Journal of VLSI Circuits and Systems, ISSN: 2582-1458 Vol. 4, No. 1, 2022 (pp. 33-41)

RESEARCH ARTICLE WWW.VLSIJOURNAL.COM

Physically Unclonable Functions Using Two-Level Finite State Machine

Vallabhuni Vijay[1,*], Kancharapu Chaitanya[1], Chandra Shaker Pittala[3], Ajitha. G[1], S. Susri Susmitha[1], J. Tanusha[1], S. China Venkateshwarlu[1], and Rajeev Ratna Vallabhuni

[1]Department of Electronics and Communication Engineering, Institute of Aeronautical Engineering, Dundigal - 500043, Hyderabad

[2] Department of Electronics and Communication Engineering, MLR Institute of technology, Hyderabad-500043, Telengana, India

[3]Bayview Asset Management, LLC Florida, USA

KEYWORDS:
Cryptographic key generation, Finite-State Machine (FSM), Local Authentication, Physically Unclonable Functions (PUF), Secure PUF.

ARTICLE HISTORY:
Received 14.01.2022
Accepted 19.05.2022
Published 20.06.2022

DOI:
https://doi.org/10.31838/jvcs/04.01.06

Abstract

The usage of physically unclonable functions is for authentications, identification applications, signature generation, I.C. metering, and cryptographic key generation. Moreover, the utilization of smart devices is also growing, which is associated with security threats and alerts. The critical feature of PUF is reliance on random variations in the fabricated hardware to derive a known random key. These acquire error-correcting methods to generate PUFs responses across different temperatures. Recently, many PUF designs concentrate on exploiting design variations intrinsic to CMOS technology. Furthermore, PUFs are emerging with nanotechnology, which is not fully developed, but they are expected to develop further. This paper discusses a two-level finite-state machine that is proposed to correct erroneous bits created by temperature variations. In contrast, every response of PUF is mapped to a key during the initial stage of design, but the actual resolution is determined after the completion of chip fabrication; this is because the key is not known to the foundry; this approach prevents counterfeiting. In addition, to change keys, the challenges of the PUF have to change. Thus, access can be modified further, which gives more flexibility in securing utilized chips. We used a cadence tool to execute this proposed design, which produces software, hardware and silicon structures for I.C. designing systems on chips and printing circuit boards.

Author's e-mail: v.vijay@iare.ac.in, chandu.p4u@gmail.com, rajeevratna@ieee.org

How to cite this article: Vijay V, Chaitanya K, Pittala CS, Ajitha. G, Susmitha SS, Tanusha J, Venkateshwarlu SC, Vallabhuni RR. Physically Unclonable Functions Using Two-Level Finite State Machine. Journal of VLSI Circuits and Systems, Vol. 4, No. 1, 2022 (pp. 33-41).

Introduction

In recent years, physical unclonable functions (PUF) are the most research areas, and the utilization of physical unclonable functions (PUF) in new cryptographic technique also increasing.[1] Whereas PUFs are noisy where the response of PUF can be affected by environmental conditions like voltage drifts, temperature changes and so on. The reported PUFs like optical PUF, ring oscillator PUF, butterfly PUF; multiplexer PUF; SRAM PUF, sensor PUF and bi-stable ring PUF[2] are not 100% stable. The existing models of PUFs are remote authentication and local authentication.

Remote Authentication

It is the most existing PUF-based authentication that contains a device and so-called server. A communication link is established between server and device during the enrollment phase to collect unpredictable response from the device by providing randomly chosen challenges by the so-called server.[8] Whereas these trusted parties store the challenge-response pairs in the database for further authentication, later, if any device requests an authentication, these parties send challenges to that particular device and obtain PUF response[8] from the communication link.[8] However, the PUF response will

receive only when it matches the past response. The drawbacks of remote authentication are man-in-the-middle attacks and modelling attacks that create a software program after collecting the challenge-response pair. Figure 1.1 shows the process of challenge-response teams through the trusted party and untrusted environment.

Local Authentication

The problem of man-in-the-middle can be solved by a local authentication outline, where a communication link is not used to transmit challenge-response pair. The system is reliable and did not have fiddled with remains verified by utilizing local authentication to each component exclusive. Furthermore, local authentication is helpful in different design layers; for example, a controller can validate every I.P. block, whereas an I.P. block can validate an individual functional unit. In addition to this, local authentication is helpful when the communication link is not established between server and device or when the server is not available. In contrast, local authentication can accomplish I.P. binding and I.C. metering. The process of local authentication is disliked the remote authentication [8] that authentication is kept on the third-party local authentication utilizes chip to store secret information. The method of local authentication is after completion of fabrication; the challenge-response key is stored in a memory chip which is considered the key required to be entered during the authentication process to protect ownership.[9]

Two-Level FSM Architecture

A two-level finite-state machine is utilized for PUF authentication, I.P. binding, and I.C. metering. To determine FSM, PUF response and authentication are required where PUF response is the input for the first level FSM, and the key is input for second-level FSM. The first level of FSM is designed as the individual intermediate state for each unique PUF response. The FSM opens the correct intermediate state; only the perfect PUF response is desired; only one key is transmitted through second level

