Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 6, 2021, Pages. 2033 - 2042
Received 25 April 2021; Accepted 08 May 2021.

PUF Based Proof-Carrying Approximate Circuits

Magesh.V*, Akash.M, Bharath Mooshik.V**, Pradeep Kumar.V**, Rishi Vignesh.D****

Assistant Professor-II, Dept of ECE, Velammal Engineering College, Chennai, Tamilnadu, India.

B.E, ECE, Velammal Engineering College, Chennai, Tamilnadu, India.

ABSTRACT

Providing protection without jeopardising a system's functionality and flexibility is a profoundly difficult challenge for researchers and practitioners. Approximate circuits (AxCs) trade off increased numerical precision for reduced hardware area, latency, and energy consumption. IP key vendors who want to develop such circuits must persuade customers of the quality of the resulting approximation. Propose PUF-based proof-carrying AxCs as a workaround. The provider produces an estimated IP core along with a certificate that verifies the accuracy of the approximation. The evidence certificate is submitted to the customer along with an estimated IP heart.The chip's feature is activated using the PUF answer. The user will formally validate the IP core's approximation accuracy for a fraction of the expense of conventional formal verification.

I. INTRODUCTION

Because of globalisation and outsourced offshore manufacturing, security has become a major concern for integrated circuits (ICs). The chip could be targeted at any point in its lifecycle, from overproduction during the manufacturing stage to illegal recycling after consumer disposal [1]. Furthermore, after obtaining the design by reverse engineering, a manufacturer could make extra chips by cloning during the manufacturing stage [2,3]. The state-of-the-art of IC reverse engineering has progressed to the point that the chips could be reverse engineered in a matter of weeks. There are firms that specialise in reverse engineering modern industrial chips. As a result, a scheme is required to avoid the reverse engineering of illegally cloned chips.

Obfuscation of hardware is a way of avoiding IC piracy and reverse engineering. There are two forms of hardware obfuscation: logic or practical locking and camouflage. The core concept behind logical locking obfuscation is that at the design period, a portion of the design is replaced with a configurable module. The chip would not work properly if the module is not triggered by the designer [4–7]. The chips can be unlocked during the post-fabrication initialization phase in a trustworthy design house by unlocking the obfuscated feature with a hidden key burned into on-chip fuses. After that, the activated chips can be sold on the open market. Without direct access to the on-chip fuses, such as poking attacks, the stored key cannot be retrieved. As a result of the obfuscation, an intruder cannot reverse engineer the specification, and the chip cannot be overproduced without knowledge of the key. In addition, layout-level tactics such as cell camouflage [8] and dummycontacts can be used to defend against attackers. In the camouflage method, the arrangement of regular cells with various functionalities is rendered to look similar. When using automatic picture software to locate camouflaged gates,

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 6, 2021, Pages. 2033 - 2042
Received 25 April 2021; Accepted 08 May 2021.

camouflaging will make it more difficult. We concentrated on logic locking-based hardware obfuscation in this article.

Several conceptually novel and intriguing ways to introducing obfuscation or logic locking to avoid reverse engineering and piracy have been introduced. In combinational architectures of configurable bits that must be set correctly to enable the chips, for example, additional exclusive OR (XOR) gates may be inserted [6]. In sequential architectures, dummy states may be introduced into finite state machines, requiring complex input sequences in order for the circuits to operate properly [9]. The circuit can only operate with a right key that will configure the interconnection network if the interconnection network is scrambled using a permutation-based technique [10]. Sections of the circuit have been concealed in a configurable lookup table (LUT) in other research [11]. The attempt for an attacker to locate the right key would become computationally infeasible with such obfuscation techniques [12]. These approaches are based on the premise that there is no direct access to the main content.

The fact that the key isn't unique in all instances of the chip is a flaw in many of these methods. As a result, if an intruder can obtain the key by any means, it can open all chips and essentially overproduce them. Physically unclonable functions (PUFs) have recently been suggested as a mechanism for producing unique keys for obfuscation [14–16]. PUFs are built-in circuit primitives that use input challenges and unpredictable output responses to derive randomness from a system's physical characteristics during the manufacturing stage [17,18]. PUFs are simple to set up, but their random nature makes their actions difficult to predict and model for an attacker.

