FPGA Based Performance Evaluation of Security in Wireleess Sensor Nodes of IOT Using Lightweight Cryptography

***Srikanth Parikibandla, #Dr Alluri Sreenivas**

*** Research Scholar, #Associate Professor, Department of Electronics and Communication Engineering, GITAM Deemed to be University, Vishakhapatnam, Andhra Pradesh, India.**

***psrikanth.bt@gmail.com, #sreenivas.alluri@gitam.edu**

ABSTRACT- IoT is the Latest Technology that enables data exchange among different kinds of devices, mostly the data is transmitted through wireless network, and there are chances of hacking. Security and Privacy are the vital elements which need to be addressed to hold up to the faith of users in IoT. Consequently, cryptography algorithms are used as solution to reinforce security. Most of the currently used cryptographic solutions are predicated on the existence of ample processing power and memory. The conventional algorithms are not suitable for IoT due to its constrained environment. Therefore, lightweight cryptographic algorithm can be used as an alternative solution for IoT security issues. In Modern cryptography, the examination of lightweight symmetric ciphers has picked up enthusiasm because of the expanding interest for security benefits in constrained computing environments. However, there are number of algorithms to choose from the distinctive execution criteria and conditions; this paper presents a PRESENT Cipher model which incorporates the encryption process by using 80bit key for 64bit input data security at hardware level. The design of the model is performed by utilizing Verilog-HDL on Xilinx 14.2 ISE platform and is implemented over Spartan FPGA board. The performance analysis is performed by key generated results with respect to slices, LUT's, Flip-flops (FF's), delay and throughput as performance parameters.

KEY WORDS -- PRESENT cipher, IoT, Wireless Sensor Nodes, Lightweight Cryptography, VLSI, FPGA

I. INTRODUCTION

The Internet of Things (IoT) paradigm connects all industrial and consumer electronics manufacturers to worldwide communication networks aims to make everything intelligent, to provide smarter services and to create new products [11, 21]. The advancement of wireless technology and sensory device has brought to the introduction towards IoT [1]. Among IoT devices; most of the data is transmitted through wireless networks, in which the system includes hardware, middleware and presentation layers. The hardware layer provides actuators and sensors, middle layer gives computation with storage and presentation layer comprise interpretation tools for taking information from various base form. Most of the popular novel attacks are executed on physical layer and major attacks are performed on software layer [14]. The whole system is secured by security implementing at lowest level that is physical layer because of this the architecture of IoT schemes cannot afford for realizing the security in middle and presentation levels due to computational restraints. The IoT devices are utilizing less hardware and devices are battery powered. Therefore security solution at hardware level requires algorithm with small footprint which all comes under the energy budget [15, 16, 17]. The use of IoT devices increased all over the world because of lightweight technique used in IoT devices which also gives an end to end communication security under computation, memory limits and low power consumption [18]. Wireless sensor networks have emerges as modern day technology in IoT technology and research involving hardware system design, data management, security and social factors. The wireless sensor networks are gathering of thousands of tiny devices called sensor node, which have the capacity of detecting, processing and transmitting data in the network. Sensor networks refer to a heterogeneous system combining tiny sensors and actuators with general purpose computing elements. Sensor node is a smart, tiny, self organizing, low cast, and multi-functional device, equipped with battery, radio communication, microcontroller and sensor. There are so many factor associated with it's as Data security, operating speed, cost, efficiency and additional sensor network constraints. Main consideration is about increase the security over existing attacks without affect the performance and complexity of overall wireless sensor network [4]. The wireless sensor networks are constantly sent in threatening and inescapable condition [8]. Security

 DOI : 10.18231/2454-9150.2018.1119

is significant worry in wireless sensor network. In order to overcome security attacks cryptography techniques were used which will provide confidentiality to the data [2].

II. CRYPTOGRAPHY

Cryptography is the science security that converts the information from a readable state (Plaint text) to unreadable form (Cipher Text). It describes the cipher texts, enabling the confidentiality of communication through an insecure channel, protects against unauthorized parties by preventing unauthorized alteration of use. It uses cryptographic algorithms to transform a plaintext into a cipher text, using most of the time a key [12] called encryption and which converts Cipher text to Plaint text is called decryption. There are two types of Encryptions one is Symmetric-key cryptosystems use the same key for encryption and decryption of a message and Asymmetric-Key Cryptosystems use different keys. In this work its proposed symmetric encryption. A large portion of the present utilized cryptographic arrangements are predicated on the presence of ample processing power and memory.

