Power Distribution Attacks in Multi-Tenant FPGAs

George Provelengios, Daniel Holcomb, and Russell Tessier

***Abstract*—The increased use of FPGAs in cloud and embedded computing environments has led to a number of potential security risks. The sizable amount of logic resources in these devices makes them amenable to sharing across multiple untrusted tenants. However, the co-location of multiple independent circuits presents the possibility of malicious fault injection into an unsuspecting circuit. In this manuscript, the ability of one tenant's FPGA circuit to inject delay faults into another tenant's application located at points across the FPGA die via deliberate supply voltage modulation is investigated. To illustrate the risks involved, an RSA encryption key extraction attack is performed by introducing delay faults in hardware via voltage manipulations. This attack does not require modification to the encryption core nor require attack activation synchronized with specific encryption operations. Our work characterizes the magnitude of on-chip voltage changes and fault injections over time in relation to the on-chip location of the malicious circuit once an attack is initiated. Strategies to identify power manipulation using low-cost monitoring circuits that can locate the source of an attack are highlighted.**[1]

***Index Terms*—embedded FPGAs, fault injection, PDN attacks**

I. Introduction

FPGAS are now widely used in a broad range of embedded and cloud computing environments for network functions [1], data search [2], and video processing [3]. While FPGA logic designs have traditionally been created by a single team of designers for dedicated single user deployment, contemporary FPGA logic design is considerably more complex. Embedded FPGA designs often contain multiple intellectual property (IP) cores created by a variety of vendors [4]. In cloud FPGA deployments, users share the FPGA substrate with support circuitry created by a potentially untrusted cloud vendor [3], [5]. Although current cloud vendors typically limit FPGA usage to a single client at a time, the size and cost of FPGAs invites simultaneous device sharing across multiple untrusting cloud users to achieve economies of scale [6]. These three *multi-tenant* use cases test the security limits of current FPGA devices.

Previous research has shown that FPGA supply voltage manipulation can cause circuit timing faults [7], [8], [9], [10], and device reset [11]. Early work showed that over-aggressive manipulation of the power supply for FPGA dynamic voltage scaling leads to delay faults [12], [13]. In the multi-tenant FPGA case, a malicious tenant may spontaneously cause the FPGA supply voltage to drop in an attempt to induce delay faults in another tenant's circuit. In many multi-tenant scenarios, the attacker does not need physical access to the FPGA to perform this type of attack due to FPGA network access, enhancing the threat [14]. Unlike multi-core microprocessors and graphics processors, FPGAs allow users to craft a broad range of computing circuits with arbitrary functionality. Additionally, all current commercial FPGAs contain a single power distribution network (PDN) for each supply voltage making on-chip supply voltage isolation impossible.

In this manuscript, we characterize the threat posed by on-chip voltage attacks in multi-tenant FPGAs and examine a low-overhead approach to detect such attacks. Specific steps include:

- We explore the on-chip voltage response to power wasters, circuits that deliberately waste power, at locations across the die. These voltage responses over time are compared to simultaneous off-chip voltage measurements for two Intel FPGA families. We characterize the voltage responses based on the distance from, and power consumption of, the power waster circuits of the attacker.
- We evaluate the ability of power wasting circuits located in one part of the die to induce timing faults in user circuits situated at locations across the die. Faults in paths with a range of slack values are considered.
- We show that power wasting circuits using a small amount of logic (e.g., thousands of logic blocks) can be used to extract the key from an RSA crypto circuit. Unlike previous approaches, our attack does not require any modifications to the encryption core, nor power wasting that is synchronized with the execution of specific rounds of the encryption operation.
- We examine the use of a network of small on-chip voltage sensors to identify the location of an attack on the FPGA die. This information could be used to mitigate the attack.

Our approaches are evaluated in FPGA hardware under typical operating conditions. DE5a-Net [15] and DE1-SoC [16] boards, containing Intel Arria 10 GX and Cyclone V FPGAs, respectively, are used to evaluate the effects. To characterize the voltage effects in the PDN from the activation of power wasters, a series of experiments were performed using portions of available on-chip logic. Our experiments show that voltage drops caused by inductance ($L\frac{di}{dt}$) can be used to create fault attacks that can even target tenants located far from the power wasting area. These attacks are shown to straightforwardly allow the determination of an RSA encryption key. To address the possibility of power wasting attacks by adversaries, we introduce a monitoring approach using FPGA logic to identify attackers attempting to deploy power wasting circuits.

The remainder of this paper is organized as follows. Background on FPGA multi-tenancy, sensors, and previous voltage

The authors are with the Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, MA 01003 USA (e-mail: gprovelengio@umass.edu; dholcomb@umass.edu; tessier@umass.edu).

[1]This manuscript extends a paper presented at the 2019 International Conference on Field Programmable Logic and Applications. Major additions include an on-chip supply voltage characterization of Intel Arria 10 FPGAs and the implementation of an RSA encryption key extraction attack.

attacks and remediation are described in Section II. Section III describes and analyzes our approach to causing voltage fluctuations. Section IV examines techniques to cause FPGA faults using voltage fluctuations. Section V describes the key extraction attack on an RSA core using PDN fluctuations caused by power wasters. Our monitor-based remediation approach is described in Section VI. Section VII concludes the paper and offers directions for future work.

II. Background and Related Work

A. Multi-Tenant FPGA Threat Model

We consider the following threat model for attacks on the FPGA PDN. Multiple independent users can implement and execute circuits in an FPGA at the same time. Their logic and interconnect resources may be isolated, and each user only has access to the logic design (i.e., bitstream) of their own circuit. There are no physical connections (i.e., wires) shared by the circuits. The software accessed by the designers which interacts with the FPGA is secure as is the interface logic provided in the FPGA. Each user has the flexibility to implement any circuit in their assigned portion of the FPGA.

This multi-tenant threat model arises in a number of user scenarios, as documented in a recent survey [14]:

Untrusted IP cores: User designs often integrate one or more intellectual property (IP) cores from untrusted vendors. Although Trojan detection techniques [17] can be used to identify malicious circuits, in many cases IP cores are distributed as obfuscated or encrypted bitstreams. IP core network connections enhance this threat for systems ranging from embedded systems to single-user cloud FPGA deployments that use IP cores.

Malicious cloud providers: Although unlikely, the possibility of a malicious cloud vendor exists. Effectively, the cloud vendor support circuitry on the FPGA can be thought of as an added tenant whose circuitry is not validated by the user.

Malicious co-tenants: Although not currently supported commercially, it is widely expected that multiple independent users will eventually be able to simultaneously share a cloud FPGA substrate [14], [18]. The ability to commercially use a cloud FPGA for multiple independent users in the future depends on a full understanding of the inherent security weaknesses of current FPGA architectures, including those exposed by the experiments described in this manuscript. Several cloud-based systems that follow this model have been presented as proofs of concept. Khawaja et al. [18] proposed the use of an operating system for shared access to a cloud-based FPGA. The system allows for multiple users to execute circuits at the same time on an FPGA. Device I/O and memory interfaces are fairly shared across users. The PDN in the Xilinx or Intel FPGA is also shared in this model. In Knodel et al. [19], the logic resources in an FPGA located in a cloud node are allocated to interface logic and a collection of virtual FPGAs (*vFPGAs*). Resources are managed by tools running on the node's microprocessor.

In Section V, we describe an attack on an RSA encryption core that involves fault injection via on-chip voltage manipulation. Among our three multi-tenant scenarios, this type of attack could be performed by either a malicious IP core with network access or in a cloud environment by a malicious co-tenant. An RSA encryption key is obtained from the erroneously-encrypted output of the circuit due to an attack.

B. FPGA Voltage Sensing

One approach to identifying FPGA voltage attacks is to implement distributed voltage sensors fashioned from FPGA logic throughout the logic fabric. The ability to identify voltage levels on an FPGA has many uses ranging from verifying safe FPGA operation [20], [21] to the extraction of secret information [22]. Contemporary FPGAs often contain at least one hardened voltage sensor [23] per chip for power supply voltage measurement. Additional on-chip FPGA voltage measurement circuits typically are based on either ring oscillators or time-to-digital converters (TDCs). A ring oscillator (RO) consists of an asynchronous loop containing an odd number of inverters. The frequency of the oscillation can be measured by connecting the RO to a counter. Although RO frequency is affected by temperature [24], voltage fluctuations have a much stronger effect [25]. TDC-based sensors are based on a combinational chain of buffers that are triggered by a clock edge [26]. The output of each buffer is sampled by a clock-triggered flip flop and voltage values can be determined by how far a rising edge propagates through the chain in a clock cycle. Although requiring more resources to implement effectively, TDCs can be used to measure instantaneous voltage changes on the order of a clock cycle [27]. Given our interest in voltage changes due to attacks, we select a network of simpler but highly-effective ROs for our monitoring system.

C. FPGA Voltage Attack Response

On-chip FPGA voltage responses to supply voltage manipulations have been previously studied, although none focus on the specific issues addressed in this manuscript. Zick et al. [26] described a new TDC-based voltage sensor that can identify on-FPGA voltage transients in the nanosecond range. A single sensor was used to characterize changes in TDC delay in the presence of significant signal switching. Although changes in TDC delay over time tracked off-chip voltage measurements taken with an oscilloscope, on-chip voltage values were not determined and voltage responses across the die were not considered. Gnad et al. [25], [28] examined the impact of power waster activation on TDC delay across an FPGA die. TDCs were distributed across the FPGA surface and average and worst-case TDC delays were evaluated over time for varying workloads. Instead, our approach considers on-chip voltage values for numerous individual sensors located columns away from the power wasting source.

D. FPGA Voltage-Induced Faults

Several studies have examined the ability of on-chip FPGA power wasters to drive a chip into reset or induce delay faults in adjacent circuitry. Gnad et al. [11] showed that the sudden activation of thousands of ROs can drive Xilinx FPGAs into

reset, requiring a bitstream reload. Although this attack results in a denial of service, it is not capable of stealthily extracting information from an unsuspecting circuit.

