Energy Efficient Low Resource Consumption Lightweight Block Ciphers for Internet of Things

Prabhat Kumar, S. Suresh

Abstract: Internet of Things (IoT) refers the system of interrelated heterogeneous uniquely identifiable devices with the ability to collect data and communicate with each other to enable applications/services. The security and privacy are the two major issues as IoT applications are providing personalized services by collecting users' private, sensitive and confidential data. The traditional cryptographic methods are not appropriate for most of the IoT applications due to resource-constraints. The lightweight block ciphers are the most common solution to provide security and privacy in IoT. In this paper, various popular lightweight block cipher algorithms and their performance metrics are discussed in detail. The comparative performance analysis of the block ciphers are investigated through experimental study in terms of various metrics and their results are presented. Based on the study, we made recommendations to choose a suitable cryptographic algorithm for an IoT environment depending upon the desired level of security and privacy requirements. The detailed directions are also given which can be considered as guidelines for designing new security solutions.

***Index Terms**: cryptography, Internet of Things, lightweight block ciphers, resources constraint devices, security and privacy.*

I. INTRODUCTION

Today's world is moving towards what we want to what we wish. A new concept "Internet of Things (IoT)" focuses to fulfill this objective. There is no commonly accepted definition for IoT worldwide. The term IoT is defined by different authors in different ways. The most popular one is "Any uniquely identifiable things that can be connected by a network and used based on anywhere and anytime for anything" [1]. IoT is an environment where physical objects interact with the real world. The IoT-devices can be inter-connected with each other without human interaction. The communication process can be accomplished with the help of sensors, actuators, processors, and transceivers [2]. The sensor and actuators are two important devices that help to connect devices with the physical world.

The sensors are responsible to collect, store and process the data and actuator helps to take decisions according to the user's response [3].

The communications among IoT devices are mostly connectionless since these devices are located geographically at different places. Being capable of connecting anything in real-time, IoT exceeds their limit and continue to cover stranger areas such as personal gadgets, smart cities, health care, agriculture etc.

While increasing the usage of IoT applications, we must familiar with limitation in the IoT environment. Further, these limitations become a barrier to the enhancement of IoT application areas. These challenges include computational capability, storage capacity, memory size, power consumption, and energy sources [4]. The IoT devices are typically built with lightweight computational capabilities. So, it is a challengeable task to maintain tradeoff between increasing computation capability and decreasing power consumption. The storage capacity of IoT devices are very less, measured in the unit of bytes or kilobytes. The energy sources to the IoT devices are not only batteries but also include human-body heat, indoor light, electrostatic vibration, magnetic fields, etc. Due to limited power capacity, these devices are bounded with fixed durability and capacity. These limitations are needed to be addressed to increase the application areas of IoT. This requires a detailed study of existing solutions to the various IoT related security problems by considering the various limitations. On the other hand, security and privacy become a major issue as IoT devices are loosely connected.

In this paper, the detailed study of various lightweight block ciphers are presented by considering the various limitations of IoT. Based on this study, we shall endorse to choose the lightweight block cipher for an IoT environment depending upon the desired level of security requirement. The detailed discussion is also given which can be considered as guidelines for designing new security solutions. The rest of the paper is ordered as introduction of the privacy and security, lightweight block ciphers definition and their working strategies with security strength, defines performance metrics and measures the performance of each block ciphers with respect to different parameters. Finally, last section concludes the paper.

II. PRIVACY AND SECURITY

To provide personalized services, IoT devices are gathering information about users' private activities. This raises questions among users whether and how their data are secured and private. The privacy and security are two important terms that are directly proportional to the user's confidence.

Manuscript published on 30 September 2019
* Correspondence Author

Prabhat Kumar*, Department of Computer Science, Institute of Science, Banaras Hindu University, Varanasi – 221 005, India.

S. Suresh, Department of Computer Science, Institute of Science, Banaras Hindu University, Varanasi – 221 005, India.

Retrieval Number: B3742078219/19©BEIESP
DOI:10.35940/ijrte.B3742.098319
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

The users want different levels of security as per their application requirements and kind of data being collected by the applications. These terms are measured based on how much their data are secured against different types of security threats. Privacy helps to protect user identification and security protects against unauthorized data accessing.

