International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

|| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512| || Volume 9, Special Issue 2, October 2020 ||

3rd International Conference on Emerging Trends in Science, Technology and Mathematics [ICETSTM 2020]

18th September 2020

Organized by

Department of Computer Science, Parvathy's Arts and Science College, Dindigul, Tamilnadu, India

A Novel S-Box Encryption Using Adaptive Composite Field Architecture (ACFA) For 5G Communications

[1]JAYASHREE CHAVAN, [2]Dr.AMIT JAIN

[1]Research Scholar, Department of ECE, Sunrise University, Alwar, Rajasthan, India

[2]Research Guide, Department of ECE, Sunrise University, Alwar, Rajasthan, India

ABSTRACT: Recently, lightweight cipher has become a hot topic, which is a key technology to ensure the security of communications among constrained devices such as WSNs (Wireless sensor Network), Radio Frequency Identification (RFID), and IoT (Internet Of Things). Lightweight cryptography is a class in cryptography that is employed in resource constrained devices like embedded systems to provide security. A novel S-BOX (Substitution–Box) encryption using Adaptive Composite Field Architecture (ACFA) for 5G communication is proposed in this research. This ACFA based S-Box encryption is one of the light weight cryptography algorithms which suits the resource constrained environments in providing security in 5G communication thus enhancing the privacy. This encryption approach employs a novel ACFA architecture with 4-stage pipelining for two S-boxes employment in the first stage of creating the confusions with Algebraic Normal Form (ANF) then key generating stage is used as another step in the encryption algorithm. The generated key can be combined with plain texts and the total process can be performed for a specific number of rounds according to the number of bits in the plain texts and the algorithm defined. The proposed S-box architectures can achieve low area than conventional approach.

KEYWORDS: Substitution–Box (S-Box), Adaptive Composite Field Architecture (ACFA), lightweight encryption, Key generation.

I. INTRODUCTION

In today's world as there is too much dependence on Internet of Things (IoT) for daily activities, there is a necessity to provide security in the operation that these perform. The IoT is comprised of resource constrained devices like sensors, embedded systems, RFID tags etc. These do not possess much of the resources so the security functions also need to be incorporated in them along with their intended purpose [1].

The classical encryption algorithms are suitable in computer systems but in the resource constrained environments newly developed class of algorithms called light weight ciphers are used which provide security as well as consume less resources. In order to provide the security the original message signal called the plain text has to be transformed into another form called cipher text. The main intention of changing the plain text is because the original message must not be accessible for the attackers. The method involved in transforming the plain text to cipher text uses many transpositions and substitutions so that it is protected from the attack [2].

The S-boxes are employed where this is the first stage of creating the confusions. The key generating blocks is another step in the encryption algorithm. There are many ways in the generation process. The key will be generated and will be combined with plain texts. There are specific numbers of rounds in the process according to the number of bits in the plain texts and the algorithm defined [3]. Cryptography is the science of sending a message to another person or party in a way that only the recipient can read the message. Traditional cryptography focus on the solutions in providing high levels of security, ignoring the requirements of constrained devices. Lightweight cryptography is a research field that has developed in recent years and focuses in designing schemes for devices with constrained capabilities in power supply, connectivity, hardware and software [4]. Schemes proposed include hardware designs, which are typically considered more suitable for ultra constrained devices, as well as software and hybrid implementations for lightweight devices. ACFA is known to be highly efficient in hardware implementations. Our implementations are based on novel serialized architectures in the data processing block. ACFA achieves enough immunity against known attacks and

|| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512| || Volume 9, Special Issue 2, October 2020 ||

3rd International Conference on Emerging Trends in Science, Technology and Mathematics [ICETSTM 2020]

18th September 2020

Organized by

Department of Computer Science, Parvathy's Arts and Science College, Dindigul, Tamilnadu, India

flexibility for efficient implementation in both hardware and software. It is reported that ACFA is highly efficient particularly in hardware implementations. Therefore in this paper ACFA based approach is used for implementing the novel S-Box confusion stage for lightweight encryption architecture.

