Silicon Demonstration of Hardware Trojan Design and Detection in Wireless Cryptographic ICs

Yu Liu, *Student Member, IEEE*, Yier Jin, *Member, IEEE*, Aria Nosratinia, *Fellow, IEEE*, and Yiorgos Makris, *Senior Member, IEEE*

***Abstract*— Using silicon measurements from 40 chips fabricated in Taiwan Semiconductor Manufacturing Company's (TSMC's) 0.35-μm technology, we demonstrate the operation of two hardware Trojans, which leak the secret key of a wireless cryptographic integrated circuit (IC) consisting of an Advanced Encryption Standard (AES) core and an ultrawideband (UWB) transmitter (TX). With their impact carefully hidden in the transmission specification margins allowed for process variations, these hardware Trojans cannot be detected by production testing methods of either the digital or the analog part of the IC and do not violate the transmission protocol or any system-level specifications. Nevertheless, the informed adversary, who knows what to look for in the transmission power waveform, is capable of retrieving the 128-bit AES key, which is leaked with every 128-bit ciphertext block sent by the UWB TX. Moreover, through physical measurements and MATLAB simulations, we show that the attack facilitated by these hardware Trojans is robust to test equipment and communication channel noise. Finally, we experimentally evaluate the effectiveness of a popular hardware Trojan detection method, namely, statistical side-channel fingerprinting via trained one-class classifiers, in detecting the hardware Trojans introduced in our fabricated IC population.**

***Index Terms*— hardware Trojan detection, side-channel fingerprinting, wireless cryptographic integrated circuit (IC).**

I. Introduction

HARDWARE Trojans are malicious modifications introduced in a manufactured integrated circuit (IC), which can be exploited by a knowledgeable adversary to cause incorrect results, steal sensitive data, or even incapacitate a chip [1]–[4]. The problem of hardware Trojans has recently caught the attention of multiple governments and industry across the globe, who are realizing the repercussions of inadvertent deployment of hardware Trojan-infested ICs in sensitive applications and are investing in understanding the risk and developing appropriate solutions. Indeed, traditional IC test methods fall short in detecting hardware Trojans, as they are mainly geared toward identifying modeled defects; therefore, they cannot reveal unmodeled malicious modifications, especially when the latter are carefully hidden within the margins allowed for process variations and do not visibly alter the functionality of the IC.

Among the various hardware Trojan detection methods proposed by researchers over the last few years, statistical analysis of side-channel measurements has received the lion's share of attention. The underlying premise of this approach is that hardware Trojans will distort the side-channel parametric profile of an IC, even if they do not alter its functionality. While for a well-designed hardware Trojan this distortion is minute and carefully hidden within the design margins allowed for process variation, it is systematic; therefore, statistical analysis should be able to identify the presence of additional structure in the side-channel parametric profile of an IC and, thereby, reveal its presence. Accordingly, assuming availability of a small, representative set of trusted Trojan-free ICs, classifiers can be trained to discern between Trojan-free and Trojan-infested chips.

Starting with the global power consumption-based method presented in [5] and the path delay-based method introduced in [6], constructing *fingerprints* of ICs based on side-channel parameters and using these fingerprints to statistically assess whether an IC is contaminated by a hardware Trojan or not became a popular direction. Indeed, numerous researchers in the hardware security and trust area developed this idea further by using various side-channel measurements, including power supply transient signals [7], [8], leakage currents [9], [10], regional supply currents [11], and temperature [12], as well as multiparameter combinations thereof [13], [14].

While all of these methods targeted digital circuits, a similar method using side-channel fingerprinting to detect hardware Trojans in analog/radio-frequency (RF) ICs, and more specifically, in wireless cryptographic ICs was also proposed in [15]. As pointed out therein, the analog/RF domain is an attractive attack target, since the wireless communication of these chips with the environment over public channels simplifies the process of staging an attack without obtaining physical access to the I/O of the chip. On the other hand, signals in an analog/RF IC are continuous and highly correlated to one another; hence, the likelihood of a modification disturbing these correlations is very high. As a result, side-channel-based hardware Trojan detection methods are very effective in this domain, as shown using simulations of a Trojan-free and two Trojan-infested versions of a wireless cryptographic IC in [15].

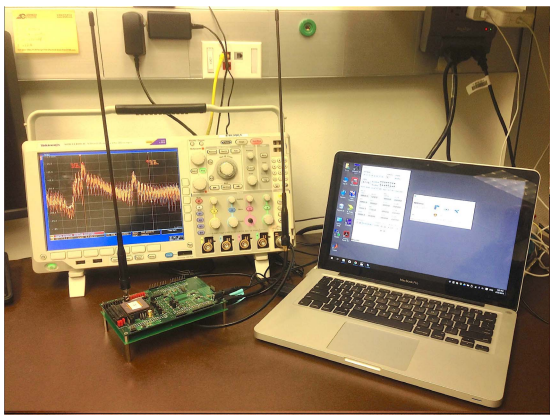
Manuscript received July 2, 2016; revised October 10, 2016; accepted November 8, 2016. Date of publication December 21, 2016; date of current version March 20, 2017. This work was supported by the National Science Foundation under Grant NSF 1149465 and Grant NSF 1514050.

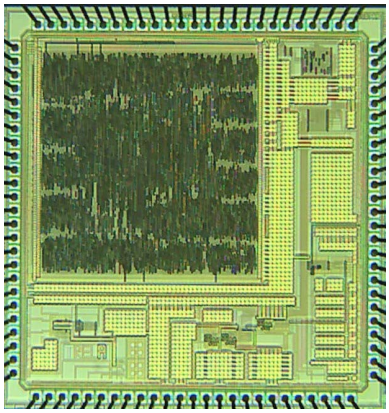
Y. Liu, A. Nosratinia, and Y. Makris are with the Department of Electrical Engineering, The University of Texas at Dallas, Richardson, TX 75080 USA (e-mail: yxl119120@utdallas.edu; aria@utdallas.edu; yiorgos.makris@utdallas.edu).

Y. Jin is with the Department of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL 32816 USA (e-mail: yier.jin@eecs.ucf.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2016.2633348





Technology	TSMC 0.35um
Chip Size	3mm X 3mm
Modulation	FSK-OOK
Data Rate	Up to 96MHz
Pulse Width	7ns – 48ns
Frequency of "0"	380MHz – 490MHz
Frequency of "1"	560MHz – 750MHz

Fig. 1. Experimental platform, die photograph, and circuit specifications.

The vast majority of hardware Trojan detection methods reported in the literature so far, whether for digital or analog/RF ICs, have only been assessed in simulation or emulation via Field Programmable Gate Arrays (FPGAs). Very few custom silicon implementations of hardware Trojans and evaluations of relevant detection solutions using actual measurements exist, mostly employing toy examples of digital hardware Trojans. The work presented herein seeks to fill this gap in the analog/RF domain by designing, fabricating, and characterizing a wireless cryptographic IC containing two hardware Trojans capable of leaking the encryption key. Indeed, silicon measurements are essential in order to convincingly assess the effectiveness of hardware Trojan operation and detection methods based on side-channel fingerprinting, especially in the analog/RF domain. Our approach follows the same general principles for leaking secret information and the same hardware Trojan detection method, as introduced in [15], although our design is slightly different. Accordingly, our results corroborate the findings of [15] through actual silicon measurements, as opposed to simulation-based results. Furthermore, we demonstrate experimentally the robustness of this hardware Trojan attack to both measurement and channel noise. To the best of our knowledge, this is the first silicon demonstration of working hardware Trojans in a wireless cryptographic IC and the first evaluation of side-channel-based statistical analysis methods for detecting them.

The remainder of this paper is structured as follows. In Section II, we introduce the chip that we designed and fabricated for the purpose of this paper. Specifically, we describe the Trojan-free and the two hardware Trojan-infested versions of an Advanced Encryption Standard (AES) + ultra-wideband (UWB) wireless cryptographic IC. In Section III, we discuss the mechanism through which the key is leaked by the two hardware Trojans, as well as issues pertaining to the robustness of the attack. Then, in Section IV, we discuss how these hardware Trojans evade detection by traditional manufacturing test methods. In Section V, we demonstrate the effectiveness of side-channel fingerprinting in revealing the presence of a hardware Trojan based on statistical analysis of transmission power using one-class classifiers. A short discussion follows in Section VI with conclusions drawn in Section VII.

II. Wireless Cryptographic IC

The wireless cryptographic IC used in this paper consists of a digital part and an analog part. The digital part, which occupies over 99% of the design, is an AES core followed by an output buffer. The analog part is a UWB transmitter (TX) [16], which is very small and easy to integrate on-chip. Our experimental platform, which is shown in Fig. 1, supports one Trojan-free and two Trojan-infested operation modes of the wireless cryptographic IC. As described in detail later in this section, in the two Trojan-infested operation modes, the added hardware Trojans leak the AES encryption key by hiding it in the wireless transmission power amplitude/frequency margins allowed for process variations, while ensuring that the circuit continues to meet all of its functional specifications.

The chip was designed in TSMC's 0.35-μm process and was fabricated through MOSIS, with all 40 chips received functioning correctly. The area of the die is 9 mm^2 and the design includes approximately 110 K gates. The digital part runs at a frequency of up to 48 MHz and consumes 66.42 mW while the UWB TX has a data rate of up to 96 Mb/s and consumes 14.72 mW. The chip specifications are listed in Fig. 1. The die is packaged in a PGA108M package. The packaged die sits in a socket of a custom FR-4 printed circuit board (PCB), which is connected to an Opal Kelly XEM 3010 FPGA board with 2.5 V power supply, through which the wireless cryptographic IC can be controlled from a PC via MATLAB. The bias voltage of the UWB TX is controlled by an 8-bit Digital to Analog Converter (DAC) (AD5668) on the PCB. The wireless transmission is carried out through a pair of dual-band handheld antennas [17]. One antenna is connected to the UWB TX output of our chip. The other antenna is connected to our oscilloscope (Tektronix MDO4104-6), which has a built-in spectrum analyzer and acts as the receiver (Rx).

