INTERNATIONAL JOURNAL OF RESEARCH
In Advanced Engineering Technologies

Volume 4, Issue 1 APR 2015

DESIGN AND IMPLEMENTATION OF PHYSICAL UNCLONABLE FUNCTIONS FOR DEVICE AUTHENTICATION AND SECRET KEY GENERATION USING ASIC

[1] **B. REKHA RANI**
PG Scholar in VLSI System Design,
[2] **S. CHAKRI SREEDHA**,
Asst. Professor, ECE Department,
[1] rekharani410@gmail.com,
[2]chakri.sreedhar@gmail.com.

ABSTRACT:

Silicon physical unclonable functions (PUF) utilize the variation during silicon fabrication process to extract information that will be unique for each chip. There have been many recent approaches to how PUF can be used to improve security related applications. However, it is well-known that the fabrication variation has very strong spatial correlation and this has been pointed out as a security threat to silicon PUF. In fact, when we apply NIST's statistical test suite for randomness against the random sequences generated from a population of 125 ring oscillator (RO) PUFs using classic 1-out-of-8 Coding and Neighbor Coding, none of them can pass all the tests. In this paper, we propose to decouple the unwanted systematic variation from the desired random variation through a regression-based distiller, where the basic idea is to build a model for the systematic variation so we can generate the random sequences only from the true random variation. Applying Neighbor Coding to the same benchmark data [2], our experiment shows that 2nd and 3rd order polynomials distill random sequences that pass all the NIST randomness tests. So does 4th order polynomial in the case of 1-out-of-8 Coding. Finally, we introduce two generic random sequence generation methods. The sequences they generate fail all the randomness tests, but with the help of our proposed polynomial distiller, all but one tests are passed. These results demonstrate that our method can provide statistically random PUF information and thus bolster the security characteristics of existing PUF schemes.

Keywords: ring oscillator (RO), physical unclonable functions (PUFs), linear regression, variation decomposition

I. INTRODUCTION

One of the most renowned principles for the design of a cryptosystem is Kerckhoff's law: "A cryptosystem should be secure even if everything about the system, except the key, is public knowledge (1883)." In order to provide a secure storage for cryptographic keys, contemporary tamper-resistant devices such as smart cards arm themselves with a number of countermeasures to defeat various kinds of invasive, semiinvasive and non-invasive physical attacks. Nevertheless, it is still possible for attackers to read, and possibly write, the secret bits in the non-volatile memory through the electron beam of a Scanning Electron Microscope (SEM) once the surface of the chip is exposed by, for instance, Focused Ion Beam (FIB).Physical unclonable functions (PUFs), in contrast, are 'inseparable' because the underlying nano-scale structural disorder will most likely be damaged during the course of physical tampering of the device, so will the keys [14]. Since the first introduction of PUFs in citepappu01, many types of circuitry have been proposed to realize the notion.

As electronic devices become ubiquitous and interconnected, people are increasingly relying on integrated circuits (ICs) for performing security sensitive tasks as well as handling sensitive information. For example, an RFID is often used as a key card to control access to buildings, smart cards carry out financial transactions, and mobile phones often contain sensitive data such as confidential documents, personal emails, etc. Therefore, it is critical for ICs to be able to perform operations such as authentication of devices, protection of confidential information, and secure communication in an inexpensive yet highly secure way.

A common ingredient that is required to enable the above security operations is a secret on each IC, which an adversary cannot obtain or duplicate. The current best practice is to place a secret key in non-volatile memory such as fuses and EEPROM, and use cryptographic primitives such as digital signature and encryption to authenticate a device and protect confidential information.

Two important metrics that are typically applied to categorize the uniqueness and robustness of PUF responses and UNO fingerprints are inter-device and intra-device distances. Inter-device distance is often quantified as the average Hamming distance between the responses to the same challenge obtained from two different PUFs/UNOs, or the average distance between the fingerprints of two unique objects measured in the same conditions. Intra-device distance is the average Hamming distance between the responses to the same challenge applied at different times and environmental conditions to the same PUF/UNO, or the average distance between the repeatedly measured fingerprint(s) of a unique object. Ideal PUFs and UNOs should lead to large inter-device and small intra-device distances. Another key requirement for PUFs and unique objects is the entropy of the resulting responses or fingerprints. The entropy quantifies the number of independent IDs that can be generated by the same device architecture.

Despite the similarities between UNOs and PUFs, there are several important differences between them that distinguish these two security primitives (and their subclasses) from each other. This chapter provides a conceptual categorization and summary of the field of physical disorder based cryptography and security, also termed physical cryptography. Whenever applicable, the concepts are interleaved with examples from the contemporary literature and implementation details. A number of surveys on the PUF subject are already existent, for example,and a recent book with several chapters dedicated to PUFs [12]. We will cite these review articles whenever applicable, and emphasize that the concepts in this chapter are complementary to the contemporary literature in this area.

