Article | Received 20 June 2025; Revised 3 January 2026; Accepted 3 January 2026; Published 22 January 2026
https://doi.org/10.55092/esp20260001

A weak FPGA physically unclonable function using multiple-parallel single delay-line based time-to-digital converters with linearity self-calibration

Kentaroh Katoh[1,*], Toru Nakura[1] and Haruo Kobayashi[2,†]

[1] Department of Electronics Engineering and Computer Science, Fukuoka University, Fukuoka, Japan
[2] Gunma University, Kiryu, Japan

† Professor Emeritus.

* Correspondence author; E-mail: kentarohkatoh@fukuoka-u.ac.jp.

Highlights:

- First proposal of weak physically unclonable function (PUF) utilizing pre-existing multiple parallel time-to-digital converters (MP TDCs).
- Applied linearity self-calibration using histogram method to calculate buffer delays of TDCs.
- Applicable to various MPSDL-TDCs on various FPGAs.

Abstract: This paper presents a Weak Field-Programmable Gate Array Physically Unclonable Function (FPGA PUF) using Multiple Parallel Single Delay-Line based Time-to-Digital Converter (MPSDL-TDC) on FPGA. In the measurement mode, the proposed PUF works as a high-resolution FPGA TDC with the MPSDL-TDCs. In the PUF mode, the proposed PUF selects a stage of a TDC in the MPSDL-TDC and another stage of another TDC in it. Next, we calculate the buffer delays of the selected stages with the linearity self-calibration. Finally, we obtain a 1-bit response output by comparing the buffer delays. With a small amount of circuit, the proposed PUF can be applied to any type of MPSDL-TDCs, even it is the one with Look-Up Table (LUT) chain based TDCs or with the dedicated carry-chains based TDCs. Consequently, the number of the extra resources for the proposed PUF is small. Evaluation using 10 Artix-7 FPGAs resulted in an inter-chip Hamming distance of 47.07%, a reliability of 93.91%, and a uniformity of 50.70%. Minimum resource overhead for the proposed PUF over original MDSDL-TDCs is 4.99%.

Keywords: PUF; weak PUF; hardware security; TDC; FPGA; MPSDL-TDC

1. Introduction

Physically Unclonable Function (PUF) is an important security primitive used for the device authentication and the generation of secret keys for encrypted communication. PUF has an entropy source that can generate reproducible random values. Using such an entropy source, a device-specific signature

can be generated [1]. Various PUFs, such as delay-based PUF and memory-based PUF, have been proposed [2,3]. PUFs are broadly classified into Strong PUF and Weak PUF based on the number of Challenge input C and Response Output R pairs (CRPs). Strong PUFs are mainly used for device authentication, while Weak PUFs are used for generating secret keys for encrypted communication through public key cryptographic authentication methods [4].

A TDC is a circuit that converts time intervals into digital values. Its applications are diverse, including Time of Flight (ToF) systems, light detection and imaging, position emission tomography, ultrasonic measurement, and diffuse optical tomography [5,6]. In the past, TDCs were implemented as full custom designs. However, in the recent years, due to the miniaturization of Field-Programmable Gate Array (FPGA) manufacturing processes, as well as their reconfigurability, shorter time to market, and lower development costs, the implementation of TDC using FPGAs (FPGA TDC) has been widely considered.

FPGAs are devices that can implement any arbitrary digital circuits, once they have been programmed. FPGAs are suitable for multi-channel applications and asynchronous architectures. Multiple Parallel Single Delay-Line (MPSDL)-TDC on FPGA provides better resolution than single SDL-TDC on FPGA [7–9].

This paper presents a Weak FPGA PUF using MPSDL-TDC on FPGA. In the measurement mode, the proposed PUF works as a high-resolution FPGA TDC with the MPSDL-TDC. In the PUF mode, the proposed PUF selects a stage of a TDC in the MPSDL-TDC and another stage of another TDC in it. Next, we calculate the buffer delays of the selected stages with the linearity self-calibration. Finally, we obtain a 1-bit response output by comparing the buffer delays. With a small amount of circuit, the proposed PUF can be applied to any type of MPSDL-TDC, even it is the one with Look-Up Table (LUT) chain based TDCs or with the dedicated carry-chains based TDCs. Consequently, the number of the extra resources for the proposed PUF is small.

The rest of this paper is organized as follows. Section 2 presents related works. Section 3 describes preliminaries. Section 4 explains the proposed PUF. Section 5 shows the experimental results. After some discussions at Section 6, Section 7 concludes this paper.

2. Related works

PUFs generate unique signatures by utilizing manufacturing variations in devices on ICs. Various PUFs have been proposed that leverage the variations in different physical quantities caused by manufacturing variations. Memory based PUFs use the variations in the characteristics of memory cells. SRAM PUF utilizes the variation in the initial values of SRAM cells immediately after power-up [3]. In Butterfly PUF, cross-coupled loops similar to SRAM cells are implemented in an array on FPGA. Like SRAM PUF, Butterfly PUF also uses the variation in the initial values of the cross-coupled loops [10]. DRAM PUF uses the variation in the capacitance of DRAM cells [11]. Many PUFs based on emerging non-volatile memories such as ReRAM and STT MRAM have also been proposed [12,13]. Chowdhury *et al.* proposed a PUF that utilizes the characteristic variations of null conventional logic gates used in asynchronous circuits [1]. Inverter/amplifier analog PUFs use the variations in threshold voltage and switching voltage of inverters and inverting amplifiers [14,15]. Xi *et al.* proposed a PUF that uses the subthreshold current of CMOS devices as an entropy source [16]. ADC PUF utilizes the mismatches among capacitances in Successive Approximation Register Analog-to-Digital Converter (SAR ADC) as

an entropy source [17]. Delay-based PUF utilizes the delay variations of gate circuits. Arbiter PUF and RO PUF are well-known delay-based PUFs [18].

FPGAs are devices for digital circuits [19]. Accordingly, PUFs composed of gate circuits can be implemented on FPGAs. These PUFs implemented on FPGAs are used for device authentication of FPGAs, generation of secret keys for encrypted communication using public key authentication methods, and IP protection as FPGA PUFs [10,18,20]. As far as we know, Butterfly PUF is the first FPGA PUF using a cross-coupled loop array [10]. Recently, Sala *et al.* proposed low-cost FPGA PUFs using cross-coupled loop array, such as DD-PUF, NAND-PUF, and XOR-PUF. They conducted comparative evaluations of these with previous FPGA PUFs using cross-coupled loop array like TERO-PUF and SS-RO-PUF [21].

Most of delay-based PUFs can also be implemented on FPGAs. RO FPGA PUFs have been widely researched over the past 20 years. Suh *et al.* were the first to propose RO PUF [18]. Maiti *et al.* improved the area efficiency by implementing 8 ROs in one Configurable Logic Block (CLB) that is a reconfigurable element of FPGA composed of LUTs and Flip-Flops to construct arbitrary sequential circuit [22]. Xin *et al.* further improved the area efficiency by implementing 256 ROs in one CLB [23]. Pei *et al.* improved area efficiency by utilizing the fact that Xilinx CLBs have two outputs, implementing two reconfigurable ROs in one CLB [24]. All of these FPGA PUFs are Weak PUFs.

On the other hand, in recent years, there have been several attempts to implement Strong PUFs on FPGAs. Gupta *et al.* realized a Strong RO PUF by using modified configurable ring oscillators [25]. Anandakumar *et al.* proposed an efficient implementation of XOR Arbiter PUF on FPGA [26]. We proposed a strong PUF using dual TDCs for AMD FPGA [27]. Ni *et al.* proposed a Strong PUF on FPGA which uses the difference of two identical delay-lines controlled by a common challenge input as a control input of an 8-bit Linear-Feedback Shift Register (LFSR) whose output is the 8-bit response output used as a pseudo-random pattern generator [28]. A LFSR is a shift register whose input bit is a linear function of its previous state. The difference is measured by a high-resolution TDC. Because TDC is composed of digital circuits, it is widely applied for various security applications without being just only applied for PUF [29,30].

