Journal of Electronics and Information Science (2023)
Clausius Scientific Press, Canada

DOI: 10.23977/jeis.2023.080409
ISSN 2371-9524 Vol. 8 Num. 4

Uniformity optimization method for arbiter physically unclonable functions

Jiang Junhao

Jining Confucius International School, Jining, China
junhao.jiang@outlook.com

Keywords: PUF, uniformity, optimization

Abstract: In recent years, hardware encryption based on physically unclonable functions (PUFs) has increased in significance in information security. In general, a PUF is evaluated by three indicators: reliability, uniqueness, and uniformity. In this paper, the arbitrator PUF is mainly studied, and a uniformity adjustment scheme is proposed. The main idea of this scheme is to add an adjustment module to the original circuit and construct an algorithm that automatically sets the adjustment signal according to the response. Finally, we use FPGAs to test the uniformity, reliability, and uniqueness of the adjusted PUF.K.

1. Introduction

With the rapid development of the Internet, the issue of information security has attracted more and more attention. As a resource, the universality, sharing, and multi-use of information are of great significance to society as a whole[6]. However, in modern society, due to various reasons, a lot of information is stolen on the network, causing immeasurable economic losses. While automation and sensing technology continue to advance, it also poses a great threat to the storage and security of information. In addition, the Internet of Things has also attracted more and more widespread social attention regarding information security. Modern information security systems are mainly based on encryption systems, which can be divided into software encryption and hardware encryption. Hardware encryption systems have encryption devices that can store information and perform encryption or decryption themselves. Functions can only be called externally through defined interfaces, so internal information and operational procedures cannot be accessed directly. Therefore, hardware encryption has advantages over software encryption in terms of security. In addition, faster decryption speed and encryption speed, less running memory, and other characteristics make hardware encryption a promising research direction in encryption technology [3]. Currently, IoT communication security is still based on software and hardware security. However, hackers can also use this technology to damage the internal structure of the chip by copying the chip or splitting the chip. In the traditional key system, NVM (Non-Volatile Memory) has non-volatile, byte access, high storage density, low energy consumption, read and write performance close to DRAM, but the read and write speed is asymmetric, with reading much faster than writing, and it has a limited lifespan [2]. eFuse has a wide range of applications, from the adjustment, calibration, and repair of analog devices to on-site updates of system software, and it is widely used in the field of security. However, because the programming nodes of the eFuse can be seen through electron microscopy, the contents

stored in it can be easily cracked [1].

During the development of hardware encryption [7], the Physical Unclonable Function (PUF) gradually gained attention. PUF is a small, non-repeatable process deviation caused by unavoidable and uncontrollable minor disturbances in the manufacturing process. This slight deviation makes it impossible to produce two identical circuits at present. As a result, PUF has become the only certified hardware "electronic fingerprint." Currently, the structural differences of PUFs can be divided into three types: non-electronic PUF, analog PUF, and digital PUF; or divided into two types based on the relationship between the logarithm of the excitation response and the size of the physical entity: strong PUF and weak PUF [4]. Physical unclonable functions (PUFs) are given special digital characteristics due to the random nature of their manufacturing processes. PUF generates fast and secure keys through the "incentive-response" mechanism, which has strong resistance to physical attacks and can effectively address the security problems existing in traditional keys [5]. In addition, PUF has certain advantages. With the advancement of science and technology, the interconnection relationship between devices has received increasing attention. For example, with PUF technology, each device has its own password and cannot be duplicated, thereby achieving the purpose of protecting the security of the device. With the integration of integrated circuits, PUF applications are becoming increasingly extensive. Based on this, the response signal is combined with the output signal, and then the PUF is used to identify and analyze the response signal, thereby enhancing the system's security.

Delay-based PUFs include RO PUF, arbitrator PUF, etc. The arbitrator PUF consists of two delay lines and an arbitrator. N switching blocks are placed throughout the circuit, each containing two two-to-one multiplexers to implement logic functions. The delay of such an arbitrator PUF includes nominal delay (nominal gate delay and nominal wiring delay) and random delay. Finally, the arbitrator compares the path delay changes and generates a one-bit response (0 or 1). When the layout of both paths is the same, the nominal delay should be the same. Therefore, due to the effect of random delay, different excitation signals produce different output responses, which constitute the electronic fingerprint of the chip.