The privacy is psychological belief and its definition varies on human being [5]. It is a fundamental right for the human being. *"Any unauthorized capturing of any human's personal information such as thoughts, objectives, opinion or some other issues, makes violation of privacy"* [6]. Information privacy was defined by Westin in 1968 as *"The right to select what personal information about me is known to what people"* [7].

The privacy always gets respect by well-defined security policies. Data processing and analysis are as important as how analyzed data can be kept secured. With the increasing use of IoT technologies, the developer and designers should be committed to embedding well-defined security solutions against threats with IoT applications [8]. The privacy and security are such incredible terms that can be used to win the user's trust by IoT developers and designers.

III. LIGHTWEIGHT CRYPTOGRAPHY IN IoT ENVIRONMENT

The encouraging of IoT technology evolution, security threats are also analogue increasing. The basic constraints with IoT embedded devices are limited resources. So, this is a challengeable task for the designer & developer to implement security techniques with limited resources. Having to win the trust of users, it is necessary to provide security against different types of threats.

The lightweight cryptography algorithms [9] are designed with the consideration of low implementation cost and low power consumption. These algorithms are also helping to keep the tradeoffs between limited resources and the desired level of security. The privacy and security threats are not limited to only the IoT environment as parallelly exists in other communication environments. The reason for using a lightweight algorithm is that there are limited resource constraints and limited energy sources. Depends on the applications' requirements, the suitable lightweight algorithms may be selected by considering the performance metrics [10].

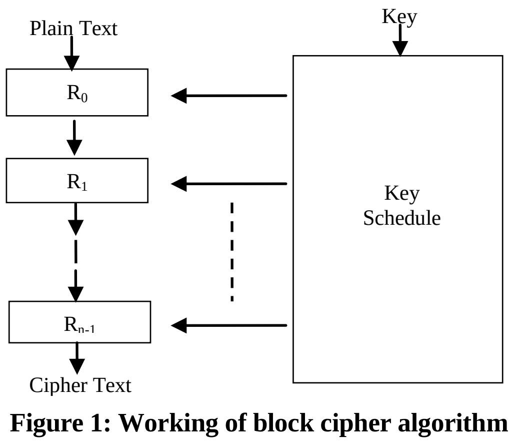


Figure 1: Working of block cipher algorithm [11].

Working strategies of lightweight block cipher algorithm are illustrated in figure 1. The process to convert from plain text to cipher text consists of R rounds and key scheduling. The round function uses previous round to get input N_b bits plaintext and sub-key then generates the output to the next round. The key schedule function adds a sub-key at each round to generate input for the next round. These two steps help to generate cipher text from input plain text.

The popular lightweight blocks cipher algorithms in the literature are given below.

A. HIGHT

The HIGHT (High Security and Light Weight) [12] algorithm comes under the feistel structure. This is suitable for cheaper, low-energy, ultra-light implementation and contains 64-bit block size, 128-bit key size and 32 rounds. This algorithm is software-oriented rather than hardware-orientation and can be implemented with 3048 gates. It is sufficient for low cost and power consumption constraints devices. The gate count value represents the required hardware complexity and associated power consumption. The encryption operation in the HIGHT algorithm consists of a key schedule, initial transformation, round function, and final transformation [12]. The decryption operation in the HIGHT algorithm is performed in the inverse order of encryption operation. The key schedule engenders sub keys in the inverse order. The round function in the decryption process has subtraction mod 28 in place of addition mod 28 and byte-swap with the opposite direction to that in the encryption procedure. It executes faster in 8-bit-oriented architecture and suitable for IoT application as most of the devices are having limited resources.Evaluating the security forte of HIGHT algorithm discloses that this is secured against various types of attacks such as Saturation Attack [13], Boomerang Attack [14], Interpolation [15], Higher-Order Differential Attack [16], etc.

B. LBlock

The LBlock [17] algorithm also comes under the family of the feistel structure. The block length is 64 bits and the key size is 80 bits with 32 rounds. LBlock can be implemented efficiently not only in hardware environments but also in software platforms. The encryption process can be accomplished by a 32-round iterative structure. The round function consists of confusion function S and diffusion function P, performed in each round of the encryption process. As parallel, eight 4-bit S-boxes are kept by confusion function S and illustrates non-linear layer of round function F. The permutation of eight 4-bit blocks are made by diffusion function P. The decryption process is same as encryption process but performed in inverse order [18].