III. LIGHTWEIGHT CRYPTOGRAPHY

The conventional algorithms provides Ciphers which have been known for providing optimum security have higher gate counts, which make them unsuitable for applications like IoT due to its constrained environment. There is need of lightweight ciphers for these highly constrained devices. These demands cannot be met by the majority of ubiquitous computing devices, thus there is a need to apply lightweight cryptography primitives that meet security demands when considering devices with low resources [6]. Lightweight Cryptography is one of the emerging research areas in cryptography, which covers cryptographic algorithms intended for use in devices with low or extremely low resources can be used as an alternative solution for IoT security issues [5, 6, 7]. The focus and discussion in this paper would be one of the algorithms of "Lightweight Cryptography (LWC)" and analyzing its behavior. Lightweight cryptography does not determine strict criteria for classifying a cryptographic algorithm as lightweight, but the common features of lightweight algorithms are extremely low requirements to essential resources of target devices. Considering area and energy consumption are the important measures to evaluate the LWC properties [3]. This work focuses on the design of hardware architecture for the lightweight symmetric key block cipher PRESENT which is implemented using Xilinx14.2 [5].

IV. PRESENT ALGORITHM

PRESENT is standardized in 2012 by ISO/IEC 29192-2:2012, as "one of the symmetric (the sender and receiver share a common key for both encryption and decryption) ultra-lightweight block ciphers suitable for lightweight cryptography, which are tailored for implementation in constrained environments" [10, 13]. The cipher is based on a Substitution-Permutation Network (SPN), with a round-based processing system. SPN features a 4 bit S-box and a permutation layer consisting only of bit shifts, with a block size of 64 bits and a key size of 80 bits or 128 bits and has 31 rounds [9]. Each round of Present cipher consists of three steps including a key-addition layer, a substitution layer which is a non-linear function, and a permutation layer.

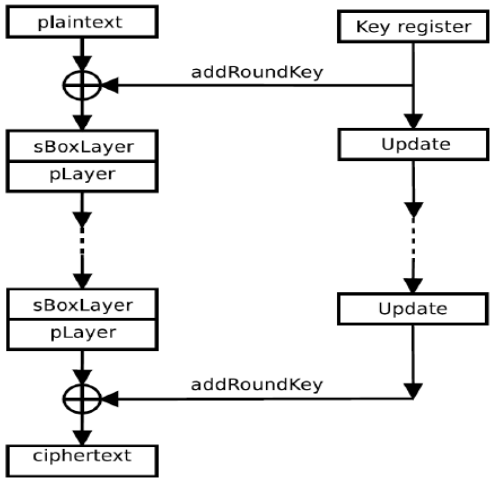

Figure 1. Round of the PRESENT algorithm

The cipher uses three basic operations to produce confidentiality over the input data:

– AddRoundKey: adds the state to a 64-bit word from the round key using finite field arithmetic.

– sBoxLayer: effectuates a 4-bit to 4-bit substitution using sixteen Substitution Box (SBOX).

– pLayer: applies bit level shifts over the state.

4.1 Substitution Layer (non-linear function (S-Box)):

The non-linear layer uses a single 4-bit S-box S which is applied 16 times in parallel in each round. So, the only nonlinear part in the PRESENT is S-box and it is called sBoxlayer. The PRESENT S-box has 4-bit input and 4-bit output and its values are in hexadecimal notation as in Table I.

Table-1 PRESENT S-box

X	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
S(X)	2	1	7	4	8	F	E	3	D	A	0	9	B	6	5	C

When the encryption algorithm is designed, the highest entropy within the substituted data is searched. In this case, the word has a size of 4 bits since it is a totally minimalist algorithm it can be implemented on a single LUT (Look Up Table) in FPGAs.

 DOI : 10.18231/2454-9150.2018.1119

4.2 *Bits Permutation: P Layer*

It is a mixing layer where a bitwise substitution is performed on 64-bit information block where the bit i of the round is moved to position P(i), the order of the substitution is shown in table 2.

