IJFANS INTERNATIONAL JOURNAL OF FOOD AND NUTRITIONAL SCIENCES

ISSN PRINT 2319 1775 Online 2320 7876

 Volume 11, Iss 7 , 2022

Analysis of Various LWE (LWE) Algorithms using different Performance Metrics

Radhika Rani Chintala, Somu Venkateswarlu

Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India

DOI : 10.48047/IJFANS/V11/ISS7/308

Abstract. Human Sensor Networks represent a burgeoning field in which resource-constrained applications are gaining significant prominence. Traditional encryption algorithms prove inadequate for such resource-limited devices due to their constrained capabilities. Consequently, the need arises for the deployment of LWE (LWE) algorithms to ensure data security during the transmission of data from sensor devices to BSs (BS). These LWE algorithms can be implemented in software or hardware, and their performance is typically evaluated using diverse metrics. Notably, each category of algorithms exhibits distinct performance metrics. This paper undertakes an exhaustive analysis of the performance of various hardware-implemented LWE algorithms employing a variety of metrics. The metrics employed for this assessment encompass Area, Energy, Throughput, Hardware Efficiency, and MSEC. This exploration seeks to shed light on the effectiveness of these algorithms and their suitability for the resource-constrained environment of Human Sensor Networks.

Keywords: Human Sensor Network**,** Resource-constrained**,** BS, Performance metric.

1. Introduction

In tandem with the escalating utilization of Human Sensor Networks (HSNs), the magnitude of data transmission between sensor devices is experiencing a significant upsurge. HSNs represent a communication framework ideally suited for resource-constrained applications, particularly in domains such as medical applications, where they operate within the proximity of human body. This intricate network comprises diminutive sensor devices that can be affixed to a person's attire or even implanted within human body [1]. An HSN device plays a dual role, serving as either a sensor or an actuator/controller device [2]. Sensor devices are designed for sensing and gathering a myriad of health parameters from human body, such as blood glucose, BP, temperature, heart rate, etc. In contrast, actuator/controller devices are responsible for processing the data collected by the sensors and initiating specific actions based on sensed data. Following the collection of data, it is then transmitted to a BS device, which typically takes the form of a handheld device like a PDA or a smartphone. To ease effective communication amongst the sensors and with the BS, sensor devices are provided with built-in radio capabilities. The BS predominantly undertakes the tasks of processing and forwarding the aggregated data to remote servers via the internet for further analysis and processing. In this manner, HSN devices enable continuous monitoring of an individual's health, offering valuable feedback to both the user and medical personnel.

The security requirements of a network vary depending on the specific application and the nature of the data being processed. For instance, in certain scenarios, an application may prioritize data integrity and accuracy over data confidentiality. In such cases, the primary concern is ensuring that the data received by other devices is correct and was not tampered, rather than keeping it secret. Many conventional cryptographic standards were originally designed with a tradeoff that optimized security, performance, and resource requirements for desktop and server environments. However, this optimization often renders them difficult or even impossible to implement on resource-constrained devices. Even when implementable, their performance may not meet acceptable standards in such environments. Consequently, lightweight cryptographic methods have been introduced to address the myriad challenges

ISSN PRINT 2319 1775 Online 2320 7876
 Volume 11, Iss 7 , 2022

posed by conventional cryptography in these contexts. These challenges encompass limitations associated with processing speed, physical dimensions, energy consumption, and memory constraints. Over the past decade, there has been a proliferation of lightweight cryptographic algorithms tailored to the unique demands of resource-constrained devices. In the realm of hardware implementations, the key considerations for lightweight cryptography revolve around chip size and energy consumption. Conversely, for software implementations, the critical factors are code size and RAM usage. Despite their lightweight nature, these cryptographic algorithms still provide a level of security that is deemed sufficient for safeguarding data in resource-constrained environments.

