Creating a Backscattering Side Channel to Enable Detection of Dormant Hardware Trojans

Luong N. Nguyen, *Student Member, IEEE*, Chia-Lin Cheng *Student Member, IEEE*, Milos Prvulovic, *Senior Member, IEEE*, and Alenka Zajić, *Senior Member, IEEE*

***Abstract*—This paper describes a new physical side channel, i.e. the backscattering side channel, that is created by transmitting a signal toward the IC, where the internal impedance changes caused by on-chip switching activity modulate the signal that is backscattered (reflected) from the IC. To demonstrate how this new side-channel can be used to detect small changes in circuit impedances, we propose a new method for non-destructively detecting hardware Trojans (HTs) from outside of the chip. We experimentally confirm, using measurements on one physical instance for training and nine other physical instances for testing, that the new side-channel, when combined with an HT detection method, allows detection of a *dormant* HT in 100% of the HT-afflicted measurements for a number of different HTs, while producing no false positives in HT-free measurements. Furthermore, additional experiments are conducted to compare the backscattering-based detection to one that uses the traditional EM-emanation-based side channel. These results show that backscattering-based detection outperforms the EM side channel, confirm that dormant HTs are much more difficult for detection than HTs that have been activated, and show how detection is affected by changing the HT's size and physical location on the IC.**

***Index Terms*—Hardware Trojan, Hardware security, hardware trust, Backscattering side channel, Trojan detection.**

I. Introduction

Integrated circuits (IC) have become an integral aspect of our lives, by controlling most of electronic devices ranging from cellphones and washing machines to airplanes and rockets. Thus, the problem of ensuring authenticity and trust for ICs is already critically important, especially for sensitive fields such as military, finance, and governmental infrastructure, and is gaining in importance as an increasing number of "things" become "smart" and connected into the Internet-of-Things (IoT). However, cost and time-to-market considerations have led IC vendors to outsource some, and in most cases many, steps in the IC supply chain. The sheer number and diversity of entities involved in modern IC supply chain, each with its own set of potentially malicious actors that can insert malicious modifications, referred as hardware Trojan (HT), in the IC [1], makes it difficult to trust the resulting ICs, especially when potentially adversarial foreign governments are among the potentially malicious actors in the IC supply chain. The potential existence of HTs significantly undermines the trust in any system that uses that IC, because the hardware usually provides the base layer of security and trust that all software layers depend and build on [2], [3], [4]. Specifically, all software protections, correctness analysis, or even proofs rely on the hardware executing instructions as specified, and by violating this assumption HTs can defeat the best software protections and/or subvert even software functionality that is otherwise completely correct and vulnerability-free.

Typically, an HT is designed to be stealthy, so it only changes the functionality of the original circuit when specific conditions have been met. Thus the design of an HT typically has two key components: the *payload*, which implements the modification of the original circuit's behavior[1], and the *trigger*, which detects when the conditions for activating the payload have been met. The conditions that activate an HT occur very rarely, and until activated the payload is usually highly inert - it simply allows the IC to follow its original input/output behavior. This makes HTs extremely challenging to detect by traditional functional verification and testing - test inputs are unlikely to activate the HT, and without activation the HT has no effect on functional behavior of the IC.

A. Prior Counter-HT Approaches

Some techniques focus on making the IC resilient to the presence of HTs, i.e. on preventing the HT's payload from modifying the behavior of the IC, mostly by using fault-tolerance-inspired approaches to operate correctly even when an HT has been able to modify some of the internal signals. However, these techniques protect only certain parts of the system, such as a bus [5] or on-chip interconnect [6], require redundant activity during normal operation [7], and/or rely on reconfigurable logic [8].

Most counter-HT techniques focus on detecting the presence of HTs. Some HT detection approaches are *destructive*, e.g. relying on successive removal of the IC's layers to scan the actual layout of the IC, reverse-engineer its GDSII and/or netlist-level design [9], and compare it to a trusted design. However, all the ICs that are found to be HT-free through such analysis are also destroyed by the scan, and the reverse-engineering is extremely expensive and time-consuming, so such destructive techniques can only be applied to a small sample of the larger population of IC.

This work has been supported, in part, by NSF grants 1651273 and 1740962, and ONR grant N00014-17-1-2540. The views and findings in this paper are those of the authors and do not necessarily reflect the views of NSF and ONR

Luong N. Nguyen, Chia-Lin Cheng, and Alenka Zajić are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA, and Milos Prvulovic is with the School of Computer Science, Georgia Institute of Technology, Atlanta, GA 30332, USA.

[1] The HT's payload can also implement a non-functional change in the IC's behavior, e.g. to increase its power consumption, increase the IC's side channel leakage of information, decrease its expected lifetime, etc.

Non-destructive HT detection approaches can be categorized according to whether they are applied to the design of the yet-to-be-fabricated IC (pre-silicon approaches), or to fabricated IC (post-silicon approaches). Pre-silicon approaches use functional validation, and code and gate-level netlist analysis [10], [11], but they cannot detect HTs that are inserted after the design stage, e.g. by editing the physical layout of the IC at the foundry. To overcome such concerns, post-silicon methods attempt to identify HTs in ICs received from the foundry.

Post-silicon non-destructive approaches detect HTs either through testing the functional properties of the IC, or by measuring non-functional (side channel) behavior of the IC as it operates. Functional testing involves finding inputs that are likely to trigger unknown HTs that may exist in the IC, causing the payload of the HT to propagate the effects of the payload to the outputs of the IC, where they can be found to differ from expected outputs [12]. However, trigger conditions for HTs are designed to be difficult to reach accidentally, so the probability of detecting HTs is extremely low for conventional functional testing techniques. Additionally, functional testing techniques are likely to fail in detecting HTs whose payload does not change the input/output behavior or the IC, but rather causes increased power consumption, side channel leakage of sensitive information, etc.

Among post-silicon approaches, HT detection through side channel analysis appears to be the most effective and widely used approach [13], [14]. These methods measure one or more non-functional properties of the IC as it operates, and compare these measurements to reference signals obtained through either simulation or measurement on a device known to be genuine. Side channels used by HT detection techniques include power consumption [15], [16], [17], [18], leakage current [19], temperature [20], [21], and electromagnetic emanations (EM) [22], [23], [24], and some approaches even combine measurements from multiple side channels [25], [26].

Among side channel-based HT detection approaches, some add the side channel measurement capability to the chip itself, while others rely on measurements that are external to the chip itself. With on-chip measurement, the measurement circuitry is added to the design [27], [28], [29], which allows the specific chosen signals to be measured close to the signal's source. However, the additional circuitry for measurement, and for routing the desired signals to the measurement circuitry, impacts chip size, manufacturing cost, performance, and power, and this impact increases as the set of individually measurable signals increases.

Finally, external-measurement side channel techniques require no modifications to the IC itself, and instead rely on externally observable side-effects of the IC's normal activity. Since an HT is typically much smaller than the original circuit, an ideal side channel signal would have little noise and interference so that the HT's small contribution to the signal is not obscured by the noise. Additionally, the HT's payload is largely inert until activated, and activation during measurement is highly unlikely, so ideally the side channel signal would be affected by the presence of the payload circuitry, even when it is inert. Finally, before activation, what little switching activity the HT does create is in its trigger component, which usually has only brief bursts of switching when the inputs it is monitoring change. Thus an ideal side channel signal would have high bandwidth, such that these brief bursts of current fluctuation due to switching activity in the HT can be identified. Unfortunately, existing externally-measurable side channel signals, such as temperature, voltage and power supply current, and electromagnetic emanations [22], tend to vary mostly in response to current variation due to switching activity. However, temperature changes slowly and has very limited bandwidth, and voltage and supply current have low bandwidth [24] because on-chip capacitances that help limit supply voltage fluctuation act as a low-pass filter with respect to both current and voltage as seen from outside the chip. Electromagnetic emanations can have high bandwidth, but their signal-to-noise ratio is affected by noise and interference.

B. Contributions

In this paper, we introduce a new physical side channel, i.e. the backscattering side channel, that is created by transmitting a signal toward the IC, where the internal impedance changes caused by on-chip switching activity modulate the signal that is backscattered (reflected) from the IC. To demonstrate how this new side-channel can be used to detect small changes in circuit impedances, we use it to implement a new proof-of-concept method for non-destructively detecting HTs from outside of the chip. The technique presented in this paper is capable of detecting different types of *inactive* HTs on multiple circuit benchmarks while tolerating variations that exist across hardware instances. To our knowledge, backscattering has never before been used as a side channel signal to infer information about the operation of electronic circuitry, even though backscattering has been used extensively for RFID tags and other short-range communications [30]. We observe that backscattering not only can be used as a side channel signal, but also that it is especially suitable for HT detection because the backscattered signal carries information about the current state of on-chip impedances, unlike traditional side channels that carry information about brief changes in current. Furthermore, like the traditional EM side channel, the backscattering side channel has high bandwidth but, unlike the traditional EM signal, the strength of the backscattered signal can be increased when needed, its frequency can be shifted to avoid noise, interference, and poor signal propagation conditions, and it can be more accurately focused on a specific part of the chip.

We test our new HT detection technique using multiple HTs from the Trusthub benchmark [31] and show that it is highly accurate in detecting even *inactive* HTs while avoiding false positives. We compare our approach to one that applies the same signal analysis to traditional electromagnetic emanations, and our results confirm backscattering yields a dramatic improvement in HT detection accuracy. We further evaluate the sensitivity of our approach by separately reducing the size of the HT's trigger and payload components, and showing that HT detection of inactive HTs largely depends on the size of the trigger component, and that our approach can detect even HTs with significantly reduced triggers. Additionally, we also evaluate how our approach is affected by manufacturing and other variations, by using different physical instances of the same design for training and testing, and find that the technique largely maintains its ability to detect HTs accurately even when trained on only one instance and used to test another.