II. LITERATURE SURVEY

Syed Farid Syed Adnan et al. [5] presented "*Timing Analysis of the Light-weight AAβ Encryption Scheme on Embedded Linux for the Internet of Things*". Here, Author present an analysis of lightweight asymmetric encryption, the AAβ (AA-Beta) .that may be feasible in the internet of thing IoT. 99% improvement in encryption time and improvement of 94% on decryption time for 2048-bit primes. The authors, Nouha Oualha et al. [6] in "Light-weight Attribute-based Encryption for the Internet of Things" proposed CP-ABE scheme using effective pre-computation technicality. The key concept behind pre-computation technicality is to pre-compute and cache set pairs collected with commonly exorbitant cryptographic operations. Pre-computation techniques based on the generator, the preprocessing algorithm of the generator is executed by the hardware devices or trusted authority. The pre-computation technique reduces the cost of (CP-ABE) encryption, the pre-computation technicality used less computation and less energy drain than original schema.

Kurniawan Nur Prasetyo ST et al. [7] presented "*An Implementation of Data Encryption for IoT Using Blowfish Algorithm on FPGA*". The author presented a Blowfish algorithm is executed on Field Programmable Get Array (FPGA) by use very high speed integrated circuit hardware description language (VHDL)it is a programming language. Using field programmable get array (FPGA) implementation is simple to implement, cheap, high speed, and reprogrammed. Decrease total encryption time, give better throughput and not affective avalanche effect significantly. A. Bogdanov et.al [8] had described a PRESENT- ultra-lightweight block cipher. They gave equal importance to both security and Hardware design during the design of the cipher and at 1570 GE the hardware that is needed for Cipher is competitive with current leading compact stream ciphers. But it consumes more area because of more number of implementations.

Mohamad Sbeiti et.al [9] explored the performance of the PRESENT block cipher on FPGAs and also they had provided the implementation results of an efficiency that is, throughput per slice and differentiated them with other block ciphers. Though this Cipher is well suited for high-speed and high-throughput applications, it consumes more power. F. Macé et.al [10] had explored the performance of scalable encryption algorithm in current field-programmable gate array (FPGA) devices which is initially designed for software implementations in smart cards, controllers, or processors. But its performance is low when compared to other algorithms.

III. NOVEL S-BOX ENCRYPTION USING ACFA

The ACFA that is implemented in our work considered to take into account 128-bit length of plain text at a time and the key which is of 128-bit in length. The 128-bit cipher ACFA is known to be highly efficient in hardware implementations.

|| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512| || Volume 9, Special Issue 2, October 2020 ||

3rd International Conference on Emerging Trends in Science, Technology and Mathematics [ICETSTM 2020]

18th September 2020

Organized by

Department of Computer Science, Parvathy's Arts and Science College, Dindigul, Tamilnadu, India