A. Trojan-Free Version

A system-level block diagram of the circuit is shown in Fig. 2. The AES core receives plaintext in blocks of 128 bits, which it encrypts using a 128-bit key that is loaded through the "key" input and stored on-chip. The width of the encryption key determines the number of transformation rounds to which the plaintext is subjected during encryption. In this case, after

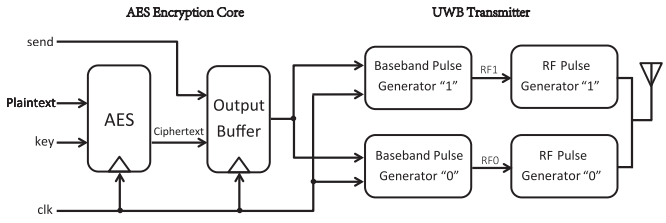


Fig. 2. System-level block diagram.

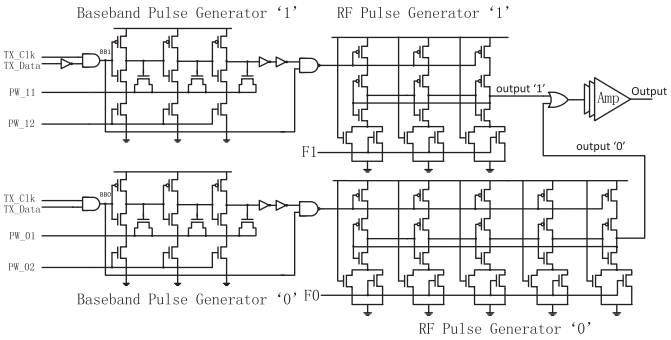


Fig. 3. UWB TX schematic.

ten rounds of transformation, the plaintext is encrypted into ciphertext, which is stored in an output buffer in blocks of 128 bits, until it is transmitted. The output buffer is a first-in first-out (FIFO) structure of 128-bit words, which is the length of the ciphertext. The output buffer sends ciphertext to the UWB TX serially. The UWB TX designed in this platform includes a baseband pulse generator and an RF pulse generator, as shown in Fig. 3. In our design, frequency-shift keying (FSK) is used to distinguish the polarity of a bit, while ON–OFF keying (OOK) is used to separate adjacent bits. Bit values of "0" and "1" are separated and converted to return-to-zero (RTZ) format in the baseband pulse generator. The pulsewidth (PW) is controlled by two types of signals. Specifically, the first type of PW control signals, PW_01 and PW_11, controls the capacitance of several voltage-controlled capacitors, which adjust the delay of the signal path, thereby controlling the PW of the baseband signal. The second type of PW control signals, PW_02 and PW_12, adjusts the current through each branch by controlling the gate voltage of the bottom nMOS transistors. With higher gate voltage, more current flows through each branch and the PW of the baseband signal becomes smaller. With lower gate voltage, the PW of the baseband signal becomes wider. The output of the baseband pulse generator controls the input of the RF pulse generator. Here, a ring oscillator is used as a voltage-controlled oscillator (VCO) to gene rate the RF pulse, and the pulses of signals "0" and "1" are assigned to two different frequencies. Signals F0 and F1 are used to control the pulse frequency by controlling the current through each branch. The higher the current through each branch, the larger the oscillation frequency. The modulation waveform of the UWB TX is shown in Fig. 4. The last part of the design is a power amplifier (PA), consisting of several stages of inverters, which are used to combine the signals "1" and "0" from the VCOs.

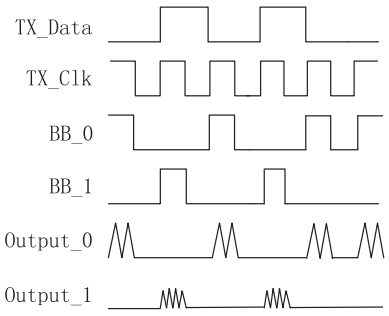


Fig. 4. Modulation waveform.

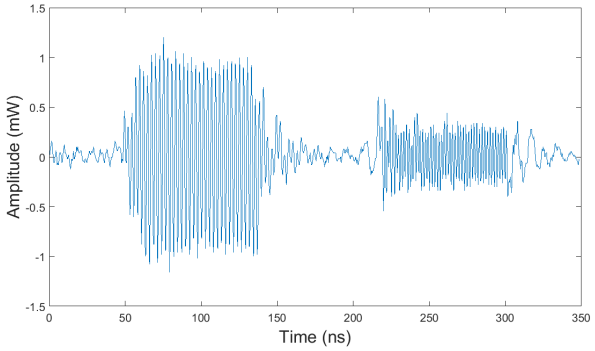


Fig. 5. Transmission power while sending "1" and "0."

The all-digital design of this UWB TX makes it compatible with the digital part and reduces overall power consumption and die area. An example of a typical transmission of a "1" and a "0" is shown in Fig. 5. We note that the transmission of signal "1" has higher amplitude and lower frequency than transmission of signal "0."

B. Trojan-Infested Versions

In order to reduce the risk of being detected, the underlying hardware changes required for introducing hardware Trojans should be very simple. Indeed, in our experimental platform, *minor* additions/modifications to the digital and analog parts of the circuit are needed in order to leak the encryption key over the public channel, as shown in Fig. 6 and as we explain in the following. Specifically, we designed two different hardware Trojans, both of which require the same simple change on the digital side, while each requires a slightly different change on the analog side. On the digital side, the added hardware taps into the register that stores the 128-bit AES key, in order to steal one bit at a time. The value of the stolen key bit is passed to the UWB TX, through which it is leaked by modulating the parameters of the wireless transmission (i.e., amplitude or frequency) during the transmission of one ciphertext bit. Overall, along with every 128-bit block transmitted by the UWB TX, the 128-bit key is also leaked. On the analog side, the modifications needed to leak a stolen key bit with each transmitted ciphertext bit by each of the two alternative hardware Trojans are also very simple, taking advantage of the design margins provided to account for fabrication process variation. Specifically, the first hardware Trojan (Trojan-I)

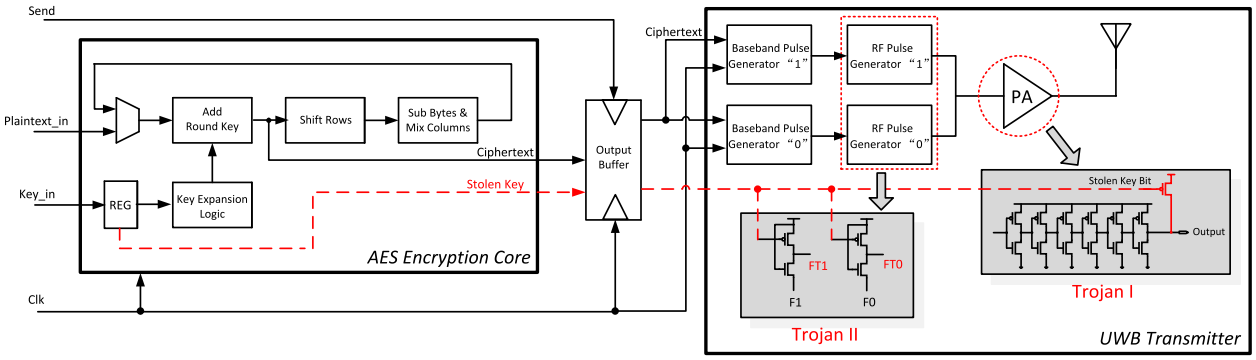


Fig. 6. Hardware Trojan modifications in digital and analog parts.

is located in the PA, while the second hardware Trojan (Trojan-II) is located in the RF pulse generators.

For Trojan-I, a pMOS transistor is added to the output of the PA of the UWB TX, and the stolen key bit is connected to the gate of this pMOS transistor. Accordingly, when the stolen key bit is "0," the pMOS transistor is turned ON and draws a small additional current from the power supply to the output, thereby slightly increasing the transmission power. Conversely, when the stolen key bit is "1," the pMOS transistor is turned OFF, so no additional current is drawn to the output, with the circuit, essentially, continuing to operate as in the Trojan-free case.

For Trojan-II, only two transistors are added at the input of each RF pulse generator. In the original design, signals F0 and F1, which control the frequency of the output signal, are connected directly to the RF pulse generators, as shown in Fig. 3. In the modified design, however, they are no longer connected to the RF pulse generators. Instead, FT0 and FT1 are connected to the RF pulse generators. When the stolen key bit is "0," the pMOS transistor in Trojan-II is turned ON, thereby resulting in

$$FT0/1 = F0/1 + (Vdd - F0/1) * \frac{ron}{ron + rop} \quad (1)$$

where ron and rop are the channel resistances of the nMOS and the pMOS transistors, respectively, and V_{dd} is the supply voltage. The sizes of the pMOS transistors and the nMOS transistors can be carefully designed, so that FT0/FT1 is just slightly higher than F0/F1, which, in turn, makes the frequency of the pulse "0"/"1" just slightly higher than its original value. Conversely, when the stolen key bit is "1," the pMOS transistor is turned OFF, in which case FT0 and FT1 are equal to F0 and F1, respectively, with the circuit essentially continuing to operate as in the Trojan-free case.

Fig. 7 shows the impact of hardware Trojan-I on the transmission power waveform of a Trojan-infested chip. Fig. 7(a) contrasts the power waveforms for transmitting a logic "0" when the stolen key bit is "1" and "0," respectively. In the latter case, the slight increase in transmission power is evident across the waveform, with the difference peaking at 0.14 mW. Similarly, Fig. 7(b) contrasts the power waveforms for transmitting a logic "1" when the stolen key bit is "1" and "0," respectively, with the difference in transmission power peaking at 0.2 mW.