The rest of the paper is organized as follows. In Section II, we present some background of HTs and the new impedance-based side channel. Section III defines our detection technique and algorithm, while Section IV describes the Trojans we use and how we implement those hardware Trojans on an FPGA. Section V evaluates the size and position of HT's trigger and payload, and the difference in HT detection by using EM versus the new backscattering side channel. Section V-A further evaluates the robustness of the technique, by testing it on multiple boards.Finally, Section VI concludes the paper.

II. Background

A. Hardware Trojans

Most software systems are built on the assumption that the underlying hardware can be trusted to perform the requested operations correctly, and even when incorrect hardware behavior is considered, it is assumed to be erroneous rather than malicious. HTs break this assumption, so the potential presence of unknown HTs in the system's hardware effectively eliminates trust in the overall system regardless of how trustworthy the system's software is. Over the past several years, numerous papers have been published on the topic of understanding the intent, behavior [14], [32] and implementation of HTs [33], [34], [35], [31]. Several studies have focused on characterizing and classifying HTs [36], [13], [37], [31] according to activation mechanism, functionality, location on the IC, the point in the IC design cycle and supply chain at which they are inserted, etc. A common characteristic of HTs is that they are designed to avoid detection, so they activate their malicious functionality rarely [32] to avoid being relatively easily detected, e.g. during functional testing of the IC. Therefore, a typical HT consists of a *trigger* circuit and *payload* circuit, as illustrated in Fig. 1. The trigger circuit is monitoring a set of signals to detect when the conditions for activation of the payload have been met, while the payload implements the actual malicious functionality. The malicious functionality can be functional, e.g. when the HT's output modifies the outputs of the overall circuit to cause harm or leak sensitive information, and/or non-functional, e.g., when the payload increases power consumption, causes excessive wear-out to reduce the lifetime of the IC, leaks sensitive information through a side channel, etc.

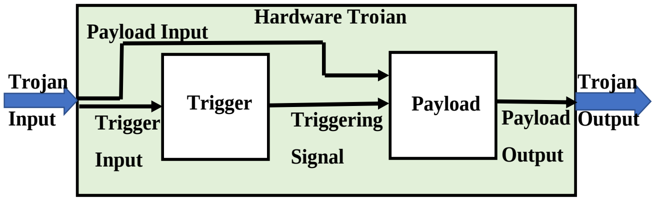


Fig. 1: Simplified Block Diagram of an HT.

B. Adversaries and Attacks

Ideally, all of the steps in this life-cycle of an IC would be performed by a single trusted entity, which would design, fabricate, test, package, and deploy the IC. However, cost-reduction, time-to-market, IC complexity, and other considerations have recently led companies to specialize in a single step in the IC design and/or manufacturing, so the overall IC is typically designed by one entity, usually includes intellectual property (IP) blocks of several other entities and design tools from yet another entity, is fabricated, tested, and packaged by one or more other entities, and is finally deployed by yet another entity. Different parts of the life cycle typically also take place in several different countries. HTs could be injected to an IC by adversaries at any stage of its design and fabrication flow. Please note that our threat model assumes a "golden" IC (known to be HT-free) can be used as a reference for training of the HT-detection mechanism. While we realize that this assumption is often unrealistic for practical deployments of HT detection, we evaluate HT detection with this assumption because it allows a fair comparison with another side channel (the EM side channel). Removing the golden-reference assumption would make the results heavily dependent on the accuracy of the model and the simulator that generate the reference signals, and different side channels would require different models/simulators that would be hard to equalize in accuracy/quality. Thus we choose to evaluate the new backscattering side channel, and to compare it to the EM side channel, under the same assumptions/conditions, in order to demonstrate the advantages of this new side-channel, namely that it can detect much smaller circuit modifications, is less susceptible to manufacturing variability, and can detect dormant HTs.

C. Backscattering

The backscattering concept has been used to enable RFID tags to transmit information with very low energy expenditure [30]. A typical RFID system based on backscattering is illustrated in Fig. 2. The data transmission requires the RFID reader to emit a continuous wave (an RF signal at some frequency f_c) toward the RFID tag. The RFID tag contains an antenna that can be connected to one of two impedances, Z_0 or Z_1, one of which is chosen to maximize the antenna's reflection coefficient (also called radar cross-section, or RCS) for frequency f_c, while the other impedance is chosen to minimize the antenna's RCS for f_c. The RFID tag typically contains an application-specific integrated circuit (ASIC) chip that can electronically switch the antenna's connection between these two impedances, which modulates the signal that reflects (backscatters) from the antenna according to the data bits the RFID tag wishes to transmit. The RFID reader then receives and demodulates the backscattered signal to retrieve the data transmitted by the tag. This enables use of very compact RFID tags, because the energy for the signal "transmitted" by the RFID tag is entirely provided by the RFID reader [2].

[2] Typically the electronic switching done by the RFID tag's ASIC is powered by energy-harvesting using the reader's signal, which completely eliminates the need for long-term energy storage (e.g. a battery) in the RFID tag.

III. New Backscattering Side Channel and Its Use for Hardware Trojan Detection

Our motivation to explore backscattering as a side channel was a hypothesis that the backscatter radio effect should be present in electronic devices. Specifically, transistors in digital circuits switch between two states (closed and open), which changes the impedances connected to wires within the IC, which should modulate a signal that is backscattered from the IC. An example of this is shown in Fig. 3 for a 2-input CMOS NAND gate, which consists of two pull-up transistors connected in parallel and two pull-down transistors connected in series, as shown in Fig. 3 (a). Depending on its output (logical 1 or logical 0), the NAND gate exhibits two impedance states shown in Fig. 3, where R_1 is the resistance of the in-parallel connection of conducting (turned-on) pull-up transistors, while R_0 is the in-series connection of conducting (turned-on) pull-down transistors. Thus the impedances “seen” from the gate’s V_{DD} and ground connections change depending on the output state of this gate, and unless the transistor geometry and doping levels are perfectly chosen to make R_1 and R_0 be exactly the same, the impedances “seen” from the gate’s output will also change with the gate’s output state [38]. Furthermore, actual impedances also have parasitic capacitances and inductances that depend on the exact geometry of the gate and its connections, making it highly likely that the overall impedances change with the gate’s output state.

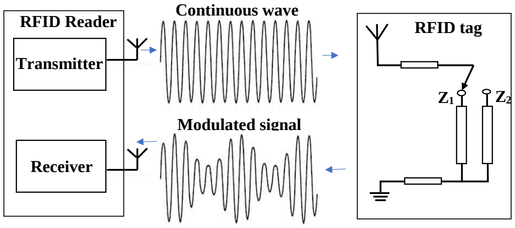
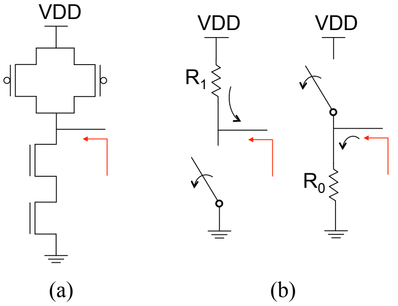


Fig. 2: An illustration of backscatter data communication.

Other types of gates exhibit similar state-dependent impedance changes, so when a continuous-wave signal is transmitted toward a set of gates, the backscattered signal can be expected to change as the gates’ states change, thus creating an *impedance-based* side channel, in contrast to the traditional EM side channel which is current-flow based.



(a) (b)

Fig. 3: CMOS NAND gate (a) and its two equivalent impedance circuits (b).

To illustrate how this concept works in practice, we implement a ring of flip-flops as shown in Fig. 4 in an Altera DE0 board with a Cyclone V FPGA (Field-Programmable Gate Array). The flip-flops are initialized with alternating values, such that each flip-flip toggles from 0 and 1 and back again with a frequency of f_m. Fig. 5 shows the resulting output voltage of a flip-flop in this ring, which has a square-wave pattern with frequency f_m.

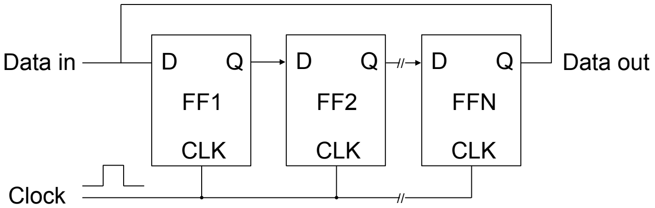


Fig. 4: Cyclical shift register.

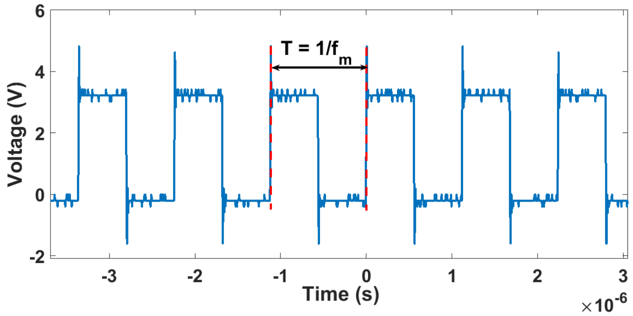


Fig. 5: Measured voltage at the output of flip-flops switching at f_m=900 kHz.

We transmit a continuous wave (sinusoidal) signal at frequency $f_{carrier}$ toward the FPGA chip, and receive the backscattered signal using the same setup as in Fig. 11.