Some of these PUF-based obfuscation techniques necessitate a complete characterization of the PUF input-output response, which necessitates a limited PUF output room. Poor PUFs are PUFs that fall under this group. Until sending the chips to the design house, an untrusted foundry may conduct the same characterization and store the challenge-response pairs (CRP) for all chips. If a key from an unlocked chip leaked, the untrustworthy foundry may use the leaked key to retrieve the entire design and, using its PUF characterizations, unlock all chips, whether approved or not.

Strong PUFs, on the other hand, have a huge input/output space, making characterization impractical and thus making them far more stable. Around the same time, it renders certain PUF-based obfuscation methods ineffective. To boost the protection of the solution, an obfuscation scheme that can take advantage of these heavy PUFs would be perfect. PUF-based proof-carrying AxCs are proposed in this article. The provider produces an estimated IP core along with a certificate that verifies the accuracy of the approximation. The evidence certificate is submitted to the customer along with an estimated IP heart. The chip's feature is activated using the PUF answer.

The remainder of the paper is structured as follows: section 2 describes the articles' associated work, and section 3 discusses the work's suggested methodology. In section 4, the effect and discussion are illustrated in depth with screenshots. Finally, in section 5, the article is brought to a close.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 6, 2021, Pages. 2033 - 2042
Received 25 April 2021; Accepted 08 May 2021.

II. RELATED WORKS

Indrasish et al. (2013) built accurate and compact mathematical models to estimate the response of FPGA-based ring oscillator PUFs using concepts from evolutionary computation, especially genetic programming (RO-PUFs). By designing novel countermeasures and threats.

Muhammad Yasin et al. (2018) advanced the state-of-the-art of logic locking. We present SARLock and SFLL, two logic locking strategies that provide quantitative security guarantees against SAT, elimination, and estimated attacks.

A portable hardware implementation of the SM3 hash algorithm was proposed by Tianyong Ao et al (2014). Instead of shift registers, which are widely found in hardware implementations, an SRAM is used to perform message expansion, and the values of the AH and V0V7 registers are changed serially as they are initialized and updated.

Anirudh Iyengar et al. (2014) developed a new PUF based on spintronic Domain Wall Memory concepts (DWM). Due to process variations caused surface roughness of the nanowire, conventional DWM is constrained by pinning.

Physical Unclonable Functions were defined by Helena Handschuh et al. and are based on intrinsic properties derived from devices and objects for the purpose of recognition. When compared to other projects.

Noor Ahmad Hazari et al. (2019) introduced an XOR-Inverter based ROPUF with increased uniformity, uniqueness, and bit-aliasing.

Adam Duncan et al. (2018) suggested a new approach for producing additional hardware protection in already-fabricated and packaged consumer off-the-shelf (COTS) system-on-a-chip (SoC) integrated circuits (ICs).

LHPUF, a lightweight and hybrid PUF developed by Sriram Sankaran et al (2018) for improved protection in IoTs, is a lightweight and hybrid PUF. LHPUF enhances security by combining the security functions of Arbiter and Ring Oscillator PUFs.

Physical unclonable functions (PUFs) are intrinsic properties derived from systems and objects for the purpose of recognition, according to Handschuh et al. (2012). SRAM is a special form of silicon PUF (static RAM)

In the Internet of Things, Kulkarni et al (2020) introduced FPGA-based Hardware Security for Edge Devices. Both smart objects and computers are linked to the internet in IoT apps. The Internet is the least secure medium for data transfer and is particularly vulnerable to cyber-attacks.

On the FPGA, Satheesh et al. (2016) created a Modified RO-PUF with Improved Security Metrics. For the same architecture, getting two PUF circuits with the same characteristics is almost impossible. PUFs take advantage of uncontrollable random process variance that occurs during IC processing. Ring oscillator (RO) PUF is the most flexible PUF, since it compares the frequencies of ring oscillators to generate the PUF answer.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 6, 2021, Pages. 2033 - 2042
Received 25 April 2021; Accepted 08 May 2021.

Rührmair et al. (2013) looked at the proper adversarial attack paradigm as well as the actual security of those protocols. We describe and compare various attack models in the first section. They range from a basic, initial setup known as the "stand-alone, successful PUF model" to more complicated scenarios such as the "weak PUF model" and the "PUF re-use model."

