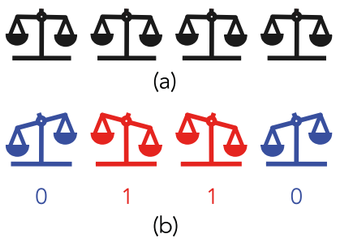
# A new stabilizer solution for Physically Unclonable Constants

*Physically Unclonable Constants, also known as Physically Obfuscated Keys, are small circuits that can be embedded in a chip to generate a secret code in the form of a bit-string that can be used to verify the chip's authenticity. These generated values are not stable and can vary at different turn-ons. A single wrong bit can make the whole system useless, so stabilizers are used to make a PUC's outcome as constant as possible. Riccardo Bernardini, an Aggregate Professor in the DPIA at the University of Udine, examines available stabilizers and proposes a new stabilizer solution with very low overheads.*

The increasing need for hardware security has motivated research into the implementation and application of various cryptographic schemes. Verifying the authenticity of a chip in a way that is both simple and secure is a necessity. Such verification, however, brings with it the problem of how to store secret information on a chip so that even an attacker who can physically open the chip and study it, is unable to read the secret information. This has led to the development of Physically Unclonable Functions (PUFs) in the field of security. The PUF acts as a function that maps the input binary words, or bit-strings. The mapping depends on fine details, such as the semiconductor's oxide thickness and the concentration of the dopant used to modify its electrical conductivity. This means that the behaviour of a particular PUF instance is very difficult to predict and reproduce. In a sense, a particular




The twin-plate example.

PUF instance can be thought of as the chip's 'fingerprint'.

A special class of PUFs that requires no inputs and always returns the same value is known as Physically Unclonable Constants (PUCs) or Physically Obfuscated Keys (POKs). These are small circuits that can be embedded in a chip to generate a secret code that can be used to verify the chip's authenticity. Not even the chip's producer can predict or reproduce the code, so they are 'unclonable'. The generated value produced by many PUC schemes is not stable, due to the uncontrollable fine details of the integrated circuit, and can vary at different turn-ons. In order to guarantee that the same secret bit-string is produced at every turn-on, stabilizer circuits are required. Riccardo Bernardini, an Aggregate Professor in the DPIA at the University of Udine, Italy, examines available stabilizers and proposes a new stabilizer solution.

## HOW IS A PUC DESIGNED?

Professor Bernardini explains that PUCs are essentially 'badly engineered' circuits. Circuits are usually designed so that their behaviour does not change as a consequence of imperfections making them resilient to small imperfections. In contrast, PUCs are designed to be very sensitive to small variations, so that even the tiny unavoidable variations that occur during productions will affect their behaviour in a consistent way. This makes an ideal PUC a random constant bit.