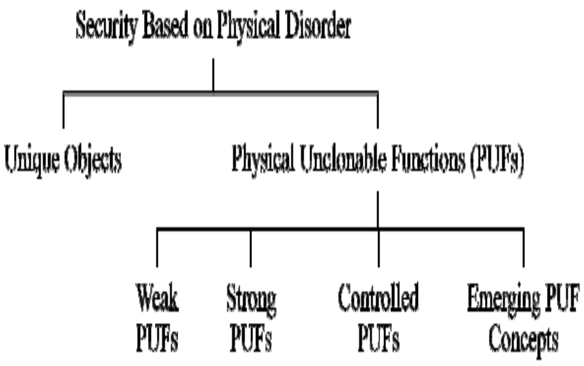

Security Based on Physical Disorder
Unique Objects
Physical Unclonable Functions (PUFs)
Weak PUFs
Strong PUFs
Controlled PUFs
Emerging PUF Concepts

Fig. 1. Organization of the Chapter.

Figure 1 gives an overview of the classes of physical disorder based security tokens discussed in this chapter. Each of the discussed subjects are shown as a branch in the chart. The next section reviews UNOs including paper-based (fiber-based) fingerprints, magnetic signatures, and RF-based Certificates of Authenticity. The weak PUF class including Physically-Obfuscated Keys, SRAM-PUFs, and butterfly PUFs. Strong PUFs are the subject of Section 4. Examples of PUF structures that can provide building blocks for Strong PUFs include optical PUFs, arbiter PUFs, XOR arbiter PUFs, and analog cellular arrays.

Physical unclonable functions (PUFs) are novel security primitives that store secret keys in physical objects by exploiting the uncontrollable randomness due to manufacturing process variations. PUFs generate signatures based on the unique intrinsic uncontrollable physical features, which can then be used for hardware authentication or the generation of secret keys. Contrary to standard digital systems, PUFs extract secrets from complex properties of a physical material rather than storing them in a non-volatile memory. It is nearly impossible to predict, clone or duplicate PUFs. Furthermore, an adversary cannot easily mount an attack to counterfeit the secret information without changing the physical randomness. Based on these advantages, PUFs can efficiently and reliably generate volatile secret keys for cryptographic operations and enable lightweight and cost-effective authentication of ICs.

II. LITERATURE SURVEY:

A. *Silicon MUX PUF:*

There are several subtypes of PUFs, each with its own applications and security features. A major type is the socalled silicon PUFs, which exploit the delay variations of circuit components to generate a unique signature for each IC. Silicon PUFs can be integrated into chips very conveniently, since these are implemented with standard digital logic and do not require any special fabrication. The examples of Silicon PUFs include: 1) MUX PUF 2) ring oscillator PUF 3) SRAM PUF and 4) butterfly PUF

A MUX PUF is an example of a Strong PUF that is unclonable due to manufacturing process variations, and can accommodate many possible challenge-response pairs (CRPs). As illustrated in Fig. 1, in a MUX PUF, each challenge creates two paths through the circuit that are excited simultaneously. The output is generated according to the delay difference between the two paths. A MUX PUF consists of N stages of MUXs and one arbiter which connects the last stage of the two paths. MUXs in each stage act as a switch to either cross or straight propagate the rising edge signals, based on the corresponding challenge bit. Each MUX should be designed equivalently, while variations will be introduced during manufacturing process. Finally, the arbiter translates the analog timing difference into a digital value. For instance, if the rising edge signal arrives at the top input of the arbiter earlier than the signal arriving at the bottom input, the output will be one; otherwise, if it reaches the bottom path first, the output will be zero. The output response depends on the applied challenge bits and will be permanent for each IC after fabrication or only vary in a small range due to environmental variations.

For transistors, manufacturing randomness exists due to variations in transistor length, width, gate oxide thickness, doping concentration density, body bias, metal width, metal thickness, and interlevel dielectric (ILD) thickness, and so on.

These manufacturing variations lead to a significant amount of variability for the MUX-based PUFs, which are sufficient to generate unique challenge-response pairs for each IC by comparing the delays of two paths.

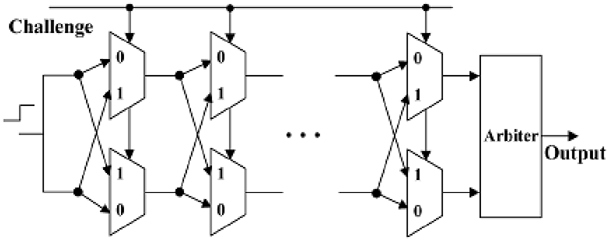


Fig. 1. Silicon MUX physical unclonable function.

