DScanPUF: A Delay-Based Physical Unclonable Function Built Into Scan Chain

Yu Zheng, Student Member, IEEE, Fengchao Zhang, Student Member, IEEE, and Swarup Bhunia, Senior Member, IEEE

Abstract—Physical unclonable function (PUF) has emerged as an attractive primitive to address diverse hardware security issues in integrated circuits, such as authentication and cryptographic key generation. Most of the existing PUFs rely on dedicated circuit structure for generating random signatures. It often causes concerns due to extra design efforts and hardware overhead. Moreover, the hardware complexity increases with higher entropy requirement, which may be unacceptable in area-constrained applications. In this paper, we propose DScanPUF, a novel PUF structure that leverages on the scan chain, a prevalent design-for-test structure in a chip. It is based on a low-overhead delay measurement structure consisting of a phase-locked loop and multiple clock delay lines to measure scan path delays at high resolution. A method is proposed to transform the responses into robust binary signatures. We note that the area of DScanPUF is only 18% of the ring-oscillator (RO) PUF with 1024 ROs. Moreover, it can be easily integrated into a design without any influence on testability. DScanPUF is evaluated with test results from 31 field-programmable gate array chips, which show good randomness, uniqueness and reproducibility under temperature, and supply voltage fluctuations. We also show that the signature is robust under aging effects on scan paths through a simulation at 45-nm CMOS process. Finally, we propose a simple structural modification to further improve the signature robustness.

Index Terms—Chip authentication, design for test, hardware security, physical unclonable function, PUF, scan chain.

I. INTRODUCTION

IN RECENT years, security has emerged as an important design parameter for integrated circuits (ICs). IC designers are increasingly required to address diverse security issues, including counterfeiting, reverse engineering, side-channel attacks, and tampering attacks [1]–[4]. Physical unclonable function (PUF) has been widely investigated as a potent security primitive for ICs. PUFs are attractive in variety of applications, such as intellectual property (IP) counter plagiarism, chip authentication, and embedded system security. PUFs transform the intrinsic random variations in device parameters (e.g., threshold voltage $V_{th}$ and channel length $L_{eff}$) arising from fundamental physical limitations in manufacturing process to variations in circuit-level parameters (e.g., current and delay) for random digital signature generation. PUF structures typically lead to the construction of unique and random challenge-response pairs (CRPs) for a chip. A random key can be derived from a digital response from the PUF triggered by a digital input as challenge. The unclonability of PUF originates from the unpredictable, random, and hard-to-copy variations in device parameters during manufacturing.

In diverse security applications, PUFs provide obvious advantages over traditional digital key storage in a nonvolatile memory (NVM). Basically, PUFs avoid the high cost of building tamper-resistant NVM system. Any invasive attack on may alter internal behavior of an IC leading to incorrect signatures as inherent proof of tampering [1], [5]. Moreover, a PUF can produce a large amount of CRPs that are random and usually difficult to predict, which comes at a lower cost compared with alternative techniques and overcomes the limitation of insufficient number (usually only one) of digital-key storage.

A majority of existing PUFs requires dedicated circuit structures [8], [9]. Apart from the substantial cost in silicon area, their integration into a design needs extra effort on the placement, routing, and verification. On the other hand, a separate class of relatively few PUF implementations generates signature from existing on-chip structures. A typical example is a PUF that exploits random mismatch in inner node voltages of a storage element - e.g., an SRAM cell or a flip-flop (FF) [16], [17], [23]. However, it often requires considerable modifications of the original design. For instance, the PUF in [16] adds four extra transistors into each 6-T SRAM cell as twisted NOR gates to initialize the inner node voltages. Although the PUF in [17] requires no such modification, the residual charge in an SRAM cell can severely reduce the power-up randomness and robustness of signature that compromises the quality. The intrinsic PUF uses the power-up state of FFs in field-programmable gate array (FPGA), but it alters the bit configuration procedure to retain the values and successfully read it out [23]. Moreover, a common disadvantage of these PUFs is small challenge-response space and hence, poor Shannon entropy. An SRAM array generates a signature with the entropy of 1-bit/cell in the best case.

In this paper, we present a novel delay-based PUF, which we refer to as DScanPUF, built into the scan chain, a prevalent design-for-test (DFT) structure in a chip. It exploits the variations in scan path delays due to manufacturing process variations to generate a large number of random signatures.
in a chip. Compared to existing delay-based PUFs that use separate circuit structures, DScanPUF incurs significantly lower hardware overhead and achieves larger number of CRPs by extracting high entropy from the scan paths. We also propose a practical scheme to measure the scan path delays with high resolution. It is based on multiple clock delay lines and a phase-locked loop (PLL) enabling dynamic phase shift. The delays of any two scan paths with similar nominal delay are compared to extract binary signature. We evaluate DScanPUF using both circuit-level simulations as well as experimental measurements with a set of 31 FPGA chips (Altera Cyclone III fabricated at TSMC 65-nm process). We show that the generated signatures have good randomness as they pass the statistical tests from National Institute for Science and Technology (NIST) [48]. In terms of uniqueness, the inter-die Hamming distance (HD), representing the difference between any two signatures, is observed to be as high as 49.9%. With respect to the reproducibility of signatures, we observe that the percentage of flipped bits under room temperature is as low as 1.8%. We also experimentally analyze the stability of the signature under supply voltage ($V_{DD}$) and temperature fluctuations, which shows promising results. In particular, this paper makes the following major contributions.

1) It presents the design, covering the architecture, enrollment and reconstruction, of a novel delay-based PUF realized into a scan chain without affecting testability or normal operation. Since the proposed PUF design utilizes an on-chip structure, the overhead remains very small while giving rise to a large challenge-response space. The PUF can be easily integrated into any design equipped with scan chain DFT structures.

2) It presents an efficient low-overhead and robust scheme to measure scan path delays. The scheme is built on the parallel scan-path delay measurement (PSDM) architecture proposed in [32]. It is based on multiple delay lines to enhance the delay resolution and mitigate the effect of temporal noises (e.g., $V_{DD}$ fluctuation) in a chip.

3) It presents the process for extraction of unique digital signatures from measured scan path delays, as well as the steps of enrollment and reconstruction for DScanPUF. Moreover, it experimentally analyzes the influence of the number of clock delay lines on the quality of signature.