More recently, several researchers have examined the possibility of injecting delay faults into neighboring circuits using power wasters. Krautter et al. [7] examined the possibility of injecting faults into an advanced encryption standard (AES) core at a number of operating frequencies and circuit minimum slack values. This work did not examine the ability of a waster to induce faults at distant locations on an FPGA's die nor consider the effects on signals with a wide range of slack values. In Mahmoud and Stojilović [8], a fault-inducing attack on true random number generators (TRNGs) using ROs was described. The ROs were placed adjacent to TRNGs and TDCs were used to evaluate induced delay changes. Recently, Alam et al. [10] showed that allowing a user to intentionally cause write collisions in FPGA dual-port block RAMs can also induce voltage and temperature fluctuations and result in circuit faults. Our work significantly extends previous fault analysis studies by considering a power waster's ability to induce faults at numerous locations on the FPGA die for paths with a spectrum of slack values.

E. FPGA Voltage Attacks on Encryption Cores

Encryption cores are a popular target for on-chip FPGA side channel or fault injection attacks. Prior work has shown that a shared FPGA PDN creates coupling between power wasters and an encryption core. This coupling has been exploited for side channel attacks [22], [29] in which an encryption key is extracted from an unsuspecting victim crypto circuit. Both RO [22] and TDC-based [29] voltage sensors were used successfully for key extraction via differential power analysis (DPA). In both cases, the power consumption of the crypto circuit was tracked on a per-cycle basis to identify specific key values. Schellenberg's TDC-based attack [29] was successfully replicated on an Amazon EC2 F1 cloud FPGA [30]. Mahmoud et al. [31] inserted a Trojan within the encryption core that is activated by a voltage drop induced by the power waster. This approach requires Trojan insertion during core design. Krautter et al. [7] extracted an AES key by enabling power wasters at specific points in encryption core operation. Our encryption core attack approach does not require core modification nor carefully-timed activation at a specific core execution point to work effectively.

F. FPGA Voltage Attack Remediation

Several studies have examined techniques to identify and suppress significant on-FPGA voltage swings. Shen et al. [32] identify voltage transients caused by user circuits. A clock edge suppressor is used to delay the circuit clock edge in an effort to control voltage drops. Krautter et al. [33] identify circuits that are likely to induce on-FPGA voltage drops (e.g., ring oscillators) from FPGA bitstreams. These circuits can be flagged and removed prior to FPGA bitstream loading in a cloud environment. Zick et al. [26] proposed using voltage information from multiple voltage sensors to monitor device health and potentially suppress malicious behavior. Our work extends these efforts by collating voltage information from numerous on-FPGA voltage sensors to localize the source of attack circuitry on the FPGA, leading to possible remediations.

G. Comparison to Previous Conference Paper

This manuscript builds on our earlier research [9] that evaluated the ability of power wasting circuits based on ring oscillators to induce faults using the FPGA PDN. Mitigation strategies based on a small on-chip voltage sensor network were deployed to identify the source of the disruption. In this manuscript we extend PDN characterization to include Intel Arria 10 GX FPGAs and to show differences across the devices in regards to the timing of fault susceptibility. To further explore PDN attacks, in this work we use power wasters to launch a fault attack against an RSA cryptographic hardware accelerator and extract the secret key. Unlike in side channel attacks, the fault attack demonstrated in this work does not require the fixed placement of sensors.

III. On-chip Attack on an FPGA PDN

The Intel Arria 10 GX (10AX115N2F45E1SG) and Cyclone V (5CSEMA5F31C6) FPGAs used for this work are located on Terasic DE5a-Net and DE1-SoC boards, respectively. Power to the DE1-SoC board is provided from a 12V DC source. The 1.1V internal FPGA core voltage (*VCCINT*) is supplied by a Linear Technology LTC3608 step-down switching regulator at 617 kHz through a 1 µH inline inductor. The Cyclone V device does not include on-chip voltage sensors or hardened monitors.

The DE5a-Net is equipped with a Texas Instruments TPS40422 switching regulator, which steps down the 12V DC input voltage to 0.9V (*VCCINT*) and supplies power to the Arria 10 GX device at 300.75 kHz through a 0.47 µH inductor. The Arria 10 GX device includes an on-chip voltage sensor and a temperature sensing diode, allowing a user to monitor the core voltage and die temperature. Both sensors are located in the upper, middle of the die.

A schematic of a typical on-chip FPGA PDN is shown in Fig. 1. Although publicly-available information about on-FPGA PDNs is limited, the PDN performance of several SRAM-based Xilinx FPGAs are characterized in Klokotov et al. [34]. FPGA PDN impedence characteristics were examined by Zhao et al. [35]. The basic components of our characterization for instantaneous current changes are similar to these works. Power is supplied through the inductor and distributed to core voltage inputs of the FPGA. The resistance and capacitance of the PCB traces and on-die PDN network allow localized voltage fluctuations to occur within the chip, such that different parts of the fabric may have different supply voltages at the same time [34].

A. Methodology and Calibration

1) On-die Voltage Sensors: A voltage monitoring system is needed to observe the PDN response to adversarial power consumption during an attack. To determine on-chip voltage, we measure the voltage at selected positions of the PDN

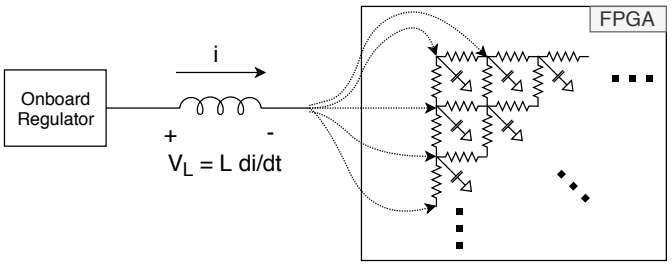


Fig. 1. Schematic of on-chip FPGA power system. A voltage drop occurs across the inductor due to di/dt. A steady-state voltage drop occurs in the PDN due to its resistance.

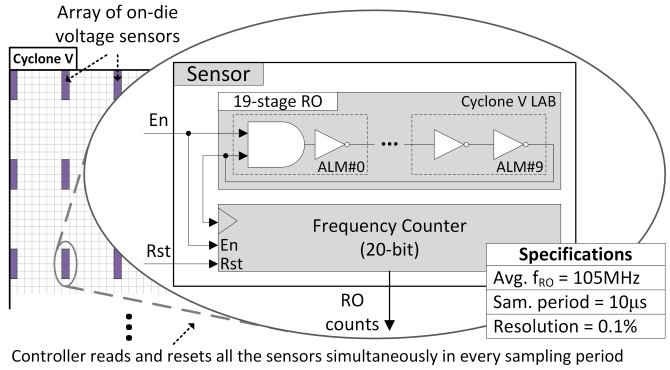
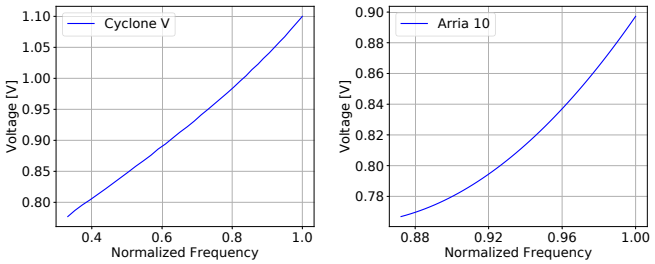


Fig. 2. Schematic of the RO-based voltage sensor.

using ring oscillator-based voltage sensors. The frequency of each oscillator decreases in a consistent way to voltage drops, and a calibration procedure is required to learn the correspondence between voltage and RO frequency. After calibration, frequency measurements made at each sensor can be translated into the voltages that cause them.

Fig. 2 illustrates the architecture of the monitoring system. The sensors are placed on the die forming a regular rectangular grid which is sufficient to perform power analysis attacks [22]. Each sensor consists of a 19-stage RO triggering a 20-bit frequency counter. With 19 inverting stages, the remaining design meets the timing constraints, local delay variations are minimized [36], and RO stacking can be used in a single logic array block (LAB). Although shorter ROs are possible by inserting open latches in the ring to increase the path delay [21], the lack of built-in latch elements in the selected devices makes this technique unsuitable. The 19 inverting stages of the RO design shown in Fig. 2 achieve an average frequency of 105 MHz for Cyclone V and 150 MHz for Arria 10. Measurement periods were 2 µs for the Arria 10 sensor calibration described in the next subsection and 10 µs for all other experiments. These periods provide the capability to detect 0.1% frequency changes, corresponding to a sub-millivolt resolution in supply voltage measurement. We found that the chosen experimental settings provide sufficient resolution for voltage characterization tests without complicating the design of the sensor. Although counting oscillations during a 2 µs or 10 µs can give an accurate estimate of voltage, it does not accurately capture short transient voltage drops with much shorter duration. We will show later in this paper that fast



(a) Cyclone V (b) Arria 10

Fig. 3. Figures show the experimentally derived Cyclone V and Arria 10 calibration curves, which relate frequency changes to the supply voltage values that account for them. The frequency of a sensor is inversely proportional to the propagation delay of the oscillating signal.

transient drops can be observed using path delay circuits which are similar to time-to-digital converters.

2) Voltage Calibration: Since the Cyclone V device on the DE1-SoC board does not include an on-chip voltage sensor, an alternate approach was needed to correlate RO count with voltage. To control the voltage when calibrating the sensors, we desoldered the switching regulator and its output inductor from one DE1-SoC board, and supplied the FPGA core voltage to that board directly from a Keysight E36312A benchtop power supply. We varied the supplied voltage, and at each step measured the FPGA input voltage with a Keysight MSOX4154A oscilloscope, and also recorded the frequency of the sensors using test logic on the FPGA. To prevent any localized voltage drops and ensure that the measured voltage matches the voltage at the sensors, only the test logic and sensors are active during calibration, which minimizes the power drawn by the FPGA. Fig. 3(a) shows the measured correspondence between voltage and frequency of the sensors. The measurements from the RO sensors exhibit a consistent trend across voltages, and the same trend is observed on all sensors, allowing us to calibrate the relationship between voltage and normalized frequency. Unless otherwise noted, all other DE1-SoC experiments described in this manuscript used an unmodified board powered by the on-board switching regulator and output inductor.