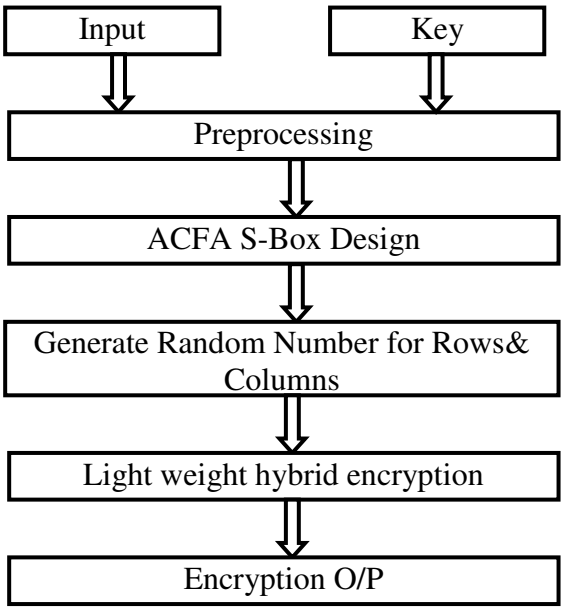

Fig. 1: Framework of ACFA based S-box Encryption

3.1 Preprocessing

ACFA-128 is divided into two parts: the data processing part and the key scheduling part. The data processing part employs a 4-branch Type-2 generalized Feistel network with two parallel F-functions F0 and F1 per round. The number of rounds r for CLEFIA-128 is 18. The encryption function ENCr takes a 128-bit plaintext $P = P_0|P_1|P_2|P_3$, 32-bit whitening keys $WK_i(0 \leq i < 4)$, and 32- bit round keys $RK_j(0 \leq j < 2r)$, as inputs, and outputs a 128-bit ciphertext $C = C0|C1|C2|C3$. The two F-functions F0 and F1 consist of round key addition, 4 non-linear 8-bit S-boxes, and a diffusion matrix. The construction of F0 and F1 is shown in Fig. b. Two kind of S-boxes S0 and S1 are employed, and the order of these S-boxes is different in F0 and F1. The diffusion matrices of F0 and F1 are also different; the matrices M0 for F0 and M1 for F1.

3.2 ACFA S-Box

S-box also known as Substitution box it is a basic accent of symmetric key algorithms as in Fig.4.1. It under goes substitution process and put back a small block of bits i.e the input bit of the S-box by another block of bits i.e S-box output bit. This method must be one-to-one, to ensure inverse operation (hence decryption). S0 and S1 are the two boxes which are used in ACFA. Let us study the details of both these.

S0 box: The implementation of this comprises of using other sub boxes termed as SS0, SS1, SS2 and SS3. These are random boxes. These are combined using the Galois field multiplier. The lookup table where the S0 values are defined in predefined tables is the most common way of implementing the S0 boxes. But here we have employed the Algebraic Normal Form (ANF) approach. The SS0, SS1, SS2 and SS3 are basically formulated using ANF expression and gates are used to represent it. This is advantageous compared to former technique as it gives the algebraic degree and linearity of the function.

S1 box: The process of designing the S1 box is done using the Galois Field (2^8) inversion process. The polynomial deployed to implement the S1 box is z8+z4+z3+z2+1. The technique in order to introduce the novelty in the design, we have used the Composite Field Approach (CFA).

$$b = g(f(a)^{-1}) \; if \; f(a) /= 0$$
$$b = g\,(0) \; if \; f(a) = 0$$

In the above equation b and a represents S-box output and input respectively

F0 and F1 designs: The F0 and F1 functions are defined using the 4 branch Generalized Fiestel Network (GFN) rather than the conventional 2 branch structure which has an advantage. Due to the 4-branch structure the F0 and F1 function

|| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512| || Volume 9, Special Issue 2, October 2020 ||

3rd International Conference on Emerging Trends in Science, Technology and Mathematics [ICETSTM 2020]

18th September 2020

Organized by

Department of Computer Science, Parvathy's Arts and Science College, Dindigul, Tamilnadu, India

are smaller compared with 2 branch structures which need double the size of input of 4-branch structure. The F-functions consists of the S0 and S1 boxes in addition to 4x4 diffusion matrices. After each round in the F-function the circular shift operation is performed.

Diffusion Matrices: The diffusion matrices namely M0 and M1 are contemplated for the generation of F-functions. The multiplication factors of 2, 4, 6, 8 and 10 are multiplied with the outputs obtained in the S-boxes. These multiplication factors are combined using the xor operation generated using the GFN structure. This GFN structure is used in the key generation block also. The intermediate key generation is done by applying the key and the 24 bit constant values to the GFN structure. The rest 36 bit values generated are employed to give the keys needed for the encryption process.