2. Literature Survey

Cazorla et al. [3] conducted a comprehensive study that involved an analysis of 17 distinct block ciphers. Their research primarily focused on 12 lightweight block ciphers, with a key emphasis on assessing memory requirements and energy consumption in relation to the CPU cycles needed. Consequently, certain algorithms exhibited a relatively high memory footprint, with a few of them consuming as much as 17K ROM. Law et al. [4] undertook an analysis of 8 block ciphers pertaining to memory requirements and CPU cycles in various modes of operation, encompassing CBC, CFB, OFB, and CTR. Hager et al. [5] have concluded that the evaluation of energy efficiency has primarily employed the parameter of CPU cycle count. L. Batina et al. [6] have demonstrated that the energy consumed may not exhibit a linear proportionality with the number of CPU cycles. James M et al. [7] introduced a hardware implementation of the standard AES 128 (Advanced Encryption Standard) block cipher using HDL (Hardware Description Language) code on Xilinx 14.2 and the results of the simulations indicated a reduction in latency, consequently leading to an improved throughput for the algorithm. Soufiane Oukili et.al [8] introduced an FPGA (Field-Programmable Gate Array) implementation of the DES (Data Encryption Standard) block cipher using the Spartan 3e version of Xilinx, and the results demonstrated a notably high encryption rate coupled with minimal hardware resource usage. Bassam Jamil Mohd et.al [9] presented an FPGA implementation of the HIGHT algorithm, employing distinct optimal designs encompassing both pipelined and scalar configurations. Their analysis revealed that scalar design implementations required less design area and power, while pipelined designs excelled in terms of low energy consumption and high throughput.

Pushpalatha GS et. al [10] introduced a hardware implementation of the IDEA block cipher on an FPGA using the Verilog HDL. The simulation happened using Xilinx ISE 14.7, and the outcomes indicated that latency had been significantly reduced to 212.5 nanoseconds, with a mere 1% of the overall device memory being utilized. Pulkit Singh et. al [11] conducted an extensive analysis of the resource utilization, power consumption, and performance characteristics of diverse hardware implementations of the KLEIN block cipher. Their findings indicated that designs employing parallel processing were demonstrated to achieve high throughput while conserving design area, making them well-suited for resource-constrained applications. Hasan M. N et. al [12] showcased the FPGA implementation of the LBlock cipher on an Altera board. Notably, in a comparative analysis with lightweight block ciphers such as Hummingbird and XTEA, LBlock exhibited superior performance on a similar FPGA platform, highlighting its efficiency and suitability for various applications. Mohammed Al-Shatari et. al. [13] detailed the hardware implementation of the round-based iterative LED (LWE Device) cipher, featuring a block size and key size of 64 bits. The simulated results demonstrated that LED achieved superior performance in both area utilization and throughput compared to alternative implementations. In the study documented in reference [14], the authors introduced hardware implementation of the mCrypton block cipher utilizing CMOS technology with a scale of 0.13µm. The implemented results underscored the suitability of the mCrypton

ISSN PRINT 2319 1775 Online 2320 7876
 Volume 11, Iss 7 , 2022

algorithm for deployment in sensor devices and RFID tags, as it demonstrated hardware resource utilization that fell within an economically feasible range.

Ayoub Mhaouch et. al [15] put forth hardware implementation of the Piccolo block cipher with keen focus on achieving a tradeoff between speed and area. This design attained 6.4 Mbps throughput, albeit with high no. of clock cycles at 496, balancing resource consumption and performance. In the work by Pandey J. G and their team [16], they introduced FPGA architectures for Present cipher, which features a 64-bit block size and keys of both 80 and 128 bits. These proposed designs were specifically tailored for applications with critical requirements in terms of both latency and resources and were implemented on a Xilinx Vertex5 platform. The findings offer insights into the performance and efficiency of the FPGA implementations for Present cipher under various scenarios. Hussain M.A et. al. [17] introduced the hardware implementation of TEA cipher on an FPGA device. The implementation was realized through VHDL coding, and the results were rigorously simulated using the Xilinx platform. Notably, this implementation achieved both encryption and decryption processes at a low overall cost.

3. Performance Analysis of LWE Algorithms

Performance analysis of various LWE algorithms is conducted, employing a range of metrics, including Throughput, Area, Hardware Efficiency, Energy and MSEC.