The impossible differential attack [19] is cryptanalytic techniques that influence the security strength of many block ciphers algorithms and proposed by Kim et al. [20]. According to the complications of the impossible differential attack on 20-round LBlock [13], we assume that the full 32-round LBlock has enough security margins in contradiction of this attack. The integral attack [20] and Related-Key Attacks [21] are also the most powerful attacks on LBlock. To get prevention from these attacks, we have to control the amount of active S-boxes.

C. ITUBee

The ITUBee [21] lightweight algorithm be appropriate to the family of feistel structure and is a software-oriented lightweight block cipher. It consists of 80 bits block size, 80 bits key size and 20 rounds. The algorithm is especially suitable for resource-constrained devices including an 8-bit microcontroller such as sensor nodes in wireless sensor networks.

To diminish the energy consumption of the cipher, the key schedule is not preferred. The encryption process is achieved as: top and bottom of encryption algorithm, (ML || MR) and (MR || ML) are used as whitening keys respectively. MR is used as round keys for odd rounds while for even rounds ML is used, where M= 80-bit master key, ML = left half of the master key, MR = right half of the master key.

The decryption process is the similar as the encryption process excluding the decryption process of key is (KR || KL) and the round constants are used in inversed order.

The differential and linear cryptanalysis is cryptanalysis methods to measure the security strength of ITUBee block cipher and committed to count the number of active S-boxes. However, it is very difficult to count the number of active S-boxes in ITUBee block cipher. Some other attacks examine the security strength i.e. related-key differential attacks, meet-in-the-middle type attacks, Self-similarity Attacks, Impossible Differential Attacks, etc.

D. LILLIPUT

The LILLIPUT [22] comes under the feistel structure. This algorithm has 64 bits block length and 80 bits key length with 30 rounds. The encryption at each round carries basic steps are achieved to generate i.e. non-linear layer, linear layer and permutation layer. The key schedule depicted produces the 30 sub-keys RK0 to RK29 from the master key M and is designed to allow on-process computations. The decryption is fairly similar to encryption but uses block permutation and sub-keys in the reverse order.

The security forte of LILLIPUT can be investigated to distinguish a permutation obtained with LILLIPUT from a random permutation. The side-channel attacks use some special knowledge about the implementation of a cipher to break its security.

E. LED

The LED (Light Encryption Device) [23] algorithm comes under the Substitution Permutation Network (SPN) structure. This is a 64-bits block length and key length from 64 bits to 128-bits with 32/48 rounds. The encryption procedure of LED contains of four rounds of operation to generate cipher state i.e. AddConstants, SubCells, ShiftRows, and MixColumnsSerial [23]. The security study of LED is used to estimate the security strength resistant in contradiction of classical attacks and several kinds of related-key attacks. The slide attack [24] is a block cipher cryptanalysis technique that feats the amount of self-similarity of a permutation. The round-dependent constants addition property of LED generates different at all rounds which marks the slide attack unbearable to perform. In addition to these attacks, the rebound attack [25] and its upgraded variations, Super-Sbox cryptanalysis [26] and Integral attacks [27] also a major attack on the LED block cipher algorithm.

F. KELEIN

The KELEIN [28] algorithm goes to the family of the Substitution Permutation Network (SPN) structure. The block size defined in this algorithm is 64 bits and different key sizes 64/80/96 with 12/16/20 rounds.

The counter mode [29] helps to avoid implementation cost of decryption in most of lightweight block cipher but KLEIN evades implementation cost deprived of fixing on any cipher mode. Its primary emphases on software implementations but also relishes hardware competence resulting from its humble structure with an involutive S-box. The numerous key sizes of KLEIN offer suppleness and a reasonable security level for omnipresent applications.

KLEIN's security analysis is used to demonstrate its confrontation with different cryptanalytic attacks. Based on the branch number, i.e. the number of active S-boxes in a certain number of rounds, the linear and differential attacks of a block cipher make resistance. By mixing the RotateNibbles [28] and MixNibbles [28] steps, the KLEIN can reach an equilibrium between the minimum amount of active S-boxes and the software performance for resource-constrained appliances.