Permutation P moves the i-th bit of the state to the position P(i):
P(i) = (16 i mod 63; if i != 63; 63; if i = 63)

Table -2 The PRESENT P-box (permutation box)T.

i	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
P(i)	**51**	**35**	**19**	**3**	**50**	**34**	**18**	**2**	**49**	**33**	**17**	**1**	**48**	**32**	**16**	**0**
i	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
P(i)	**55**	**39**	**23**	**7**	**54**	**38**	**22**	**6**	**53**	**37**	**21**	**5**	**52**	**36**	**20**	**4**
i	**47**	**46**	**45**	**44**	**43**	**42**	**41**	**40**	**39**	**38**	**37**	**36**	**35**	**34**	**33**	**32**
P(i)	**59**	**43**	**27**	**11**	**58**	**42**	**26**	**10**	**57**	**41**	**25**	**9**	**56**	**40**	**24**	**8**
i	**63**	**62**	**61**	**60**	**59**	**58**	**57**	**56**	**55**	**54**	**53**	**52**	**51**	**50**	**49**	**48**
P(i)	**63**	**47**	**31**	**15**	**62**	**46**	**30**	**14**	**61**	**45**	**29**	**13**	**60**	**44**	**28**	**12**

4.3 Password Expansion Function (Add-Round-Key Layer):

While the generation of the key 80-bit is done in each round using the following steps:

- Rotate left the key (80 bit) by 61 bit positions.
- Pass the left most four bits of the key [K79, K78, K77, K76] to PRESENT S-box.
- XOR between right least significant bits of round counter with five bits [K15 to K19] of key.

PRESENT may have keys going from 80 to 128 bits of length. However, this design and implementation considered 80 bit Key that will be stored in a register K of such size and will be numbered K79, K78 …K0. In each round, only the 64 most significative bits of the new calculated key will be mixed after applying the password expansion function so that the new key Ki = K79, K78.... K0, is determined by the following bit rotation:

Ki = K63 K62 ....K0 = K79 K78 ....K0

After this rotation has been performed on the input block, the following operations must be performed for each new generated subPassword Ki:

Bit rotation of the input Key:

[K79 K78 ....K0] = [K18 K17 ...K0 K79 K78…K20 K19]

Substitution using S-Box for the nibble from K79 to K76 of the password:

[K79 K78 K77K76] = *S* [K79 K78 K77 K76]

Addition or mix of nibbles K19 a K15 of the key with the round counter, through the addition on finite field GF (21) or XOR operation:

[K19 K18 K17 K16 K15] = [K19 K18 K17 K16 K15] ⊕ Round Counter Binary Value

This function allows the generation of information blocks that are useful as sub-keys starting from the system Key K. The first Nk words of this array contain the key used for encryption, since the user key is mapped to the array W while the rest of the words are generated from these first Nk words.

This function takes consecutive bytes from the sequence derived from the key expansion function and assigns them to each sub- key Ki, to form blocks of the same size than the state matrix. This means that it takes Nb * *4* bytes for each round, here Nb is 16.

The generation of the key (key expansion) for the decryption process is performed in the same way as in the encryption process. The difference is in the key selection function. In the decryption process, blocks from the key list are taken starting from the final values up to the initial ones, which is the user's personal key. This means that the last sub- key Ki used to decrypt will be the first to be used to decrypt. Hence, the encryption process has to include all key generation rounds to begin from this last key Ki, performing the reverse process until the original key is reached. The round counter must be therefore decremented and perform its mixing in each round with the previously indicated nibble [19].

V. DESIGN AND IMPLEMENTATION

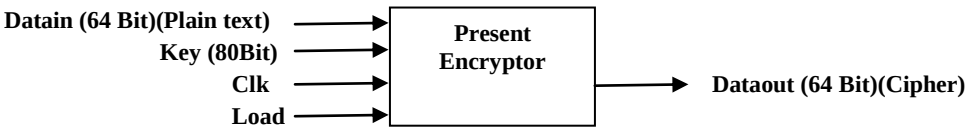

Figure 2. Schematic Diagram of PRESENT Encryptor

 DOI : 10.18231/2454-9150.2018.1119

5.1. ***Environment used for Design, Simulation and Synthesis:***

The architecture of PRESENT Algorithm mentioned in the earlier section was modeled using Verilog-HDL using Xilinx Design Suite 14.2, the FPGA device Xilinx Spartan3E, XC3S100E with Package-CP132 was used for Synthesis [20].

VI. RESULTS AND DISCUSSIONS

The plain text and Key values are taken as # is the symbol used in Verilog to denote delay which 1 time unit according to the present simulation it has been considered as 1st nano second (1ns), the presented architecture computes one round per clock cycle, initially the output will be resets at 10th nano second the data which is plain text is loaded into the encryption process it takes 32 clock cycles to complete the encryption process and delivers cipher text at 335ns, again to load the second plain text 10ns seconds will be consumed in a clock cycle from then again it takes 32 clock cycles to retrieve 2nd cipher text which is 745ns. The results are shown as below