An effective PUF should meet the three key evaluation indicators of reliability, uniqueness, and uniformity. Reliability refers to the probability that the PUF's response remains unchanged after accepting the same challenge many times under different conditions. Ideally, the response should be stable. Uniqueness is an indicator of the nonclonality of PUF, measuring differences between chips. The ideal value for the average Hamming distance is 50%. Uniformity refers to the ratio between 1 and 0 responses under different challenges, ideally 50%. The better the consistency, the harder it is to predict a certain motivation-response pair. This paper takes the arbitrator PUF as the research object and is divided into five parts. The first part introduces the background of hardware encryption and PUF. The second part explains the basic principles of PUF for arbitrators and the algorithm for PUF for adjustable arbitrators. In the third part, FPGA is used to verify the feasibility of the algorithm and process the experimental data. Finally, the experimental results are analyzed in the fourth part.

2. Method

Due to uncontrollable process deviations, the uniformity of one PUF usually deviates slightly by 50%. However, if the deviation is large, the probability of one response will be significantly greater than that of another response, which breaks the unpredictability of response information and greatly reduces the difficulty of cracking PUF. This paper proposes an adjustable arbitrator PUF to solve this problem. The core idea is to increase the cache to balance the latency of the two paths. The process deviation of different chips is not the same, and the increased cache requirements are also different, so we need to propose a unified design method that can be oriented to different chips, and the proposed

circuit design diagram is shown in Figure 1.

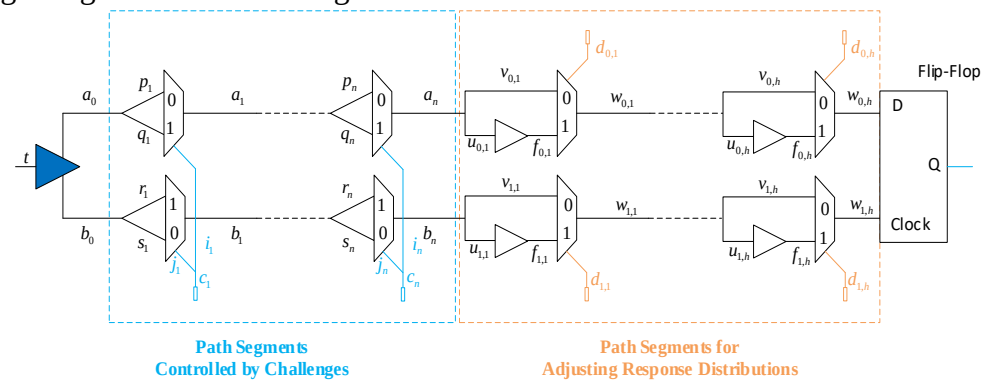

Figure 1: Circuit design diagram

The right half of Figure 1 is the adjustment circuit. In order to increase the application range of the tuning module, the adjustment circuit does not directly increase the specific delay of a specific path but changes the time delay through a control signal. As shown in Figure 1, when the control signal is 1, it connects to the h-path and inserts a buffer to add delay. The number of adjustment units and the delay range should be determined according to different requirements. The adjusted algorithm flow chart is shown in Figure 2.

