Security Oriented Design Framework for EM Side-Channel Protection in RTL Implementations

Jiaji He, Haocheng Ma, Max Panoff, Hanning Wang, Yiqiang Zhao, Leibo Liu*, *Senior Member, IEEE*, Xiaolong Guo, *Member, IEEE*, and Yier Jin, *Senior Member, IEEE*

***Abstract*—Electromagnetic (EM) side-channel analysis is a powerful attack for extracting secret information from cryptographic hardware implementations. Countermeasures have been proposed at the register-transfer level (RTL), layout level, and device level. However, existing EM radiation modeling and side-channel vulnerability mitigation methods do not consider the structural resilience of original designs, nor do they provide fine-grained security enhancements to those vulnerable submodules/components. These universal solutions may introduce unnecessary overheads on the circuit under protection and may not be optimized for individual designs. In this paper, we propose a design/synthesis for side-channel security evaluation and optimization framework based on the *t*-test evaluation results derived from RTL hardware implementations. While the framework apply to different side-channel leakage, we focus more on EM side channels. Supported by this framework, different RTL implementations of the same cryptographic algorithm will be evaluated for their side-channel resistance. In vulnerable implementations, submodules with the most significant side-channel leakages will be identified. Security design/synthesis rules will then be applied to these vulnerable submodules for security enhancements against side-channel attacks (SCAs). Experiments, including simulations and FPGA implementations on different AES designs, are performed to validate the effectiveness of the proposed framework as well as the security design/synthesis rules.**

***Index Terms*—Cryptographic hardware, Electromagnetic side-channel, Side-channel security, Leakage model, *t*-test.**

I. Introduction

The fast growth of hardware devices results in increased demand for intellectual property (IP) cores with complex connectivity. A lack of security consideration when creating these IPs opens potential vulnerabilities to any design wherein they are used. The impact of cyber-attacks and design flaws in IP cores threatens to overturn the credibility of 3rd-party vendors and increases security risks for end users. One common example of these security risks can be found in cryptographic algorithm implementations. These are widely used to protect information confidentiality and are proven secure against algorithmic attacks. However, the sensitive information in these implementations is still vulnerable to various attacks, such as SCAs [1].

SCAs aim at recovering the information processed in devices by monitoring electrical conditions on a operating device, including timing differences, power consumption, EM radiation, etc. Compared with other physical measurements, the EM-based SCA has significant advantages including non-contact detection, location awareness, and rich frequency information. EM radiations can be generally categorized into two types: direct radiation and modulated radiation [2]. *Direct radiation* is caused directly by current flow with sharp rising/falling edges, while *modulated radiation* occurs when a signal modulates carrier signals which in turn generate EM radiations propagating outwards. Direct EM radiation arises as a consequence of current flows within logic gates or other logic parts inside a circuit, where the currents correlate with the underlying logic operations. There are different implementations of arithmetical operations on hardware, including instruction- or circuit-based operations [3], [4]. For instruction-based operations, SCAs aim at extracting keys from the differences of pipelined execution of instructions in microprocessors. For circuit-based operations, side-channel information is leaked from the differences of logic state changes of physical components.

Countermeasures against SCAs have been proposed in almost all design layers [5]. However, few of them have considered side-channel resilience from different RTL implementations, nor do they provide fine-grained security enhancements to the most vulnerable circuit submodules at RTL. In this paper, we define fine-grain as the submodule level and coarse-grain as the whole circuit level. If circuit designers can identify and address security vulnerabilities at the design stage, circuit implementations' SCA resistance will be improved at low cost [6]. However, with the growing complexity of hardware system designs, the workload is overwhelming for the designers to manually diagnose security vulnerabilities. A few frameworks for RTL side-channel modeling and security evaluation are proposed in [7], [8], [9], but security enhancement strategies are not discussed based on identified fine-grained vulnerabilities. The work demonstrated in paper [10] paves the way for a quantitative security evaluation on a specific cryptographic hardware implementation. In this papeIr, we extend the work in [10] from the following aspects. First, we will try to identify information leakage sources of vulnerable submodules in the circuit. Second, we will integrate the security enhancement into the overall RTL side channel analysis framework to mitigate potential vulnerabilities. Third, although generic

Jiaji He, Hanning Wang and Leibo Liu are with the School of Integrated Circuits, Tsinghua University, Beijing, 100084 China (e-mail: jiaji_he@mail.tsinghua.edu.cn, {wanghn, liulb@tsinghua.edu.cn}).

Haocheng Ma and Yiqiang Zhao are with the School of Microelectronics, Tianjin University, Tianjin, 300072 China (e-mail: {hc_ma, yq_zhao}@tju.edu.cn).

Xiaolong Guo is with the Department of Electrical and Computer Engineering, Kansas State University, USA (e-mail: guoxiaolong@ksu.edu).

Max Panoff and Yier Jin are with the Department of Electrical and Computer Engineering, University of Florida, USA (e-mail:m.panoff@ufl.edu, yier.jin@ece.ufl.edu).

* Corresponding author.

security rules are outlined in paper [10], we present the design/synthesis for trust rules more precisely. Meanwhile, we will validate the effectiveness of these rules. Thus, an efficient RTL evaluation mechanism is urgently needed to elaborately assess the capability of hardware implementations against SCAs and to guide designers in implementing more secure hardware designs.

In this paper, we propose an EM side-channel security evaluation and enhancement framework at RTL to perform a quantitative security assessment, vulnerability identification, and rule-based security enhancement[1]. Circuit designers can leverage the framework to identify potential side-channel vulnerabilities and enhance the security. As the theoretical foundation of the proposed framework, an information leakage model is established based on widely accepted Hamming models and is constructed utilizing the RTL code of a design considering the circuit's structural and functional parameters, including logic components, drive capabilities, switching activities and logic states. Note that these simulated EM radiation traces, referred to as *L-traces*, are not real side-channel radiation. Instead, *L-traces* represents the chip's side-channel behaviors [11], which can be utilized to evaluate EM side-channel information leakage. The framework also adopts the *t*-test [12] together with Test Vector Leakage Assessment (TVLA) [13] to quantify the information leakage. Supported by the framework, SCA resistance levels of different implementations or instances for the same specification are evaluated. Further, for vulnerable implementations, submodules with the most significant side-channel leakage will also be identified. Another key feature of the proposed framework is a set of security-oriented design/synthesis rules. These rules are utilized to optimize the implementations for side-channel security enhancement, following a trade-off between security and other traditional hardware metrics such as power, delay, and area. The contributions of our paper are listed as follows.

- A quantitative EM information leakage model at RTL is established to evaluate the circuit's EM side-channel vulnerabilities.
- The side-channel security level of a circuit design is evaluated using *t*-test metrics. Further, information leakage sources of vulnerable submodules are identified.
- Security rules and security enhancement strategies are proposed to guide circuit designers to realize more side-channel secure hardware implementations.
- The feasibility of the proposed framework is validated both through simulations and FPGA implementations.

The rest of the paper is organized as below. Section II presents the background. The proposed framework and design/synthesis for security rules are presented in Section III. Section IV demonstrates our framework by evaluating representative AES benchmarks. In Section V, we will prove that the design for security rules can enhance the side-channel resistance with intrinsic security. Then in Section VI, we will demonstrate the effectiveness of security synthesis strategies. Conclusions are drawn in Section VII.

[1]The proposed CAD for EM side-channel toolset is open sourced in this link: https://github.com/jinyier/CAD4EM/

II. Background

A. Side-Channel Attacks

The first successful SCA was performed by Kocher [14] in 1996, and in 2001 Quisquater et al. [15] extended SCA to EM radiation and showed that attackers can obtain similar results to other side-channel parameters, such as power consumption. Similar to power SCA, EM SCA methods can be divided into simple EM attack (SEMA), differential EM attack (DEMA), and correlation EM attack (CEMA) based on the statistical algorithms applied to EM radiation signals. SEMA is performed through directly interpreting cryptographic operations by observing different signal patterns of measured EM traces [16]. DEMA can extract secret information from cryptographic devices by comparing measured EM traces with predicted values based on potential key candidates and then searching for peaks that indicate the correct prediction [17]. CEMA, on the other hand, does not exploit the difference of means but utilizes the Pearson correlation coefficient as a distinguisher to quantify the measured EM traces with predicted intermediate values. EM SCA has been proven to be a serious threat to cryptographic algorithms (e.g., AES, DES, RSA) [18] implemented on different types of electronic devices containing smart cards, microprocessors, ASIC, FPGAs, etc. In [19], EM SCA is performed using high-resolution magnetic probes at a close distance to an ASIC die, proving the feasibility of restricting EM SCA to specific parts of a design after locating correct positions. Further, localized EM SCA with high-resolution equipment nullifies validated countermeasures of power SCA, including dual-rail logic [20] and threshold implementations [21]. The majority of existing EM SCA methods aim to crack cryptographic implementations, where the goal is to obtain secret keys, and thus these methods cannot be utilized to identify information leakage sources.

B. Side-Channel Attack Countermeasures

To prevent and mitigate side-channel vulnerabilities, various countermeasures have been proposed targeting different side-channel leakage channels. These side-channel countermeasures can be deployed at different design stages with the key idea to eliminate dependencies between side-channel leakage and internal sensitive data. Popular countermeasures include masking and hiding.