Unlike the Cyclone V device, the Arria 10 FPGA is equipped with an on-chip voltage sensor [23] that can be used to calibrate the RO sensors. In a series of calibration experiments, we varied the number of power wasters placed and triggered on the Arria 10 device (Fig. 4) from 8,000 up to 28,000, while monitoring readings from both the on-chip voltage sensor and an RO sensor adjacent to it. Turning on a different number of wasters at each step causes a variation in the reported RO counts and measured voltages by the on-chip sensor allowing us to identify the relationship between RO frequency and voltage on the Arria 10 device. The resulting calibration curve is shown in Fig. 3(b). The curve exhibits a similar trend to the one extracted from the Cyclone V device. The Arria 10 board was unmodified for all experiments, including calibration.

3) Minimizing Temperature Effects: Although RO operation can potentially influence chip temperature, voltage gradients

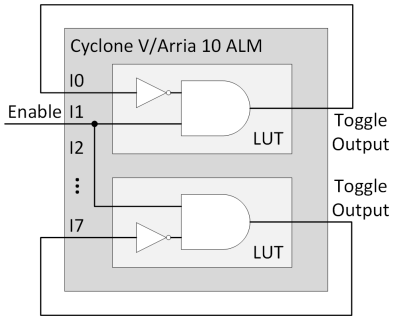


Fig. 4. Power waster circuit mapped to a Cyclone V/Arria 10 ALM device.

TABLE I
POWER CONSUMED BY EACH POWER WASTER INSTANCE VARIES AS THE NUMBER OF INSTANCES IS INCREASED.

Cyclone V		Arria 10	
Number of PW instances	Power / Instance [mW]	Number of PW instances	Power / Instance [mW]
160	1.13	12,000	2.17
1,600	1.02	16,000	2.18
3,200	0.91	20,000	2.21
4,800	0.84	24,000	2.18
6,400	0.75	28,000	2.20

have a much more immediate impact on the measured RO delay than temperature [37], [38]. To minimize heating effects, our experiments were conducted using sampling periods in the sub-millisecond range (e.g., less than $10\,\mu s$) with no more than a hundred samples taken each time. An idle period of a few seconds between iterations was introduced. The ambient temperature during the calibration and characterization experiments was kept at $24\,^{\circ}C$. Neither the on-board nor on-chip temperature sensor of the Arria 10 device reported temperature fluctuations during the calibration process. This result indicates that thermal effects are negligible in our characterization.

B. Adversarial Power Consumption Circuit

We assume that an application on one part of the FPGA is adversarial, and implements a design capable of high power consumption to disturb the PDN. For initial experiments, an area of 1,408 LABs (44 rows by 32 columns) was arbitrarily chosen as a representative example of the Cyclone V FPGA real estate an adversary might occupy, which is 32.8% of the total LABs on the chip. To evaluate the Arria 10 PDN, an area of 11,424 LABs (168 rows by 68 columns) was arbitrarily allocated to the adversary, occupying 23% of the FPGA real estate. In Section VI experiments that consider different attacker area sizes for each device are analyzed. Dynamic power is maximized by circuits with a high amount of switching, so we allow the adversary to instantiate various quantities of single-stage ring oscillators as power waster circuits. Fig. 4 shows an adaptive logic module (ALM) implementing two power wasters. Up to 20 power wasters can be implemented in each Cyclone V or Arria 10 LAB. When instantiating a desired number of power wasters, a script places them uniformly at random locations throughout the allocated region.

$$p_{dyn} = C * V_{DD}^2 * f_{SW} \quad (1)$$

The power consumed by each instance in both examined devices is shown in Tab. I. Power consumption in the Cyclone V device was measured using the modified DE1-SoC board and benchtop supply. Note that the power consumed per instance is diminished as the number of instances grows. This result occurs because the power wasters cause a local drop in supply voltage which slows down their oscillation (reducing f_{SW} in Eq. 1) and causes the switching to occur at lower voltage (reducing V_{DD}^2). Although our later experiments use up to 12,000 power waster instances on Cyclone V with an unmodified board, Tab. I ends at 6,400 because the 5A current limit is reached on the benchtop supply that powers the modified board.

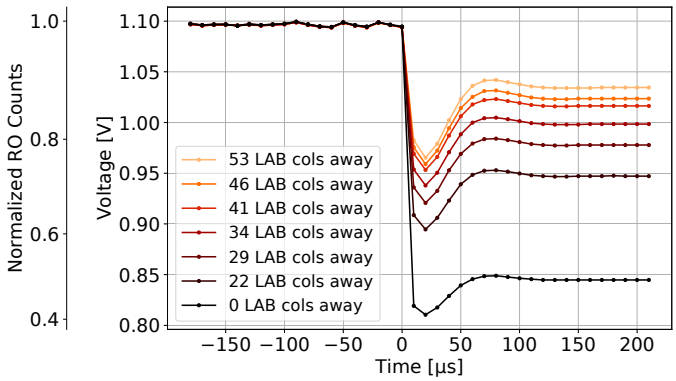
Power consumption in the Arria 10 device was measured using an unmodified DE5a-Net board via an on-board Texas Instruments INA231 [39] power monitor chip on the 12V supply. Unlike the Cyclone V results, increasing the number of wasters in the Arria 10 device appears to have a negligible impact on the power consumed by each instance (two rightmost columns of Tab. I), although this finding is inferred from 12V power measurement and is therefore less direct than the Cyclone V measurements. The INA231 reported that the power consumed reached 78W when 28,000 wasters were activated. Beyond that point, attempts to further increase the number of instances caused a device crash and the loss of the FPGA configuration image.

As mentioned earlier, the modified DE1-SoC board was used only for sensor calibration experiments and measuring the power consumption of the power wasters on Cyclone V (Sections III-A and III-B). All experiments in the remainder of the paper were performed on unmodified DE1-SoC and DE5a-Net boards with their original switching regulators.

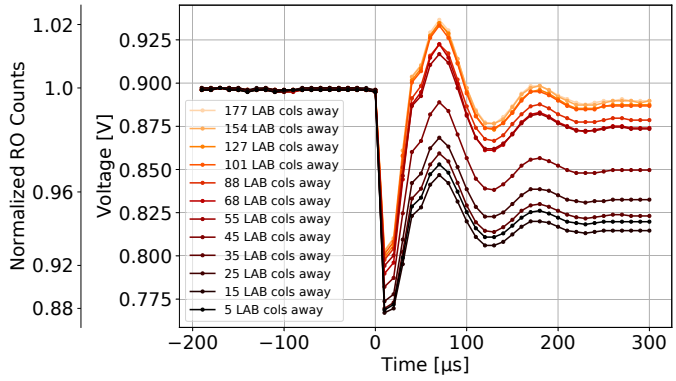
It is important to note that the voltage sensors in this work are only used to measure the effects of the power consumption circuits, not to perform the attack itself. Voltage sensors are calibrated and used to measure on-chip voltage at various time points at locations across the die surface of the FPGA. Such information is used to characterize the effects of the power consumption and potentially perform remediation, not to perform the attack itself.

C. Physical Characterization of Voltage Drop

To evaluate the PDN response of the two devices to high power consumption, experiments are performed with sensors placed at various distances away from the attack circuitry. In the Cyclone V device, 12,000 power wasters turn on at time 0 and the frequency of the sensors, or equivalently their supply voltages (Fig. 5a), drop in response to the attacker's power consumption. The supply voltage measured by each sensor initially drops, undershoots, and then settles back to a steady-state voltage that is lower than the nominal 1.1V for as long as the power wasters remain active. At the center of the power consumption area, the supply voltage drops to a minimum of 811mV and reaches a steady state of 846mV. Sensors farther



(a) Cyclone V



(b) Arria 10

Fig. 5. Normalized RO sensor counts (left axis) and their corresponding voltages (right axis) measured by sensors before and during a power wasting attack that begins at time 0. The legend shows the distance between each sensor and the center of the power wasting region.

away observe a similar behavior but a smaller magnitude of voltage drop.

Similarly, 28,160 power wasters are placed on the Arria 10 fabric and are simultaneously activated while 12 sensors, placed at different distances to the center of the attack, capture the PDN response. The measured voltage at the 12 different sensor locations is shown in Fig. 5b. To a greater extent than in Cyclone V, the voltage drop in Arria 10 is followed by an overshoot before settling back to a steady-state voltage. The sensor farthest from the center of the attack observes a peak-to-peak voltage swing of 125mV, corresponding to 14% of the nominal 0.9V supply voltage. The magnitude of the voltage drop in the Arria 10 device becomes smaller with increasing distance to the power wasting region. This result is consistent with the Cyclone V observations shown in Fig. 5a.

1) Varying the Amount of Power Consumed: As one might expect, attacks wasting more power cause larger voltage drops. The voltage drops are observed at the site of the attack and also in the surrounding area of the die. Fig. 6 shows voltage plotted against distance from the center of the attack on the Cyclone V device; each line in the figure corresponds to a different number of power wasters being instantiated and used in the attack. We can observe in each attack that the supply voltage change can have a far-reaching impact on other circuitry. Even 53 columns away from the center of attack, the supply voltage is reduced from 1.1V to 967mV in the strongest attack.

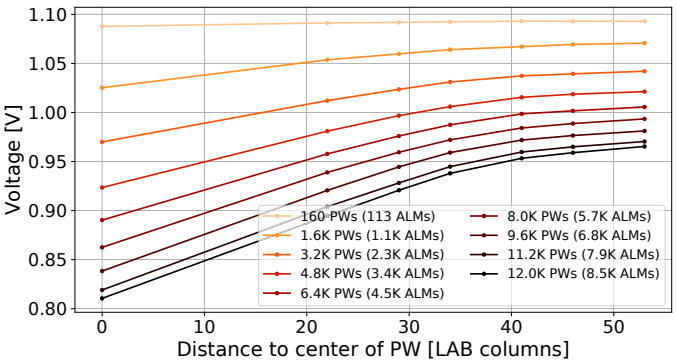


Fig. 6. Voltage change across distance for various number of power wasters instantiated in the Cyclone V device.