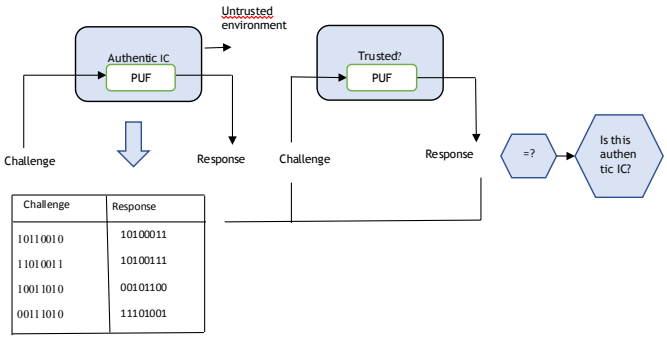

Fig.1.1: PUF-based authentication

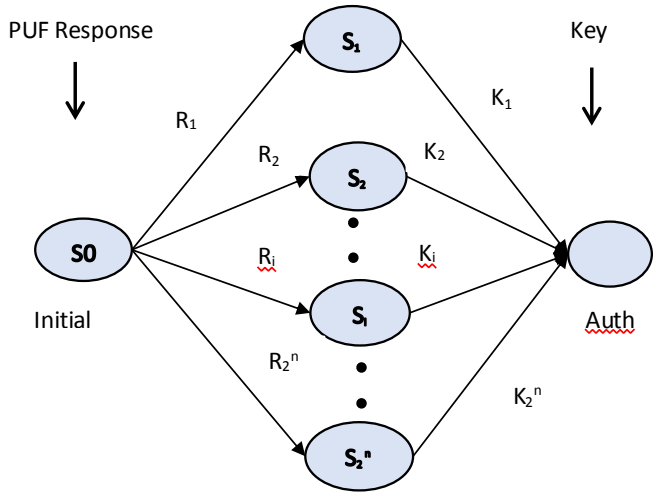

Fig.1.2: Asynchronous state machine

FSM from the intermediate state to the desired shape. In contrast, two-level FSM requires individual PUF response and key authentication pairs; they can be designed arbitrarily. All these pairs are known only to the designer, whereas the N-bit PUF response requires Intermediate states and the key will be N (size of key and length of PUF response are not required to be identical); the longer key will map to complexity in structure. Furthermore, the mappings of PUF response and keys differ from chip to chip. Figure 1.2 indicates the two-level finite state machine where PUF response is the input for the second state and key is the output.

The circuit (Fig. 1.2) is the fundamental mode circuit where flipflops are not being used. And also, the circuit contains two feedback paths and two state variables, i.e., X0 and X1, whereas feedback paths are necessary to generate latching operations needed to produce a sequential circuit. In addition, I0 and I1 are two input variables, and z is the output variable. The primary state (input state) is relay on I0 and I1, the secondary form is relay on X0 and X1, whereas the complete state is dependent on I0, I1, X0 and X1. Figure 1.3 express the present, next and stable states of the finite state machine.

PUF DESIGNS

Parallel Puf

This parallel PUF design on the left side is the initial design that there is no change made on their, right side one is our attempt at generating up more output bits, and while indigent, this method is more effective than replicated to eight times and contain eight output bits that can be reused the same challenges into each subblock that each subblock uses a group of ring oscillators which can be expected to be very random and very unique, but there is a vast hardware overhead especially in the use of more

Present total state				*Next total state*				*Stable total state*	*Output*
X1	X0	I1	I0	X1	X0	I1	I0	Yes/No	Z
0	0	0	0	0	0	0	0	Yes	0
0	0	0	1	0	1	0	1	No	0
0	0	1	1	0	0	1	1	Yes	0
0	0	1	0	0	0	1	0	Yes	0
0	1	0	0	0	0	0	0	No	0
0	1	0	1	0	1	0	1	Yes	0
0	1	1	1	1	1	1	1	No	1
0	1	1	0	1	1	1	0	No	1
1	1	0	0	0	0	0	0	No	0
1	1	0	1	1	0	0	1	No	0

Fig. 1.3 Tabular column of FSM

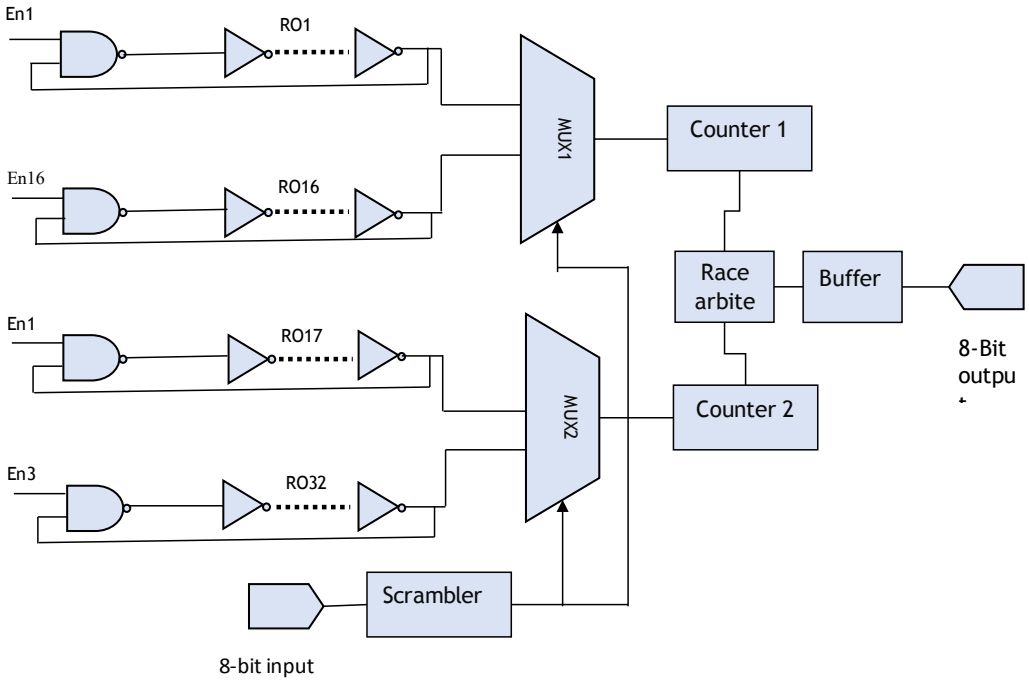

Fig. 2.1 Block diagram of parallel PUF

realistic size such as 128 or maybe 256 that the hardware cost would be very high. Fig.2.1 shows the block diagram of the parallel diagram of PUF that contains a ring oscillator connected to mux, mux output connected to counter, and race arbiter.