3.3 Random Number Generation

A pseudorandom number generator (PRNG) is an algorithm that takes a small amount of truly random bits as input and outputs a long sequence of pseudorandom bits. The initial truly random input is called the seed. PRNG to take in an initial seed and then be available to generate as many pseudorandom bits as needed on demand. To achieve this, the PRNG maintains some internal state and updates the state any time the PRNG generates new bits or receives a seed as input.

A pseudo Random Number Generator (RNG) is defined by a structure $(S, \mu, f, U\ g)$ where S is a finite set of states, μ is a probability distribution on S, called the initial distribution. A transition function $f : S \rightarrow S$, A finite set of output symbols U, An output function $g : S \rightarrow U$. Then the generation of random numbers is as follows:

- Generate the initial state (called the seed) s0 according to m and compute $u_0 = g(s_0)$
- Iterate for $i = 1, 2, 3, \ldots, s_i = f(s_{i-1}), u_i = g(s_i)$.

Generally, the seed s_0 is determined using the clock machine, and so the random variables $u_0, u_1, \ldots, u_i, \ldots$ seems "real" uniform random variables. The period of a RNG, a key characteristic, is the smallest integer p ∈ N, such that ∀n ∈ N, $u_{n+p} = u_n$. Two important statistical properties of the pseudo random number generator are uniformity and independence.

3.4 Lightweight hybrid Encryption

The design of Lightweight hybrid Encryption takes into consideration the following modifications: S-box arrangements, operation of finding the inverse in GF(2^5), affine transformation, and key expansion process. These changes allow the algorithm to operate on 128-bit plaintext input. This section presents an overview of the main components of the proposed 128-bit Lightweight hybrid Encryption cryptographic algorithm that uses a 128-bit encryption key. The Lightweight hybrid Encryption consists of ten rounds that use an expanded key generated from the initial key. The Lightweight Encryption consists of four transformation components: LSubNibble, LShiftRows, LMixColumns, and LAddRoundKey. The final round of LAES contains only three transformations components, which are LSubNibble, LShiftRows, and LAddRoundKey, as described below.

LSubNibble: This component is used to perform the simple transformation of the state matrix bits into another matrix based on the S-box lookup table. Different bits of the state matrix are transformed into different bits from the S-box lookup table. The S-box is intended to provide an invertible transformation of the state matrix entries during this component, which is inverted by the LInvSubNibble in the decryption process by the entries of the Inv S-box (see Table 6). In lightweight encryption, the S-box and the Inv S-box are reduced into 16 nibbles, indexed from 1 to 16, which is expected to improve the processing speed of lightweight encryption and provide the required confusion for lightweight applications.

LShiftRows: This component operates similar to the ShiftRows in AES, by shifting the entries of each row of the state matrix cyclically. The LShiftRows operates on a state matrix of 64 bits. The shift is made by certain offsets of 0, 1, 2, and 3 for rows 1 to 4 of the state matrix, respectively. The objective of this component is to avoid the columns of the state matrix form being linearly independent, and to enhance the overall diffusion of the encryption process. For the decryption process, the LInvShiftRows component cyclically shifts the rows of the state matrix to the right by certain offsets of 0, 1, 2, and 3 for rows 1 to 4 of the state matrix, respectively.

LMixColumns: In this component, the state matrix obtained from the LSubNibble is diffused to provide a further level of diffusion following the operation performed by LShiftRows component. The LMixColumns performs permutation at the column level of the state matrix, which provides another level of diffusion over the LShiftRows component.

|| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512| || Volume 9, Special Issue 2, October 2020 ||

3rd International Conference on Emerging Trends in Science, Technology and Mathematics [ICETSTM 2020]

18th September 2020

Organized by

Department of Computer Science, Parvathy's Arts and Science College, Dindigul, Tamilnadu, India

IV. RESULTS

The Xilinx design environment was used to implement and examine the developed algorithm. The below Fig. 2 and Fig. 3 shows the RTL schematic and technology schematic of ACFA based S-box encryption. RTL schematic is the combination of inputs and outputs. Register-transfer logic deliberation is utilized in equipment portrayal dialects (HDLs) to make elevated level portrayals of a circuit, from which lower-level portrayals and at last genuine wiring can be determined.