4) It analyzes the robustness of the signature under aging effect through simulations in HSPICE by MOS Reliability Analysis (MOSRA) under different duty cycles on scan paths. Finally, it proposes a simple low-cost design method to improve signature robustness under aging effect.

The rest of this paper is organized as follows. Section II discusses prior work on PUFs and provides motivation behind the proposed DScanPUF structure. Section III describes the basic method to measure scan path delay in the presence of nonideality in PLL. The design of DScanPUF is described in detail in Section IV. The experimental results are presented in Section V. The hardware overhead with error-correcting code (ECC) based robustness enhancement is shown in Section VI. Section VII analyzes typical attacks to DScanPUF. Finally, the conclusion is drawn in Section VIII.

II. BACKGROUND AND MOTIVATION

A. Related Work

In recent years, PUF has emerged as an important research topic in the field of hardware security. Researchers have addressed the problems of design, evaluation, and application of different types of PUFs. Optical PUF depends on laser speckle fluctuation of coherent radiation to disordered media [6]. In the coating PUF, a sensor array on top of metal layers measures the unit capacitance of a coating of random dielectric particles for signature generation [7]. Both of them, however, require special material or equipment to generate and/or extract signature. Delay-based PUFs constitute a dominant class of silicon PUFs. They exploit the random variations in signal propagation delay due to variations in device parameters (e.g., $L_{eff}$ and $V_{th}$). A common delay PUF, called the ring-oscillator PUF (RO-PUF) [9]–[11] produces a set of signatures by comparing the frequency of two ROs with identical structure. On the other hand, an arbiter-PUF [8] employs an arbiter to compare the analog delay of two paths in cascaded switches to generate digital bits.

The glitches in combinational circuits can be characterized by a delay line to produce unique and random bits [14]. A delay-measurement structure can be embedded into a design (e.g., crypto-core) to measure path delays and produce signatures [15]. Several delay-based PUF implementations specifically target authentication of FPGA chips. For example, lookup tables (LUTs) can be configured into shift mode to compare the transition delay and thus generate random bits of a signature [12]. Moreover, the delay of configurable LUTs can be measured with high accuracy using a launch-capture circuit and converted into random signature [13].

Another major category of silicon PUFs is constructed in memory or storage cells. The random power-up state of SRAM cell or FF can be employed as the fingerprint of a chip [16]–[20], [23], [24]. In addition, the write failure in SRAM cells caused by reduced write duration or scaled $V_{DD}$ has been exploited to create random digital responses [21], [22]. NVM structures, including emerging ones, e.g., FLASH and phase change memory, are used to construct PUF [25], [26]. The resistance variation in metal interconnects on die has been utilized to design PUF due to its low sensitivity to environmental (e.g., voltage) variations. In [27], the stimulus/measure circuits are placed into the power grid of a chip to measure $V_{DD}$ drop across a metal wire and convert it into a signature.

Two PUFs based on scan chain have been proposed earlier [28], [29]. In [28], the power-up states of scan FFs (SFFs) are used as signature, which is similar to FF-based intrinsic PUFs [23], [24]. They are fundamentally different in operational principle from the delay-based DScanPUF proposed here. In [29], we introduced ScanPUF that performs a scan shift operation under a short clock cycle, which matches the nominal delay of majority of the scan paths. The failure or success of capturing the transition generates random bits in
the SFFs. However, to achieve good uniqueness and robustness of signature, the clock delay line in this approach needs to track well the inter-die corner of a chip as well as temperature- and voltage-induced change in the nominal path delay. It makes the design of clock delay line challenging. The problem arises from the fact that each path independently contributes to one signature bit instead of differential analysis between a pair, as adopted in most delay-based PUFs. It also affects the randomness of signature and results in low entropy (1-bit/FF). In this paper, building on the original approach [29], we propose a novel scan-based PUF implementation that greatly improves the entropy and randomness. It also eliminates the requirement of precise timing control on clock delay lines, which may be difficult to realize by a designer.

B. Motivation of Building PUF Into Scan Chain

As an industrial DFT standard, scan chain is commonly inserted into most ICs to facilitate post-manufacturing test. Since no additional gates but for some occasional buffers are inserted in a scan chain, most of the scan paths have small delays. As a result, scan paths are greatly affected by random intra-die variations, which is helpful for unique and random signature generation. Moreover, since scan paths are often globally distributed across a die and the number of scan paths can be quite large for a realistic design, they can generate high volume of random signatures.

Furthermore, delay measurement of scan paths is much simpler compared with the conventional at-speed scan testing of combinational paths [30]. This is due to: 1) it avoids the difficulty of appropriate test vector generation by an automatic test pattern generation tool to sensitize combinational paths; and 2) it eliminates the need of fast switching on the scan enable signal or an enhanced-scan architecture in order to achieve high combinational path delay testability. Hence, it is attractive to embed simple structures to enable delay measurement of all scan paths in a die.

Since scan paths are typically shorter than functional ones, there may be concern on precise delay characterization of these paths. Note that even though SFFs may have physical proximity, scan path delays cannot be very small to avoid hold timing violations, a common problem for scan chains [31]. Usually, designers target keeping good hold margin (typically by inserting buffers) for scan paths to avoid zero-yield scenario due to inability to perform structural test on scan failure.

III. DELAY MEASUREMENT ON SCAN PATHS

In this section, we introduce the basic method to measure scan path delay in parallel. The factors (e.g., temporal noises) that influence measurement results are analyzed. Next, to address the possible insufficient resolution on delay measurement for a given PLL, we propose a low-overhead approach to improve delay-measurement resolution at the fixed phase-shift step in PSDM by increasing the number of clock delay lines.

A. Preliminary

A SFF is typically realized by adding a 2:1 multiplexer to the input of D flip-flop (DFF). In Fig. 1(a), the output of SFF

![Image](313x550 to 561x731)

Fig. 1. (a) Structure of typical scan paths. (b) Generation of a clock pulse with period $t_{meas}$ for scan path delay measurement.

