Physical Side-Channel Attacks and Covert Communication on FPGAs: A Survey

Seyedeh Sharareh Mirzargar and Mirjana Stojilović
School of Computer and Communication Sciences
École Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Switzerland
seyedeh.mirzargar, mirjana.stojilovic@epfl.ch

***Abstract*—Field-programmable gate arrays (FPGAs) are, like CPUs, susceptible to side-channel information leakage and covert communication. The malleability of FPGAs enables users to create and control physical effects, and sense and measure the consequences. With FPGAs becoming integrated into the cloud, a range of hardware- and software-based attacks may be waiting to be discovered. In this survey, we focus on physical channels used for side-channel attacks or covert communication. Physical channels are those that exist due to the physical properties of FPGAs, for example: power consumption, temperature, or electromagnetic emission. We include the most recent demonstrations of malicious or unintended use of physical channels in remote and/or shared FPGAs, propose taxonomies, compare the efficiency and feasibility of the attacks, and discuss challenges in preventing them.**

***Index Terms*—covert communication, crosstalk, electromagnetism, FPGA, power, side-channel attacks, temperature**

I. Introduction

FPGAs, with their flexible computing fabric, offer lower design costs, reduced system complexity, and decreased time to market, while achieving performance gains due to abundant hardware parallelism. Given the large number of bit- and byte-level operations required in modern block ciphers, FPGAs are a natural platform for implementing cryptographic algorithms. The growth in application space of FPGAs puts a lot of pressure on FPGA- and system developers to ensure security and protect both the development investment and the end users.

Design-tool subversion, (un)trusted foundries, tampering, and bitstream reverse engineering are only some of the known security threats associated with reconfigurable hardware [1]–[5]. In this paper, we focus on those security threats that do not require injecting a fault or tampering with the design (e.g., by inserting a Trojan) to retrieve a secret information: side-channel analysis (SCA). In SCA, while an FPGA is performing cryptographic computation, an adversary exploits external, measurable, and benign manifestations of internal processes of the FPGA with the goal of inferring secrets. Side channel attacks first appear in Kocher et al. [6] as **timing attacks**, in which an adversary measures the time a device takes to perform the computations and deduces additional information about the crypto-system. Another example of SCA is the differential **power analysis attack** [7], where an adversary measures and analyzes the device power consumption to deduce the secret key. Yet another side channel is the one that measures and exploits the electromagnetic (EM) emanations from a device: **EM analysis attack**. Besides attacking, these and other side channels can be used to communicate, i.e., to intentionally leak secret information to someone who is eavesdropping the channel properties. This **covert communication** demands for a team: a source and a destination. Normally, the source is prevented from writing to destination directly; instead, it uses indirect means to leak classified data. For example, a covert communication channel could be a shared memory, such as DRAM or cache memory.

Since big datacenter and cloud providers decided to add FPGAs to their portfolio, researchers have been actively looking into the security threats that entails and how to best implement FPGA-accelerated clouds. In the past couple of years, several side-channel threats have been discovered. Given that all of them are **physical**, i.e., they rely on sensing a physical phenomena (power, current, electromagnetic emanations, crosstalk coupling, heat), it is timely to revisit the related work on physical side-channels.

There are several ways to categorize physical side-channel attacks. First, they can be classed based on the transmission medium or physical phenomenon being observed as **power**, **electromagnetic**, **thermal**, **sound**, **crosstalk coupling**, and **photonic emission**. Additionally, they can be classed as **active** (invasive) or **passive** (non invasive). Active attacks include tampering with the device to increase side-channel leakage or to monitor its internal signals, while passive attacks only observe internal information of the device with a measuring instrument or a sensor. An entirely new way of categorizing SCAs in the datacenter could be into those requiring physical **proximity** to the device and those that can be performed **remotely**.

The contributions of this survey, besides revisiting physical side-channel attacks and covert-communication successfully demonstrated on FPGAs, are as follows:

- a survey of the recently shown physical side- and covert-communication channels in shared and/or remote FPGAs,
- a comprehensive list of FPGA devices and platforms that have been successfully attacked or used to perform covert communication, and
- a discussion, based on qualitative and quantitative data, about the threat that the FPGA physical channels pose.

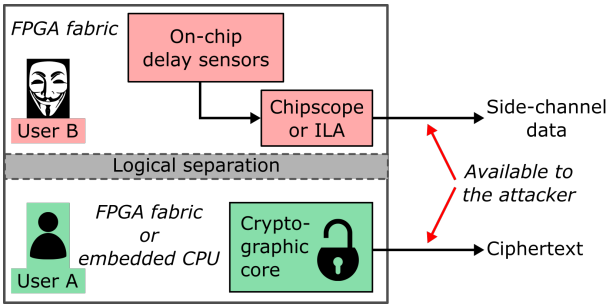

Fig. 1. An illustration of the remote power analysis attack. The assumption is that two users—the victim (in green) and the adversary (in red)—are using the same FPGA at the same time. While a cryptographic core in the victim's space is encrypting a number of plaintexts, the adversary collects the ciphertexts and the power-consumption traces recorded on the chip, to apply off-chip differential or correlation power analysis attack. Even with logical/physical separation between the two users, these attacks have been shown to succeed.

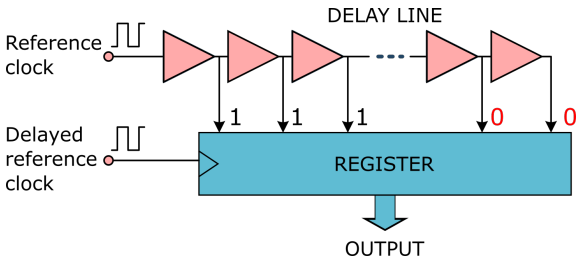

Fig. 2. Simplified architecture of a delay-line sensor. A reference clock signal is connected to a delay line, composed of a uniform chain of buffers. A delayed clock, of the same frequency, is used to sample the propagation delay of the reference clock. Hence, one value is recorded every clock cycle. Since the buffer delay depends on the supply voltage, the output of the register reflects local voltage fluctuations.