The backscattered signal, if it is modulated by the switching activity, should contain not only a component at $f_{carrier}$, but also side-band components at frequencies $f_{carrier} - f_m$ and $f_{carrier} + f_m$. The $f_{carrier}$=3.031 GHz in this experiment was chosen to avoid interference from other periodic signals on the DE0-CV board, e.g. the crystal-oscillator-controlled 50 MHz clock and its harmonics. To ensure that the side-channel created by the backscattering effect corresponds to on-chip activity, none of the flip-flop outputs is used to control any off-chip activity, and all of the FPGA chip’s output pins are kept in a constant state throughout the experiment. Fig. 6 plots the spectra of the backscattered signal in this experiment. The first spectrum was collected for f_m=900 kHz. This spectrum contains a strong component at $f_{carrier}$, which represents the unmodulated part of the backscattered (reflected) signal, and also side-band signals 900 kHz to the left and to the right of $f_{carrier}$. These side-band signals are a consequence of the carrier signal being modulated by on-chip toggling activity through the backscattering effect. To further increase confidence that these side-band signals are indeed a consequence of the backscattered signal being modulated by on-chip toggling, we change the f_m to 1.2 MHz, and observe that the spectral component at $f_{carrier}$ remains at the same frequency, the frequencies of side-band components change with f_m as predicted by the modulation hypothesis (sidebands at $f_{carrier} \pm f_m$). We note that these measurements were conducted in an indoor office environment, in the presence of measurement instruments, LCD monitors, mobile phones, WiFi routers, etc. that all create interference at various frequencies. While this can be a problem for measurements using the traditional electromagnetic side channel, where some of the interference may be in the same frequency bands in which the chip produces side-channel emanations, with the backscattering side channel such interference can be avoided by selecting $f_{carrier}$ such that no strong interference is present in a wide frequency band around it. Finally, please note that signal we are injecting into the board is well below levels that may cause faults (whether transient or permanent) on the FPGA chip or elsewhere on the board.

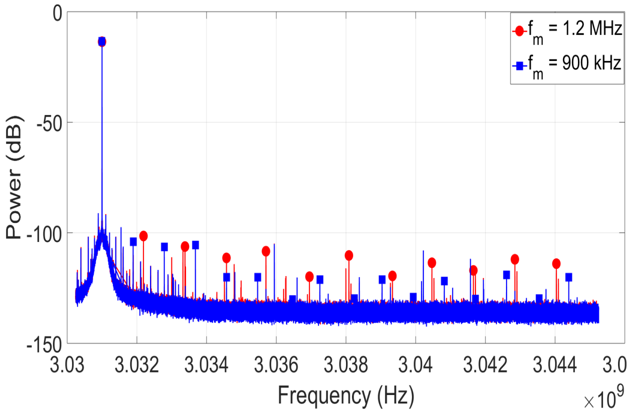


Fig. 6: Measured backscattered power with $f_{carrier}$=3.031 GHz and f_m=900 kHz (blue), 1.2 MHz (red), respectively.

A. Hardware Trojan Detection Using The New Backscattering Side Channel

Switching in digital circuits causes internal impedances to vary, which causes changes in the circuit's radar cross-section (RCS), and thus modulates the carrier wave that is backscattered by the circuit. This new side channel is impedance-based, so it can be beneficial to detection of HTs because the HTs added circuitry, and also the additional connections attached to existing circuitry, result in modifications to the chip's RCS and in how that RCS changes as the on-chip circuits switch. Note that although the HT's trigger tends to be small, it exhibits switching activity as its logic reacts to inputs from the original circuitry, and it adds connections to the chip's original circuitry to obtain those inputs.

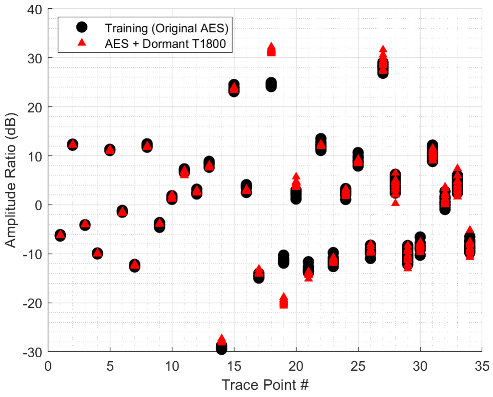


Fig. 7: Amplitude ratios for HT-free and HT-afflicted AES.

Most digital logic circuits are synchronous, so the overall switching pattern follows the clock cycle. Furthermore, the clock cycle usually accommodates switching delays along entire paths of logic gates, which means that the impedance changes of individual gates occur abruptly at some point in the clock cycle, i.e., they have a square-wave-like waveform. This implies that the backscattered signal will contain side-band components for several harmonics of the circuit's clock frequency f_C. These side-band components will be at $f_{carrier} \pm f_C$, $f_{carrier} \pm 2f_C$, $f_{carrier} \pm 3f_C$, etc., and the components at $f_{carrier} \pm f_C$ (that correspond to the first harmonic of the clock frequency) will mostly follow the overall RCS change during a cycle, while the components for the remaining harmonics will be influenced by the rapidity (rise/fall times) and timing of the impedance changes within the clock cycle.

Therefore, our detection of HTs using the backscattering side channel will rely on measuring the amplitude of the backscattered signal at $f_{carrier} \pm f_C$, $f_{carrier} \pm 2 * f_C$, ..., $f_{carrier} \pm m * f_C$, i.e. the side-bands for the first m harmonics of the clock frequency. We use only the amplitude (i.e. we ignore the signal's phase and other properties), mainly because the amplitude at some desired frequency is relatively easy to measure, whereas the phase and other properties require much more sophisticated tuning, phase tracking, etc. Furthermore, we note that each clock harmonic produces two side-band components that have the same amplitude, so the measurement can be made more efficient by only measuring m points to the left, or m points to the right, of $f_{carrier}$. In this paper we measure points to the right of the carrier, i.e. $f_{carrier} + f_C$, $f_{carrier} + 2f_C$, etc.

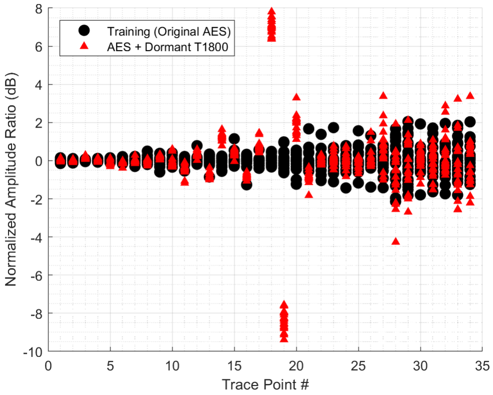


Fig. 8: Amplitude ratios for HT-free and HT-afflicted AES, with each point normalized to the mean of its HT-free measurements.

We call the m amplitudes measured for a given circuit a *trace*, and each trace characterizes the circuit's overall amount, timing, and duration of impedance-change activity during a clock cycle. Intuitively, HTs can then be detected by first collecting training traces, using one or more ICs that are known to be HT-free, and then HT detection on other ICs would consist of collecting their traces and checking if they are too different from the traces learned in training.

However, the amplitude of a received signal declines rapidly with distance. Our measurements are performed close to the chip, so even small variations in positioning of the probes create significant amplitude changes, and would result in numerous false positives when training and detection are not using identical probe positioning (which is very hard to achieve in practice).

Fortunately, the distance affects all of the points in a trace similarly, i.e. distance attenuates all amplitudes in the trace by the same multiplicative factor. Therefore, rather than using amplitudes for trace comparisons, we use amplitude ratios, i.e. amplitude of a harmonic divided by the amplitude of the previous harmonic[3], which cancels out the trace's distance-dependent attenuation factor. The resulting $m-1$ amplitude ratios are then used for comparing traces.

To illustrate amplitude ratios and how they are affected by differences in the tests circuit, Fig. 7 shows the statistics (mean and standard-deviation error bars) of each amplitude-ratio point, for a genuine AES circuit [31], and for the same AES circuit to which the T1800 Trojan from TrustHub [39] has been added but remains inactive throughout the measurement. In this experiment the carrier frequency is $f_{carrier}$=3.031 GHz, the AES circuit is clocked at f_C=20 MHz, and amplitudes for $m = 35$ right-side-band harmonics are measured to obtain the 34 amplitude ratios shown in Fig. 7.

We observe that different amplitude-ratio points for the same trace vary significantly, from -30dB to 35dB in Fig. 7, and that different measurements for the same amplitude-ratio point tend to vary much less than that, making these differences difficult to see in Fig. 7, except for the very large differences between the HT-free and HT-afflicted design at the 18th and 19th amplitude ratio. This indicates that the impedance change is very small and the differences can be observed only at higher harmonics of the clock.

To more clearly show the differences at other harmonic-ratio points, Fig. 8 shows amplitude-ratio points that have been normalized to the mean amplitude ratio for the genuine AES circuit, i.e. for each amplitude ratio the logarithmic-scale points are shifted such that the genuine AES circuit's mean amplitude ratio becomes zero. It can now be observed that, in addition to the 18th and 19th point, which exhibit very large differences between the HT-free and the HT-afflicted measurements, the two circuits differ significantly in a number of other points, e.g. measurements for the two circuits are fully separable using the 14th point or the 20th point, and numerous other points have very little overlap between the HT-free and the HT-afflicted sets of measurements.

From Fig. 8, it can also be observed that the variance among measurements for the same design tends to increase with the index of the amplitude-ratio point, i.e. for points that correspond to higher harmonics.

The primary cause of this increased variance is that higher harmonics of the signal tend to have lower amplitude, which makes their measurement less resilient to noise. Another factor that helps explain this increase in variance among higher harmonics is that they are affected by very small differences in timing of impedance changes during the clock cycle, and factors such as temperature and power supply voltage fluctuation can create small changes in the switching speed of the gates, and thus in the timing of the resulting impedance changes.