Mahmoud Khalafalla et al. (2019) published an in-depth study of modelling attacks against double arbiter PUFs using deep learning (DL) techniques (DA-PUFs). DL findings show improved prediction accuracy of the attacked PUFs, compared to more traditional machine learning approaches like logistic regression and help vector machines, stretching the limits of simulation attacks to smash more complex architectures.

Wenjie Che et al. (2017) used hardware data obtained from 45 Xilinx Zynq FPGAs that incorporate a Hardware-Embedded Delay PUF named HELP to examine security metrics such as Entropy, uniqueness, and randomness.

Becker et al. (2015) investigated the security of IBS when used with a k-sum PUF, as suggested at CHES 2011. Since the expectation of equal and separately distributed answers does not hold for a k-sum PUF, the term "leaked bits" was coined at CHES 2011 to describe the security of such structures. We demonstrate that the entropy of the key obtained is substantially lower than predicted, based on a refined analysis using hamming distance characterization and machine learning techniques.

Jiang et al. (2017) suggested a hardware architecture for cerebellar models that uses estimated circuits with a limited region and low power consumption. Approximate adders and multipliers are cautiously tested for deployment in an adaptive filter dependent cerebellar model to obtain a good tradeoff in precision and hardware use, using the cerebellum's intrinsic error tolerance.

III. PROPOSED METHODOLOGY

IC piracy is a major security threat in this article, where malicious manufacturers may generate unauthorized extra chips and/or steal design information through reverse engineering attempts. As a countermeasure, hardware obfuscation schemes typically replace a portion of the architecture (which then becomes the "key") with configurable modules. Enforcing the filling in of the configurable module with the concealed key details allows for post-manufacturing activation of each authenticate chip, but the possibility of a compromised common key must be mentioned. Physically Unclonable Functions (PUFs) have been suggested to be combined with hardware obfuscation and even used for generating a license key to ensure that each chip has a unique key. Since the designer must thoroughly define the PUFs for all the chips in order to uniquely set the key (the content of the configurable module) for each chip, such a paradigm is restricted to use weak PUFs. We argue in this paper that a powerful attacker in the place of a producer will thoroughly characterize all the weak PUFs and crack the obfuscation mechanism using some leaked key. This paper proposes a PUF-based hardware obfuscation scheme that generates a license key to effectively deter IC theft even though the key from an active chip is leaked.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 6, 2021, Pages. 2033 - 2042
Received 25 April 2021; Accepted 08 May 2021.

The suggested scheme uses two components to generate a specific key per chip: Key1, which is the content of the Collection bits, and Key2, which is the content of the LUT. Create the worst-case scenario, in which an attacker obtains a copy of the whole card, including Key1 and Key2, for a particular chip. Obviously, this key cannot be used to explicitly open other chips. In order to recover the master key, the intruder must examine the CRP space of the subset of the PUF used by the vendor by the leaked chip. After that, we'll go over some of the possible attacks in this scenario:

1) Chip PUF characterization in the open market: Using a leaked key Key1 and Key2 from a chip, an attacker can decide which subset of the PUF is used by the manufacturer for that chip (from analyzing Key1). Characterizing the PUF is no longer possible since all of the PUFs' characterization channels were deleted at the end of the activation step.

2) PUF characterization of chips during the manufacturing stage: The intruder has access to the characterization networks during the manufacturing stage. However, since the chips have not yet been activated, the intruder is unable to use the "leaked key" to determine which subset of the PUF will be used. Because of the large CRP room of strong PUFs, exhaustively examining all CRPs for even a single PUF is prohibitively costly. It is also prohibitively costly for a manufacturer to execute machine learning attacks on all fabricated chips when a stable strong PUF is used, as discussed in section II-B.

3) SAT-based attacks: In the absence of a clear method of obtaining the CRP space of the PUFs, the attacker will simulate the entire protection block (Obfuscator, PUF, and LUT) with a simulated LUT, and then attempt to deduce the content of that LUT by adding carefully constructed primary inputs to a working chip and evaluating the values of the primary outputs. The trick to overcoming such SAT-based attacks is to carefully pick the withheld feature during the design stage, resulting in highly correlated LUT outputs. The suggested scheme in this paper can be paired with a number of SAT-based preventive schemes to form a more comprehensive structure. Furthermore, the designer will increase the amount of q dummy fan-outs fed to the Obfuscator (as described in section IV) to reduce the expense of SAT-based attacks by exponentially raising the size of the simulated LUT at a linear cost.