Masking-based countermeasures are developed based on secret sharing [22], where intermediate values are shared using random numbers called *masks*, such that each share alone is independent of the secret. Two masking schemes are employed to strengthen several countermeasures [23], which are based on dynamic relocation of the data within single encryption round and between different round instances at the same time. In [24], a masking structure is introduced to enhance lightweight encryption algorithms against DPA, which can also be applied to DEMA. A protection scheme for DES that blend with rotating masks and secured S-boxes is proposed in [25], and two d_{th} order masking schemes are developed in [26]. Although masking based countermeasures are effective, masking-based modifications generally require

circuit designers to know well the countermeasures' working mechanisms which may not be true for most cases.

Hiding-based countermeasures are applied to make physical leakages of integrated circuits independent of intermediate values and operations performed during cryptographic implementations. Works are trying to randomize the physical leakages, including random precharge logic (RPL) [27], and random delay insertion (RDI) [28]. Others aim to make side-channel information constant, such as wave dynamic and differential logic (WDDL) [29], sense amplifier based logic (SABL) [30], delay based dual-rail (DDPL) [31], etc. Note that the majority of hiding-based countermeasures focus only on power-based SCAs. In [32], the authors proposed a randomization method to counter against localized EM attacks. This method requires additional permutation network circuits which will cause significant performance and area overheads. In [33], an analysis regarding the root-cause of EM leakage from an integrated circuit layout is performed, and a low-level metal routing method is proposed to suppress critical information leakage. Among existing solutions, few of them have focused on the mitigation of circuit structure's vulnerabilities to provide intrinsic resilience against SCAs.

C. Pre-Silicon EM Radiation Modeling and Side-Channel Vulnerability Identification

Identifying information leakage sources is key to apply SCA countermeasures. To evaluate one circuit's EM information leakage at the pre-silicon stage, layout-, gate- and register-transfer level EM radiation modeling and simulation are proposed. At the layout level, the most popular method is to calculate EM radiation using a 3D or planar EM simulator. Although this process is proved accurate, it requires massive computation resources and is very time-consuming, not to mention the fact that layout information is not available at the design stage. At the gate-level and RTL, EM radiation simulations follow similar leakage models, i.e. Hamming Distance (HD) model, Hamming Weight (HW) model, or other improved Hamming models. As for the EM radiation modeling, Peeters et al. [34] present an improved EM leakage model called switching distance model based on the hypothesis that charging (or discharging) the capacitance involves leakage of +1 (or -1). However, the weight allocation does not consider the circuit's structural characteristics and thus limits the accuracy of simulation results.

Supported by the EM side-channel modeling, researchers start to identify leakage sources that are responsible for generating the majority of EM radiation. It is demonstrated that signal leakage is exclusively restricted to a time-span after the active clock edges [19]. Inspired by this observation, the authors in [35] developed a framework utilizing design data from RTL to generate circuit's EM radiation. This framework was validated on FPGA platforms, and the simulated EM traces are consistent with physical measurements. However, this framework cannot be used to evaluate circuit's intrinsic resilience to SCAs. Researchers have studied EM radiation models [36] to analyze side-channel vulnerabilities but no design/synthesis for security rules have been developed to help improve EM side-channel resistance of a target circuit. Still, there lacks a fine-grained EM side-channel radiation analysis framework that can identify the source of information leakage and provide specific feedback on which submodules/parts of the circuit implementation need to be optimized.

Generally, two security metrics are often used to quantitatively evaluate the level of information leakage, namely measurements to disclosure (MtD) and Welch's *t*-test. MtD value represents the least number of waveforms required to break cryptographic implementations by SCA. MtD is usually applied in correlation SCA [37] that calculates the Pearson correlation function between each measured EM trace and a hypothetical EM value. A secret key will be recovered from a large number of correlation computations. Welch's *t*-test [38] and its variants [39], [40] are common approaches to assess if two sets of data are significantly different from each other. For the assessment of information leakage based on simulation methods, *t*-test is more applicable because different variables and stimuli can be applied in the evaluation process. Furthermore, TVLA methodology can also be utilized together with the *t*-test distinguisher to detect statistical dependencies between internal sensitive data and side-channel leakage.

III. Security Oriented Design Framework

The security-oriented design framework for EM side-channel security quantitative evaluation and security enhancement is illustrated in Figure 1 where a cryptographic circuit is used as a sample design under protection. The developed framework mainly includes four parts:

1) RTL design analysis to extract the implementation information for *L-traces* generation (shown in ①);
2) EM radiation simulation and *L-traces* generation (shown in ②);
3) Quantitative assessment on side-channel leakage and vulnerable submodules identification (shown in ③);
4) Security enhancement through the developed design/synthesis for security rules (shown in ④).

In ①, following the TVLA methodology, the RTL implementations are analyzed to extract the logic gates which contribute directly to EM radiation and their drive capabilities. In ②, a set of stimuli are applied to the circuits. Logic gates' switching activities/toggling counts and corresponding logic states are extracted. Using the collected data, *L-traces* can then be calculated utilizing leakage models. Details of ① and ② will be introduced in Section III-A. In ③, a quantitative security assessment is performed for the entire circuit as well as individual submodules within the circuit. Our quantitative assessment leverages the *t*-test method. We will first calculate *t*-test value for the entire circuit as a global security metric indicating whether exploitable EM information leakage exists. If such information leakage is observed, i.e. a large $|t|$ value is derived for the entire circuit, univariate *t*-test evaluations will then be performed on each submodule to decide how much each submodule contributes to the overall information leakage. In this way, circuit designers can identify the major source of information leakage so that corresponding countermeasures can be applied to these submodules, rather than to the whole

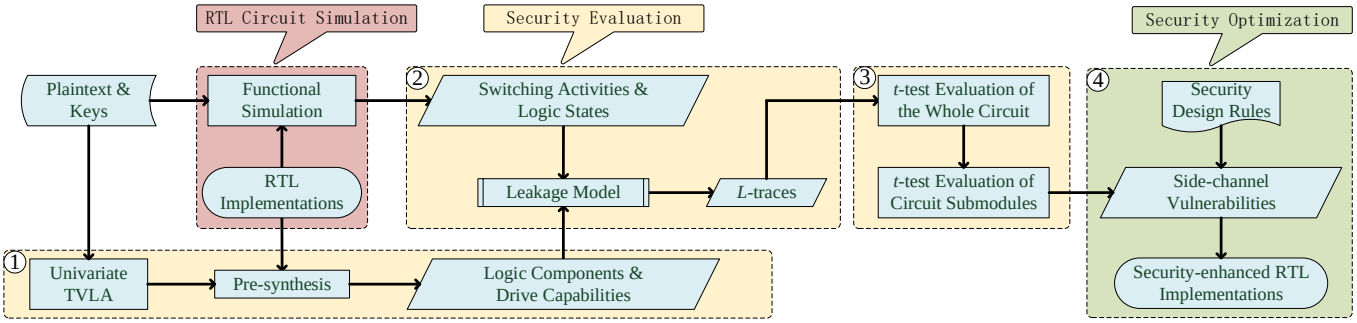


Fig. 1: The proposed RTL security oriented design framework for EM side-channel protection.

circuit. Details about the assessment metrics and the process can be found in Section III-B. Although the evaluation happens at RTL, the identified vulnerable submodules could be mapped to a corresponding radiation heatmap for the mitigation of EM radiation hot-spots [41], which are considered to be the point that has the most obvious EM radiation after configuration. Except for the pre-synthesis in ①, other operations within the above parts can be automatically supported by the framework. In ④, security rules and security enhancement strategies are developed. For the identified submodules with the highest information leakage, circuit designers can enhance the circuit implementation using proposed security rules as elaborated in Section V and VI. Our solution adopts a conditional hiding strategy that tries to implement a secure circuit structure with intrinsic resilience rather than simply adding masks/noise to mask/hide the side-channel leakage.

A. EM Information Leakage Calculation

As shown in ① and ② in Figure 1, in order to establish the correlations between the EM information leakage and the RTL design, we will leverage the concept of *partial Boolean difference* [42]. The *partial Boolean difference* of function $f(x_0, x_1, \ldots, x_{n-1})$ with respect to one variable or a subset of variables is defined in Equation (1):

$$\frac{\partial f}{\partial x_i} = f_{x_i} \oplus f_{x_i'} \tag{1}$$

where $\oplus$ represents exclusive-OR operator and f_{x_i} is the co-factor of f with respect to x_i with:

$$\begin{aligned} f_{x_i} &= f(x_0, x_1, \ldots, 1, \ldots, x_{n-1}) \\ f_{x_i'} &= f(x_0, x_1, \ldots, 0, \ldots, x_{n-1}) \end{aligned} \tag{2}$$

The *partial Boolean difference* of f with respect to x_i shows the condition under which f is sensitive to a change in the input variable x_i. More precisely, if logic values of $\{x_0, \cdots, x_{i-1}, x_{i+1}, \cdots, x_{n-1}\}$ fulfill the condition that $\partial f / \partial x_i = 1$, then any changes in the input value x_i will alter the output value of f.

According to TVLA theory [13], the operations of circuit implementations vary when different input vectors are applied. To assess the side-channel information leakage of a specific circuit implementation, the f is utilized as an information leakage function and x_i is utilized to represent input vectors. The information leakage function f is constructed based on specific leakage models, e.g. HD and HW models, and circuit implementation details that can be extracted from the RTL code. In the proposed framework, the function f is elaborately designed to reflect the differences in EM information leakage, so the calculation results of function f represents *L-traces*. Please note that as long as the target device is fabricated utilizing a standard CMOS technology, the behavior can be approximated by either HD or HW models [43]. Under certain circumstances, the leakage models used by an adversary may not be sufficiently precise, thus mutual information analysis (MIA) or other distinguishers could also be utilized [44] in the proposed framework.