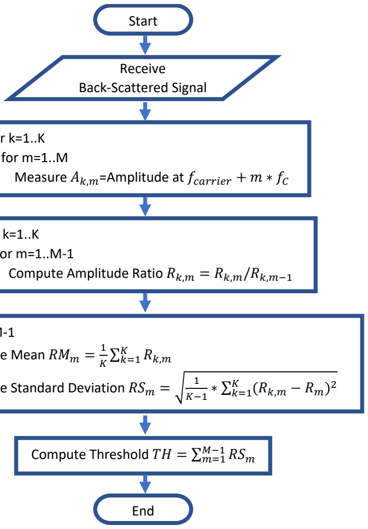

Start
Receive
Back-Scattered Signal
for k=1..K
for m=1..M
Measure $A_{k,m}$=Amplitude at $f_{carrier} + m * f_C$
for k=1..K
for m=1..M-1
Compute Amplitude Ratio $R_{k,m} = R_{k,m}/R_{k,m-1}$
for m=1..M-1
Compute Mean $RM_m = \frac{1}{K}\sum_{k=1}^{K} R_{k,m}$
Compute Standard Deviation $RS_m = \sqrt{\frac{1}{K-1} * \sum_{k=1}^{K}(R_{k,m} - R_m)^2}$
Compute Threshold $TH = \sum_{m=1}^{M-1} RS_m$
End

Fig. 9: Training algorithm.

Regardless of the reason for the increasing variance among measurements of higher harmonics, the fact that the variance does increase is an important motivation for using an impedance-based side channel rather than one created by bursts of current. Specifically, for each gate that switches, the impedance change persists for the rest of the cycle, while the burst of current is very brief in duration. This means that the impedance-change contributes to lower frequencies than the current-burst signal. When activity from cycle to cycle is repetitive, the spectrum of the signal's within-a-cycle waveform is projected onto the harmonics of the clock frequency, so gate-switching activity tends to affect lower harmonics of the clock frequency in impedance-based than in current-burst based side channels. As lower harmonics tend to have less variance from measurement to measurement, impedance-based side channels can be expected to perform better for HT detection than current-burst based side channels, and our results in Section V-C confirm that.

B. HT Detection Algorithm

Our HT detection algorithm has two phases: *training*, where a circuit that is known to be HT-free is characterized, and *detection*, where an unknown circuit is classified into one of the two categories – HT-free or HT-afflicted, according to how much its measurements deviate from the statistics learned in training.

1) Training: Fig. 9 details the training for the prototype implementation of backscattering-based HT detection. This training consists of measuring K times the signal backscattered from an IC known to be HT-free, each time collecting the m amplitudes at frequencies that correspond to the lowest m harmonics of the IC's clock frequency in the side-band of the received backscattered signal. The $m-1$ amplitude ratios are then computed from these amplitudes.

Next, for each of the $m-1$ amplitude ratios, the mean and standard deviation across the M measurements are computed, and the detection threshold for HT detection is computed as the sum of the $m-1$ standard deviations.

[3] Measurement of signal amplitude are often expressed in decibels, i.e. on a logarithmic scale, and for these measurements subtraction of logarithmic-scale amplitude values yields the logarithmic-scale value for the amplitude ratio

2) Detection: Figure 10 details how the prototype implementation of backscattering detection decides whether to classify an IC as HT-free of HT-afflicted. First, a single measurement is obtained of the m amplitudes that correspond to the lowest m harmonics of the IC's clock frequency in the side-band of the signal that is backscattered from the IC under test, and $m-1$ amplitude ratios are computed from these amplitudes.

Next, for each of the $m-1$ amplitude ratios, we compute how much it deviates from the corresponding mean computed during training. This deviation is computed as the absolute value of the difference, and intuitively it measures how much that amplitude ratio differs from what would be expected from an HT-free IC. Finally, this sum of these deviations is compared to the sum of standard deviations from training. Intuitively, the sum of the differences for the IC under test is a measure of how much its overall backscattering "signature" differs from what would be expected from an HT-free IC, and the sum of standard deviations from training corresponds to how much an individual measurement of an HT-free IC can be expected to differ from the average of HT-free measurements. The IC under test is labeled as HT-free if its sum of amplitude-ratio deviations is lower than this detection threshold (sum of standard deviations from training).

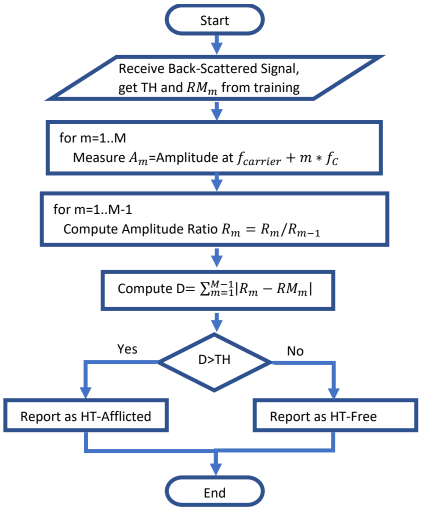

Start
Receive Back-Scattered Signal, get TH and RM_m from training
for m=1..M
Measure A_m=Amplitude at $f_{carrier} + m * f_C$
for m=1..M-1
Compute Amplitude Ratio $R_m = R_m/R_{m-1}$
Compute D= $\sum_{m=1}^{M-1} |R_m - RM_m|$
Yes
D>TH
No
Report as HT-Afflicted
Report as HT-Free
End

Fig. 10: Detection algorithm.

IV. Experimental Setup

A. Backscattering Side Channel Measurement Setup

Figure 11 shows the measurement setup that we use to evaluate the performance of the proposed prototype backscattering-based HT-detection. The carrier signal is a sinusoid at $f_{carrier}$=3.031 GHz produced by an Agilent MXG N5183A signal generator and transmitted toward the FPGA chip using an Aaronia E1 electric-field near-field probe. To select $f_{carrier}$, we have measured signal strength at the frequency of the reflected carrier signal (the signal we were injecting into the board), the first several harmonics of the modulated FPGA board clock (e.g. 50 MHz away from the carrier), and of the noise floor of the instrument using AARONIA Near Field Probes (0 to 10 GHz). We have found that the side-band signal for the first harmonic of the board's clock is strongest when $f_{carrier}$ is around 3 GHz, but we also found that traditional EM emanations create interference at frequencies that are multiples of the board's clock frequency (50MHz). Thus we choose $f_{carrier}$=3.031 GHz, a frequency close to 3GHz that avoids interference from the board's traditional EM emanation. The device-under-test (DuT) is the FPGA chip on the Altera DE0-CV board, and it is positioned using a right-angle ruler so that different DE0-CV boards can be tested using approximately the same position of probes. The backscattered signal is received with an Aaronia H2 magnetic field near-field probe, and this signal pre-amplified using an EMC PBS2 low-noise amplifier and then the signal amplitudes at desired frequencies are measured using an Agilent MXA N9020A Vector Signal Analyzer.

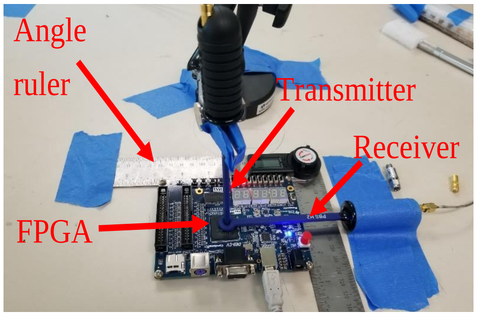


Fig. 11: Measurement setup for hardware Trojan detection using back-scattering side channel.

B. Training and Testing Subject Circuit Designs

All circuits used in our experiments are implemented on a Field Programmable Gate Array (FPGA), which allows rapid experimentation by changing the circuit and/or its physical placement and routing, unlike hard-wired ASIC designs that would require fabrication for each layout variant of each circuit. The specific FPGA board we use is the Altera DE0-CV board, and within it the IC on which our backscattering measurement setup focuses is the Altera 5CEBA4F23C7N, an FPGA in Altera's Cyclone V device family.

For our HT detection experiments, we use AES-T1800, AES-T1600, and AES-T1100 hardware Trojan benchmarks from TrustHub [39]. For all three of these HTs, the original HT-free design is an AES-128 cryptographic processor, which uses an 11-stage pipeline to perform the 10 stages of AES encryption on 128-bit block. Since numerous HTs in the TrustHub repository are similar to each other, we selected these three HT benchmarks because they exhibit different approaches for their triggers and payloads:

- T1800: The payload in this HT is a cyclic shift register that, upon activation, continuously shifts to increase power drain consumption, which would be a serious problem for small battery-powered or energy-harvesting devices in e.g., medical implants. The HT's trigger circuit consists of combinatorial logic that monitors the 128-bit input of the AES circuit, looking for a specific 128-bit plaintext value, and the occurrence of that 128-bit value at the input activates the payload. The size of T1800's trigger circuit is 0.27% of the original AES circuit, and the size of its payload is 1.51% of the size of the AES circuit. Because this HT's trigger and payload can be resized easily, we use this HT to study how our HT detection is affected by HT size and physical location.

- T1600: The payload in this HT creates activity on an otherwise-unused pin to generate an RF signal that leaks the key of the AES circuit. The HT's trigger circuit consists of sequential logic which activates the payload when a predefined *sequence* of values is detected at input of the AES circuit. The size of T1600's trigger circuit is 0.28% of the size of the original AES circuit, while the size of its payload is 1.76% of the size of the original AES circuit.
- T1100: The payload of this HT modulates its activity using a spread-spectrum technique to create a power consumption pattern that leaks the AES key. The trigger is a (sequential) circuit that looks for a predefined sequence of values at the input of the AES circuit to activate the payload. The size of T1800's trigger circuit is 0.28% of the size of the original AES circuit, while the size of its payload is 1.61% of the size of the AES circuit.