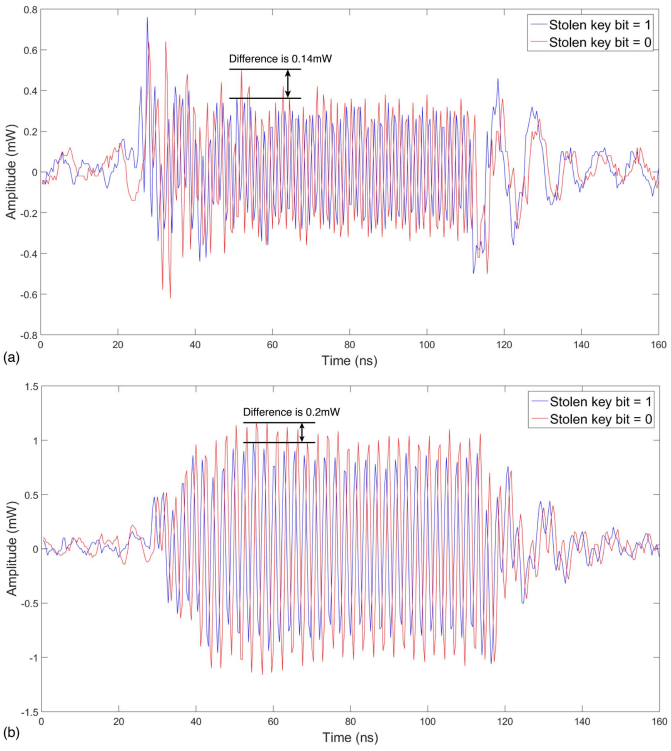


Fig. 7. Difference in transmission power waveform of Trojan-I infested chip when the stolen key bit is "0" and "1" while transmitting (a) ciphertext bit of value "0" and (b) ciphertext bit of value "1."

Similarly, Figs. 8 and 9 show the impact of hardware Trojan-II on the transmission power waveform of a Trojan-infested chip. Fig. 8 contrasts the power waveforms for transmitting a logic "0" when the stolen key bit is "1" and "0," respectively. In the latter case, the slight increase in transmission frequency is evident across the waveform, with the difference measured at 24 MHz. The left plot in Fig. 8 shows the two waveforms in the time domain. The frequency difference is too small to be distinguished visually, so we convert the waveforms into the frequency domain for easier observation. The right plot in Fig. 8 contrasts the power waveform for transmitting a logic "0" in the frequency domain, when the stolen key bit is "1" and "0," respectively. The difference due to the hardware Trojan is now evident when

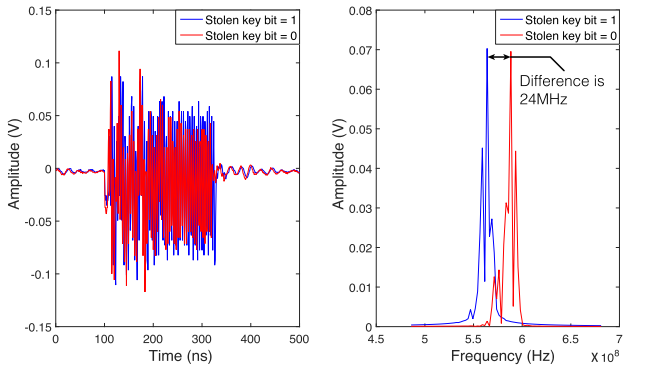


Fig. 8. Difference in transmission power waveform of Trojan-II infested chip when the stolen key bit is "0" and "1" while transmitting a ciphertext bit of value "0."

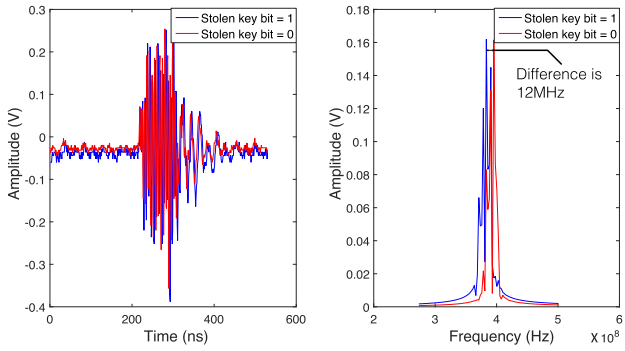


Fig. 9. Difference in transmission power waveform of Trojan-II infested chip when the stolen key bit is "0" and "1" while transmitting a ciphertext bit of value "1."

comparing these two waveforms. Similarly, Fig. 9 contrasts the power waveforms for transmitting a logic "1" when the stolen key bit is "1" and "0," respectively, with the difference in transmission frequency measured at 12 MHz. Once again, the left plot in Fig. 9 shows the two waveforms in the time domain, while the right plot in Fig. 9 shows the same information transformed into the frequency domain, with the difference incurred due to the hardware Trojan becoming evident.

We emphasize that the slight increase in transmission power amplitude (by Trojan-I) or frequency (by Trojan-II) when the hardware Trojan is turned ON (i.e., when the stolen key bit is "0") is very small and leaves the circuit well within its functional specification margins allowed for process variations and operating condition fluctuations. In other words, all of these transmissions, when considered individually, appear perfectly legitimate and do not raise any suspicions, as they do not violate any specifications and could have been produced by a chip originating from the Trojan-free distribution, as we will demonstrate further in Section IV.

Finally, we also point out that the overall area overhead incurred by Trojan-I and Trojan-II is 0.005% and 0.025%, respectively, which is extremely small and hard to detect through visual inspection. Similarly, despite being always ON, the overall power consumption of Trojan-I and Trojan-II is 360 and 90 μW, respectively. When expressed as a percentage of the 81.14 mW of total power consumed by the chip, this

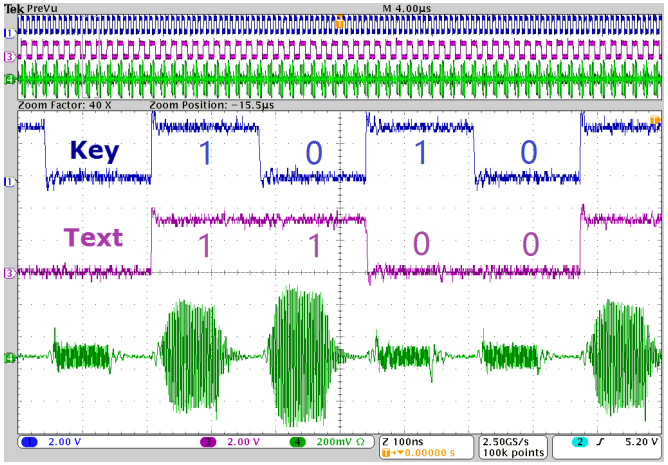


Fig. 10. Received waveform of a 4-bit ciphertext block transmitted by Trojan-I infested chip.

amounts to 0.4% and 0.1%, respectively, making it difficult to detect by conventional or statistical methods.

III. Hardware Trojan Operation

We now proceed to describe how the hardware Trojan-induced modifications in the transmission power waveform can be used for stealing the encryption key and we evaluate the robustness of the attack to measurement and channel noise.

A. Stealing the Key

Despite being hidden in the process variation margins, the impact of the hardware Trojan on the transmission power waveform suffices for the informed adversary to obtain the secret key and, by extension, the plaintext by deciphering the ciphertext. We emphasize that the attacker does not need to know the exact shape of the waveform when a key bit of value "0" and a key bit of value "1" is leaked. In fact, it is impossible to know this information, since every chip will be affected differently by process variations. Indeed, the attacker does not rely on absolute values. Rather, it is the minute relative difference between transmissions by the same chip that gives away the secret. All the attacker has to do is listen to the public wireless transmission channel, focusing on the parameter manipulated by the hardware Trojan (i.e., amplitude or frequency), in order to observe the different levels, which correspond to a key bit of "1" and "0," respectively, when a ciphertext bit of value "0" and a ciphertext bit of value "1" are transmitted (i.e., the waveforms of Figs. 7–9). Once these four waveforms are known to the attacker, observing the transmission of a 128-bit block suffices to obtain the entire 128-bit AES key.

Fig. 10 shows an example of how the encryption key is leaked by a Trojan-I infested chip. We transmitted a ciphertext through the antenna connected to the UWB TX on our chip and we received it through the second antenna, which was connected to the oscilloscope. Fig. 10 zooms in on a 4-bit portion of the 128-bit ciphertext transmitted by the UWB TX. The value of this 4-bit snippet is "1100," which is shown

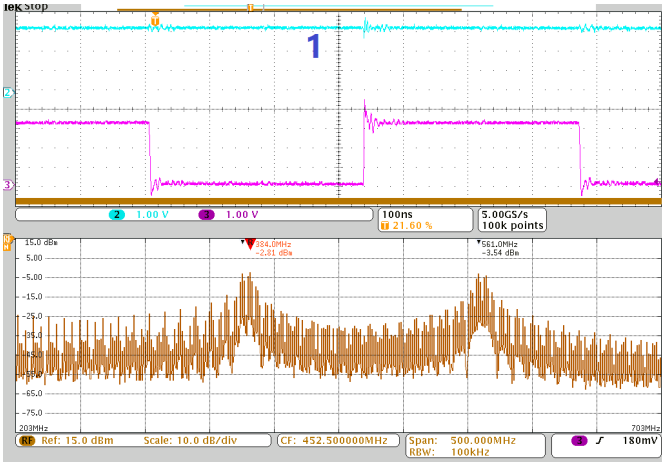


Fig. 11. Received waveform of ciphertext bitstream transmitted by Trojan-II infested chip, when leaked key bit is "1."

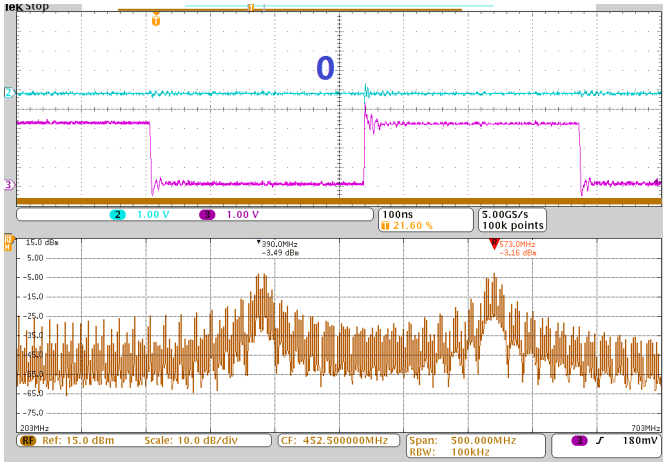


Fig. 12. Received waveform of ciphertext bitstream transmitted by Trojan-II infested chip, when leaked key bit is "0."

in the purple (middle) trace. The corresponding 4-bit key portion leaked during transmission of this ciphertext snippet is "1010" and is shown in the blue (top) trace. The received ciphertext signal is shown in the green (bottom) trace. This signal is perfectly legitimate for every bit transmitted in this example, as it stays well within the specifications of the circuit. Indeed, due to its simple structure and careful transistor sizing, the changes that Trojan-I imposes on the parameters of the transmission power waveform (e.g., power gain, power efficiency, and power consumption) when it is enabled (i.e., when the stolen key bit is "0") are very small, measured by the oscilloscope at 81 and 32 mV for ciphertext bit values of "0" and "1," respectively. However, comparative observation of the transmission power amplitude in the received waveforms reveals the values of the key bits to the attacker.

