Multi-valued Arbiters for Quality Enhancement of PUF Responses on FPGA Implementation

(Invited Paper)

Siarhei S. Zalivaka[1], Alexander V. Puchkov[2], Vladimir P. Klybik[2], Alexander A. Ivaniuk[2], and Chip-Hong Chang[1]
[1]School of Electrical and Electronic Engineering, Nanyang Technological University, 639798 Singapore
[2]Belarusian State University of Informatics and Radioelectronics, P. Browki Str., 6 Minsk 220013, Belarus
e-mail: zali0001@e.ntu.edu.sg, alexander.v.puchkov@gmail.com, {klybik, ivanuik}@bsuir.by, ECHChang@ntu.edu.sg

Abstract— One main problem encountered in the FPGA implementation of Arbiter based Physical Unclonable Function (A-PUF) is the response instability caused by the metastability of delay flip-flop. This paper presents a new multi-arbiter approach to extract more entropy to extend the number of response bits to a single challenge. New multi-arbiter schemes based on the insertion of either a four-flip-flop arbiter or SR latch arbiter after each pair of multiplexers in the configurable paths are proposed to detect the metastable state when two copies of test pulse arrive at the arbiter inputs almost simultaneously. The detected metastable states are distinguishable by the encoded multiple valued outputs of the arbiter. The codes corresponding to the metastable states collectively form a deterministic ternary state that can be recoded to one of the stable states to improve the uniqueness and reliability of the PUF. Our analysis shows that the proposed design can generate robust and reliable challenge-response pairs with a uniqueness of 0.4982 and a reliability of 0.9985 at the expense of a relatively small FPGA resource overhead.

I. Introduction

The increase in random and unpredictable variability of electrical parameters due to the decreasing device feature size in nano-scale integrated circuit (IC) manufacturing technology has given rise to its exploitation for the development of different types of silicon-based Physical Unclonable Function (PUF) as emerging cryptographic primitive for device authentication and electronic tagging. A PUF can be represented by the mapping of an external input challenge C to an output response R. The response R to the challenge C is random but unique for an IC that contains the same PUF circuit due to their mismatches in the electrical parameters caused by the manufacturing process variations. The challenge-response pairs (CRPs) that uniquely describe an IC are prohibitively difficult to be reproduced by a physical clone of the same circuit. The mismatches in physical parameters that can be used for PUF implementation include the reset state of memory cell (e.g., SRAM-PUF [1], Butterfly PUF [2] and SR latch PUF [3]), signal propagation delays (e.g., Arbiter PUF [4]), oscillation frequency [5], aging effect [6], etc..

One main issue associated with the use of PUF as a cryptographic key or unique chip identifier in an authentication protocol is the minute mismatches of these exploited electrical characteristics within an IC are often instable and sensitive to temporal operating condition variations. This causes the responses to the same challenges to change from time to time. This problem is especially severe for delay-based PUF with a large CRP space such as the arbiter PUF (A-PUF). To overcome this imperfection, error correction codes (ECC) and post processing can be applied to reduce the error rate of PUF response (e.g. [7]). Alternatively, existing structure of the PUF could be modified to stabilize the response by using devices with opposite parametric sensitivity to counteract the changes in certain operating condition such as [8]. If the native PUF quality is poor, the reliance on strong ECC to achieve high reliability is discouraged due to its high hardware overheads.

In this paper, we propose two main techniques to overcome the poor reliability of native A-PUF by replacing the DFF by enhanced 4-DFF and SR latch based arbiters in a multi-arbiter PUF (MA-PUF) scheme. The proposed techniques are able to detect and encode the metastable outputs of the PUF, which allows them to be isolated or recoded into the stable logic zero or one to improve the randomness, uniqueness and stability of the response. In addition, the flexibility of selecting different arbiters of the MA-PUF chain as outputs also expands the CRP space and enables better trade-offs to address different security requirements.

The rest of this paper is organized as follows. The classical arbiter PUF architecture is introduced in Section II. Multi-arbiter PUF scheme and its quality analysis are presented in Section III. Two different multi-valued arbiters for the detection and encoding of metastable bits of MA-PUF are proposed in Section IV. Experimental results are analyzed and discussed in Section V. Different ways to generate unique identifiers are given in Section VI. Finally, Section VII concludes the paper.