$j - 1$ is connected with the SD port of SFF $j$, $j = 1, 2, \ldots$ Hence, the scan chain works in scan shift mode by setting signal TD to 1. Assuming 0 and 1 are alternatively input into the scan chain, the outputs of SFF $j - 1$, $j$, and $j + 1$ can be 1, 0, and 1, respectively. After rising edge $t_1$, $0 \rightarrow 1$ transition will occur on scan path $j$. To measure the propagation delay of $0 \rightarrow 1$ transition on scan path $j$, the rising edge $t_1 + 1$ is inserted into the clock after the interval of $t_{meas}$, as shown in Fig. 1(b). If ignoring clock skew, SFF $j + 1$ of chip $i$ outputs

$$O_{i,j+1} = \begin{cases} 
1, & \text{if } t_{meas} - d_{i,j+1,\text{setup}} > d_{i,j,\text{clk2q}} + d_{i,j,\text{com}} \\
0, & \text{if } t_{meas} + d_{i,j+1,\text{hold}} < d_{i,j,\text{clk2q}} + d_{i,j,\text{com}} \\
\text{uncertain otherwise} 
\end{cases}$$

where $d_{i,j+1,\text{setup}}$ and $d_{i,j+1,\text{hold}}$ are the setup and hold time of SFF $j + 1$, respectively; $d_{i,j,\text{clk2q}}$ is the clock-to-q delay of SFF $j$; and $d_{i,j,\text{com}}$ is the combinational delay of path $j$. Assume $t_{meas}$ starts from an initial value $t_{\text{init}}$ and the resolution of $t_{meas}$ is $\Delta t$. $k$ is called the switch point of scan path $j$, if $t_{meas} = t_{\text{init}} + (k - 1)\Delta t$ and $t_{meas} = t_{\text{init}} + k\Delta t$ lead to $O_{i,j+1} = 0$ and $O_{i,j+1} = 1$, respectively. The delay of path $j$ is estimated as

$$d_{i,j} = t_{\text{init}} + (k + k - 1)\Delta t/2 = t_{\text{init}} + (k - 0.5)\Delta t. \quad (2)$$

Note for $1 \rightarrow 0$ transition, we can similarly redefine (1) and the switch point. The delay measurement on all the scan paths can be completed in multiple iterations. In each iteration, since $1 \rightarrow 0$ and $0 \rightarrow 1$ transitions occur on the scan paths simultaneously after the rising edge $t_1$ in Fig. 1(b), all the scan paths can be evaluated under the same $t_{meas}$, which reduces the test time significantly. $t_{meas}$ is increased by $\Delta t$ for the next iteration, until all the switch points are found. The measurement can be repeated to average out the effect of setup/hold timing violation and environmental noises. Zheng et al. [32] proposed a PSDM structure that incorporates the above procedures.

To implement a PSDM, two clocks with tunable phase difference are employed to insert a delay-measurement cycle of period $t_{meas}$ in each iteration, as shown in Fig. 2.
This method has also been used earlier to generate glitchy clock for fault injection attack [33]. The advantage is that short paths can be measured accurately without the requirement of high-frequency clock. Fig. 2(a) shows the generation of delay-measurement cycle by the switch between c1 and c0 employing the structure, as shown in Fig. 2(b). In the beginning, the output clock c3 is connected with c0. The rising edge A3 in c3 is generated by rising edge A0 in c0. Triggered by A0, signal sel becomes 1 after a small delay due to the red line. Hence, c3 is switched to c1 and becomes 0. With the arrival of rising edge A4, the delay-measurement cycle (A3 and A4) is formed. Freq can be adjusted with the phase difference between c0 and c1. After A0, the input of DFF in Fig. 2(b) becomes 0 for a new iteration. In PSDM, Δt is decided by the resolution of phase shift in c1.

**B. Nonideality**

The nonideality in PLL affects the accuracy and robustness of delay measurement. In [36], clock jitter is defined as variation of active (rising or falling) edge from ideal position in time. It can be evaluated by time interval error (TIE), which is the interval between the actual position of active edge and its ideal position. Fig. 3 shows the delay measurement of two scan paths with clock jitter in iteration \( k-1 \) and iteration \( k \). Here, assume path 2 is shorter than path 1. If the jitter achieves TIE2 or more (in the right direction), the switch point of scan path 1 is \( k \), otherwise \( k + 1 \). Hence, clock jitter may lead to unstable switch point. On the other hand, since the switch point of path 2 keeps \( k \), we can identify that path 1 is longer, if measuring the delay of path 1 and path 2 several times in the presence of clock jitter. Hence, clock jitter leads to a smaller \( \Delta t \) under multiple measurements.

With insufficient resolution, the delay measurement cannot track temporal noises. In the experiments using Cyclone III FPGA with \( \Delta t = 97 \text{ ps} [37] \), we observe that the switch points of some paths remain unchanged with increase in temperature. For example, under the room temperature (i.e., 25 °C), the switch points of both path 1 and path 2 are \( k \). When the temperature rises to 40 °C, the switch point becomes \( k + 1 \) for path 1, but still \( k \) for path 2, regardless of the fact that the delay of path 2 also increases with temperature. The signature robustness in Section IV-C is reduced in such scenario. Hence, it is necessary to achieve a \( \Delta t \) smaller than phase-shift step in PLL.

**C. Improvement in Resolution**

We show that different \( t_{\text{init}} \) values can improve the actual resolution of \( t_{\text{meas}} \). Assuming the \( m \)th initial value \( t_{\text{init}} \) is \( t_{\text{init}, m} \), \( t_{\text{meas}} \) can achieve \( t_{\text{init}, m} + \Delta t \), \( t_{\text{init}, m} + 2\Delta t \), \ldots .

Given \( M \) initial values, if modulo(\( t_{\text{init}, m}, \Delta t \)) = \( m(\Delta t/M) \), \( m = 0, 1, \ldots, M - 1 \), the actual resolution can be enhanced as \( \Delta t/M \), when \( t_{\text{meas}} \) is larger than \( \max(t_{\text{init}, 0}, \ldots, t_{\text{init}, M-1}) \).

In the case of \( M = 2 \), as shown in Fig. 4, if \( t_{\text{init}, 0} = 2\Delta t \) and \( t_{\text{init}, 1} = 4.5\Delta t \), then the resolution becomes \( \Delta t/2 \), when \( t_{\text{meas}} \) is larger than \( 4.5\Delta t \).