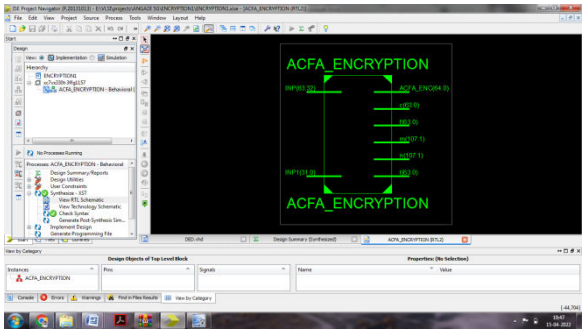

Fig. 2: RTL Schematic of ACFA based S-box encryption

Technology schematic is the combination of Look up tables, Truth Tables, K-Map and equations. The below figure (3) shows the Technology schematic of ACFA based S-box encryption. This schematic is generated after the optimization and technology targeting phase of the synthesis process. It shows a representation of the design in terms of logic elements optimized to the target Xilinx device or example, in terms of LUTs, carry logic, I/O buffers, and other technology-specific components.

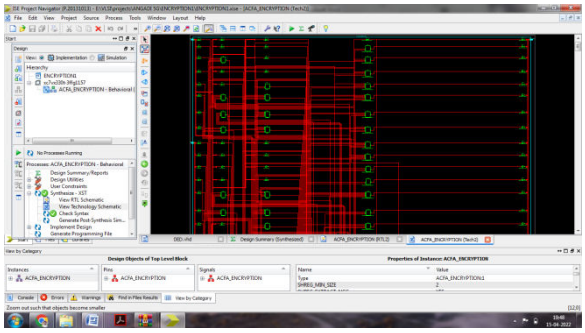

Fig. 3: Technology Schematic of ACFA based S-box encryption

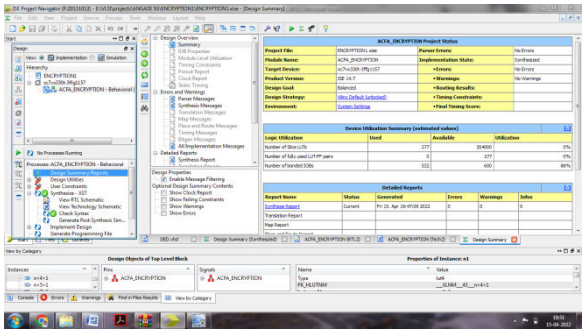

Fig. 4: Synthesis Report of ACFA based S-box encryption

|| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512| || Volume 9, Special Issue 2, October 2020 ||

3rd International Conference on Emerging Trends in Science, Technology and Mathematics [ICETSTM 2020]

18th September 2020

Organized by

Department of Computer Science, Parvathy's Arts and Science College, Dindigul, Tamilnadu, India