II. Classical Arbiter PUF Architecture

The Arbiter PUF was coined in [4]. It is classified as a delay-based PUF as the response is generated based on the intrinsic timing differences of two topologically and functionally identical paths in an IC due to its intra-die variations. As shown in Fig. 1, a set of symmetrical paths can be designed by N multiplexer stages. Each stage L_n consists of a pair of multiplexers configured as a cross-bar switch. The data select lines of the two multiplexers in stage L_n are both fed by Ch_n, where Ch_n is the n-th bit of an N-bit challenge. Altogether 2^N different symmetrical paths can be selected by an N-bit challenge.

978-1-4673-9569-4/16/$31.00 ©2016 IEEE

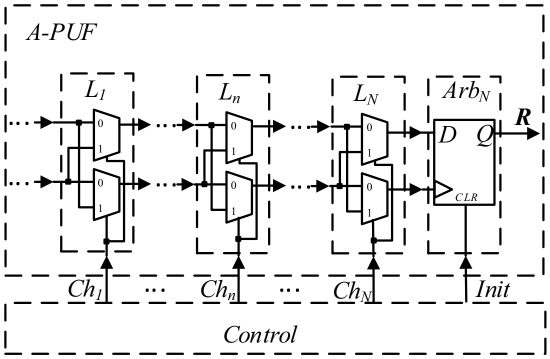


Fig. 1. Classical arbiter PUF circuit.

The output signals of the two multiplexers at the last stage are connected respectively to the data and clock inputs of a DFF (Arb_N). The DFF acts as an arbiter to determine which of the signals along the two symmetrical paths is faster. When the output signal in the upper multiplexer is faster, the arbiter outputs a logic 1; otherwise, it outputs a logic 0.

The main disadvantage of classical A-PUF is the relatively low reliability [9] due to the metastability of DFF. Our experiments have shown that the average and minimum reliability of an N = 128 stage classical A-PUF are 0.5769 and 0.5648, respectively. Due to the routing constraints, it is especially hard to design symmetric electrical paths on a FPGA platform [10]. The timing mismatches in asymmetrically designed paths may outweigh the mismatches due to the unpredictable process variations, resulting in low unpredictability.

III. Multi-arbiter PUF Architecture

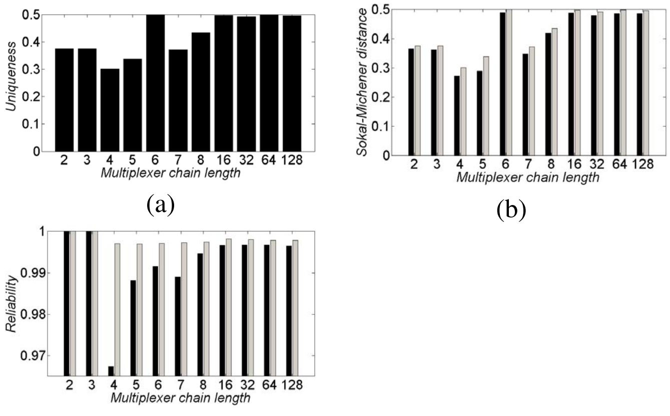
To enhance the reliability and unpredictability of classical A-PUF, a multi-arbiter PUF (MA-PUF) is proposed in [11], where each pair of multiplexers is connected to an arbiter as in the last stage of classical A-PUF, as shown in $MA\text{-}PUF_1$ of Fig. 2. Multiple response bits to an input challenge can be extracted from different stages of the multiplexer chain in any order by changing the address Adr to the end multiplexer.

Instead of generating multiple response bits from one A-PUF chain, one response bit can also be selected from each MA-PUF to generate a multi-bit response vector from several MA-PUF chains, as shown in Fig. 2. All MA-PUF chains share a common pseudorandom N-bit $Challenge$ (Ch_1, Ch_2, ..., Ch_N) generated by a linear feedback shift register (LFSR). A test pulse generator (TPG) produces the common input signal S to the first stage of each MA-PUF chain to launch the propagation delay race on the rising edge of S. The response vector to each challenge is stored in the output register REG.