To deter piracy and overbuilding attacks, we use the configuration of OCs of obfuscated nature to communicate with the PUF response in order to generate a chip-dependent license and provide the pay-per-device licensing service. Without knowing the OCs' key, an intruder cannot compute the right license to open the pirated/overproduced chips. As a result, only the designer has the authority to grant the chip's activation license.When the chip is turned on, the PUF answer is XORed with the license to create the right OC setup, which is then stored in the flip-flops to enable the chip. When the chip is turned on, the PUF answer is XORed with the licence to produce the right OC key bits, which are then stored in flip-flops to open the chip. The obfuscated gate-level netlist can be collected by RE, but the extracted netlist does not include the key bits.

The chip's feature is activated using the PUF answer. Since the PUF output is difficult to keep perfectly constant due to noise or other causes of physical instability, the designer often computes the error correcting code (ECC) to compensate for any bit flips to the PUF output (response). The overhead of applying PUFs and ECC approaches is

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 6, 2021, Pages. 2033 - 2042
Received 25 April 2021; Accepted 08 May 2021.

not mentioned in this brief. The costs of introducing PUF and ECC are readily accessible in recent literatures that have been summarized. The mistake has been fixed. The PUF response is used to activate the chip's operation; if the PUF response is incorrect, the function will not work properly. As a result, the circuit is kept locked until the proper license is obtained. It's worth mentioning that issued licenses should be made public, and that different PUF responses can be used to measure different licenses.

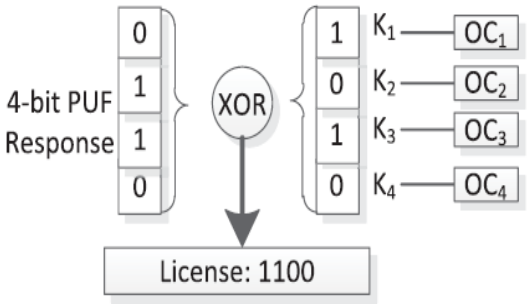


Figure 1: Obfuscation using PUFs and the development of licenses.

Figure shows an example of creating a license to demonstrate the core concept of our method. In the case of the four OCs shown in Figure, the primary bits of the OCs are OC1–OC4 and K1–K4. Assuming K1–K4 = 1010, the OC can be used to replace or add wires to either inverter. Assume that the performance value of the PUF is 0110. The 4-bit PUF output 0110 should be XORed with a 4-bit license that will produce the result 1010 to probably enable the chip (in this case, the license should be 1100). With the measured license and the PUF answer, the chip can be correctly activated. On each powered IC, nonvolatile on-chip memories will be used to store the PUF challenges, the license, and the related ECC bits. From then on, the IC will automatically read the PUF challenge and ECCs and use them to open the chip once it started up.

IV. RESULT AND DISCUSSIONS

Modalism and xilinx12.1 were used to replicate and synthesize the proposed circuit, respectively. Figures 2 and 3 display the structure simulation effects as well as waveforms. The results of actual and planned area use are shown in table 1. Figures 4 and 4 provide a description of the current and planned schemes. Finally, figure 6 depicts the project's success map.

Figure 2 Output without PUF

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 6, 2021, Pages. 2033 - 2042
Received 25 April 2021; Accepted 08 May 2021.