Additionally, KLEIN also secured against key schedule attacks, integral attack, algebraic attack, side-channel attack, etc. Summarized description of light weight block cipher algorithms are given is in the table 1.

Table 1: Summarization of Lightweight Block Ciphers

Cipher	Year	Technique	Key Size (bits)	Block Size (bits)	No. of Rounds
HIGHT	2006	Feistel network	64	128	32
LBlock	2011	Feistel network	80	64	32
LED	2011	Substitution and permutation network	64/128	64	32/48
KLEIN	2011	Substitution and permutation network	64/80/96	64	12/16/20
ITUBee	2014	Feistel network	80	80	20
Improved Lilliput	2015	Feistel network	80	64	30

IV. DESIGN PARAMETERS AND PERFORMANCE METRICS

The performance metrics are suffering from a lack of universal evolution platform to measure the performance systematically and consistently. There are several performance metrics defined by various authors to measure the performance of lightweight algorithms [11] [30].

These metrics depend on the various other factors and mostly specific to a particular platform/application. The performance metrics are separately designed for software and hardware implementation platforms. The research challenges vary for software platforms and hardware platforms.

A. Software Platform Metrics

In order to optimize the use of memory size, speed, power and energy, the software platform metrics play a significant role. The execution speed can be enhanced by minimizing the number of execution cycles i.e. reliant on instruction set architecture, coding style, and code structure. The performance metrics used to calculate the encryption time, throughput, area, power, and energy are completely dependent on number of rounds and block size [11]. If two or more block ciphers have the same block size and round value then it becomes difficult to differentiate which is better, becomes a problem. The solution of this problem can be handled by using the key size that plays an energetic role to measure which algorithm is well in which situation.

a. Timing

The time required to encrypt a single block can be obtained by cycle time multiply with amount of cycles to encode one block.

$$\mathrm{T_{block}} = C_B \times T_{cycle}$$

The value of $\mathrm{T_{cycle}}$ is obtained by timing delay of register ($\mathrm{T_{reg.}}$) adding with combinational logic.

$$\mathrm{T_{cycle}} = T_{reg} + T_{comb}$$

The specified constant $\mathrm{C_B}$ means amount of cycles to encode one block, can be obtained by

$$C_B = {}^{R}/_{r} + C_{idle}$$

where $\mathrm{C_{idle}}$ = 2 (typically value), R = amount of cipher rounds, r = amount of rounds implemented in hardware. Finally, the period to encode a single block can be calculated as

$$T_{block} = C_B \times (T_{reg} + r \times (T_1 \times T_2 \times N_b))$$

b. Throughput

The throughput (bits/second) is a significant metric to measure the competence of the algorithm. There may be different throughput value while computing across platforms which depends on the frequency and block size. The throughput can be calculated as

$$Th = (N_B \times F)/C_B$$

where $\mathrm{N_B}$ = block size, F = frequency

B. Hardware Platform Metrics

The hardware implementation platform is dedicated to optimal use of area, speed, and energy. The metrics are inter-related to each other when adjusting some metrics may be influenced by the performance value of other metrics [11].

a. Area

The design area covered by block length can be measured by area. The designed area is also getting affected by different architecture types and implementation procedures. In the case of ASIC (Application-Specific Integrated Circuit) implementation, the occupied area is measured by physical design tools (denoted by um^2). In the case of FGPA (Field Programmable Gate Array) implementation, the resource utilization represents the area, where the resource is vendor dependent basic blocks/elements.

The implementation area depends on the value of r, $\mathrm{N_b}$ and overhead logic. The area can be expressed as

$$\mathrm{A} = \mathrm{r}^{(\rho 2 \times \mathrm{N_b} + \rho 1)} \times \mathrm{A_{r1}} + \nu \times \mathrm{N_b} + \mathrm{A_{r0}}$$

where $\mathrm{A_r}$ = area of implemented r rounds, ρ1 = $\mathrm{A_r}$ growth with respect to r, when $\mathrm{N_b}$ = 0, ρ2 = $\mathrm{A_r}$ growth with respect to r, when $\mathrm{N_b}$ increases, $\mathrm{A_{r0}}$ = area of overhead logic, which includes control and key scheduling, $\mathrm{A_{r1}}$ = area of single round, assuming minimum block size, $\mathrm{N_b}$ = block size, ν = $\mathrm{AN_b}$ growth per bit.