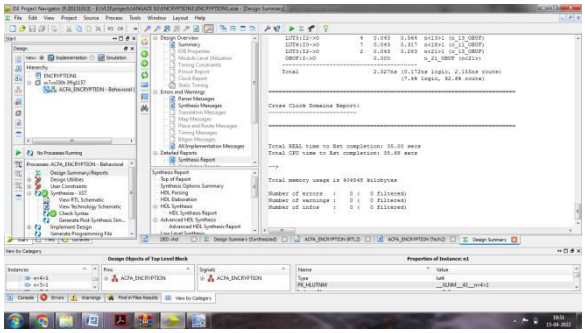

Fig. 5: Delay and Memory usage Report of ACFA based S-box encryption

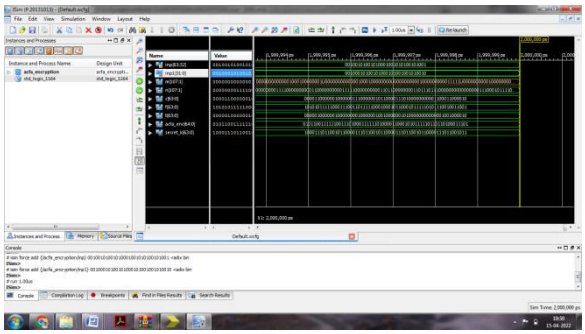

Fig. 6: Output Waveform of ACFA based S-box encryption

The following Graphs indicate the delay, memory and look Up Table (LUT) utilization for implementing the encryption. Here the ACFA based s-box encryption is compared with the AES based encryption in terms of resources utilization like delay, memory and LUTs. It can be seen that ACFA has a better performance i.e. low delay as seen in Fig. 7, less memory as in Fig. 8 and LUTs utilization as in Fig. 9 compared to the AES encryption.

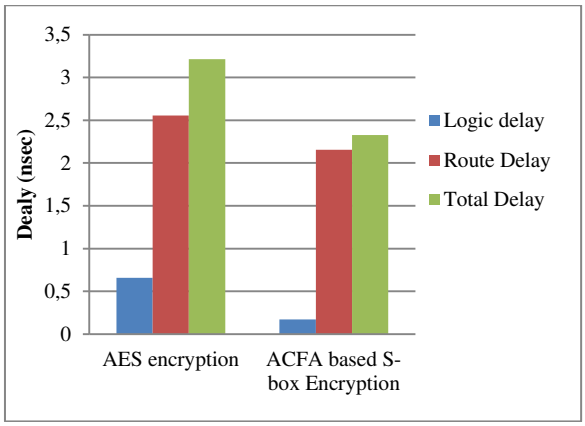

Fig. 7: Delay Comparison

|| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512| || Volume 9, Special Issue 2, October 2020 ||

3rd International Conference on Emerging Trends in Science, Technology and Mathematics [ICETSTM 2020]

18th September 2020

Organized by

Department of Computer Science, Parvathy's Arts and Science College, Dindigul, Tamilnadu, India