Based on the f function, different variables, i.e. input stimuli, are important for the assessment of side-channel leakage. Regarding leakage models, we have the state-based HW model and the transition-based HD model. Note that the leakage function f is related to the energy required to accomplish the Boolean difference, i.e., state transitions of variables. Therefore, the HD model is more suitable to assess information leakage [45]. Still, we will consider both HD and HW models in the paper. Note that the information leakage function f can be applied to both ASIC and FPGA applications. For ASIC, circuit implementation details may be influenced by both technology libraries and synthesis strategies, thus the leakage function f will be more complicated. For FPGA, fundamental logic components are typically look-up-tables (LUTs) and registers, thus the leakage function f is primarily determined by RTL design. In this paper, the FPGA platform is utilized to validate the feasibility of the information leakage evaluation framework[2].

For circuit implementations, the variables in Equation (1) are stimuli that are provided to the design for extracting switching activities and logic states of all logic components. The process to simulate *L-traces* is illustrated in Algorithm 1, where *RTL_imp* is RTL implementations of the circuits, and $input_v(t)$ represents input vectors including the plaintext and other input stimuli. As shown in the algorithm, RTL implementations will be synthesized first to extract static circuit information including logic components and drive capabilities. Note that the synthesis process is technology node agnostic

[2]The validation and possible adjustments to the framework for ASIC implementations will be investigated in our future work.

Algorithm 1 *L-traces* simulation

Input:
RTL_{imp} ▷ Original circuit implementation.
$input_v(t)$ ▷ Input vectors at time point t.
Output: $L\text{-}traces(input_v, t)$ ▷ The simulated EM radiation at time point t under the stimuli $input_v$.
$logiccomponents \leftarrow RTL_{imp}$;
$DriveCapabilities \leftarrow logiccomponents$;
for Each t **do**
  $List_{SwitchingActivities} \leftarrow RTL_{imp}|input_v(t)$;
  $List_{LogicState} \leftarrow RTL_{imp}|input_v(t)$;
end for
$H(A_i)$, $F_i \leftarrow list_{SwitchingActivities}$; ▷ HW model
A_i, B_i $F_i \leftarrow list_{LogicState}$; ▷ HD model
$L\text{-}traces(input_v, t) \leftarrow H(Ai), A_i, B_i, F_i$.

since we only aim to extract logic components states and their transitions. Under different stimuli, the RTL code is simulated to collect dynamic information such as switching activities and logic states of internal signals. The switching activities within the circuit will cause changes in register states. During each clock cycle, the states of signals under evaluation and their drive capabilities are collected. With all the information, *L-traces* are then calculated to evaluate side-channel vulnerabilities with a chosen leakage model, i.e. HD model or HW model. The same algorithm could be used for either power or current trace generation.

Considering the HW model, the input vector of the circuit is denoted as $input_v$. EM radiation R of the target logic components in the circuit under test is modeled as follows:

$$R(input_v, t) = \sum_{i=1}^{n} H(A_i)|_{input_v} \tag{3}$$

where A_i denotes the logic states of the i_{th} register, t denotes the time of simulation, and $H(A_i)$ means the HW of A_i. The fanout number of the i_{th} register is denoted as F_i, and the simulated EM side-channel leakage, i.e. *L-traces*, is modeled in Equation (4).

$$\textit{L-traces}(input_v, t)_{HW} = \sum_{i=1}^{n} F_i \times H(A_i)|_{input_v} \tag{4}$$

An $N \times L$ matrix T is generated where N denotes the number of different plaintexts and L is the number of sampling points in the simulation. Considering the HD model, the initial and final states of the i_{th} register are denoted as A_i and B_i, respectively, and t represents the moment of the transition under the input vector $input_v$. The transitions of all registers in the circuit under test are modeled as $D(t)$ in Equation (5). The fanout number of the i_{th} register is denoted as F_i, then the simulated EM side-channel trace is modeled in Equation (6).

$$D(t) = \sum_{i=1}^{n} (A_i \oplus B_i)|_{input_v} \tag{5}$$

$$\textit{L-traces}(input_v, t)_{HD} = \sum_{i=1}^{n} F_i \times (A_i \oplus B_i)|_{input_v} \tag{6}$$

Algorithm 2 Identifying Vulnerable Submodules

Input:
$L\text{-}traces(input_v, t)$ ▷ Simulated EM traces.
$Ptext_1$, $Ptext_2 \in input_v(t)$ ▷ Two sets of plaintexts.
τ ▷ Leakage significant level.
Output: M_{vul} ▷ Vulnerable submodules.
/* t-test evaluation of the whole circuit. */
$M_{vul} \leftarrow \emptyset$;
$L\text{-}traces_1 \leftarrow L\text{-}traces(input_v, t)|Ptext_1$;
$L\text{-}traces_2 \leftarrow L\text{-}traces(input_v, t)|Ptext_2$;
t value $\leftarrow$ TFUNCTION($L\text{-}traces_1, L\text{-}traces_2$);
if t value $< \tau$ **then** break;
else
/* t-test evaluation of circuit submodules. */
  divide the whole circuit into n submodules;
  for each $submodule_n$ **do** TFUNCTION;
    if t value$|submodule_n > \tau$ **then**
      M_{vul} = $M_{vul} \cup \{submodule_n\}$;
    else continue;
    end if
  end for
end if
/* Calculate t-test function. */
function TFUNCTION($traces_1$, $traces_2$):
  calculate corresponding μ_i, s_i^2 and n_i parameters
  compute t value
  return t value
end function

B. Side-Channel Vulnerabilities Identification

As demonstrated in ③ in Figure 1, t-test evaluations are performed. Having μ_i, s_i^2 and n_i to be sample mean, variance, and cardinality of set i, respectively, where $i \in \{1, 2\}$, the t value is computed in Equation (7). The p-value of the t-test is calculated as the probability under a t-distribution that the random variable exceeds observed statistic value [38]. The null hypothesis, i.e. no leakage is detected, will be rejected when the p-value is smaller than a threshold, or equivalently when the t value exceeds a corresponding threshold. The rejection criterion of the t value exceeding ± 4.5 is often used in practice [46]. If the t value is outside ± 4.5 range, the test rejects the null hypothesis with confidence greater than 99.999% for a large number of measurements, indicating that the mean of the sets at a particular sample is distinguishable and thus highlighting the existence of side-channel leakage. Since all design decisions along with design flow will impact the final side-channel vulnerability of a fabricated circuit [47], a high $|t|$ value at RTL design may not guarantee the success of SCAs. Also, one design could still be attacked with a $|t|$ value below ± 4.5 threshold. Nonetheless, a high $|t|$ value is generally accepted to be interpreted that one design implementation is more vulnerable to SCAs than other implementations of the same design.

$$t = \frac{\mu_2 - \mu_1}{\sqrt{\frac{s_1^2}{n_2} + \frac{s_1^2}{n_1}}} \tag{7}$$

The calculated t-test value is based on the whole circuit's side-channel information, and thus indicates the level of side-channel vulnerability of the entire circuit. We denote this t calculation process as a coarse-grained side-channel

evaluation in which all submodules of the circuit contribute to global vulnerability together. The reality is that different submodules may have different correlations with sensitive intermediate values, and thus make varying degrees of vulnerability contributions [8]. Therefore, a fine-grained side-channel evaluation method is proposed through which the *t*-test will be applied in a univariate manner for each submodule of the circuit if information leakage is detected in the whole circuit. This fine-grained evaluation can quantify the leakage of submodules in a design, enabling the identification of major side-channel information leakage sources. The *t*-test values for each submodule serve as an indicator of levels of side-channel vulnerability. The calculated *t*-test values of individual submodules will guide circuit designers for more efficient protections in applying countermeasures on these submodules. For large-scale circuit designs, especially system-on-chip (SoC) designs equipped with cryptographic functionalities, there are possibilities that the coarse-grained evaluation indicates no leakage, but the fine-grained evaluation should still be performed for a complete understanding. The fine-grained side-channel evaluation process is illustrated in Algorithm 2. The *t*-test evaluations of circuit submodules are performed according to Equation (7), where the μ_i, s_i^2 and n_i parameters will be replaced with those extracted from submodules under evaluation. The identified vulnerable submodules are listed in M_{vul}, and security enhancement will be performed at the submodule level.

C. Design for Trust and Synthesis for Trust Rules

As we mentioned earlier, existing side-channel security countermeasures are general solutions applicable to different circuits, often at the cost of high overheads. Contrary to these practices, we try to leverage the knowledge that some circuit implementations are more resilient to SCAs than other circuit structures performing the same functionality. Our analyses of circuit implementations help us develop security-oriented design rules and strategies to enhance the circuit's resistance to SCAs. The proposed design/synthesis for security rules can significantly improve circuit security with low-performance overheads by only enhancing the vulnerable circuit submodules rather than protect the entire design. These security rules may also be combined with existing security countermeasures to provide more robust enhancement.

