# A lightweight and collapse-response-resistant PUF using obfuscation-feedback-shift-register

Chen, Z., Lee, W., Hong, Q., Gu, C., Guan, Z., Ding, L., & Zhang, J. (2022). A lightweight and collapse-response-resistant PUF using obfuscation-feedback-shift-register. *IEEE Transactions on Circuits and Systems II: Express Briefs*, *69*(11), 4543-4547. https://doi.org/10.1109/TCSII.2022.3193002

**Published in:**
IEEE Transactions on Circuits and Systems II: Express Briefs

**Document Version:**
Peer reviewed version

# A Lightweight and Collapse-Response-resistant PUF using Obfuscation-Feedback-Shift-Register

Zhuojun Chen, Wenshang Lee, Qinhui Hong, Chongyan Gu, Zhenyu Guan, Lin Ding, and Jiliang Zhang

***Abstract*—Physical Unclonable Function (PUF) is a lightweight hardware security primitive and suitable for device authentication in Internet of Things (IoT). However, each strong PUF instance is required to store at least $10^6$ reliable challenge-response pairs (CRPs) in the center nodes, which brings an excessive storage overhead since such nodes connect massive remote PUFs. In this paper, an obfuscation-feedback-shift-register (OFSR) PUF is designed, which consists of certain numbers of weak PUF cells working with an obfuscation mechanism. It can efficiently overcome the collapse response resulted from normal linear-feedback-shift-register (LFSR), and provide higher security. Experimental results show that the proposed PUF has ideal performance on reliability, uniqueness, uniformity, randomness, and good resistance to machine learning attacks.**

***Index Terms*—Physical Unclonable Function, Hardware Security, Obfuscation, Machine Learning.**

## I. Introduction

IN the Internet of Things (IoT), secret key generation and device authentication are two crucial technologies to protect information security and privacy. The traditional schemes store keys in EEPROM or Battery-powered SRAM and employ crypto modules to implement secure data transmission and authentication. However, continuous power consumption and complex operations of crypto modules bring unacceptable overhead to IoT devices that commonly afford limited resources, e.g., CPU, memory, and battery power.

Physical Unclonable Function (PUF) [1] is an emerging hardware security primitive and provides a candidate solution for lightweight key generation and authentication. Due to uncontrolled and unpreditable process variations, each PUF instance generates a unique mapping relationship between the challenge and response. Generally, PUFs are categorized into two groups: strong PUFs and weak PUFs, according to the number of challenge-response pairs (CRPs).

Manuscript received July 30, 2021. This work is supported by the National Natural Science Foundation of China (No. 62122023, No. U20A20202, No. 61874042 and No. 61804053), the Science and Technology Innovation Program of Hunan Province under Grant No. 2021RC4019, the Hunan Natural Science Foundation for Distinguished Young Scholars under Grant No. 2020JJ2010, key laboratory of network assessment technology, CAS (Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093), State Key Laboratory of Computer Architecture (ICT, CAS) (No. CARCHB202011), and Hangzhou Innovation Institute and Beihang University (No. 2020-Y9-A-012). (*Corresponding author: Jiliang Zhang*)

Z. J. Chen, W. S. Lee, Q. H. Hong, L. Ding, and J. L. Zhang are with College of Semiconductors (College of Integrated Circuits), Hunan University, Changsha 410082, China. (e-mails: zhangjiliang@hnu.edu.cn)

C. Gu is with the Centre for Secure Information Technologies (CSIT), Institute of Electronics, Communications & Information Technology (ECIT), Queen's University Belfast (QUB), U.K., BT3 9DT. (e-mails: c.gu@qub.ac.uk)

Z. Y. Guan are with Beihang University, Beijing 100191, China.

Strong PUFs provide an exponential number of CRPs, which can resist reuse attacks, and hence are employed for authentication. Arbiter PUF (APUF) and its variants [2, 3] are classic strong PUFs. Those PUFs employ the arbiter to generate a Boolean bit as the PUF's response according to delay difference between parallel paths controlled by multiplexers and challenge. On the contrary, weak PUFs [4, 5] only have limited CRPs, and are suitable for secret keys generation.