b. Power

The power consumption measurement is essential for defining the performance metrics for the energy of the algorithms. The power consumption can be optimized by lowering clock frequency and reducing the throughputs. The power consumption can be defined as

$$P = (\alpha_1 \times r + \alpha_2) \times F \times A$$
$$= \frac{((\alpha_1 \times r + \alpha_2) \times A)}{T_{cycle}}$$

where α = power per unit area, α_1 = r-dependent power per unit area factor, α_2 = r-independent power per unit area factor, F = maximum frequency (F = 50 typically value)

c. Energy

The energy per bit metrics formalizes the energy concerning the number of bits in a cipher block. The energy to encode a single block is estimated as

$$E_{block} = T_{block} \times P$$

The energy consumption to encode a single plaintext bit for a specific cipher $\mathrm{E_b}$ is stated as

$$E_b = E_{block}/N_b$$

V. COMPARATIVE ANALYSIS OF LIGHTWEIGHT BLOCK CIPHER ON ASIC AND FPGA MODEL

The objective of comparative analysis is to illustrate the variation of performance matrices on ASIC and FPGA models.

Table 2: ASIC and FPGA Designs Constants [11]

Constant	ASIC [30]	FPGA [31]
N_b	64/80	64/80
T_{reg}	11.77	0.0140
$\tau1$	0.5	0.000143
$\tau2$	0.00375	8.9375e-07
$\rho1$	0.60	0.61
$\rho2$	0.0014	0.005
A_{r0}	2445	80
A_{r1}	2042	30
ν	10	2
$\alpha1$	0.0002	0.003
$\alpha2$	0.001	0.5

The ASIC design is an integrated circuit that can be modified for a specific use than proposed for general-purpose use and FPGA design can also be organized by a customer or a designer after built-up. Both designs carry Hardware Description Language (HDL) for hardware configuration. The hardware implementation is highly prejudiced by platform, front-end tools, physical design, etc. The in-depth analysis of design options and results about lightweight block ciphers are presented in [30] [31]. The constant values for ASIC and FPGA are given in table 2[11]. By using these constant values, the software and hardware performance metrics are calculated.

A. Lightweight Block Cipher Implementation in ASIC Model

The performances of lightweight algorithms (HIGHT, LBlock, ITUBee, Improved Lilliput, LED and KLEIN) are measured with the implemented in the ASIC model, performance metrics introduced in the previous section. The encryption time, energy consumption and throughput (bits/sec.) concerning different values of r are presented in figure 2, figure 3 and figure 4 respectively.

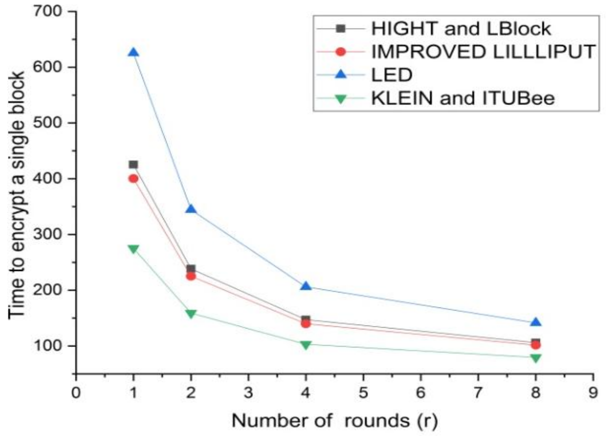


Figure 2: Encryption time for single block versus r

Figure 2 illustrates the encryption time for a single block regarding different values of r. The LED consumes high encryption time and ITUBee and KLEIN block cipher takes less amount of encryption time as compare to all other algorithms. The HIGHT and LBlock consist of the same block size (N_b= 64-bits) and the rounds (R=32), consumes the same amount of encryption time for a single block. The key size (bits) is used to measure the strength of the HIGHT (128-bits key size) and LBlock (80-bits key size) block cipher algorithms.

Figure 3 expressions that the relation between throughput and hardware implementation rounds values (r) is directly proportional to each other. If the value of r is increasing then the throughput is also increasing.

The ITUBee algorithm gives the highest throughput value as compared to all other algorithms. According to the throughput value, the performance of the LED algorithm is very poor on FPGA design. It has been discussed that HIGHT and LBlock give the same throughput value due to its similar properties (block size and key size).