IV. Causing Faults via PDN Manipulation

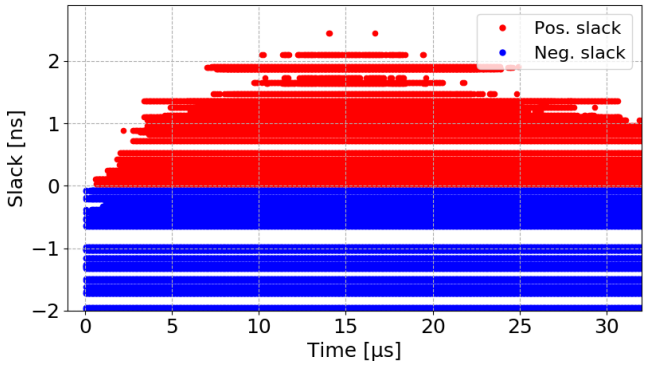
A decrease in supply voltage causes an increase in the propagation delay of combinational logic. Path delay faults will be caused by a reduced supply voltage if the completion time of the combinational results do not satisfy the setup time requirement of the capturing flops. Having shown that aggressive power consumption can cause a far-reaching drop in supply voltage, we now turn to examining whether the voltage drop can induce path delay faults in a victim circuit. For simplicity, we use ripple carry adders as test circuits since their carry chains can provide differing path lengths.

A. Demonstration of Path Delay Faults

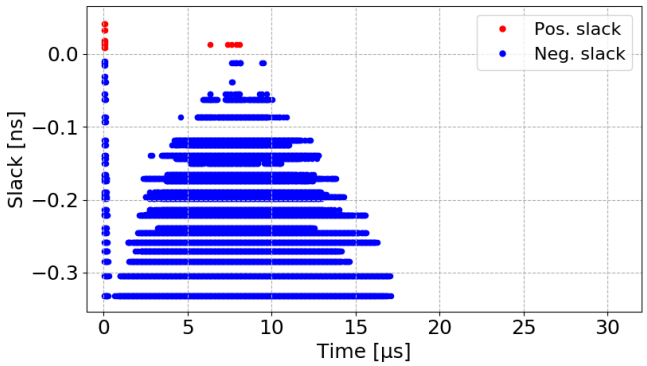
Our first path delay experiments use 12,000 power wasters within a block of 1,408 LABs in a Cyclone V device and 28,160 wasters within a block of 11,424 Arria 10 LABs. The victim (i.e., the ripple carry adder) has been hand placed adjacent to the attack area in a single LAB column, which in the Cyclone V and Arria 10 experiments is 23 and 38 LAB columns away from the center of the attacker, respectively. A script generates vectors that sensitize paths with slack ranging from $+3\,\text{ns}$ to $-2\,\text{ns}$ in the Cyclone V device and from $+0.2\,\text{ns}$ to $-0.5\,\text{ns}$ in the Arria 10 device. The timing slack of each path in an adder instance is reported using the TimeQuest Timing Analyzer [40]. The slow 1100mV $85\,^{\circ}\text{C}$ model is used for the Cyclone V implementation of the adder and the slow 900mV $100\,^{\circ}\text{C}$ model for the Arria 10 implementation. The vectors are repeatedly applied during power attacks and a log is kept with the faults and their timestamps.

Fig. 7 shows the faults that occur from the attack. The X and Y coordinates of each point denote the time and reported slack of the path on which the fault occurred. Paths with more slack are less susceptible to delay faults. Every point on the plot depicts the capture of an incorrect result. Red points denote faults on paths with positive slack, which are paths that meet timing constraints according to the conservative timing model. Blue points originate from paths that have negative slack according to the conservative timing model, but are error free in the absence of an attack.

The results in Fig. 7 indicate that in both devices there is a period in which faults occur (e.g., $10\,\mu\text{s}$ to $20\,\mu\text{s}$ for the Cyclone V device and $6\,\mu\text{s}$ to $10\,\mu\text{s}$ for the Arria 10

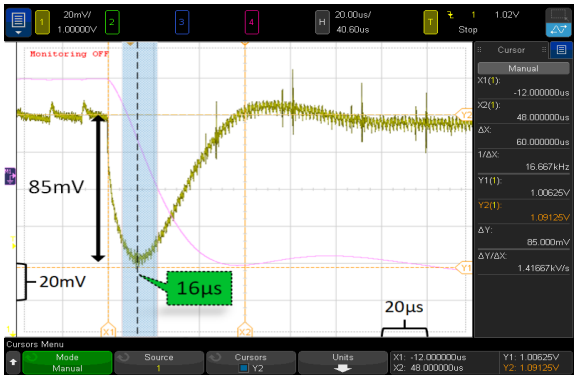


(a) Cyclone V delay fault test. Adder is placed 23 LAB columns away from the center of the attack.

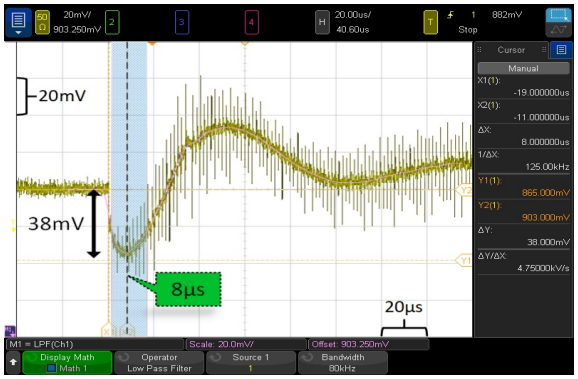


(b) Arria 10 delay fault test. Adder is placed 38 LAB columns away from the center of the attack.

Fig. 7. Delay faults on adder circuits placed outside the wasting area when the adversary at time 0 turns on 12,000 and 28,160 power wasters in Cyclone V and Arria 10 devices, respectively. X-coordinate denotes the time the fault occurred during the attack. Y-coordinate is the reported timing slack of the exercised path.



(a) DE1-SoC/Cyclone V: voltage drop measured at test pad VCC1P1.



(b) DE5a-Net/Arria 10: voltage drop measured at the positive terminal of on-board decoupling capacitor labeled as C371.

Fig. 8. Turning on power waster circuit causes a large instantaneous change in current. The instantaneous change causes a voltage drop on the off-chip inductor which effects all parts of the chip.

device). The Arria 10 results (Fig. 7b) however, show an additional peak of faults immediately following the enabling of the wasters. These faults are attributed to the initial response of the DE5a-Net/Arria 10 PDN to the sudden activation of the wasters that led to a large but brief voltage drop, also observed by Zick et al. [26] in a Xilinx Kintex-7 device.

As shown in our previous work [9], the voltage drop observed in Fig. 5 is responsible for many of the timing faults induced on the positive-slack paths. The simultaneous activation of all the power wasters causes a large and sudden change in the current drawn by the FPGA. The sudden change in current creates a voltage drop across the inline inductor of the switching regulator, which thereby reduces the voltage supplied to the chip (Eq. 2). Fig. 8 shows the core voltage dropping in DE1-SoC and DE5a-Net boards when the power wasters turn on, as captured by a Keysight MSOX4154A oscilloscope. In the Cyclone V device, the waveform in Fig. 8a shows that the peak voltage drop of 85mV occurs roughly 16 µs after the power wasters turn on. In the Arria 10 device (Fig. 8b), the peak voltage drop of 38mV occurs roughly 8 µs after the activation of the wasters. For each device, the timing of the minimum voltage as measured on the scope (Fig. 8) corresponds to the timing of the minimum voltage observed in on-chip sensors (Fig. 5), and the time at which the most severe delay faults occur (Fig. 7).

$$V_{core} = V_{reg} - V_L = V_{reg} - L\frac{di}{dt} \tag{2}$$

The 85mV and 38mV voltage drops measured across the inductors impact every part of the FPGA that shares the same supply, which can allow an attacker to affect victim circuits regardless of their position on the chip. Unlike the $L\frac{di}{dt}$ drop, the iR voltage drop due to resistances in the PDN depends only on the current, and not on the change in current. Therefore, $L\frac{di}{dt}$ drop is maximal when the current is changing, and iR drop is maximal after the current has changed, so they do not both contribute their peak values at the same time. The largest total voltage drop is observed to be a combination of $L\frac{di}{dt}$ drop from the inductor combined with a iR drop of the power grid.

To examine the spatial impact of the on-chip voltage drop, we placed ripple-carry adders in the Cyclone V device at distances 23, 26, 31, 35, 37, 40, 44, 47, and 52 LAB columns away from the center of the power waster region. Similarly, in the Arria 10 device, we instantiated adders at distances 38, 48, 60, 70, 76, 87, 97, 107, 118, 138, 148, and 160 columns from the region center. Fig. 9a shows that in the Cyclone V device the attack causes faults on legal paths with positive timing slack that are 40 LAB columns away from the center of the wasting area. The attack impact gradually diminishes with

increased distance from the waster (Fig. 5a). Adders placed farther away exhibit fewer faults.

Fig. 9b focuses on the first 1 µs of the attack in the Arria 10 device. Although the impact of the attack weakens at increasing distance from the wasters, faults in paths with positive slack are observed at all examined distances. Since faults were induced on legal paths at the device outskirts in both tested devices it is apparent that spatial isolation between tenants is insufficient to protect against PDN attacks in multi-tenant FPGA applications.

B. Relating Voltage and Timing Slack to Fault Sensitivity

Having demonstrated the capability to cause delay faults, and characterizing PDN voltage in response to power consumption, we now connect the two by using the Cyclone V device to show experimentally the combinations of slack and voltage that lead to faults. In this experiment, 1,024 random attack scenarios are created and implemented by choosing at random the following parameters:

- The position of the victim adder circuit (between 23 and 53 columns from center of attacker).
- The sensitized path of the victim adder (uses between 53 and 64 stages of carry logic implemented on the hardened carry circuitry of the FPGA).
- The number of power wasters used by the attacker (between 3,200 and 12,000 instances).

The minimum voltage at the victim circuit during each attack is inferred by interpolation on the data shown in Fig. 6 according to the victim location and number of power wasters in the attack. As in the prior subsection, the path is repeatedly sensitized during the attack and the result is checked for faults. Red and green marks in Fig. 10 denote attack scenarios in which faults did or did not occur, respectively. The X-coordinate of each point is the minimum voltage at the victim during the attack. The Y-coordinate of each point is the timing slack of the victim path as reported by the TimeQuest Timing Analyzer using the slow 1100mV 85 °C timing model.