3.1 Algorithms with 64-bit block size

Area :

A smaller effective design area is indicative of greater energy efficiency. Specifically, when design area exceeds 3000GE, algorithm is categorized as inefficient [19]. The metric values computed w.r.t area are detailed in Table 1, and a graphical representation depicting algorithm performance in relation to area is provided in Figure 1. Among the thirteen algorithms with a 64-bit block size, the calculation of the effective area indicates that LED stands out with its minimal consumption, totaling only 1040GE, thereby classifying it as an efficient algorithm.

Table 1 Values w.r.t Area metric

Algorithm	Block Size	Area(GE) (Smaller)
DES	64	2762
DESXL	64	3082
HIGHT	64	3901
IDEA	64	3082
KLEIN	64	1528
LBLOCK	64	1320
LED	64	1040
mCRYPTON	64	2500
PICCOLO	64	1135
PRESENT	64	1704
TEA	64	2820
XTEA	64	3490
EELWE	64	1086

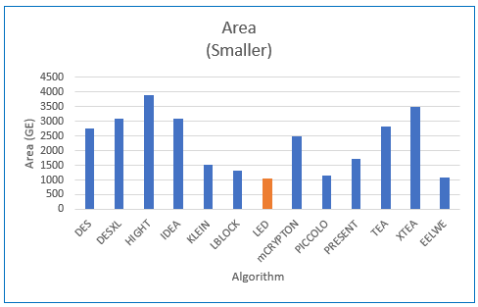

Figure 1 Performance of LWE w.r.t Area

Energy

The algorithm suitability for deployment in resource-constrained devices hinges on its energy efficiency in data processing. Reduced energy consumption not only enhances the operational longevity of such devices but also contributes to their effectiveness. Detailed metric values related to energy are featured in Table 2, while a graphical representation illustrating the algorithm performance concerning energy is provided in Figure 2. In the evaluation of energy consumption among the thirteen algorithms with a 64-bit block size, it is evident that EELWE stands out as the most efficient, having the lowest energy consumption at 0.0406 µJ/byte. This observation leads to the conclusion that EELWE demonstrates superior performance in terms of energy efficiency.

Table 2 Values w.r.t Energy consumed.

ISSN PRINT 2319 1775 Online 2320 7876
Research paper **Volume 11, Iss 7 , 2022**

Algorithm	Block Size	Energy (µJ/byte) (Smaller)
DES	64	21.8
DESXL	64	24.15
HIGHT	64	5.37
IDEA	64	9.87
KLEIN	64	7
LBLOCK	64	21.5
LED	64	0.292
mCRYPTON	64	0.506
PICCOLO	64	0.125
PRESENT	64	9.758
TEA	64	0.182
XTEA	64	0.258
EELWE	64	0.0406

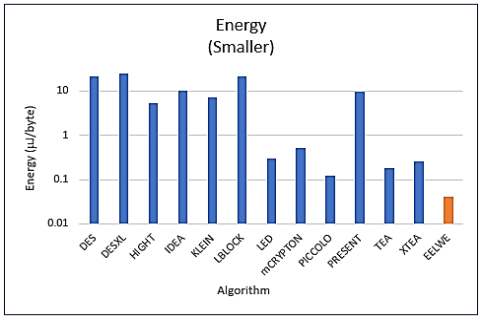

Figure 2 Performance of LWE w.r.t Energy consumed.

Throughput

An efficiency of a LWE algorithm is typically assessed based on its ability to deliver high throughput. However, in the context of resource-constrained devices, the energy exhausted by an algorithm takes precedence in determining its effectiveness. The metric values related to throughput are detailed in Table 3, and a visual representation depicting the performance of these algorithms concerning throughput is provided in Figure 3. In the assessment of throughput among the thirteen algorithms with a 64-bit block size, IDEA has demonstrated notably high throughput, clocking in at 640 kbps. This performance characteristic positions IDEA as an efficient choice for data processing.