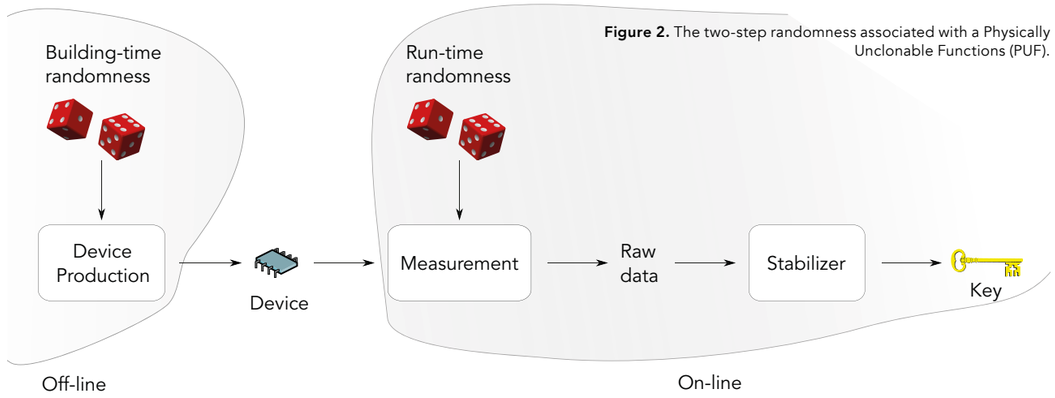



**Figure 2.** The two-step randomness associated with a Physically Unclonable Functions (PUF).

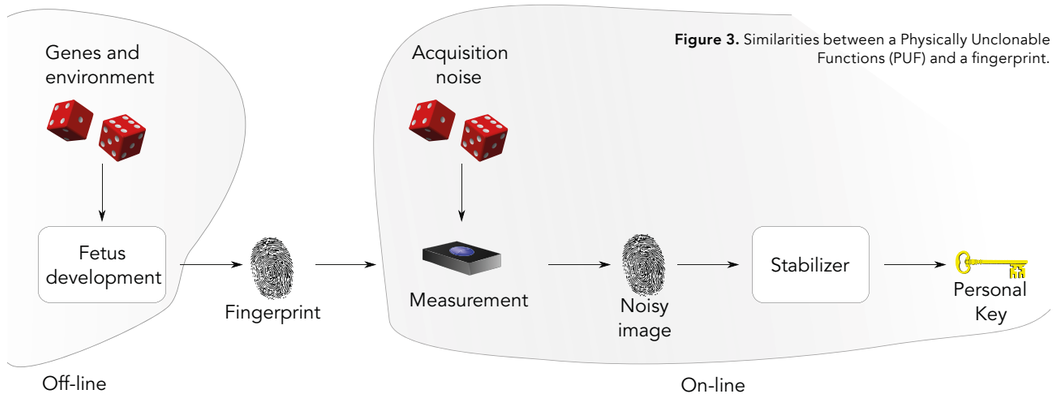



**Figure 3.** Similarities between a Physically Unclonable Functions (PUF) and a fingerprint.

He offers an informal example of a twin-pan balance to illustrate how PUCs work. Ideally, when a twin-pan balance is centred with zero forces acting on the fulcrum it would be perfectly symmetrical and remain in that position forever. No balance is absolutely symmetrical however, and unavoidable random imperfections will cause one side to be slightly heavier. Regardless of how small the difference is, the balance will drop on one side. These random imperfections can occur on either side of the balance scales, so the probabilities of finding a balance that drops to the left side or to the right side are equal. This makes the balance a 'random constant bit'.

## RANDOM CONSTANT BIT

While 'random constant bit' could be considered an oxymoron, the balance is random in the sense that one cannot predict which side of the balance will drop without 'querying' it by centring the balance and then letting it go. It is also constant in that every time the balance is queried it will return the same 'bit' by dropping to the same side.

***In order to guarantee that the same secret bit-string is produced at every turn-on, stabilizer circuits are required.***

The random part is easily achieved in most PUC schemes. The constant part, however, is more delicate. This is because the query result is influenced by both the construction time variations and query-time noises.

If we return to the balance example, we note that the reasoning assumes that the balance is frictionless. If, however, friction is present and the unbalance is very small, the outcome could also be influenced by additional variables, such as air flows, temperature differences or the surface supporting the balance not being exactly horizontal.

## STABILIZATION

In a cryptography-based protocol a single wrong bit can make the whole system useless. Stabilization

techniques are, therefore, required to make a PUC's outcome as constant as possible. Professor Bernardini notes that Forward Error Correction (FEC) codes are employed by many stabilization schemes to protect the data from error. Forward Error Correction is used in data transmission and involves adding some redundancy to the transmitted data which is then used to correct errors. This form of correction, however, is quite expensive both in terms of computing time and the space taken up on chip.

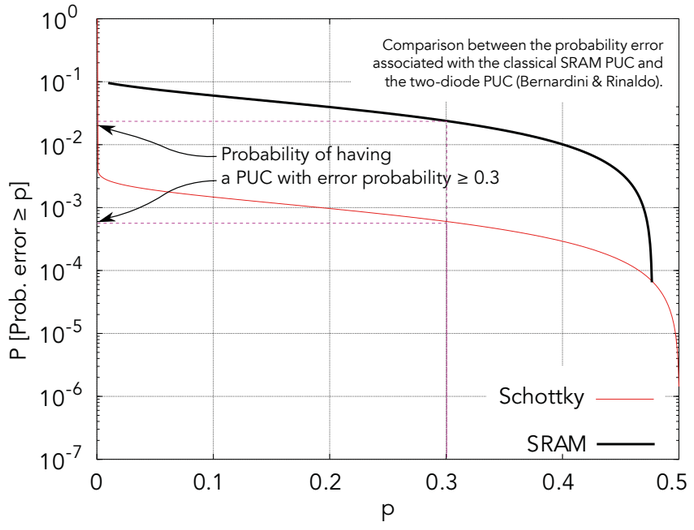
**AN ALTERNATIVE APPROACH**

Professor Bernardini has developed an alternative approach. Instead of adding redundancy, additional PUCs are embedded on the chip. The PUCs that are more likely to generate an error (similar to the balances with small asymmetries that make them drop to one side) are forced to zero. This could affect the randomness by making zero slightly more probable, but the randomness is recovered by mixing up all the bits and taking combinations of those that have been forced to zero and those that have not.