Timing models are conservative with respect to operating conditions and process variation, and the effects of the conservative timing model can be seen in Fig. 10. Paths reported as having 0 slack are typically fault free even when their voltage drops by 140mV, although Fig. 3a shows that a 140mV drop should cause a significant increase in propagation delay.

The pattern of faulty and fault-free points in Fig. 10 shows that voltage and timing slack are largely sufficient to explain which adder paths will experience faults during an attack. This finding supports the supply voltage drop being the cause of the fault, and not some other artifact of power consumption. The results also show that conservative timing models provide some inherent margin against attack.

C. Relationship to FPGA Logic Isolation and Active Fencing

The results shown in Figure 9 indicate the ability of power wasters to induce faults over a wide extent of the FPGA die. The shared nature of the supply voltage PDN on the FPGA causes the voltage drop induced by the wasters to impact supply voltage across the chip, even though the waster and target logic components are logically isolated. Although leading FPGA companies, such as Xilinx and Intel, allow for the isolation of logic design subcomponents [41], [42], our results and those of other groups indicate that logic isolation is not effective in preventing these types of attacks.

Recently, Krautter et al. [43] proposed to use ROs to perform active fencing against side channel attacks on encryption cores. In these experiments, voltage sensors are used to identify small changes in on-chip voltage due to encryption operations. These perturbations are then used to identify the encryption key. To prevent such attacks, ROs located close to the encryption core are enabled, making it difficult to identify the small voltage changes induced by the encryption core. In our attack, the ROs used by active fencing would enhance the attack, rather than prevent it. The active fence would further reduce on-chip voltage to induce additional faults in the target circuit.

V. Exploiting Voltage Drops for Security Breaches

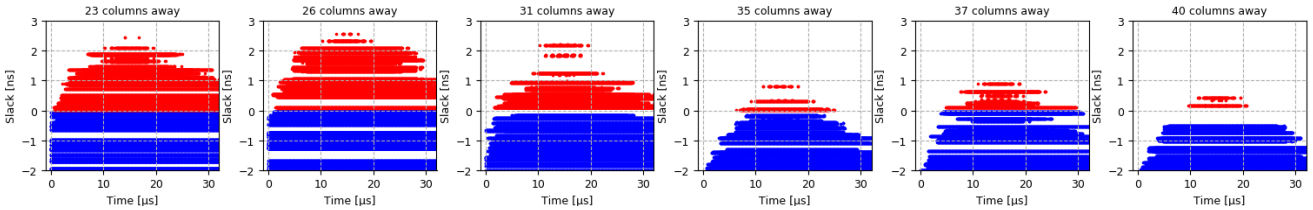
To demonstrate the risk of a malicious user manipulating an FPGA's core voltage, we conduct a fault injection attack on an FPGA-based RSA implementation. The attack is able to extract private RSA keys from the faulty outputs produced by the circuit. The following subsections give details of the RSA key recovery attack on a DE1-SoC board with power wasters implemented as discussed in Section III-B.

A. RSA Cryptosystem Background

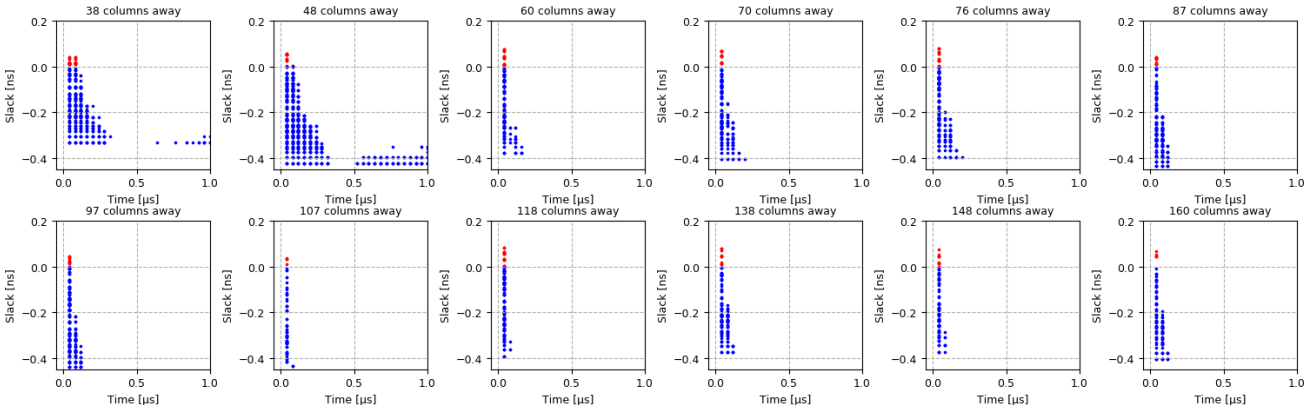
The RSA cryptosystem [44], based on an asymmetric cryptographic algorithm proposed in 1977, is still widely used for secure data transmission. The RSA encryption of a message involves the computation of the modular exponentiation $Y = X^e \ mod \ N$, where X is the message to be encrypted, e is the *public exponent*, N is the *RSA modulus*, and Y is the resulting ciphertext. Identically, the decryption function is described as $X = Y^d \ mod \ N$, where d is the *private exponent*. The public exponent e along with the modulus N compose the public key $k_{pub} = (N, e)$ which can be known by everyone and used for encrypting messages. The private exponent d is kept secret as the private key $k_{pr} = (N, d)$ and used for decryption. The RSA modulus N is used as the modulus for both the public and private keys and is computed as $N = p \cdot q$, where p and q are two large, randomly generated prime numbers. The primes p and q are usually 512 to 2,048 bits long and must be kept secret.

Performing modular exponentiation with large exponents can become impractical for conventional processors and hence modern systems often use dedicated hardware to accelerate the computationally expensive operations. One common technique to further speed up modular exponentiation of long numbers is based on the Chinese Remainder Theorem (CRT). The CRT can be applied for encrypting a message X as follows:

$$Y = X^e mod \ N = (a \cdot q) \cdot Y_p + (b \cdot p) \cdot Y_q \ mod \ N \quad (3)$$



(a) Cyclone V with victim 23-40 LAB columns away from the center of the attack.



(b) Arria 10 with victim 38-160 LAB columns away from the center of the attack.

Fig. 9. Examining timing faults at different distances between the adder and center of attack in Cyclone V and Arria 10 devices.

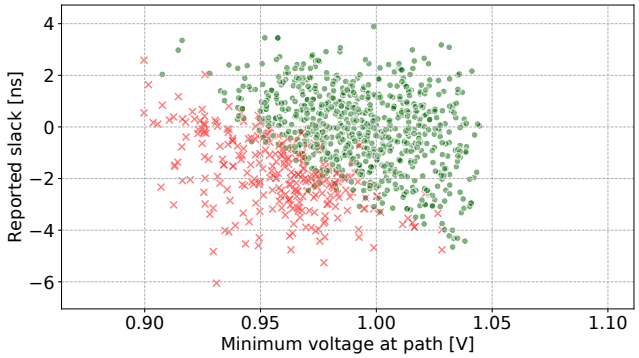


Fig. 10. Scatter plot shows which randomly generated attack scenarios caused faults and which did not. X-coordinate denotes voltage in victim circuit during attack. Y-coordinate is the reported timing slack of path exercised during attack.

where a, b are predefined constants and Y_p, Y_q are computed as:

$$\begin{aligned} Y_p &= (X \ mod \ p)^{(e \ mod \ p-1)} \ mod \ p \\ Y_q &= (X \ mod \ q)^{(e \ mod \ q-1)} \ mod \ q \end{aligned} \tag{4}$$

CRT avoids performing computations with the full-length exponents e, d, and modulus N, and instead performs two separate and faster modular exponentiations with numbers bounded by the "shorter" prime numbers p and q (see Eq. 4). In the last step of CRT, the two partial results Y_p and Y_q are combined (see Eq. 3) to construct the encrypted message Y. CRT exponentiation is shown to be four times faster than direct exponentiation [45] but can only be used by the party who possesses the private key k_{pr} and two prime numbers p and q.

B. Hardware Implementation

To investigate if the deliberately caused fluctuations of the FPGA's core voltage can reveal the RSA private key when the CRT is used, we implemented a parameterizable RSA core on the Cyclone V device. The CRT-based RSA core consists of a single modular exponentiation unit and control-path state machine for calculating sequentially Y_p and Y_q, described in Eq. 4. Modular exponentiation is realized using the standard square and multiply algorithm and a Montgomery multiplier for eliminating the requirement for the division operation [46]. Interfacing with the host PC is accomplished through a JTAG-accessible on-chip memory that controls the RSA core, writes in its inputs (e.g., p, q, d, X), and reads out its outputs (e.g., Y_p, Y_q).

Tab. II shows the resource utilization and maximum clock speed of the RSA core for three different key lengths. The critical path of the architecture resides in the control-path state machine implementing the Montgomery multiplier. The 128-bit implementation on average completes a single RSA operation (e.g., encryption or decryption) in $0.59\,\mathrm{ms}$ (1,695 ops/sec). Due to the combined use of the square-and-multiply method and Montgomery multiplication, doubling the length of the key quadruples the required clock cycles for a single RSA operation. In addition, the larger designs must be clocked at lower frequency. The 256- and 512-bit implementations complete on average a single RSA operation in $3\,\mathrm{ms}$ (333 ops/sec) and $14.68\,\mathrm{ms}$ (68 ops/sec), respectively.

C. Fault Injection Attack

Cloud FPGAs have steered attention to a new class of attacks that require neither physical device access nor expensive lab equipment. It has been shown that FPGA implementations

TABLE II
RESOURCES USED IN RSA CORE AND CORRESPONDING REPORTED FMAX FOR THE THREE SUPPORTED KEY LENGTHS IN THE CYCLONE V DEVICE.