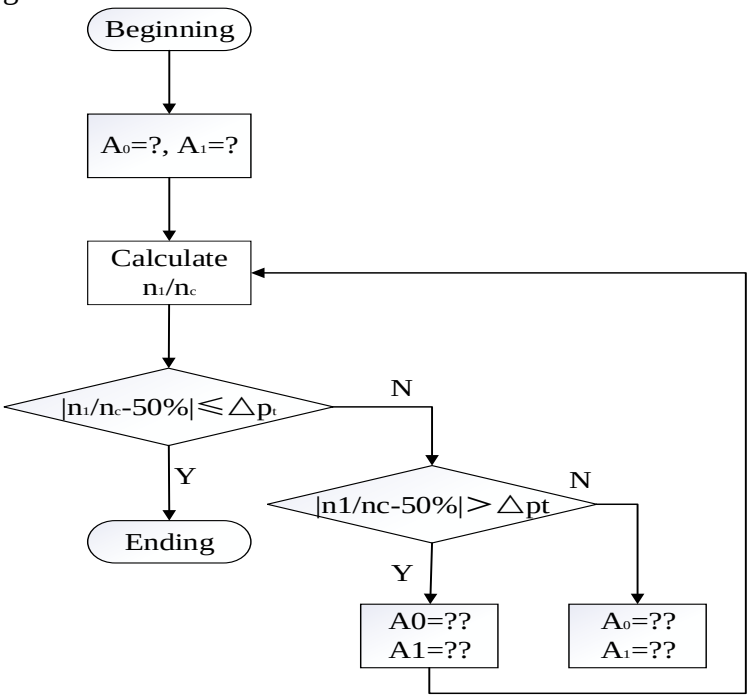

Figure 2: Flow chart of the adjusted algorithm

The algorithm gives random values from the adjustment signal (A_1、A_0、p_t、a_0、a_1). The proportion of 1 response is calculated after multiple runs and then compared to 50%. If the difference is less than or equal to a threshold (Δ), PUF uniformity is considered to have been adjusted to good. Otherwise, the program varies depending on the proportion of responses. If more than half of the response is "1", the signal will be adjusted randomly, x from 0 to 1, or x from 1 to 0. We re-run the

new circuit and evaluate its uniformity. If the new uniformity is close to 50%, the modification is retained, otherwise, until the difference is within the threshold. In some special cases, multiple loops may still not bring the difference to the threshold. Then, we take the one with the least difference in the whole process as the final result.

3. Experimental results and analysis

During the experiment, we use Xilinx Vivado tools to design on the Zedboard development board.

3.1. Code design

I directly call the Programmable Logic Gate (Look Up Table, LUT) to avoid comprehensive optimization of the circuit.