The rest of the paper is structured as follows. In Sections II–V we discuss power, crosstalk, electromagnetic, and thermal side channels, respectively. Section VI compares the threats and addresses prevention and protection mechanisms, while Section VII presents concluding remarks.

II. Power Analysis

Since their discovery in late nineties [7], power analysis attacks have attracted significant attention within the cryptographic community. They have been used to efficiently break a wide variety of crypto devices: smart cards [8], ASICs [9], and even FPGAs [10]–[15]. First experimental results on FPGAs were published by Örs et al. in 2003 [10].

A. Power Analysis Attacks

Traditional power analysis attacks assume having **physical access** to the victim to measure power consumption (at a power supply pin or over a shunt resistor, with an oscilloscope). However, FPGAs can be programmed to allow power analysis attacks to be executed **remotely**, i.e., without physical proximity. In the rest of this section, our focus is on remote attacks, because they have been discovered only recently. Figure 1 illustrates a typical attack scenario.

1) Fabric-to-Fabric Power Analysis Attack: Schellenberg et al. [16] and Zhao et al. [17] in two contemporary works demonstrated remotely-controlled power analysis attacks.

To measure voltage without an oscilloscope, Schellenberg et al. [16] implemented in FPGA fabric a sensor that indirectly measures core-voltage variations. This delay-line based sensor, first published by Zick et al. [18], was also used by Gnad et al. to analyze transient voltage fluctuations in FPGAs [19]. In its simplest configuration, illustrated in Figure 2, the sensor is a sequence of buffers (a delay line), where the output of every buffer is connected to a flip-flop. The input of the first buffer in the line is driven by a reference clock signal, while the register is clocked with a signal of the same frequency but delayed with respect to the reference clock. This allows the reference clock to propagate through the delay line for a part of the period, before the buffer outputs get sampled by the registers. As a result, the information on the signal's propagation depth is recorded once every clock period.

How is this related to voltage fluctuations? In typical CMOS circuits, combinational logic delay d can be modeled as inversely proportional to voltage V supplying each gate [20]:

$$d \propto \frac{1}{V}.$$

Hence, a change in the buffer delays, observed through the change in register values, indirectly exposes the core voltage behavior, which reflects the switching activity and the chip power consumption.

Only 5k AES-128 encryptions were sufficient for Schellenberg et. al [16] to retrieve the secret key using a standard correlation power analysis attack.

2) Fabric-to-CPU Power Analysis Attack: In systems-on-chip (SoCs), it is common for FPGAs and CPUs to share the same power supply rails [21]. Therefore, switching activities inside the CPU may cause a drop in voltage supply of the FPGA. Zhao et al. [17] tested and demonstrated two successful attacks: first, on an RSA cryptomodule implemented in FPGA and, second, on the embedded ARM of Xilinx Zynq SoC while it was performing RSA encryption.

To measure voltage variations, they used a slightly different sensor than Schellenberg et al. [16]. The main component of their sensor is a ring oscillator (RO) composed of a single inverter closed in a loop. In a more general case, an RO can be assembled using buffers, flip-flops, and an odd number of inverters, provided they are all closed in a loop (Figure 3).

The oscillation frequency f_{RO} of the RO is inversely proportional to the time a signal takes to complete the loop. Since voltage fluctuations influence the delay of inverters and buffers, they affect the oscillation frequency as well. To measure f_{RO}, the authors used a frequency counter to count the number of oscillations C_{RO} during a fixed period of time. In parallel, they used a reference counter, clocked at a reference frequency f_{REF}, to count the number of elapsed reference clock periods. When the reference counter would reach a predetermined value C_{REF}, frequency counter would be disabled and read, and the frequency computed as follows:

$$f_{\mathrm{RO}} \approx C_{\mathrm{RO}} \frac{f_{\mathrm{REF}}}{C_{\mathrm{REF}}}.$$

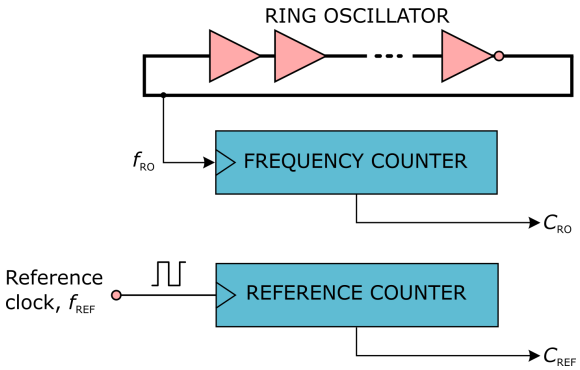

Fig. 3. Simplified architecture of a sensor based on a ring oscillator. The frequency counter keeps track of the number of RO oscillations. The reference counter measures elapsed time. When a fixed-number of reference clock periods expires, frequency counter is disabled and both counters are read to compute frequency f_{RO}, which reflects voltage variations.

Additionally, the authors instantiated a network of twenty sensors, distributed throughout the FPGA fabric, and combined their readings to reduce the result dependence on spatial proximity to switching logic.

3) FPGA-to-FPGA Power Analysis Attack: Schellenberg et al. [22] went a step further to show that an FPGA with on-chip sensors for voltage variations may be used for power analysis attacks on cryptographic modules running on another FPGA, as long as the two chips share the same power-delivery network and sit on the same printed circuit board (PCB).

As the experimental platform, they used Sakura-G, which contains two independently-programmable FPGAs on a single PCB. Since the two FPGAs were powered from two different core voltages, they disconnected one FPGA from its power supply and connected it to the power supply of the other FPGA. For a successful differential power analysis attack on a 128-bit AES, they reported that at least 40× more traces were required than when the sensor and the victim shared the same FPGA, and this under the condition that all small decoupling capacitors are removed from the board. With all decoupling capacitors present on the PCB, the AES key could still be guessed, but as many as 2.5 million traces were required.