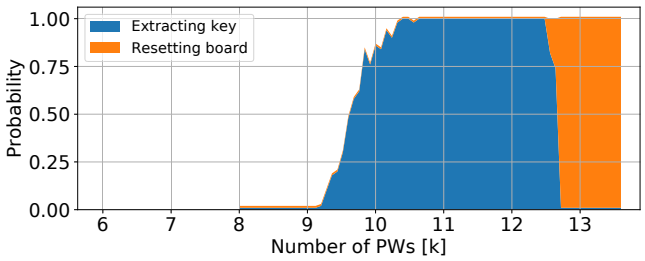
Key Length	ALMs (Avail.: 32k)	Flip-flops (Avail.: 128k)	Mem. (Avail.: 3,970 Kb)	Fmax [MHz]
128-bit	1,236 (3.9%)	1,925 (1.5%)	16 Kb (<1%)	94.74
256-bit	2,003 (6.2%)	3,463 (2.7%)	16 Kb (<1%)	77.12
512-bit	4,030 (12.6%)	6,537 (5.0%)	16 Kb (<1%)	61.50

of AES are susceptible to both power analysis attacks [29] using on-chip sensors, and fault injection [7] using power wasters. RSA implementations have been exposed to power analysis attacks [22] where the private key was successfully extracted using power traces meticulously captured by on-chip ring oscillators. In contrast, fault injection attacks, like the one described in this section, do not require calibrated sensors to detect information leakage.

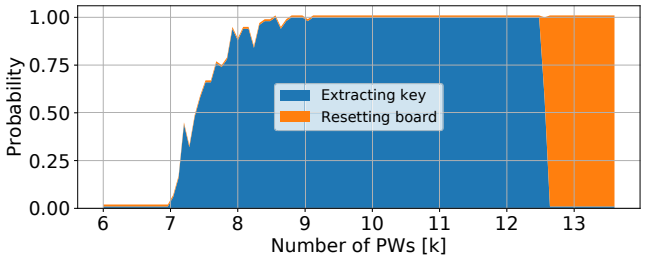
Boneh et al. [47] showed that arbitrary errors in computations of RSA with CRT make the factorization of the modulus N feasible. It can be shown that if an error occurs while computing one of the modular representations Y_p or Y_q (see Eq. 4) then the secret prime numbers p, q can be recovered. Lenstra [48] showed that if the RSA input is known then the prime numbers can be recovered using only faulty outputs. Assuming a faulty Y_p, Lenstra's approach provides q as $gcd(X - \widehat{Y}^e,\ N)$, where X is the RSA input, $\widehat{Y}$ is the faulty output, and e is the public exponent. Then the private exponent d can be derived as $d = e^{-1}\ mod\ ((q-1)\cdot(\frac{N}{q}-1))$. Similarly, a Y output composed using a faulty Y_q reveals the prime number p and the private exponent d can be then derived as $d = e^{-1}\ mod\ ((p-1)\cdot(\frac{N}{p}-1))$. Note that since $N = p \cdot q$ the attacker needs to know neither which prime number (p, q) has been exposed nor which of the partial results (Y_p, Y_q) is incorrect. As long as only one of the two partial results composing the final output Y is incorrect, a single interaction with the cryptosystem is sufficient for extracting the private key $k_{pr} = (N, d)$.

A potential scenario where this attack may apply involves a *Certification Authority* (CA) service that uses hardware to accelerate an RSA signing. Assume that an adversary who can induce faults in the hardware of the service requests a certificate and then sends message X to be signed. If the returned certificate Y is assembled with an erroneous Y_p or Y_q, then CA's private key $k_{pr} = (N, d)$ will be exposed. That is, the adversary is in possession of the already known CA's public key $k_{pub} = (N, e)$, initial message X, and a faulty output Y. By applying Lenstra's approach the adversary can extract one of the two primes and use it to reconstruct CA's private exponent d which then can be used for issuing fake certificates. A similar use-case scenario is discussed in Pellegrini et al. [49] where, however, the authors attack a SPARC-based RSA implementation by manipulating the supply voltage of the system in order to inject faults.

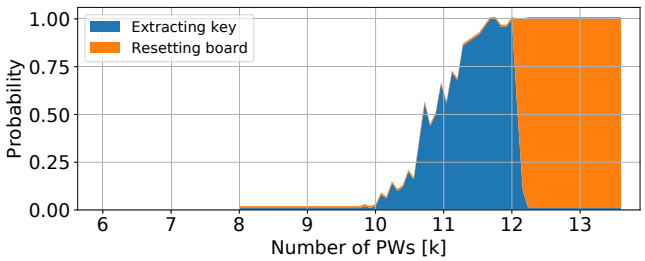
In the Cyclone V device, Lenstra's approach is put to the test by instantiating the RSA core described in Section V-B along with power wasters placed at random locations in the surrounding area. A script running on the host PC generates a set of RSA variables (e.g., message X and keys k_{pub}, k_{pr}), passes to the RSA core its inputs, triggers an RSA operation, and activates the wasters. When the RSA operation is over, the script reads the output of the RSA core and attempts to extract the private key k_{pr} using Lenstra's approach. A precompiled library containing bitstreams with various numbers of wasters is used to examine different attack magnitudes. Each bitstream undergoes 50 trials using randomly generated RSA inputs and a log is kept with the outcome of each attempt.



(a) 128-bit RSA core



(b) 256-bit RSA core



(c) 512-bit RSA core

Fig. 11. The stack plot shows the probability of different outcomes when attacking RSA using various numbers of wasters instantiated in the Cyclone V device. Successfully extracting the RSA private key constitutes the blue part of the plot. Unwantedly resetting the board due to the attack constitutes the orange part of the plot.

The activation of the wasters during the RSA operation can result in three outcomes: (1) the attack has no impact on the RSA core and thus it outputs the expected Y, (2) the attack induces timing faults resulting in a faulty output Y which reveals the private key k_{pr}, and (3) the voltage drop due to the attack triggers an undesirable board reset and loss of the FPGA configuration image. The probability of these three outcomes are summarized in Fig. 11. The X-axis denotes the number of wasters that are activated during the RSA operation, blue corresponds to the probability of successfully extracting the private key k_{pr}, and orange is the probability of resetting the board. Although in theory, the attack should work for any key length, we examined the three key lengths (128-, 256-,

and 512-bit) discussed in Section V-B. The datapath for each key length is found to be susceptible to fault injection. The probability of successfully extracting the key is maximized when activating roughly 11k-12k wasters, and beyond this number of wasters the board typically resets during an attack.

VI. Monitoring System for PDN Attacks

PDN attacks require power consumption, transiently or in steady-state, beyond what the power distribution network can handle. Our results have shown that the power consumption of one adversarial block can cause a measurable and significant difference in the voltage of other blocks. Circuits closest to the power consumption experience the largest voltage drop, and the voltage drop becomes smaller moving farther away (Fig. 6). The voltage gradients effectively provide a map pointing toward the center of the attack, which will have the lowest voltage. A spatially distributed network of voltage sensors can enable a resource manager to monitor voltage gradients and identify the source of any attacks that occur. The resource manager can then prevent further instances of the same attack by taking the offending application offline, or banning it from co-tenant settings.

A. Monitor Network

A network of 46 and 132 sensors in Cyclone V and Arria 10 devices, respectively, monitor voltage fluctuations and log the data. The area cost of the monitor network is given in Tab. III. In both devices, each sensor uses 20 ALMs and 20 flip-flops. In Cyclone V, the 46 sensors collectively consume 2.9% of the ALMs whereas the 132 sensors implemented on the Arria 10 fabric consume less than 1% of its ALMs. Both implementations use less than 1% of the flip-flops on the chip. Note that Tab. III shows the resources required for the controller logic that logs the sensor data to memory only for the full sensor networks of 46 and 132 sensors in Cyclone V and Arria 10 devices, respectively. A controller synthesized for a network composed of fewer sensors would consume fewer resources as well.

Fig. 12 shows the voltage contours of the two devices based on sensor data during two different power attacks. The specific data used to generate the plot is the minimum value observed by each sensor in the $500\,\mu s$ time period that contained the attack. A cubic interpolation algorithm reconstructs the smoothed voltage contours from the samples collected at the discrete sensor locations.

The two power attacks on each chip vary in the magnitude of the PDN disturbance and location of the attacker on the chip. The details of each attack are shown in Tab. IV. In the Cyclone device, as denoted on the voltage contour lines in Fig. 12a, when the attacker turns on 12,000 power wasters the voltage at the center of the attack drops below 825mV, and the voltage at the farthest corner of the FPGA drops to 975mV. In the weaker attack (Fig. 12b), the enabling of 3,200 wasters drops the voltage below 975mV near the attack and the voltage at the farthest corner of the FPGA remains above 1.050V.

TABLE III
Resources used in voltage monitoring network for various numbers of sensors.

Num. RO sensors	ALMs (Avail.: 32,070)	Flip-flops (Avail.: 128,280)
10	200 (<1%)	200 (<1%)
20	400 (1.2%)	400 (<1%)
30	600 (1.9%)	600 (<1%)
40	800 (2.5%)	800 (<1%)
46	920 (2.9%)	920 (<1%)
Controller	430 (1.3%)	111 (<1%)

(a) Intel Cyclone V SE (5CSEMA5F31C6) FPGA.

Num. RO sensors	ALMs (Avail.: 427,200)	Flip-flops (Avail.: 1,708,800)
64	1,280 (<1%)	1,280 (<1%)
132	2,640 (<1%)	2,640 (<1%)
Controller	1,008 (<1%)	134 (<1%)

(b) Intel Arria 10 GX (10AX115N2F45E1SG) FPGA.