In MPSDL-TDC, time intervals are applied to the multiple parallel identical low-resolution SDL-TDCs. By averaging the measurement result of each TDC, the resolution and precision are improved. Usually MPSDL-TDC is implemented on FPGA.

Daigneault *et al.* realized a TDC with 10 ps resolution and 24 ps precision using 10 parallel dedicated carry-chains based TDCs on Virtex-II Pro FPGA [7]. Szplet *et al.* analyzed the influence of the number of parallel TDCs on the resolution and precision. They considered 14 different variants of multiple parallel dedicated carry-chains based TDCs up to the 16 parallel TDCs. They confirmed that the resolution was improved up to 11 times and the precision was improved about 2 times [8]. Shen *et al.* realized a TDC with 1.7 ps equivalent bin size, 1.5 ps averaged bin size, and 4.2 ps RMS using 16 parallel dedicated carry-chains based TDCs on Virtex-6 FPGA [9]. The PUF proposed by Ni *et al.* measures the difference of delay of identical delay lines by the multilayer TDC that is 3 parallel TDCs to generate response outputs [28].

The proposed FPGA PUF is applied to MPSDL-TDC. The PUF is a Weak PUF that uses the buffer delay of the internal SDL-TDCs as an entropy source. On the other hand, the PUF proposed in [27] is a

Strong PUF that uses the difference of the measurement results of a common time interval by dual parallel identical TDCs with identical multiplexer-chain based delay-lines as an entropy source.

3. Preliminaries

This section gives the prior knowledge to understand the proposed PUF. Subsection 3.1 overviews the basics of SDL-TDC. Subsection 3.2 describes the linearity self-calibration using histogram method for SDL-TDC. Subsection 3.3 explains MPSDL-TDC.

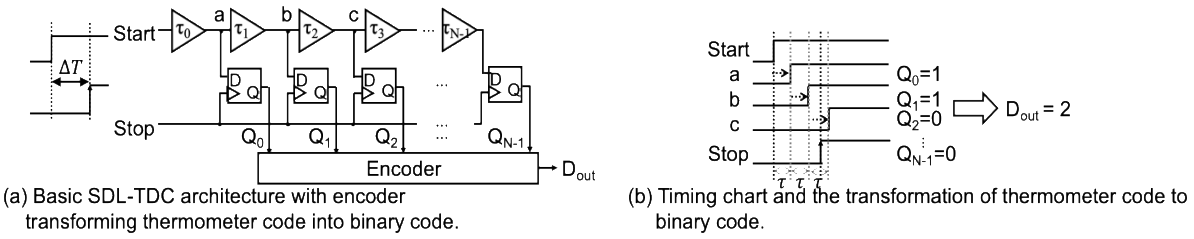
3.1. Single Delay-Line based TDC (SDL-TDC)

SDL-TDC is the most basic TDC [31]. Figure 1a shows a typical N-stage, which is composed of a buffer-chain with N buffers and N Flip-Flops (FF) [31]. Start input is provided to the buffer-chain. Stop input is connected to each clock input of each FF. Each stage of the TDC consists of a buffer and an FF. The buffer delay and the output of the FF of the i-th stage are τ_i and Q_i, respectively. In ideal TDC with resolution τ, $\tau_i = \tau$ in all the stages. Let the buffer outputs of the first, second, and third stages be a, b, c, respectively.

A positive transition TR_U is provided to Start input, and then, a positive transition TR_L is to Stop input. The TDC measures their time interval ΔT. At the rising edge timing of TR_L, all the FFs capture the input data. Figure 1b shows the timing chart in case the rising edge timing of Stop is between those of a and c. There, $Q_0 = Q_1 = 1$, $Q_2 = Q_3 = \cdots = Q_{N-1} = 0$, and thus, ΔT is measured as 2τ. The bit string $Q_0Q_1Q_2Q_3\cdots Q_{N-1}$ in thermometer code format is transformed to the corresponding integer D_{out} in binary code format by Encoder.

The structure of the TDC is simple. Therefore, it is often implemented on FPGA [32–34]. A buffer chain is implemented with LUT-chain or dedicated carry-chains in FPGA.

In this paper, a TDC with a single LUT-chain is called LUT-chain based TDC, while the one with a single dedicated carry-chains is dedicated carry-chains based TDC. Vernier Delay-Line based TDC (VDL-TDC) using two LUT-chains was also proposed for higher resolution [35].



(a) Basic SDL-TDC architecture with encoder transforming thermometer code into binary code.

(b) Timing chart and the transformation of thermometer code to binary code.

Figure 1. Typical N-stage SDL-TDC. **(a)** Basic SDL-TDC architecture with encoder transforming thermometer code into binary code; **(b)** Timing chart and the transformation of thermometer code to binary code.

3.2. SDL-TDC with linearity self-calibration using histogram method

In the SDL-TDC, ideally, the buffer delay is equal in all the stages. However, in reality, τ_i varies due to manufacturing variation, which yields non-linearity of the TDC. The self-calibration using histogram method can compensate for the non-linearity.

Figure 2 shows a four-stage SDL-TDC with linearity self-calibration using histogram method, where the gray-colored extra components as well as four inputs Test mode, CLK_U, CLK_L, and Stg, and the output of 1's counter Bin are added. The input Test mode controls the upper and lower 2-to-1 multiplexer (MUX). When Test mode = 0, the TDC is in the measurement mode, and Start input and Stop input signals are provided to the TDC.

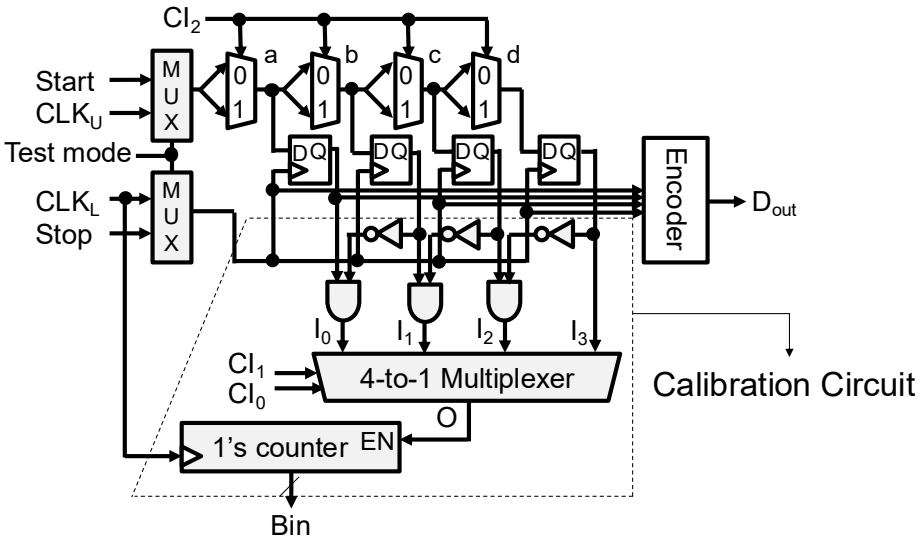


Figure 2. Four-stage SDL-TDC with linearity self-calibration using histogram method.

When Test mode = 1, the TDC is in the self-calibration mode. CLK_U is the upper input and CLK_L is the lower input, respectively. During self-calibration process, different oscillation signals are input to CLK_U and CLK_L, respectively. As a result, pseudo-random time interval sequence like that in Figure 3 where Δt_i is the i-th time interval is applied to the SDL-TDC. The time intervals follow a probabilistic uniform distribution with the minimum 0 and the maximum the period of CLK_U. The SDL-TDC measures each time interval sequential one by one and a histogram is constructed.