Although strong PUFs show great advantages over traditional security schemes, at least $10^6$ reliable CRPs must be stored for state-of-the-arts [2, 3, 6], which results in unreasonable storage overhead for service nodes of IoT connecting with mass devices. To reduce the storage space, this paper develops an OFSR-PUF employing the ES unit designed by [7]. Its main contribution and innovation are as follows.

- A new strong PUF is designed, in which dozens of weak PUF cells are embedded into the linear-feedback-shift-register (LFSR) structure. Since the reliability of the proposed PUF relies on such cells, only reliability data of cells is required to store in the enrollment phase, which is linear in the cell number.
- A collapse response generated by the LFSR-based PUF designs is first revealed in this paper, which weakens the security of the response. In order to address the issue, a nonlinear obfuscation mechanism is proposed.
- To output reliable response, an ES selection scheme is presented, which utilizes a reliability information matrix and a certain numbers of AND gates acting as switches to ensure that the unreliable outputs are shielded.

The remainder of this paper is organized into four sections. Section II describes the related works. The proposed scheme is represented in section III. The following section shows and discusses the experiment results. At last, section V presents the conclusion of the entire work.

## II. Related works

### A. PUF Designs for Storage Space Reduction

The traditional strong PUF-based authentication protocols commonly require storing vast reliable CRPs, bringing extreme storage overhead to center nodes and servers. To overcome this issue, some schemes storing the soft PUF models have been proposed in the literature [8, 9]. However, such works induce extra and high overhead. For example, [8] contains 4KB RAM and 64KB ROM blocks; and [9] demands a expensive secure memory to store secret seed and a amount of training time (hours to weeks) in the registration stage. Another type of design utilizes weak PUF cells and cryptography

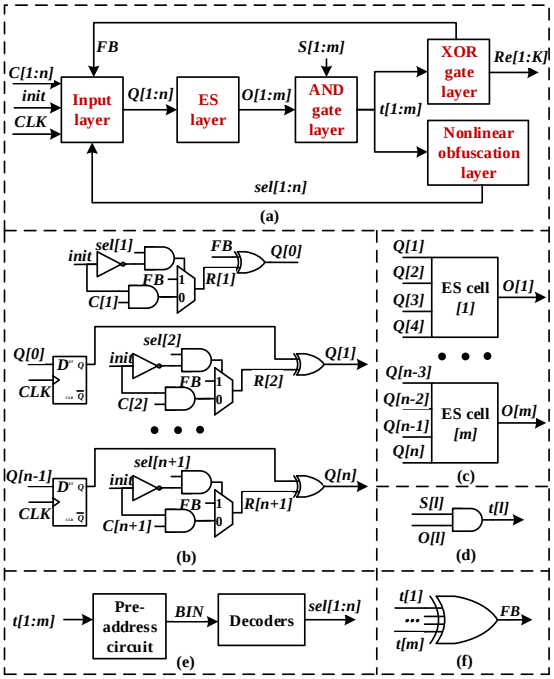


Fig. 1: (a) The top structure of OFSR-PUF; (b) the input layer; (c) the ES layer; (d) the AND gate layer; (e) the nonlinear obfuscation layer; and (f) the XOR gate layer.

circuits to form a strong PUF. Since the reliability relies on employed cells, center nodes only store reliability data of cells, which is linear to cell number. [10] devises a core logic unit based on inverters and multiplexers, which is inserted into an LFSR. Each core logic generates a 1-bit response according to the initial challenge. Its reliability is controlled by clock cycles, and hence only reasonably short clock data are stored. Recently, some similar works [11, 12] have been presented. However, such LFSR-based arts overlook that the collected output sequences of LFSR-PUFs may leak information about unused CRPs and lower the security in the authentication. This case is called collapse response in this paper.

### B. Configurable Cross-Coupled Inverter

The ES cell [7] consists of a modified cross-coupled inverter-pair and two configurable clock delay circuits on both sides, in which a 4-bit challenge selects components to generate different voltages marked as "01" or "10" at output nodes. Hence, such a cell is characterized by a 16-row unique look-up table recording an unique input-output mapping. Although a framework is provided to form a strong PUF using such cells, the employed 16 Sboxs and 128 cells cause large overhead. Meanwhile, the challenge selection scheme requires verifiers calculate proper challenges by a loop process for each PUF instance, which increases the server load in IoT scenario. This paper provides an optimization version with lower overhead and removes the load of this scheme from center devices.