A multi-chain MA-PUF circuit was implemented with D = 8 MA-PUF chains ($MA\text{-}PUF_d$ $\forall d \in [1, 8]$). Each chain contains N = 128 cross-bar arbiters. To evaluate the uniqueness and reliability of this multi-chain MA-PUF scheme as the number of stages n in each chain changes, this PUF was tested E = 30 times on C = 10,000 pseudorandom challenges. The parameters involved in this experiment is summarized in Table I.

TABLE I
Parameters used for multi-chain MA-PUF experiments

Parameter	Definition
N	The maximum length of challenge vector, which is also the maximum number of cross-bar connected multiplexer stages in each MA-PUF chain.
n	The index of stage length, n = 1, . . ., N
C	The number of applied challenge vectors, $C \leq 2^N$. Each challenge vector will produce one response vector.
c	The index of a challenge vector, which is also the index of its corresponding response vector, c = 1, . . ., C
D	The number of MA-PUF chains on an FPGA, which is also the length of the response vector.
d	The index of a MA-PUF chain, d = 1, . . ., D
F	The number of tested FPGAs
f	The index of an FPGA
E	The number of tests. Each test involves the application of the same set of C challenge vectors to all multi-chain MA-PUF implemented on F FPGAs.
e	The index of a test, e = 1, . . ., E
$R_{e,f,d,n}(c)$	The response vector to the applied challenge vector c generated by multiplexer stage length n in chain d of the multi-chain MA-PUF implemented on FPGA f $R_{e,f,d,n}(c) \in \{0, 1\}$ for classical A-PUF circuit, $R_{e,f,d,n}(c) \in \{0, 1\}^d$ for multi-chain MA-PUF)



(a)

(b)

(c)

Fig. 3. Multi-chain MA-PUF characteristics versus multiplexer chain length: (a) Uniqueness, (b) Sokal-Michener distance (average and minimum), (c) Reliability (average and minimum)

Our experiments show that the quality of MA-PUF depends on the multiplexer chain length n. The uniqueness calculated based on Hamming distance and the dissimilarity calculated based on Sokal-Michener distance [12] are plotted against n in Figs. 3(a) and 3(b), respectively. For multiplexer chain of less than 16 stages, there are clear signs of ill metastability effects that exhibit no specific trend of variation. For n greater than 16, the uniqueness and Sokal-Michener distance approximate the ideal value of 0.5 and vary within a narrow range of [0.491, 0.498]. The same tendency and narrow spread are observed for the reliability plotted in Fig. 3(c) after the 16-th cross-bar arbiter stage. The multi-chain MA-PUF provides stable response values with an average reliability of 0.9982 and a minimum reliability of 0.9965. Such unique, random and stable response values can only be generated by classi-

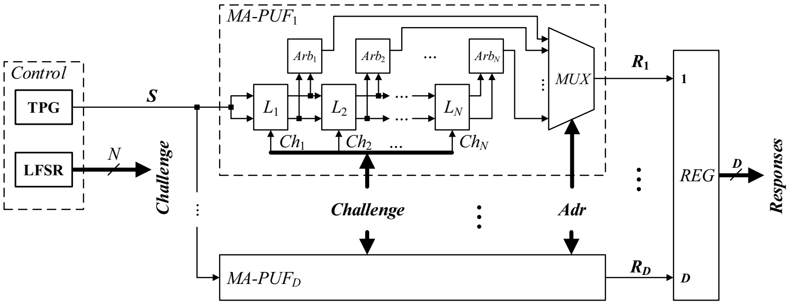


Fig. 2. Multi-chain Multi-arbiter PUF Circuit.

cal FPGA-based A-PUF with ECC or manual routing. Thus multi-chain MA-PUF enables an FPGA chip to be uniquely identified with a great flexibility in CRP generation by changing the chain length n. The MA-PUF response values can be chosen from among many stable and uncorrelated arbiter responses from each chain by changing the address value Adr of Fig. 2, making it useful as a reconfigurable strong PUF. In what follows, methods to identify and resolve the metastable outputs of arbiters are proposed.