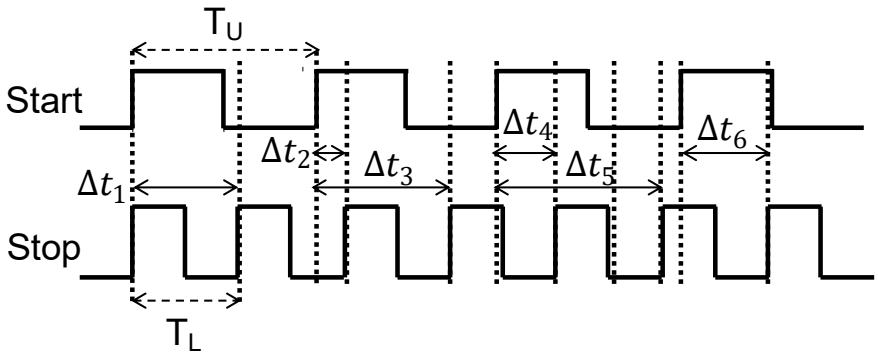


Figure 3. Example of applied pseudo-random time interval.

When the delay sequence is sufficiently random and the length is sufficiently long, the normalized distribution of the histogram reflects that of the buffer delay. For example, when the distribution of the constructed histogram is uniform, that of the buffer delay is uniform, too. Consequently, we can estimate each buffer delay from the constructed histogram.

When the effect of noise can be ignored and the initial time interval is 0, pseudo-random time interval sequence generated by two oscillation signals like those shown in Figure 3 is applied to TDC. The i-th time interval Δt_i is expressed by the following formula.

$$\Delta t_i = iT_L mod(T_U) \tag{1}$$

where T_U is the period of the oscillation signal applied to CLK_U, and T_L is that applied to CLK_L. When $\sum_{j=0}^{N-1} \tau_j < T_U/2$, the bin length of the i-th stage Bin_i except the last one is expressed by the following formula.

$$Bin_i = \tau_i N_{SMP}/T_U \tag{2}$$

where N_{SMP} is number of the applied time interval during the self-calibration.

The N-1-th stage bin length Bin_{N-1} is expressed by the following formula.

$$Bin_{N-1} = (T_U - \sum_{j=0}^{N-2} \tau_j)\, N_{SMP}/T_U \tag{3}$$

The bin length of each stage of a 4-stage SDL-TDC is shown in Figure 4.

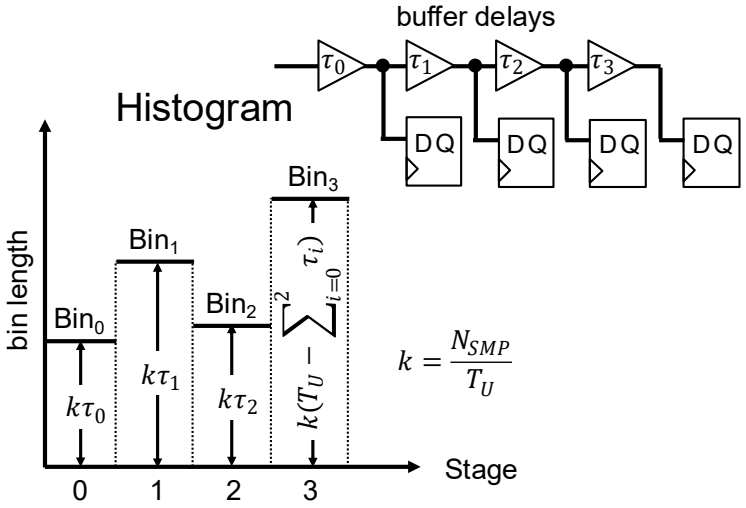


Figure 4. Relation between buffer delay and bin length of histogram of 4-stage SDL-TDC with self-calibration.

Transforming Equation (2) for τ_i gives

$$\tau_i = \frac{Bin_i}{N_{SMP}} T_U \tag{4}$$

The Equation (4) shows that the buffer delay is in proportion to the bin length.

The bin length is counted with the 1's counter. The following is the procedure to obtain the buffer delay τ_i.

Step 1: Set Test mode to 0 and initialize the 1's counter and assign i to Stg.

Step 2: After that, set Test mode to 1. Then calibration process starts in synchronization with the positive edge of Test mode. N_{SMP} pseudo random time intervals are measured. When D_{out} is i, EN input of the 1's counter becomes 1, and thus the 1's counter is incremented by 1.

Step 3: The output Bin after Step 2 is Bin_i. Substituting the bin length Bin_i into Equation (4), we obtain the buffer delay τ_i.

Figure 5 shows a timing chart of the self-calibration process for the target buffer delay of τ_2.

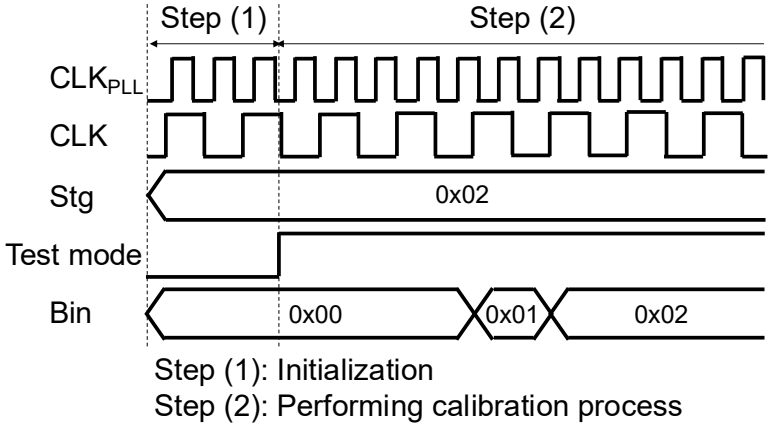


Figure 5. Timing chart of self-calibration process when target buffer delay is τ_2.

Linearity self-calibration can be used to observe variation of the buffer delay over resolution. It is called code density test [7–9,34].

A common approach for delivering two oscillation signals to the SDL-TDC is to place ring oscillators adjacent to CLK_U and CLK_L, respectively. However, the oscillation signals are weak and susceptible to external noise, which adversely affects the self-calibration process.

3.3. MPSDL-TDC

Since an FPGA has a huge amount of resources in general, it is suitable for implementing multi-channel TDCs. In multi-channel TDCs, equivalent fine resolution is realized with parallel delay measurement of a time interval with parallel identical TDCs even the resolution of the TDCs is low.

Figure 6 shows an MPSDL-TDC composed of three SDL-TDCs; it is composed of SDL-TDC_0, SDL-TDC_1, and SDL-TDC_2. A time interval is input to these TDCs in parallel. The buffer delays of the TDCs are variated although the TDCs have common ideal resolution. Let $t_{m,n}$ denote the minimum time interval for which the output D_{out} of SDL-TDC_m is equal to n. The upper three rows of the figure show an example of the distribution of $t_{m,n}$ ($0 \le m \le 2, 1 \le n \le 15$) in each SDL-TDC. Let t_n denote the minimum time interval for which the output D_{out} of MPSDL-TDC is equal to n. The last row shows the distribution of t_n. In this example, the resolution of the MPSDL-TDC is about one third of each SDL-TDC. Further, the number of the possible output values D_{out} is three times of that of each SDL-TDC.

We define MPSDL-TDC with M N-stage SDL-TDCs as M-parallel N-stage MPSDL-TDC. The resolution of M-parallel N-stage MPSDL-TDC is 1/M of that of each SDL-TDC. The number of the possible output values is MN.

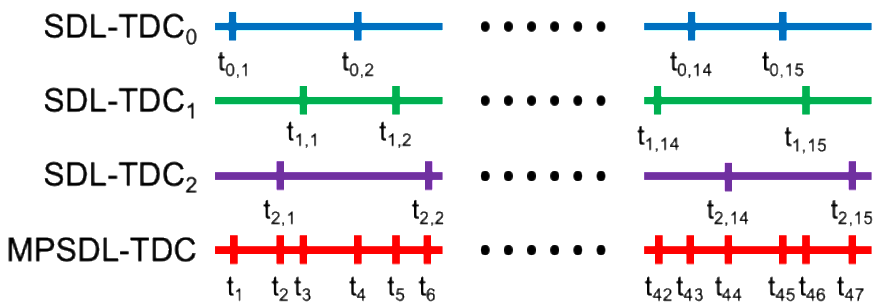


Figure 6. Basic idea of 3-parallel 16-stage MPSDL-TDC.