**USE CASE**

The researcher has provided some reference use cases so that the PUC can be considered in context: It is assumed that the PUC will be deployed in a user device such as a smart key that opens a door, a USB dongle that mutually authenticates with the host, or a disk controller that encrypts the data sent to the disk. The PUC is used for authentication and/or encryption purposes. The PUC is queried once at turn-on. Moreover, when the chip is tested at construction time, the quality of the PUC is also measured, and the chip is discarded if this does not satisfy some quality constraint.

It is also assumed that the user will notice when a key generation error occurs. For example, the door will not open, the dongle will not connect to the host, or the PC with the encrypted disk will not boot. It is likely that the user will try again and that the error will be considered an annoyance.




**TWO-STEP RANDOMNESS MODEL**

Professor Bernardini models the PUC as a two-step experiment. The first step occurs at construction time and determines the statistical behaviour of a specific instance. The second step occurs when the PUC instance is queried. The query outcome is then generated according to the statistical behaviour established in the first step. This new approach also takes into consideration that should environmental conditions such as temperature or supply voltage change, then the behaviour of the cell may also change.

Of particularly interest is the fundamental robustness of the proposed PUC, in terms of how difficult it is to guess the generated value, in order to ensure that the PUC is not the weak link of the system that it is embedded in.

**ADVANTAGES**

Professor Bernardini has compared the performance of his new approach with other existing schemes. Even though this solution uses more PUCs. it is more efficient overall because the step where the outcomes of the zero and non-zero cells are mixed is much simpler than Forward Error Correction. The main advantage of this new scheme is that unlike the majority of stabilizers described in the literature, an Error Correction Code (ECC) circuit is not required. This contributes to the very low overheads of the new scheme. Performance analysis has also demonstrated this new approach to be very secure. Furthermore, unlike some of the other ECC-less schemes, this new solution is not limited to the contexts used in its authentication; it is suitable for all PUF applications.

***This new solution is not limited to the contexts used in its authentication; it is suitable for all PUF applications.***




# Behind the Research

## Riccardo Bernardini

**E:** riccardo.bernardini@uniud.it **T:** +39-320-436-5972
**W:** https://www.linkedin.com/in/riccardobernardini/

## Research Objectives

Riccardo Bernardini's research interests include multidimensional signal processing, wavelets, filter banks, multimedia coding, robust transmission, bio-engineering, chaotic systems, P2P streaming and some security-related areas (such as random number generation, PUFs).

## Detail

**Adress**
Riccardo Bernardini
DPIA – Università di Udine
Via delle Scienze 208
33100 Udine – Italy

**Bio**

Riccardo Bernardini completed his PhD at the University of Padua (Prof. G. Cortelazzo). At the end of his studies, he worked on filter banks and wavelets at AT&T (Prof J. Kovačević). He then completed his post-doc at EPFL (Prof M. Vetterli). Some of the results have been patented. After a brief period in Padova where he worked on 3D acquisition and transmission, Riccardo moved to the University of Udine, where he is an Aggregate Professor in the Polytechnical Department of Engineering and Architecture (DIPA).

**Collaborator**

Roberto Rinaldo

## References

Bernardini, R., Rinaldo, R. (2020). Analysis of some simple stabilizers for physically obfuscated keys. *International Journal of Information Security,* [online] 19, 547–565. Available at: https://doi.org/10.1007/s10207-019-00473-8 [Accessed 4th January 2021].

Bernardini, R., Rinaldo, R. (2017). Making random permutations from physically unclonable constants. *International Journal of Information Security,* [online] 16, 249–261. Available at: https://doi.org/10.1007/s10207-016-0324-2 [Accessed 4th January 2021].

Bernardini, R., Rinaldo, R. (2017). A very stable diode-based physically unclonable constant. Integration, 59, 179–189.

Bernardini, R., Rinaldo, R. (2014). Theoretical limits of helperless stabilizers for physically unclonable constants. *IEEE Transactions on Emerging Topics in Computing,* 4, 73–87.

## Personal Response

***What inspired you to embed extra PUCs in the chip as an alternative to adding redundancy?***

"We developed some PUC schemes that gives, with overwhelming probability, very stable PUC instances, that is, instances very insensitive to query-time noises. We estimated that the stability was so good that one could skip the stabilization step altogether, at the price of a low yield (if a single "bad" cell was produced the whole chip would be discarded). The most obvious solution was to include more cells in order to increase the probability of having enough working cells. The necessity of guaranteeing a good statistical distribution triggered the research."