## III. The Proposed OFSR-PUF using Configurable Cross-Coupled Inverters

This section introduces the proposed OFSR-PUF structure, the collapse response, the obfuscation mechanism, and the tailor-made ES selection scheme. Note that for two ordered sequences $A = a_1a_2\cdots a_{|A|}$ and $B = b_1b_2\cdots b_{|B|}$, a sign $\boxplus$ marks a concatenation operation and follows such rules: $A \boxplus B = a_1a_2\cdots a_{|A|}b_1b_2\cdots b_{|B|}$ and $B \boxplus A$=$b_1b_2\cdots b_{|B|}a_1a_2\cdots a_{|A|}$. Let A[$l$] and $a_l$ represent the $l^{th}$ bit of $A$. $A[z:x]$ denotes $z^{th}$ to $x^{th}$ bits of $A$.

### A. The Top Structure of OFSR-PUF

As shown in Fig. 1(a), OFSR-PUF consists of five layers: the input layer, the ES layer, the AND gate layer, the nonlinear obfuscation layer, and the XOR gate layer. The input layer, as shown in Fig. 1(b), comprises $n$ D flip-flops, $n$+1 Inverter gates, 2($n$+1) AND gates, $n$+1 multiplexers and $n$+1 XOR gates to store and update challenges. Every four bits of a challenge *Q[1:n]* are fed into an ES cell to yield a 1-bit output $O$, shown in Fig. 1(c). Then, this bit is input to AND gate with $S[l]$ to generate a reliable bit $t_k[l]$,$l$={1,2,$\cdots$,$m$}, in which $S[l] = 0$ shields the unreliable ES bits, as Fig. 1(d) shows. Where-after, all $t_k[l]$ are inputted into the XOR gate layer to generate a feedback bit (FB), shown in Fig. 1(f). All FBs compose a muti-bit response *Re[1:K]* of this PUF with $K$ loops. In addition,the obfuscation layer in Fig. 1(e) isolates the challenge and response to avoid collapse response generation, according to $FB$ and $t_k$.

### B. Collapse Response

**Definition 1** (Guess space of response) Let $Re \in \{0,1\}^K$ be a response and contains $V \in \{0,1,2,\cdots,K\}$ secret bits, and then guess space of $Re$ is $2^V$ for an adversary, which is noted as $G(Re) = 2^V$.

**Definition 2** (Collapse response) Let $\hat{C} \in \{0,1\}^n$ and $\hat{Re} \in \{0,1\}^K$ be a challenge and response, respectively, if $\hat{Re} = puf(\hat{C})$ and $G(\hat{Re})<2^K$, and then $\hat{Re}$ is a collapse response.

Such two definitions imply leakage of partial bits results in the actual security of response relies on *unknown remainder terms (URT)*. To analyze the collapse response generated by the design based on LFSR, let us consider a situation without the nonlinear obfuscation layer, in which each output of AND gate is only fed into the XOR gate layer and all XOR gates in the first layer are removed. In this case, the structure is linear, and each feedback bit $FB_k$ is characterized by equation (1),

$$FB_k = \bigoplus_{l=1,2,\cdots,m} \delta_l(C[k+4(l-1):k+4l-1])\&S[l], \quad (1)$$

where $\delta_l$ denotes the $l^{th}$ ES cell; $C$[$k$+4($l$-1):$k$+4$l$-1] represents four inputted bits; $k$ marks the $k^{th}$ loop; $n$ denotes the length of the challenge; $k$ = {1,2,$\cdots$,$K$} counts the loop number; and $m$ is the number of the ES cells, which equals $n$/4. Moreover, the response $Re$ of the linear PUF is shown in the next expression,

$$Re = puf(C[1:n]) = \boxplus_{k=1,2,\cdots,K} FB_k. \quad (2)$$

Since $FB_k$ is assigned to challenge $C[n+k]$ per loop, the equation (2) can be written as the following version,

$$Re = \boxplus_{k=1,2,\cdots,K} C[n+k]. \quad (3)$$

Equation (3) illustrates that the response and challenge share the same bit sequence. Namely, an arbitrary $n$-bit subsequence

of CRP can be used as the challenge of its next bit(s) in the multi-bit response of LFSR-based PUFs. In this case, even $C[1:n]$ is only single-use (to resist reuse attacks) and still leakage at most $n$+$k$-1 sets of information of unused CRPs. It weakens the security of LFSR-based PUFs. A formal description is demonstrated by Theorem I.