IV. Identification of Metastable Arbiter Bits

A. 4-DFF based arbiter

Although the CRP space has been enlarged by MA-PUF, the metastable operation of arbiter affects the reliability of MA-PUF responses, especially when the length n of the multiplexer chain is short. The metastable states of MA-PUF can be detected by substituting the DFF arbiters by multi-valued arbiters as shown in Fig. 4. The connections of the two racing signal pulses to adjacent DFFs alternate between the D and clock inputs. The outputs, $R^1_{d,n}$ and $R^0_{d,n}$, of the first two FFs are triggered by the rising edges of the signals in the upper and lower multiplexer paths, respectively. The outputs, $R^3_{d,n}$ and $R^2_{d,n}$ of the last two FFs are triggered by the falling edges of the signals in the upper and lower multiplexer paths, respectively. Each of these output bits is asserted high at their respective active clock edge when the launching signal in the upper path is faster and asserted low if it is slower. The distribution of different quaternary logic outputs collected from a single arbiter in E = 30 tests of 10,000 CRPs is shown in Fig. 5(a). Only the values ”0110” and ”1001” occur with approximately equal high probability and can be taken as the stable one and zero bits, respectively. Due to the routing restriction, the timing difference between two long signal paths may be negligibly small that causes the absence of some response values like ”0000”, ”0100”, ”1000”, ”1100”, ”1101”, ”1110” and ”1111”. The rest of the values that rarely occur are due to the metastable outputs of the arbiter. To improve the reliability and unpredictability of MA-PUF, these codes can be converted to ”0110” as its probability of occurrence is less than that of ”1001” (see Fig. 5(a)).

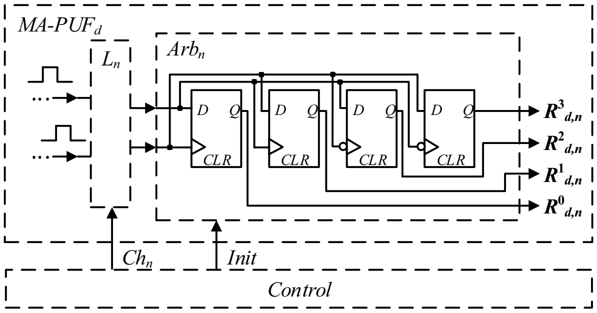


Fig. 4. MA-PUF with enhanced arbiter.

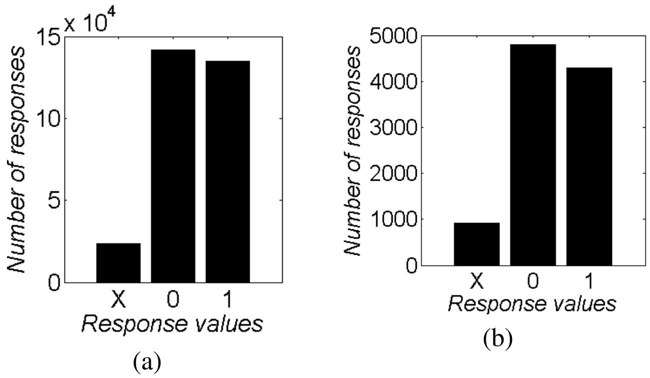


Fig. 5. Distribution of 0, 1 and X outputs of MA-PUF with (a) 4-DFF based arbiter or (b) SR latch based arbiter.

B. SR latch based arbiter

Alternatively, the metastable oscillatory outputs of an arbiter can be detected and isolated by using a SR latch in place of a DFF, as shown in Fig. 6. The proposed SR latch arbiter consists of two cross-coupled NOR gates and two DFFs. Unlike the classical A-PUF, the proposed SR latch based arbiter is triggered by the falling edge of the test pulse. This is because metastable operation occurs when both S and R inputs transit from logic one to logic zero simultaneously and stay at logic zero for a long time. A frequency counter with at least two DFFs is required to capture the oscillatory metastable operation of SR latch. Three output states of SR latch are possible: stable logic zero, stable logic one and high frequency oscillatory (HFO) states [13]. These output states are encoded by a 2-bit binary counter. Stable zero state is encoded as $(R^0_{d,n} = 0, R^1_{d,n} = 0)$, stable one state as $(R^0_{d,n} = 1, R^1_{d,n} = 0)$ and HFO state as $(R^0_{d,n} = 1, R^1_{d,n} = 1)$. Thus, the output of the lower DFF