PLAINTEXT0 64'h0000000000000000 KEY0 80'h00000000000000000000

PLAINTEXT1 64'h6954aa91c3592d27 KEY1 80'h1b2c31902a56d4a762f1

```
#10  load = 1'b1;     idat = PLAINTEXT0;     key = KEY0;
#10  load = 1'b0;
#400  load = 1'b1;    idat = PLAINTEXT1;     key = KEY0;
#10  load = 1'b0;
```

TESTVECTOR=> 335 plaintext=0000000000000000 key=00000000000000000000 cipher text=5579c1387b228445
TESTVECTOR=> 745 plaintext=6954aa91c3592d27 key=00000000000000000000 cipher text=bebc5996c52d7bc6

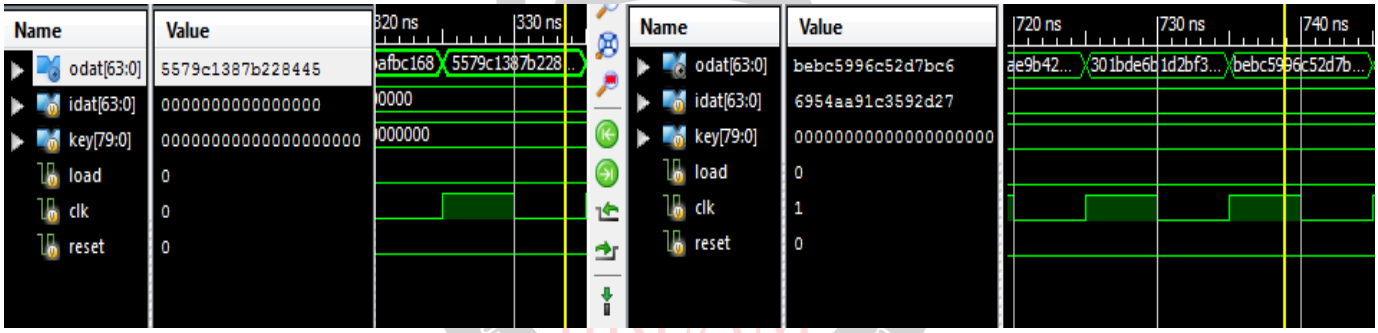

Figure 3. Waveform contains 2 ciphers according to given 2 plain texts at 330ns and 740ns.

VII. CONCLUSION

This work presented the Design and implementation of the lightweight symmetric block cipher PRESENT on FPGA which provides a secure communication in wireless sensor nodes of IoT. It designed in Verilog using a tool Xilinx 14.2 on FPGA family Spartran3E device XC3S100E with a package CP132 and Speed Grade: -5. The results obtained as Clock period: 3.804ns (frequency: 262.895MHz), Data Path: Input to output 5.663ns (4.295ns logic, 1.368ns route) (75.8% logic, 24.2% route). Number of Slices: 190 out of 960 which is 19%, Number of Slice Flip Flops: 149 out of 1920 which is 7%, Number of 4 input LUTs: 349 out of 1920 which is 18%, Number of IOs: 210. As Latency, Area, and Power are three important metrics that a VLSI designer wants to optimize. The PRESENT cipher model can be conquered the hardware design metrics like area, performance, energy and efficiency by optimizing the code in future by reducing port count with conditional inputs and using user defined primitives.

REFERENCES

[1] D. Akash, et al, **LIGHTWEIGHT SECURITY ALGORITHM FOR WIRELESS NODE CONNECTED WITH IOT**, **Indian Journal of Science and Technology,** Vol 9(30), DOI: 10.17485/ijst/2016/v9i30/99035, August 2016.

[2] Yashaswini R, et al, **WIRELESS SENSOR NETWORK SECURITY USING CRYPTOGRAPHY, IJARCST 2016, Vol. 4, Issue 2 (Apr. - Jun. 2016)**

[3] Citra B, et al, **A SURVEY ON VARIOUS LIGHTWEIGHT CRYPTOGRAPHIC ALGORITHMS ON FPGA,** e-ISSN: 2278-2834,p-ISSN: 2278-8735.Volume 12, Issue 1, Ver. II (Jan.-Feb. 2017), PP 54-59

[4] Mr. Bhavin N Patel, et al, **SECURE DATA TRANSFER USING CRYPTOGRAPHY WITH**

 DOI : 10.18231/2454-9150.2018.1119

VIRTUAL ENERGY FOR WIRELESS SENSOR NETWORK, **(IJETT) – Volume 4 Issue 8- August 2013**

[5] Mathumitha Venugopal, et al, **LIGHTWEIGHT CRYPTOGRAPHIC SOLUTION FOR IOT – AN ASSESSMENT,** International Journal of Pure and Applied Mathematics, Volume 117 No. 16 2017, 511-516.