Table 3 Values w.r.t Throughput metric

Algorithm	Block Size	Throughput (Kbps) (Larger)
DES	64	400
DESXL	64	400
HIGHT	64	200
IDEA	64	640
KLEIN	64	376
LBLOCK	64	200
LED	64	133
mCRYPTON	64	492
PICCOLO	64	237
PRESENT	64	207
TEA	64	200
XTEA	64	200
EELWE	64	356

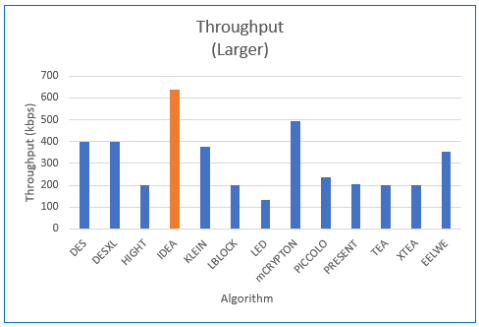

Figure 3 Performance of LWE w.r.t Throughput

Hardware Efficiency

Hardware efficiency, as measured by Throughput/Area, is a key indicator of an algorithm's effectiveness. An algorithm is thought efficient when it achieves high hardware efficiency.

Table 4 Values w.r.t H/W Efficiency

Algorithm	Block Size	Hardware Efficiency (Kbps/GE) (Larger)
DES	64	0.1448
DESXL	64	0.1297
HIGHT	64	0.0512
IDEA	64	0.2076
KLEIN	64	0.246
LBLOCK	64	0.1515
LED	64	0.1923
mCRYPTON	64	0.1968
PICCOLO	64	0.2088
PRESENT	64	0.1214
TEA	64	0.0709
XTEA	64	0.0573
EELWE	64	0.3278

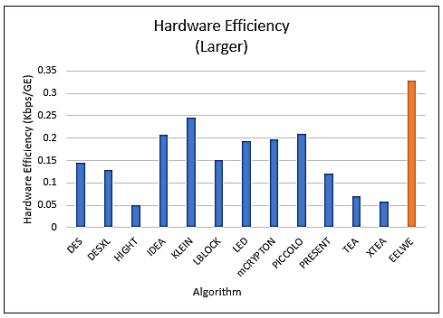

Figure 4 Performance of LWE w.r.t Hardware Efficiency

The specific metric values for hardware efficiency are outlined in Table 4, and a graphical representation illustrating the algorithms' performance with respect to hardware efficiency is depicted in Figure 4. In the evaluation of hardware efficiency among the thirteen algorithms utilizing a 64-bit block size, EELWE has demonstrated a notably higher hardware

ISSN PRINT 2319 1775 Online 2320 7876
Research paper **Volume 11, Iss 7 , 2022**

efficiency, measuring at 0.3278 kbps/GE. This observation leads to the conclusion that EELWE showcases superior performance in terms of hardware efficiency.

MSEC

MSEC serves as a metric for assessing the performance of LWE algorithms with regard to both security and energy consumption. An algorithm is deemed superior in terms of security when it generates a higher MSEC value. The specific metric values associated with MSEC are detailed in Table 5, and a visual representation illustrating the performance of the algorithms in relation to MSEC is provided in Figure 5.

Table 5 Values w.r.t MSEC metric

Algorithm	Block Size	MSEC Value (Larger)
DES	64	-0.18
DESXL	64	7.78
HIGHT	64	19.37
IDEA	64	10.84
KLEIN	64	5
LBLOCK	64	264
LED	64	1.13
mCRYPTON	64	-0.09
PICCOLO	64	71.15
PRESENT	64	116.44
TEA	64	38.46
XTEA	64	403.1
EELWE	64	928.13

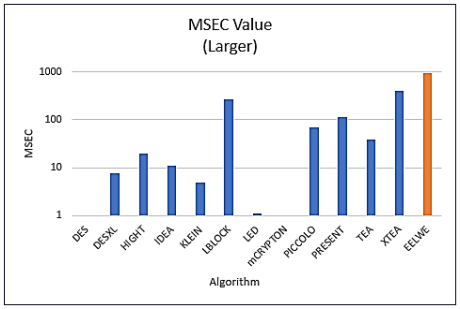

Figure 5 Performance of LWE w.r.t MSEC

A greater metric value indicates a more favorable trade-off between security and energy consumption, while a negative metric value suggests that, given the current technological context, the cipher may not be suitable for secure usage. In the calculation of MSEC values among the thirteen algorithms using a 64-bit block size, EELWE has shown a notably higher MSEC value, measuring at 928.13. This finding leads to the conclusion that EELWE's performance excels in terms of security and energy consumption trade-offs, positioning it as a superior choice in this regard.