Similarly, Figs. 11 and 12 show an example of how the encryption key is leaked by a Trojan-II infested chip. Once again, we transmitted a ciphertext through the antenna connected to the UWB TX on our chip, we received it through the second antenna connected to the oscilloscope, and we converted it to the frequency domain through the built-in spectrum analyzer of our oscilloscope. These examples also zoom in on a snippet of the ciphertext including both signal "0" and signal "1." Specifically, Fig. 11 shows the received signal when the leaked bit is "1," while Fig. 12 shows the received signal when the leaked bit is "0." In Figs. 11 and 12, the top half shows the ciphertext bits, shown in the purple (bottom) trace, along with leaked key bits, shown in the blue (top) trace, in the time domain. The bottom half of Figs. 11 and 12 shows the received signal in the frequency domain. In order to be compatible with the antennas, the transmission frequency of signal "1" was tuned to be centered at 380 MHz, while the transmission frequency of signal "0" was tuned to be centered at 561 MHz. As in the case of Trojan-I, the impact of Trojan-II on the transmission parameters is inconspicuous and the received signals for each transmitted ciphertext bit are perfectly legitimate, remaining well within the specifications of the circuit. Specifically, when the leaked key bit is "0," the transmission frequency of ciphertext bits "1" and "0" is increased to 390 and 573 MHz, respectively, as shown in Fig. 12. Once again, while the difference when the leaked key bit is "1" and when the leaked key bit is "0" is very small, it is systematic; therefore, comparative observation of the transmission frequency reveals the values of the key bits.

B. Attack Robustness

For the designed hardware Trojans to facilitate a robust attack, the difference between the transmission power waveforms when the stolen key bit is "0" and "1" should be discernible even in the presence of measurement noise and environmental variations. When measuring transmission power/frequency, unavoidable measurement noise is introduced due to the accuracy of the test equipment (i.e., starting point and step size precision), resulting in slightly different outcomes for the same waveform. Environmental conditions, such as temperature, electromagnetic interference, and test-board setup, may also impact the measurements. To assess the robustness of the introduced hardware Trojans to noise, we conducted ten repetitions of the same measurements while placing the chip in different locations with variable ambient noise and after operating the chip for a variable time to introduce differences in operating temperature. While this was not a controlled-temperature experiment and no temperature sensors are available on chip, the on-chip voltage variation caused by the different temperature conditions resulted in transmission power variation in the range of 5%–7%.

Fig. 13(a) shows the ten power waveforms obtained from a Trojan-I infested chip while transmitting a "1" and a "0" when the stolen key bit is a "1." Similarly, Fig. 13(b) shows the same measurements when the stolen key bit is "0." As may be observed, the lowest peak amplitude among the ten repetitions shown in Fig. 13(b) is always above 1.16 and 0.48 mW for transmitting a "1" and a "0," respectively, when the leaked key bit is "0." In contrast, the corresponding highest peak amplitude shown in Fig. 13(a) never exceeds 1.07 and 0.46 mW, respectively, when the leaked key bit is "1." Hence, the difference is clearly distinguishable and Trojan-I robustly leaks the encryption key.

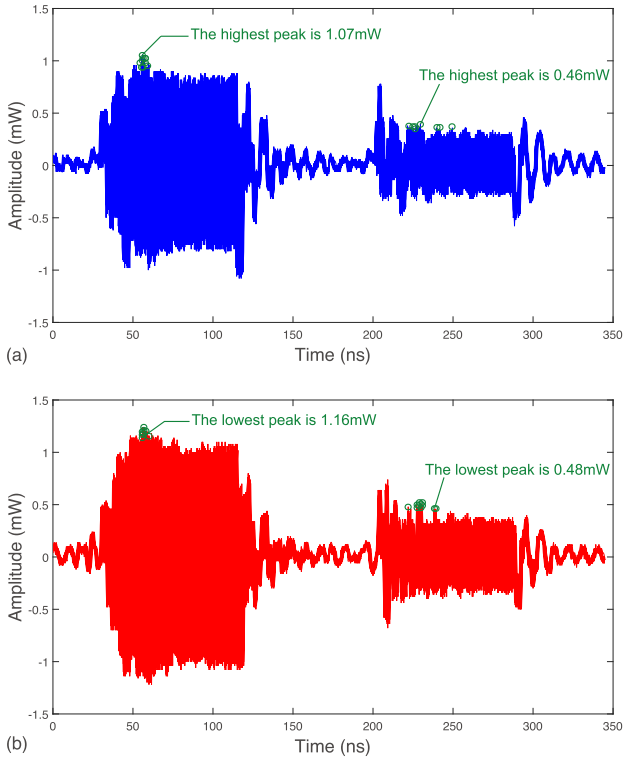


Fig. 13. Ten repetitions of measuring transmission amplitude while transmitting a "1" and a "0" when the stolen key bit is (a) "1" and (b) "0."

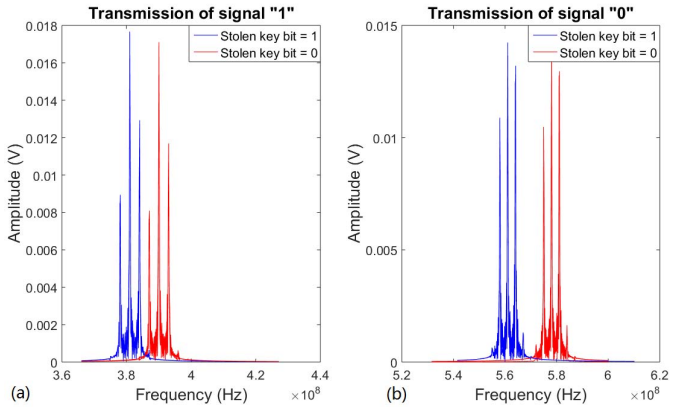


Fig. 14. Ten repetitions of measuring transmission frequency while (a) transmitting a "1" when the stolen key bit is "1" and "0" and (b) transmitting a "0" when the stolen key bit is "1" and "0."

Fig. 14(a) shows ten transmission frequency plots obtained from a Trojan-II infested chip while transmitting a "1" when the leaked key bit is "1" and "0," while Fig. 14(b) shows the same measurements while transmitting a "0" when the leaked key bit is "1" and "0." As may be observed, the highest frequency among the ten repetitions shown in the blue (left) traces of Fig. 14(a) is below 385 MHz when the leaked key bit is "1." In contrast, the lowest frequency among the ten repetitions shown in the red (right) traces is always above 385 MHz when the stolen key bit is "0." Hence, the difference is clearly distinguishable. Similarly, as shown in Fig. 14(b), when transmitting a "0," the transmission frequency is always below 570 MHz when the leaked key bit is "1," shown in blue (left), and always above 570 MHz when the leaked key bit is "0," shown in red (right). Hence, Trojan-II is also robust to measurement noise while leaking the encryption key.

The above-mentioned experiment evaluated the robustness of our Trojan-infested UWB TX against measurement noise, while leaking the encryption key. Besides measurement noise, however, a robust attack should also be able to withstand noise on the communication channel, as well as on the Rx circuitry, which may contribute to an increase in bit error rate (BER). The ability of an Rx to robustly obtain the key leaked through our Trojan-infested UWB TX over the air, as shown in Figs. 10–12, was demonstrated in a controlled laboratory environment where channel noise is low. In case of arbitrary channel conditions or variable transmission distance, however, such robustness may be jeopardized. Therefore, in order to evaluate robustness of both ciphertext reception and stolen key reception in a noisy environment, we perform an experiment, which combines on-chip instrumentation for controlling the intensity level of the Trojan impact on the transmitted signal, with MATLAB simulation for introducing various levels of channel noise.

Specifically, instead of the single pMOS transistor shown in Fig. 6 for Trojan-I, we have implemented an array of four pMOS transistors (X1, X2, X4, and X8), each being twice the width of the previous one, as shown in Fig. 15(a). The gates of these four transistors are still driven by the leaked key bit but only after going through four two-input OR gates, the other inputs of which are controlled through a 4-bit control code (en4, en3, en2, and en1). When all four control bits are "0," the circuit operates as Trojan-free, since the OR gates produce a logic "1" and all pMOS transistors are OFF, so no additional current is drawn, independent of the key value. For all other values of the 4-bit control code, at least one pMOS transistor turns ON when the stolen key bit value is "0." The sizes of the four pMOS transistors were carefully chosen, such that the 4-bit code can control the transmission power amplitude difference when the stolen key bit is "0" and when it is "1" in one of 15 distinct levels. The same capability was also designed for Trojan-II, by replacing each of the two pMOS transistors shown in Fig. 6, which generate signals FT0 and FT1, respectively, with four transistors (X1, X2, X4, and X8), as shown in Fig. 15(b). In this case, the 4-bit code controls the difference in transmission power frequency when the stolen key bit is "0" and when it is "1" in one of 15 distinct levels. In the rest of this paper, we will use Level 0 to denote the case when all pMOS transistors are OFF, Level 1 to denote the case where only the smallest pMOS transistor is ON (i.e., minimal Trojan impact), and so on, until Level 15 which denotes the case where all four pMOS transistors are ON (i.e., maximal Trojan impact). We emphasize that even at Level 15, all the UWB transmissions remain within their specifications.