Rule 1: *Registers that store sensitive information should not have high fanout numbers. The rule can be enforced either by duplicating relevant logic components or using max fanout constraints at the synthesis process.*

Justification. The secret information or sensitive intermediate value related registers contribute directly to side-channel information leakage, especially those with large fanout numbers. Registers that have high fanout numbers are utilized for more accessible timing closure. However, this also makes potential side-channel leakage concentrated. By removing sensitive information related to high fanout registers while keeping the overall timing constraints satisfied, the load capacitance of these registers is reduced, resulting in a lower information leakage over critical time points. Thus, side-channel leakage will become more dispersed. Following this rule, we try to reduce the fanout values only for those registers with high fanout numbers. Take HD model-based EM side-channel trace generation for example. The corresponding impact on side-channel leakage is shown as Equation (8), where Δ denotes the reduced fanout value of the i_{th} register. The overall leakage level will be reduced.

$$\textit{L-traces}(input_v, t)_{HD} = \sum_{i=1}^{n} (F_i - \Delta) \times (A_i \oplus B_i)|_{input_v} \quad (8)$$

Rule 2: *Registers that store sensitive information should have similar side-channel behaviors under different stimuli vectors. The rule can be applied by using the stimuli-aware balancing strategy on registers storing sensitive information.*

Justification. As shown in Equations (4) and (6), no matter what leakage models are used, *L-traces* are determined by both logic components and the stimuli $input_v$. For the SCA to succeed, one important precondition is that attackers can control the stimuli to get as many side-channel traces as possible. The differences in side-channel information under different stimuli can then be accumulated and be utilized to break the secret information, with the help of statistical analysis methods. Through eliminating the differences of operations induced by different stimuli, the secret information related registers' side-channel information will remain identical from the perspective of information leakage, thus reducing side-channel leakages. Note that the stimuli-aware balancing strategy only targets those registers showing different side-channel behaviors under different stimuli vectors. The proposed countermeasure is considered to be one conditional hiding countermeasure which will be applied only to those registers that store or process sensitive information. Thus, the proposed countermeasure provides a constant behavior pattern regardless of the stimuli vectors. The proposed method achieves this by considering multiple registers' states simultaneously and introducing balancing registers. Take HD model-based EM side-channel trace generation for example. As depicted in Equation (9), the side-channel leakage of overall registers will keep constant as C after introducing δ registers with fanout F_i' as complementing leakage. The condition $A_i \oplus B_i = 0$ means that the initial and final states of the i_{th} logic component remain unchanged under the input vector $input_v$.

$$\sum_{i=1}^{n} F_i \times (A_i \oplus B_i)|_{input_v} + \sum_{i=1}^{\delta} F_i'|_{input_v \&\& (A_i \oplus B_i = 0)} = C \quad (9)$$

Rule 3: *Large-size register arrays should not be used to process sensitive information. The rule can be enforced by adjusting the synthesis process to use BRAMs (in FPGA implementations) or embedded RAM (in ASIC implementations) instead of large-size register arrays.*

Justification. Large-size register arrays will leak concentrated side-channel information. If large-size register arrays are processing sensitive information, then the information leakage

level will be significant. To implement large-size register arrays, the FPGA development tool will select which RAM to infer based on heuristics that give the best results by default. In modern FPGAs, LUTs in a SLICEM [48] can be utilized to implement distributed RAMs, and FPGAs are generally equipped with plenty of BRAMs for dedicated data storage. The data storing and loading operations from the BRAM will have trivial side-channel leakage compared to distributed RAM, thus the information leakage will be reduced. Therefore, for FPGA implementations, it is encouraged to use BRAMs instead of large-size register arrays or distributed RAM. A similar design rule also applies to ASIC implementations where embedded RAM can be used to replace large-size register arrays[3]. In this way, logic operations related with large-size register arrays will be replaced by memory queries, with much less side-channel footprints [49].

IV. EM Side-Channel Vulnerability Assessment

The experimentation to demonstrate the feasibility of the proposed security-oriented design framework will be divided into three parts. In this section, we will show the evaluation process of the EM side-channel resistance with one cryptographic algorithm's different RTL implementations. For all demonstrations, three representative AES hardware implementations will be evaluated and SCAs will be performed on their FPGA implementations. Please note that the framework can be applied to other cryptographic algorithms.

A. AES Benchmarks

Three open source AES benchmarks are used in our experiments. The first is designed in the light of NIST standard [50], denoted as AES_PUB. The second AES implementation is designed by the developers with Satoh Lab [51], denoted as AES_LUT. The third AES implementation is extracted from the release of a common evaluation platform (CEP) by MIT Lincoln Lab [52], denoted as AES_CEP. These AES implementations are selected to represent different applications of AES modules.

Different RTL descriptions can lead to diverse circuit structures and finally result in different side-channel leakages. The resource utilization is listed in Table I. The corresponding RTL schematics after synthesis are demonstrated in Figure 2. To demonstrate the key related RTL implementations, we focus on the S-box output and related signal processing units. As shown in AES_PUB schematic, signals from the S-box block are stored in registers of a cipher block first and then passed through following encryption operations. In AES_LUT, the signals from the S-box block are passed to mixcolumns operation directly, and in AES_CEP, the signals from S-box are passed directly to an internal flip-flop. While the AES_CEP has a similar schematic with the AES_PUB, the AES_CEP is a pipelined version so the circuit area of AES_CEP is much larger than the other two designs.

[3]Applying this rule in ASIC implementations will be limited by the fabrication technology.

TABLE I: AES Implementation on FPGA Platforms.

	AES_PUB	AES_LUT	AES_CEP
Total registers	1595	649	4798
LUTs	1393	2985	13102

B. Leakage Evaluation of Different AES Implementations

The corresponding EM traces under the same secret keys and plaintexts are simulated in MATLAB based on Equations (4) and (6), and the simulation results are shown in Figure 2(d). The simulated EM radiation from three AES implementations is diverse due to their different RTL schematics. It is clear that compared with AES_LUT, extra EM leakage is generated between adjacent encryption rounds of AES_PUB circuit due to data storage in registers. Under the same working environment and sampling rate, the AES_CEP takes less time to perform the encryption operation due to its pipeline structure, thus as shown in the simulation results of AES_CEP in Figure 2(d), the test points are also less.

To quantitatively evaluate the level of information leakage, *t*-test results are calculated for different AES implementations based on simulation as described in Section III-B, and the *t*-test evaluation results are demonstrated in Figure 3. Please note that the horizontal axis represents the encryption process along with time, and the vertical axis shows the calculated *t*-test values. 10,000 *L*-traces are randomly generated and utilized in the evaluation process. Based on the calculated *t*-test values, we notice that the AES_CEP has the most obvious information leakage as the absolute *t*-test value with HD model exceeds 4.5 already. Although the absolute *t*-test results of AES_PUB and AES_LUT do not reach 4.5, the AES_PUB has more obvious information leakage than the AES_LUT. Besides, for all three AES benchmarks, the *t*-test values with HD model are higher than those with HW model. Our later FPGA implementations verify that the inference in Section III-A that HD model is more suitable to assess the EM information leakage in the proposed framework.

C. Experiments on FPGA Platforms

To demonstrate the effectiveness of the EM side-channel resistance evaluation framework, as mentioned earlier, we perform FPGA measurements on AES_PUB, AES_LUT and AES_CEP implementations. The correlation analysis is performed to check the correspondence of the *t*-test evaluation results with actual attack results.

The experimental setup is illustrated in Figure 4. The device under test (DUT) is a SAKURA-G board [53]. Two FPGAs are integrated and internally connected on the board, where the main FPGA is in charge of conducting operations and the controller FPGA provides the digital stimuli for the main FPGA and controls its operating conditions. When configuring the Xilinx FPGA, the *PlanAhead* [54] software is used to restrict the circuit into certain areas of the FPGA, a common design practice for SoC designs where each IP is allocated separately. Another reason for the area constraint is to make