B. Power Supply as Covert Communication Channel

Communication over power delivery network (PDN) is widely used in overhead power lines. However, communication over power lines on PCBs or inside integrated circuits is very uncommon. Ziener et al. demonstrated that the PDN of a PCB can be used to send information from the FPGA [23]. As experimental platforms, they used two boards: one with a Xilinx Spartan-3 and one with a Xilinx Virtex-II FPGA.

To send a logical 1, they would let a large shift register, initialized with a sequence of alternate ones and zeros, operate for a number of cycles; this would cause increased power consumption. To transmit a logical 0, no shifting would be performed. In order to successfully decode the transmitted data from the FPGA power supply voltage, one needs the FPGA impulse response. For measuring it, the authors again

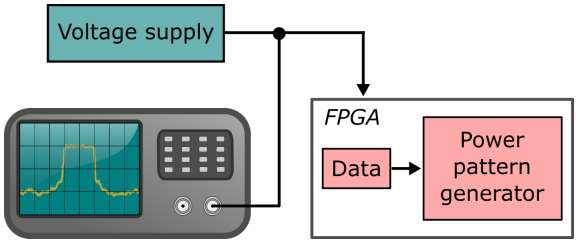

Fig. 4. Covert communication using FPGA power supply. Data being sent is used as a binary enable signal for a power pattern generator. This pattern generator can be, for example, a large shift register, which while enabled shifts a sequence of alternate zeros and ones, thus increasing current consumption and modulating the shape of the power supply voltage. The oscilloscope records the power supply voltage trace which, together with a known system's impulse response, can then be used to decode the transmitted data.

used a large shift register, enabled for exactly one clock cycle, and recorded the system's response at a power-supply voltage via underneath the FPGA. After correlating the FPGA power supply voltage during the data transmission with the acquired system's impulse response, the location of correlation peaks would reveal the cycles during which a logical 1 was transmitted; similarly, the absence of correlation peaks would reveal the cycles during which a logical 0 was transmitted. The reported data rate is ≈500 kb/s. Figure 4 illustrates the experimental setting.

III. Crosstalk Coupling Channel

It is known that interconnect crosstalk inside an FPGA can affect signal delays [24]. Provelengios et al. recently examined long wire coupling on various types of wires across three FPGAs implemented in technology nodes ranging from 60 to 20 nm (Cyclone IV, Stratix V, Arria 10), and demonstrated that information leakage exists in all of them [25]. These unintentional transmissions pose new risks for multi-user scenarios, including FPGA/CPU hybrids and cloud infrastructures offering FPGA solutions. In these setups, an adversarial receiver can be placed next to long wires used by other third-party vendors and eavesdrop on the signals carried by them.

A. Crosstalk Coupling Attacks

Giechaskiel et al. observed that a long routing wire carrying a logical 1 reduces the propagation delay of another adjacent, but unconnected, long wire in the FPGA routing network [26]. As a consequence, the information that 1 is being transmitted can be inferred. This effect may be undesired (side-channel leakage) or intentional (covert communication). The attack setting is illustrated in Figure 5.

The transmitter of information consists of a buffer driving one or more long-wire segments connected end-to-end. The receiver uses long wires that are adjacent to the transmitter's wires, and employs a ring oscillator (a closed chain of an inverter and, for example, two buffers) to measure the change in wire delay due to crosstalk effect [27]. Since the wire delay influences the frequency of oscillation of the ring oscillator, the receiver is equipped with a counter to measure the RO frequency. When logical 1 is transmitted, the counter of

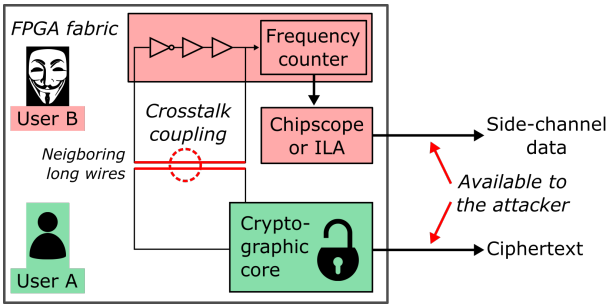

Fig. 5. An illustration of the crosstalk coupling attack. Here, an adversary creates a ring oscillator from an odd number of inverters, one or more buffers, and a long wire. The long wire is chosen carefully, to be very close to another long wire, which is part of the victim. By observing the change in the oscillation frequency of the ring oscillator, the adversary can infer the secret information sent over the victim's long wire.

RO oscillations reports higher values than when logical 0 is transmitted. Additionally, the longer the overlap between the neighboring long wires, the more pronounced the crosstalk effect is. This phenomenon is still measurable, although 20× weaker, when the transmitter and receiver wires are separated by a single long wire. When the transmitter and the receiver pair are separated even farther, the coupling is too weak and the transmitted data cannot be reliably inferred.

Since the delay of a long wire depends only on the proportion of time for which the nearby wire is carrying a logical 1, and not on the signal switching frequency, the counter keeping track of the number of RO oscillations in effect contains information on the Hamming weight of the transmitted sequence. To extract the bits being transmitted, one can therefore observe the sequence of bits through a sliding window. For illustration, let us suppose that in one measurement period the long wire carries w consecutive bits of the N-bit key K. By repeating the measurements in sliding windows of size w bits (Figure 6) and by comparing the Hamming weights (measured by the RO oscillation counter) of the bits in all subsequent windows, an attacker can infer the relationship between the bits in these windows. For example, subtracting the Hamming weight of bits in windows W_0 and W_1 will reveal the relationship between key bits K_0 and K_w: if the result is positive, $K_0 = 1$ and $K_w = 0$; if the result is negative, $K_0 = 0$ and $K_w = 1$. Repeating this procedure, one can guess the remaining key bits. Giechaskiel et al. estimate the probability of successfully recovering all N bits of a key as a function of the window size w: a window of 10 bits can fully recover a 64-bit key with 78% probability, while a 30-bit window can fully recover a 264-bit key with 87% probability [26].