In MPSDL-TDCs, the output value is often calculated as the average of the output values of the internal TDCs. Figure 7 shows the architecture of a M-parallel N-stage MPSDL-TDC of this case. In this figure, TDC_i is the i-th SDL-TDC. D_{out} is the average of the output of all the SDL-TDCs that is calculated in Averaging Block.

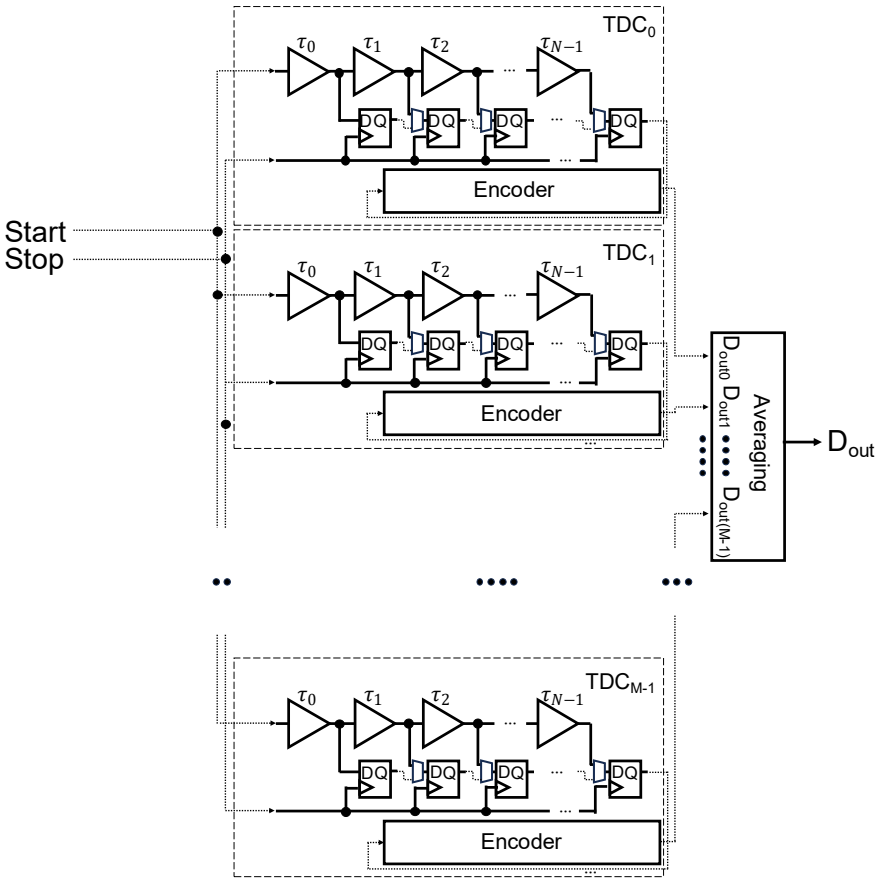


Figure 7. Architecture of M-parallel N-stage MPSDL-TDC.

4. Proposed PUF

This section describes the proposed PUF, which calculates two buffer delays in TDCs with the calibration process for the generation of a 1-bit response output from their difference. Subsection 4.1 presents its architecture. Subsection 4.2 explains its FPGA implementation using LUT-chain. Subsection 4.3 presents the scheme of the response output generation. Subsection 4.4 outlines the limitation of the proposed PUF.

4.1. PUF architecture

This subsection explains the architecture of the proposed PUF. Figure 8 shows the proposed PUF based on the MPSDL-TDC of Figure 7. It can be applied to other MPSDL-TDCs such as those of [7–9], too.

The proposed PUF has the external inputs, Start input, Stop input, PUF input, and the challenge input CI, and the external outputs, D_{out} and Bin. This PUF has M-parallel N-stage TDCs with linearity self-calibration using histogram method. Although the TDCs of [7] use carry-chains as the delay-lines, the proposed PUF uses buffer-chains implemented with LUTs for the delay-lines of the SDL-TDCs. Consequently, the proposed PUF can be implemented on most of FPGAs from high-end to low-cost FPGAs.

Let i-th SDL-TDC be TDC_i, and let the output of Encoder of TDC_i be D_{outi}. Let the output of Calibration Circuit of TDC_i be Bin_i. The output D_{out} is the average of D_{out} of all the M SDL-TDCs that is calculated by Averaging Block. The output Bin is the output of the M-to-1 multiplexer whose inputs are the bin length of all the M SDL-TDCs. A part of CI is the control input of the M-to-1 multiplexer.

The input PUF is the mode control input of the PUF. When PUF = 0, each SDL-TDC is in the measurement mode. Then the proposed PUF works as a normal MPSDL-TDC. Positive transitions are input from Start input and Stop input. After measurement, the fine-resolution measurement result is output to D_{out}. Calibration Circuit of TDC_i outputs Bin_i.

When PUF = 1, each SDL-TDC is in the calibration mode. Then the proposed PUF works as a PUF. Arbitrary buffer delay except τ_{N-1} of arbitrary TDC can be calculated with the self-calibration. Let $m = \lceil \log_2 M \rceil$ and $n = \lceil \log_2 N \rceil$. The output Bin reflects the $CI_{n-1}\ldots CI_0$-th buffer delay of the $CI_{m+n-1}\ldots CI_n$-th SDL-TDC after the calibration. The binary value $CI_{m+n-1}\ldots CI_n$ is a control value of the control input of the M-to-1 multiplexer. Calibration Circuit of TDC_i outputs Bin_i which reflects the $CI_{n-1}\ldots CI_0$-th buffer delay when the control value is $CI_{n-1}\ldots CI_0$.

During calibration, oscillation signals are input to CLK_U and CLK_L of the SDL-TDCs. They are sent from the outputs of the T Flip-Flops. All the T Flip-Flops connected with CLK_L work in synchronization with common CLK, and all the T Flip-Flops connected with CLK_U work in synchronization with common CLK_{PLL}. CLK and CLK_{PLL} are sent from PLLs. Thus, common time intervals are always sent to the SDL-TDCs.