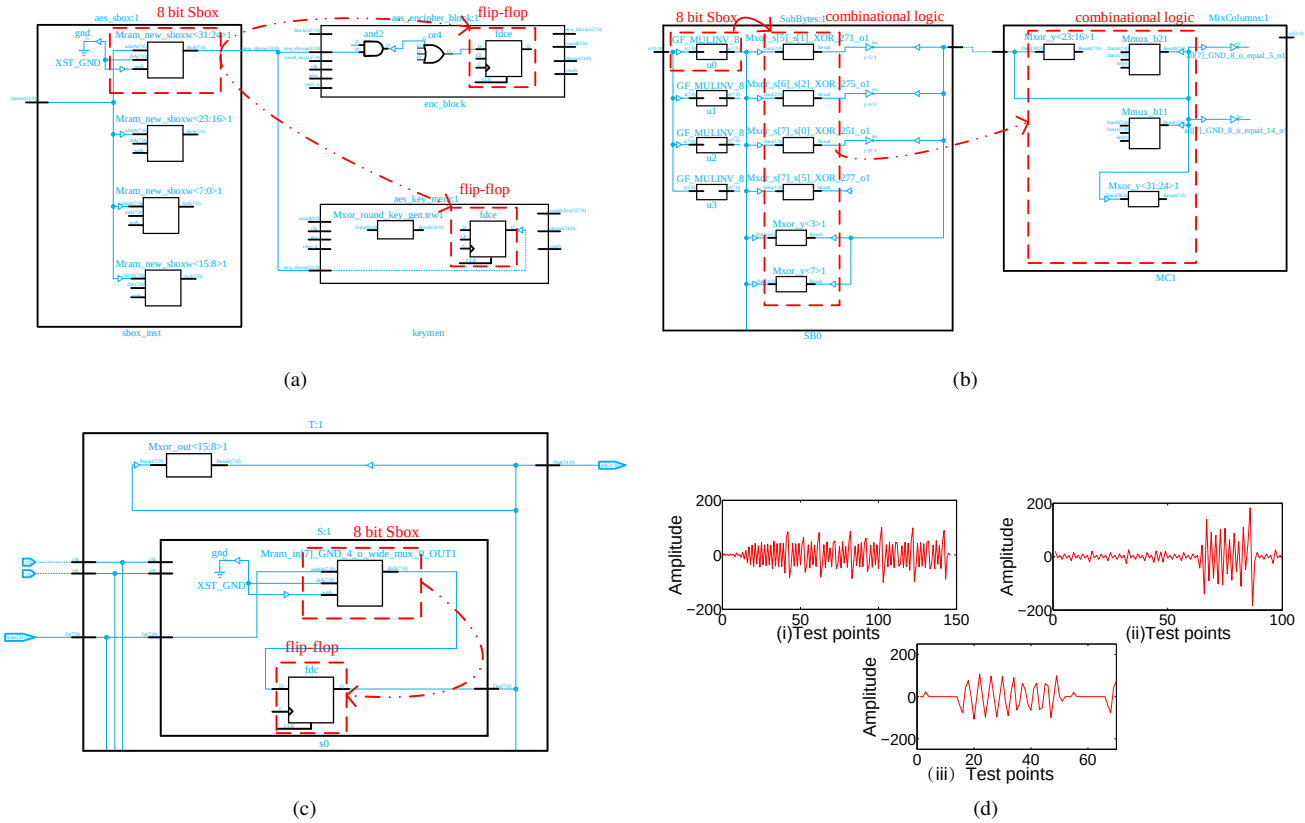


Fig. 2: RTL schematics of (a) AES_PUB, (b) AES_LUT, and (c) AES_CEP. And simulation traces of (d-i) AES_PUB, (d-ii) AES_LUT, and (d-iii) AES_CEP.

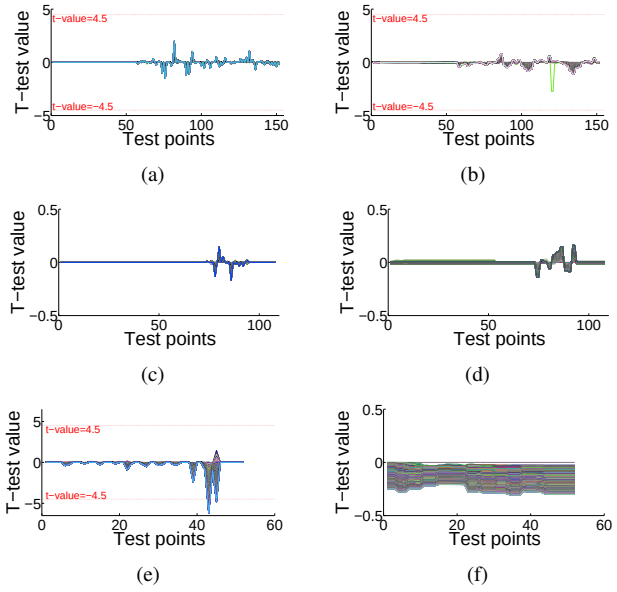


Fig. 3: HD model based *t*-test evaluation results of (a) AES_PUB, (c) AES_LUT, and (e) AES_CEP. And HW model based *t*-test evaluation results of (b) AES_PUB, (d) AES_LUT, and (f) AES_CEP.

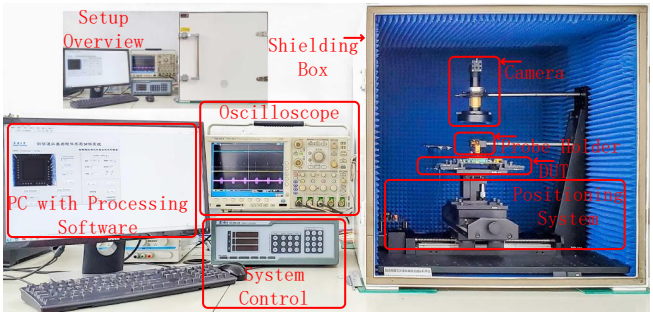


Fig. 4: Experimental setup using FPGA platforms.

sure that the EM radiation of the circuit will be more concentrated. One RF-R LANGER near field probe [55] is utilized, and the probe is fixed on a customized positioning system. The loop of the probe is placed right above the surface of the main FPGA to collect EM radiations. Adjustments of the probe position are made till the amplitude of the waveform in oscilloscope gets the maximum value. This specific position of the probe during the whole experiments should be fixed. The collected EM signals are amplified using an amplifier up to +30 dB magnification. All the equipment used for EM radiation collection is placed in a shielding box to reduce environmental noise.

In total, 640,000 traces are collected for each of the three

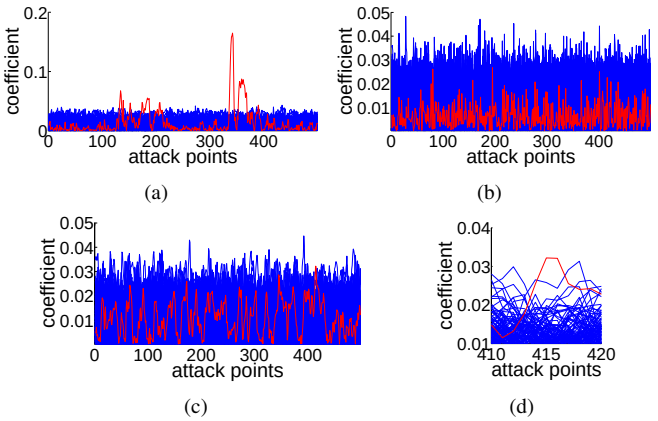


Fig. 5: FPGA CEMA analysis results of (a) AES_PUB, (b) AES_LUT, (c) AES_CEP implementations, and (d) local enlarged AES_CEP implementation.

AES implementations. For the CEMA analysis, we target the S-box output in the last round, thus the hypothetical intermediate values of the S-box output are utilized to calculate the correlation coefficient with the collected traces. The correlation is calculated according to Equation (10), where c is the Pearson correlation coefficient between the *L-traces* and hypothetical intermediate values, n is the number of observations, $\overline{x}$ and $\overline{y}$ are the mean of x and y.

$$c = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}} \tag{10}$$

The CEMA results are illustrated in Figure 5. The horizontal axis indicates the sampling points for one encryption process and the vertical axis shows the correlation coefficient values. From the CEMA results of the AES_PUB, the right key guess (red line) stands out visibly indicating the attack is successful, while the CEMA results of the AES_LUT indicate that the right key guess is still mixed with the wrong key guesses. For the CEMA results of the AES_CEP, the right key guess cannot be easily distinguished from other guesses, so instead of finding the correct key guess from the image, we would look into the theoretical leakage period. The theoretical leakage period is defined as the time when sensitive data are processed by the S-box and stored in related state registers, thus potential information leakage can be observed. In this case, the theoretical leakage period is around the horizontal axis coordinate 415 in Figure 5(c), and we learned that the right key guess is on top of all other guesses as demonstrated in Figure 5(d). The FPGA experimental results of three AES implementations validate the feasibility of the proposed framework in EM side-channel security evaluation. According to the leakage evaluation simulation, the AES_CEP should be most vulnerable to SCAs among all three benchmarks. However, the large circuit size due to the pipelined structure of the AES_CEP helps reduce actual information leakage in CEMA. Overall, the proposed framework provides a circuit designer a reasonable prediction of the EM information leakage level even though the hardware instantiation is still vital to the level of information leakage.

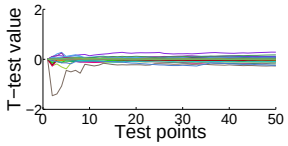


Fig. 6: t-test evaluation results of the *encipher_block*.

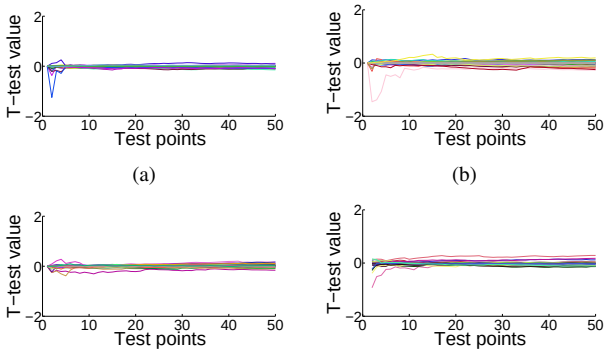


Fig. 7: t-test evaluation results of (a) *S-box0*, (b) *S-box1*, (c) *S-box2*, and (d) *S-box3*.

V. EM Side-Channel Vulnerability Mitigation through Design for Trust

From both simulation and FPGA measurements, we learn that the AES_PUB implementation is more vulnerable to EM SCAs. Therefore, we choose this implementation to further validate the design for security rules. Specifically, security enhancement is performed on the AES_PUB implementation according to the proposed security Rule 1 and Rule 2 in Section III-C.

A. Fine-Grained EM Side-Channel Vulnerability Evaluation

Through parsing the AES_PUB RTL code, it can be divided into several modules along with some peripheral logic. We need to first identify the source of side-channel information leakage from these modules. Based on the circuit hierarchy, the *encipher_block* is a module that implements initial round, main rounds, and final round logic for encryption operations. Other modules include a round key generator and a 32-bit S-box. Similar to the evaluation process for the whole AES design demonstrated in Section IV-B, the HD leakage model is adopted to evaluate the level of information leakage of each module. The leakage evaluation results of the *encipher_block* module alone is shown in Figure 6. Meanwhile, the t-test values of other modules are close to zero, which means that no direct EM information leakage is observed from other modules.