As shown in Fig. 5, we place multiple clock delay lines for c0 or c1 to change \( t_{\text{init}} \) in (2). In Fig. 5(a), c1 becomes c3 after passing through a M:1 multiplexor denoted as MUX1 and a 2:1 multiplexor (denoted as MUX2). In Fig. 5(b), c1 only passes through MUX2. As a result, the placement of clock delay lines for c0 is better due to a shorter latency on clock switching that leads to a smaller \( t_{\text{init}} \). A small \( t_{\text{init}} \) in \( t_{\text{meas}} \) is good for measuring short path, which is common in case of scan paths. In an application specific IC (ASIC) design, the \( t_{\text{init}} \) for each delay lines can be precisely changed by modifying the layout (e.g., rerouting or gate sizing). In an FPGA, the placement of inverters in delay lines with different wiring can be used to accomplish it. Here, we place two inverters into each clock delay line. Note that if the resolution of phase shift in PLL is good enough, the number of clock delay lines can be reduced in DScanPUF, which can reduce the hardware complexity.
In this section, we describe the procedure of delay measurement with the help of multiple clock delay lines as well as an efficient architecture for DScanPUF. The challenge-response space in DScanPUF is analyzed to extract stable binary signatures.

A. Delay Measurement on Scan Path

The repeated measurement based on multiple clock delay lines can improve the resolution and robustness. Assume DScanPUF uses \( M \) clock delay lines. \( N_p \) scan paths are measured by \( L \) times under each delay line. The switch point of path \( j \) in chip \( c \) is \( k_{i,j,m,l} \) for the \( l \)th measurement under the delay line \( m \). Considering each delay line can correspond to a different \( t_{\text{init}} \), \( k_{i,j,m,l} \) reflects the delay of path \( j \). The accumulation of switch points for path \( j \) is defined as

\[
 s_{i,j} = \sum_{m=1}^{M} \sum_{l=1}^{L} k_{i,j,m,l} \tag{3}
\]

where \( j = 1, 2, \ldots, N_p \). It can be noted that \( s_{i,j} \) increases with \( d_{i,j} \) for all the paths. Hence, \( s_{i,j} \) represents the delay information of each path, which does not require the actual \( d_{i,j} \) or the \( t_{\text{init}} \) of a delay line. In (3), each path is measured by \( ML \) times in total. Using the PSDM, \( k_{i,j,m,l} \) (\( j = 1, 2, \ldots, N_p \)) for \( N_p \) paths can be identified in parallel. Such procedure is repeated by \( ML \) times to compute \( s_{i,j} \).

B. Architecture

DScanPUF embeds a PUF into scan chain, which is originally employed to enhance testability. Hence, besides the normal mode and scan mode, we define a new mode called PUF mode. Correspondingly, the output of scan chain is controlled by two 2:1 multiplexers, as shown in Fig. 6. When \( \text{scan}_{\text{en}} = 0 \) and \( \text{puf}_{\text{en}} = 0 \), the scan chain is in normal mode, and the output port holds zero. During manufacturing test, the scan chain is configured into scan mode by setting \( \text{scan}_{\text{en}} = 1 \) and \( \text{puf}_{\text{en}} = 0 \). As a result, test vectors can be scanned into a chip with the scan-out results for fault detection and analysis. PUF function is achieved when \( \text{puf}_{\text{en}} = 1 \) and the output port also holds zero as that in normal mode to prevent possible information leakage through scan chain. The signatures from DScanPUF can be applied in the device authentication or secret key generation. If DScanPUF is only used in authentication, scan chain is switched between PUF mode and normal mode through external controls (via input/output ports). The temporal noises, such as temperature and \( V_{\text{DD}} \) variation, can be controlled during authentication with few flipped bits. If it is employed in key generation, a finite-state machine can be inserted into the chip that forces the scan chain into PUF mode when the chip is powered up. After the key bits are generated, it makes the scan chain switch into normal mode. Hence, no external operation is required.

The architecture of DScanPUF is shown in Fig. 6, which comprises a main controller, clock generator, clock delay line, clock switch, and post-processor. The clock generator includes a PLL with tunable clock phase. It generates clock \( c_0 \), \( c_1 \), and \( c_2 \). The phases of \( c_0 \) and \( c_2 \) are fixed with 180° difference, while the phase of \( c_1 \) is tunable. \( c_0 \) and \( c_1 \) pass through clock delay lines for multiple \( t_{\text{init}} \) in (2) to improve the resolution of \( t_{\text{meas}} \). The particular structure of clock switch is shown in Fig. 2(b) that inserts a delay measurement cycle into \( c_3 \). The main controller, post-processor, and memory are synchronized with the rising edge of \( c_2 \). The main controller communicates with the clock generator to shift the phase of \( c_1 \). It also inputs test vectors into the scan chain to generate 0 \( \rightarrow \) 1 transition or 1 \( \rightarrow \) 0 transition on the scan paths in each iteration. Finally, it selects a clock delay line in each iteration with the help of signal \( \text{dline}_{\text{num}} \). Post-processor identifies switch point and completes the operation, as shown in (3), for each path. Memory stores the \( s_{i,j} \) in (3). DScanPUF requires no modification on the core logic of a design under test (DUT), which facilitates its integration in a chip. A reconfigurable soft IP can be designed to add PUF functionality into any DUT with scan chain. All scan paths are employed in the PUF implementation to construct a large challenge-response space.

The measured switch points for scan paths, i.e., the \( s_{i,j} \) values, are directly stored in an on-chip memory for signature extraction. The storage space would increase linearly with the number of scan paths. The memory requirement in DScanPUF can be reduced by the following two ways. First, we only store \( s_{i,j} \) associated with a challenge. The architecture in Fig. 6 is modified by inserting an equality comparator to compare a challenge with scan path locations. The memory requirement can be reduced, since only a part of scan paths, selected by a challenge, is employed for signature generation at a time. For example, if 20 scan paths out of 100 scan paths are selected for signature generation, the memory overhead is reduced to 20% of the maximum required storage (for 100 paths). We keep changing the challenge vector to obtain entropy of all scan paths. As a result, each storage unit stores the results for multiple paths in a time-multiplexed fashion. Second, when DScanPUF is integrated into a system-on-chip (SoC) chip, which typically includes a memory (e.g., SRAM and DRAM) array, it can employ existing memory array to temporarily store the path delays, which leads to virtually zero storage overhead. After the signature is generated, if the path delays are still present in memory, an attacker can potentially read them and hence can access the signature bits. To prevent such attack, we propose to flush the contents in memory (e.g., by storing all-zero) after binary signature extraction is completed.
The location of scan path is considered as a part of challenge vector in DScanPUF. Usually, a complex IC, e.g., a modern processor, has several thousand SFFs, which leads to a large number of CRPs. Furthermore, in our experiments, we observe that the propagation delays for $0 \rightarrow 1$ and $1 \rightarrow 0$ transitions on scan path are often different. Hence, a challenge can incorporate the type of transition on scan path. $s_{i,j}$ is regarded as response that includes the entropy of DScanPUF.