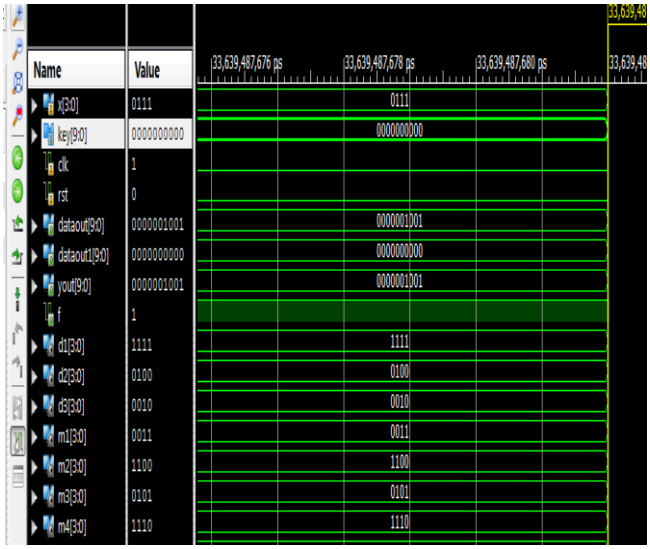
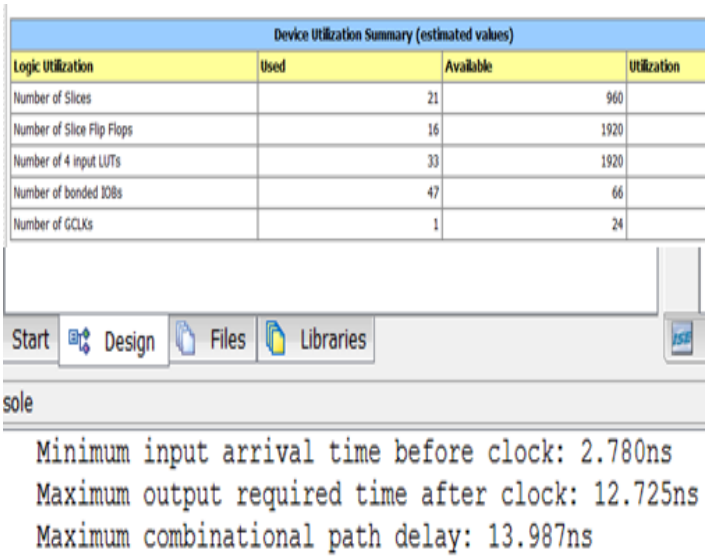


Figure 3 Output with PUF



Device Utilization Summary (estimated values)

Logic Utilization	Used	Available	Utilization
Number of Slices	21	960	
Number of Slice Flip Flops	16	1920	
Number of 4 input LUTs	33	1920	
Number of bonded IOBs	47	66	
Number of GCLKs	1	24	

Start Design Files Libraries

sole

Minimum input arrival time before clock: 2.780ns
Maximum output required time after clock: 12.725ns
Maximum combinational path delay: 13.987ns

Figure 4 existing synthesis report

parameter	existing	proposed
slices	21	19
LUT	33	31
IOB	47	27
time(ns)	13.98	12.07

Table 1 comparison table

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 6, 2021, Pages. 2033 - 2042
Received 25 April 2021; Accepted 08 May 2021.

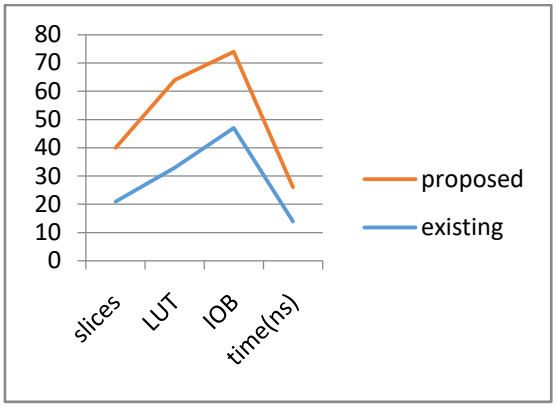


Figure 6 performance chart

V CONCLUSION

In this paper, a PUF-based proof-carrying AxCs was successfully designed. Producers of approximated IP cores were able to have structured guarantees for their circuits' error bounds using this innovative concept, allowing customers to validate these error bounds without having to trust the manufacturer or transmission networks. Consumers built confidence at a fraction of the computational effort required for complete verification. With bench mark filter architecture, we verified a proof-carrying AxCs and experimentally checked the method.

REFERENCES

1. P. Yellu, M. R. Monjur, T. Kammerer, D. Xu and Q. Yu, "Security Threats and Countermeasures for Approximate Arithmetic Computing," 2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC), Beijing, China, 2020, pp. 259-264, doi: 10.1109/ASP-DAC47756.2020.9045385.
2. T. Song, Y. Xue and D. Wang, "An Algorithm of Large-Scale Approximate Multiple String Matching for Network Security," 2006 First International Conference on Communications and Networking in China, Beijing, 2006, pp. 1-5, doi: 10.1109/CHINACOM.2006.344838.
3. M. Gao, Q. Wang, M. T. Arafin, Y. Lyu and G. Qu, "Approximate computing for low power and security in the Internet of Things," in Computer, vol. 50, no. 6, pp. 27-34, 2017, doi: 10.1109/MC.2017.176.
4. P. Yellu, Z. Zhang, M. M. R. Monjur, R. Abeysinghe and Q. Yu, "Emerging Applications of 3D Integration and Approximate Computing in High-Performance Computing Systems: Unique Security Vulnerabilities," 2019 IEEE High Performance Extreme Computing Conference (HPEC), Waltham, MA, USA, 2019, pp. 1-7, doi: 10.1109/HPEC.2019.8916503.
5. H. Martin, L. Entrena, S. Dupuis and G. Di Natale, "A Novel Use of Approximate Circuits to Thwart Hardware Trojan Insertion and Provide Obfuscation," 2018 IEEE 24th International Symposium on On-Line Testing And Robust System