The *encipher_block* is composed of several subfunctions, including S-boxes, mixcolumns, shiftrows, and addroundkey. Given that the function of a 128-bit S-box is performed by four 32-bit operations, denoted as *S-box0*, *S-box1*, *S-box2*, and *S-box3*. The t-test simulations are performed on all these submodules included in the *encipher_block*, including *mixcolumns*, *shiftrows*, *addroundkey*, *S-box0*, *S-box1*, *S-box2*,

Algorithm 3 Security Enhancement of RTL implementations

Input:
1: $RTL_{initial}$ ▷ Initial RTL code of original circuit.
2: $v(reg_{(i)}, t)$ ▷ Value of the i_{th} register at time point t.
Output: RTL_{opt} ▷ Optimized RTL code.
3: $netlist \leftarrow RTL_{initial}$;
4: $reg_{vul(i)}, fanout(reg_{vul(i)}) \leftarrow netlist$;
5: $reg_{add(i)} \leftarrow reg_{vul(i)}$;
6: $fanout(reg_{add(i)}) \leftarrow fanout(reg_{vul(i)})$;
7: **for** Each i **do**
8: **if** $v(reg_{vul(i)}, t) \oplus v(reg_{vul(i)}, t+1)$ **then** none;
9: **else** $v(reg_{add(i)}, t+1) = !\, v(reg_{add(i)}, t)$;
10: **end if**
11: **end for**
12: $RTL_{opt} \leftarrow RTL_{initial} + reg_{add(i)} | fanout(reg_{add(i)})$.

and *S-box3*. The evaluation results of *S-box0*, *S-box1*, *S-box2*, and *S-box3* are shown in Figure 7. The *t*-test values of other logic components are all close to zero.

This fine-grained side-channel vulnerability identification process reveals that the synthesis process may perform different levels of optimizations on different circuit components and often disturb the original circuit structure. For example, *S-box0*, *S-box1*, *S-box2*, and *S-box3* should behave the same at the netlist level because they are crafted the same way in RTL code, however, the reality is that *S-box1* has the highest *t*-test value as demonstrated in Figure 7(b) thanks to the synthesis process. With the consideration that *S-box1* is responsible for the computation of the [095:064] bits within the 128-bit key, if an attacker starts with the breaking of the keys that fall into this range, the success rate will be higher. As a consequence, this logic component should be protected first.

B. Security Enhancement of Vulnerable RTL Implementations

After the most vulnerable submodules are identified, design for security rules can then be applied to help enhance the design for security protection. Again, the security enhancement process using these design rules does not require the circuit designers to have background knowledge of side-channel countermeasures. The design for security rule-based security enhancement process is illustrated in Algorithm 3. In Line 3, the netlist file can be obtained from a technology node agnostic pre-synthesis process. Based on the evaluation framework proposed in Section III, the logic components and drive capabilities will be identified and extracted. Following the security design rules proposed in Section III-C, as shown from line 7 to line 12, the added logic components are integrated with the original circuit for intermediate value hiding.

For the AES_PUB design, the security enhancement happens at RTL to eliminate potential vulnerability and to reduce side-channel leakage. To make sure the security enhancement optimization will be reflected at the netlist level, we use low-level primitive descriptions when enhancing the RTL design, as illustrated in Listing 1. Vulnerable registers are optimized through adding extra registers and *if* statement controlled logic to *S-box1* related logic, and LUTs are added to balance the fanouts, as illustrated in Listing 2. In total, 32 6-input LUTs and 8 4-input LUTs are utilized to optimize 28 registers that have 8 fanout numbers in the original RTL code, and a total 14 4-input LUTs are utilized to optimize 4 registers that have 14 fanout numbers in the original RTL code. To reduce the added logic's influence on the original implementation, *if* statements are utilized for the monitoring of intermediate value changes within the nonblocking assignments related to *S-box1*, thus the newly introduced LUTs will only be activated when there are vulnerable operations.

Listing 1: Code Segment of the Added LUT Primitive.

```
reg [31:0] block_w1_reg_inv;
/*newly added reg*/
/* part1 32 fanout balancing */
parameter N_0=4;
genvar i_0;
generate
for (i_0=0;i_0<N_0;i_0=i_0+1)
begin
    LUT6 #(.INIT(64'h0101101010110100))
    /* specify LUT Contents 1*/
    LUT6_inst_0 (
    .O(block_w1_reg_inv_o[0+i_0*8]), /*LUT general output*/
    .I0(block_w1_reg_inv[0+i_0*8]),  /*LUT input*/
    .I1(block_w1_reg_inv[1+i_0*8]),  /*LUT input*/
    .I2(block_w1_reg_inv[2+i_0*8]),  /*LUT input*/
    .I3(block_w1_reg_inv[3+i_0*8]),  /*LUT input*/
    .I4(block_w1_reg_inv[4+i_0*8]),  /*LUT input*/
    .I5(block_w1_reg_inv[5+i_0*8])   /*LUT input*/
    );
    LUT6 #(.INIT(64'h0101101010110100))
    /* specify LUT Contents 2*/
    LUT6_inst_1 ( ... )
...
end
```

Listing 2: Code Segment of the Added Control Logic.

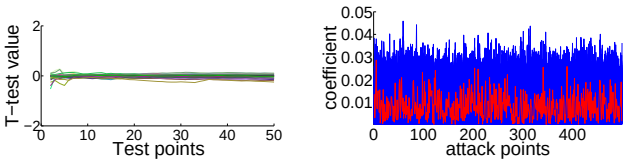
```
if (block_w1_we)
begin
    block_w1_reg <= block_new[095 : 064];
if ( !(block_new[64]^block_w1_reg[0]) )
    begin block_w1_reg_inv[0] <= !block_w1_reg_inv[0]; end
if ( !(block_new[65]^block_w1_reg[1]) )
    begin block_w1_reg_inv[1] <= !block_w1_reg_inv[1]; end
if ( !(block_new[66]^block_w1_reg[2]) )
    begin block_w1_reg_inv[2] <= !block_w1_reg_inv[2]; end
if ( ... ); end
...
end
```

C. Evaluation of the Enhanced RTL Implementations

Two assessments, simulation and actual CEMA, are performed to validate the effectiveness of the design security enhancement process. The *t*-test evaluation simulation is performed first on the enhanced AES_PUB implementation. The *t*-test results of the single *S-box1* is reduced to close to zero, and the whole enhanced AES_PUB implementation's *t*-test result is shown in Figure 8(a). Compared with the HD results of the AES_PUB shown in Figure 3(a), the information leakage is significantly reduced compared with the non-enhanced version. The enhanced AES_PUB implementation is then configured into the FPGA board, and the CEMA with the same experiment setup demonstrated in Section IV-C. The attack result is shown in Figure 8(b), where the right key guess (red line) is hidden in the correlation results. It is a clear indication that the AES_PUB implementation after the

TABLE II: Security Enhancement Results Comparison.

Module name	Total registers	Δ	High fanout registers	LUTs	Δ	Power consumption (W)	Δ	Net skew (ns)	Δ	Max delay (ns)	Δ
AES_PUB	1595	/	13	1393	/	0.153	/	0.421	/	1.896	/
Optimized AES_PUB	1660	+4.08%	10	1502	+7.82%	0.162	+5.88%	0.412	-2.14%	1.896	0%
AES_CEP	4798	/	1364	13102	/	0.747	/	0.415	/	1.890	/
Optimized AES_CEP	4798	0%	0	3108	-76.27%	0.695	-6.96%	0.423	+1.93%	1.911	+1.11%



(a) (b)

Fig. 8: Side-channel evaluation of the enhanced AES_PUB (a) *t*-test evaluation results, and (b) CEMA results of the optimized AES_PUB.

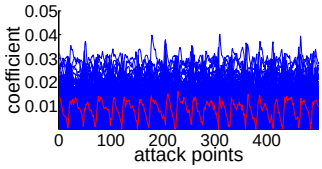


Fig. 9: CEMA results of the optimized AES_CEP.

security enhancement has a much higher SCA resistance level compared to the original design (see Figure 5(a)).

During the security enhancement process, more logic components are added to the original circuit implementation, thus there will be inevitably area and power overheads. As listed in Table II, the original AES_PUB is implemented using 1595 total registers with 13 high fanout registers and 1393 LUTs. Note that in this paper, the high fanout register is defined as a register that has a fanout number $\geq$ 25. The FPGA power consumption at 48MHz is 0.153W. For the optimized version of the AES_PUB implementations, 1660 total registers with 10 high fanout registers and 1502 LUTs are utilized. The FPGA power consumption at 48MHz is 0.162W. The optimized version consumes 4.08% more registers and 7.82% more LUTs, while the power consumption overhead is 5.88%. Here we exploit two main metrics, i.e. net skew and maximum delay, to evaluate the timing performance of different design implementations. The net skew and maximum delay are the maximum delay difference and timing delay taken by the clock signals from the clock source to the sequential cells, respectively. For the original AES_PUB, the net skew and maximum delay are $0.421ns$ and $1.896ns$. After optimization, the net skew is changed to $0.412ns$, while the maximum delay remains unchanged. Thus, it can be concluded that the optimized implementation has no performance degradation.

VI. EM Side-Channel Protection through Synthesis for Trust

From the above experiment results, the AES_CEP implementation has distinguishable information leakage from the perspectives of both simulation and FPGA measurement. Note that the circuit size of AES_CEP implementation is much bigger than the other two implementations, and the large size itself provides a level of protection (by adding more noise). In this section, we will use the AES_CEP as an example to demonstrate the effectiveness of Rule 3 from Section III-C.