Due to temporal noises, $s_{i,j}$ may change for the same chip in different measurements. Hence, it cannot be directly employed in authentication or secret key generation without any post-processing. To suppress the common-mode noises, we select path $j$ and $j'$ ($j \neq j'$) with similar nominal delay to produce a binary bit as

$$\text{sig}_j = \begin{cases} 1, & \text{if } s_{i,j} > s_{i,j'} \\ 0, & \text{else.} \end{cases}$$

The enrollment and reconstruction of signatures in DScanPUF are shown in Fig. 7. After manufacturing, $s_{i,j}$ is computed for each path of all chips. We bin the paths by paths within the same group. Considering the order of $s_{i,j}$ may change under temporal noises, the generated bits flip, thus reducing robustness. Hence, the last step in enrollment is the secure sketch, as proposed in [50], for error correction during reproducing signatures. The sketch method can be built on ECCs. If it is a linear block code (e.g., Bose-Chaudhuri-Hocquenghem codes and repetition codes), the syndrome $\text{syn}_1$ is public information with limited entropy leakage, which is also referred to as helper data [45], [46]. For example, assuming the parity check matrix is $H$, the syndrome of the generated signature $\text{sig}_1$ is computed as $\text{syn}_1 = \text{sig}_1 \cdot H^T$. The challenges, signatures, and helper data are stored for reconstruction in future. Usually, the signatures are stored in a secure database, while challenges and helper data can be publicized. During reconstruction, the challenges are used to generate the raw bits. With the helper data, the error bits are corrected by decoding the ECC codeword to recover the original signature. For example, if $\text{sig}_1'$ is generated ($\text{sig}_1' \neq \text{sig}_1$), we decode the codeword with the syndrome $\text{syn}_1 + \text{sig}_1' \cdot H^T = (\text{sig}_1 + \text{sig}_1') \cdot H^T$ to find the flipped bits between $\text{sig}_1$ and $\text{sig}_1'$.

For $N_p$ scan paths in group $p$, the entropy is $\log_2(N_p!)$ bits in the best case, which is the same as that in RO-PUF. An important difference between them is that the frequency variation in RO-PUF is obtained by counting the pulses of ROs, while scan path delay variation in DScanPUF is captured by PLL and delay line based measurement. In Section VI, we show that repetition code can be employed to bring the robustness of DScanPUF to the level of RO-PUF, at the cost of very modest overhead.

V. EXPERIMENTAL RESULTS

In this section, we present the experimental results for 31 DE0 FPGA boards equipped with Altera Cyclone III FPGA.

A. Experimental Setup

In validating DScanPUF, circuit s9234 with 211 SFFs (210 scan paths) from the benchmark suite ISCAS-89 is selected as the DUT (as shown in Fig. 6). Two inverters (a buffer) are inserted into each scan path. The period of $c_0$, $c_1$, and $c_2$ is $1/17.637$ MHz = 56.7 ns, and the resolution of dynamic phase shift in the PLL is 0.097 ns (1/8th of the voltage-controlled oscillator period 0.777 ns). The number of clock delay lines is $M = 32$ and all scan paths are measured $L = 8$ times for each line.

The setup time of DScanPUF for all $s_{i,j}$ (210 scan paths) is $\sim152$ ms. In case of RO-PUF, [45] shows that the time for generating 128-bit RO-PUF output is 4.59 ms, which is faster than DScanPUF. However, the setup time of DScanPUF can be significantly reduced without performance degradation, if we increase the clock frequency (e.g., a different PLL) or the phase adjustment times (e.g., if the range of scan path delays is known). For example, if the clock frequency improves to 68 MHz using a different PLL with 10 times phase adjustment in each delay measurement, the setup time can be reduced to 7.6 ms.

The hardware resources required for implementing the DScanPUF architecture, as shown in Fig. 6, include 284 logic elements or logic element (LEs) (2% of total 15 408 LEs), and 4608 memory bits (<1% of total memory bits) in Cyclone III FPGA device (3C16F484C6), which is much smaller than
The number of paths in each group is between 9 and 29. We divide 152 paths out of the total 210 paths into nine groups. 2146 slices in 2Probased FPGA [15].

the delay-based PUF HELP (1297 FFs, 3122 LUTs, and 2146 slices in 2Probased FPGA) [15].

Following the bootstrapping procedure, as shown in Fig. 7, we divide 152 paths out of the total 210 paths into nine groups. The number of paths in each group is between 9 and 29. \( N_b = 140 \) bits are extracted for each chip according to (4). In [45], the entropy density \( \rho \) is defined as \( \rho = N_b / \sum_{k=1}^{N_b} \log_2(k) \). Each group has different \( \rho \). On average, \( \rho \) is 38.2% in the DScanPUF constructed on s9234, which is implemented in Cyclone III FPGA.

B. Randomness

A signature generated from DScanPUF should be random with high Shannon entropy to make it difficult for an adversary to predict [46], [48], [49]. We have evaluated the randomness of signatures by the NIST statistical test suite [48] under different temperatures (i.e., 10 °C, 25 °C, 40 °C, 55 °C, and 70 °C) and three \( V_{DD} \) corners (i.e., 1.1, 1.2, and 1.3 V). As shown in Table I, the 140-bit signatures obtained from 31 Altera DE0 boards can pass the NIST tests ApproximateEntropy, CumulativeSums, Frequency, LongestRun, Runs, BlockFrequency, Serial, and the test FFT with over 98% success rate on average. Since the tests Rank, LinearComplexity, OverlappingTemplate, NonOverlappingTemplate require >140 bits, we combined the signatures from multiple boards and the pass proportion is >98% for the default parameters. Finally, the DScanPUF built on s9234 cannot complete the evaluation under RandomExcursion, RandomExcursionVariant, and Univeral tests, since they require at least one million (i.e., \( 10^6 \)) bits. We used five of the FPGA boards to evaluate the randomness at two additional voltage points: 1) \( V_{DD} = 1.1 \) V and 2) \( V_{DD} = 1.3 \) V. All five boards passed the tests listed in Table I except for Board 1 failing the test FFT. As DScanPUF depends on the process variation effect in scan path delays, we can expect significant enhancement in signature randomness with more scan paths (thus higher Shannon entropy) in large-scale realistic designs, which are likely to be much larger than s9234 benchmark.