Besides controlling the Trojan impact level, as explained earlier, we also used MATLAB simulations to introduce additive white Gaussian noise (AWGN) to the transmission waveform, thereby modeling communication channel noise and Rx circuitry noise. With these capabilities at hand, we conducted

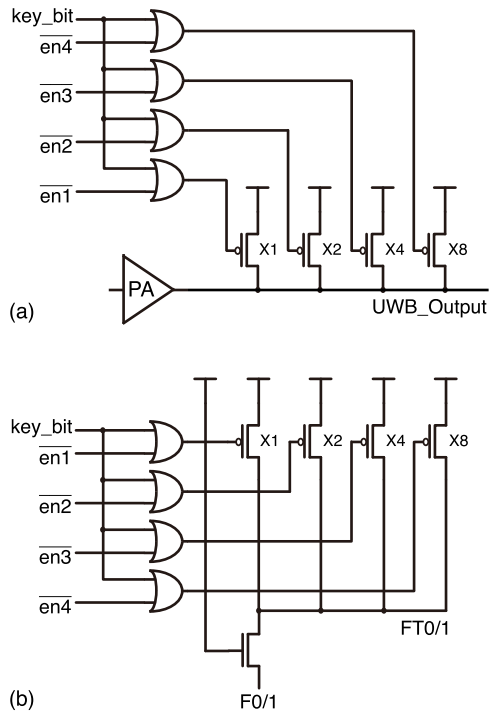


Fig. 15. On-chip provisions for controlling the impact of (a) Trojan-I and (b) Trojan-II in 15 distinct levels.

the following experiment to assess robustness of ciphertext reception and leaked key reception in a noisy environment. We started by using one of our chips to transmit a 128-bit block of ciphertext 15 times, while leaking the same 128-bit AES encryption key, first through Trojan-I and then through Trojan-II. For each of these 15 transmissions, a distinct impact level was used to control the transmission amplitude or frequency incurred by Trojan-I and Trojan-II, respectively. All in all, for each of the two Trojans, we collected 15 transmission power waveforms. To each of these waveforms, we added nine different levels of AWGN, with signal-to-noise ratio (SNR) ranging from 0 to 40 dB with a step size of 5 dB. Two bandpass filters, with a central frequency of 380 and 560 MHz, respectively, were then used to model Rx functionality in MATLAB and distinguish between ciphertext signals "0" and "1." After filtering, the method of Section III-A is applied to extract the leaked key bit values. Comparison to the correct ciphertext and key values is then performed to compute BER. The experiment was repeated 30 times and the results reported next reflect the average over these 30 repetitions.

Fig. 16 shows the ciphertext BER results for Trojan-I infested transmissions. Results for the Trojan-free case (i.e., Trojan impact Level 0) are also provided for the purpose of comparison. In the worst case scenario, where SNR is 0 dB and the impact level of the Trojan is minimal (i.e., Level 1), more than half of the ciphertext bits (i.e., 72/128) are incorrectly received. As the channel conditions improve, at an SNR of 15 dB, BER drops to 36/128 incorrectly received ciphertext bits, while at 20 dB, it drops further to 20/128. Beyond that SNR point, BER becomes negligible, even for the lowest Trojan impact level. At each SNR point, increasing the Trojan impact level slightly reduces BER. This is explained by noticing that whenever the leaked key bit is "0," the Trojan-infested transmission signal has increasingly higher amplitude as we increase the Trojan impact level. Thus, it can withstand more noise, essentially helping the Rx correctly interpret the ciphertext waveform and reducing BER. Overall, these results confirm that the presence of Trojan-I does not jeopardize the robustness of ciphertext reception; in fact, it actually helps in slightly improving it.

Fig. 17 shows the leaked key BER results for Trojan-I infested transmissions. When the impact level of the Trojan is minimal (i.e., Level 1), the BER is very high, with 49/128 leaked key bits incorrectly received even when the channel is very clean at an SNR of 40 dB. This is expected, as the extra transmission amplitude when the leaked key bit is "0" is extremely small in this case and becomes easily lost even in relatively little channel noise. Similarly, when the channel is very noisy, at an SNR of 0 dB, even the maximal level of Trojan impact (i.e., Level 15) results in a very high BER of 61/128 incorrectly received key bits. The situation improves quickly as the SNR increases, becoming negligible at 25 dB for the maximal level Trojan impact. The results clearly depict the joint impact of SNR and Trojan intensity, revealing that the leaked key can be received very robustly as the channel conditions improve and as we approach the maximal impact level (Level 15), for which Trojan-I was originally designed. We remind that, even at this impact level, the transmissions are well within their specifications and appear legitimate.

Figs. 18 and 19 show the same results for Trojan-II infested transmissions. Regarding ciphertext reception, in the worst case of SNR of 0 dB and minimal Trojan impact level (i.e., Level 1), 70/128 ciphertext bits are incorrectly received. The situation improves quickly as the SNR and the impact level of the Trojan increase, though not always as quickly as in the case of Trojan-I. This is expected, since Trojan-II modulates transmission power frequency rather than amplitude; hence, it only indirectly counteracts channel AWGN. Overall, the results confirm again that the presence of Trojan-II does not reduce the robustness of ciphertext reception; in fact, it improves it. Regarding leaked key reception, in the worst case of SNR of 0 dB and minimal Trojan impact level (i.e., Level 1), 82/128 key bits are incorrectly received. As expected, increasing SNR and/or the impact level of the Trojan quickly reduces leaked key BER, becoming negligible at an SNR of 30 dB even for the minimal Trojan impact level (i.e., Level 1). Overall, as can be observed in the results, Trojan-II is more robust than Trojan-I in leaking the key. This is, again, attributed to the type of noise introduced in the channel (i.e., AWGN), which masks slight amplitude differences much more effectively than slight frequency differences.

Finally, with respect to correct retrieval of the leaked encryption key, we should also point out that the infrequent change of the key offers another opportunity for dealing with BER. Specifically, since the 128-bit key is leaked with every 128-bit ciphertext, an attacker could repeat the process an odd number of times and take a majority vote to decide the value

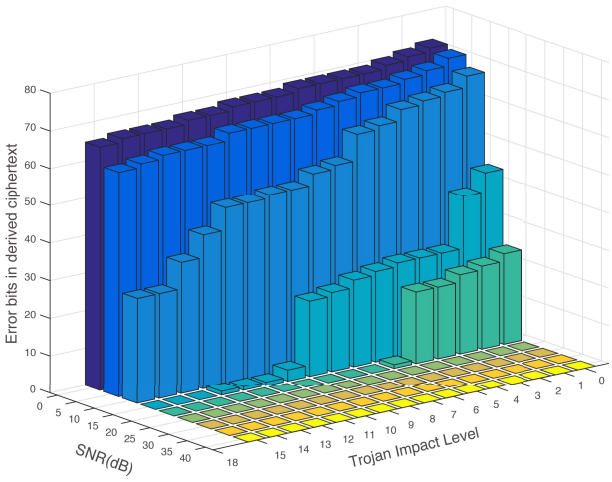


Fig. 16. Error bits in retrieved ciphertext for Trojan-I infested transmissions.

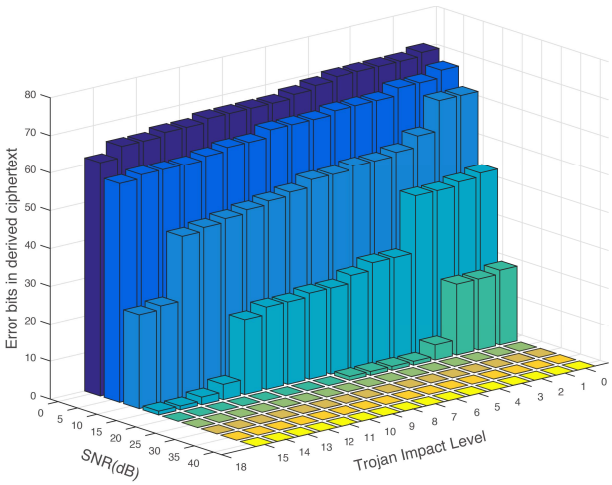


Fig. 18. Error bits in retrieved ciphertext for Trojan-II infested transmissions.

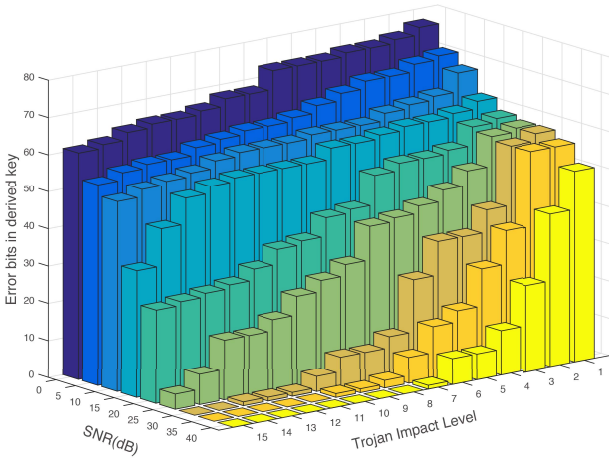


Fig. 17. Error bits in retrieved encryption key for Trojan-I infested transmissions.

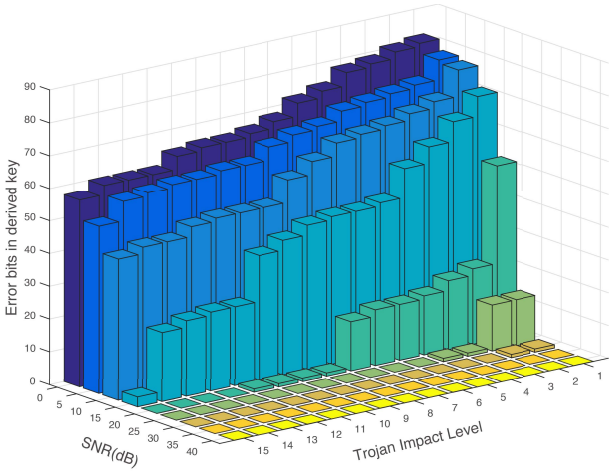


Fig. 19. Error bits in retrieved encryption key for Trojan-II infested transmissions.

of each key bit. In our experiments, after three repetitions, this simple process eliminated almost all of the leaked key BER reported earlier.