**Theorem 1** Let $CRP^{adv}$ be a set of CRPs collected by an adversary, in which an arbitrary element $CRP_r^{adv}$ = $C_r \boxplus Re_r$. If a given challenge $C_u$ is subject to the next expression,

$$(\exists CRP_{\hat{r}}^{adv}[\hat{k}:n+\hat{k}-1]=C_u) \wedge (CRP_u \notin CRP^{adv}), \tag{4}$$

and then $puf(C_u)$ is a collapse response for the adversary.

**Proof**.

$$\begin{aligned} &C_u = CRP_{\hat{r}}^{adv}[\hat{k}:n+\hat{k}-1] \\ \Longrightarrow &puf(C_u) = puf(CRP_{\hat{r}}^{adv}[\hat{k}:n+\hat{k}-1]) \\ \Longrightarrow &puf(C_u) = CRP_{\hat{r}}^{adv}[n+\hat{k}:K] \boxplus URT \\ \Longrightarrow &G(puf(C_u)) = 2^{n+\hat{k}-1} < 2^K. \end{aligned}$$

### C. The Obfuscation Mechanism

As discussed above, the direct cause of collapse response is that challenge and response share the same bit sequence. In order to overcome this issue, a nonlinear obfuscation structure is devised as shown in Fig. 1(e), in which $t[1:m]$ are used to attain multiple $log_2 n$-bit index numbers (BIN) according to formula (5),

$$BIN_h^k = t_k[h:(h+log_2 n)\ mod\ m-1], \tag{5}$$

where $h$ = {1,2,$\cdots$,$H$} denotes the $h^{th}$ index number consisting of $log_2 n$ bits. Then, such $BIN_h^k$ are translated into an $n$-bit selection sequence (marked as $sel$ [1:$n$]) by the decoder as formula (6),

$$sel[i] = \begin{cases} 1, & i \in decBIN^k; \\ 0, & otherwise. \end{cases} \tag{6}$$

where $decBIN^k$ is the set of decimalism values of $BIN_h^k$. Consequently, selection and init bits choose $H$ bits in the challenge to be XORed with the feedback bit. This process is characterized as the next expression (7),

$$R[i] = \begin{cases} 0, & sel[i]=0 \text{ and } init[i]=0; \\ FB, & sel[i]=1 \text{ and } init[i]=0; \\ C[i], & init[i]=1. \end{cases} \tag{7}$$

When $init[i]$ = 0, the nonlinear obfuscation layer updates the selected bits of challenge according $sel$ [1:$n$], in which $sel[i]$ = 1 makes the output of the obfuscation layer $R[i]$ is assigned $FB$, and then is XORed with $C[i]$ in the input layer. After the bit obfuscation, all init bits are assigned ones. In this case, there are no any changes in the challenge, and then the input layer starts to execute the shift operation.

The obfuscation structure makes that there are nearly $H$/2 random bits of challenge to reverse with the aid of XOR operation in each loop. Simultaneously, the last bit of challenge is assigned the feedback bit. Hence, the bit-flipping ratio $r_{bf}$ is estimated by formula (8),

$$r_{bf} = \frac{H}{2(H+1)}. \tag{8}$$

That is, if value of $H$ is big enough, the hamming distance between $C$ [$n$+$k$:2$n$+$k$-1] and $Re$ [$k$:$n$+$k$-1] is approximate to 0.5 after several loops, which demonstrates the nonlinear obfuscation structure mixes and isolates challenge and response, while they once shared the same sequence.