Serial Puf

Fig. 2.2 shows the block diagram of serial PUF, which contains scrambler and buffer in addition to parallel PUF.

The serial scheme kind of combat overhead the hardware where we added two more blocks to the parallel scheme, i.e., buffer and scrambler. Scrambler is fed in one 8-bit challenge input and generates eight serial 8-bit challenges; it cuts the hardware use because it can reuse the identical ring oscillators in the same counters and multiplexers; however, it is less random and less unique than the parallel scheme because the usage of ring oscillators and buffers collides eight sequential outputs into one group. The

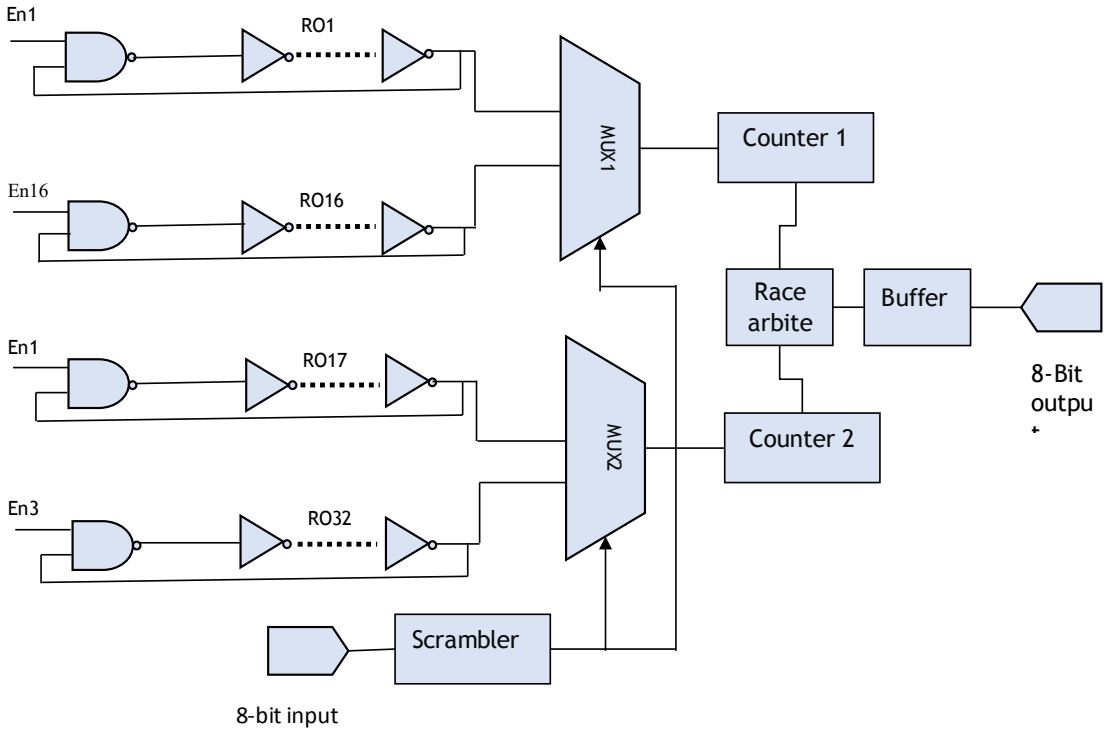

Fig. 2.2 Block diagram of serial PUF

usage of circular registers is not a good choice because it contains a lot of quotes and non-quotes dire challenges, and a counter is another option; it is a much better choice as it removes terrible challenges there's is no issues of patterns since the counter is very linear and counts from 0 to 1 to 2 and so on that's what counter does. In this linear feedback shift register(LFSR) and linear feedback add record (LFAR) is used instead of a counter; this LFAR runs through the same input space as a counter but in a pseudo-random order; however it still has adjacency issues; it just simply shuffles them around so that in next step non-linear behaviour is added to the LFAR and deemed the entire block with a scrambler, it takes the initial challenge that sent in XOR all of the LFAR output with that initial challenge to create this non-linear behaviour that no longer has any adjacency issues faced by counters.

In this Figure 2.3, two groups of 16 ring oscillators are contained on the left side, which feeds into multiplexers, and then the counters and the race referee and the buffer and right in the centre are the scramblers.

Ring Oscillator

In-ring oscillator, not gates, are to be connected in a ring to be known as a ring oscillator. The feedback signal generates a clock signal; the number of not gates in-ring should be odd (i.e., 1,3,5,7). The clock signal is process dependent, so the clock signal may get unstable to indicate buffers to drive the load. Figure 2.4 is the ring oscillator that contains three not gates (i.e., odd number)

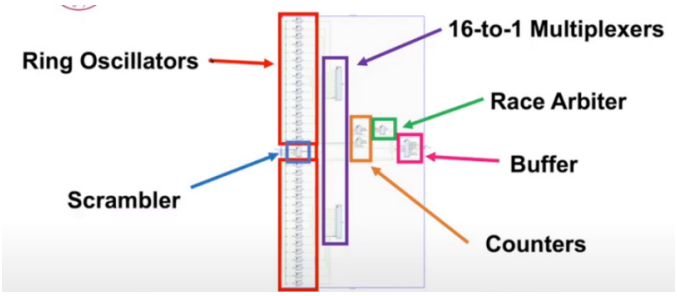

Fig. 2.3: Synthesized design of PUF

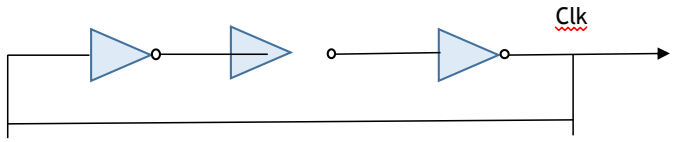

Fig. 2.4: Ring oscillator

In not gate is having propagation delay to with each of these not gates in this ring oscillator. It takes propagation delay, amount of time to have a transition from input to output. Initially, the clock is equal to zero. Zero is getting back into feedback. Then zero will get translated into one after this, not gate and translated into zero translators that will get translated into one after another, not gate. So now we see our clock will become one, but to have that transition of the clock from zero to one takes repropagation delay. If we say that the Clock signal is 0, it is one over here that will go in feedback, so we'll get to 0 over here that will get to one over here, and that will

get to 0 over here. So, to have it zero from one, it takes another three to period, so in total one clock cycle, there is propagation delay which is 6τ for this ring oscillator and operation frequency of oscillations and is 1/6τ.