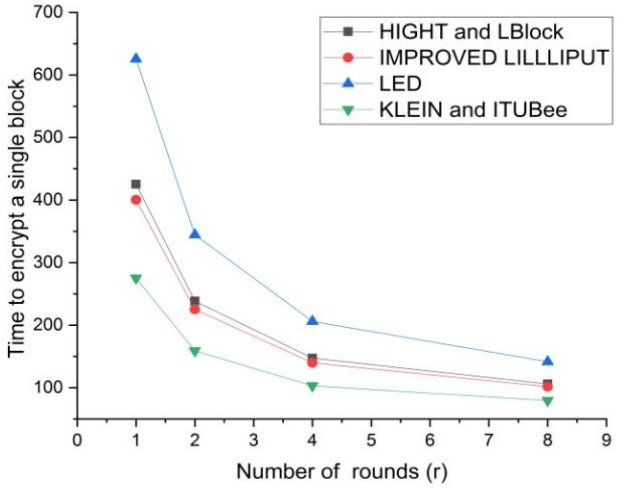


Figure 3: Throughput time (bit/sec.) versus r

Figure 4 shows energy consumption per block regarding the different values of r. The energy consumption value of HIGHT and LBlock are also the same due to their similar properties. The energy consumption rate is high in the LED block cipher algorithm and the KLEIN algorithm consumes less energy as compared to other algorithms r. When the value of r goes above 4, the energy consumption remains constant.

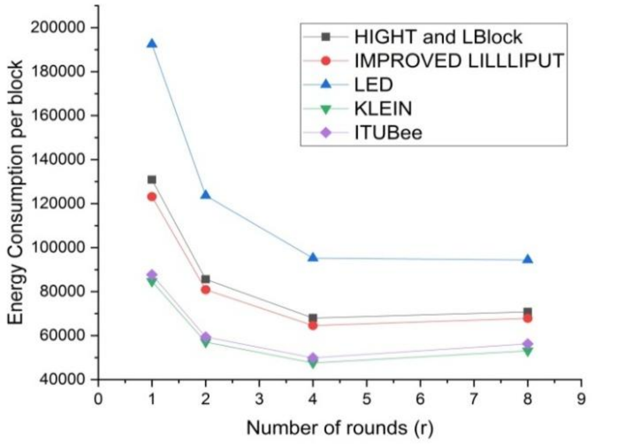


Figure 4: Energy consumption per block versus

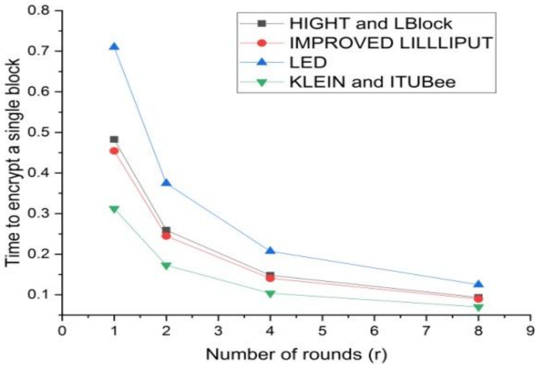
B. Lightweight Block Cipher Implementation in FPGA Model

All the lightweight block cipher algorithms discussed in the previous section, are also implemented in the FPGA model. The encryption time, energy consumption and throughput (bits/sec.) of various algorithms concerning different values of r and the results are presented in figure 5, figure 6 and figure 7 respectively.

The encryption time for a single block regarding different values of r is shown in figure 5.

The HIGHT and LBlock consume high encryption time as compare to other algorithms (Improved Lilliput, LED, KLEIN and ITUBee). In the FPGA model, the KLEIN (Nb=64, R=20) and ITUBee (Nb=80, R=20) consume the approximately same amount of encryption time for a single block.

Figure 5: Encryption time for single block versus r.



The HIGHT and LBlock also consist of the same block size (N_b = 64-bits) and the rounds (R = 32), consumes the same amount of encryption time for a single block. Due to this similar property (same block size and rounds), the key size plays a vital to measure the security strength of these lightweight block cipher algorithms.