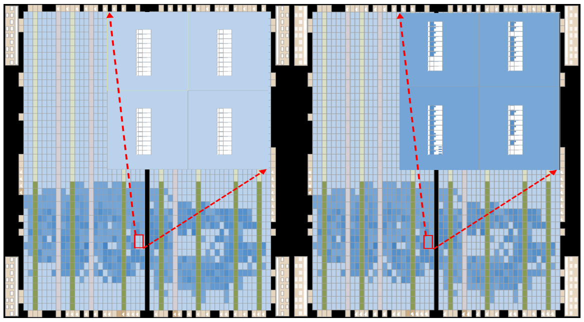
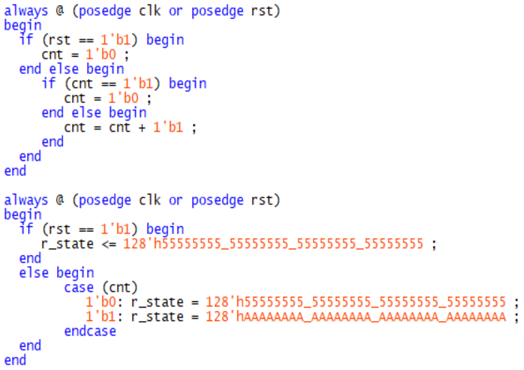


Fig. 12: (a) Genuine AES circuit (b) Hardware Trojan infected AES circuit.

A key challenge we faced when implementing the HT-afflicted circuits was that these HTs are specified at the register-transfer level, as modifications to the original AES circuit's Verilog HDL source code. If the modified source code is subjected to the normal compilation, placement, and routing, we found that the addition of the HT causes the EDA tool to change the placement and routing of most logic elements in the overall circuit, and this extensive change makes the modification very easy to detect regardless of the HT's actual size and activity. The next approach we tried was to compile the AES circuit using the normal compilation, placement, and routing, and then for each HT-afflicted design we used the ECO (Engineering Change Order) tool in Altera's Quartus II suite to add the HT's circuitry while leaving unchanged the placement of logic elements (and the routing of their connections) that belong to the original AES circuit. However, we found that this approach makes it very hard to place the HT's logic elements close to the inputs of the original AES circuit, and (as will be demonstrated in Section V-E), the HT is easier to detect when its trigger is placed away from where it is connected to the original circuit. To make the HTs more stealthy, we instead compile, place, and route the *HT-afflicted* circuit, then create the HT-free circuit by removing (using the ECO tool) the HT's logic elements and their connections. This models the HT "dream scenario" for the malicious entity that wishes to insert the HT, as there is just enough space in the HT-free layout to insert the HT in just the right place to have very short connections to the original circuit. To illustrate this, the placement of the HT-free circuit and the T1800-afflicted circuit are shown in Fig. 12, with a zoom-in to show the details where the HT's logic elements are placed.

Finally, for HT detection, the circuit must be supplied with inputs during the evaluation. Since we evaluate our HT detection approach in the dormant-HT scenario, any input sequence that causes logic gates in the original AES circuit to change state can be used, so each cycle we simply flip all of the AES circuit's input bits, as shown in Fig. 13.[4]



```
always @ (posedge clk or posedge rst)
begin
  if (rst == 1'b1) begin
     cnt = 1'b0 ;
  end else begin
     if (cnt == 1'b1) begin
        cnt = 1'b0 ;
     end else begin
        cnt = cnt + 1'b1 ;
     end
  end
end

always @ (posedge clk or posedge rst)
begin
  if (rst == 1'b1) begin
     r_state <= 128'h55555555_55555555_55555555_55555555 ;
  end
  else begin
        case (cnt)
           1'b0: r_state = 128'h55555555_55555555_55555555_55555555 ;
           1'b1: r_state = 128'hAAAAAAAA_AAAAAAAA_AAAAAAAA_AAAAAAAA ;
        endcase
  end
end
```

Fig. 13: Feeding inputs to the AES circuit.

V. Evaluation

Because it is very difficult to activate an HT without a priori knowledge of its trigger conditions, it is highly desirable for an HT detection scheme to provide accurate detection of *dormant* HTs, i.e., to detect HTs whose payload is never activated while it is characterized by the HT detection scheme. However, a dormant HT is typically more difficult to detect compared to an activated HT. For side channel-based detection methods, in particular, the switching activity in the activated payload, and/or the changes it creates in the switching activity of the original circuit, have more impact on the side channel signal than an inert payload (no switching activity in the payload and no changes to the original circuit's functionality).

Another important practical concern for HT detection is robustness to manufacturing variations and other differences between different physical instances of the same hardware design. Thus our evaluation focuses on detection of *dormant* HTs with *cross-training*, i.e. training for HT detection is performed on one hardware instance, and then HT detection is performed on others.

Our experimental results (Section V-A) show that our prototype backscattering-based HT detection, after training with an HT-free design on one DE0-CV board, accurately reports the presence of dormant HTs, for each of three different HT designs, on nine other DE0-CV boards, while having no false positives when the HT-free design is used on those nine other DE0-CV boards.

[4] Note that hexadecimal 3 and C correspond to binary 0011 and 1100, while hexadecimal A and 5 correspond to 1010 and 0101, respectively. Thus the inputs we feed to the AES circuit simply toggle each of the input bits, while avoiding all-ones and all-zeros patterns.

Next, we perform additional experiments to experimentally confirm that dormant HTs are indeed more difficult to detect than activated ones (Section V-B), and also to confirm that a similar detection approach with the traditional EM side channel would still be able to detect activated HTs, but would be unreliable for detection of dormant HTs (Section V-C). Finally, we experimentally evaluate how the accuracy of dormant-HT detection changes when changing the size (Section V-D) and physical placement (Section V-E) of the hardware Trojan's trigger and payload components.

A. Dormant-HT Detection with Cross-Training Using the Backscattering Side Channel Signal

We evaluate the effectiveness of our HT detection prototype by training it on one DE0-CV FPGA board with an HT-free AES circuit, then applying HT detection to several test subject circuits implemented on nine DE0-CV FPGA boards, none of which is the same as the one used for training.

The test subject designs are:

- *Original AES*. This is the same HT-free AES circuit that was used in training, and we use it to measure the false positive rate of our HT detection,
- *AES + Dormant T1800*. This is the same AES circuit, with the same placement and routing, that was used for training, but with additional logic elements and connections that implement the AES-T1800 Trojan from TrustHub. The size of this HT's trigger (in FPGA logic elements) is 0.27% of the original AES circuit, and we use a payload that was reduced to only 0.03% of the original AES circuit. The reduced payload size helps fit this HT closer to where its input signals can be connected to the original AES circuit, making the HT significantly more difficult to detect (as will be shown in Section V-E).
- *AES + Dormant T1600*. This is the same AES circuit, with the same placement and routing, that was used for training, but with additional logic elements and connections that implement the AES-T1600 Trojan from TrustHub. The size of this HT's trigger is 0.28% of the original AES circuit, while its payload's size is 1.76% of the original AES circuit.
- *AES + Dormant T1100*. This is the same AES circuit, with the same placement and routing, that was used for training, but with additional logic elements and connections that implement the AES-T1100 Trojan from TrustHub. The size of this HT's trigger is 0.28% of the original AES circuit, while its payload's size is 1.61% of the original AES circuit.

For each measurement, the previously measured FPGA board is removed from the measurement setup, and then a different board is positioned using an angle ruler to model a realistic measurement scenario when each measurement uses a very similar but not identical relative position of the chip and the probes. Each test subject design is measured 20 times on each board, and each measurement is used for HT detection in isolation, i.e. for each test subject the detection makes 20 classification decisions (HT-free or HT-afflicted) on each of the 9 boards, resulting in a total of 720 decisions. Among these decisions, 180 were on the *Original AES* test subject, and in all 180 of them our prototype has correctly classified the design as HT-free, i.e., the HT detection prototype had no false-positive detections. In the remaining 3 sets of 180 decisions, each test subjects design was HT-afflicted (180 decisions with T1800, 180 decisions with T1600, and 180 with T1100), and in all of them our prototype has correctly classified the design as HT-afflicted, i.e. the HT detection prototype has detected the presence of an HT in each measurement in which an HT was present.

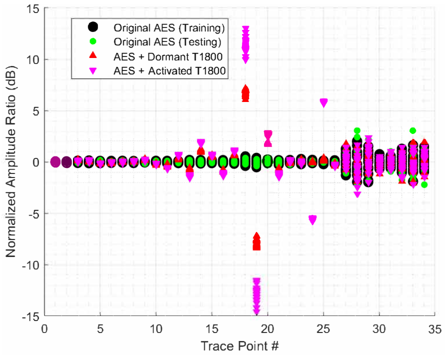


Fig. 14: Normalized amplitude ratios for backscattering side channel measurements.

Since our HT detection prototype using the backscattering side channel **achieves 100% detection of three kinds of dormant HTs, with 0% false positives, in the cross-training measurement scenario**, we focus the rest of our experimental evaluation on getting more insight into why our HT detection performs so well and how sensitive it is to changes in the position and size of the HT.

B. HT Detection of Dormant vs. Active HTs Using the Backscattering Side Channel

Figure 14 compares the normalized amplitude ratios for an HT-free AES design and for the same AES design (and layout) to which the AES-T1800 Trojan has been added. Two separate sets of 20 measurements are shown for the HT-free design, one that is used for training and one that is used to detect false positives when evaluating HT detection (on another DE0-CV board). For the HT-afflicted design, one set of 20 measurements is collected when the HT is dormant (its payload has not been activated), and another set of 20 measurements is collected with the same HT after its payload is activated.

We can observe that there are a number of trace points where both sets of HT-afflicted measurements deviate significantly from HT-free measurements, and that this deviation tends to be larger for measurements in which the HT has been activated. The higher deviation from HT-free measurements seen for active-HT measurements agrees with the intuitive reasoning that an HT is easier to detect when active then when it is dormant. Even so, our backscattering-based HT detection prototype successfully reports the existence in each dormant-HT experiment (100% detection rate), while correctly reporting all 20 HT-free measurements as HT-free (no false positives).