Ramesh et al. successfully performed a crosstalk side-channel attack to recover the encryption key from an AES circuit on Intel Cyclone IV and Stratix V FPGAs [28]. To quantify the number of encryptions performed before the correct guess can be distinguished, they used the metric of *measurements-to-disclosure* (MTD). It took them 217 and 1.5 million encryptions, respectively, to extract a key byte at

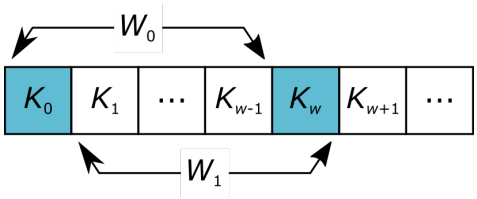

Fig. 6. Sliding-window approach for guessing the values of the bits K_i transmitted on the neighboring long wire. For example, subtracting the Hamming weight of all bits in windows W_0 and W_1 suggests the value of bits K_0 and K_w: if the result is positive, $K_0 = 1$ and $K_w = 0$; if the result is negative, $K_0 = 0$ and $K_w = 1$. The Hamming weight is measured by the counter of the RO oscillations.

operating frequencies of 10 kHz and 4 MHz. The higher the clock frequency, the smaller the side channel signal and thus higher required MTD. Similarly, the longer the wire, the lower the MTD: 328k for a length of one C4 long wire and 40k for a length of 10 C4 wires.

B. Crosstalk Coupling as Covert Communication Channel

Giechaskiel et al. demonstrated that two neighboring long wires in FPGA routing network can be used for covert communication [26]. To facilitate information transfer even in the presence of environmental changes (temperature and voltage variations) they proposed a Manchester encoding scheme: logical 0 transmitted as a pair of bits (0,1) and logical 1 transmitted as the opposite par (1,0). Under this encoding scheme, the channel bandwidth was estimated to ≈6.1 kb/s. To further distinguish between the noise and the signal, they introduced N-bit start-of-frame and end-of-frame patterns, leading to more accurate communication but also reduced bandwidth to ≈4.9 kb/s.

To enable crosstalk covert communication, both parties need to use long wires and constrain their placement to adjacent wires. However, the channel itself requires very little logic: 71 LUTs and 66 registers, excluding resources occupied by ChipScope to transfer measurements to a PC.

IV. Electromagnetic Emission Channel

Electrical current flowing through a conductor induces electromagnetic (EM) emanations; they may be intentional (direct emanations) or unintentional (due to nonlinear coupling between signals) [29]. Each active component of the device produces its own emanations, but also affects emanations from other components. Hence, these multiple emanations provide multiple views of events unfolding within the device. Views emphasizing different active components can be obtained by using different types and positions of current probes [30] or even by focusing on different types of emanations that can be captured by a single probe. This is in contrast to the power side channel where there is only a single aggregated view of net current flow. The presence of multiple views make the EM side channel(s) more powerful than the power SCA [29].

EM signals propagate not only via radiation, but also via conduction. To capture radiated signals, probes are placed as close as possible (not more than a wavelength away), while

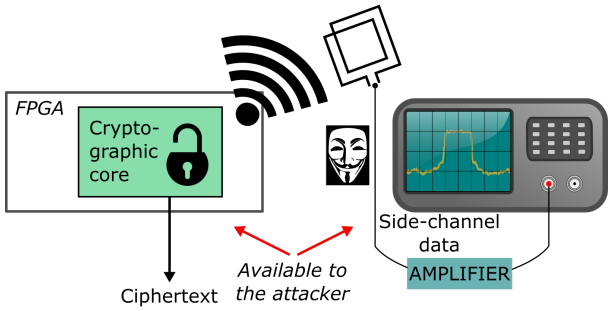

Fig. 7. An illustration of the electromagnetic side-channel attack. The adversary uses electromagnetic-field sensors to capture radiated emissions. To preserve as many frequency components in the spectra as possible, high-quality amplifiers are often used in addition. EM side-channel attack requires physical proximity to the device.

to capture conductive signals one uses current probes. In any case, physical proximity to the device is required (Figure 7).

A local power SCA is considered easier to perform than an electromagnetic attack (EMA) even though it often requires a slight modification to the device printed circuit board (PCB): for example, the preparation of a point at which to monitor the device core voltage. EM radiation, in contrast, can be measured without any modification to the PCB, and can even be measured at some distance. In addition, monitoring the power consumption of a device is becoming increasingly difficult due to the decrease of signal-to-noise ratio (SNR) brought by core voltage reductions, on-chip decoupling capacitors, and system-on-chip implementations. EMA has consequently become a greater threat.

Two types of EM-analysis attacks are distinguished. In a **simple** EMA attack, an attacker uses the side-channel information from one measurement directly to determine (parts of) the secret key. In a **differential** EMA (DEMA) attack, many measurements are made in order to filter out noise.

Kim et al. investigated the differential side channel resistance of the block cipher ARIA [31] implemented on Altera APEX 20K device [32] . They demonstrated successful differential EM-attack (DEMA) both in near-field (with 200 traces only) and in far-field (2k traces), which is harder as the emissions in the far field include more noise. To measure the near-field EM traces, they used LANGER RF-R 400-1 EM probe and a LANGER preamplifier. The far-field EM traces were measured usig a directional antenna with a frequency range from 200 MHz to 1 GHz, connected to the oscilloscope via a preamplifier (30 dB). Further, they succeeded in second order DEMA against masked ARIA implementation, at the cost of significantly more traces (100k traces).

Hori et al. developed Sasebo-GIII board (equipped with Xilinx 28-nm Kintex-7 FPGA) to perform correlation-based electromagnetic analysis (CEMA) under the Hamming-distance model [33]. They measured EM radiation emissions from the AES circuit in the Kintex-7 FPGA and compared them to those of the same AES circuit in 65nm Virtex-5 FPGA (Sasebo-GII). Kintex-7 FPGA was fabricated in newer technology, with lower core voltage and, consequently, lower side-channel information. The waveforms of the emitted EM radiation were acquired using a Langer LF-B 3 EM probe, a Miteq AU-3A-0150 amplifier (50 dB, 0.3–600 MHz), a fifth-order Bessel low-pass filter, and an Agilent DSO6104A oscilloscope. Contrary to expectations and despite $5\times$ lower measured voltage on the newer platform, only 7k traces were sufficient to recover a key, compared to 19k traces on the older platform. One of the causes to this, the authors claim, could be the AES structure and the physical positions of the subkey bytes.