IV. Detection Evasion

The hardware Trojans introduced in our design evade traditional manufacturing test, since they do not change the functionality of the circuit and they do not violate any circuit- or system-level specifications. As commonly practiced in mixed-signal ICs, we generated test vectors that cover both stuck-at and transition faults in the digital portion of the design by using a full-scan chain of enhanced scan flip-flops. However, since our hardware Trojan only taps into the register holding the encryption key and does not alter the functionality of the AES circuit, no functional or structural digital test targeting stuck-at or transition faults is going to expose it. Furthermore, the added capacitive load for leaking the key, one bit at a time, is very low to make the circuit fail any delay tests or to be picked up by statistical side-channel fingerprinting methods for hardware Trojan detection, such as [6].

Similarly, on the UWB side, the impact of the introduced hardware Trojan (i.e., one pMOS transistor for Trojan-I and two pairs of transistors for Trojan-II) is hidden within the process variation margins. In other words, for the vast majority of fabricated devices, the transmission power waveform will continue to be within the UWB transmission specifications. It is possible, however, that for a very small number of chips at the tails of the distribution, the extra nudge provided by the hardware Trojan might push them outside the specifications, thereby slightly reducing yield. Nevertheless, such yield loss could be caused by many other reasons (process drifts, material impurities, mask misalignment, measurement noise, and so on) and there is no way to attribute it to the presence of a hardware Trojan. In our case, none of the 80 Trojan-infested circuits ended up outside the specifications, while all of them could robustly leak the secret key. System-level test is also not going to reveal these hardware Trojans, since they do not transmit any additional bits and they do not violate the transmission protocol in any way.

To demonstrate the difficulty in detecting these hardware Trojans, in Fig. 20(a)–(c), we show the measured transmission power for transmitting a ciphertext bit of “0” and “1” by each of the 40 hardware Trojan-free, hardware Trojan-I

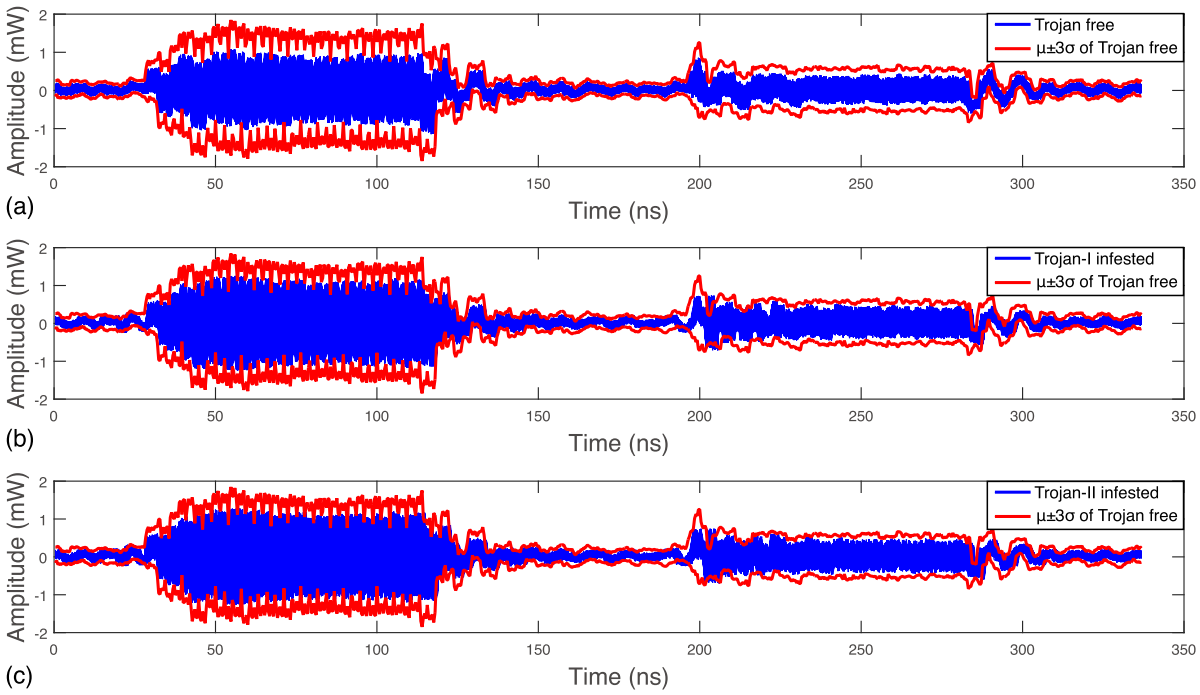


Fig. 20. Transmission power of (a) 40 Trojan-free circuits, (b) 40 Trojan-I infested circuits, and (c) 40 Trojan-II infested circuits enclosed in the $\mu \pm 3\sigma$ envelop of the Trojan-free circuits.

infested, and hardware Trojan-II infested circuits. During these transmissions, the maximal impact level of each Trojan is employed. Each of these three distributions is enclosed in the $\mu \pm 3\sigma$ transmission power envelop of the hardware Trojan-free circuits. The key observation based on Fig. 20 is that the three distributions are very similar. Clearly, given any one of these 120 transmission power waveforms, it is very difficult, if not impossible, to definitively tell whether it comes from a hardware Trojan-free circuit or a hardware Trojan-infested circuit.

We note that the specification margins allowed for interdie process variation are much wider than the margins needed for a single chip to deal with variable SNR on the communication channel. Therefore, while these measurements were taken over an actual communication channel with typical ambient noise, higher or lower SNR would not adversely impact the ability of the Trojans to remain hidden.

V. Side-Channel Fingerprinting Evaluation

Having demonstrated the robustness of our hardware Trojans in stealing the cryptographic key, we now proceed to evaluate the effectiveness of a popular hardware Trojan detection method, namely, side-channel fingerprinting, in detecting them. As mentioned in Section I, the underlying premise of statistical side-channel-based hardware Trojan detection methods is that the distortion imposed by hardware Trojans on the parametric profile of an IC is *systematic*, even though it is hidden within the design margins allowed for process variations. For example, the hardware Trojans introduced in this paper increase slightly the transmission amplitude/frequency when the stolen key bit value is "0," without violating any transmission specifications. This systematic impact of the attack is indispensable, since the adversary relies on it in order to discern the hidden information. However, any systematic component, subtle as it might be, imposes added statistical "structure" to the transmission power of a population of chips. This added "structure" is precisely what statistical side-channel fingerprinting methods for hardware Trojan detection rely on.

In the following, we first employ a simple statistical analysis method, namely, principal component analysis (PCA) [18], to visualize in a 3-D space the existence of such statistical "structure" in the silicon measurements obtained from our fabricated Trojan-free and Trojan-infested wireless cryptographic ICs. We then show that one-class classifiers, such as a simple minimum volume enclosing ellipsoid (MVEE) [19] drawn in the 3-D space of the first three principal components of the data, or a more advanced one-class support vector machine (SVM) [20] trained with the original multidimensional data, can be used to effectively distinguish between Trojan-infested and Trojan-free circuits, with the SVM being able to do so in lower dimensionality (i.e., with fewer measurements).

A. Hardware Trojan Detection via PCA and MVEE

In order to visualize through PCA the added statistical "structure" imposed by hardware Trojans, we randomly selected six different blocks of plaintext, which we encrypted through the AES using a randomly chosen 128-bit key. Each of the resulting six blocks of ciphertext was then transmitted by the UWB TX and the total transmission power for each block over the public channel was measured for each of the 40 Trojan-free, 40 Trojan-I infested, and 40 Trojan-II infested

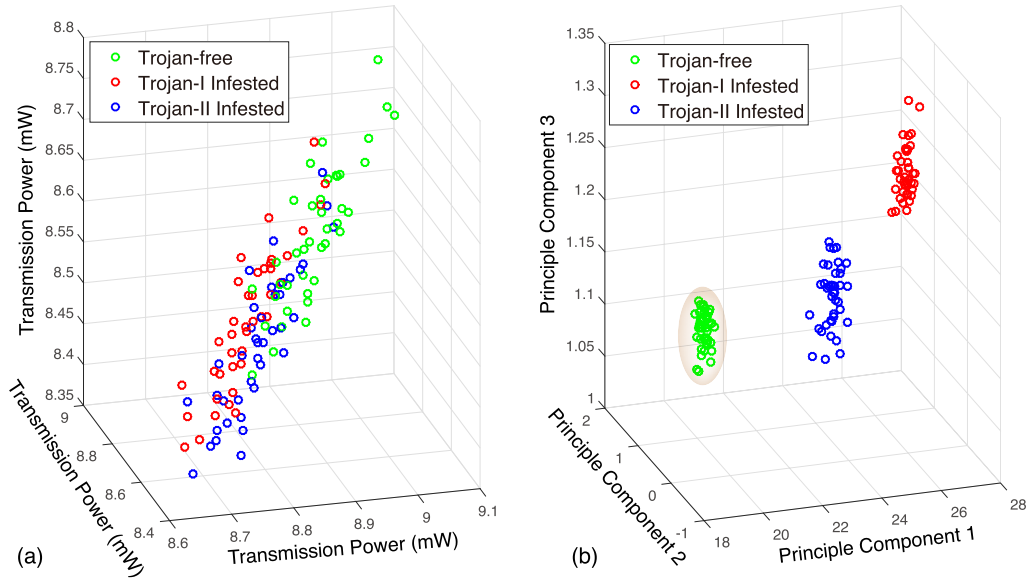


Fig. 21. Projection of hardware Trojan-free and hardware Trojan-infested circuits on a 3-D space where each dimension corresponds to (a) total transmission power for transmitting one ciphertext block, demonstrating that the populations are indistinguishable (three out of the six blocks used in our experiment were randomly chosen; results are similar for any other subset of three) and (b) one of the three top principal components yielded by performing PCA on the total transmission power for transmitting each of the six blocks for all the chips. The MVEE enclosing the hardware Trojan-free population, which can be used to classify a chip as hardware Trojan-free or hardware Trojan-infested, is also shown.

circuits. In Fig. 21(a), we project these populations to a randomly chosen subset of three out of these six measurements (i.e., each dimension reflects the total transmission power when transmitting a 128-bit ciphertext block). Evidently, the three populations fall upon each other and are not distinguishable in this space. This remains the case for all other subsets of three out of the six measurements. This is expected, as the transmission power for each individual block remains within the acceptable specification margins for all of these circuits. In other words, by simply examining transmission power of blocks by individual chips and comparing to some threshold, it is not possible to reveal the presence of a hardware Trojan.