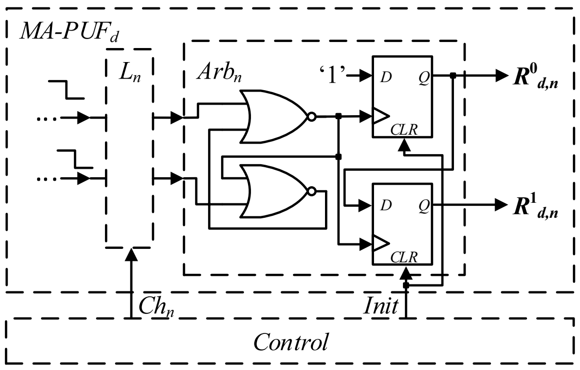


Fig. 6. SR latch based arbiter.

($R^1_{d,n}$) indicates if an arbiter response is stable ($R^1_{d,n} = 0$) or metastable ($R^1_{d,n} = 1$).

The distribution of the ternary response values {stable zero, stable one, HFO} is shown in Fig. 5(b). Around 10% of the response values are detected to be the unstable HFO. The ternary code ($R^0_{d,n} = 1, R^1_{d,n} = 1$) corresponding to HFO can be arbitrarily but deterministically converted to either ($R^0_{d,n} = 0, R^1_{d,n} = 0$) or ($R^0_{d,n} = 1, R^1_{d,n} = 0$) to balance the probabilities of stable zero and stable one states (all metastable values will be converted to stable zero values when the number of occurrences of stable zero is less than stable one and vice versa). It is also observed that the oscillation frequency before the metastable operation settles to its final stable value changes with the input challenge to the MA-PUF. A more accurate characterization of the HFO states for different input challenges can be accomplished by increasing the length of the frequency counter.

V. Experiment Results

The same single-chain MA-PUF designs with 4-DFF and SR latch arbiters were coded in VHDL, synthesized by CAD Xilinx ISE 14.7 and implemented on 10 Digilent Nexys-4 Artix-7 FPGA boards. Transfer of challenges and responses between the FPGA boards and personal computer was made via the UART interface. Every PUF instance was tested in E = 30 tests with C = 10,000 challenges generated by a 128-bit LFSR counter. It takes around 25 minutes to collect 10,000 CRPs from this system. The uniqueness, reliability and randomness for both enhanced MA-PUFs are then evaluated by considering both cases of keeping the codes of multi-valued arbiters corresponding to the metastable states as a ternary state (which means that X is equally probable to be recoded as a 1 or a 0 state) as well as converting them all to 1 state. The resource overheads of MA-PUF and enhanced MA-PUFs are also compared with the classical A-PUF implementation on the same FPGA.

A. Uniqueness

The uniqueness quantifies the dissimilarity of the responses generated by different PUF-instances to the same challenge. Average inter-die Hamming Distance (HD) can be used to estimate the uniqueness. Let R_u and R_v be the n-bit responses generated from two different PUF-instances u and v respectively to the same challenge c. The uniqueness U for m PUF-instances can be computed by [14]:

$$U = \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} \frac{HD(R_u, R_v)}{n} \quad (1)$$

TABLE II
Hamming distances between 0, 1 and X

Values	**0**	**1**	$\boldsymbol{X}$
0	0	1	0.5
1	1	0	0.5
$\boldsymbol{X}$	0.5	0.5	0

TABLE III
Different combinations of ternary values for Sokal-Michener distance calculation

Variable	$v_1(i)v_2(i)$
a	11
b	01
c	10
d	00
e	XX
f	$0X$
g	$X0$
h	$1X$
i	$X1$

Ideally, U = 0.5, which is the maximum percentage of bit differences that can be obtained between both the true and complement forms of two binary vectors. The average uniqueness for 10,000 CRPs obtained from 10 MA-PUFs is 0.4972 with 4-DFF arbiter and 0.4982 with SR latch arbiter, which is very close to the ideal value.

If the codes corresponding to the metastable states identified by the 4-DFF arbiter and the HFO state of the SR latch based MA-PUF can be arbitrarily assigned to either a zero or a one state, then they can be represented by a ternary state X. The Hamming distance between two ternary vectors with each element belongs to the set {0, 1, X} is redefined according to Table II, where the Hamming distances between X and 0, and between X and 1 are both 0.5.