In our experiment, the number of FPGA boards (samples) was inadequate for the \( \chi^2 \) test (i.e., \( P \)-value of the \( P \)-values, \( P \)-value\(_T\)). Hence, to estimate the \( P \)-value distribution, we remapped the DScanPUF for the s9234 design into different regions of the FPGA chips under the normal corner (i.e., 1.2 V, 25 °C). It helped us to generate total 55 samples with 140-bit length per sample. The \( P \)-value\(_T\) for the test ApproximateEntropy, CumulativeSums, Frequency, LongestRun, Runs, BlockFrequency, and Serial are observed to be 0.4457, 0.1160, 0.0502, 0.0031, 0.0014, 0.1223, and 0.0083, respectively. All of these values are larger than the threshold 0.0001. Hence, they can be considered as uniform distribution. Since the recommended minimum bit length for FFT test is 1000, we increased the signature length by combining multiple 140-bit signatures and obtained the \( P \)-value\(_T\) = 0.0089. The remaining tests require even larger number of bits for an accurate \( P \)-value\(_T\) estimation, which are, therefore, not considered in this paper.

C. Uniqueness

The uniqueness of signatures is evaluated by HD, which is basically the number of different bits between any two signatures. Fig. 9(a) shows the histogram of 31 signatures. We can observe that most inter-die HDs concentrate in the range of 0.5, which means the signature is not biased toward 1 or 0.

Assuming \( HD_{i,j} \) is the HD between chip \( i \) and chip \( j \), the average HD for \( m \) chips, denoted by \( HD_{avg} \), is calculated as [34]

\[
HD_{avg} = \frac{2}{m \cdot (m - 1)} \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} HD_{i,j}. \tag{5}
\]

The \( HD_{avg} \) is 49.9%, and therefore, DScanPUF can provide a signature with excellent uniqueness to identify each chip.

D. Robustness

In this section, the robustness of DScanPUF is evaluated by flipping rate (i.e., bit error rate) of signature under temporal noises (e.g., temperature and \( V_{DD} \) fluctuation) as well as aging effects.

1) Robustness Under Temporal Noises: First, we show that the resolution of delay measurement can be improved with multiple clock delay lines, so that \( s_{i,j} \) can track the change of temperature and \( V_{DD} \). \( s_{i,j} \) is normalized by dividing it with its value obtained in 25 °C with \( V_{DD} = 1.2 \) V. Path delay increases with temperature, which leads to a larger \( s_{i,j} \). An increase in \( V_{DD} \) results in a smaller \( s_{i,j} \). Fig. 10 shows that the normalized \( s_{i,j} \) tracks the temperature fluctuations between 10 °C and 70 °C and \( V_{DD} \) variations between 1.1 and 1.3 V.

<table>
<thead>
<tr>
<th>Table I: Pass proportion of NIST tests for DScanPUF under ( V_{DD} = 1.2 ) V when ( \alpha = 0.01 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
</tr>
<tr>
<td>App.Intropy</td>
</tr>
<tr>
<td>CumSums</td>
</tr>
<tr>
<td>Frequency</td>
</tr>
<tr>
<td>LongestRun</td>
</tr>
<tr>
<td>Runs</td>
</tr>
<tr>
<td>BlockFreq</td>
</tr>
<tr>
<td>Serial</td>
</tr>
<tr>
<td>FFT</td>
</tr>
<tr>
<td>Rank</td>
</tr>
<tr>
<td>LinearComp.</td>
</tr>
<tr>
<td>Overlap</td>
</tr>
<tr>
<td>NonOverlap</td>
</tr>
</tbody>
</table>
Since $s_{i,j}$ and $s_{i,j'}$ ($j \neq j'$) change in the same direction, the signature based on (4) is robust. Furthermore, comparing Fig. 10(a) with Fig. 10(b), we can observe that $s_{i,j}$ is more sensitive to $V_{DD}$ than temperature.

To evaluate the robustness, we use the same set of challenges for all the FPGAs, which is also same as those used in the evaluation of uniqueness. The percentages of flipped bits in signature are shown in Fig. 12. The error bits in the signatures are only 1.8% under 25 °C, which become 11% when the temperature increases to 70 °C. It is similar to the reported results for RO-PUF [45]. Fig. 11 shows the influence of clock delay lines (i.e., 1, 9, 17, 25, and 32) on the percentage of unstable bits at 40 °C, 55 °C, and 70 °C. Signature robustness is generally enhanced with more clock delay lines, since minor common variations on scan path delays can be detected accurately.

We modified five Altera DE0 boards with a new regulator to adjust the core $V_{DD}$ of FPGA chips, as shown in Fig. 8. As $V_{DD}$ is changed from 1.3 and 1.1 V from 1.2 V, 21% and 5.6% bits, respectively, flip on average. This can be explained from two perspectives. First, the characteristics of PLL (e.g., frequency, jitter) change with $V_{DD}$, which leads to the instability of switch points. Second, $V_{DD}$ has more significant influence on delay variation due to $s_{i,j} \propto V_{DD}/(V_{DD} - V_{th})^\alpha$ ($\alpha > 1$). Hence, the original order of $s_{i,j}$ becomes more difficult to keep with the increase of delay variation. We also combine the $V_{DD}$ and temperature variation to form fast corner (1.3 V, 10 °C) and slow corner (1.1 V, 70 °C) of DScanPUF. We note that the bit errors are 23.5% and 8.1% for slow and fast corner, respectively. Hence, the bit error (e.g., $>15\%$) due to $V_{DD}$ variation primarily comes from $V_{DD}$ drop.