However, when we perform even a very simple statistical processing, such as PCA, of the same information from all the circuits (i.e., the total transmission power for transmitting each of the same six ciphertext blocks as described earlier), things start to become very interesting. In Fig. 21(b), we project again the three chip populations, this time on the three principal components of the original data, which are essentially three eigenvectors of the data covariance matrix and, hence, orthogonal (linearly uncorrelated). Evidently, in this space, the three populations are clearly separable. For example, in Fig. 21(b), we show how a simple one-class classifier, such as an MVEE, can be trained to enclose the population of Trojan-free chips and, subsequently, serve the purpose of deciding whether a new chip under evaluation is Trojan-free (i.e., inside the MVEE) or Trojan-infested (i.e., outside the MVEE).

The reason why PCA can separate the Trojan-free chips from the Trojan-infested ones lies in the orthogonality of the dimensions in the transformed space wherein the original data are projected. Due to this orthogonality, the minute but systematic differences incurred by the hardware Trojans, which are indistinguishable from the random differences incurred by process variation, are amplified and become clearly visible in the transformed space. In essence, projection of the data from the original 6-D linearly correlated space to the transformed 3-D linearly uncorrelated space reveals that the distribution of measurements from Trojan-infested chips does not follow the same characteristics as that of Trojan-free chips. In our case, this is the result of the systematic modulation of amplitude/frequency by hardware Trojan-I/II, respectively, in order to leak the encryption key.

Since each added principal component accounts for as much of the variance left in the data set as possible, while maintaining orthogonality with prior principal components, a small number of dimensions will typically suffice for distinguishing the Trojan-infested from the Trojan-free chips. In our case, the six original measurements resulted in overlapping populations in the 2-D space of the top two principal components and required the third one to be fully separable.

We also attempted to reduce the dimensionality of the original data set in order to expedite detection and reduce the hardware resources required for obtaining and processing the measurements. Specifically, we randomly selected a subset of five out of the six ciphertext blocks of our original data set and used PCA to project this data set to its top three principal components, with the results shown in Fig. 22. As can be observed, when training an MVEE one-class classifier in the transformed 3-D space, it is not possible to enclose the hardware Trojan-free population without also including numerous Trojan-infested chips. This result was consistent for all subsets of five out of the six original measurements and for a large number of different sets of initial six measurements that we experimented with. Overall, for the given hardware Trojan-free chip population, a set of six transmission power measurements

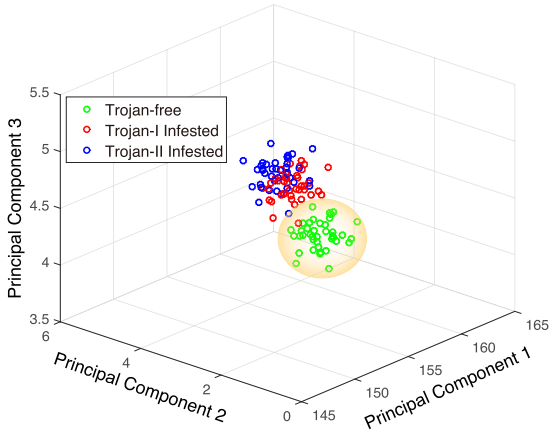


Fig. 22. Projection of Trojan-free and Trojan-infested circuits on a 3-D space of the top principal components after running PCA on a randomly selected subset of five out of the six transmission power measurements.

projected through PCA on a 3-D space appears to be the minimum required for perfectly separating Trojan-free from Trojan-infested chips in our data set using a trained MVEE one-class classifier.

We emphasize that, in the above-mentioned method, training of the MVEE one-class classifier is Trojan-agnostic, i.e., it is performed using measurements obtained *only* from Trojan-free chips, in an unsupervised learning fashion. This is particularly important, as its detection ability is not geared toward specific hardware Trojans. In fact, the only assumption in this case is that a hardware Trojan in wireless cryptographic ICs will have to somehow distort transmission power, which is the only parameter an attacker has access to. Accordingly, the trained MVEE should effectively detect any hardware Trojan that systematically distorts transmission power, independent of how it encodes the leaked information. The two hardware Trojans designed and implemented on our experimentation platform are not used in training the MVEE but are only used to demonstrate the trained classifier's ability to detect hardware Trojans of which it has no prior knowledge.

B. Hardware Trojan Detection via SVM

As an alternative to the combination of PCA and MVEE, which was used mainly for visualizing the statistical "structure" imposed by hardware Trojans and for demonstrating feasibility of hardware Trojan detection through trained classifiers, we also experimented with a more advanced classifier, namely, a one-class SVM. The key strength of SVMs is that they use nonlinear transformation kernels, such as a radial basis function in our experiment, in order to project the original data into a higher dimensional space. In the case of two-class classification (i.e., supervised learning), the objective of the transformation is to make the two populations linearly separable through a hyperplane in the transformed space. In the case of one-class classification, the objective of the transformation is to fit a hyperplane that separates the data from the origin, such that the distance between the hyperplane and the origin is maximized, or to enclose the population within a minimal-volume hypersphere. When projected back to the original (lower dimensional) feature space, this boundary (hyperplane or hypersphere) becomes nonlinear and it is precisely this nonlinearity of the transformation that gives a one-class SVM an edge over the PCA and MVEE approach, as we show next.

TABLE I

TROJAN DETECTION METRICS FOR DIFFERENT NUMBERS OF TRANSMISSION POWER MEASUREMENTS USED TO TRAIN ONE-CLASS SVM

Dimensions used to train the 1-Class SVM	FP	FN
6	0/10	0/80
5	0/10	0/80
4	0/10	0/80
3	0/10	1/80
2	0/10	29/80

We start with the same data set as before, which consists of transmission power measurements for six ciphertext blocks, transmitted by each of our 40 Trojan-free, 40 Trojan-I infested, and 40 Trojan-II infested circuits. We select 30 out of the 40 Trojan-free circuits as the training set and we train a one-class SVM to learn the trusted boundary enclosing this population. Once again, only Trojan-free circuits are used for training, retaining the Trojan-agnostic aspect of this detection method. The remaining 10 Trojan-free circuits along with the 80 Trojan-infested circuits are used to evaluate the performance of the trained classifier. This performance is quantified using two metrics, namely, false positive rate and false negative rate. The former reflects Trojan-free circuits, which are classified as Trojan-infested, while the latter reflects Trojan-infested circuits, which evade detection. This experiment is first performed with all six measurements for each chip and, subsequently, repeated for randomly chosen subsets of five, four, three, and two ciphertext blocks, with the results reported in Table I.

As may be observed, just as in the PCA and MVEE approach, training the one-class SVM with transmission power measurements from six ciphertexts results in a classification boundary that perfectly labels not only the ten previously unseen Trojan-free circuits but also all 40 Trojan-I and 40 Trojan-II infested circuits. In this case, however, training the one-class SVM with transmission power measurements from a subset of five or even four ciphertexts also results in perfect labeling of the validation set, which could not be achieved with the PCA and MVEE approach. In fact, even when a subset of three of these measurements are used for training, all ten Trojan-free circuits in the validation set are still labeled correctly, and only 1 out of 80 Trojan-infested circuits evades detection. When dimensionality of the training set is further reduced to transmission power measurements from two ciphertext blocks, however, the error increases to 29 out of 80 Trojan-infested circuits evading detection, even though none of the 10 Trojan-free circuits is misclassified as Trojan-infested. Overall, these results corroborate experimentally the expectation that, due to the nonlinear nature of the underlying transformation, the more advanced one-class SVM will require

fewer measurements in order to effectively separate the Trojan-free from the Trojan-infested circuits than the simpler PCA and MVEE approach.

VI. Discussion

Before concluding, we would like to note the following.

1) The simplicity of the hardware Trojans described in this paper makes them very stealthy and practical designs. Significantly, these are also the first hardware Trojans reported and demonstrated in actual silicon for attacking wireless cryptographic ICs. Encoding of the leaked information is very straightforward, through modulation of transmission amplitude or frequency, and is easy to decode by a simple Rx who is aware of the rogue encoding. Other options, such as phase modulation and Rx impedance modulation, or more advanced schemes, such as code division multiple access [21] or orthogonal frequency division multiplexing, may also be used to further reduce the distortion induced by the hardware Trojan on the transmission power, at the expense of more complex hardware for staging the attack.
2) The popular statistical side-channel fingerprinting method employed herein for hardware Trojan detection is Trojan-agnostic, i.e., it has no knowledge of and makes no assumptions about the Trojan functionality. The two hardware Trojans designed and implemented in this paper were only used to demonstrate that a one-class classifier, trained with measurements from only Trojan-free designs, can detect these Trojans. In fact, this method will detect any Trojan, which systematically distorts transmission power through any other mechanism, as long as the statistics of the contaminated transmission exhibit different structure than the statistics of the Trojan-free measurements. We note that if the above-mentioned condition does not hold, it will be extremely difficult, if not impossible, for an attacker to decode the leaked information. We also clarify that the use of transmission power as a side-channel measurement for statistical fingerprinting can only protect against attacks which assume that the over-the-air transmission waveform is the only physical parameter the attacker has access to. Hardware Trojans, which use other side channels, relying on physical access to the I/O of the chip, such as MOLES [10], require fingerprinting of other side-channel measurements, such as power consumption, in order to be detected.
3) Statistical side-channel fingerprinting assumes access to measurements from a set of known Trojan-free chips in order to train the one-class classifier. Meeting this requirement, which is prevalent in the hardware Trojan detection literature, can be cumbersome. An expensive option involves destructive delayering and imaging of the ICs in the training set, after the measurements are obtained, in order to certify that they are Trojan-free. Nonintrusive options, including self-referencing [11], self-consistency [22], and the use of trusted on-die process control monitors [23], have also recently been proposed. As such, while not trivial, the obstacle of obtaining trusted fingerprints is possible to overcome.