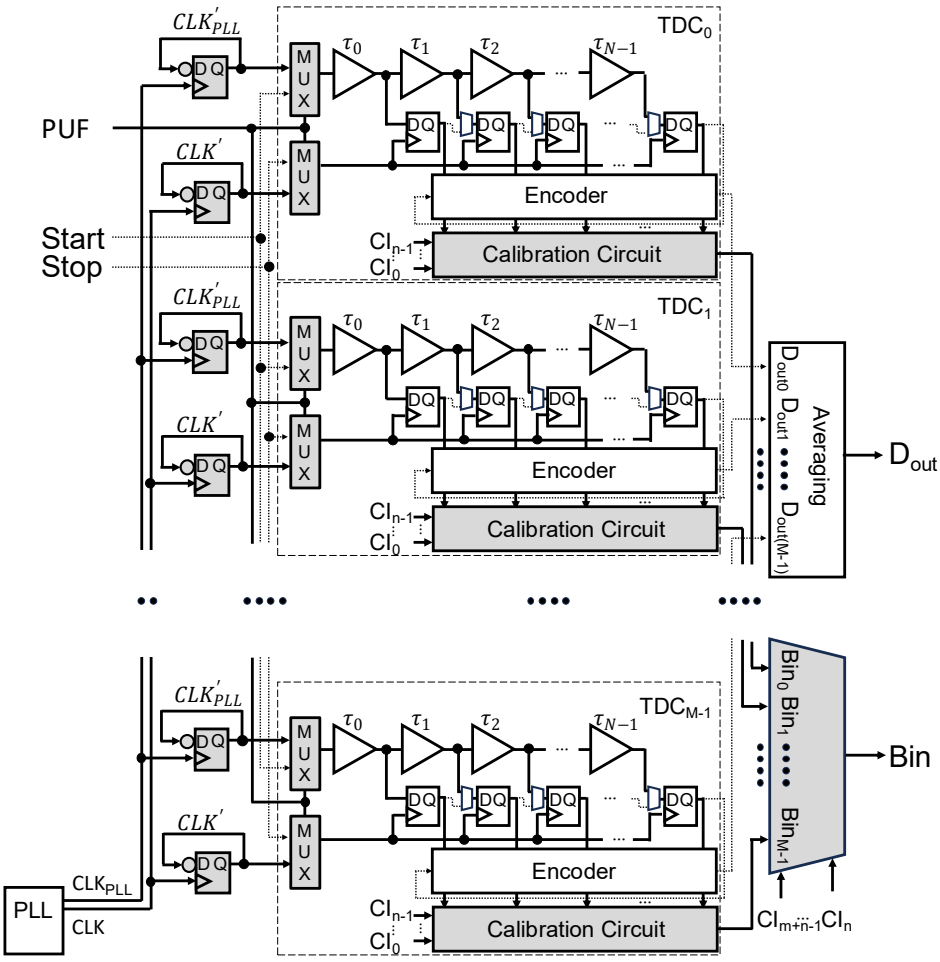


Figure 8. Proposed PUF based on M-parallel N-stage MPSDL-TDC.

4.2. Response output generation

The proposed PUF generates a 1-bit response output by comparing a pair of the buffer delays of the internal SDL-TDCs. The challenge input decides the two buffers. The delays are calculated with the bin length obtained with the calibration.

The n-bit response output $R = R_0R_1\cdots R_{n-1}$ is generated with the challenge input $CI = CI^0CI^1\cdots CI^{n-1}$, where R_k is the k-th bit of R and CI^k is the challenge input to generate R_k. The n-bit response output R is generated with each 1-bit response output from R_0 to R_{n-1} obtained with the corresponding challenge input sequentially one by one.

Figure 9 shows the format of CI^k. The challenge input CI^k consists of the first half challenge sub-input CI^{k0} and the second half challenge sub-input CI^{k1}. The first sub-input selects the first buffer. The second sub-input selects the second buffer. Here both sub-inputs are $(m + n)$-bit bitstrings. The higher m bits select SDL-TDC, and the lower n bits select the stage.

Figure 9. Format of challenge input C^k for generating 1-bit response output R_k.

First, we obtain Bin_0 with the calibration after CI^{k0} is set to CI of the proposed PUF. Then we obtain the first delay d_0 by substituting Bin_0 to Equation (4).

Second, we obtain Bin_1 with the calibration after CI^{k1} is set to CI of the proposed PUF. Then we obtain the second delay d_1 by substituting Bin_1 to Equation (4).

If $d_0 - d_1 > 0$, then the response output is 0, otherwise it is 1. Since the delay is in proportion to the bin length, it is equivalent to determine the response output based on the difference between Bin_0 and Bin_1.

The scheme for generating n-bit response output is as follows:

Step 1. Prepare CI.

Step 2. Set PUF active.

Step 3. $k = 0$.

Step 4. Execute calibration operation with CI^{k0}. Obtain Bin_0.

Step 5. Execute calibration operation with CI^{k1}. Obtain Bin_1.

Step 6. $R_k = 0$ if $Bin_0 - Bin_1 > 0$, otherwise $R_k = 1$.

Step 7. Increment k by 1.

Step 8. Finish if $k = n$, otherwise go to Step 4.

As operating voltage and temperature fluctuate, the delay characteristics of the buffer chains within the internal SDL-TDCs also change. However, the proposed PUF remains robust against such variations, as it derives its 1-bit response output from the relative difference in buffer delays. In this context, we assume that environmental variations affect all buffers uniformly—a condition that holds true in most cases of voltage and temperature changes within FPGAs.

Let the effect of environmental variation to the bin length obtained by the calibration be a constant K. Then the length of bins obtained by the calibration becomes $KBin_0$ and $KBin_1$. In this case, K gives no effect to the response output in theory.

The entropy source of the proposed PUF is the buffer delay. Accordingly, the performance as a PUF becomes greater as variation of the buffer delay larger.

On the other hand, when variation of delay of the buffers is large, the performance as a TDC is decreased for its serious non-linearity. In this case, D_{out} of each TDC is compensated with the compensation method such as [36] before averaging.

4.3. Limitation of proposed PUF

The performance of the proposed PUF depends on the architecture of the target MPSDL-TDC. Table 1 shows the target FPGA and the parameters of the conventional MPSDL-TDCs. The 1st column is the reference of each MPSDL-TDC, and the 2nd column is the implemented FPGA. The 3rd column is the number of TDCs M and the 4th column is the number of the stages of a TDC N. Here, the maximum M is 16, and the maximum N is 260.

The 5th column is the number of the buffers. Then number of candidates of the buffer is 4144 in maximum. In case of RO PUF, it is equivalent to the number of ROs. Compared with well-known RO PUFs from [18,22,23], the order of the number of the delay elements is equal or rather larger.

Table 1. Device and parameters of the Conventional MPSDL-TDCs.

Reference	Device	Number of TDCs M	Number of stages N	Number of buffers
[7]	VirtexII-Pro	10	256	2550
[8]	Spartan 6	16	64	1008
[9]	Virtex-6	16	260	4144

5. Experimental results

We evaluated the proposed PUF based on a 15-parallel 16-stage MPSDL-TDC.

5.1. Experimental setup

We have implemented the proposed PUFs on 10 AMD Artix-7 (XC7A35T) FPGAs. The implementation used AMD Vivado 2023.1. The reference clock was 100MHz, and the system clock CLK was also 100 MHz. The PLL clock CLK_{PLL} was 105.7 MHz. The sampling number N_{SMP} of D_{out} in the calibration process was 2^{21}. MicroBlaze which is AMD soft core processor also implemented to control the proposed PUF. Pseudo random number generated with the emulation of LFSR on MicroBlaze was used as the challenge inputs. The length of a challenge input was 8 bits. The length of a response output was 128 bits. 1-out-8 masking scheme is applied for the response output generation [18].

5.2. Basic evaluation metrics

We evaluated uniqueness, reliability, randomness, and uniformity as the basic evaluation metrics at room temperature and nominal voltage. In this evaluation, uniqueness was evaluated with inter-chip Hamming distance (HD). Let the number of PUFs used for the evaluation be N_{PUF}, and let the length of the response output be n (bits). Then the inter-chip HD HD_{INTER} (%) is expressed as the following formula [37]:

$$HD_{INTER} = \frac{2}{N_{PUF} \cdot (N_{PUF} - 1)} \sum_{i=0}^{N_{PUF}-2} \sum_{j=i+1}^{N_{PUF}-1} \left(\frac{HD(R_i, R_j)}{n} \right) \times 100.0 \quad (5)$$

where $HD(R_i, R_j)$ is the Hamming distance between the response outputs R_i and R_j.

The intra-chip HD of the k-th PUF $(HD_{INTRA})_k$ is expressed as the following formula [37]:

$$(HD_{INTRA})_k = \frac{1}{N_q}\sum_{t=0}^{N_q-1}\frac{HD(R_k, R'_{k,t})}{n} \times 100.0 \quad (6)$$

where N_q is times of queries, and $R'_{k,t}$ is the response output of the t-th query.