The average uniqueness computed with the modified HD for 10 PUF instances with 10,000 CRPs is 0.4959 for the 4-DFF arbiter based PUF and 0.4929 for the SR latch arbiter based PUF, which are also very close to the ideal value.

To estimate the uniqueness of generated unique identifier obtained by the concatenated responses for 10,000 challenges, the modified Sokal-Michener [12] distance between two ternary vectors is used. The variables that denote the nine possible combinations of every possible pairs of ternary values obtained at the same position i of two ternary vectors, v_1 and v_2, are listed in Table III.

To account for the fact that state X can be recoded into either state 0 or state 1, the Sokal-Michener distance is modified in a similar way to:

$$D_{Sokal-Michener} = \frac{b+c}{m} + \frac{0.5\cdot(f+g+h+i)}{m} = \tag{2}$$
$$= \frac{2\cdot(b+c)+f+g+h+i}{2\cdot m}$$

If the state X for the multi-valued arbiters is recorded into the stable 1 state, the similarity of two binary vectors v_1 and v_2 of length m can be expressed as:

$$S_{Sokal-Michener} = \frac{a+d}{m} \tag{3}$$

The distance between two binary vectors can then be computed as $1 - S_{Sokal-Michener}$.

For MA-PUF with 4-DFF based arbiter, the average and minimum Sokal-Michener distances are 0.4971 and 0.4868, respectively if all Xs are recoded as 1s, and 0.4954 and 0.4867, respectively if all Xs are preserved. On the other hand, for the MA-PUF with SR latch based arbiter, the modified average and minimum Sokal-Michener distances are 0.4983 and 0.4888, respectively if X is recoded as 1, and 0.4929 and 0.4854, respectively if X is preserved.

B. Reliability

The reliability of a PUF measures the temporal reproducibility or stability of the responses to the same input challenge. The reliability can be quantified using the Bit Error Rate (BER), which is the average percentage of erroneous response bits obtained at different periods of time or operating environments. Let R_i be a reference response vector of size n to some challenge C produced by a PUF instance. Each element of R_i can be computed by a majority voting algorithm as follows:

$$R_i = \frac{max(n_0, n_1, n_X)}{n_0 + n_1 + n_X} \tag{4}$$

where n_0, n_1 and n_X are the numbers of zero, one and metastable states, respectively in E tests, and $n_0 + n_1 + n_X = E$.

Let $R_{i,e}$ be the response to the same challenge generated by the same PUF at different time $e = 1, 2, \ldots, E$. The reliability S can be computed by [14]:

$$S = 1 - BER = 1 - \frac{1}{E}\sum_{e=1}^{E}\frac{HD(R_i, R_{i,e})}{n} \tag{5}$$

10,000 CRPs are collected in 30 experiments from the MA-PUF implemented on 10 FPGA boards. The average and minimum S obtained for MA-PUF with 4-DFF based arbiter are 0.9986 and 0.9979, respectively by recoding X to 1, and 0.9985 and 0.9978, respectively by retaining X. The proposed MA-PUF with SR based arbiter has very similar high average and minimum reliability S of 0.9985 and 0.9978, respectively with recorded X, and 0.9975 and 0.9965, respectively with X retained.

TABLE IV
NIST TEST RESULTS

Test Description	Passed/Total		P-value	
	4-DFF	SR	4-DFF	SR
Frequency	100/100	100/100	0.74	0.53
Frequency block	100/100	100/100	0.12	0.53
Runs	100/100	100/100	0.74	0.12
Longest run	100/100	100/100	0.53	0.99
Binary matrix rank	100/100	100/100	0.35	0.91
Spectral (DFT)	100/100	100/100	0.53	0.21
Non-overlapping	97/100	98/100	0.73	0.76
Overlapping	100/100	100/100	0.74	0.91
"Universal"	100/100	100/100	0.07	0.02
Serial	100/100	100/100	0.21	0.74
Cumulative sums	100/100	100/100	0.12	0.07
Random exc.	10/10	10/10	—	—
Random exc. var.	10/10	10/10	—	—