A. Synthesis for Trust of Vulnerable RTL Implementations

In the AES_CEP implementations, circuit designers adopted *case* statements among the clock-controlled *always* block. A segment of the RTL code is shown in Listing 3. Other implementations, i.e. AES_PUB and AES_LUT, utilize *assign* statements and other logic operators. This *case* coding style can be instantiated into a single-port distributed RAM in FPGA implementation [56]. Thus, in order to use BRAMs instead of large-size register arrays or distributed RAM, we guide the synthesis process to instantiate the *case* statement into BRAMs on the FPGA to realize corresponding logic functions. Following Rule 3 in Section III-C, the security of AES_CEP is enhanced by resynthesizing the RTL code using BRAM logic components to reduce potential information leakage. Although the security optimization that utilizes Rule 3 may not be as accurate as Rule 1 and Rule 2, the synthesis for trust can help enhance the overall security of the RTL implementations.

Listing 3: Segment of RTL code in AES_CEP.

```
1  module S (clk, rst, ins, outs);
2      input clk;
3      input rst;
4      input [7:0] ins;
5      output reg [7:0] outs;
6      always @ (posedge clk or posedge rst)
7          if (rst)
8              outs <= 8'd0;
9          else
10     /* Total 256 lines of case statements */
11         case (ins)
12             8'h00: outs <= 8'h63;
13             8'h01: outs <= 8'h7c;
14             8'h02: outs <= 8'h77;
15              ...
16         endcase
17 ...
18 endmodule
```

B. Evaluation of the Enhanced RTL Implementations

The security enhancement happens during the synthesis process, so the *t*-test evaluation on the RTL code will not be affected. After the synthesis for trust, the CEMA failed as shown in Figure 9. As shown in the CEMA results, the right key guess is mixed with the wrong key guesses, indicating that the attack is failed. Compared with the results of Figure 5(c),

the resistance against CEMA is significantly improved after the security enhancement process.

The original AES_CEP is implemented using 4798 total registers with 1364 high fanout registers and 13102 LUTs. The FPGA power consumption at 48MHz is 0.747W. For the optimized version of the AES_CEP implementations, 4798 total registers without high fanout register and 3108 LUTs are utilized. The FPGA power consumption at 48MHz is 0.695W. The optimized version does not consume more registers and consumes only 23.72% LUTs compared with the original implementations, while the power consumption is 93% compared with the original implementation. The results are listed in Table II. For the original AES_CEP, the net skew and maximum delay are $0.415ns$ and $1.890ns$. After optimization, the two metrics are changed to $0.423ns$ and $1.911ns$, respectively. The higher maximum delay value is caused by the increase of the net skew. If the net skew is higher, the performance will be worse. Thus, it can be concluded that our security enhancement introduces trivial performance degradation.

VII. Conclusion and Future Work

In this paper, an EM side-channel security aware design framework is developed and validated. This framework employs quantitative assessment and security enhancement. Side-channel vulnerabilities of three different AES benchmarks are evaluated and major leakage sources are identified to validate the practicality of the proposed framework. Design for security and synthesis for security rules are proposed to guide the security enhancement of vulnerable implementations, and assessment results validate the effectiveness of the proposed design rules. In the future, the framework will be further enhanced to automate the security enhancement process. Automatic EDA tools will be developed to help optimize vulnerable RTL code. More security rules will be proposed to reduce side-channel information leakage, and new designs will also be evaluated. The combination of the proposed framework with other countermeasures will be investigated.

Acknowledgments

This work is supported in part by the National Key R&D Program of China (Grant No. 20218YFB3100903)and in part by the National Natural Science Foundation of China (Grant No. 620041122202101), the National Science and Technology Major Project of the Ministry of Science and Technology of China (Grant No. 2018ZX01027101-002), the National Natural Science Foundation of China (Grant No. 62004112), and the China Postdoctoral Science Foundation (Grant No. 2019TQ0167).

References

[1] G. Joy Persial, M. Prabhu, and R. Shanmugalakshmi, "Side channel attack survey," *Int. J. Adv. Sci. Res. Rev*, vol. 1, no. 4, pp. 54–57, 2011.

[2] H. Li, A. T. Markettos, and S. Moore, "Security evaluation against electromagnetic analysis at design time," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2005, pp. 280–292.

[3] K. Atasu, L. Breveglieri, and M. Macchetti, "Efficient AES implementations for ARM based platforms," in *Proceedings of the 2004 ACM Symposium on Applied Computing*, ser. SAC '04. New York, NY, USA: ACM, 2004, pp. 841–845.

[4] P. Chodowiec and K. Gaj, "Very compact FPGA implementation of the AES algorithm," in *Cryptographic Hardware and Embedded Systems - CHES 2003*, C. D. Walter, Ç. K. Koç, and C. Paar, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 319–333.

[5] F.-X. Standaert, "Introduction to side-channel attacks," in *Secure Integrated Circuits and Systems*. Springer, 2010, pp. 27–42.

[6] L. Batina, N. Mentens, and I. Verbauwhede, "Side-channel issues for designing secure hardware implementations," in *11th IEEE International On-Line Testing Symposium*, July 2005, pp. 118–121.

[7] S. A. Huss, M. Stöttinger, and M. Zohner, "AMASIVE: an adaptable and modular autonomous side-channel vulnerability evaluation framework," in *Number Theory and Cryptography*. Springer, 2013, pp. 151–165.

[8] M. T. He, J. Park, A. Nahiyan, A. Vassilev, Y. Jin, and M. Tehranipoor, "RTL-PSC: Automated power side-channel leakage assessment at register-transfer level," in *2019 IEEE 37th VLSI Test Symposium (VTS)*. IEEE, 2019, pp. 1–6.

[9] D. Šijačić, J. Balasch, B. Yang, S. Ghosh, and I. Verbauwhede, "Towards efficient and automated side-channel evaluations at design time," *Journal of Cryptographic Engineering*, pp. 1–15, 2020.

[10] J. He, H. Ma, X. Guo, Y. Zhao, and Y. Jin, "Design for EM side-channel security through quantitative assessment of RTL implementations," in *25th Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2020, pp. 62–67.

[11] S. B. Rs, F. Grkaynak, E. Oswald, and B. Preneel, "Power-analysis attack on an ASIC AES implementation," in *International Conference on Information Technology: Coding and Computing*, 2004, p. 546.

[12] T. Schneider and A. Moradi, "Leakage assessment methodology - a clear roadmap for side-channel evaluations," 2015.

[13] G. C. Becker, J. Cooper, E. DeMulder, G. Goodwill, J. Jaffe, G. Kenworthy, T. Kouzminov, A. Leiserson, M. E. Marson, P. Rohatgi, and S. Saab, "Test vector leakage assessment (TVLA) methodology in practice (extended abstract)," 2013.

[14] P. C. Kocher, "Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems," in *Annual International Cryptology Conference*. Springer, 1996, pp. 104–113.

[15] J.-J. Quisquater and D. Samyde, "Electromagnetic analysis (EMA): Measures and counter-measures for smart cards," in *International Conference on Research in Smart Cards*. Springer, 2001, pp. 200–210.

[16] K. Gandolfi, C. Mourtel, and F. Olivier, "Electromagnetic analysis: Concrete results," in *International workshop on cryptographic hardware and embedded systems*. Springer, 2001, pp. 251–261.

[17] J. Pan, J. G. Van Woudenberg, J. I. Den Hartog, and M. F. Witteman, "Improving DPA by peak distribution analysis," in *International Workshop on Selected Areas in Cryptography*. Springer, 2010, pp. 241–261.

[18] T. Kasper, D. Oswald, and C. Paar, "EM side-channel attacks on commercial contactless smartcards using low-cost equipment," in *International Workshop on Information Security Applications*. Springer, 2009, pp. 79–93.

[19] J. Heyszl, D. Merli, B. Heinz, F. De Santis, and G. Sigl, "Strengths and limitations of high-resolution electromagnetic field measurements for side-channel analysis," in *International Conference on Smart Card Research and Advanced Applications*. Springer, 2012, pp. 248–262.

[20] V. Immler, R. Specht, and F. Unterstein, "Your rails cannot hide from localized EM: how dual-rail logic fails on FPGAs," in *International Conference on Cryptographic Hardware and Embedded Systems*. Springer, 2017, pp. 403–424.

[21] R. Specht, V. Immler, F. Unterstein, J. Heyszl, and G. Sig, "Dividing the threshold: Multi-probe localized EM analysis on threshold implementations," in *2018 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. IEEE, 2018, pp. 33–40.

[22] S. Chari, C. S. Jutla, J. R. Rao, and P. Rohatgi, "Towards sound approaches to counteract power-analysis attacks," in *International Cryptology Conference on Advances in Cryptology*, 1999, pp. 398–412.

[23] P. Maistri, S. Tiran, P. Maurine, I. Koren, and R. Leveugle, "Countermeasures against EM analysis for a secured FPGA-based AES implementation," in *2013 International Conference on Reconfigurable Computing and FPGAs (ReConFig)*. IEEE, 2013, pp. 1–6.

[24] W. Gong, P. Choi, B. C. Kim, and K. K. Dong, "Analysis of masking effects on DPA countermeasure for lightweight cryptographic algorithms," in *Soc Design Conference*, 2016, pp. 315–316.

[25] Z. Zhuang, J. Chen, and H. Zhang, "A countermeasure for DES with both rotating masks and secured S-boxes," in *Tenth International Conference on Computational Intelligence and Security*, 2014, pp. 410–414.