### D. ES Selection Scheme

Reliability of OFSR-PUF depends on that of all ES cells. The test results on PUF chips fabricated in 14nm CMOS [7] shows the worst native bit error rate (BER) of ES cells is 14.5% at 0 $\sim$ 100°C and 750mV, while is reduced to 0.26% with the aid of a mask matrix that records the reliability information of each cells.

In this paper, the mask matrix is attained as the same way in [7], while a new masking scheme (named ESS) for the working stage is designed to shield the unreliable CRPs. Let $s_l$[1:16] present a row vector of the recorded matrix and covers the reliable information of the $l^{th}$ cell. $C$[$k$+4($l$-1):$k$+4$l$-1] where $l$={1,2,$\cdots$,$L$} selects one bit from each $s_l$ to form a vector $S$[1:$L$]. It is inputted into AND gate layer to guarantee that the values of $t_k[l]$ of unreliable CRPs are always stuck at zeros. That is, all unreliable CRPs are shielded. A prominent advantage is that it is not necessary to compute reliable challenges in server for remote PUF and each remote node shields unreliable CRPs locally, which saves amounts of computing resource for center node of IoT.

Although the ESS forces unreliable bits into zeros, it would not influence the uniformity, according to the following analysis based on Piling up Lemma.

**Piling up Lemma** let $X_i$ ($1 \le i \le n$) be an independent random variable whose values are 0 with probability $p_i$ and 1 with probability 1 - $p_i$. Then,

$$P[X_1 \oplus X_2 \oplus \cdots \oplus X_n = 0] = \frac{1}{2} + 2^{n-1}\prod_{i=1}^{n}(p_i - \frac{1}{2})^n.$$

Suppose that for a 16-cell OFSR-PUF, all ES outputs are uniform while the ESS forces 10% of outputs into 0s. Namely, $p_i$ increases to 0.55 from 0.5. In this case, $P[X_1 \oplus X_2 \oplus \cdots \oplus X_{16} = 0] = \frac{1}{2} + \frac{1}{2\times 10^{16}}$. It illustrates that our ESS only leads to an ignorable impact on uniformity.

## IV. Experiment

In this experiment, 1000 PUF instances are implemented on Cyclone IV FPGA to test our OFSR-PUF, in which each instance is characterized by 8 random LUTs. In the first loop, the init bit is set to 1 (but to 0 in others), and sel[1:n] is assigned 0s; each PUF instance is fed with challenges from a pre-generated random number table. Then, 14.5% of BER is recorded in the mask matrix to configure s[1:m] of AND gate layer. Moreover, 5-6 incorrect bits are randomly inserted into 2048 bits (generated by 256 instances) in each loop to simulate the 0.26% of BER. The remainder context of this section presents experimental results, including uniformity, uniqueness, reliability, randomness, resistance to machine learning attacks, and effect of the obfuscation mechanism.

TABLE I: NIST 800-22 randomness test.

| Test | P-value | Pass? | Test | P-value | Pass? |
|---|---|---|---|---|---|
| Frequency | 0.585 | Yes | Rank | 0.057 | Yes |
| BF | 0.041 | Yes | FFT | 0.312 | Yes |
| RandExVar | 0.350 | Yes | NOT | 0.689 | Yes |
| RandEx | 0.534 | Yes | Serial | 0.484 | Yes |
| CumSums | 0.876 | Yes | OT | 0.106 | Yes |
| Universal | 0.187 | Yes | LRO | 0.187 | Yes |
| ApprEntropy | 0.485 | Yes | Runs | 0.312 | Yes |
| linearCom | 0.311 | Yes | | | |

### A. Uniformity and Uniqueness

Uniformity is used to describe the distribution of 0 and 1 in responses. Its value range is [0, 1] and 0.5 is the ideal value. An inadequate uniformity means would result in that attackers guess the correct bit values in a high probability. Our OFSR-PUF achieves 0.4998 of uniformity, which is extremely approaching the ideal value.