B. *Feed-Forward MUX PUF*

In order to improve the security, a feed-forward structure has been proposed to add non-linearity into the original MUX PUF. In a feed-forward MUX PUF, the output of a feed-forward arbiter (FF arbiter) from an intermediate stage is used as a challenge to a subsequent stage. Fig. 2 shows one basic structure of the feed-forward MUX PUF, which uses the racing result of an intermediate stage as the select signal for a later MUX stage. This structure increases the complexity of numerical modeling attacks [18]. However, the reliability of the PUF has been degraded since some select signals of the MUXs may also be affected by environmental variations.

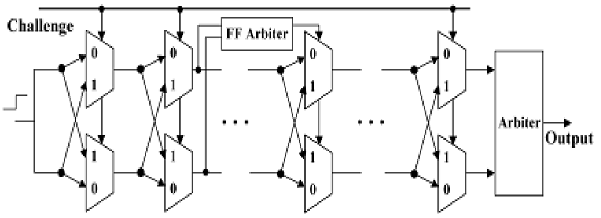


Fig. 2. Feed-Forward MUX PUF structure.

C. *MUX-based Reconfigurable PUFs:*

Based on the MUX PUF and its feed-forward variants, we have proposed several novel reconfigurable PUFs, where the CRPs can be reconfigured. Reconfigurable PUFs satisfy the updatable key requirement for PUF-based authentication systems. Furthermore, reconfigurability improves the security against modeling attacks by limiting the amount of information leaked for each configuration. Such architectures are classified into the following two categories.

1) CRP-Reconfigurable PUF: The challenge-response pairs are reconfigured directly by adding some additional configure circuits into the structure, but without configuring the main PUF structure. This can

be achieved by preprocessing the challenge before applying to the PUF or preprocessing the response before using it for authentication.

2) Logic-Reconfigurable PUF: The underlying logic of the PUF circuit is reconfigured in these structures; therefore, the challenge-response pairs are reconfigured.

Logic-reconfigurable PUFs have better performance from a security perspective, as reconfiguration leads to a different mathematical model of the PUF circuit, while the CRP-reconfigurable PUFs only update the CRPs. The CRP reconfigurable PUFs are not studied in this paper. The examples of logic-reconfigurable PUFs include logic-reconfigurable feed-forward MUX PUF and MUX/De MUX PUF.

1) Logic-Reconfigurable Feed-Forward MUX PUF: We had introduced three different types of feed forward MUX PUFs. These structures include feed-forward overlap (FFO), feed-forward cascade (FFC), and feed-forward separate (FFS). These structures are classified by the nature of interconnections of various feed-forward patterns in these PUFs. We had also shown that the performance of a feed forward MUX PUF depends on locations and the number of feed-forward paths (sometimes referred as feed-forward loops). The three feed-forward structures are described below.

a) FFO: This structure has at least one stage overlap between two feed-forward paths as illustrated in Fig. 3.

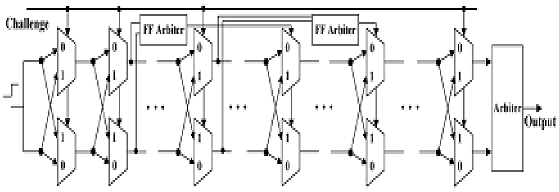


Fig. 3. Feed-Forward MUX PUF overlap structure

a) FFC: The ending stage of a feed-forward path will be the starting stage of another feed-forward path. This is illustrated in Fig. 4.

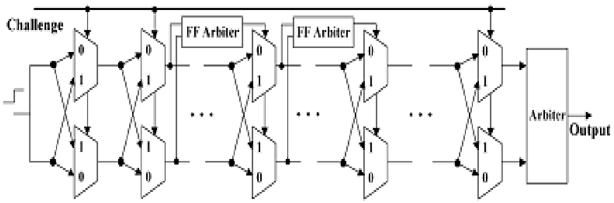


Fig. 4. Feed-Forward MUX PUF cascade structure

a) FFS: Different feed-forward paths are separate; thus, no stage overlap exists between the two feed-forward paths. This is illustrated in Fig. 5.

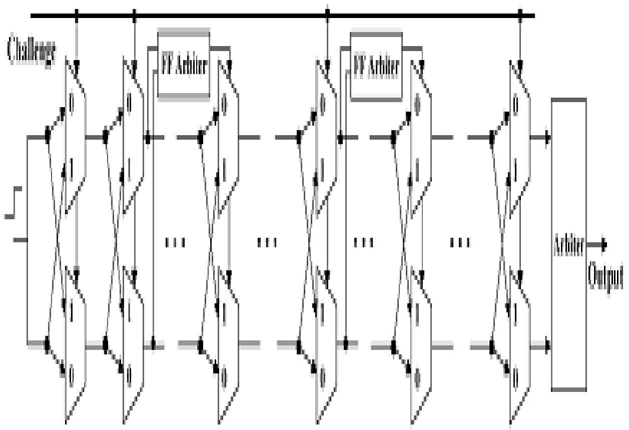


Fig. 5. Feed-Forward MUX PUF separate structure.