3.2 Algorithms with 64-bit block size and 80-bit key size

Area

The calculated metric values pertaining to area are delineated in Table 6 for algorithms characterized by a 64-bit block size and an 80-bit key size and the performance of these algorithms concerning area is provided in Figure 6.

Table 6 Values w.r.t Area

Algorithm	Block Size	Key Size	Area(GE) (Smaller)
KLEIN	64	80	1528
LBLOCK	64	80	1320
PICCOLO	64	80	1135
PRESENT	64	80	1704
EELWE	64	80	1086

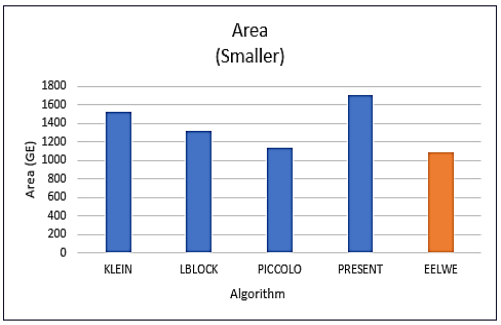

Figure 6 Performance of 64-bit LWE w.r.t Area

Upon computing the effective area among the five algorithms with a 64-bit block size and 80-bit key size, it is discerned that EELWE boasts a smaller effective area. As a result, it can be concluded that EELWE delivers superior performance in this context.

ISSN PRINT 2319 1775 Online 2320 7876
Research paper **Volume 11, Iss 7 , 2022**

Energy

The calculated metric values with respect to energy are detailed in Table 7 for algorithms characterized by a 64-bit block size and an 80-bit key size. Additionally, a visual representation illustrating the performance of these algorithms in terms of energy is provided in Figure 7. In the computation of energy consumption among the five algorithms employing a 64-bit block size and an 80-bit key size, it is evident that EELWE stands out by consuming less energy. This finding leads to the conclusion that EELWE's performance excels in terms of energy efficiency, positioning it as a superior choice within this context.

Table 7 Values w.r.t Energy consumed.

Algorithm	Block Size	Key Size	Energy (µJ/byte) (Smaller)
KLEIN	64	80	7
LBLOCK	64	80	21.5
PICCOLO	64	80	0.125
PRESENT	64	80	9.758
EELWE	64	80	0.0406

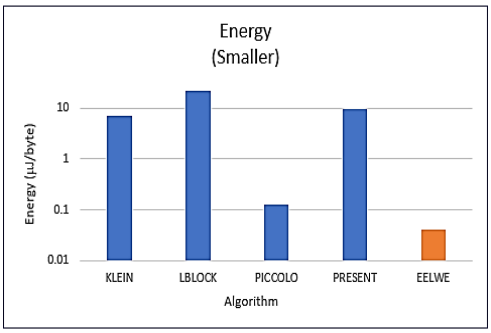

Figure 7 Performance of 64-bit LWE w.r.t Energy consumed.

Throughput

The metric values related to throughput are presented in Table 8 for algorithms featuring a 64-bit block size and an 80-bit key size.

Table 8 Values w.r.t Throughput

Algorithm	Block Size	Key Size	Throughput (Kbps) (Larger)
KLEIN	64	80	376
LBLOCK	64	80	200
PICCOLO	64	80	237
PRESENT	64	80	207
EELWE	64	80	356

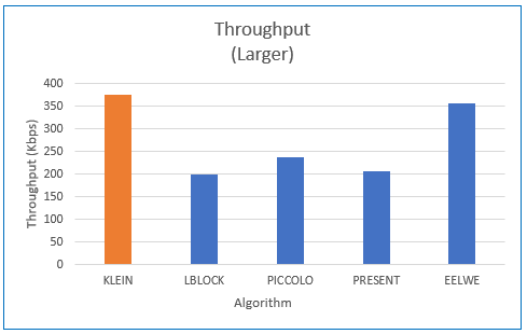

Figure 8 Performance of 64-bit LWE w.r.t Throughput

Additionally, a graphical representation illustrating the performance of these algorithms with respect to throughput is provided in Figure 8. In the calculation of throughput among the five algorithms utilizing a 64-bit block size and an 80-bit key size, KLEIN has demonstrated notably high throughput. This observation leads to the conclusion that KLEIN's performance excels, positioning it as the top-performing choice within this specific context.