Carlier et al. investigated how to attack an FPGA implementation of AES where all bytes are processed in parallel [34]. They concluded that high frequency and parallel computations are not a sufficient protection against DEMA, as by moving the probe one can detect specific bit leakage.

Mulder et al. performed DEMA on a hardware implementation of an elliptic curve cryptosystem [35], [36]. Using the distance of mean test and as few as 2k measurements, they managed to retrieve the right key bit.

V. Thermal Channel

Some physical channels have the property of keeping their state longer than others. One such channel is the thermal covert channel, as it takes time to warm up or cool down a device. In 2011, Iakymchuk et al. discovered the temperature-based covert communication channel in FPGAs [37]. The channel enabled bidirectional exchange of an arbitrary bitstream between two electrically separated parts of the FPGA during its normal operation. Transmitter and receiver modules were both based on ring-oscillators. They report $1/8$ bps transmission speed between Xilinx Spartan-IIE and an external trasceiver and up to 1 bps for internal communication.

Very recently, Tian et al. have demonstrated that it is possible to use temperature to leak sensitive data in the cloud FPGAs [38]. Their platform of choice were Microsoft Catapult servers in the Texas Advanced Computing Center (TACC). To warm up an FPGA, they would instantiate circuits that would toggle frequently, consume a lot of power and, consequently, warm up the FPGA. Their heater was composed of an array of ring oscillators, enabled using a control signal. After some time (heating period), they would disable the heater, and reconfigure the very same FPGA with another design, thus simulating a different user occupying the same FPGA at a later point in time. This user would be the receiver of information: it too would use a ring oscillator, for only a very short time (not to affect the device temperature) but this time to measure the frequency of oscillation which is inversely proportional to the temperature:

$$T \nearrow \Longrightarrow d \nearrow \Longrightarrow f \searrow .$$

Here d is the inverter gate delay and T the temperature. To transmit information, they used simple on-off keying: logical 1 would correspond to signal presence, while logical 0 to the absence of a signal, where signal presence corresponds to high temperature of the FPGA chip. There are many factors that make this **temporal** covert communication channel

difficult to implement and unreliable; for example, it is hard to imagine that one would be given the possibility to choose a specific FPGA instance in the datacenter or the cloud. Then, although heat takes time to dissipate, the two participants in the communication need to synchronize well, as the receiver needs to reconfigure the FPGA very quickly after the sender has vacated it, to prevent the temperature from stabilizing and the error rate from becoming too high. Additionally, in the scenario where FPGA is shared, another unknown user may be heating the FPGA die, thus polluting the thermal covert channel. Finally, the communication bandwidth is very low, as for every bit to be transmitted, the FPGA needs to be reconfigured twice and warmed up in the meantime. To increase the bandwidth, one could employ multiple FPGAs at a time, at the cost of higher price for using cloud resources. The maximum bandwidth reported, when no waiting or idling time is considered, is $\approx$1 bps when as many as 256 FPGAs are used in parallel.

VI. Discussion

Common questions one tries to address when discussing side channels are how to prevent or protect from an attack or covert communication. Before exploring these topics, we look at some seldom regarded and yet interesting properties of physical side channels.

Figure 8 lists FPGA families that were experimentally proven to be susceptible to physical SCAs and shows that FPGAs are vulnerable to physical side channel attacks regardless of the **process node**. However, crosstalk SCAs appear more likely to succeed in FPGAs in newer technology nodes, thanks to reduced spacing between neighboring wires.

Figure 9 illustrates the **timeline** of the key research contributions we present in this survey. Recently, research focus shifted towards security vulnerabilities of remote and/or shared FPGAs in datacenters and the cloud.

A. Experimental Equipment Cost and Complexity

If we were to rank the physical channels based on the complexity and cost of the equipment for measurement and analysis, then the EM channel would be on top of the list. First, EMA is performed in the vicinity of the device. Then, although magnetic field probes can be made in-house and quite cheaply (a wire loop, with ends soldered to a BNC connector, for instance), the emanation from the device is of very low amplitude and wide spectra. Hence, a high-quality and thus expensive amplifier needs to be used. Finally, a high-end oscilloscope is mandatory to record a good-quality data for subsequent statistical differential side-channel analysis. Comparably or somewhat less expensive are power SCA attacks performed in the vicinity of the device, as they require an oscilloscope and, sometimes, modifications to the PCB to enable current or voltage monitoring. In comparison, remote power, thermal, and crosstalk SCA are all quite cheap, because all they require is access to a remote FPGA and a way of sending the collected data to the adversary.

B. Portability

In terms of portability, electromagnetic SCA attacks are very convenient, as they do not require changing the experimental setup or methodology: regardless of the FPGA device at hand, the procedure is the same. Power SCAs, when physical access to the device is available, are also convenient, except that every board may require different modifications in order to enable quality current or voltage measurements. Unlike the above, remote power and crosstalk SCAs require careful redesign and recalibration of sensors, tuned for the target FPGA family, printed circuit board, and environmental conditions (temperature, for instance). Thermal covert communication is even more challenging, as not only that the transmitter and the receiver pair need to use the exact same remote FPGA, they need to be placed inside the same FPGA region, to reduce the risk of measuring heat produced by un unrelated circuit.

C. Prevention and Protection

Preventing EM analysis or local power SCAs can be done by restricting the access to the device, but that is often not possible; for example, smart cards and a variety of embedded devices are meant to be handled by anyone and are thus at risk. Preventing remote power SCA is best done by not allowing neither FPGA sharing nor board sharing. This would prevent any form of information leakage through the common power supply. However, multitenancy is one of the fundamental features of datacenters and the cloud, and it helps amortizing the investment and reducing the costs for maintenance, electricity, and cooling. Once FPGA or board sharing will be widely enabled, strategies for protecting from remote power SCA attacks will have to be considered. For example, while the user design is being placed and routed, the FPGA primitives that may be used to create a voltage sensor (buffer delay lines or ring oscillators) can be detected. Additionally, these primitives require careful placement constraints, which, if detected, could help identifying potentially malicious circuits.