C. Comparison to EM-based HT Detection

As discussed in Section III, the impedance-based backscattering side channel should be more effective for HT detection than existing current-burst-based (e.g. traditional EM) side channels. To confirm this, we repeat the same experiment, but this time use amplitudes of EM emanations at the clock frequency and its harmonics, instead of using the clock-frequency harmonics in the side-bands of the backscattered signal. The normalized amplitude ratios from these measurements are shown in Fig. 15. We can observe that the HT-afflicted measurements are much less separated from HT-free ones than they were with backscattering – for most trace points even active-HT measurements are all within ±1dB from the HT-free ones, although for several trace points there is still some separation between the active-HT and HT-free measurements. More importantly, nearly all dormant-HT measurements have a lot of overlap with HT-free measurements, which makes the dormant-HT measurements difficult to distinguish from HT-free ones.

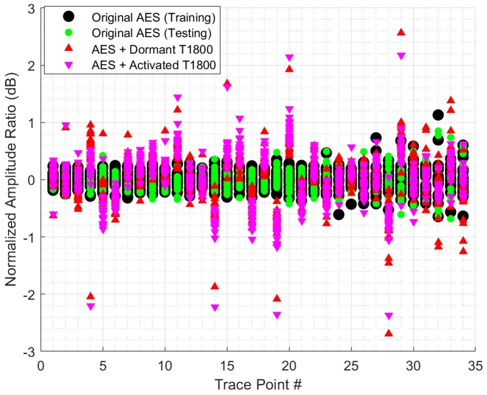


Fig. 15: Normalized amplitude ratios for traditional electromagnetic side channel measurements.

This is confirmed by the results of applying our HT detection prototype to these measurements. The ROC (Receiver Operating Characteristic) curves for HT detection using backscattering and EM side channels are shown in Fig. 16. Backscattering-based detection correctly identifies the presence of an HT in each HT-afflicted measurement, without false positives in HT-free measurements, in both active-HT and dormant-HT scenarios. In contrast, detection based on the EM side channel performs less well in the active-HT case, reporting only 70% of the active-HT measurements as HT-afflicted using the default threshold (which produces no false positives). More importantly, EM-based detection in the dormant-HT case performs poorly – in the absence of false positives, only 15% of the dormant-HT measurements are correctly reported as HT-afflicted, and when the detection threshold is reduced to a point where all dormant-HT measurements are reported as HT-afflicted, 50% of the HT-free measurements are also reported as HT-afflicted (a 50% false-positive rate).

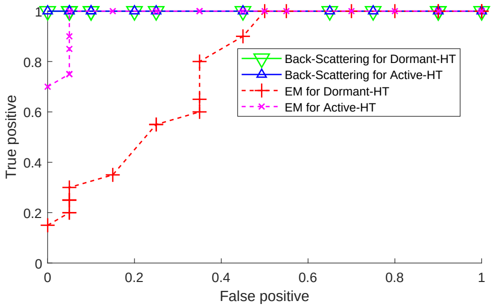


Fig. 16: Detection performance (ROC curve) comparison of backscattering-based and EM-based detection in active-HT and dormant-HT scenarios.

In conclusion, these experiments indicate that our HT detection technique's ability to detect dormant HTs comes, at least in large part, from using the backscattering (impedance-based) side channel instead of traditional current-burst-based (EM and power) side channels.

D. Impact of Hardware Trojan Trigger and Payload Size

To provide more insight into which factors influence our HT detection prototype's ability to detect dormant HTs, we perform experiments in which we reduce the size of the T1800 hardware Trojan's trigger and payload. The T1800 was chosen because it has the smallest trigger among the HTs we used in our experiments, and because both its payload and its trigger can be meaningfully resized.

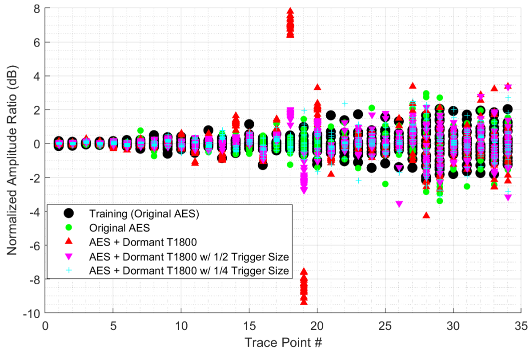


Fig. 17: Normalized amplitude ratios for different sizes of T1800's trigger input.

The T1800 monitors the 128-bit data input of the AES-128 circuit, comparing it to a specific hard-wired 128-bit value, and it activates the payload when that 128-bit value is detected. In terms of logic elements (gates), the size of this 128-bit trigger is only 0.27% of the size of the original AES circuit, i.e. even this full-size trigger is much smaller than the AES circuit to which the HT has been added, and its activity (while the HT is dormant) is difficult to detect using existing side channels. We implement reduced-trigger variants of this HT by monitoring only the 64 least significant bits (the "1/2 Trigger Size" variant, where the trigger circuit size is only 0.15% of the original AES circuit's size), and then only the 32 least significant bits (the "1/4 Trigger Size" variant, where the trigger circuit size is only 0.08% of the original AES circuit size). The normalized harmonic ratio traces for 20 measurements of each design, along with 40 HT-free measurements (20 for training and 20 for false-positives testing) are shown in Fig. 17. We observe that smaller trigger sizes result in trace points that are closer to HT-free ones, i.e. that trigger size directly impacts the side-channel-based separation between dormant-HT and HT-free circuits. These results match the intuition that the HT's influence on impedance changes should increase as more input bits are monitored by the HT's trigger, both because of the increased number of connections to the original circuit (which can change impedances "seen" by gates that belong to the original circuit) and because of the increased number of gates whose values can change (switching activity) within a cycle in the HT's trigger circuit itself.

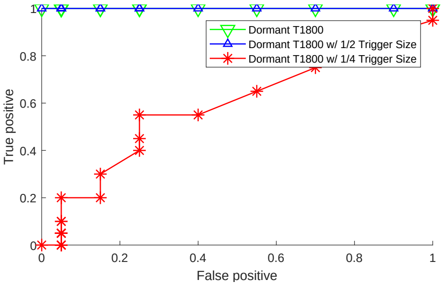


Fig. 18: ROC curves for HT detection for different sizes of the HT's trigger circuit.

The ROC curves for HT detection with different trigger sizes (Fig. 18) confirm that, while the HT with the original-size and even 1/2-size trigger can be detected in each measurement with no false positive, the detection accuracy suffers significantly as the HT's trigger is further reduced to 1/4 of the original size.

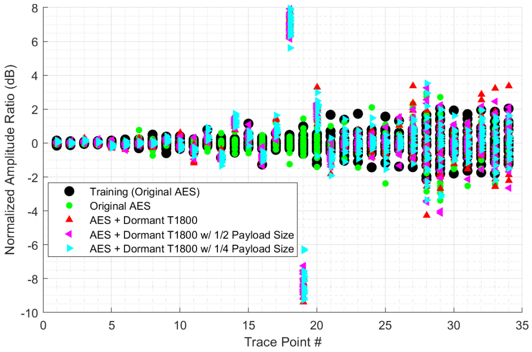


Fig. 19: Normalized amplitude ratios for different sizes of T1800's (dormant) payload.

We perform additional experiments in which we keep the trigger at full size, but reduce the size of the payload to 50% and then 25%. Our dormant-HT measurement results for these variants are not noticeably different from each other (Fig. 19), which implies that the payload size has little impact on our HT detection. This agrees with our theoretical and intuitive expectations: the payload in T1800 has little impact on the impedance changes during a clock cycle, as it has no switching activity (until activated), and has no connections to the gates in the original AES circuit (T1800's payload is designed to produce a lot of power-draining switching activity upon activation, not to change the functionality of the AES circuit).

Since the measurements of the full-trigger-and-reduced-payload variants of T1800 HT are very similar to the full-size T1800 HT, they provide the same ROC curves (complete detection without false positives) as the full-size T1800 HT, as shown in Fig. 18.

E. Impact of HT Trigger and Payload Position

We next investigate how the backscattering-based HT detection is influenced by the physical location and routing of the HT's connection to the original circuit. For this, we start with the AES circuit with the T1800 HT, whose trigger logic was placed at Position 1 shown in Fig. 20 by the placement and routing tool very close to where its 128-bit input can be connected to the original AES circuit.

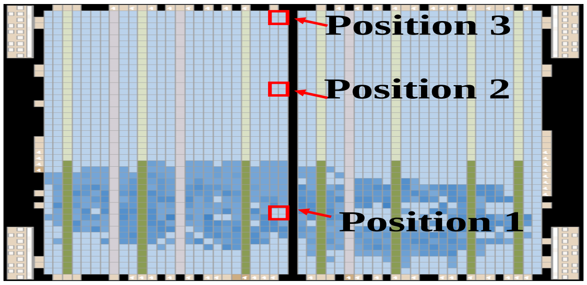


Fig. 20: Changing the physical position off the HT's trigger logic.

We then create a variant of this HT by moving the HT's trigger logic to Position 2, keeping the logic elements and the connections between them in the same position relative to each other, but making the trigger's 128 connections to the original AES circuit much longer. Another variant is similarly created by moving the HT's trigger logic to Position 3.

The dormant-HT measurement results for these three positions are shown in Fig. 21. We observe that, at many trace points, in terms of separation of HT-afflicted measurements HT-free ones, Position 2 is significantly more separated than Position 1, and Position 3 provides an additional small increase in separation. This means that HTs placed close to their connection points in the original circuit are more difficult to detect than HTs that require long connections. All of our prior experiments used HTs that were placed by the placement and routing tool in a way that attempts to minimize overall cost (which tends to minimize the total length of the HT's connections to the original circuit), we can thus expect the Position 2 and Position 3 variants to also be detected correctly in each dormant-HT measurement (with no false positives in HT-free measurements), and our HT detection results confirm this.