We have simulated these three feed-forward structures and have shown that these structures satisfy different interchip and intrachip characteristics. Based on this property, we have proposed a logic-reconfigurable feed-forward MUX PUF, which can be configured to any of these three different structures (i.e., FFO, FFC, and FFS).

2) MUX/DeMUX PUF: Another MUX-based logicreconfigurable PUF is the MUX/DeMUX PUF, which alters the PUF logic by using DeMUX. DeMUX enables the circuit to select the direction of the propagating signals, and makes the original MUX PUF reconfigurable.

A basic structure is shown in Fig. 6. Instead of propagating the rising edge signal successively, some stages can be skipped by the DeMUX, which allows the challenge-response behavior to be reconfigurable.

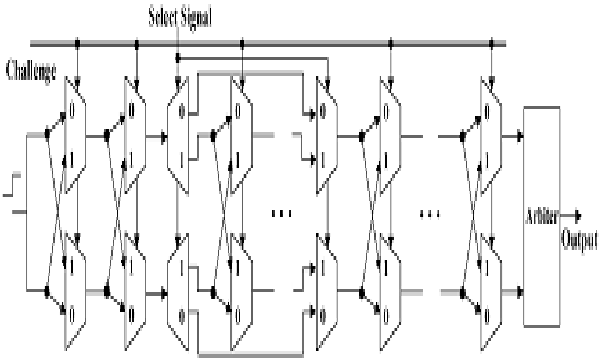


Fig. 6. MUX/DeMUX PUF.

III. Modified Feed-Forward MUX PUFs

A. Modified Feed-Forward Path

In this paper, we propose a novel modified feed-forward MUX PUF structure shown in Fig. 7, which is motivated by our statistical analysis results. In this structure, the output of a feed-forward arbiter from an intermediate stage is input as the challenge bit to two consecutive late MUX stages. By employing this modified feed-forward path, the reliability of the feed-forward PUF structure can be improved, while the same level of security will be retained.

This structure is analyzed statistically in this paper. The complexity of the modified feed-forward MUX PUFs can be further improved by using several modified feedforward paths in a PUF circuit.

Note that if we want to maintain the length of challenge bits asN, we need to increase the number of MUX stages to N+2Mfor the modified feedforward structure, compared to N+Mof the standard feedforward PUF,

whereMrepresents the number of feed-forward paths.

Additionally, the design overhead will also include arbiters for both the standard feed- forward MUX PUF and the modified feed-forward MUX PUF.

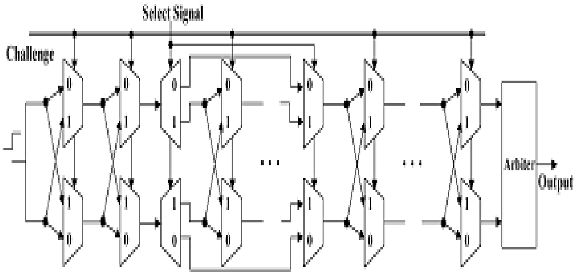


Fig. 6. MUX/DeMUX PUF.

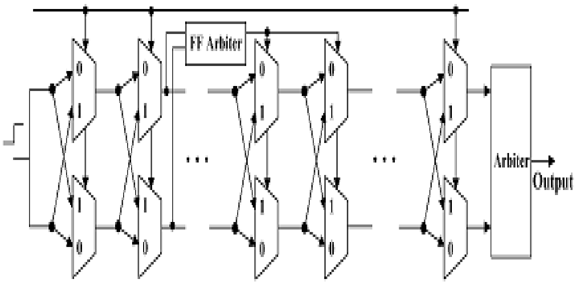


Fig. 7. Modified Feed-Forward MUX PUF structure.

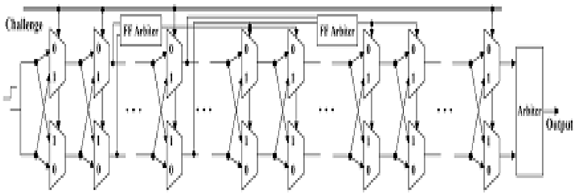


Fig. 8. Modified Feed-Forward MUX PUF overlap structure.