To increase the difficulty of power SCA attacks, various hiding and masking schemes have been developed. Unfortunately, these countermeasures require considerable area overhead and often lower design performance. An example is Wave-Dynamic Differential Logic; a hiding measure based on ensuring constant power consumption [39]. It may sound easy to implement, but it is in fact very hard to achieve on FPGAs, as even the slightest difference in routing or placement makes the protection imperfect and renders the design vulnerable [40], [41]. Masking countermeasure, on the other hand, removes the correlation between the device power consumption and the secret key by XOR-ing the key with a random mask and by modifying the cryptographic design to ensure symmetry of operations on the masked key [42]. Masking, if well implemented, can protect cryptographic cores from first-order attacks, but the higher-order attacks remain an active threat.

Given that crosstalk coupling requires the use of long wires in very close proximity (precisely, with none or at most one wire between the two wires that take part in information

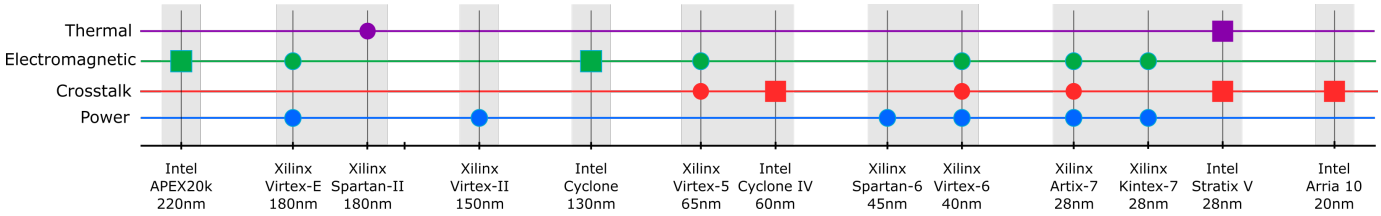

Fig. 8. FPGA devices successfully used to demonstrate a physical side-channel attack or covert communication. The circles represent Xilinx FPGAs, whereas the squares represent Intel FPGAs. For most of the experiments, Xilinx FPGAs were the platform of choice. As a general observation, FPGAs seem to be susceptible to physical SCAs regardless of the technology node. In particular, newer FPGAs are more susceptible to crosstalk SCAs, due to the reduced spacing between neighboring routing wires.

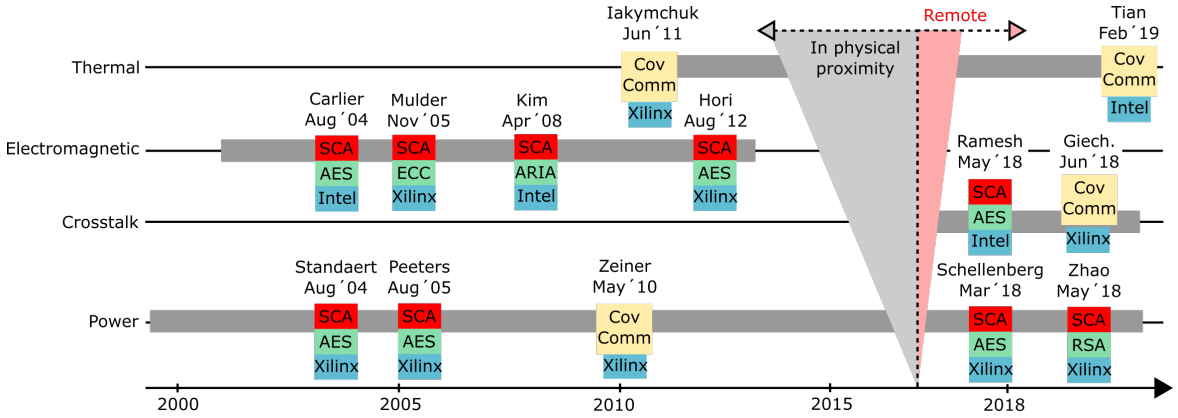

Fig. 9. Timeline of the key research contributions we present in this survey (not an exhaustive list). Gray horizontal bars start at the earliest reported successful attack. Since 2018, the focus shifted towards side-channel vulnerabilities in remote and/or shared FPGAs in datacenters and the cloud.

transfer), adding isolation between two circuits of two different users [43] can prevent crosstalk side-channel attack. However, the information leakage may happen between two IP modules that are placed and routed as part of a single design. Leaving nearby long wires unoccupied or not using the configurable logic blocks accessible by those long wires, to avoid eavesdropping, could take considerable toll on dense designs. As a consequence, crosstalk channel should be addressed with care in circuits assembled of IPs of untrusted origin.

Preventing temporal thermal covert communication in the cloud setting is not a hard task. All one would need to do is disable users from selecting themselves the FPGA board or to enforce a minimum idle period between users, so that the FPGA cools down and reaches a steady state temperature before a new user uploads the code. Preventing thermal covert communication when the receiver is in the physical proximity to the FPGA is more challenging, as it is hard to distinguish intentional from unintentional heating. Given the extremely low channel bitrate—whopping 1 bps when 256 FPGAs are used in parallel—one can imagine that the adversaries would try to heat the FPGA as quickly as possible. This sudden and excessive heating could be detected indirectly, as abrupt and long-lasting current consumption, and the affected FPGA could be put to a power-down state to protect from further information leakage.

Preventing designs containing combinational loops, such as ring oscillators, makes it harder to implement power wasting circuits or some types of voltage sensors, and thus improves FPGA security. This quite radical strategy is already applied in Amazon AWS [44]. However, removing ring oscillators means losing their various legitimate uses: from thermal and device health monitors [45] to hardware Trojan detectors [46], true random number generators [47], and physical unclonable functions [48]. Moreover, voltage sensors and heaters can be implemented even without combinational loops, examples being the delay-line sensors and large shift registers.