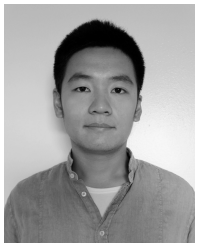
VII. Conclusion

Wireless cryptographic ICs provide a tangible objective and constitute an attractive target for hardware Trojans. Not only do these ICs hold valuable secret information, but also they communicate over public channels, thereby simplifying the attack. Indeed, as shown in this paper, leaking the secret key by hiding it in the wireless transmission power, to which an adversary has access, is fairly straightforward and requires very little circuit modification. More importantly, this can be done without violating any digital, analog, or system-level specifications, rendering traditional test methods ineffective in detecting such hardware Trojans, the impact of which is carefully concealed within the design margins allowed for process variations. In this sense, transmission by a hardware Trojan-infested wireless cryptographic IC appears perfectly legitimate and, in isolation, cannot be differentiated from that of a hardware Trojan-free chip. Nevertheless, due to the systematic nature of the hardware Trojan impact, statistical analysis through either a simple PCA and MVEE or through a more advanced SVM is capable of revealing the presence of a hardware Trojan, without requiring any $a\ priori$ knowledge about the particulars of the attack. The above-mentioned observations were demonstrated using 40 chips from a wireless cryptographic IC design, consisting of an AES encryption core and a UWB TX, which we designed and fabricated in TSMC's 0.35-μm process. To the best of our knowledge, this is the first silicon demonstration of hardware Trojans in wireless cryptographic ICs and the first evaluation of the popular side-channel fingerprinting hardware Trojan detection method using actual measurements.

References

[1] S. Adee, "The hunt for the kill switch," *IEEE Spectr.*, vol. 45, no. 5, pp. 34–39, May 2008.
[2] R. Karri, J. Rajendran, K. Rosenfeld, and M. Tehranipoor, "Trustworthy hardware: Identifying and classifying hardware Trojans," *Computer*, vol. 43, no. 10, pp. 39–46, Oct. 2010.
[3] S. Bhunia, M. S. Hsiao, M. Banga, and S. Narasimhan, "Hardware Trojan attacks: Threat analysis and countermeasures," *Proc. IEEE*, vol. 102, no. 8, pp. 1229–1247, Aug. 2014.
[4] K. Xiao, D. Forte, Y. Jin, R. Karri, S. Bhunia, and M. Tehranipoor, "Hardware Trojans: Lessons learned after one decade of research," *ACM Trans. Design Autom. Electron. Syst.*, vol. 22, no. 1, pp. 6:1–6:23, Nov. 2016.
[5] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, "Trojan detection using IC fingerprinting," in *Proc. IEEE Symp. Secur. Privacy*, May 2007, pp. 296–310.
[6] Y. Jin and Y. Makris, "Hardware Trojan detection using path delay fingerprint," in *Proc. IEEE Int. Workshop Hardw.-Oriented Secur. Trust*, Jun. 2008, pp. 51–57.
[7] R. M. Rad, X. Wang, M. Tehranipoor, and J. Plusquellic, "Power supply signal calibration techniques for improving detection resolution to hardware Trojans," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, Nov. 2008, pp. 632–639.
[8] R. Rad, J. Plusquellic, and M. Tehranipoor, "Sensitivity analysis to hardware Trojans using power supply transient signals," in *Proc. IEEE Int. Workshop Hardw.-Oriented Secur. Trust*, 2008, pp. 3–7.
[9] C. Lamech, J. Aarestad, J. Plusquellic, R. Rad, and K. Agarwal, "REBEL and TDC: Two embedded test structures for on-chip measurements of within-die path delay variations," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, Jun. 2011, pp. 170–177.

[10] L. Lin, W. Burleson, and C. Paar, "MOLES: Malicious off-chip leakage enabled by side-channels," in *Proc. Int. Conf. Comput.-Aided Design*, 2009, pp. 117–122.
[11] D. Du, S. Narasimhan, R. S. Chakraborty, and S. Bhunia, "Self-referencing: A scalable side-channel approach for hardware Trojan detection," in *Proc. Cryptograph. Hardw. Embedded Syst.*, 2010, pp. 173–187.
[12] K. Hu, A. N. Nowroz, S. Reda, and F. Koushanfar, "High-sensitivity hardware Trojan detection using multimodal characterization," in *Proc. IEEE/ACM Design, Autom. Test Eur. Conf. Exhibit. (DATE)*, Mar. 2013, pp. 1271–1276.
[13] S. Narasimhan *et al.*, "Hardware Trojan detection by multiple-parameter side-channel analysis," *IEEE Trans. Comput.*, vol. 62, no. 11, pp. 2183–2195, Aug. 2012.
[14] F. Koushanfar and A. Mirhoseini, "A unified framework for multi-modal submodular integrated circuits Trojan detection," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 1, pp. 162–174, Mar. 2011.
[15] Y. Jin and Y. Makris, "Hardware Trojans in wireless cryptographic ICs," *IEEE Des. Test Comput.*, vol. 27, no. 1, pp. 26–35, Jan./Feb. 2010.
[16] T. Wei and E. Culurciello, "A non-coherent FSK-OOK UWB impulse radio transmitter for clock-less synchronization," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2012, pp. 1295–1298.
[17] [Online]. Available: http://www.powerwerx.com/two-way-radios/handheld-wouxun-radios/high-gain-dual-band-handheld-antenna-reverse-sma.html
[18] I. T. Joliffe, *Principal Component Analysis*. Berlin, Germany: Springer-Verlag, 1986.
[19] N. Moshtagh. (2006). Minimum volume enclosing ellipsoid. GRASP Lab, University of Pennsylvania. [Online]. Available: http://www.seas.upenn.edu/~nima/papers/Mim_vol_ellipse.pdf
[20] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.
[21] D. Chang, B. Bakkaloglu, and S. Ozev, "Enabling unauthorized RF transmission below noise floor with no detectable impact on primary communication performance," in *Proc. IEEE 33rd VLSI Test Symp.*, Apr. 2015, pp. 1–4.
[22] S. Wei and M. Potkonjak, "Self-consistency and consistency-based detection and diagnosis of malicious circuitry," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 9, pp. 1845–1853, Sep. 2014.
[23] Y. Liu, K. Huang, and Y. Makris, "Hardware Trojan detection through golden chip-free statistical side-channel fingerprinting," in *Proc. ACM/IEEE 51st Annu. Design Autom. Conf.*, Jun. 2014, pp. 1–6.



Yu Liu (S'12) received the B.S. degree in electronic science and technology and the M.S. degree in microelectronics and solid-state electronics from Xidian University, Xi'an, China, in 2008 and 2011, respectively. He is currently pursuing the Ph.D. degree with the Electrical Engineering Department, Erik Jonsson School of Engineering and Computer Science, The University of Texas at Dallas, Richardson, TX, USA.

His current research interests include trustworthy wireless cryptographic ICs, analog/mixed signal IC design, VLSI design, and machine learning.

Yier Jin (S'07–M'12) received the B.S. and M.S. degrees in electrical engineering from Zhejiang University, Hangzhou, China, in 2005 and 2007, respectively, and the Ph.D. degree in electrical engineering from Yale University, New Haven, CT, USA, in 2012.

He is currently an Assistant Professor with the Electrical and Computer Engineering Department, University of Central Florida, Orlando, FL, USA. His current research interests include the areas of trusted embedded systems, trusted hardware intellectual property (IP) cores, and hardware–software co-protection on computer systems. He proposed various approaches in the area of hardware security, including the hardware Trojan detection methodology relying on local side-channel information, the postdeployment hardware trust assessment framework, the proof-carrying hardware IP protection scheme, the security analysis on Internet of Things (IoT), and wearable devices with particular emphasis on information integrity and privacy protection in the IoT era.

Dr. Jin is the recipient of Best Paper Awards from the 2015 Design Automation Conference and the 2016 Asia and South Pacific Design Automation Conference.

Aria Nosratinia (S'87–M'97–SM'04–F'10) received the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana–Champaign, Champaign, IL, USA, in 1996.

He has held visiting appointments with Princeton University, Princeton, NJ, USA, Rice University, Houston, TX, USA, and the University of California at Los Angeles, Los Angeles, CA, USA. He is currently the Erik Jonsson Distinguished Professor and an Associate Head of the Electrical Engineering Department with The University of Texas at Dallas, Richardson, TX, USA. His current interests include the broad area of information theory and signal processing, with applications in wireless communications.

Dr. Nosratinia has been a recipient of the National Science Foundation CAREER Award. He is an Area Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, and has served as an Editor of the IEEE TRANSACTIONS ON INFORMATION THEORY, the IEEE SIGNAL PROCESSING LETTERS, the IEEE TRANSACTIONS ON IMAGE PROCESSING, and the *IEEE Wireless Communications*.

Yiorgos Makris (SM'08) received the Diploma degree in computer engineering and informatics from the University of Patras, Patras, Greece, in 1995, and the M.S. and Ph.D. degrees in computer engineering from the University of California at San Diego, San Diego, CA, USA, in 1998 and 2001, respectively.

He has held a faculty appointment with Yale University, New Haven, CT, USA, and a visiting faculty appointment with The University of Washington, Seattle, WA, USA. He is currently a Professor of Electrical Engineering at The University of Texas at Dallas, Richardson, TX, USA, where he leads the Trusted and RELiable Architectures Laboratory. His current research interests include the applications of machine learning and statistical analysis in the development of trusted and reliable integrated circuits and systems, with particular emphasis in the analog/RF domain.

Dr. Makris is a recipient of the 2006 Sheffield Distinguished Teaching Award and Best Paper Awards from the 2013 Design Automation and Test in Europe Conference and the 2015 VLSI Test Symposium. He serves as an Associate Editor of the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY and the *IEEE Design & Test of Computers* Periodical. He has also served as a Guest Editor for the IEEE TRANSACTIONS ON COMPUTERS and the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF VLSI CIRCUITS AND SYSTEMS.