B. Different Types of Modified Feed-Forward MUX PUFs

The modified feed-forward MUX PUFs can also be classified as modified feed-forward overlap (MFFO), modified feed-forward cascade (MFFC), and modified feed-forward separate (MFFS) as shown in Figs. 8–10, respectively. These three different structures also have different interchip and intrachip behaviors. Additionally, the modified feed-forward paths can also be used in the logic-reconfigurable feed-forward MUX PUF to improve the reliability while retaining the high security.

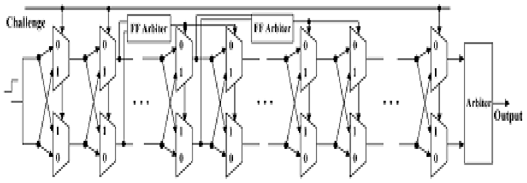


Fig. 9. Modified Feed-Forward MUX PUF cascade structure.

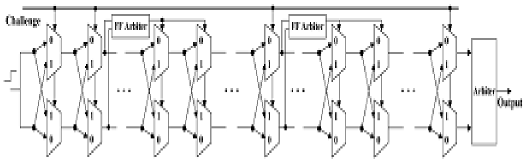


Fig. 10. Modified Feed-Forward MUX PUF separate structure.

IV. Definition of PUF Performance

The Monte-Carlo simulation can be used to provide performance indicators for different PUF designs. For example, by simulating the three feed-forward MUX PUF structures with the same parameter variations and environmental conditions, we were able to conclude in [19],
[20] that the FFO structure is the most reliable among the three feed-forward structures. In this paper, we focus on analyzing the quantitative performance of various MUX-based PUFs through statistical modeling of the delay variations and environmental variations. Performance indicators ranging from zero to one with one representing the best performance are generated through a theoretical analysis.

A. Reliability

Intra chip variation is a measure of the reliability of PUF, which is determined by comparing the digital signatures of the PUF to the same challenge under different environmental conditions. LetPintra represent the probability that a certain bit of a response will flip when applying a randomly selected challenge multiple times. All the bits of a PUF response have the same value ofPintra, since each bit is generated independently by a same PUF instance (i.e., the effects of manufacturing process variation and environmental variation for all the bits are the same). As a result,Pintra can be used to represent the intrachip variation for the entire L-bit response. In particular, the average Hamming distance (HD) between the responses is used to measure the intrachip variations of MUX-based PUFs. The Pintra and the averaged HD are described by

$$E(HD_{intra}) = P_{intra} = E\left(\frac{1}{m}\sum_{i=1}^{m}\frac{HD(R,R')}{L}\times 100\%\right) \quad (1)$$

Where m is the number of HD comparisons, and R and R' represent two measurements of the PUF response under different conditions. The expected value ofHDintra is equal to Pintra . If the responses are sampled sufficient number of times, the averaged intrachip variation would be close to the value of Pintra . As smaller intrachip variation means better reliability, the reliability indicator is defined as

Reliability=1−Pintra. (2)

B. Uniqueness

Interchip variation is a measure of the uniqueness of PUF, which is determined by comparing the digital signature of a PUF to that of another. Similarly, we can also definePinter as the probability that the bits generated by the same challenge for different PUF instances are different. Since uniqueness is a measure of interchip performance, all possible chipcombinations should be considered. Therefore, the average interchip HD of KPUFs can be described as

$$E(HD_{inter}) = P_{inter}$$
$$= E\left(\frac{2}{(K-1)K}\sum_{i=1}^{K-1}\sum_{j=i+1}^{K}\frac{HD(R(i),R(j))}{L}\times 100\%\right). \quad (3)$$

It can also be seen thatPinter represents the expected value of the interchip variation. Since Pinter = 50% represents the best uniqueness for a PUF, the uniqueness indicator can be defined by

Uniqueness=1−|2Pinter −1|. (4)

C. Randomness

A MUX-PUF is expected ideally to produce unbiased 0’s and 1’s. Randomness represents the ability of the PUF to output 0 and 1 response with equal probability. One measurement of the randomness can be expressed as

$$\text{Randomness}=1-|2P(R=1)-1|. \quad (5)$$

Therefore, a randomness of one indicates unbiased PUF responses.

NOVEL RECONFIGURABLE PUFS

In order to add reconfigurable property into general MUX based silicon PUFs, we must make the challenge-response pairs (CRPs) reconfigurable, which can be used to update the database for an authentication system. The methods can be classified into two categories:

(a) Make the challenge-response pairs reconfigurable directly, by adding some extra circuits into the structure, but without configuring the main PUF circuit. This can be achieved by utilizing some techniques to pre-process the challenge before applying to PUF or pre-process the response before using it for authentication.

(b) Make the PUF circuit reconfigurable, therefore the challenge response pairs will be reconfigurable as well.

We propose several novel non-FPGA reconfigurable PUFs implementations for the above two categories, which would be more suitable for practical use than FPGA-based techniques. Furthermore, we address the reliability and the security of the PUF performance, as some information of the hidden secrets that an adversary can take advantage of may leak out during reconfigurations.