Hardware Efficiency

The metric values associated with hardware efficiency are outlined in Table 9 for algorithms characterized by a 64-bit block size and an 80-bit key size. Furthermore, a visual representation depicting the performance of these algorithms in terms of hardware efficiency is provided in Figure 9. In the evaluation of hardware efficiency among the five algorithms featuring a 64-bit block size and an 80-bit key size, it is evident that EELWE stands out by demonstrating larger hardware efficiency. This observation leads to the conclusion that EELWE's performance excels in terms of hardware efficiency, positioning it as a superior choice within this specific context.

ISSN PRINT 2319 1775 Online 2320 7876
Research paper **Volume 11, Iss 7 , 2022**

Table 9 Values w.r.t H/W Efficiency

Algorithm	Block Size	Key Size	Hardware Efficiency (Kbps/GE) (Larger)
KLEIN	64	80	0.246
LBLOCK	64	80	0.1515
PICCOLO	64	80	0.2088
PRESENT	64	80	0.1214
EELWE	64	80	0.3278

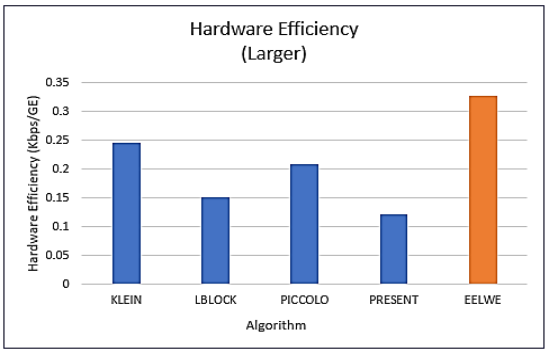

Figure 9 Performance of 64-bit LWE w.r.t Hardware Efficiency

MSEC

The calculated metric values associated with MSEC are presented in Table 10 for algorithms characterized by a 64-bit block size and an 80-bit key size. Additionally, a visual representation illustrating the performance of these algorithms concerning MSEC is provided in Figure 10. In the assessment of MSEC values among the five algorithms with a 64-bit block size and an 80-bit key size, it is evident that EELWE has demonstrated a notably larger MSEC value. This observation leads to the conclusion that EELWE excels in terms of performance and security trade-offs, positioning it as a superior choice within this specific context.

Table 10 Values w.r.t MSEC

Algorithm	Block Size	Key Size	MSEC Value (Larger)
KLEIN	64	80	5
LBLOCK	64	80	264
PICCOLO	64	80	71.15
PRESENT	64	80	116.44
EELWE	64	80	928.13

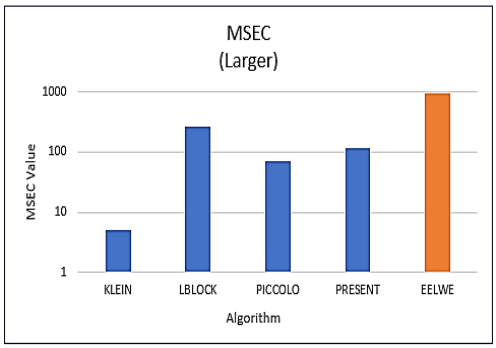

Figure 10 Performance of 64-bit LWE w.r.t MSEC

4. Conclusion

This paper undertakes an extensive performance analysis of various LWE algorithms. The study focuses on algorithms with identical block and key sizes, namely 64-bit blocks and 80-bit keys. Among these algorithms, the EELWE algorithm has consistently demonstrated superior outcomes in terms of Area, Energy efficiency, Hardware Efficiency, and MSEC. Although there is a minor variance in throughput when compared to the KLEIN algorithm, the comprehensive evaluation leads to the unequivocal conclusion that EELWE is the most favorable LWE algorithm for ensuring data security within the context of Human Sensor Networks.