A generalized formula for ring oscillator is the frequency of operations

$$f = \frac{1}{2N\tau} \quad (1)$$

Counter

The synchronous counter is the most famous because the propagation delay is minor for the synchronous counter than the asynchronous counter. Performance is also better because of the absence of glitch in the synchronous counter.

The above Figure 2.6 contains three AND, four XOR, and four D flipflops; only one pulse is given to all flipflops for every pulse; the counter counts the up-counter that starts from 0000(i.e., LSB) and counts one step up to 1111(i.e., MSB). In addition to there is a enable pin where E=0 then counter stops depending and if E=1 counter result in counting. And D flip flop works as rising the clock of the timing diagram.

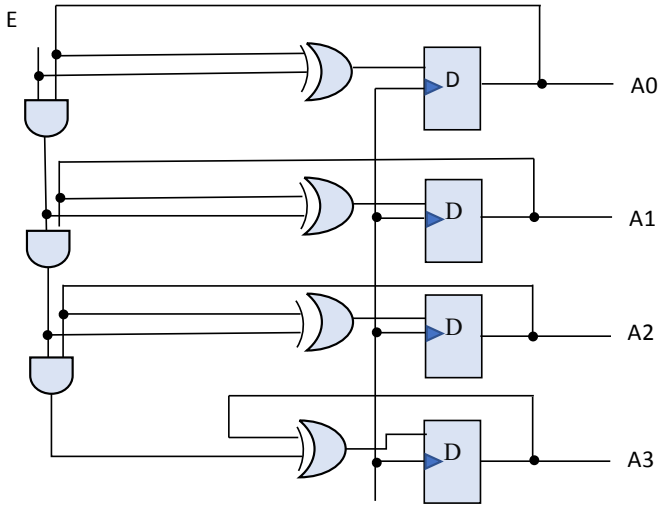

Fig. 2.6: Circuit diagram of the counter

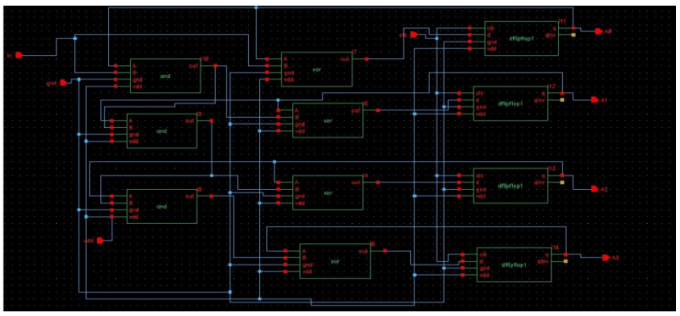

Fig. 2.7 Schematic diagram of the counter

Multiplexers

To reduce the utilize of logic gates or select data lines in a circuit, multiplexers are used. It is also used to send more than one analogue or digital signals through one transmission line at different speeds. The multiplexer is a device that converts multiple inputs into single output; there will be 2:1 multiplexer, 3:1 multiplexer, 4:1 multiplexer and n:1 multiplexer. Figure 3.7 shows the cadence design of a multiplexer that uses four NAND gates known as 3:1 multiplexer. Table 2 reveals the truth table for 3 to 1 multiplexer.

To design a multiplexer in cadence, we must first add one invertor, two AND gates, two NOR gates and four NAND gates to the virtuoso schematic editing window. Connect all four NAND gates using wires, as shown in the following Figure 2.8. Be careful at wiring, as it is an essential part of the design.

Whereas Figure 2.9 is the output of the 3:1 multiplexer, after getting the successful production of the multiplexer, we have to move to another step, i.e., arbiter.

Arbiter

Arbitrary based PUF is a strong PUF, which generates more challenge-response pairs. The basic design of the arbiter contains N number of multiplexers arranged as shown in the below Figure 1.14, and at the end, an S.R. flipflop or