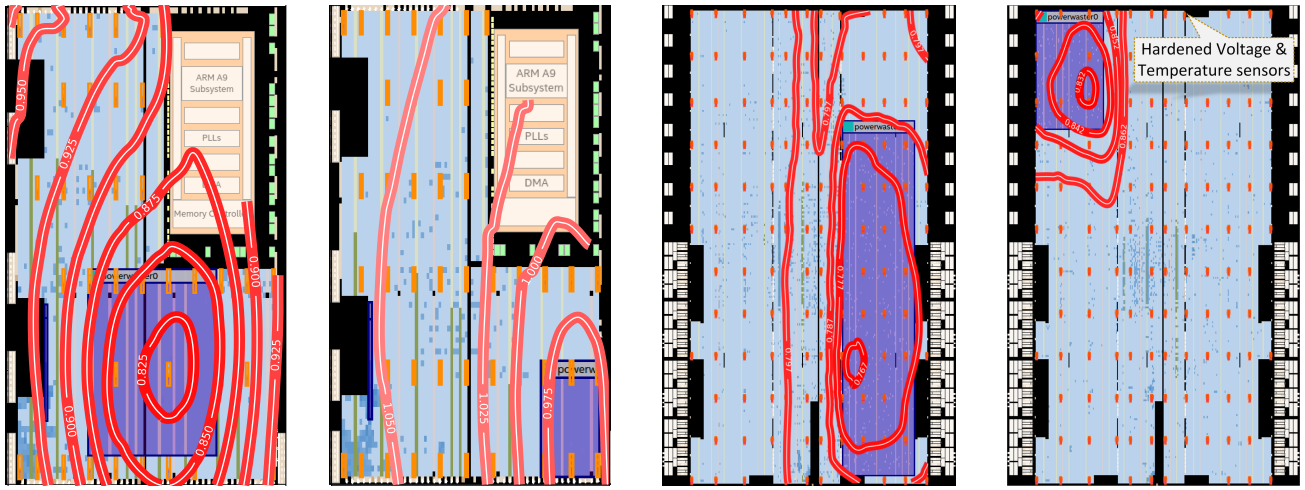
TABLE IV
Attack scenarios used for evaluating the monitor network.

Device	Allocated Area [rows by cols LABs]	Number of PW instances	Type of Attack
Cyclone V	32 x 44 (1,408)	12,000	Strong
	20 x 20 (400)	3,200	Weak
Arria 10	168 x 68 (11,424)	28,160	Strong
	56 x 64 (3,584)	8,000	Weak

In the Arria 10 device, the 28,160 power wasters drop the voltage to 767mV at the center of the stronger attack (Fig. 12c) while the voltage in more than half of the FPGA fabric is 100mV below the nominal 0.9V. In the weaker attack (Fig. 12d), the 8,000 wasters have a lower impact dropping the voltage below 862mV only in the vicinity of the attacker.

B. Attack Attribution

A goal for the monitoring network is to determine the source of any attacks that occur. In the case of PDN attacks, the attacker cannot easily mask their identity, because of the spatial extent of the voltage drops that they cause. Here we evaluate, as a cost tradeoff, the number of sensors required to find an attacker based on voltage contours. For each attack scenario (Tab. IV) examined in the previous subsection, we consider how precisely the attacker can be located using 10, 20, 30, or 40 of the 46 sensors in the Cyclone V device (Tab. III) and 32, 64, 96, or 128 of the 132 sensors in the Arria 10 device, which would reduce the cost of the monitoring network. For each number of sensors, we randomly choose 100 different subsets containing that number of sensors, and from each subset try to predict the location of the attacker. Fig. 13 shows the results of this analysis. The dots on each plot are the 100 different predictions of the attacker location. As one might expect, the chance of successfully locating the attack increases with the number of sensors. In the Cyclone V device (Fig. 13a and Fig. 13b), predictions based on 20 or more



(a) 12,000 power waster attack on Cyclone V.
(b) 3,200 power waster attack on Cyclone V.
(c) 28,160 power waster attack on Arria 10.
(d) 8,000 power waster attack on Arria 10.

Fig. 12. Map of voltage contours on chip during power attacks, reconstructed from sensor data. Purple rectangle denotes location of the attacker's power waster circuits. Orange rectangles are the sensors.

sensors converge to a location within the attacking circuit. Similarly, in the Arria 10 device (Fig. 13c and Fig. 13d), using 64 or more sensors causes predictions to converge to a location within the attack area. These results show that a network of monitors can locate the attacker with less than 46 sensors in the Cyclone V device and with less than 132 sensors in the Arria 10 device. The overall low hardware overhead of the monitoring system should not interfere with the design of other circuits.

VII. Conclusion and Future Work

The increased logic capacity and performance of FPGAs have made them attractive implementation options for a wide range of digital circuits. As the application domain of FPGAs has grown to include cloud computing and a broad range of embedded systems, scenarios have emerged in which circuits from multiple designers are deployed on an FPGA at the same time. As shown in this manuscript, FPGA multi-tenancy in contemporary FPGA architectures can lead to security risks due to a shared power distribution network (PDN). Our work shows that power wasters in one portion of the FPGA can lead to faults in distant FPGA locations, even for circuit paths with significant slack. These effects are carefully characterized for the Intel Cyclone V and Arria 10 FPGAs located on the Terasic DE1-SoC and DE5a-Net boards, respectively. Specifically, we characterize the magnitude of the disturbance as a function of time, power consumed by attacker, and position of the victim relative to the attacker.

To show the breadth of the threat, we perform a power-based fault injection attack on an RSA cryptosystem that reveals the private key without synchronizing the attack with specific encryption rounds nor embedding Trojans within the core. This attack is performed remotely and does not require physical access to the device. For mitigation, we propose the use of a network of small voltage sensors that collect voltage information and pass it to a central controller. We demonstrate that the source of a voltage-altering attack can be easily identified by a small number of sensors consuming less than 3% of FPGA logic.

In future work, the effects of using different types of power wasters and their fault inducing capabilities could be explored. Although RO-based circuits are not currently allowed in cloud FPGA platforms such as Amazon EC2 F1 [3], other power wasters that are allowed [33], [50] have been recently reported. These wasters could be evaluated in the context of current safeguards provided by FPGA cloud vendors such as external power monitoring [51]. Experiments using FPGAs from other vendors could also be performed.

We hope that the results of our experiments and those other researchers can be used to architect FPGAs that will make multi-tenancy safer.

Acknowledgment

This research was funded by NSF/SRC grant CNS-1619558 and a grant from Intel's Corporate Research Council.

References

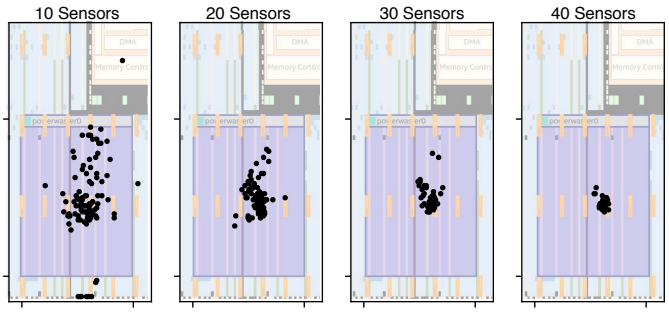
[1] X. Zhang, X. Shao, G. Provelengios, N. K. Dumpala, L. Gao, and R. Tessier, "Scalable network function virtualization for heterogeneous middleboxes," in *IEEE International Symposium on Field Programmable Custom Computing Machines (FCCM)*, 2017, pp. 219–226.

[2] A. M. Caulfield, E. S. Chung, A. Putnam, H. Angepat, J. Fowers, M. Haselman, S. Heil, M. Humphrey, P. Kaur, J.-Y. Kim, D. Lo, T. Massengill, K. Ovtcharov, M. Papamichael, L. Woods, S. Lanka, D. Chiou, and D. Burger, "A cloud-scale acceleration architecture," in *IEEE/ACM International Symposium on Microarchitecture (MICRO)*, 2016, pp. 1–13.

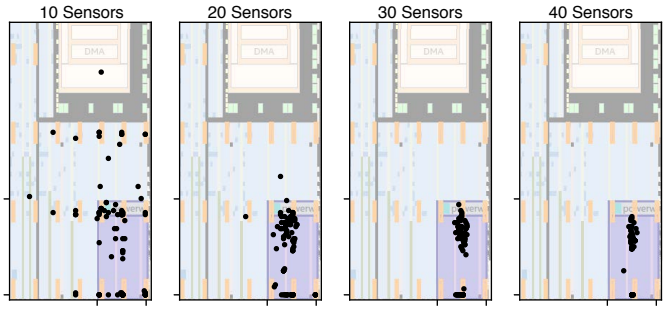
[3] "Amazon EC2 F1 instances," https://aws.amazon.com/ec2/instance-types/f1/.

[4] R. Maes, D. Schellekens, and I. Verbauwhede, "A pay-per-use licensing scheme for hardware IP cores in recent SRAM-based FPGAs," *IEEE Transactions on Information Forensics and Security (TIFS)*, vol. 7, no. 1, pp. 98–108, 2011.

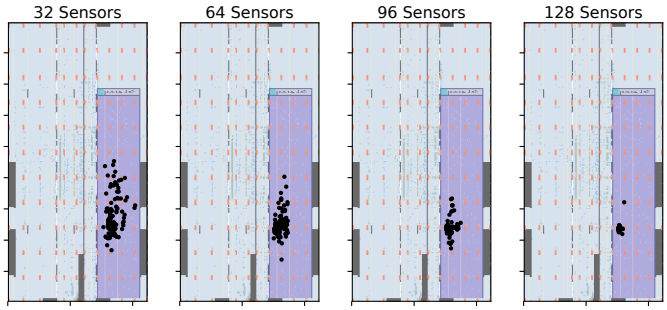
[5] "Deep dive into Alibaba Cloud F3 FPGA as a service instances," https://www.alibabacloud.com/blog/deep-dive-into-alibaba-cloud-f3-fpga-as-a-service-instances_594057.



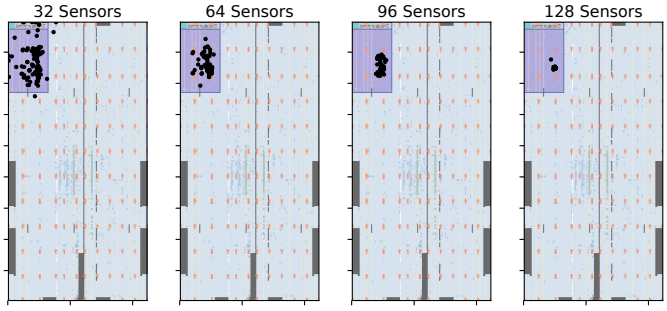
(a) Cyclone V: locating attack that uses 12,000 power wasters.



(b) Cyclone V: locating attack that uses 3,200 power wasters.



(c) Arria 10: locating attack that uses 28,160 power wasters.



(d) Arria 10: locating attack that uses 8,000 power wasters.

Fig. 13. Marks represent predicted center of attack based on a randomly selected subset of sensors. Each subplot contains 100 points.