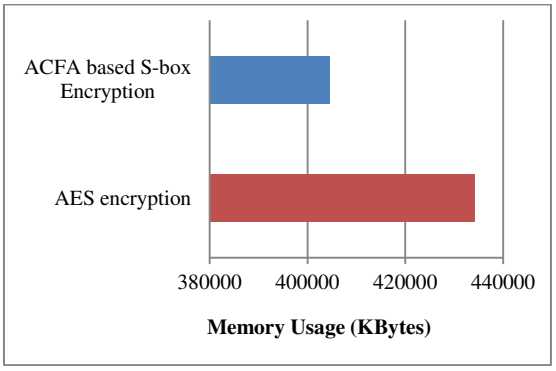

Fig. 8: Comparison of Memory Utilization

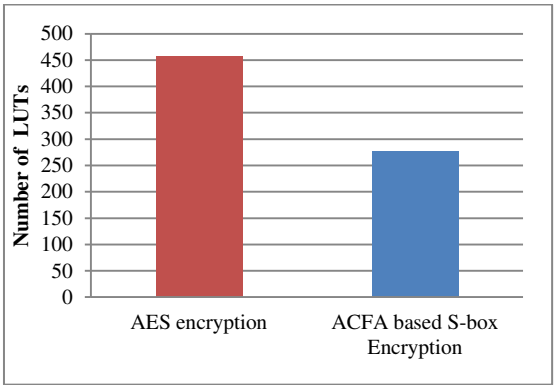

Fig. 9: Comparison of LUTs utilization

V. CONCLUSION

Lightweight cryptography is a cryptographic algorithm used for the implementation in resource limited, constrained environments including RFID tags, contactless smart cards, and sensors. The designed novel s-box encryption using ACFA architecture is implemented and compared with the conventional Encryption architecture. In this paper, very compact hardware architectures of ACFA with 128-bit keys based on 8-bit shift registers was presented. Three types of hardware architectures such as novel s-box, random number generator and lightweight encryption were implemented according to required cycles for one block process by adaptively applying clock gating technique. The ACFA architecture based s-box approach in light weight encryption algorithm can be used for the resource constrained devices. The novel sub-blocks are integrated at the top level and the implementation results are obtained. This suits the new class of light weight cryptography. This gives a good tradeoff between speed and memory requirement. From the results, it is obtained that the execution time of the new architecture is reduced as compared to the conventional architecture. It is observed that as the size of the input data increases, the throughput is increased for encryption. It is also found that the throughput of the novel architecture is enhanced.

REFERENCES

[1] Ritchell S. Villafuerte, Ariel M. Sison and Ruji P. Medina, "An Improved 3d Playfair Cipher Key Matrix With Dual Cipher Block Chaining Method", International Journal of Scientific & Technology Research, vol. 8, no. 10, October 2019

[2] R. Ueno, N. Homma, T. Iida and K. Minematsu, "High throughput/gate FN-based hardware architectures for AES-OTR", *Proc.* IEEE Int. Symp. Circuits Syst. (ISCAS), pp. 1-4, 2019.

[3] A. Subandi, R. Mieyanti, C. L. M. Sandy and R. W. Sembiring, "Three-Pass Protocol Implementation in Vigenere Cipher Classic Cryptography Algorithm with Keystream Generator Modification", Advances in Science Technology and Engineering Systems Journal, vol. 2, no. 5, pp. 1-5, 2017.

|| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512| || Volume 9, Special Issue 2, October 2020 ||

3rd International Conference on Emerging Trends in Science, Technology and Mathematics [ICETSTM 2020]

18th September 2020

Organized by

Department of Computer Science, Parvathy's Arts and Science College, Dindigul, Tamilnadu, India

[4] T. P. Berger, J. Francq, M. Minier and G. Thomas, "Extended generalized Feistel networks using matrix representation to propose a new lightweight block cipher: Lilliput", IEEE Transactions on Computers, *vol*. 65, pp. 2074-2089, 2016

[5] S. FaridSyed, M. Anuar, M. Isa, H. Hashim. "Timing analysis of the light-weight AAβ encryption scheme on embedded Linux for Internet of Things," In Computer Applications & Industrial Electronics (ISCAIE), 2016 IEEE Symposium on, pp. 113- 116. IEEE, 2016.

[6] O.Oualha, N .Nuha, K. Nguyen, "Light-weight Attribute-BasedEncryption for the Internet of Things," In Computer Communication andNetworks (ICCCN), 2016 25th International Conference on, pp. 1-6.IEEE, 2016.

[7] K. Nur, Y. Purwanto, D. Darlis. "AnImplementation of Data Encryption for Internet of Things using BlowFishAlgorithm on FPGA,"In Information and Communication Technology 2014, 2nd International Conference on, pp. 75-79. IEEE, 2014

[8] Sbeiti M, Michael S, Poschmann A and Paar C., "Design space exploration of present implementations for FPGAS", Springer, pp. 141-145, 2009

[9] Mace F, Standaert F.X, and Quisquater J.J, "FPGA implementation(s) of a scalable encryption algorithm", IEEE Trans. Very Large Scale Integr. Syst., vol. 16, no. 2, pp. 212–216, 2008

[10] BogdanovA , Knudsen L R , Leander G , Paar C, Poschmann A, Robshaw M.J. B, Seurin Y and VikkelsoeC, "PRESENT: An ultralightweight block cipher", Springer, pp. 450-466, 2007