Table 2: Truth table for 3:1multiplexer

A	I1	I0	Q
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

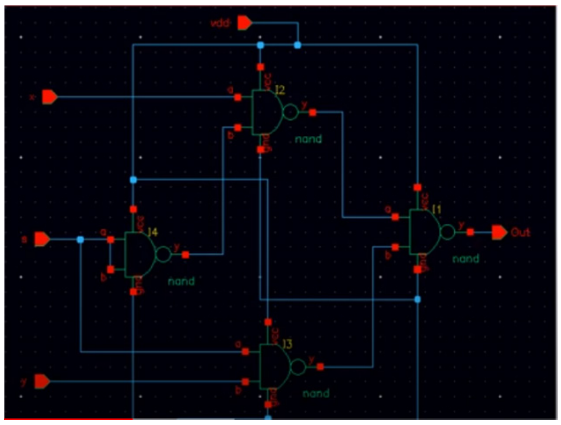

Fig. 2.8: Design of multiplexer

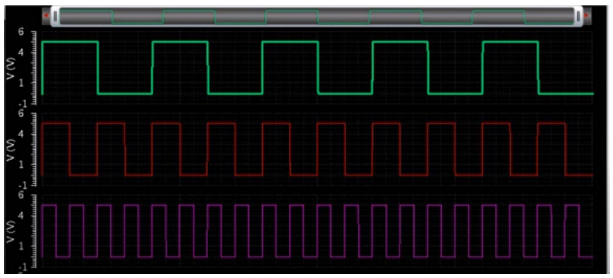

Fig. 2.9: Output response of multiplexer

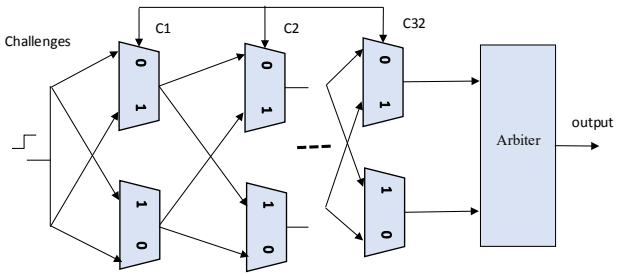

Fig. 2.10: Arbitrary PUF

a D flip flop is placed to finalize the response. The high level of input (i.e., C1, C2, C3) is applied to every switching element. The final response is evaluated by the timing diagram that reaches the arbiter at the end, and that arbiter converts the analogue signal into corresponding digital output values. The delay caused by the multiplexer is labelled as d1, d2, d3and so on, as shown in the below Figure 2.10.

A feed-forward arbiter PUF (FF PUF) is introduced by taking the PUF security into account. An arbiter is introduced between the multiplexer (i.e., mux); this arbiter in between mux acts as a switch for future stages to improve circuit complexity, uniqueness, randomness and reliability, instead of configuring the challenge-response pairs directly, configuring the PUF circuit to enhance the performance of security purpose and also secures information. Furthermore, FFO (feed-forward overlap), FFC (feed-forward cascades), FFS (feed-forward separate) are introduced using SSTA (i.e., statistical static timing analysis).

Scrambler

To encrypt our data with security at a low cost, we have to utilize a scrambler. Scramblers are the crucial components for the physical layer and highly used in VLSI designs, mainly used in data communication to secure data. Scrambler is used at the transmitter side to have accurate input. Scramblers have a lot of use in communication protocols.

Simulation Results

In this project, the two-level finite state machine is designed in cadence. For this project, the layout of the multiplexer, ring oscillator, and 4-bit counter was done. DRC was used to check errors format vs schematic is also used to check schematic and layout match. The performance of the HVT, LVT is measured with a propagation delay of various loads. Below graphs 3.1, 3.2, 3.3 shows the average power versus delay register, delay HVT, delay LVT.

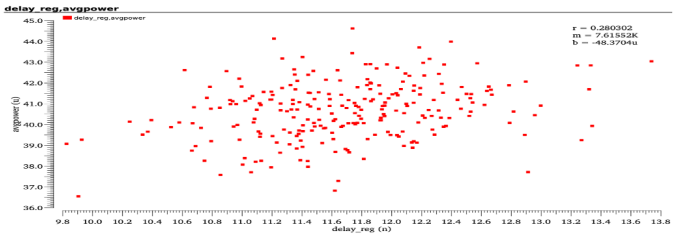

Fig. 3.1: Delay in between register and average power

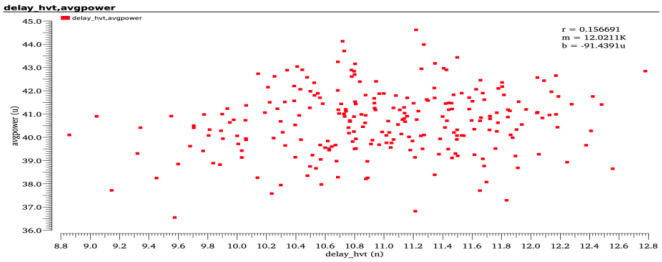

Fig. 3.2 Delay in between HVT and averag

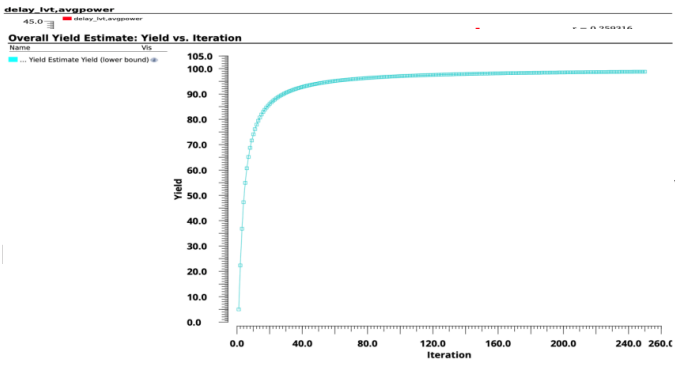

Fig. 3.4: Monte-Carlo analysis

In figure 3.4, MonteCarlo analysis involves simulating an assumed number of trials where the statistical values diverge erratically about nominal values with statistical models and probability functions. Passing and failing numbered attempts are recorded and used to estimate the yield; we plotted this using standard deviation and calculated the number of iterations to produce minimum delay. After reaching a certain iteration point, the result gets the constant value; we did this in Montecarlo analysis, where Montecarlo proposed this design using different parameters.

$$\text{Monte-Carlo estimator } G_N = \frac{1}{N}\sum_{i=1}^{N} g(X_i) \qquad (1)$$

In figure 3.5, the output sample is delayed up to 9.85n, and after one selection again, the delay has occurred. And

for the delay register, 200 values are given, whereas the mean is 11.69n and the standard deviation is 644.220p.

For HVT, Power consumption is more. In figure 3.6, HVT is delayed up to 8.85n, and the highest sample is 34.0, whereas 200 samples are given, the mean value is 10.9n, and the standard deviation is 730p.