[6] J. M. Mbongue, A. Shuping, P. Bhowmik, and C. Bobda, "Architecture support for FPGA multi-tenancy in the cloud," in *IEEE International Conference on Application-Specific Systems, Architectures and Processors (ASAP)*, 2020, pp. 125–132.

[7] J. Krautter, D. R. Gnad, and M. B. Tahoori, "FPGAhammer: Remote voltage fault attacks on shared FPGAs, suitable for DFA on AES," *IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)*, vol. 2018, no. 3, pp. 44–68, 2018.

[8] D. Mahmoud and M. Stojilović, "Timing violation induced faults in multi-tenant FPGAs," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2019, pp. 1745–1750.

[9] G. Provelengios, D. Holcomb, and R. Tessier, "Characterizing power distribution attacks in multi-user FPGA environments," in *International Conference on Field Programmable Logic and Applications (FPL)*, 2019, pp. 194–201.

[10] M. M. Alam, S. Tajik, F. Ganji, M. Tehranipoor, and D. Forte, "RAM-Jam: Remote temperature and voltage fault attack on FPGAs using memory collisions," in *Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC)*, 2019, pp. 48–55.

[11] D. R. Gnad, F. Oboril, and M. B. Tahoori, "Voltage drop-based fault attacks on FPGAs using valid bitstreams," in *International Conference on Field Programmable Logic and Applications (FPL)*, 2017, pp. 1–7.

[12] C. Chow, L. Tsui, P. Leong, W. Luk, and S. Wilton, "Dynamic voltage scaling for commercial FPGAs," in *IEEE International Conference on Field Programmable Technology (FPT)*, 2005, pp. 173–180.

[13] I. Ahmed, S. Zhao, O. Trescases, and V. Betz, "Measure twice and cut once: Robust dynamic voltage scaling for FPGAs," in *International Conference on Field Programmable Logic and Applications (FPL)*, 2016, pp. 1–11.

[14] C. Jin, V. Gohil, R. Karri, and J. Rajendran, "Security of cloud FPGAs: A survey," *arxiv*, vol. arXiv:2005.04867, 2020. [Online]. Available: http://arxiv.org/abs/2005.04867

[15] *DE5a-Net DDR4 User Manual*, Terasic Technologies, Aug. 2018.

[16] *DE1-SoC User Manual*, Terasic Technologies, Feb. 2014.

[17] S. Mal-Sarkar, A. Krishna, A. Ghosh, and S. Bhunia, "Hardware Trojan attacks in FPGA devices: Threat analysis and effective countermeasures," in *ACM Great Lakes Symposium on VLSI (GLSVLSI)*, 2014, pp. 287–292.

[18] A. Khawaja, J. Landgraf, R. Prakash, M. Wei, E. Schkufza, and C. J. Rossbach, "Sharing, protection, and compatibility for reconfigurable fabric with AmorphOS," in *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2018, pp. 107–127.

[19] O. Knodel, P. Lehmann, and R. G. Spallek, "RC3E: Reconfigurable accelerators in data centres and their provision by adapted service models," in *IEEE International Conference on Cloud Computing (CLOUD)*, 2016, pp. 19–26.

[20] K. M. Zick and J. P. Hayes, "On-line sensing for healthier FPGA systems," in *ACM/SIGDA International Symposium on Field Programmable Gate Arrays (FPGA)*, 2010, pp. 239–248.

[21] ——, "Low-cost sensing with ring oscillator arrays for healthier reconfigurable systems," *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, vol. 5, no. 1, pp. 1:1–1:26, 2012.

[22] M. Zhao and G. E. Suh, "FPGA-based remote power side-channel attacks," in *IEEE Symposium on Security and Privacy (S&P)*, 2018, pp. 229–244.

[23] *Intel Arria 10 Core Fabric and General Purpoe I/Os Handbook*, Intel Corporation, May 2018.

[24] M. Barbareschi, G. Di Natale, and L. Torres, "Implementation and analysis of ring oscillator circuits on Xilinx FPGAs," in *Hardware Security and Trust*. Springer, 2017, ch. 12, pp. 237–251.

[25] D. R. Gnad, F. Oboril, S. Kiamehr, and M. B. Tahoori, "Analysis of transient voltage fluctuations in FPGAs," in *International Conference on Field-Programmable Technology (FPT)*, 2016, pp. 12–19.

[26] K. M. Zick, M. Srivastav, W. Zhang, and M. French, "Sensing nanosecond-scale voltage attacks and natural transients in FPGAs," in *ACM/SIGDA International Symposium on Field Programmable Gate Arrays (FPGA)*, 2013, pp. 101–104.

[27] M. Ueno, M. Hashimoto, and T. Onoye, "Real-time supply voltage sensor for detecting/debugging electrical timing failures," in *IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum (IPDPS)*, 2013, pp. 301–305.

[28] D. R. E. Gnad, F. Oboril, S. Kiamehr, and M. B. Tahoori, "An experimental evaluation and analysis of transient voltage fluctuations in FPGAs," *IEEE Transactions on Very Large Scale Integration Systems (TVLSI)*, vol. 26, no. 10, pp. 1817–1830, 2019.

[29] F. Schellenberg, D. R. Gnad, A. Moradi, and M. B. Tahoori, "An inside job: Remote power analysis attacks on FPGAs," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2018, pp. 1111–1116.

[30] O. Glamocanin, L. Coulon, F. Regazzoni, and M. Stojilović, "Are cloud FPGAs really vulnerable to power analysis attacks?" in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2020, pp. 1007–1010.

[31] D. G. Mahmoud, W. Hu, and M. Stojilović, "X-Attack: Remote activation of satisfiability don't-care hardware Trojans on shared FPGAs," in *International Conference on Field Programmable Logic and Applications (FPL)*, 2020, pp. 1–8.

[32] L. L. Shen, I. Ahmed, and V. Betz, "Fast voltage transients on FPGAs: Impact and mitigation strategies," in *IEEE International Symposium on Field-Programmable Custom Computing Machines (FCCM)*, 2019, pp. 271–279.
[33] J. Krautter, D. R. Gnad, and M. B. Tahoori, "Mitigating electrical-level attacks towards secure multi-tenant FPGAs in the cloud," *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, vol. 12, no. 3, pp. 12:1–12:26, 2019.
[34] D. Klokotov, J. Shi, and Y. Wang, "Distributed modeling and characterization of on chip/system level PDN and jitter impact," in *DesignCon*, 2014, pp. 1–22.
[35] S. Zhao, I. Ahmed, V. Betz, A. Lotfi, and O. Trescases, "Frequency-domain power delivery network self-characterization in FPGAs for improved system reliability," *IEEE Transactions on Industrial Electronics (TIE)*, vol. 65, no. 11, pp. 8915–8924, 2019.
[36] T. Takahashi, T. Uezono, M. Shintani, K. Masu, and T. Sato, "On-die parameter extraction from path-delay measurements," in *IEEE Asian Solid-State Circuits Conference (ASSCC)*, 2009, pp. 101–104.
[37] S. Xie and W. T. Ng, "Delay-line temperature sensors and VLSI thermal management demonstrated on a 60nm FPGA," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2014, pp. 2571–2574.
[38] A. Amouri, J. Hepp, and M. Tahoori, "Built-in self-heating thermal testing of FPGAs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 35, no. 9, pp. 1546–1556, 2015.
[39] *INA231 High- or Low-Side Measurement, Bidirectional Current and Power Monitor With 1.8-V I2C Interface*, Texas Instruments, Mar. 2018.
[40] *TimeQuest Timing Analyzer Quick Start Tutorial*, Intel Corporation, Dec. 2009.
[41] *Vivado Isolation Verifier*, Xilinx Corporation, Aug. 2018.
[42] *Enabling Design Separation for High-Reliability and Information-Assurance Systems*, Altera Corporation, Jun. 2009.
[43] J. Krautter, D. R. Gnad, F. Schellenberg, A. Moradi, and M. B. Tahoori, "Active fences against voltage-based side channels in multi-tenant FPGAs," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2019, pp. 1–8.
[44] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.
[45] C. Paar and J. Pelzl, "The RSA cryptosystem," in *Understanding cryptography: a textbook for students and practitioners*. Springer Science & Business Media, 2009, ch. 7, pp. 173–204.
[46] J. Fry and M. Langhammer, "RSA & public key cryptography in FPGAs," *Altera document*, pp. 1–8, 2005.
[47] D. Boneh, R. A. DeMillo, and R. J. Lipton, "On the importance of eliminating errors in cryptographic computations," *Journal of cryptology*, vol. 14, no. 2, pp. 101–119, 2001.
[48] A. K. Lenstra, "Memo on RSA signature generation in the presence of faults," 1996.
[49] A. Pellegrini, V. Bertacco, and T. Austin, "Fault-based attack of RSA authentication," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2010, pp. 855–860.
[50] G. Provelengios, D. Holcomb, and R. Tessier, "Power wasting circuits for cloud FPGA attacks," in *International Conference on Field Programmable Logic and Applications (FPL)*, 2020, pp. 231–235.
[51] "AWS general F1 FAQs," https://github.com/aws/aws-fpga/blob/master/FAQs.md.

George Provelengios received the B.S. degree in informatics engineering from the Technological Educational Institute of Western Greece, in 2011, and the M.S. degree in microelectronics from the National and Kapodistrian University of Athens, Greece, in 2015. He is working towards the Ph.D. degree in the Electrical and Computer Engineering Department at the University of Massachusetts Amherst, Amherst, MA, USA. His research interests include cloud FPGA security and embedded systems security.

Daniel Holcomb (M07) received the B.S. and M.S. degrees in electrical and computer engineering from the University of Massachusetts Amherst, Amherst, MA, USA, and the Ph.D. degree in electrical engineering and computer sciences from the University of California at Berkeley, Berkeley, CA, USA. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, University of Massachusetts Amherst. His research interests are in hardware security and embedded systems.

Russell Tessier (M00-SM07) received the B.S. degree in computer and systems engineering from Rensselaer Polytechnic Institute, Troy, NY, USA, in 1989, and the S.M. and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1992 and 1999, respectively. He is currently Professor of Electrical and Computer Engineering with the University of Massachusetts, Amherst, MA. His current research interests include computer architecture and FPGAs.