References

[1]. C. Buratti, A. Conti, D. Dardari, and R. Verdone, “An overview on wireless sensor networks technology and evolution,” Sensors, vol. 9, no. 9, pp. 6869–6896, 2009.

[2]. HamidrezaSalarian, Kwan-WuChin, FazelNaghdy, “Coordination in wireless sensor–actuator networks: A survey”, Journal of Parallel and Distributed Computing, Vol. 72, No. 7, pp. 856-867, 2012.

[3]. Mickael Cazorla, Kevin Marquet and Marine Minier, “Survey and benchmark of lightweight block ciphers for wireless sensor networks”, LNCS, SECRYPT, 2013.

[4]. Law Y. W., Doumen, J., & Hartel, P., “Survey and benchmark of block ciphers for wireless sensor networks”, ACM Transactions on Sensor Networks, Vol. 2, No. 1, pp. 65–93, 2006.

ISSN PRINT 2319 1775 Online 2320 7876
 Volume 11, Iss 7 , 2022

[5]. C. T. R. Hager, S. F. Midkiff , J. min Park, and T. L. Martin, "Performance and energy efficiency of block ciphers in personal digital assistants," in In Proceedings of the 3rd IEEE International Conference on Pervasive Computing and Communications (PerCom 2005), pp. 127–136, IEEE Computer Society, 2005.

[6]. L. Batina, A. Das, B. Ege, E. B. Kavun, N. Mentens, C. Paar, I. Verbauwhede, and T. Yalcin, "Dietary recommendations for lightweight block ciphers: Power, energy and area analysis of recently developed architectures." Cryptology ePrint Archive, Report 2013/753, 2013.

[7]. James M, & Kumar D. S, "An Implementation of Modified Lightweight Advanced Encryption Standard in FPGA", Procedia Technology, Vol. 25, pp. 582–589, 2016.

[8]. Soufiane Oukili, Seddik Bri, "High Throughput FPGA Implementation of Data Encryption Standard with Time Variable Sub-Keys", Int. Journal of Electrical and Computer Engineering, Vol.6, No.1, 2016.

[9]. Bassam Jamil Mohd, Thaier Hayajneh, Zaid Abu Khalaf and Khalil Mustafa Ahmad Yousef, "Modeling and optimization of the lightweight HIGHT block cipher design with FPGA implementation", Security and Communication Networks, Vol. 9, No. 13, pp. 2200-2216, 2016.

[10]. Pushpalatha G S , Harshitha N G , Rashmi C , Rashmi P K , Preksha S, "Performance Analysis of Idea Algorithm on FPGA for Data Security", International Journal for Research in Applied Science & Engineering Technology (IJRASET), Vol. 7, No. 5, pp. 2635 – 2638, 2019.

[11]. Pulkit Singh, B. Acharya, R. K. Chaurasiya, "High Throughput Architecture for KLEIN Block Cipher in FPGA", Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON), 2019.

[12]. Hasan M. N., Hasan M. T., Toma R. N., Maniruzzaman M, "FPGA implementation of LBlock lightweight block cipher", 3rd International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), 2017.

[13]. Mohammed Al-Shatari, Fawnizu Azmadi Hussin, Azrina Abd Aziz, Gunawan Witjaksono, Mohd Saufy Rohmad , Xuan-Tu Tran, "An Efficient Implementation of LED Block Cipher on FPGA", International Conference of Intelligent Computing and Engineering (ICOICE), 2019.

[14]. C. Lim and T. Korkishko, "mCrypton - A lightweight block cipher for security of low-cost RFID tags and sensors," in Information Security Applications, vol. 3786 of LNCS, pp. 243–258, Springer Berlin Heidelberg, 2006.

[15]. Ayoub Mhaouch, Wajdi Elhamzi, Mohamed Atri, "Lightweight Hardware Architectures for the Piccolo Block Cipher in FPGA", International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), 2020.