We also performed experiments in which the trigger part of the HT is kept in Position 1, while its payload was moved to Position 2 and then Position 3. Our results show that the payload position has little impact on the measurements, which is as expected given that, in our dormant-HT experiments, the 1-bit "activate" signal between the trigger and the payload

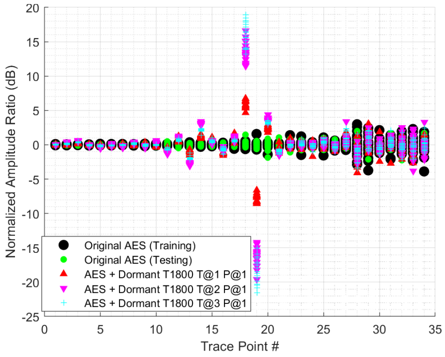


Fig. 21: Normalized amplitude ratios for different locations of T1800's trigger logic.

never changes its value (it stays at 0, i.e. inactive), and that the payload has no switching activity.

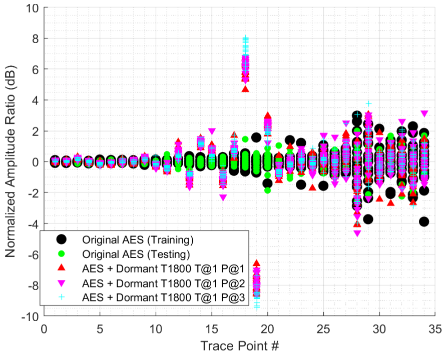


Fig. 22: Normalized amplitude ratios for different locations of T1800's (dormant) payload.

F. Further Evaluation of HT Detection Using More Benchmarks

To further evaluate the effectiveness of our HT detection prototype, we implement two different circuits, RS232 and PIC16F84, each with three HTs, from TrustHub [39]. We use the same HT detection prototype described in Section III-B and the setup described in Section IV.

1) RS232 circuit: We use RS232-T500, RS232-T600, and RS232-T700 HT benchmarks from TrustHub [39]. For all three of these HTs, the original HT-free design is a RS232 micro-UART core consisting of a transmitter and a receiver. The transmitter takes input words (128-bit length) and serially outputs each word according to the RS232 standard, while the receiver takes a serial input and output 128-bit words.

- RS232-T500: The payload in this HT is a circuit that, upon activation, causes the transmission to fail. The trigger is sequential circuit that increments its counter every clock cycle, and activates the payload activated when this counter reaches a certain value. The size of the trigger circuit is 1.67%, and the size of the payload circuit is 1.48% of the size of the RS232 circuit.
- RS232-T600: The payload in this HT is a circuit that, upon activation, makes the transmitter's "ready" signal become stuck-at-1, and changes specific bits in the transmitted data. The trigger is a sequential circuit that looks for a specific sequence of UART states to activate the payload. The size of the trigger circuit is 1.54%, and the size of the payload circuit is 1.52% of the size of the RS232 circuit.
- RS232-T700: The payload of this HT is a circuit that, upon activation, makes the transmitter's "finished" signal become stuck-at-0. The trigger is sequential circuit that looks for a predefined sequence of UART states to activate. The size of the trigger circuit is 1.54%, and the size of the payload circuit is 1.48% of the size of the RS232 circuit.

The results in Figs. 23 and 24 show the ratios of harmonics and ROC curve, respectively. The results show that we can detect each of these three Trojans with 100% accuracy and 0% false positives.

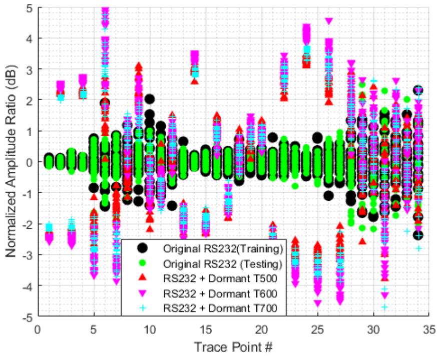


Fig. 23: Normalized amplitude ratios for different HTs in the RS232 circuit.

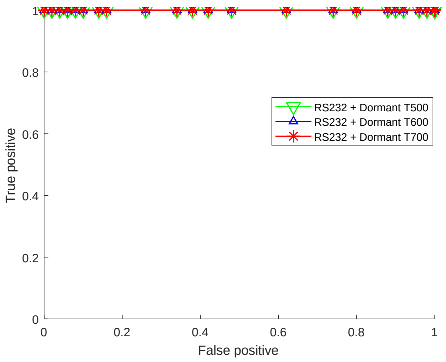


Fig. 24: ROC curves for detection of HTs in the RS232 circuit.

2) PIC16F84 circuit: We use PIC16F84-T100, PIC16F84-T200, and PIC16F84-T400 hardware Trojan benchmarks from TrustHub [39]. For all three HTs, the original HT-free design is PIC16F84 circuit, a RISC micro-controller whose functions and instruction set are very similar to those of the Microchip 16F84 chip.

- PIC16F84-T100: Once activated by its (sequential) trigger circuit, the payload changes the address to PIC16F84's program memory (causing denial of service). The size of the trigger circuit is 1.34%, while the size of the payload circuit is 1.81% of the size of the PIC16F84 circuit.
- PIC16F84-T200: Once activated by its (sequential) trigger circuit, the payload in this HT replaces the instruction register with a sleep command (causing denial of service). The size of the trigger circuit is 1.35%, and the size of the payload circuit is 1.93% of the size of the PIC16F84 circuit.
- PIC16F84-T400: Once activated by its (sequential) trigger circuit, the payload of this HT changes the address lines to the external EEPROM to 0 (causing denial of service). The size of the trigger circuit is 1.35%, while the size of the payload circuit is 1.75% of the size of the PIC16F84 circuit.

The results in Figs. 25 and 26 show the ratios of harmonics and ROC curve, respectively. The results show that we can detect each of these three Trojans with 100% accuracy and 0% false positives.

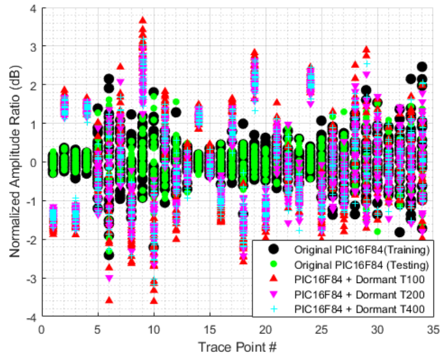


Fig. 25: Normalized amplitude ratios for different Trojans on PIC16F84 circuit.

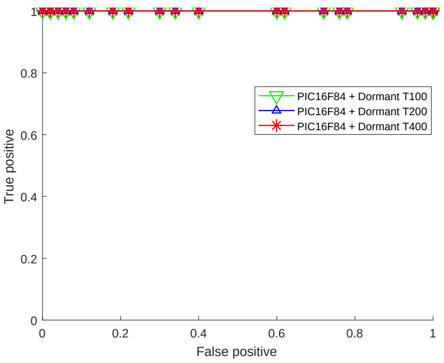


Fig. 26: ROC curves for different Trojans on PIC16F84 circuit.

VI. Conclusion and Future Directions

This paper describes a new physical side channel, i.e. the backscattering side channel, that is created by transmitting a signal toward the IC, where the internal impedance changes caused by on-chip switching activity modulate the signal that is backscattered (reflected) from the IC. To demonstrate how this new side-channel can be used to detect small changes in circuit impedances, we propose a new method for non-destructively detecting HTs from outside of the chip. We experimentally confirm, using measurements on one physical instance for training and nine other physical instances for testing, that the new side-channel, when combined with an HT detection method, allows detection of a *dormant* HT in 100% of the HT-afflicted measurements for a number of different HTs, while producing no false positives in HT-free measurements. Furthermore, additional experiments are conducted to compare the backscattering-based detection to one that uses the traditional EM-emanation-based side channel. These results show that backscattering-based detection outperforms the EM side channel, confirm that dormant HTs are much more difficult for detection than HTs that have been activated, and show how detection is affected by changing the HT's size and physical location on the IC.

This paper presents preliminary results on using a new physical side channel for HT detection. As a part of our future work, we plan to do more detailed testing on ASIC hardware, design specialized probes and use probe station to enhance spatial resolution, and develop new techniques that do not rely on golden example.

References

[1] K. Xiao, D. Forte, Y. Jin, R. Karri, S. Bhunia, and M. Tehranipoor, "Hardware trojans: Lessons learned after one decade of research," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 22, no. 1, p. 6, 2016.

[2] W. K. Clark and P. L. Levin, "Securing the information highway," *Foreign Aff.*, vol. 88, p. 2, 2009.

[3] J. Villasenor, *Compromised by design?: Securing the defense electronics supply chain*. Center for Technology Innovation at Brookings, 2013.

[4] ——, "The hacker in your hardware," *Scientific American*, vol. 303, no. 2, pp. 82–87, 2010.

[5] L.-W. Kim, J. D. Villasenor *et al.*, "A trojan-resistant system-on-chip bus architecture," in *Military Communications Conference, 2009. MILCOM 2009. IEEE*. IEEE, 2009, pp. 1–6.

[6] Q. Yu and J. Frey, "Exploiting error control approaches for hardware trojans on network-on-chip links," in *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2013 IEEE International Symposium on*. IEEE, 2013, pp. 266–271.

[7] D. McIntyre, F. Wolff, C. Papachristou, S. Bhunia, and D. Weyer, "Dynamic evaluation of hardware trust," in *Hardware-Oriented Security and Trust, 2009. HOST'09. IEEE International Workshop on*. IEEE, 2009, pp. 108–111.

[8] L.-W. Kim and J. D. Villasenor, "Dynamic function replacement for system-on-chip security in the presence of hardware-based attacks," *IEEE Transactions on Reliability*, vol. 63, no. 2, pp. 661–675, 2014.

[9] R. Torrance and D. James, "The state-of-the-art in ic reverse engineering," in *Cryptographic Hardware and Embedded Systems-CHES 2009*. Springer, 2009, pp. 363–381.

[10] A. Waksman, M. Suozzo, and S. Sethumadhavan, "Fanci: identification of stealthy malicious logic using boolean functional analysis," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. ACM, 2013, pp. 697–708.