TABLE V
HARDWARE OVERHEAD COMPARISON

Component	# slice LUTs	# slice registers
A-PUF	256 / 63400	1 / 126800
MA-PUF	298 / 63400	128 / 126800
4 DFF	302 / 63400	512 / 126800
SR latch	506 / 63400	256 / 126800
Entire system	2494 / 63400	1263 / 126800

C. Randomness

The randomness of a sequence of numbers can be assessed by the statistical test package NIST [15]. This package contains 13 statistical tests. If the sequence passes every test from the package, it can be considered as a true random number sequence. Ten million response bits were generated and split into 100 blocks of 100,000 bits each. For MA-PUF with SR latch based arbiters, the X-bits were arbitrarily substituted by zero or one bits by a pseudorandom number generator. The results of NIST test are shown in Table IV.

The outputs of both MA-PUFs have successfully passed all NIST tests and hence, their responses can be considered as a true random number sequence.

D. Hardware overhead analysis

The hardware overheads for different types of single-chain arbiter PUF implementation are shown in Table V. Despite large number of registers used compared to the classical A-PUF, these additional resources are negligible compared with the expanded CRP space and their enhanced reliability and uniqueness for highly robust cryptographic key generation. The entire MA-PUF system includes the UART controller, 128-bit LFSR counter, PUF storage registers and the PUF itself. The complete MA-PUF system consumes only less than 4% of the total logic slices and less than 1% of registers available in the target FPGA chip. In fact, the additional arbiters of 4-DFF and SR latch based MA-PUF over the basic A-PUF consume less than 0.08% and 0.4% of logic slices and less than 0.4% and 0.2% of registers, respectively, of the total resources available on the FPGA.

By sacrificing a relatively small part of the total FPGA hardware resources, MA-PUFs provides greater flexibility to use

TABLE VI
CORRELATION BETWEEN ARBITER OUTPUTS

Arb_i	121	122	123	124	125	126	127	128
121	■	N	N	N	N	N	S	N
122	N	■	N	N	S	N	N	N
123	N	N	■	N	S	N	S	N
124	N	N	N	■	S	N	N	N
125	N	S	S	S	■	N	S	N
126	N	N	N	N	N	■	N	N
127	S	N	S	N	N	N	■	N
128	N	N	N	N	N	N	N	■

any arbiter outputs from the 16th to 128th multiplexer stages as entropy sources to generate better quality multi-bit response to a given challenge.

VI. GENERATION OF UNIQUE CHIP IDENTIFIER BY MA-PUF

The proposed MA-PUF systems can be used for unique chip identification and strong authentication. There are three possible ways for the generation of chip identifiers:

1. The initial challenge is used as a seed of the LFSR, which generates the select signals to the multiplexers and Adr to select arbiters for the response bit generation. The LFSR generates additional k Adr words from the input challenge to produce a k-bit identifier.

2. The applied challenge is used directly to configure the signal paths. Specific arbiters in the multiplexer chain are chosen at design time. The arbiters can be selected based on inter-arbiter uniqueness, high reliability and better randomness after the characterization the MA-PUF. The resources occupied by the unselected arbiters can then be freed.

3. Similar to the second method except that part of the challenge is used for arbiter selection. This approach is more secure, but requires a longer challenge.

Table VI shows the results of the hypothesis test for correlation of eight adjacent arbiters in a single chain of MA-PUF. The correlation between the two arbiters with the row and column indexes is marked S for statistically significant and N for statistically insignificant. The black colored cells represent self-correlation of the same arbiter, which is always 1. The results show that uncorrelated adjacent arbiters are rare.

The arbiters should preferably be selected so that the minimum chain length is 16. In addition, one should also avoid addressing adjacent correlated arbiters in sequence to generate a multi-bit response. The distance between sequentially addressed arbiters should preferably be random and large.

VII. CONCLUSIONS

This paper presents and analyzes two techniques to identify and exploit metastable states to improve the uniqueness and reliability of multi-arbiter PUF design. The proposed 4-DFF and SR latch based MA-PUF designs have been implemented on FPGA platform (Digilent Nexys-4 Artix-7 FPGA board). Our experimental results show that these MA-PUFs possess very high-quality uniqueness, reliability and randomness compared with the classical arbiter PUF. The quality of these MA-PUFs has insignificant fluctuation (around 0.001) if the first 16 arbiters are excluded. With a small hardware overhead, these MA-PUFs greatly expand the CRP space for strong PUF applications and provide additional agility in generating unique chip identifiers for FPGA device authentication and integrity protection.