In Fig. 10(b), $V_{DD}$ variations are reflected on $s_{i,j}$ deviation from the measured $s_{i,j}^{(nom)}$ at the nominal corner (1.2 V, 25 °C). $V_{DD}$ varies over time when a chip is running [47]. For example, compared with $s_{i,j}^{(nom)}$, $s_{i,j}$ is changed by more than 10% with 0.1 V reduction on $V_{DD}$, which is only $\sim4\%$ when the temperature rises up to 70 °C with the same $V_{DD}$. Hence, we propose an iterative procedure, as shown in Fig. 13, to drop all the $s_{i,j}$ (i.e., signatures), if $\left|\sum_j s_{i,j} - \sum_j s_{i,j}^{(nom)}\right|$ is out of a specific range $\beta$, where path $j$ is within the same group. $\sum_j s_{i,j}^{(nom)}$ and $\beta$ are written into an NVM after post-silicon test. Furthermore, it is worth noting that the knowledge of $\sum_j s_{i,j}^{(nom)}$ and $\beta$ would not help an attacker, due to the absence of information about the order of $s_{i,j}$. Moreover, the induced computation complexity is small with only simple addition and subtraction. Here, $\beta$ is determined based on a reasonable $V_{DD}$ range so that the flipped bits are dominated by temperature fluctuation (i.e., $11\%$ with 70 °C) in real application of DScanPUF. For example, as shown in Fig. 10, if $\beta = 0.05 \times s_{i,j}^{(nom)}$, the drop of $V_{DD}$ to 1.1 V can be detected so that the generated signature is discarded, while DScanPUF can still work within a large temperature fluctuation.

2) Robustness Under Aging Effect: In the nanometer technology regime, a dominant contributor to the aging effect is negative-biased temperature instability (NBTI) in pMOS transistor. The $V_{th}$ shift of pMOS transistors in a gate
due to NBTI is related to a series of factors, such as $V_{DD}$, duty cycle, and circuit structure [35]. As shown in Fig. 14, for a scan path, the output of SFF S1 (0 or 1) puts different inverters under different stress mode. If the first (second) inverter is aged, the propagation delay of falling (rising) edge will be affected severely. Moreover, the output of S1 may not have the same duration of 0 (duty cycle) during life time. Hence, the extent of NBTI effect on inverters in scan paths can be different. To obtain the unstable bits under various duty cycles, we analyze it with Monte Carlo simulation in predictive technology model 45-nm CMOS process [38] using HSPICE platform. The profile of $V_{th}$ shift due to NBTI is generated under different duty cycles for one-year aging in MOSRA [39] tool. For the process variation, we consider 15% inter-die and 10% intra-die variations in $V_{th}$ as used in [29]. The DUT includes 100 scan paths with identical nominal delays and 200 instances are produced in the Monte Carlo simulation. The signature of 128 bits is generated for each instance after obtaining the delay of rising edge.

Fig. 15. Change of delay (rising edge) under NBTI with different duty cycles.

Fig. 14. Inverters under stress associated with the output of SFF S1.

Fig. 15. Change of delay (rising edge) under NBTI with different duty cycles.

Table II

<table>
<thead>
<tr>
<th>Comparison of Number of LEs between RO-PUF and DScanPUF</th>
</tr>
</thead>
<tbody>
<tr>
<td>RO-PUF</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>64 scan paths or 64 ROs</td>
</tr>
<tr>
<td>128 scan paths or 128 ROs</td>
</tr>
<tr>
<td>256 scan paths or 256 ROs</td>
</tr>
<tr>
<td>1024 scan paths or 1024 ROs</td>
</tr>
</tbody>
</table>

VI. HARDWARE COMPLEXITY

In this section, DScanPUF is compared with RO-PUF in terms of area overhead under the same number of ROs and scan paths, since they can generate the same number of raw bits. We assume the nominal delay of each scan path is nearly the same. At the beginning, we list the hardware complexity of implementing DScanPUF and RO-PUF, including the same number of ROs and scan paths, without considering the reliability of generated bits. Table II shows the number of LEs in RO-PUF and DScanPUF for 64, 128, 256, and 1024 scan paths/ROs, synthesized and mapped into Cyclone-III FPGA device. RO-PUF is implemented according to [9] and each RO contains five inverters. The memory overhead in DScanPUF is not listed in Table II, since one can employ the embedded memory in SoC chips. DScanPUF keeps constant number of LEs with the increase of scan paths, because the delay information is collected by a PLL-based structure addressing all scan paths. The percentage continues to reduce with the increase of scale. As a result, we conclude
that DScanPUF is a low-overhead delay-based PUF, especially at a large scale, i.e., when high entropy is required.

Next, we compare the reliability of signatures. The reported number of flipped bits in RO-PUF varies across different studies in open literatures. For example, in [45], the unstable bits of RO-PUF are \( \sim 2\% \) under room temperature, which rise up to 5% and 11% in low and high temperature, respectively. In [9], it becomes 0.48% at the worst temperature and \( V_{DD} \) fluctuation, when a 128-bit signature is extracted from 1024 ROs (five inverters in each RO). Hence, we set 0.48% as the reference for comparison. Assuming large \( V_{DD} \) fluctuation can be detected by the iterative procedure in Fig. 13, the flipping rate of DScanPUF is \( p_e = 11\% \) in high temperature. As the challenge-response space of DScanPUF is large, we adopt the repetition code to improve the reliability of DScanPUF. Since \( (n, k, \min) = (7, 1, 3) \) repetition code can correct up to \( \min - 1 \) errors, the bit error rate becomes \( 1 - \sum_{k=0}^{\min - 1} \binom{n}{k} (p_e)^k (1 - p_e)^{n-k} \) after decoding based on syndrome. In particular, when \( n = 7 \), \( p_e \) becomes 0.39% (<0.48%). The decoding circuit of \( (7, 1, 3) \) repetition code is fully combinational, which takes five LEs in Cyclone III.

Finally, we consider path grouping, entropy density \( \rho = 38.2\% \) and \( p_e = 0.39\% \); 1024 scan paths are assigned into groups and each group includes the same number of scan paths. Fig. 17 shows the length of signature with \( p_e = 0.39\% \) under different number of groups. A 478-bit signature can be generated if all the scan paths have the same nominal delay. It reduces to 182 bits when the 1024 scan paths are assigned into 40 groups (25 paths in each group). According to Table II, the area ratio between DScanPUF with repetition code \( (7, 1, 3) \) and the RO-PUF of 1024 ROs is \( (1308 + 5)/7306 = 17.8\% \).