VII. Conclusions

In this paper, we present and discuss FPGA security vulnerabilities caused by FPGA design and run-time physical properties: power consumption, temperature, electromagnetic emission, and long-wire crosstalk coupling. We give special attention to most recent findings related to shared and/or remote FPGAs. We discuss prevention and protection strategies, but the conclusion remains that no perfect countermeasure exists, let alone a universal one. Since FPGAs have been added to datacenters and the cloud, researchers have discovered new ways for covert communication and side-channel attacks, and this trend will continue.

References

[1] T. Huffmire, B. Brotherton, T. Sherwood, R. Kastner, T. Levin, T. D. Nguyen, and C. Irvine, "Managing security in FPGA-based embedded systems," *IEEE Design and Test of Computers*, vol. 25, no. 6, pp. 590–598, Nov. 2008.

[2] S. M. Trimberger and J. J. Moore, "FPGA security: Motivations, features, and applications," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1–15, Aug. 2014.

[3] F. Durvaux, S. Kerckhof, F. Regazzoni, and F.-X. Standaert, *A Survey of Recent Results in FPGA Security and Intellectual Property Protection*. New York, NY: Springer New York, 2014.

[4] S. Drimer, "Volatile FPGA design security—A survey," Jan. 2008. [Online]. Available: http://www.cl.cam.ac.uk/ sd410

[5] J. Fan, X. Guo, E. D. Mulder, P. Schaumont, B. Preneel, and I. Verbauwhede, "State-of-the-art of secure ECC implementations: A survey on known side-channel attacks and countermeasures," in *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, Anaheim, CA, USA, Jun. 2010, pp. 76–87.

[6] P. C. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems," in *Advances in Cryptology—CRYPTO '96*, ser. Lecture Notes in Computer Science, N. I. Koblitz, Ed. Berlin: Springer, Sep. 1996, vol. 1109, pp. 104–13.

[7] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO '99*, Santa Barbara, CA, USA, Aug. 1999, pp. 387–397.

[8] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Transactions on Computers*, vol. 51, no. 5, pp. 541–552, May 2002.

[9] S. B. Örs, F. Gürkaynak, E. Oswald, and B. Preneel, "Power analysis attack on an ASIC AES implementation," in *Proceedings of the International Conference on Information Technology: Coding and Computing*, Las Vegas, NV, USA, Aug. 2004, pp. 1–7.

[10] S. B. Örs, E. Oswald, and B. Preneel, "Power-analysis attacks on an FPGA—first experimental results," vol. 2779, pp. 35–50, 2003.

[11] F.-X. Standaert, L. van Oldeneel tot Oldenzeel, D. Samyde, and J.-J. Quisquater, "Power analysis of FPGAs: How practical is the attack?" Lisbon, Portugal, Sep. 2003, pp. 701–710.

[12] F.-X. Standaert, B. Örs, and B. Preneel, "Power analysis of an FPGA: Implementation of rijndael: Is pipelining a DPA countermeasure," in *Proceedings of the Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, Boston, MA, USA, Aug. 2004, pp. 30–44.

[13] F.-X. Standaert, S. B. Örs, J.-J. Quisquater, and B. Preneel, "Power analysis attacks agains FPGA implementation of the des," in *Proceedings of the 14th International Conference on Field-Programmable Logic and Applications*, Leuven, Belgium, Sep. 2004, pp. 84–84.

[14] E. Peeters, F.-X. Standaert, N. Donckers, and J.-J. Quisquater, "Improved higher-order side-channel attacks with FPGA experiments," *CHES 2005*, vol. 3659, pp. 309–323, 2005.

[15] J. Waddle and D. Wagner, "Towards efficient second-order power analysis," *CHES 2004*, vol. 3156, pp. 1–15, 2004.

[16] F. Schellenberg, D. R. Gnad, A. Moradi, and M. B. Tahoori, "An inside job: Remote power analysis attacks on FPGAs," in *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition*, Dresden, Germany, Mar. 2018, pp. 1111–1116.

[17] M. Zhao and G. E. Suh, "FPGA-based remote power side-channel attacks," in *Proceedings of IEEE Symposium on Security and Privacy*, San Francisco, CA, USA, May 2018, pp. 805–820.

[18] K. M. Zick, M. Srivastav, W. Zhang, and M. French, "Sensing nanosecond-scale voltage attacks and natural transients in FPGAs," in *Proceedings of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, Monterey, CA, USA, Feb. 2013, pp. 101–104.

[19] D. R. Gnad, F. Oboril, S. Kiamehr, and M. B. Tahoori, "Analysis of transient voltage fluctuations in FPGAs," in *Proceedings of the IEEE International Conference on Field Programmable Technology*, Xi'an, China, Dec. 2016, pp. 1–8.

[20] S. Pant, "Design and analysis of power distribution networks in VLSI circuits," Ph.D. Thesis, University of Michigan, Ann Arbor, MI, 2008.

[21] Avnet, *ZedBoard Power Distribution and Decoupling System*, Avnet, Phoenix, AZ, 2012. [Online]. Available: http://zedboard.org

[22] F. Schellenberg, D. R. Gnad, A. Moradi, and M. B. Tahoori, "Remote inter-chip power analysis side-channel attacks at board-level," in *Proceedings of the International Conference on Computer Aided Design*, San Diego, CA, USA, Nov. 2018.

[23] D. Ziener, F. Baueregger, and J. Teich, "Using the power side channel of FPGAs for communication," in *Proceedings of the 18th IEEE Symposium on Field-Programmable Custom Computing Machines*, Charlotte, NC, USA, May 2010, pp. 237–244.

[24] S. J. E. Wilton, "A crosstalk-aware timing-driven router for FPGAs," in *Proceedings of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, Monterey, CA, USA, Feb. 2001, pp. 21–28.

[25] G. Provelengios, C. Ramesh, S. B. Patil, K. Eguro, R. Tessier, and D. Holcomb, "Characterization of long wire data leakage in deep submicron FPGAs," in *Proceedings of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, Seaside, CA, USA, Feb. 2019, pp. 292–297.