With $(HD_{INTRA})_k$, Reliability (%) is defined as:

$$Reliability = 100.0 - \frac{1}{N_{PUF}}\sum_{k=0}^{N_{PUF}-1}(HD_{INTRA})_k \quad (7)$$

The metric Randomness (%) is the frequency of 1 in all the evaluated PUFs, which is expressed as the following formula [38]:

$$Randomness = \frac{1}{nN_{PUF}}\sum_{k=0}^{N_{PUF}-1}\sum_{l=0}^{n-1} r_{k,l} \times 100.0 \quad (8)$$

The metric uniformity is the frequency of 1 in the response output of a PUF. The uniformity of the k-th PUF $(Uniformity)_k$ (%) is expressed as the following formula [37]:

$$(Uniformity)_k = \frac{1}{n}\sum_{l=0}^{n-1} r_{k,l} \times 100.0 \quad (9)$$

where $r_{k,l}$ is the l-th bit of R_k. In this evaluation, N_{PUF} = 10, n = 128, N_q = 128.

Table 2 shows the evaluation result of uniqueness, randomness, and reliability of the proposed PUF and the conventional weak FPGA PUFs. The first and second columns are the reference of each PUF and type of each PUF, respectively. The third and fourth columns are evaluation results of HD_{INTER} and Reliability, respectively.

In terms of uniqueness, the HD_{INTER} of the proposed PUF is moderate compared to other PUFs. With respect to reliability, it performs slightly lower than conventional designs. However, in terms of randomness, the proposed PUF exhibits comparable characteristics to those of conventional PUFs.

Table 2. Evaluation result of uniqueness, reliability, and randomness.

Reference	Type	HD_{INTER}	Reliability	Randomness
[18]	RO PUF	46.15	99.52	-
[23]	RO PUF	41	99.29	-
[24]	RO PUF	50.013	98.875	-
[21]	NAND-PUF	49.50	98.62	50.20
[21]	XOR-PUF	49.47	98.94	50.24
[21]	DD-PUF	46.88	98.33	50.29
Proposed	TDC PUF	47.07	93.91	50.70

The distributions of inter-chip and intra-chip Hamming distances are shown in Figure 10. Figure 11 shows the uniformity of the evaluated PUFs. The horizontal axis is ID of the PUFs, and the vertical axis is uniformity of each PUF. The uniformity distributes around the ideal value 50%, and the standard deviation of the distribution is 3.3%.

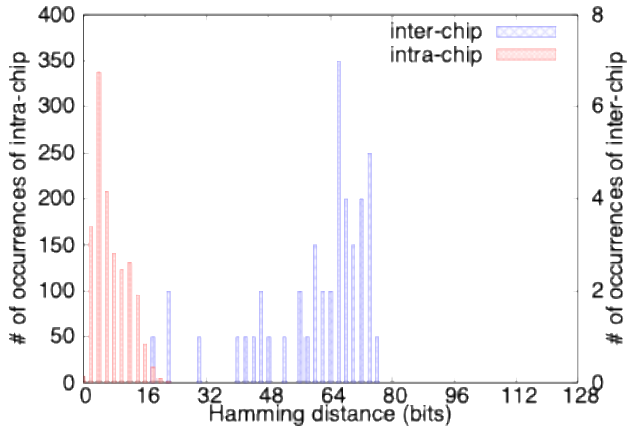


Figure 10. Distribution of inter-chip and intra-chip Hamming distance.

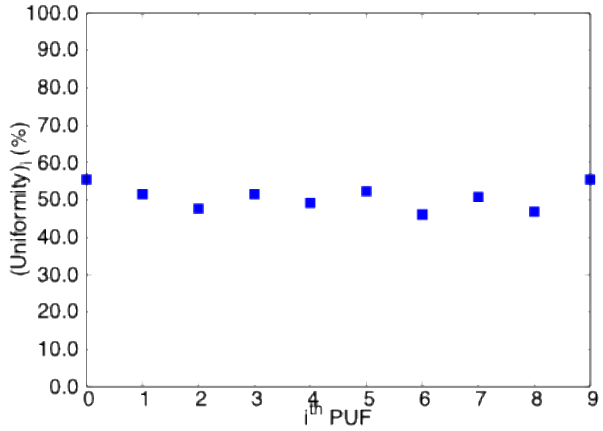


Figure 11. Uniformity of evaluated PUFs.

5.3. Resource overhead

We evaluated the resource overhead of the proposed PUF shown in Figure 8 over the applied original MPSDL-TDC in Figure 7. There, we examined the resource overhead of two implementations. One implements Calibration Circuits on FPGA. Let the resource overhead of the implementation be ro_0.

The other ran the process in Calibration Circuit on MicroBlaze. Let the resource overhead of the implementation be ro_1. Both implementations ran the process in Averaging Block on MicroBlaze. The resource overhead of a resource ro_0 was defined as $ro_0 = 100.0 \times (res_{PUF0} - res_{TDCs})/res_{TDCs}$, where res_{PUF0} was the target resource for the proposed PUF of the 1st implementation and res_{TDCs} is the resources for the original MPSDL-TDC. The resource overhead of a resource ro_1 was defined as $ro_1 = 100.0 \times (res_{PUF1} - res_{TDCs})/res_{TDCs}$, where res_{PUF1} was the target resource for the proposed PUF of the 2nd implementation. The amounts of resources for these implementations are estimated with the report of resource utilization after synthesis.

Table 3 shows the result. The first column is the resource name. The second, third, and fourth columns are res_{TDCs}, res_{PUF0}, and res_{PUF1}, respectively. The 5th and 6th column are ro_0 and ro_1, respectively. The number of resources F7 MUXes and F8 MUXes in res_{TDCs} is 0. Accordingly, we do not calculate ro_0 and ro_1 of these resources. The overhead ro_0 of Slice LUTs is 102.62%, which means that the number of the resources for Calibration Circuit is almost equivalent to that for TDCs.

The overhead ro_0 of Slice Registers is 270.15%, which means that the number of the resource for Calibration Circuit is more than twice of that for TDCs. According to above results, the implementation of Calibration Circuit on FPGA needs extra resources more than twice of the original circuit.

The overhead ro_1 of Slice LUTs and Slice Registers are about 5%, which means that the overhead for LUTs and Registers is quite low. Further, as the range of TDCs is longer, the overhead becomes lower.

Table 3. Resource utilization in Artix-7 FPGA.

Resource	res_{TDCs}	res_{PUF0}	res_{PUF1}	ro_0	ro_1
Slice LUTs	762	1544	800	102.62%	4.99%
Slice Registers	268	992	282	270.15%	5.22%
F7 Muxes	0	19	4	-	-
F8 Muxes	0	2	2	-	-

5.4. Thermal specification

The thermal specification of the proposed PUF was evaluated with the PUF of ID 4. First, the thermal specification of the histogram obtained with the self-calibration was evaluated. With the self-calibration, the histograms of TDC_0 of the PUF of ID 4 were obtained at 30 ℃, 60 ℃, and 80 ℃. 80 ℃ is the maximum temperature of Artix-7 [39]. Figure 12 shows the histograms. The horizontal axis is the stage number of the TDC, and the vertical axis is bin length. Each stage has 30 ℃, 60 ℃, and 80 ℃ bins from left to right. The bin length decreases as temperature increases in most of the stages. The length of 10-th bin is rather larger for non-linearity of the TDC.

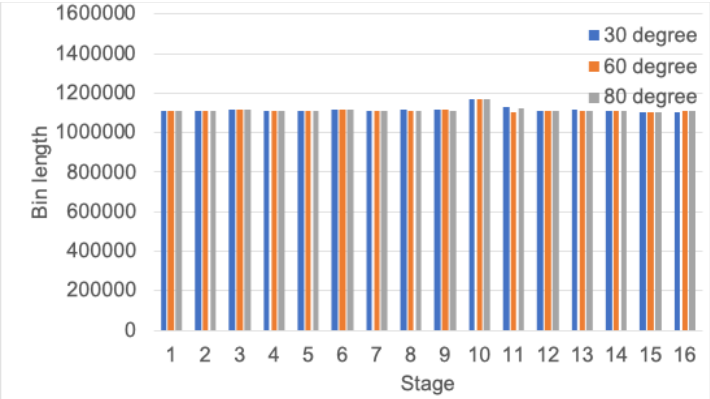


Figure 12. Histograms at 30, 60, and 80 ℃ of TDC_0 of 4-th PUF.