Reconfigurable Challenge and/or Response Structures

The reconfigurable structures of PUF are built on the prior work in Physical Unclonable Function, which can also be applied to various types of silicon PUFs as well as other challenge-response based PUFs. Our goal is to develop reconfigurable PUF which is a PUF with a mechanism to transform it into a new PUF with an unpredictable and uncontrollable challenge-response behavior, even if the challenge-response behavior of the original PUF is already known. Additionally, the new PUF inherits all the security properties of the original one. An early reconfigurable design PUF [9] in the literature treated some challenge bits as the configure data. As an example, the last 10 bits of a 100-bit challenge can be fixed as the configure data, leaving only 90 bits for actual challenge. A user can update the CRPs by applying another 10-bit stream to the last 10 stages of the PUF. However, it is very clear that the reconfigured PUF will have high correlation between different configurations and will be vulnerable to attacks, as this method is similar to adding a certain time difference between the two paths or introducing an interval between the two rising edge signals. Even worse, the performance of the PUF will be greatly degraded, if the cumulative variations in the last 10 stages are relatively large. Due to these disadvantages, this architecture of reconfigurable PUFs cannot generate unpredictable challenge-response behaviors. Intuitively, adding reconfigurable elements before the challenges applied to the PUF can definitely make the PUF reconfigurable. At the same time, the performance of the original PUF will be preserved. The main structure of this type of reconfigurable PUF is shown in Figure

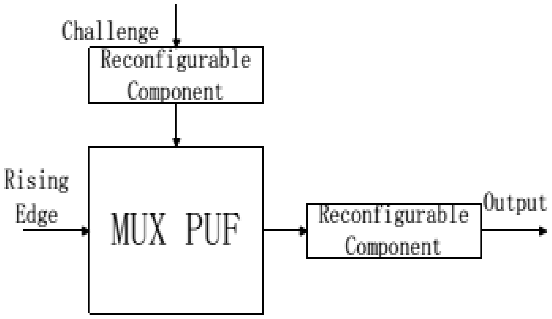


Fig.11 Reconfigurable Challenge and Response PUF Structure.

PUF with LFSR

We can adopt the linear feedback shift register (LFSR) as the reconfigurable component. Such a structure is shown in Figure . LFSR is an important part of sequence cipher and can be used to generate pseudo-random key stream. We can apply different seeds to the IC to generate various random patterns. Furthermore, we can also alter the characteristic polynomial by utilizing the properties of reconfigurable linear feedback shift register [18, 19]. Such capability makes it extremely difficult for adversaries to obtain PUF signature. It is important to point out that we can improve the security of the PUF system, by benefiting from the property of the LFSR in cryptography.

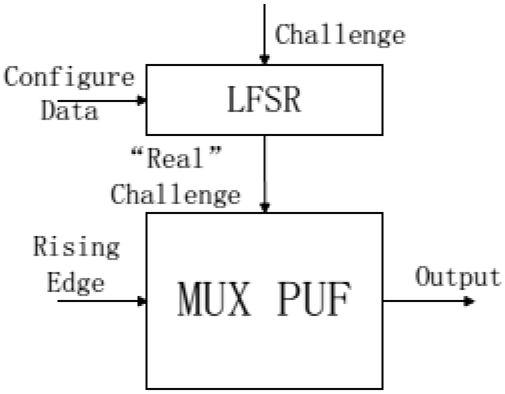


Fig.12 PUF Structure of Using LFSR to Configure the Challenge.

PUF with Hash Function

Hash function is a kind of "one-way" function, which means it is easy to compute the hash value for a given message, but hard to find a message with a given hash. Due to the random property of hash function, we can employ a hash function as the reconfigurable element to generate a reconfigurable PUF. This structure can be reconfigured very easily, such as by adding several different lengths of 0's at the end of every challenge. Additionally, the security of PUF can be increased, due to the "one-way" property of hash function. Many hash algorithms have been investigated and developed in the last years. Currently, the SHA-1 algorithm is the National institute of Standards and technology (NIST) secure hash standard. Several reconfigurable hash function unit architectures have been published in past years [20]. In fact, this structure has already been named as Controlled Physical Unclonable Function in [21], which was described as adding control logic to a PUF structure to prevent an adversary from accessing the PUF directly. Instead of doing a simple hash before the challenges applied to the PUF, we can consider adding another control logic, which would make the CRPs updatable. We propose several reconfiguration methods:

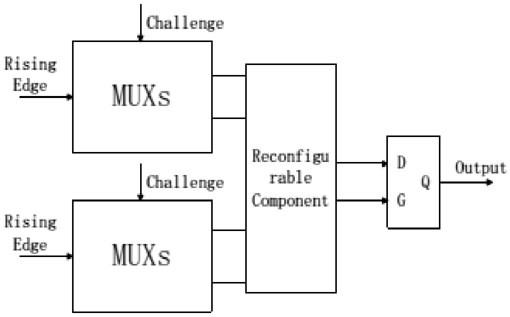
(a) Adding different bit streams into the challenges, e:g:; adding different numbers of 0's at the end of the challenges.