[26] I. Giechaskiel, K. B. Rasmussen, and K. Eguro, "Leaky wires: Information leakage and covert communication between FPGA long wires," in *Proceedings of 13th ACM ASIA Conference on Information, Computer and Communications Security*, Songdo, Incheon, Republic of Korea, Jun. 2018, pp. 15–27.

[27] M. Gag, T. Wegner, A. Waschki, and D. Timmermann, "Temperature and on-chip crosstalk measurement using ring oscillators in FPGA," in *Proceedings of the 15th IEEE International Symposium on Design and Diagnostics of Electronic Circuits & Systems*, Tallinn, Estonia, Apr. 2012, pp. 1–4.

[28] C. Ramesh, S. B. Patil, S. N. Dhanuskodi, G. Provelengios, S. Pillement, D. Holcomb, and R. Tessier, "FPGA side channel attacks without physical access," in *Proceedings of the 26th IEEE Symposium on Field-Programmable Custom Computing Machines*, Boulder, CO, USA, May 2018, pp. 1–8.

[29] D. Agrawal, B. Archambeault, J. R. Rao, and P. Rohatgi, "The EM side-channel(s)," *CHES 2003*, vol. 2523, pp. 29–45, 2003.

[30] K. Gandolfi, C. Mourtel, and F. Olivier, "Electromagnetic analysis: Concrete results," in *Proceedings of the Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, Paris, France, May 2001, pp. 251–261.

[31] D. Kwon, J. Kim, S. Park, S. H. Sung, Y. Sohn, J. H. Song, Y. Yeom, E.-J. Yoon, S. Lee, J. Lee, S. Chee, D. Han, and J. Hong, "New block cipher: ARIA," in *Proceedings of the 6th International Conference on Information Security and Cryptology*, Seoul, Korea, Nov. 2003, pp. 1–14.

[32] C. Kim, M. Schläffer, and S. Moon, "Differential side channel analysis attacks on FPGA implementations of ARIA," *ETRI Journal*, vol. 2, no. 30, pp. 315–325, Apr. 2008.

[33] Y. Hori, T. Katashita, A. Sasaki, and A. Satoh, "Electromagnetic side-channel attack against 28-nm FPGA device," in *Proceedings of the 13th International Workshop on Information Security Applications*, Jeju Island, Korea, Aug. 2012, pp. 1–9.

[34] V. Carlier, H. Chabanne, E. Dottax, and H. Pelletier, "Generalizing square attack using side-channels of an AES implementation on an FPGA," in *International Conference on Field Programmable Logic and Applications*, Tampere, Finland, Aug. 2004, pp. 1–16.

[35] E. D. Mulder, P. Buysschaert, S. B. Örs, P. Delmotte, B. Preneel, G. Vandenbosch, and I. Verbauwhede, "Electromagnetic analysis attack on an FPGA implementation of an elliptic curve cryptosystem," in *EUROCON*, Belgrade, Serbia, Nov. 2005, pp. 1879–1882.

[36] E. D. Mulder, S. B. Örs, B. Preneel, and I. Verbauwhede, "Differential electromagnetic attack on an FPGA implementation of elliptic curve cryptosystems," in *2006 World Automation Congress*, Budapest, Hungary, Jul. 2006, pp. 1–7.

[37] T. Iakymchuk, M. Nikodem, and K. Kepa, "Temperature-based covert channel in FPGA systems," in *6th International Workshop on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC)*, Montpellier, France, Jun. 2011, pp. 1–7.

[38] S. Tian and J. Szefer, "Temporal thermal covert channels in cloud FPGAs," in *Proceedings of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, Seaside, CA, USA, Feb. 2019, pp. 298–303.

[39] K. Tiri and I. Verbauwhede, "A logic level design methodology for a secure DPA resistant ASIC or FPGA implementation," in *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition*, Paris, France, Feb. 2004, pp. 1–6.

[40] L. Sauvage, S. Guilley, J.-L. Danger, Y. Mathieu, and M. Nassar, "Successful attack on an FPGA-based WDDL DES cryptoprocessor without place and route constraints," in *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition*, Nice, France, Apr. 2009, pp. 1–6.

[41] P. Yu and P. Schaumont, "Secure FPGA circuits using controlled placement and routing," in *Proceedings of IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS)*, Salzburg, Austria, Sep. 2007, pp. 45–50.

[42] F. Regazzoni, Y. Wang, and F.-X. Standaert, "FPGA implementations of the AES masked against power analysis attacks," in *Second International Workshop on Constructive Side-Channel Analysis and Secure Design (COSADE)*, Darmstadt, Germany, Feb. 2011, pp. 1–11.

[43] T. Huffmire, B. Brotherton, G. Wang, T. Sherwood, R. Kastner, T. Levin, T. Nguyen, and C. Irvin, "Moats and drawbridges: An isolation primitive for reconfigurable hardware based systems," in *IEEE Symposium on Security and Privacy*, Berkely, CA, USA, May 2007, pp. 1–15.

[44] A. W. S. (AWS), "Combinational loops disabled," Amazon Web Services, Seatle, WA, USA, 2017. [Online]. Available: https://forums.aws.amazon.com/message.jspa?messageID=806151

[45] K. M. Zick and J. P. Hayes, "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, Mar. 2012.

[46] S. Kelly, X. Zhang, M. Tehranipoor, and A. Ferraiuolo, "Detecting hardware trojans using on-chip sensors in ASIC design," *Journal on Electronic Testing: Theory and Applications*, vol. 31, no. 1, pp. 11–26, Feb. 2015.

[47] I. Vasyltsov, E. Hambardzumyan, Y.-S. Kim, and B. Karpinskyy, "Fast digital TRNG based on metastable ring oscillator," in *CHES 2008*, vol. 5154, Washington DC, USA, Aug. 2008, pp. 164–180.

[48] A. Maiti, J. Casarona, L. McHale, and P. Schaumont, "A large scale characterization of RO-PUF," in *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, Anaheim, CA, USA, Jun. 2010, pp. 94–99.