VII. ATTACKS ON DScanPUF AND COUNTERMEASURES

In this section, we analyze some possible attacks on DScanPUF, as well as countermeasures that can enhance its security.

A. Cloning Attack Through Reverse Engineering

The chip cloning based on reverse engineering is a basic and direct possible attack to DScanPUF, which requires an adversary to manufacture a duplicated PUF with the same challenge-response space as that of a genuine one. To achieve this, he/she should first identify the nominal delay of each scan path in the original design and then modify the layout to match the nominal delays. Next, a duplicated PUF chip is manufactured under process variation (may be different parameters) with the actual path delays same as genuine PUF chips. To know the nominal delay of scan path, the attacker has to access scan chain from outside, and work on a reasonably large number of genuine chips. In real scenario, scan chain is usually protected by key-based authentication or cutoff after manufacturing test with an electrical fuse [43], [44]. With knowledge of nominal delay values, the layout modification can be highly time-consuming and even practically infeasible with increasing of path number. Finally, manufacturing a chip with identical challenge-response space as a target chip has a negligible probability of success due to the random nature of process variations. Hence, it is infeasible for an attack to clone a DScanPUF integrated into an SoC chip with thousands of scan paths.

B. Machine Learning Attack

The attack based on machine learning (ML) is mounted when an attacker knows a partial list of CRPs. A model (e.g., logistic regression, artificial neural networks) can be built to predict the responses of remaining challenges. In open literatures, ML has been shown effective on several common PUFs, such as arbiter-PUF, XOR-PUF, and RO-PUF [40]–[42]. Considering DScanPUF is a delay-based PUF, ML can potentially be employed. The effectiveness will be explored quantitatively through simulations and measurements in future. Here, we consider some key points described in [40] and [41] to analyze the feasibility of ML attack to DScanPUF. First, the increase of entropy can help a PUF in resisting ML attack. In [41], under 95% success rate, the number of CRPs is 640 for an arbiter-PUF of 64 stages, which rises to 1350 when the number of stages becomes 128. Moreover, the size of training set in ML affects prediction accuracy. In [42], the success rate clearly declines with the number of CRPs due to insufficient \textit{a priori} information. Hence, a direct and simple method to resist ML attack is enlarging the scale of a PUF (e.g., more stages/ROs in arbiter-RO-PUF). However, it results in a large silicon area and thus manufacturing cost. Fortunately, DScanPUF employs the scan paths already in a chip, which incurs minimal hardware overhead to utilize a large number of scan paths of a modern SoC in signature generation. Hence, compared with other delay-based PUFs, DScanPUF is expected to have significantly higher defense against ML attacks at iso-overhead.

VIII. CONCLUSION

We have presented an approach to build a PUF, referred to as DScanPUF, into the scan chain by exploiting random variations in scan path delays. It does not affect the testability of the chip as well as its normal operation. We have presented a practical method to extract robust binary signature. We have also presented an efficient low-overhead method to enhance the resolution of path delay measurement. Since it utilizes the scan paths already in a chip, DScanPUF requires low-complexity on-chip structure. It, however, provides a large...
challenge-response space. Measured results from a set of 31 FPGA devices show that the signatures have high level of uniqueness and randomness as determined by the NIST statistical tests on randomness. Through experiments with FPGAs, we have also shown that the effect of delay variation due to temporal noises (e.g., resulting from temperature and voltage fluctuations) can be effectively mitigated. We have proposed an iterative procedure to improve the robustness under VDD variations. Moreover, through extensive HSPICE simulations, we show that the number of flipped bits under aging effect with different duty cycles on scan paths is not significant. To reduce the impact of unbalanced aging on scan paths, a modified low-overhead DScanPUF structure is proposed. Future work would include further improvement on robustness of signature and quantitative analysis on different attacks on the PUF, e.g., ML-based attack.

REFERENCES


Yu Zheng (S’12) received the B.S. degree in communication engineering and the M.S. degree in communication and information systems from the University of Electronic Science and Technology of China, Chengdu, China, in 2007 and 2010, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering and Computer Science, Case Western Reserve University, Cleveland, OH, USA. His current research interests include hardware security (e.g., physical unclonable function and counterfeit chip detection), VLSI architecture design for communications, and signal processing.

Fengchao Zhang (S’15) received the B.E. degree in optoelectronic information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2012. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering and Computer Science, Case Western Reserve University, Cleveland, OH, USA. His current research interests include hardware security and nanoelectromechanical switch.

Swarup Bhunia (M’05–SM’09) received the B.E. (Hons.) degree from Jadavpur University, Kolkata, India, the M.Tech. degree from IIT Kharagpur, India, and the Ph.D. degree from Purdue University, West Lafayette, IN, USA. He is currently T. and A. Schroeder Associate Professor of Electrical Engineering and Computer Science with Case Western Reserve University, Cleveland, OH, USA. He has over ten years of research and development experience with over 150 publications in peer-reviewed journals and premier conferences. His current research interests include hardware security and trust, adaptive nanocomputing, and novel test methodologies.

Dr. Bhunia was a recipient of the IBM Faculty Award in 2013, the National Science Foundation Career Development Award in 2011, the Semiconductor Research Corporation Inventor Recognition Award in 2009, the SRC Technical Excellence Award in 2005, and several best paper awards/nominations. He has served as an Associate Editor of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN, the IEEE TRANSACTIONS ON MULTI-SCALE COMPUTING SYSTEMS, the ACM Journal of Emerging Technologies, and the Journal of Low Power Electronics. He served as a Guest Editor of the IEEE Design and Test of Computers in 2010 and 2013, and the IEEE JOURNAL ON EMERGING AND SELECTED TOPICS IN CIRCUITS AND SYSTEMS in 2014. He served as the Co-Program Chair of the IEEE International Mixed-Signal, Sensors, and Systems Test Workshop in 2011, the IEEE Nanoscale Architectures in 2013, and the IEEE International Symposium on VLSI Design and Test in 2014. He serves as the Vice-Program Chair of the IEEE HOST in 2015. He has served on the program committee of many IEEE/ACM conferences.