Uniqueness is also a crucial metric for PUFs. A non-ideal uniqueness represents that different PUF instances generate identical or similar responses when receiving the same challenge. It leads to security risks during device authentication. Uniqueness is calculated as formula 9,

$$u = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=j+1}^{k} \frac{HD(R_i, R_j)}{n} \times 100\%, \qquad (9)$$

where $k$ marks the total amount of tested PUF instances, $n$ is the length of bits yielded from each instance, and HD($r_i$, $r_j$) represents the hamming distance between the $i^{th}$ and $j^{th}$ PUF instances. The range of $u$ is [0,1] and 0.5 denotes the optimal value. As shown in Fig. 2(a), the average HD of such 1000 OFSR-PUF instances is 0.5001. It illustrates our proposed PUF achieves a satisfactory uniqueness.

### B. Reliability

Reliability measures the consistency of PUF responses in various environments. Ideally, PUF always generate the same responses no matter when, where and how many times the input is presented. Reliability is calculated as formula (10),

$$Reliability = 1 - \frac{1}{k} \sum_{j=1}^{k} \frac{HD(R_i, R_{i,j})}{n} \times 100\%, \qquad (10)$$

where $n$ is the length of PUF response, $k$ marks the number of samples, and $HD(R_i, R_{i,j})$ is the Hamming distance between the responses $R_i$ and the $j^{th}$ sampling $R_{i,j}$. In this paper, four groups of BER are tested for 1-bit, 8-bit, 16-bit, and 32-bit output, respectively. Fig. 2(b) shows that the BER of OFSR-PUF is close to 0.26% for 1-bit response generation. Since a bit error induces following bits to be incorrect in a multi-bit response, the BER of OFSR-PUF reduces to 5.3% for the 8-bit response. However, 0.26% of BER is generated by cells whose output flips in a tiny probability, since the bit matrix shields markedly unreliable cells. Namely, the proportion of correct responses is much higher during repeated bit generation with the same challenge for such cells. Therefore, TMV effectively reduces the BER of the 8-bit response to 0.19%. Similarly, it improves the reliability to 99.2% and 96.7% for OFSR-PUF with 16-bit and 32-bit response, but once the data are only 90.8% and 83.0%, respectively.

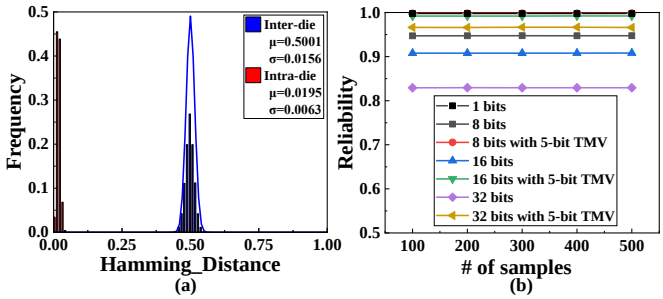


Fig. 2: (a) Inter-die and intra-die Hamming distances within 1000 dies; and (b) Reliability of 1-bit, 8-bit, 16-bit, and 32-bit outputs.

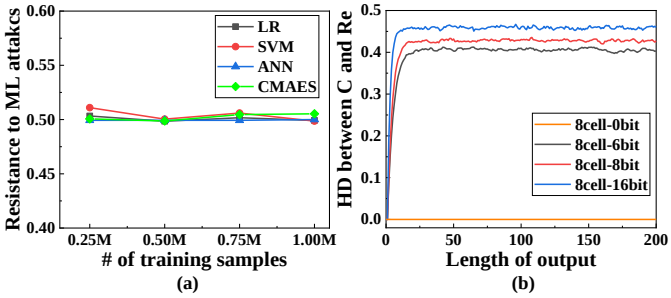


Fig. 3: (a) Ability against ML attacks, including LR, SVM, ANN, and CMAES; (b) Hamming distance between challenge and response for an 8-cell OFSR-PUF with 0/6/8/16 bits XORed in the challenge.

### C. Randomness

To evaluate the randomness of OFSR-PUF, an 800-22 NIST randomness test is conducted in 10M bits from such 1000 PUF instances. Table I exhibits our PUF design pass all sub-tests.