[6] Piotr Książak, et al **A LIGHTWEIGHT AUTHENTICATION PROTOCOL FOR SECURE COMMUNICATIONS BETWEEN RESOURCE-LIMITED DEVICES AND WIRELESS SENSOR NETWORKS,** International Journal of Information Security and Privacy, 8(4), 62-102, October-December 2014

[7] Sergey Panasenko, et al, **LIGHTWEIGHT CRYPTOGRAPHY: UNDERLYING PRINCIPLES AND APPROACHES,** International Journal of Computer Theory and Engineering, Vol. 3, No. 4, August 2011

[8] Abhijit Patil, et al, **HYBRID LIGHTWEIGHT AND ROBUST ENCRYPTION DESIGN FOR SECURITY IN IOT,** International Journal of Security and Its Applications Vol.9, No.12 (2015), pp.85-98

[9] G. Jyothirmayi, et al, **PRESENT CIPHER ARCHITECTURE IMPLEMENTATION ON XILINX 14.3, IJEECS ISSN 2348-117X Volume 7, Issue 4 April 2018.**

[10] Suresh.H1, et al, **Lightweight Hardware Architectures for PRESENT Cipher in FPGA,** © 2018 IJEDR | Volume 6, Issue 1 | ISSN: 2321-9939

[11] Tadashi Okabe, et al, **EFFICIENT FPGA IMPLEMENTATIONS OF PRINTCIPHER, April 2016, Volume 3, Issue 4 JETIR (ISSN-2349-5162)**

[12] P Penchala Reddy, et al, **IMPLEMENTATION OF MULTI MODE AES ALGORITHM USING VERILOG,** International Journal of Engineering Research ISSN:2319-6890)Volume No.3, Issue No.12, pp : 780-785

[13] Edwar Gomez, et al, **PERFORMANCE EVALUATION OF THE PRESENT CRYPTOGRAPHIC ALGORITHM OVER FPGA, Contemporary Engineering Sciences, Vol. 10, 2017, no. 12, 555 – 567, HIKARI Ltd, doi.org/10.12988/ces.2017.7653**

[14] McKay, Kerry A, et al. **REPORT ON LIGHTWEIGHT CRYPTOGRAPHY**. US Department of Commerce, National Institute of Standards and Technology, 2017.

[15] Yalla, Panasayya, and Jens-Peter Kaps. "**LIGHTWEIGHT CRYPTOGRAPHY FOR FPGAS**." Reconfigurable Computing and FPGAs, 2009. ReConFig'09. International Conference on. IEEE, 2009.

[16] Banik, Subhadeep, et al. "**MIDORI: A BLOCK CIPHER FOR LOW ENERGY**." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2014.

[17] Karri, Ramesh, Grigori Kuznetsov, and Michael Goessel. "**PARITY-BASED CONCURRENT ERROR DETECTION OF SUBSTITUTION-PERMUTATION NETWORK BLOCK CIPHERS.**" International Workshop on Cryptographic Hardware and Embedded Systems. Springer, Berlin, Heidelberg, 2003

[18] Hengameh Delfan Azari, et al, **AN EFFICIENT IMPLEMENTATION OF PRESENT CIPHER MODEL WITH 80 BIT AND 128 BIT KEY OVER FPGA BASED HARDWARE ARCHITECTURE,** International Journal of Pure and Applied Mathematics Volume 119 No. 14 2018, 1825-1832

[19] A. Bogdanov, G. Leander, L. R. Knudsen, C. Paar, A. Poschmann, M. J. Robshaw, Y. Seurin, and C. Vikkelsoe, "**PRESENT - AN ULTRA-LIGHTWEIGHT BLOCK CIPHER**," in Proceedings of CHES 2007, ser. LNCS, no. 4727. Springer-Verlag, 2007, pp. 450 – 466. [Online]. Available: http://dx.doi.org/10.1007/ 978-3-540-74735-2 31

[20] Lara-Nino, et al, **NOVEL FPGA-BASED LOW-COST HARDWARE ARCHITECTURE FOR THE PRESENT BLOCK CIPHER**, 2016 Euromicro Conference on Digital System Design 978-1-5090-2817-7/16 IEEE DOI 10.1109/DSD.2016.46

[21] Hala Tawalbeh1, et al, **SECURITY IN WIRELESS SENSOR NETWORKS USING LIGHTWEIGHT CRYPTOGRAPHY,** Journal of Information Assurance and Security. ISSN 1554-1010 Volume 12 (2017) pp. 118-123

 DOI : 10.18231/2454-9150.2018.1119