Design (IOLTS), Platja d'Aro, 2018, pp. 41-42, doi: 10.1109/IOLTS.2018.8474077.

6. W. Liu, C. Gu, M. O'Neill, G. Qu, P. Montuschi and F. Lombardi, "Security in Approximate Computing and Approximate Computing for Security: Challenges and Opportunities," in Proceedings of the IEEE, doi: 10.1109/JPROC.2020.3030121.
7. M. Ye, X. Feng and S. Wei, "Runtime Hardware Security Verification Using Approximate Computing: A Case Study on Video Motion Detection," 2019 Asian Hardware Oriented Security and Trust Symposium (AsianHOST), Xi'an, China, 2019, pp. 1-6, doi: 10.1109/AsianHOST47458.2019.9006675.
8. Gupta and K. Suneja, "Hardware Design of Approximate Matrix Multiplier based on FPGA in Verilog," 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2020, pp. 496-498, doi: 10.1109/ICICCS48265.2020.9121004.
9. G. S. Rodrigues, J. Fonseca, F. Benevenuti, F. Kastensmidt and A. Bosio, "Exploiting Approximate Computing for Low-Cost Fault Tolerant Architectures," 2019 32nd Symposium on Integrated Circuits and Systems Design (SBCCI), Sao Paulo, Brazil, 2019, pp. 1-6.
10. C. Li, D. Sengupta, F. S. Snigdha, W. Xu, J. Hu and S. S. Sapatnekar, "Special session: a quantifiable approach to approximate computing," 2017 International Conference on Compilers, Architectures and Synthesis For Embedded Systems (CASES), Seoul, 2017, pp. 1-2, doi: 10.1145/3125501.3125511.
11. S. Hashemi and S. Reda, "Generalized Matrix Factorization Techniques for Approximate Logic Synthesis," 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE), Florence, Italy, 2019, pp. 1289-1292, doi: 10.23919/DATE.2019.8715274.
12. S. Lee, L. K. John, and A. Gerstlauer, “High-level synthesis of approximate hardware under joint precision and voltage scaling,” in *Proc. Design, Autom. Test Eur. Conf. Exhibiting (DATE)*, Mar. 2017, pp. 187–192.
13. Jiang, H., Liu, L., & Han, J. (2017). *An efficient hardware design for cerebellar models using approximate circuits. Proceedings of the Twelfth IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis Companion - CODES ’17.* doi:10.1145/3125502.3125537
14. H. Saadat and S. Parameswaran, "Special session: hardware approximate computing: how, why, when and where?," 2017 International Conference on Compilers, Architectures and Synthesis For Embedded Systems (CASES), Seoul, 2017, pp. 1-2, doi: 10.1145/3125501.3125518.
15. Murugan, S., Jayarajan, P., & Sivasankaran, V.Majority Voting based Hybrid Ensemble Classification Approach for Predicting Parking Availability in Smart City based on IoT.
16. Efficient Contourlet Transformation Technique for Despeckling of Polarimetric SyntheticApertureRadarImageRobbi Rahim, S. Murugan, R. Manikandan, andAmbeshwarKumarJ. Comput. Theor. Nanosci. 18, 1312–1320(2021)

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 6, 2021, Pages. 2033 - 2042
Received 25 April 2021; Accepted 08 May 2021.

17. Eye blink controlled virtual keyboard using brain sense B kavitha vp, janani meganathan, sreehoshini j, mounika International research journal of engineering and technology (irjet) 7,08-2020
18. Cryptosystem design based on Hermitian curves for IoT security OA Alzubi, JA Alzubi, O orgham, M AlsayyedThe Journal of Supercomputing 76 (11), 8566-8589