(b) Reordering the challenge stream by certain rules.

(c) Reconfiguring the hash function, by using the reconfigurability of these reconfigurable hash function implementations.

Due to the property of hash function, it is extremely hard for an adversary to model the PUF, even after we configure it several times, since the output of hash function is unpredictable.

PUF with Output Recombination



Fig,13 Two Parallel MUX PUF Structure

Another idea is to add an extra reconfigurable component to preprocess the output of the arbiter before using it as an authentication key. One simple example is to use two parallel MUX PUFs to update the CRPs, as shown in Figure 5. In this case, the signal (rising edge) will propagate through 4 paths which are selected by challenges. Then we can select two of the four paths using the configure data and forward to the arbiter to generate the response. We will have a total of 12 possible combinations if we use a 2 parallel MUX PUFs. Therefore, we can reconfigure this architecture 12 times. However, there will be very high correlations among these 12 different combinations. For example, if we know that path 1 is faster than path 2, and path 2 is faster than path 3, then we can conclude that path 1 will be faster than path 3. Therefore, there should be some constraints for the pre-processing, which will decrease the total number of reconfigurations. In fact, there are N ! possible cases for ordering N paths based on their arrival time. Therefore, log2(N !) independent bits can be produced by N paths. We can increase the number of parallel PUFs to obtain more possible combinations to meet the practical application needs. If we want to achieve the entropy limit as log2(N !), we need to choose the output comparison pairs adaptively, which would increase the design complexity and fabrication area significantly.

However, there will be a problem by

employing this structure, since the pre-processing component after the last stage also has variations, which will affect the performance of the PUF. To solve this problem, we can add pre-processing components after the arbiters, as in structure of Figure 6. If we use N parallel MUX-based PUF, we will need 2N-1 arbiters, where we only compare the neighbor paths. This is a concept borrowed from ring oscillator PUF which could ensure there will be no correlation between the output bits of the arbiters, as the comparison pairs are non-cyclic. Therefore, this structure can update its challenge-response behavior in an unpredictable manner.

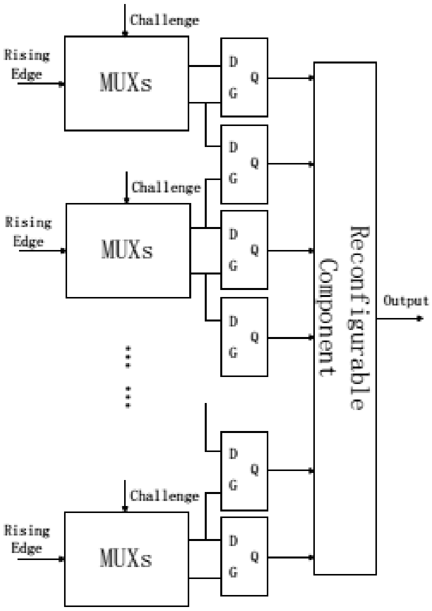


Figure 14: MUX PUF Structure of Output Recombination.

IV.EXPERIMENTAL RESULTS

All of our experiments have been carried out using SPICE simulations on a 65-nm technology process. We use Monte Carlo method to simulate the effect of process variations and environmental variations. In our simulation, we set up the transistor parameters and process variations based on a major industrial standard model. Each proposed structure has been simulated over at least 20 Monte Carlo runs in SPICE. We simulated 100 MUXs stages for each structure of these silicon PUFs. Accordingly, we need to apply a 100-bit challenge to the PUF to produce a 1-bit response; as a result, 100 different challenges were required to generate the final 100-bit digital signature for each IC.

Simulated PUF Structures: In our experiment, we added 10 feed-forward arbiters into each feed-forward structure of MUX PUF. For instance, the feed-forward arbiters were from stage 1 to stage 11, from stage 11 to stage 21 ... from stage 91 to stage 100 in a feed-forward cascade structure. The feed-forward arbiters were from stage 1 to stage 7, from stage 11 to stage 17 ... from stage 91 to stage 97 in feed-forward separate structure. In a feed-forward overlap structure, the feed-forward arbiters were from stage 1 to stage 51, from stage 6 to stage 56 ... from stage 46 to stage 96. For the reconfigurable feed-forward structure, we also added 10 such arbiters and MUXs structures (as discussed in Section 4.2.1) into the original PUF circuit, which can switch among the 3 different feed-forward structures. Moreover, we also simulated 10 DeMUX components in the MUX and DeMUX PUF. The inputs and the outputs of the DeMUXs were from stage 3 to stage 8, from stage 13 to stage 18 ... and from stage 93 to stage 98. Finally, we simulated an output recombination structure with 20 parallel MUX PUFs. We derived the digital signature by comparing adjacent paths among the total 40 paths. Therefore, except for the first and the last paths, each path was compared to two other paths.