### D. Resistance to Machine Learning Attacks

Resistance to ML attacks [13, 14] is an important security criterion for PUFs. Linear regression (LR), support vector machine (SVM), covariance matrix adaptation evolutionary strategies (CMAES), and artificial neural network (ANN) are common models to test the ML-resistance. This paper conducted above models according to [14, 15]. Fig. 3(a) presents that the results from 0.25 $\sim$ 1M datasets. The precision of the four ML methods is always around the ideal value of 0.5, which are equivalent to random guess. And there is no evident growth trend. The results illustrate our proposal is able to resist such machine learning attacks.

### E. Effect of the Obfuscation Mechanism

To evaluate the obfuscation mechanism, the hamming distance between $C$ [$n$+$k$:2$n$+$k$-1] and $Re$ [$k$:$n$+$k$-1] is calculated with $n$ = 32 and $k$ = {1,2,$\cdots$,32}. Fig. 3(b) shows that the result of the obfuscation mechanism, including 4 cases: 0 bit, 6 bits, 8 bits, and 16 bits XORed in challenge. The experimental HDs are close to the theoretical values: 0.42, 0.44, and 0.47 are those of the 6-bit, 8-bit, and 16-bit obfuscations according to formula (8), respectively, while all HDs ever were zeros. Namely, the mechanism efficiently isolates the challenge and response from a shared sequence.

TABLE II: On chip resource usage of FPGA.

| ES stage | obfuscation stage | Logic cells | LUT-Only LCs | Register-Only LCs | LUT/Register LCs |
|---|---|---|---|---|---|
| 32 | 32 | 4727 | 4595 | 0 | 132 |
| 32 | 0 | 728 | 599 | 0 | 129 |
| 8 | 8 | 411 | 377 | 0 | 34 |
| 8 | 0 | 215 | 181 | 0 | 34 |

TABLE III: Comparison with the state-of-the-art PUFs.

| | Storage cost (bit) | Max BER (%) | Modeling precision | Area per bit |
|---|---|---|---|---|
| VLSI'17 [16] | $10^n$ | 11 | 0.94 | 1.45 $MF^2$ |
| DAC'20 [7] | 4096 | 0.26 | $\sim 0.5$ | 0.28 $MF^2$ 4600 LCs |
| TIFS'19 [9] | - | $\sim 0.1$ | $\sim 0.5$ | 419 LUTs + 264 FFs |
| VLSI'17 [17] | $10^n$ | 2.6 | $\sim 0.6$ | 2.64 $MF^2$ |
| 8-cell OFSR-PUF | 256 | $\sim 0.26$ | $\sim 0.5$ | 215 LCs |
| 32-cell OFSR-PUF | 1024 | $\sim 0.26$ | - | 728 LCs |

0.28 $MF^2$ is reported by [7] and 4600 LCs is an estimated value with the FPGA implementation, which is the bridge for the comparison.

### *F. Comparison with Prior-Art*

Table II records the hardware overhead of our design. The last line shows an ES cell needs ∼27 logic cells (LCs) in average, and each Sbox needs 48 LUTs [18]. Based on such data, DAC'20 [7] can be estimated as ∼4600 LCs for the 1-bit response generation. In contrast, a 32-cell OFSR PUF provides the same CRP space but only costs 728 LCs for the 1-bit case. The obfuscation with 4727 LCs is designed for the multi-bit case against the collapse response. Hence, our design is more efficient than [7] on the overhead per bit.

Finally, we compare our OFSR-PUF with the state-of-the-arts in table III. It shows that our proposed OFSR-PUF is outperformance on reliability and resistance to ML attacks with the smallest storage overhead for enrollment data. Moreover, our design has a comparable area overhead with [9] and is much more lightweight than other works. However, [9] requires the expensive overhead on secure memory and training time, as mentioned above. Another potential advantage of our design is that it occupies a much lower bandwidth versus such PUFs in multi-bit-based authentication protocols, because it generates a multi-bit response for each challenge.

## V. Conclusion

This paper proposes an OFSR-PUF employing configurable weak PUF based entropy source. It efficiently overcomes the collapse response issues, which also exist in conventional LFSR-based PUF designs and is first revealed in this paper. Moreover, compared with recently proposed strong PUFs, our design exhibits satisfactory performance, including 0.4998 of uniformity, 0.5001 of uniqueness, 0.26% of reliability, ideal randomness, and desired resistance to machine learning. Meanwhile, it reduces the storage overhead of center nodes and moves the computation load of challenge selection from server to remote devices, both of which make the centre device can connect more nodes in parallel.