In figure 3.7, the low voltage threshold is delayed up to 10.1n, and 200 samples are given, whereas 32 is the highest sample. The LVT mean is 12.0n and the standard deviation is 713.3p.

For figure 3.8, the average power is delayed up to 36.5, and 200 samples are given, whereas 34 is the highest sample. For average power mean is 40.66n, and the standard deviation is 1.37u.

Generally, fixing the supply voltage and output response is observed. Still, here we various supply voltages input and follow the output responses, whereas the coloured lines in the "ylv" indicate the output response of several input parameters. Using this transient response in cadence software, we can fix the supply voltage by observing various output responses, whereas we can also neglect the previous voltages.

To achieve primary PUF security, we need authentication and secure key storage. Further developments are trying to secure objectives PUFS. The former direction which is not discussed in this paper is cryptography generation, which is directly deployed on PUF as a building block. Applications of the PUF ranging from cryptographic key storage to security protocols like oblivious transfer schemes.[3]

PUFs are the best option for low-cost authentication instead of cryptographic generation, whereas data encryption. In addition, PUFs generate individual responses for each integrated circuit that are hard to predict. PUF responses are easy to store in the database compared to regenerated responses whereas, a single PUF response is insufficient to use device authentication. The PUFs contain an M number of challenge-response pairs where the responses are identical for each challenge. The authentication process is carried out during the manufacturing integrated circuits; the third party keeps the challenge-response couples in data encryption and supplies the key to the client; after the request sends by the client, these third parties send challenges to the PUF presents the response for authentication purpose. Later

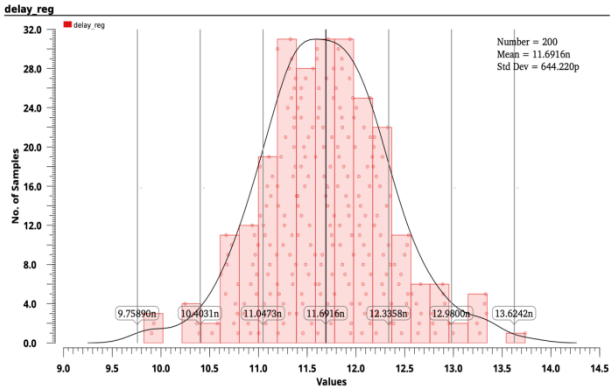

Fig. 3.5: Delay register

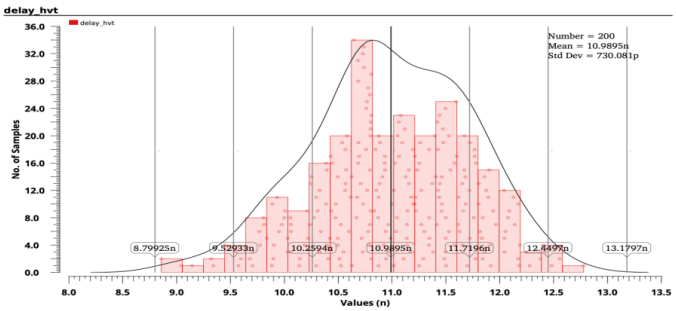

Fig. 3.6: Delay high voltage threshold (HVT)

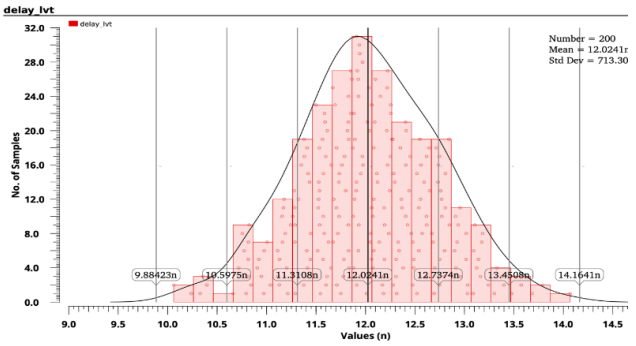

Fig. 3.7 Delay low voltage threshold (LVT)

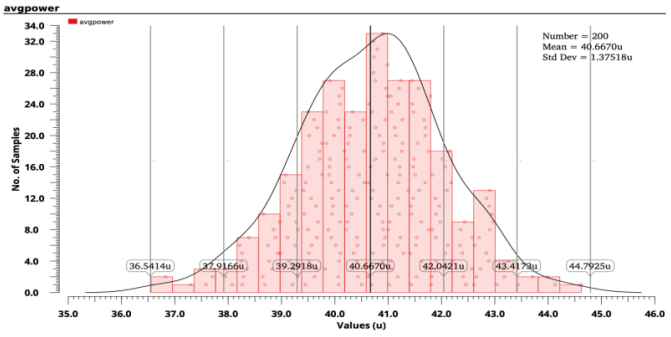

Fig. 3.8: Average power

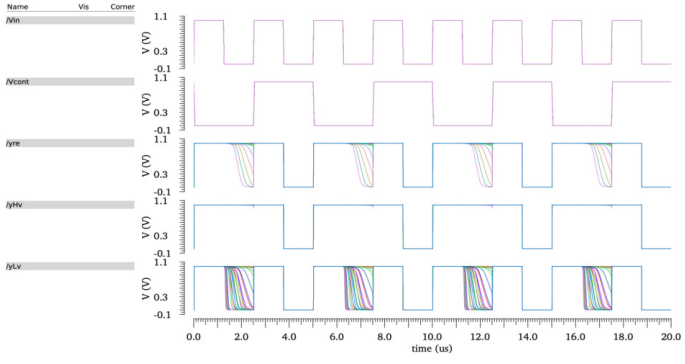

Fig. 3.9 Transient response

on, observing the results, if it matches the previously stored values, these third parties send the required key.[2]

PUFs get stability over the readouts by utilising private key generation due to temperature variations, noise, and voltage variations, whereas PUF responses are not used directly as cryptographic keys. To overcome the problem of not using PUF responses, the two-phase algorithm is implemented, i.e., key generation and key extraction phases.

