be performed, that previously required special circuits such as TDCs or ROs [1, 2, 5, 6].

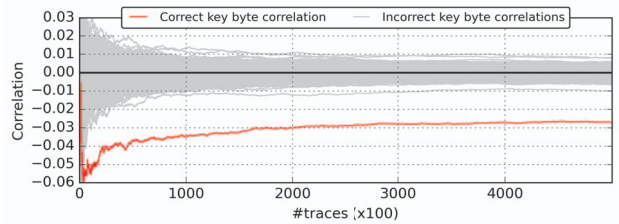
Because even a single bit (path endpoint) of the tested ALU circuit can be used for a successful CPA, more complex structures like carry-chains used in the TDC are not even



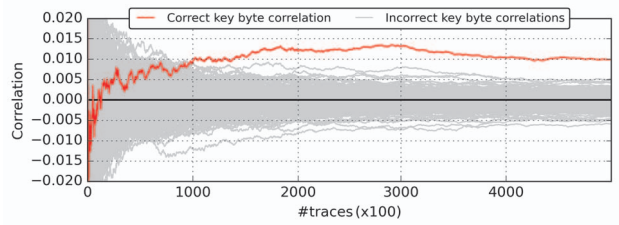
(a) TDC-based sensor measurements at 150 MHz. After the first **few hundred traces**, the correct secret key is already clearly distinguished.



(b) Traces derived from ALU results, clocked at 300 MHz with an effective sampling rate of 150 MHz. The correct secret key is revealed after about **150k traces**.



(c) Measurements with only a single output (**bit 32**) of a TDC-based sensor at 150 MHz. Even when just using a single bit, the correct secret key is already clearly distinguished after a **few hundred traces**.



(d) Traces from a single path endpoint (**bit 21**) of the ALU; clocked at 300 MHz with an effective sampling rate of 150 MHz. The correct secret key is revealed after about **200k traces**.

Fig. 7: CPA attack results on the 1st bit of the 4th byte of the last secret round key of AES. Various correlation progress plots using 500k traces for all 256 key byte candidates are shown; The correlation with the correct key byte is marked red.

necessary. In our experiments, using an ALU Adder merely facilitated the stimuli we had to choose to stimulate all critical path endpoints. In a more complex circuit, Automatic Test Pattern Generation (ATPG) tools and path delay testing can be used to find such stimuli to activate the critical path. But finding the stimuli for a single bit or path can even be done manually.

One approach of bitstream checking [9] uses a strict timing analysis that would indeed detect the approach presented in this paper. For that to work, a mechanism would need to be established, such that a circuit can not select a faster clock than timing analysis suggests. However, we believe that level of strictness is very unrealistic to apply for complex real-world designs. In a real circuit, there are typically many false paths or non-functional paths that are ignored during timing closure, since they have no impact on the correct functionality of the circuit. However, they can potentially be used as sensors for power analysis attacks, and pose a security threat.

VII. Conclusion

Using FPGAs as multi-tenancy devices is highly interesting to further increase computing efficiency, with their security being an important aspect if deployed in cloud computing platforms. In this paper we have shown that even more stealthy attacks are feasible which will be very hard to detect under normal circumstances. By using any benign-looking circuit under false timing assumptions, some critical path endpoints can be used as sensors sensitive to voltage fluctuations, which this paper proves to be sufficient to perform on-chip power analysis attacks, and key recovery on an AES module.

References

[1] F. Schellenberg, D. R. E. Gnad, A. Moradi, and M. B. Tahoori, "An Inside Job: Remote Power Analysis Attacks on FPGAs," in *DATE*, 2018.

[2] M. Zhao and G. E. Suh, "FPGA-Based Remote Power Side-Channel Attacks," in *Symposium on Security and Privacy (S&P)*, IEEE, 2018.

[3] S. A. Fahmy, K. Vipin, and S. Shreejith, "Virtualized FPGA Accelerators for Efficient Cloud Computing," in *CloudCom*. IEEE Computer Society, 2015.

[4] A. Khawaja, J. Landgraf, R. Prakash *et al.*, "Sharing, Protection, and Compatibility for Reconfigurable Fabric with AmorphOS," in *USENIX OSDI*, 2018.

[5] O. Glamočanin, L. Coulon, F. Regazzoni, and M. Stojilović, "Are cloud FPGAs really vulnerable to power analysis attacks?" in *DATE*, IEEE, 2020.

[6] J. Gravellier, J.-M. Dutertre, Y. Teglia *et al.*, "Remote Side-Channel Attacks on Heterogeneous SoC," in *CARDIS*, Nov. 2019.

[7] J. Krautter, D. R. E. Gnad, and M. B. Tahoori, "FPGAhammer: Remote Voltage Fault Attacks on Shared FPGAs, suitable for DFA on AES," *TCHES*, 2018.

[8] T. Sugawara, K. Sakiyama, S. Nashimoto *et al.*, "Oscillator without a Combinatorial Loop and its Threat to FPGA in Data Center," *Electronics Letters*, 2019.

[9] J. Krautter, D. R. E. Gnad, and M. B. Tahoori, "Mitigating Electrical-Level Attacks towards Secure Multi-Tenant FPGAs in the Cloud," *TRETS*, Aug. 2019.

[10] T. M. La, K. Matas, N. Grunchevski *et al.*, "FPGADefender: Malicious Self-Oscillator Scanning for Xilinx UltraScale + FPGAs," *TRETS*, 2020.

[11] K. M. Zick, M. Srivastav, W. Zhang, and M. French, "Sensing Nanosecond-scale Voltage Attacks and Natural Transients in FPGAs," in *FPGA*. ACM, 2013.

[12] P. Kocher, J. Jaffe, and B. Jun, "Differential Power Analysis," in *Advances in Cryptology*. Springer Berlin Heidelberg, 1999.

[13] S. Chari, C. S. Jutla, J. R. Rao, and P. Rohatgi, "Towards Sound Approaches to Counteract Power-Analysis Attacks," in *Advances in Cryptology*. Springer Berlin Heidelberg, 1999.