## References


[1] I. Tsiokanos, J. Miskelly, C. Gu, M. O'neill, and G. Karakonstantis, "Dta-puf: Dynamic timing-aware physical unclonable function for resource-constrained devices," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 17, no. 3, pp. 1–24, 2021.

[2] J. Lee, D. Lim, B. Gassend, G. Suh, M. van Dijk, and S. Devadas, "A technique to build a secret key in integrated circuits for identification and authentication applications," in *2004 Symposium on VLSI Circuits. Digest of Technical Papers*, 2004, pp. 176–179.

[3] D. P. Sahoo, D. Mukhopadhyay, R. S. Chakraborty, and P. H. Nguyen, "A multiplexer-based arbiter puf composition with enhanced reliability and security," *IEEE Transactions on Computers*, vol. 67, no. 3, pp. 403–417, 2018.

[4] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *2007 44th ACM/IEEE Design Automation Conference*, 2007, pp. 9–14.

[5] D. E. Holcomb, "Power-up sram state as an identifying fingerprint and source of true random numbers," *IEEE Transactions on Computers*, vol. 58, 2009.

[6] J. Zhang and G. Qu, "Physical unclonable function-based key sharing via machine learning for iot security," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 8, pp. 7025–7033, 2020.

[7] V. B. Suresh, R. Kumar, and S. Mathew, "Invited: A 0.26% ber, machine-learning resistant 1028 challenge-response puf in 14nm cmos featuring stability-aware adversarial challenge selection," in *2020 57th ACM/IEEE Design Automation Conference (DAC)*, 2020, pp. 1–3.

[8] M. Rostami, M. Majzoobi, F. Koushanfar, D. S. Wallach, and S. Devadas, "Robust and reverse-engineering resilient puf authentication and key-exchange by substring matching," *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 1, pp. 37–49, 2014.

[9] S. S. Zalivaka, A. A. Ivaniuk, and C.-H. Chang, "Reliable and modeling attack resistant authentication of arbiter puf in fpga implementation with trinary quadruple response," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 4, pp. 1109–1123, 2019.

[10] Y. Hori, H. Kang, T. Katashita, and A. Satoh, "Pseudo-lfsr puf: A compact, efficient and reliable physical unclonable function," in *2011 International Conference on Reconfigurable Computing and FPGAs*, 2011, pp. 223–228.

[11] T. Zhou, Y. Ji, M. Chen, and Y. Li, "Pl-mro puf: High speed pseudo-lfsr puf based on multiple ring oscillators," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2020, pp. 1–5.

[12] S. Hou, Y. Guo, and S. Li, "A lightweight lfsr-based strong physical unclonable function design on fpga," *IEEE Access*, vol. 7, pp. 64 778–64 787, 2019.

[13] J. Shi, Y. Lu, and J. Zhang, "Approximation attacks on strong pufs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 10, pp. 2138–2151, 2020.

[14] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, and J. Schmidhuber, "Modeling attacks on physical unclonable functions," in *Acm Conference on Computer Communications Security*, 2010.

[15] J. Zhang, C. Shen, Z. Guo, Q. Wu, and W. Chang, "Ct puf: Configurable tristate puf against machine learning attacks for iot security," *IEEE Internet of Things Journal*, pp. 1–1, 2021.

[16] S. Jeloka, K. Yang, M. Orshansky, D. Sylvester, and D. Blaauw, "A sequence dependent challenge-response puf using 28nm sram 6t bit cell," in *2017 Symposium on VLSI Circuits*, 2017, pp. C270–C271.

[17] X. Xi, H. Zhuang, N. Sun, and M. Orshansky, "Strong subthreshold current array puf with 265 challenge-response pairs resilient to machine learning attacks in 130nm cmos," in *2017 Symposium on VLSI Circuits*, 2017, pp. C268–C269.

[18] D. Canright, "A very compact s-box for aes," in *International Workshop on Cryptographic Hardware and Embedded Systems*, 2005.