Internet of things (IoT) systems are the most popular in recent decades; before sending and receiving data, devices authenticate with each other. At the same time, I.C. counterfeit mitigation to make sure secure supply chain [4]. For encryption techniques, the internet of things requires ample space to load the secret keys, it needs more power, and it has attacks like semi-invasive and invasive. Providing a massive level of security to the internet of things using tamper sensitive circuit utilizes a large amount of power and money. Whereas PUFs offer a high level of protection within low-cost authentication and without loading secret keys to the internet of things (IOT).[2]

Conclusions

This paper discussed serial physically unclonable functions (PUF), similar unclonable functions, and the design of a two-level finite state machine (FSM) using ring oscillators, scrambler, counters, and delay-based arbiter. In addition to this, we also focused on methodology, properties and applications of physically unclonable functions. Using the cadence software, we implemented our proposed design, i.e., two-level FSM, and we also designed lightweight PUF that is directly intended on integrated chips. The existing model is remote authentication and local authentication. In contrast, in the authentication model, the third party includes storing the key, and that key is provided to the client after sending unexpected challenges to the PUF. If the obstacles are not satisfied by the PUF, the key does not ship to the client. So, to overcome this problem, PUF is introduced.

In contrast, the ticket is automatically generated during manufacturing and stored in the chip, and for n number of challenge-response pairs, PUF creates a unique response for each fragment. In addition to this, by using PUF, we can get authentication at a low cost and security key generation. In recent days, PUFs are used highly in research areas, and their use in cryptographic key generation also increases. And also, one of the drawbacks of PUF is altering response due to environmental changes, and voltage variations are overcome in this paper. And applications of the PUFs are low-cost authentication and security key generation.

References

[1] Kurra, Anil Kumar, and Usha Rani Nelakuditi. "A secure arbiter physical unclonable functions (PUFs) for device authentication and identification." *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)* 7, no. 1 (2019): 117-127.

[2] Halak, Basel. "Hardware-based security applications of physically unclonable functions." In *Physically Unclonable Functions*, pp. 183-227. Springer, Cham, 2018.

[3] Aniello, Leonardo, Basel Halak, Peter Chai, Riddhi Dhall, Mircea Mihalea, and Adrian Wilczynski. "Anti-BlUFf: towards counterfeit mitigation in I.C. supply chains using blockchain and PUF." *International Journal of Information Security* (2020): 1-16.

[4] Deutschmann, Martin, Lejlalriskic, Sandra-Lisa Lattacher, Mario Münzer, Felix Stornig, and Oleksandr Tomashchuk. "Research on the Applications of Physically Unclonable Functions within the Internet of Things." (2018).

[5] Garcia-Bosque, M., G. Díez-Señorans, C. Sánchez-Azqueta, and S. Celma. "Introduction to Physically Unclonable Fuctions: Properties and Applications." In *2020 European Conference on Circuit Theory and Design (ECCTD)*, pp. 1-4. IEEE, 2020.

[6] Shiozaki, Mitsuru, Yohei Hori, Tatsuya Oyama, Masayoshi Shirahata, and Takeshi Fujino. "Cause Analysis Method of Entropy Loss in Physically Unclonable Functions." In *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5. IEEE, 2020.

[7] Lao, Yingjie, Bo Yuan, Chris H. Kim, and Keshab K. Parhi. "Reliable PUF-based local authentication with self-correction." *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 36, no. 2 (2016): 201-213.

[8] Oriero, Enahoro, and Syed Rafay Hasan. "All Digital Low Power Aging Sensor for Counterfeit Detection in Integrated Circuits." In *2018 IEEE 61st International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 33-36. IEEE, 2018.

[9] V. Vijay, et al., "Energy efficient CMOS Full-Adder Designed with TSMC 0.18µm Technology," International Conference on Technology and Management (ICTM-2011), Hyderabad, India, June 8-10, 2011, pp. 356–361.

[10] Ch. Srivalli, et al., "Optimal design of VLSI implemented Viterbi decoding," National conference on Recent Advances in Communications & Energy Systems, (RACES-2011), Vadlamudi, India, December 5, 2011, pp. 67–71.

[11] Chandra Shaker Pittala, and Vallabhuni Vijay, "Design Of 1-Bit FinFET Sum Circuit For Computational Applications," In International Conference on Emerging Applications of Information Technology, pp. 590–596. Springer, Singapore, 2021.

[12] Rajeev Ratna Vallabhuni, M. Saritha, Sruthi Chikkapally, Vallabhuni Vijay, Chandra Shaker Pittala, and Sadulla Shaik, "Universal Shift Register Designed at Low Supply Voltages in 15nm CNTFET Using Multiplexer," Lecture Notes in Networks and Systems, 2021.

[13] Chandra Shaker Pittala, J. Sravana, G. Ajitha, P. Saritha, Mohammad Khadir, V. Vijay, S. China Venkateswarlu, Rajeev Ratna Vallabhuni, "Novel Methodology to Validate DUTs Using Single Access Structure," 5th International

Conference on Electronics, Materials Engineering and Nano-Technology (IEMENTech 2021), Kolkata, India, September 24-25, 2021, pp. 1-5.

[14] B. M. S. Rani, et al., "Disease prediction based retinal segmentation using bi-directional ConvLSTMU-Net," Journal of Ambient Intelligence and Humanized Computing, 2021.

[15] Ch. Srivalli, et al., "Low power based optimal design for FPGA implemented VMFU with equipped SPST technique," National Conference on Emerging Trends in Engineering Application (NCETEA-2011), India, June 18, 2011, pp. 224–227.