ACKNOWLEDGEMENTS

This work was supported by the Singapore Ministry of Education Academic Research Fund Tier I Grant No. MOE 2014-T1-002-141.

REFERENCES

[1] D. E. Holcomb, W. P. Burleson, and K. Fu, "Initial SRAM state as a fingerprint and source of true random numbers for RFID tags," in *Proc. Conf. RFID Security*, vol. 7, Malaga, Spain, Jul. 2007, p. 1-2.

[2] S. S. Kumar, J. Guajardo, R. Maes, G. J. Schrijen, and P. Tuyls, "The butterfly PUF protecting IP on every FPGA," in *Proc. IEEE Int. Workshop Hardware-Oriented Security and Trust (HOST'08)* , Anaheim, USA, Jun. 2008, pp. 67–70.

[3] D. Yamamoto et al. , "Uniqueness enhancement of PUF responses based on the locations of random outputting RS latches," in *Proc. Int. Workshop Cryptographic Hardware and Embedded Syst. (CHES'11)*, Nara, Japan, Sep. 2011, pp. 390–406.

[4] J. Lee et al., "A technique to build a secret key in integrated circuits for identification and authentication applications," in *Proc. Int. Symp. VLSI Cir. (VLSI'04)*, Honolulu, USA, Jun. 2004, pp. 176–179.

[5] B. Gassend, D. Clarke, M. Van Dijk, and S. Devadas, "Silicon physical random functions," in *Proc. ACM Conf. Comput. And Comms. Security*, Washington DC, USA, Nov. 2002, pp. 148–160.

[6] S. Meguerdichian and M. Potkonjak, "Device aging-based physically unclonable functions," in *Proc. IEEE Design Automation Conf. (DAC'11)*, San Diego, USA, Nov. 2011, pp. 288–289.

[7] R. Maes, A. Van Herrewege, and I. Verbauwhede, "PUFKY: A fully functional PUF-based cryptographic key generator," in *Proc. Int. Workshop Cryptographic Hardware and Embedded Syst. (CHES'12)*, Leuven, Belgium, Sep. 2012, pp. 302–319.

[8] Y. Cao, L. Zhang, C. H. Chang, and S. Chen, "A low-power hybrid RO PUF with improved thermal stability for lightweight applications," *IEEE Trans. on CAD*, vol. 34, no. 7, pp. 1143–1147, Jul. 2015.

[9] M. Majzoobi, F. Koushanfar, and S. Devadas, "FPGA PUF using programmable delay lines," in *Proc. IEEE Int. Workshop Info. Forensic Security (WIFS'10)*, Seattle, USA, Dec. 2010, pp. 1–6.

[10] S. Morozov, A. Maiti, and P. Schaumont, "An analysis of delay based PUF implementations on FPGA," in *Proc. Int. Symp. Applied Reconfigurable Computing (ARC'10)*, Bangkok, Thailand, Mar. 2010, pp. 382–387.

[11] V. P. Klybik and A. A. Ivaniuk, "Use of arbiter physical unclonable function to solve identification problem of digital devices," *Auto. Contr. and Comput. Sci.* , vol. 49, no. 3, pp. 139–147, 2015.

[12] R. R. Sokal and C. D. Michener, "A statistical method for evaluating systematic relationships," *The Univ. of Kansas Sci. Bullutin*, vol. 38, no. 22, pp. 1409–1438, 1958.

[13] T. Kacprzak, "Analysis of oscillatory metastable operation of an RS flip-flop," *IEEE J. Solid State Cir.*, vol. 23, no. 1, pp. 260–266, 1988.

[14] Y. Hori, T. Yoshida, T. Katashita, and A. Satoh, "Quantitative and statistical performance evaluation of arbiter physical unclonable functions on FPGAs," in *Proc. Int. Conf. Reconfigurable Computing and FPGAs (ReConFig'10)*, Quantana Roo, Mexico, Dec. 2010, pp. 298–303.

[15] A. Rukhin et al., "A statistical test suite for random and pseudorandom number generators for cryptographic applications," *NIST Special Publication*, 2010.