[11] H. Salmani, "Cotd: reference-free hardware trojan detection and recovery based on controllability and observability in gate-level netlist," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 2, pp. 338–350, 2017.

[12] J. Zhang, F. Yuan, L. Wei, Y. Liu, and Q. Xu, "Veritrust: Verification for hardware trust," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 7, pp. 1148–1161, 2015.

[13] M. Tehranipoor and F. Koushanfar, "A survey of hardware trojan taxonomy and detection," *IEEE design & test of computers*, vol. 27, no. 1, 2010.

[14] R. S. Chakraborty, S. Narasimhan, and S. Bhunia, "Hardware trojan: Threats and emerging solutions," in *High Level Design Validation and Test Workshop, 2009. HLDVT 2009. IEEE International*. IEEE, 2009, pp. 166–171.

[15] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, "Trojan detection using ic fingerprinting," in *Security and Privacy, 2007. SP'07. IEEE Symposium on*. IEEE, 2007, pp. 296–310.

[16] M. Banga and M. S. Hsiao, "A region based approach for the identification of hardware trojans," in *Hardware-Oriented Security and Trust, 2008. HOST 2008. IEEE International Workshop on*. IEEE, 2008, pp. 40–47.

[17] ——, "Vitamin: Voltage inversion technique to ascertain malicious insertions in ics," 2009.

[18] C. He, B. Hou, L. Wang, Y. En, and S. Xie, "A failure physics model for hardware trojan detection based on frequency spectrum analysis," in *Reliability Physics Symposium (IRPS), 2015 IEEE International*. IEEE, 2015, pp. PR–1.

[19] S. Narasimhan, D. Du, R. S. Chakraborty, S. Paul, F. Wolff, C. Papachristou, K. Roy, and S. Bhunia, "Multiple-parameter side-channel analysis: A non-invasive hardware trojan detection approach," in *Hardware-Oriented Security and Trust (HOST), 2010 IEEE International Symposium on*. IEEE, 2010, pp. 13–18.

[20] C. Bao, D. Forte, and A. Srivastava, "Temperature tracking: Toward robust run-time detection of hardware trojans," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 10, pp. 1577–1585, 2015.

[21] D. Forte, C. Bao, and A. Srivastava, "Temperature tracking: An innovative run-time approach for hardware trojan detection," in *Proceedings of the International Conference on Computer-Aided Design*. IEEE Press, 2013, pp. 532–539.

[22] J. He, Y. Zhao, X. Guo, and Y. Jin, "Hardware trojan detection through chip-free electromagnetic side-channel statistical analysis," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 10, pp. 2939–2948, 2017.

[23] J. Balasch, B. Gierlichs, and I. Verbauwhede, “Electromagnetic circuit fingerprints for hardware trojan detection,” in *Electromagnetic Compatibility (EMC), 2015 IEEE International Symposium on*. IEEE, 2015, pp. 246–251.

[24] X. T. Ngo, Z. Najm, S. Bhasin, S. Guilley, and J.-L. Danger, “Method taking into account process dispersion to detect hardware trojan horse by side-channel analysis,” *Journal of Cryptographic Engineering*, vol. 6, no. 3, pp. 239–247, 2016.

[25] K. Hu, A. N. Nowroz, S. Reda, and F. Koushanfar, “High-sensitivity hardware trojan detection using multimodal characterization,” in *Proceedings of the Conference on Design, Automation and Test in Europe*. EDA Consortium, 2013, pp. 1271–1276.

[26] A. N. Nowroz, K. Hu, F. Koushanfar, and S. Reda, “Novel techniques for high-sensitivity hardware trojan detection using thermal and power maps,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 33, no. 12, pp. 1792–1805, 2014.

[27] B. Cha and S. K. Gupta, “Efficient trojan detection via calibration of process variations,” in *Test Symposium (ATS), 2012 IEEE 21st Asian*. IEEE, 2012, pp. 355–361.

[28] ——, “Trojan detection via delay measurements: A new approach to select paths and vectors to maximize effectiveness and minimize cost,” in *Proceedings of the conference on design, automation and test in Europe*. EDA Consortium, 2013, pp. 1265–1270.

[29] M. Lecomte, J. Fournier, and P. Maurine, “An on-chip technique to detect hardware trojans and assist counterfeit identification,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 12, pp. 3317–3330, 2017.

[30] P. V. Nikitin and K. S. Rao, “Theory and measurement of backscattering from rfid tags,” *IEEE Antennas and Propagation Magazine*, vol. 48, no. 6, pp. 212–218, 2006.

[31] B. Shakya, T. He, H. Salmani, D. Forte, S. Bhunia, and M. Tehranipoor, “Benchmarking of hardware trojans and maliciously affected circuits,” *Journal of Hardware and Systems Security*, vol. 1, no. 1, pp. 85–102, 2017.

[32] S. Bhunia, M. S. Hsiao, M. Banga, and S. Narasimhan, “Hardware trojan attacks: threat analysis and countermeasures,” *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1229–1247, 2014.

[33] J. Zhang, F. Yuan, and Q. Xu, “Detrust: Defeating hardware trust verification with stealthy implicitly-triggered hardware trojans,” in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014, pp. 153–166.

[34] Z. Chen, X. Guo, R. Nagesh, A. Reddy, M. Gora, and A. Maiti, “Hardware trojan designs on basys fpga board,” *Embedded system challenge contest in cyber security awareness week-CSAW*, 2008.

[35] R. S. Chakraborty, I. Saha, A. Palchaudhuri, and G. K. Naik, “Hardware trojan insertion by direct modification of fpga configuration bitstream,” *IEEE Design & Test*, vol. 30, no. 2, pp. 45–54, 2013.

[36] X. Wang, M. Tehranipoor, and J. Plusquellic, “Detecting malicious inclusions in secure hardware: Challenges and solutions,” in *Hardware-Oriented Security and Trust, 2008. HOST 2008. IEEE International Workshop on*. IEEE, 2008, pp. 15–19.

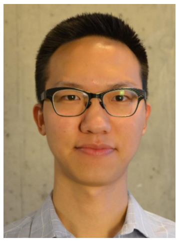
[37] R. Karri, J. Rajendran, K. Rosenfeld, and M. Tehranipoor, “Trustworthy hardware: Identifying and classifying hardware trojans,” *Computer*, vol. 43, no. 10, pp. 39–46, 2010.

[38] J. M. Rabaey, A. P. Chandrakasan, and B. Nikolic, *Digital integrated circuits*. Prentice hall Englewood Cliffs, 2002, vol. 2.

[39] “Trusthub,” http://www.trust-hub.org/benchmarks/trojan.

[40] U. Guin, K. Huang, D. DiMase, J. M. Carulli, M. Tehranipoor, and Y. Makris, “Counterfeit integrated circuits: A rising threat in the global semiconductor supply chain,” *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1207–1228, 2014.

Luong N. Nguyen (S’18) received the B.Sc. degree in Electrical and Computer Engineering from the Hanoi University of Science and Technology in 2013 and the M.Sc. degree in Electrical and Computer Engineering from the Seoul National University in 2016. Since 2016, he has been a Graduate Research Assistant, pursuing the Ph.D. degree in the School of Electrical and Computer Engineering, Georgia Institute of Technology focusing on digital circuit design, software and hardware security, and embedded system. His current research interests span areas of ASIC design, computer architecture, and electrical engineering. He is a past recipient of the Korean Government Scholarship Program, and the best paper award from the 2014 Korean SoC conference.



Chia-Lin Cheng (S’17) received the B.Sc. degree in electrical engineering from the National Taiwan University in 2013 and the M.Sc. degree in electrical engineering from the Georgia Institute of Technology in 2017, respectively. He is currently pursuing his PhD in the Electromagnetic Measurements in Communications and Computing (EMC^2) Lab at the Georgia Institute of Technology focusing on THz chip-to-chip channel measurements and modeling. Previously, he worked on signal integrity and nonlinear circuits I/O modeling by using machine learning. His research interests span areas of electromagnetics, wireless channel measurements and modeling.

Milos Prvulovic (S’97-M’03-SM’09) received the B.Sc. degree in electrical engineering from the University of Belgrade in 1998, and the M.Sc. and Ph.D. degrees in computer science from the University of Illinois at Urbana-Champaign in 2001 and 2003, respectively. He is a Professor in the School of Computer Science at the Georgia Institute of Technology, where he joined in 2003. His research interests mainly focus on the interaction between computer architecture, computer system security, and software engineering.

Dr. Prvulovic is recipient of the following awards/honors: NSF CAREER Award (2005), Best Paper Award at the 49th Annual IEEE/ACM International Symposium on Microarchitecture, 2016, and Distinguished Alumni Educator Award, 2012, from the Department of Computer Science at the University of Illinois at Urbana-Champaign.

Alenka Zajić (S’99-M’09-SM’13) received the B.Sc. and M.Sc. degrees form the School of Electrical Engineering, University of Belgrade, in 2001 and 2003, respectively. She received her Ph.D. degree in Electrical and Computer Engineering from the Georgia Institute of Technology in 2008. Currently, she is an Associate Professor in the School of Electrical and Computer Engineering at Georgia Institute of Technology. Her research interests span areas of electromagnetics, wireless communications, signal processing, and computer engineering.

Dr. Zajić is the recipient of the following awards: NSF CAREER Award (2017), Best Paper Award at the 49th Annual IEEE/ACM International Symposium on Microarchitecture, 2016, 2012 Neal Shepherd Memorial Best Propagation Paper Award, the Best Student Paper Award at the IEEE International Conference on Communications and Electronics 2014, the Best Paper Award at the International Conference on Telecommunications 2008, the Best Student Paper Award at the 2007 Wireless Communications and Networking Conference, LexisNexis Dean’s Excellence Award 2016, and Richard M. Bass/Eta Kappa Nu Outstanding Teacher Award 2016. She has been an editor for IEEE Transactions on Wireless Communications 2012-2017.