[16] Vallabhuni Vijay, C. V. Sai Kumar Reddy, Chandrashaker Pittala, P ASHOK BABU, "System to Obtain Finite Gain and Noise of an Electrocardiogram Amplifier," The Patent Office Journal No. 43/2019, India. International classification: H03F3/38. Application No. 201941042674 A.

[17] Vallabhuni Vijay, C. V. Sai Kumar Reddy, Veerastu Sivanagaraju, Chandrashaker Pittala, "System for Minimizing Crosstalk Effects of Shells and Designing Multiwalled Carbon Nanotube Models," The Patent Office Journal No. 43/2019, India. International classification: B82Y10/00. Application No. 201941042460 A.

[18] Bandi Mary Sowbhagya Rani, Vasumathi Devi Majety, Chandra Shaker Pittala, Vallabhuni Vijay, Kanumalli Satya Sandeep, Siripuri Kiran, "Road Identification Through Efficient Edge Segmentation Based on Morphological Operations," Traitement du Signal, vol. 38, no. 5, Oct. 2021, pp. 1503-1508.

[19] K.H. Bindu, et al., "FINFET Technology in Biomedical-Cochlear Implant Application," International Web Conference on Innovations in Communication and Computing, ICICC '20, India, October 5, 2020.

[20] Chandra Shaker Pittala, et al., "Novel Architecture for Logic Test Using Single Cycle Access Structure," Journal of VLSI Circuits And Systems, vol. 3, iss. 1, 2021, pp. 1–6.

[21] Vallabhuni Vijay, et al., "ECG Performance Validation Using Operational Transconductance Amplifier with Bias Current," International Journal of System Assurance Engineering and Management, vol. 12, iss. 6, 2021, pp. 1173–1179.

[22] Swathi, S., et al., "A hierarchical image matting model for blood vessel segmentation in retinal images," International Journal of System Assurance Engineering and Management, 2021, pp. 1–9.

[23] Rajeev Ratna Vallabhuni, S. Lakshmanachari, G. Avanthi, and Vallabhuni Vijay, "Smart Cart Shopping System with an RFID Interface for Human Assistance," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), Thoothukudi, India, 2020, pp. 165-169, doi: 10.1109/ICISS49785.2020.9316102.

[24] Chandra Shaker Pittala, et al., "Energy Efficient Decoder Circuit Using Source Biasing Technique in CNTFET Technology," 2021 Devices for Integrated Circuit (DevIC), Kalyani, India, May 19-20, 2021, pp. 610–615

[25] Chandra Shaker Pittala, et al., "Biasing Techniques: Validation of 3 to 8 Decoder Modules Using 18nm FinFET Nodes," 2021 2nd International Conference for Emerging Technology (INCET), Belagavi, India, May 21–23, 2021, pp. 1–4.

[26] Vallabhuni Rajeev Ratna, M. Saritha, Saipreethi. N, V. Vijay, P. Chandra Shaker, M. Divya, and Shaik Sadulla, "High Speed Energy Efficient Multiplier Using 20nm FinFET Technology," Proceedings of the International Conference on IoT Based Control Networks and Intelligent Systems (ICICNIS 2020), Palai, India, December 10-11, 2020, pp. 434-443. Available at SSRN: https://ssrn.com/abstract=3769235 or http://dx.doi.org/10.2139/ssrn.3769235

[27] Manchala Sreeja, and Vallabhuni Vijay, "A Unique Approach To Provide Security For Women By Using Smart Device," European Journal of Molecular & Clinical Medicine, vol. 7, iss. 1, 2020, pp. 3669-3683.

[28] Vallabhuni Vijay, C. V. Sai Kumar Reddy, Chandrashaker Pittala, and Sonagiri China Venkateswarlu, "System for Reducing Crosstalk Delays In Electronic Devices Using A CMOS Inverter," The Patent Office Journal No. 43/2019, India. International classification: H03B5/18. Application No. 201941042515 A.

[29] Rajeev Ratna Vallabhuni, Jujavarapu Sravana, Chandra Shaker Pittala, Mikkili Divya, B.M.S.Rani, and Vallabhuni Vijcaay, "Universal Shift Register Designed at Low Supply Voltages in 20nm FinFET Using Multiplexer," In Intelligent Sustainable Systems, pp. 203-212. Springer, Singapore, 2022.

[30] Vallabhuni Vijay, Pittala Chandra shekar, Shaik Sadulla, Putta Manoja, Rallabhandy Abhinaya, Merugu rachana, and Nakka nikhil, "Design and performance evaluation of energy efficient 8-bit ALU at ultra low supply voltages using FinFET with 20nm Technology," VLSI Architecture for Signal, Speech, and Image Processing, edited by Durgesh Nandan, Basant Kumar Mohanty, Sanjeev Kumar, Rajeev Kumar Arya, CRC press, 2021.

[31] Vallabhuni Vijay, C. V. Sai Kumar Reddy, Chandrashaker Pittala, "System and Method to Improve Performance of Amplifiers Using Bias Current," The Patent Office Journal No. 43/2019, India. International classification: C12Q1/6869. Application No. 201941042648 A.

[32] S.V.S Prasad, Chandra Shaker Pittala, V. Vijay, and Rajeev Ratna Vallabhuni, "Complex Filter Design for Bluetooth Receiver Application," In 2021 6th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, July 8-10, 2021, pp. 442-446.

[33] V. Siva Nagaraju, P. Ashok Babu, Vallabhuni Rajeev Ratna, Ramya Mariserla, "Design and Implementation of Low Power 32-bit Comparator," Proceedings of the International Conference on IoT Based Control Networks and Intelligent Systems (ICICNIS 2020), Palai, India, December 10-11, 2020, pp. 459-468. Available at SSRN: https://ssrn.com/abstract=3769748 or http://dx.doi.org/10.2139/ssrn.3769748.