Next the thermal specification of the reliability of the proposed PUF was evaluated. Table 4 shows the result. As temperature increases, reliability decreases. However, the decrease is 2.70% in spite of variation of bin length shown in Figure 12. It is because the proposed PUF generates response output with the difference of delays.

Table 4. Thermal specification of reliability at 30, 60, and 80 ℃.

30 ℃	**60** ℃	**80** ℃
92.48%	91.82%	89.78%

5.5. Voltage specification

The voltage specification of reliability of the proposed PUF was evaluated with the PUF of ID 9. The normal core voltage of Artix-7 is 1.0 V. Reliability of the PUF was evaluated with 0.9 V 10% smaller core voltage and 1.1 V 10% larger core voltage as well as 1.0 V core voltage. Table 5 shows the result. Reliability becomes lower under both 0.9 V and 1.1 V core voltages. Comparing with the result of Table 4, we know that variation of core voltage is more effective for reliability than variation of temperature.

Table 5. Voltage specification of reliability at 0.9, 1.0, and 1.1 V.

0.9 V	1.0 V	1.1 V
75.06%	92.83%	67.87%

6. Discussion

Table 6 shows the comparison of the performance of the proposed PUF with major FPGA PUFs. The first column is the type of each PUF. The second column shows whether the PUF is a Weak PUF (W) or a Strong PUF (S). The third column shows if the PUF needs extra configuration or not. The 4th, 5th, 6th, and 7th columns are uniqueness, reliability, area overhead over functional circuit, and power consumption, respectively. Here we list Arbiter PUF, RO PUF, and Butterfly PUF as major FPGA PUFs. We compare the proposed PUF with these PUFs with scoring them from 1 to 5 in terms of each metric.

Table 6. Comparison of performance of major FPGA PUFs.

Type	W/S	Extra Configuration	Uniqueness	Reliability	Area Overhead	Power Consumption
Arbiter	S	not needed	4	5	4	4
RO	W	needed	4	5	5	2
Butterfly	W	needed	4	5	5	4
Proposed	W	not needed	4	4	4	2

5: Excellent, 4: Good, 3: Fair, 2: Poor

A major advantage of the proposed PUF is that it does not require extra configuration to generate response outputs. Extra configuration will be a unique evaluation metric for FPGA PUFs. FPGA PUFs that have a substantial area such as RO PUF and Butterfly PUF needs extra configuration. To obtain response outputs, an extra configuration data for PUFs is written to FPGA after the existing configuration data for functional circuit is backed up. Then response outputs are generated. Finally, the configuration data backed up for the functional circuits is written back to the FPGA. As a result, the above process causes overhead of time for generating response outputs and data for the extra configurations.

On the other hand, since the proposed PUF is based on existing TDCs, it does not require extra configuration. Consequently, it does not have such overhead. Uniqueness of the proposed PUF is equivalent to the other FPGA PUFs. Reliability is a little bit lower. One of the reasons is sequential execution of the two calibrations for generating 1-bit response output generation. Voltage bounce or fluctuation during the calibrations surfaces as variation of the external environment during the two calibrations. Probably, it causes degrading of its reliability. Reliability may improve if the two calibrations are executed in parallel under the same external environment. Since TDCs require Flip-Flops of the same number as the number of stages, the area is larger than Arbiter PUF of the equivalent number of stages.

However, since the proposed PUF reuses an existing high-resolution TDC, the area overhead, and power consumption as a PUF is low. Area overhead of RO PUF and Butterfly PUF over the functional circuit is considered to be zero since it is implemented with different configuration data [40].

The proposed PUF is a kind of delay PUF. Aging of PUF is an important and active research topic in the research area of hardware security [41]. Evaluation of effect of aging and the mitigation against it are future works.

7. Conclusion

This paper has presented a Weak FPGA PUF using MPSDL-TDC on FPGA. In the measurement mode, the proposed PUF works as a high-resolution FPGA TDC with the MPSDL-TDC. In the PUF mode, the proposed PUF selects a stage of a TDC in the MPSDL-TDC and another stage of another TDC in the MPSDL-TDC. Next, we calculate the buffer delays of the selected stages with the linearity self-calibration. Finally, we obtain a 1-bit response output by comparing the buffer delays. With a little extra circuit, the proposed PUF can be applied to any type of MP-SDL TDCs, even it is the one with LUT chain based TDCs or with the dedicated carry-chains based TDCs. Consequently, the number of the extra resources for the proposed PUF is small. Evaluation using 10 Artix-7 FPGAs resulted in an inter-chip Hamming distance of 47.07%, a reliability of 93.91%, and a randomness of 50.70%. Minimum resource overhead for the proposed PUF over original MDSDL-TDCs is 4.99%.

Data availability statement

No supplementary or additional data were generated in this study.

Authors' contribution

Conceptualization, KK and HK; writing—original draft preparation, KK; writing—review and editing, TN and HK; visualization, KK; supervision, TN and HK; project administration, KK. All authors have read and agreed to the published version of the manuscript.

Conflicts of interests

The authors declare no conflict of interest.

References

[1] Chowdhury S, Acharya R, Boullion W, Felder A, Howard M, *et al*. A weak asynchronous RESet (ARES) PUF using start-up characteristics of null conventional logic gates. In *Proceedings of 2020 IEEE International Test Conference (ITC)*, Washington, USA, November 1–6, 2020, pp. 1–10.

[2] Prabhunath GC, Shah AP. Fredkin gate-based feed-forward arbiter PUF design on FPGA. In *Proceedings of IEEE International Conference on Microelectronics*, Doha, Qatar, December 14–17, 2024, pp. 1–5.

[3] Abideen ZU. NIST 800-22 statistical validation of SRAM-based PUFs for hardware security. In *Proceedings of 2025 IEEE 34st Microelectronics Design & Test Symposium*, Albany, USA, May 19–20, 2025, pp. 1–4.

[4] Georgoulas D, Tsiatouhas Y, Tenentes V. CAS-PUF: current-mode array-type strong PUF for secure computing in area constrained SoCs. In *Proceedings of IEEE Design, Automation & Test in Europe Conference*, Lyon, France, March 31–April 2, 2025, pp. 1–7.

[5] Szyduczynski J, Koscielnik D, Miskowicz M. Time-to-digital conversion techniques: a survey of recent developments. *Measurement* 2023, 214(112762):1–17.

[6] Machado R, Cabral J, Alves FS. Recent developments and challenges in FPGA-based time-to-digital converters. *IEEE Trans. Instrum. Meas.* 2019, 68 (11):4205–4221.

[7] Daigneault M, David JP. A novel 10 ps resolution TDC architecture implemented in a 130 nm process FPGA. In *Proceedings of the 8th IEEE International NEWCAS Conference 2010*, Montreal, Canada, June 20–23, 2010, pp. 281–284.

[8] Szplet R, Kwiatkowski P, Jachna Z, Rozyc K. Several issues on the use of independent coding lines for time-to-digital conversion. In *Proceedings of 2013 IEEE Nordic-Mediterranean Workshop on Time-to-Digital Converters*, Perugia, Italy, October 3, 2013, pp. 1–8.

[9] Shen Q, Liu S, Qi B, An Q, Liao S, *et al*. A 1.7 ps equivalent bin size and 4.2 ps RMS FPGA TDC based on multichain measurements averaging method. *IEEE Trans. Nucl. Sci.* 2015, 62(3):947–954.

[10] Kumar SS, Guajardo J, Maes R, Schrijen GJ, Tuyls P. The butterfly PUF: protecting IP on every FPGA. In *Proceedings of 2008 IEEE International Workshop on Hardware-Oriented Security and Trust*, Anaheim, USA, June 9, 2008, pp. 67–70.