Inter-chip Variations: The inter-chip variations were evaluated by the Hamming distance between two digital signatures which were generated by a same challenge and configure data from different chips. Since we simulated 20 chip instances, we had 20*19/2, i.e., 190 possible digital signature comparisons. We provide the maximum and the minimum of these numbers for the inter-chip variations.

Intra-chip Variations: The intra-chip variations were determined by comparing the digital signatures of the same IC under different environmental conditions; in our case, we consider the temperature as the primary environmental factor. It has been shown in our experiment that the intra-chip variations introduced by different temperatures from 0o C to 100oC were more significant then the intra chip variations caused by voltage varying from 1V to 1:2V . The digital signatures of the PUF at 0oC, 20oC, 40oC, 80oC, 100oCwere obtained; however, we only present the comparisons between0oC and 100oC, as those exhibited the largest variations. We applied 10 different challenges for each IC, and simulated 20 different IC instances. Therefore we had 200 comparisons in total. We provide the maximum and the average of these values of Hamming distance for the intra-chip variations.

Reconfigurability: The reconfigurability was determined by the variations of digital signatures generated by different configure data in a same IC. We fixed the challenge for a reconfigurability test, while we fixed the configure data of the different structures when examining the inter-chip variations and the intra-chip variations. In fact, the challenge-bit lengths were decreased by adding reconfigurable components in the feed-forward structures; therefore, we need to adjust the challenge bits when simulating these reconfigurable structures. We also applied 10 different configure data for each IC, and simulated total 20 different IC instances, which are similar to intra-chip variation test. All the simulations were done under the environmental condition of 25oC and 1:1V.

CONCLUSION

We have presented several reconfigurable silicon MUX Physical Unclonable Functions based on two major approaches and demonstrated their effectiveness by experimental results via inter-chip variation and intra-chip variation. We also have discussed the reliability perspective of PUFs and proposed several methods to increase the security. Ongoing work includes novel highly secure and reliable reconfigurable PUF designs and their mathematical analysis. Furthermore, we are also interested in developing an authentication scheme for reconfigurable PUFs, which will use several pairs of CRPs as a set for authentication by utilizing the reconfigurable property of reconfigurable PUFs.

REFERENCES

[1] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, “Physical one-wayfunctions.” Science, vol. 297(5589), p. 2026, 2002.

[2] B. Gassend, D. Clarke, M. V. Dijk, and S. Devadas, “Silicon physicalunclonable functions,” the 9th ACM Conference on Computer andCommunications Security, p. 160, 2002.

[3] ——, “Controlled physical unclonable functions,” in Computer SecurityApplication Conference, 2002, pp. 149–160.

[4] S. Kumar, J. Guajardo, R. Maesyz, G. Schrijen, and P. Tuyls, “Extendedabstract: The butterfly PUF protecting IP on every FPGA,” Hardware-OrientedSecurity and Trust (HOST 2008), pp. 67–70, 2008.

[5] R. Maes, P. Tuyls, and I. Verbauwhede, “Intrinsic PUFs from flip-flops onreconfigurable devices,” in Benelux Workshop Information and System Security(WISSec 08), 2008.

[6] D. E. Holcomb, W. P. Burleson, and K. Fue, “Initial SRAM state as a fingerprintand source of true random numbers,” in Conference on RFID Security, 2007.

[7] U. Ruhrmair, F. Sehnke, J. Solter, G. Dror, S. Devadas, and J. Schmidhuber,“Modeling attacks on physical unclonable functions,” in Conference on RFID Security, 2010.

[8] M. Majzoobi, F. Koushanfar, and M. Potkonjak, “Techniques for design and implementation of secure reconfigurable PUFs,” ACM Transactions on Reconfigurable Technology and Systems, vol. 2, no. 1, pp. 1–33, 2009.

[9] D. Lim, J. W. Lee, B. Gassend, G. E. Suh, M. V. Dijk, and S. Devadas, “Extracting secret keys from integrated circuits,” IEEE Transaction on Very Large Scale Integration Systems, vol. 13, no. 10, p. 1200, 2005.

[10] H. Chang and S. Sapatnekar, “Statistical timing analysis considering spatial correlation in a pert-like traversal,” in IEEE International Conference Computer-Aided Design Integrated Circuits and Systems, 2003, pp. 621–625.