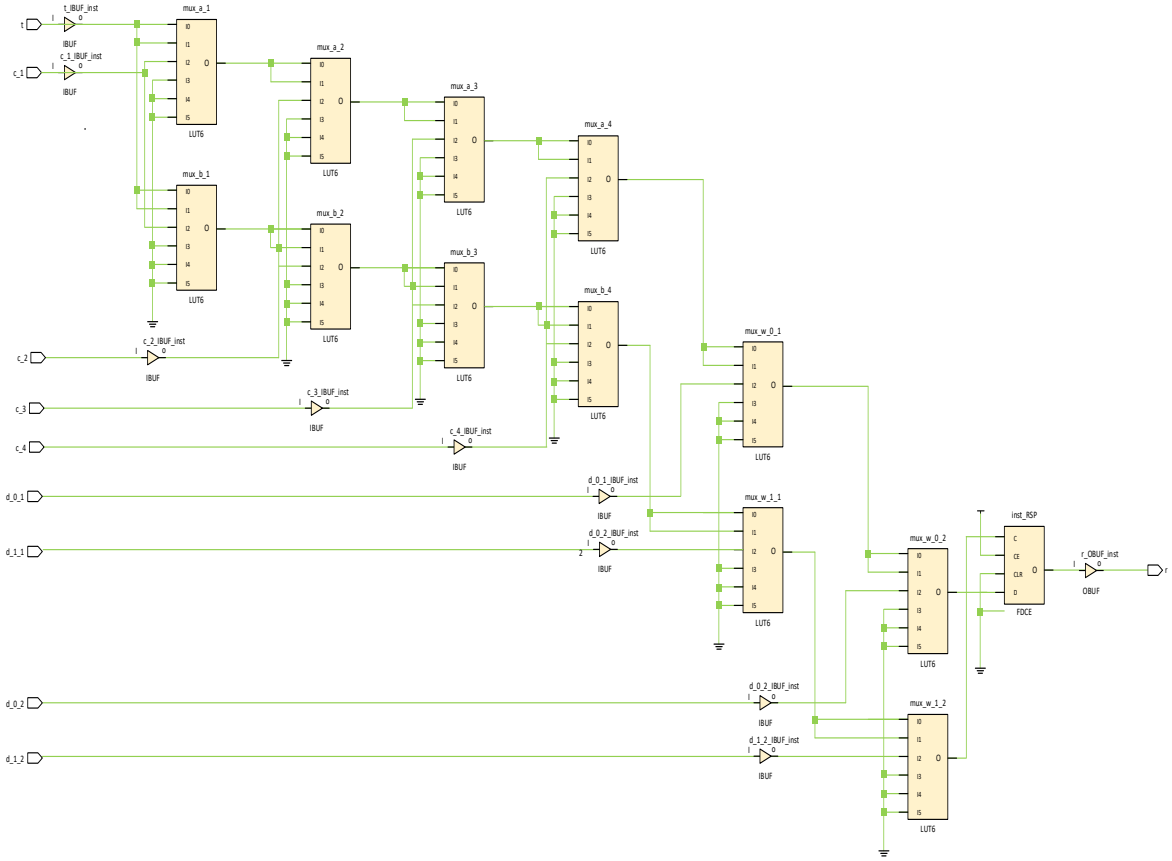

Figure 3: Programmable logic gates

3.2. Layout design

In order for the arbitrator PUF to be subject only to random process deviations when the FPGA implements the target circuit, it is assumed that the same chip model of the FPGA has the same nominal delay (generally automatically selects the shortest path) when using the same connection resources and logic gate resources. In this article, we use two identical modules to achieve the same spatial layout and line layout. As shown in Figure 3, P1 equals R1, and Q1 equals S1. While differences between P1 and Q1, and R1 and S1 are inevitable in this wiring mode, the differences are very small based on the net latency provided by the FPGA, so they are considered the same.

3.3. Experimental results

Table 1: The results of experiment

Uniformity, uniqueness and reliability of PUF	PUF1	PUF2
Uniformity	0.527	0.473
Uniqueness	0.502	
reliability	0.510	0.490

In PUF1, uniformity, uniqueness and reliability are all around 0.50. The uniformity in PUF1 is 0.054 higher than that in PUF2. But both of them have the same uniqueness. The difference between PUF1 and PUF2 in reliability is 0.20.

4. Conclusion and outlook

Ideally, the uniformity of PUF is 50%. The uniformity of the adjusted PUF achieved in the FPGA increases in practical applications as its value approaches 50%. Table 1 shows the uniformity, reliability, and uniqueness of two PUFs (PUF1 and PUF2) at the same challenge bit. It can be seen from Table 1 that after inserting a buffer in the low uniformity PUF, the uniformity of PUF1 and PUF2 is 0.526 and 0.493, respectively, while the reliability of PUF1 and PUF2 is 0.932 and 0.937, respectively, and the uniqueness of PUF is 0.503. The calculation results are in good agreement with the theoretical analysis results. It is verified that the overall delay difference between the two paths can be improved by inserting the buffer. Therefore, it is proved that the adjustment circuit proposed in this paper has high theoretical feasibility and practical operability, and the uniformity and uniqueness are close to the ideal value.

References

[1] Han Lijuan, Qian Lei, Yao Enyi, et al. Design of physical unclonable functional circuit based on hierarchical transformation[J]. Computer Science and Technology Research, 2021, 7 (2):86-93.

[2] IEEE Asian Solid-State Circuits Conference (A-SSCC), 2020, J. C. H. Chang, "Intelligent Chips and Techniques for AIoT Era" Note 1 to 4, doi: A-SSCC48613. 2020. 9336122.

[3] Liu Hailong. Research on Key Technologies and FPGA Implementation of PUF-based Key Generation [D]. Hubei: Huazhong University of Science and Technology, 2018.

[4] Ophey W., Skoric B., T. T. Integrated Physical Unclonable Function (Puf) with Combined Sensor and Display: US, 20080231418A1 [P]. 2008-10-22.

[5] Zhang Junqin, Gu Dawu, Hou Fangyong. Research and application of PUF in improved arbitration procedures [J]. Computer Engineering, 2010 (3):3.

[6] Wang Yi. The Research and Appucation of Hardware Cryptographic System [D]. Beijing: Beijing University of Posts and Telecommunications, 2006.

[7] W. Zhenyu, D. Ding, G. Yang and L. Shaoqing, "HCRO-LKSM: A Lightweight Key Sharing and Management Protocol Based on HCRO-PUF for IoT Devices, " 2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys), Hainan, China, 2022, pp. 1366-1373, doi: 10. 1109/HPCC-DSS-SmartCity-DependSys57074. 2022. 00212.