[11] Talukder BMSB, Ray B, Forte D, Rahman MT. PreLatPUF: exploiting DRAM latency variations for generating robust device signatures. *IEEE Access* 2019, 7:81106–81120.

[12] Ahsan SMM, Hossain T, Hasan MS, Hoque T. Resistive RAM-based PUF: challenges and opportunities. In *Proceedings of 2023 IEEE 16th Dallas Circuits and Systems Conference (DCAS)*, Denton, USA, April 14–16, 2023, pp. 1–6.

[13] Hu Y, Wu L, Chen Z, Huang Y, Xu X, *et al*. STT-MRAM based reliable weak PUF. *IEEE Trans. Comput.* 2022, 71(7):1564–1574.

[14] Taneja S. Energy-efficient and low-cost hardware security primitives for secure ubiquitous computing. In *Proceedings of IEEE International Midwest Symposium on Circuits and Systems*, Fukuoka, Japan, August 7–10, 2022.

[15] Li D, Yang K. A self-regulated and reconfigurable CMOS physically unclonable function featuring zero-overhead stabilization. *IEEE J. Solid-State Circuits* 2020, 55(1):98–107.

[16] Xi X, Zhuang H, Sun N, Orshansky M. Strong subthreshold current array PUF with 2^{65} challenge-response pairs resilient to machine learning attacks in 130 nm CMOS. In *Proceedings of 2017 Symposium on VLSI Circuits*, Kyoto, Japan, June 5–8, 2017, pp. C268–C269.

[17] Chen Y, Chang S. A physically unclonable function embedded in a SAR ADC. In *Proceedings of 2022 IEEE International Test Conference in Asia*, Taipei, China, August 24–26, 2022, pp. 85–89.

[18] Suh GE, Devadas S. Physical unclonable functions for device authentication and secret key generation. In *Proceedings of 2007 44th ACM/IEEE Design Automation Conference*, San Diego, USA, June 4–8, 2007, pp. 9–14.

[19] Eddla A, Pappu VYJ. Low area and power-efficient FPGA implementation of improved AM-CSA-IIR filter design for the DSP application. *Int. J. Electr. Electron. Eng. Telecommun.* 2022, 11(4):294–303.

[20] Anandakumar NN, Hashmi MS, Sanadhya SK. Efficient and lightweight FPGA-based hybrid PUFs with improved performance. *Microprocess. Microsyst.* 2020, 77:103180.

[21] Sala RD, Scotti G. Evaluation and comparison of physical unclonable functions suitable for FPGA implementation. In *Proceedings of 2024 39th Conference on Design of Circuits and Integrated Systems*, Catania, Italy, November 13–15, 2024, pp. 1–6.

[22] Maiti A, Schaumont P. Improving the quality of a physical unclonable function using configurable ring oscillators. In *Proceedings of 2009 International Conference on Field Programmable Logic and Applications*, Prague, Czech Republic, August 31–September 2, 2009, pp. 703–707.

[23] Xin X, Kaps JP, Gaj K. A configurable ring-oscillator-based PUF for Xilinx FPGAs. In *Proceedings of 2011 14th Euromicro Conference on Digital System Design*, Oulu, Finland, August 31–September 2, 2011, pp. 651–657.

[24] Pei S, Zhang J, Wang R. A low-overhead RO PUF design for Xilinx FPGAs. *IEICE Electron. Express* 2018, 15(5):20180093–20180093.

[25] Gupta A, Naz SF, Shah AP. Configurable RO-PUF with improved thermal stability for lightweight applications. In *Proceedings of 2024 International Conference on Microelectronics (ICM)*, Doha, Qatar, December 14–17, 2024, pp. 1–6.

[26] Anandakumar NN, Hashmi MS, Chaudhary MA. Implementation of efficient XOR arbiter PUF on FPGA with enhanced uniqueness and security. *IEEE Access* 2022, 10:129832–129842.

[27] Katoh K, Nakura T, Kobayashi H. A strong physical unclonable function using dual time-to-digital converters for AMD FPGAs. In *Proceedings of 2025 IEEE 14th International Conference on Communications, Circuits and Systems (ICCCAS)*, Wuhan, China, May 23–25, 2025, pp. 35–40.

[28] Ni T, Wu H, Nie M, Yan A, Wang S, *et al*. A response-nonlinearized DEMUX-TDC PUF for resistance against modeling attacks and secure authentication protocols. *IEEE Trans. Very Large Scale Integr. VLSI Syst.* 2025, 33(10):2606–2619.

[29] Li M, Huang PX, Park J, Mathew SK, De V, *et al*. GUARD: a fully-digital TDC-based clock and voltage glitch detector with on-demand protection in a 28 nm CMOS. In *Proceedings of 2025 Symposium on VLSI Technology and Circuits (VLSI Technology and Circuits)*, Kyoto, Japan, June 8–12, 2025.

[30] Ebrahimabadi M, Viera R, Guilley S, Danger JL, Dutertre JM, *et al*. Multi-sensor data fusion for enhanced detection of laser fault injection attacks in cryptographic hardware: practical results. In *Proceedings of 2025 Design, Automation & Test in Europe Conference*, Lyon, France, March 31–April 2, 2025, pp. 1–2.

[31] Katoh K, Yamamoto S, Zhao Z, Zhao Y, Katayama S, *et al*. A physically unclonable function using time-to-digital converter with linearity self-calibration and its FPGA implementation. In *Proceedings of 2023 IEEE International Test Conference in Asia (ITC-Asia)*, Matsue, Japan, September 12–14, 2023, pp. 1–6.

[32] Wu J, Shi Z. The 10-ps wave union TDC: improving FPGA TDC resolution beyond its cell delay. In *Proceedings of 2008 IEEE Nuclear Science Symposium Conference Record*, Dresden, Germany, October 19–25, 2008, pp. 3440–3446.

[33] Chaberski D. Time-to-digital-converter based on multiple-tapped-delay-line. *Measurement* 2016, 89:87–96.

[34] Hua Y, Chitnis D. A highly linear and flexible FPGA-based time-to-digital converter. *IEEE Trans. Ind. Electron.* 2022, 69(12):13744–13753.

[35] Katoh K, Nakura T, Kobayashi H. A 10ps-order flexible resolution time-to-digital converter with linearity calibration and legacy FPGA. In *Proceedings of 2025 Design, Automation & Test in Europe Conference*, Lyon, France, March 31–April 2, 2025, pp. 1–2.

[36] Ito S, Nishimura S, Kobayashi, H, Uemori S, Tan Y, *et al*. Stochastic TDC architecture with self-calibration. In *Proceedings of 2010 IEEE Asia Pacific Conference on Circuits and Systems*, Kuala Lumpur, Malaysia, December 6–9, 2010, pp. 1027–1030.

[37] Maiti A, Gunreddy V, Schaumont P. A systematic method to evaluate and compare the performance of physical unclonable functions. In *Embedded Systems Design with FPGAs*, 1st ed. New York: Springer New York, 2012. pp. 245–267.

[38] Chen Y, Cui X, Ye W, Cui X. The security enhancement techniques of the double-layer PUF against the ANN-based modeling attack. In *Proceedings of 2021 IEEE International Test Conference (ITC)*, Anaheim, USA, October 10–15, 2021, pp. 63–72.

[39] AMD. Artix-7 FPGAs Data Sheet: DC and AC Switching Characteristics v1.27. 2022. Available: https://docs.amd.com/v/u/en-US/ds181_Artix_7_Data_Sheet (accessed on 27 October 2025).

[40] Zeng G, Ito H. Efficient test data decompression for system-on-a-chip using an embedded FPGA core. In *Proceedings 2023 IEEE Symposium on Defect and Fault Tolerance in VLSI Systems*, Boston, USA, November 5, 2003, pp. 503–510.

[41] Karimi N, Danger JL, Guilley S. Impact of aging on the reliability of delay PUFs. *J. Electron. Test.* 2018, 34:571–586.