[26] R. Zhang, S. Qiu, and Y. Zhou, "Further improving efficiency of higher order masking schemes by decreasing randomness complexity," *IEEE Transactions on Information Forensics & Security*, vol. 12, no. 11, pp. 2590–2598, 2017.
[27] M. Bucci, M. Guglielmo, R. Luzzi, and A. Trifiletti, "A power consumption randomization countermeasure for DPA-resistant cryptographic processors," in *Integrated Circuit and System Design, Power and Timing Modeling, Optimization and Simulation; International Workshop, Patmos 2004, Santorini, Greece, September 15-17, 2004, Proceedings*, 2004, pp. 481–490.
[28] M. Bucci, R. Luzzi, M. Guglielmo, and A. Trifiletti, "A countermeasure against differential power analysis based on random delay insertion," in *IEEE International Symposium on Circuits and Systems*, 2005, pp. 3547–3550 Vol. 4.
[29] K. Tiri, M. Akmal, and I. Verbauwhede, "A dynamic and differential CMOS logic with signal independent power consumption to withstand differential power analysis on smart cards," in *Solid-State Circuits Conference, 2002. Esscirc 2002. Proceedings of the European*, 2002, pp. 403–406.
[30] K. Tiri and I. Verbauwhede, "A logic level design methodology for a secure DPA resistant ASIC or FPGA implementation," in *Design, Automation and Test in Europe Conference and Exhibition, 2004. Proceedings*, 2004, p. 10246.
[31] M. Bucci, L. Giancane, R. Luzzi, G. Scotti, and A. Trifiletti, "Delay-based dual-rail precharge logic," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 19, no. 7, pp. 1147–1153, 2011.
[32] G. Li, V. Iyer, and M. Orshansky, "Securing AES against localized EM attacks through spatial randomization of dataflow," in *2019 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, 2019, pp. 191–197.
[33] D. Das, M. Nath, B. Chatterjee, S. Ghosh, and S. Sen, "Stellar: A generic em side-channel attack protection through ground-up root-cause analysis." in *HOST*, 2019, pp. 11–20.
[34] E. Peeters, F.-X. Standaert, and J.-J. Quisquater, "Power and electromagnetic analysis: Improved model, consequences and comparisons," *Integration, the VLSI journal*, vol. 40, no. 1, pp. 52–60, 2007.
[35] 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 System (TVLSI)*, vol. 25, no. 10, pp. 2939–2948, 2017.
[36] A. Kumar, C. Scarborough, A. Yilmaz, and M. Orshansky, "Efficient simulation of EM side-channel attack resilience," in *2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, Nov 2017, pp. 123–130.
[37] T. Sugawara, Y.-i. Hayashi, N. Homma, T. Mizuki, T. Aoki, H. Sone, and A. Satoh, "Spectrum analysis of cryptographic modules to counteract side-channel attacks," *EMC '09*, vol. 6, 01 2009.
[38] A. A. Ding, C. Chen, and T. Eisenbarth, "Simpler, faster, and more robust t-test based leakage detection," in *International workshop on constructive side-channel analysis and secure design*. Springer, 2016, pp. 163–183.
[39] A. A. Ding, L. Zhang, F. Durvaux, F.-X. Standaert, and Y. Fei, "Towards sound and optimal leakage detection procedure," in *International Conference on Smart Card Research and Advanced Applications*. Springer, 2017, pp. 105–122.
[40] W. Yang, H. Zhang, Y. Gao, A. Fu, and S. Wei, "Side-channel leakage detection based on constant parameter channel model," in *2020 IEEE 38th International Conference on Computer Design (ICCD)*. IEEE, 2020, pp. 553–560.
[41] J. Heyszl, S. Mangard, B. Heinz, F. Stumpf, and G. Sigl, "Localized electromagnetic analysis of cryptographic implementations," in *Cryptographers' track at the RSA conference*. Springer, 2012, pp. 231–244.
[42] N. Mohyuddin, E. Pakbaznia, and M. Pedram, *Probabilistic Error Propagation in a Logic Circuit Using the Boolean Difference Calculus*, 2011.
[43] B. Gierlichs, L. Batina, P. Tuyls, and B. Preneel, "Mutual information analysis," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2008, pp. 426–442.
[44] N. Veyrat-Charvillon and F.-X. Standaert, "Mutual information analysis: how, when and why?" in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2009, pp. 429–443.
[45] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2004, pp. 16–29.
[46] B. J. Gilbert Goodwill, J. Jaffe, P. Rohatgi *et al.*, "A testing methodology for side-channel resistance validation," in *NIST non-invasive attack testing workshop*, vol. 7, 2011, pp. 115–136.
[47] H. Ma, J. He, Y. Liu, Y. Zhao, and Y. Jin, "CAD4EM-P: Security-driven placement tools for electromagnetic side channel protection," in *Asian Hardware Oriented Security and Trust (AsianHOST)*, 2019.
[48] Xilinx, "UltraScale Architecture Configurable Logic Block," https://www.xilinx.com/support/documentation/user_guides/ug574-ultrascale-clb.pdf, 2017.
[49] L. Zhang, D. Mu, W. Hu, Y. Tai, J. Blackstone, and R. Kastner, "Memory-based high-level synthesis optimizations security exploration on the power side-channel," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2019.
[50] Secworks, "NIST document FIPS 197 based AES design," https://github.com/secworks/aes, 2014.
[51] S. Lab, "Lookup tabel based AES design," http://satoh.cs.uec.ac.jp/SAKURA/hardware/SAKURA-G.html, 2017.
[52] L. LABORATORY, "CEP AES design," https://github.com/mit-ll/CEP, 2018.
[53] SAKURA, *http://satoh.cs.uec.ac.jp/SAKURA/index.html*.
[54] PlanAhead, *https://www.xilinx.com/products/design-tools/planahead.html*.
[55] LANGER, *https://www.langer-emv.com/en/index*.
[56] XSTUserGuide, *https://www.xilinx.com/support/documentation/sw_manuals/xilinx11*.

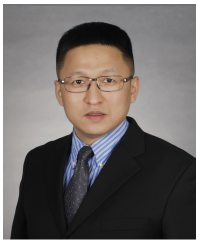
Jiaji He received the B.S. degree in electronic science and technology and the M.S. and Ph.D. degrees in microelectronics from Tianjin University, Tianjin, China, in 2013, 2015 and 2019, separately. He is currently a postdoctoral research fellow since 2019 with the School of Integrated Circuits, Tsinghua University, China. His research interests are digital circuit design, cryptographic chip design, hardware security, on-chip security primitive design, side-channel vulnerabilities mitigation, etc.

Haocheng Ma received the B.S. degree in microelectronics from Tianjin University, Tianjin, China in 2017, where he is currently pursuing the Ph.D. degree at the School of Microelectronics. His current research interests include digital circuit design, hardware security, and EDA for security.

Max Panoff received the B.E. in electrical engineering from Stevens Institute of Technology in Hoboken, New Jersey, USA in 2018. He is currently pursuing a Ph.D in electrical engineering at the University of Florida. His research focuses in hardware security, especially side channel analysis.

Hanning Wang received the B.S. degree in Network Engineering and the M.S. degree in Computer Science and Technology from the China University of Geosciences in 2009 and 2012, respectively. He is currently working as a senior engineer for the School of Integrated Circuits, Tsinghua University. His current research interests include hardware security and cryptographic engineering.

Yiqiang Zhao received the B.S. in semiconductor physics and device, the M.S. in microelectronics, and the Ph.D. degree in microelectronics and solid-state electronics from Tianjin University, Tianjin, China, in 1988, 1991 and 2006, respectively. In 1991, he joined Jinhang Technical Physics Institute, Tianjin, China, where he was responsible for analog and mixed signal circuit design. Since 2001, he has been with the School of Electronic Information Engineering and the School of Microelectronics, Tianjin University, where he is currently a professor. His research interests include mixed-signal integrated circuits, security chips and hardware security.



Leibo Liu (M'10-SM'17) received the B.S. degree in electronic engineering and the Ph.D. degree with the Institute of Microelectronics, both from Tsinghua University, Beijing, China, in 1999 and 2004, respectively. He is currently a Full Professor with the School of Integrated Circuits, Tsinghua University. His current research interests include reconfigurable computing, mobile computing, and very large-scale integration digital signal processing.

Xiaolong Guo (S'14-M'20) is an assistant professor in the Department of Electrical and Computer Engineering at Kansas State University (KSU). He received his PhD degree in Electrical and Computer Engineering from University of Florida (UF) in 2019. His research focuses on detecting hardware or computer vulnerabilities using formal verification and program analysis. He has been recognized with Best Paper Awards at the 2020 AsianHost and the 2019 DATE, and Best Paper Candidate at ASP-DAC 2021.

Yier Jin (M'12-SM'19) is an Associate Professor and IoT Term Professor in the Department of Electrical and Computer Engineering (ECE) in the University of Florida (UF). He received his PhD degree in Electrical Engineering in 2012 from Yale University after he got the B.S. and M.S. degrees in Electrical Engineering from Zhejiang University, China, in 2005 and 2007, respectively. His research focuses on the areas of hardware security, embedded systems design and security, trusted hardware intellectual property (IP) cores and hardware-software co-design for modern computing systems. He is also interested in 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 a recipient of the DoE Early CAREER Award in 2016 and ONR Young Investigator Award in 2019. He received Best Paper Award at DAC'15, ASP-DAC'16, HOST'17, ACM TODAES'18, GLSVLSI'18, DATE'19, and AsianHOST'20. He is also the IEEE Council on Electronic Design Automation (CEDA) Distinguished Lecturer.