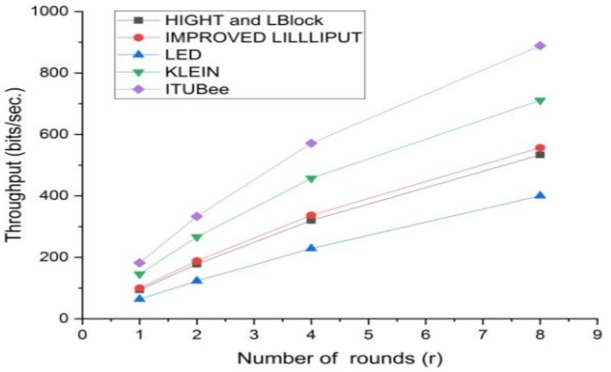


Figure 6: Throughput (bit/sec.) versus r.

The lightweight block cipher algorithms are implemented in the FPGA model gives the same result as the ASIC model. The ITUBee algorithm achieves the highest throughput (bits/sec.) value and the LED algorithm gets the lowest throughput value as compare to other algorithms (figure 6). As the value of r increases, the throughput is also linearly increasing for all the algorithms.

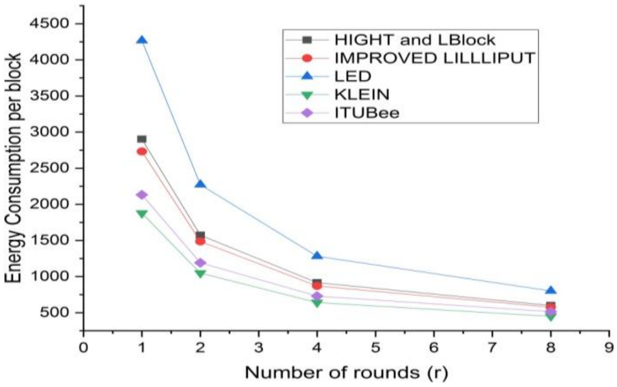


Figure 7: Energy consumption per bit versus r.

The energy consumption (figure 7) value for HIGHT and LBlock are also the same due to their similar properties. The energy consumption rate is high in the LED block cipher algorithm and the KLEIN algorithm consumes less energy as compared to other algorithms. When the value of r increases, the energy requirements of all algorithms are coming very close to each other.

VI. SUMMARY AND DISCUSSION

The comparative performance analysis of performed to arrange the lightweight ciphers according to their ranks with respect of encryption time, throughput and energy consumption. The encryption time and throughput are related to the software performance metrics and the energy consumption is related to hardware performance metrics. At the result discussion phase, it is very difficult to analyze software performance because it is completely dependent on coding style and platforms. In hardware performance metrics, the measurement of energy consumption concludes area and power evolution. If the area and power constraint get influenced due to any reason, directly affects energy consumption.

The summarized results are not exactly consistent because of the differences in platforms. This is not a trivial task to analysis on heterogeneous software and hardware platforms. However, we are drawing the general conclusion and designate the top lightweight block ciphers with respect of time, throughput and energy consumption. The performance of lightweight block cipher algorithms in both ASIC and FPGA models are identical and concise in figure 8 and table 3.

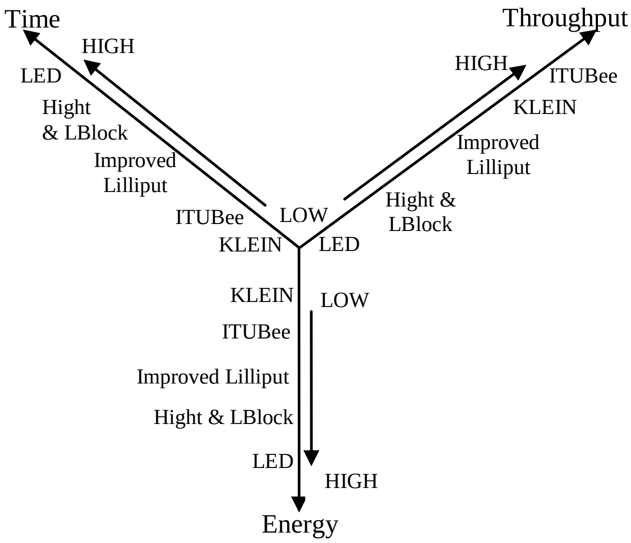


Figure 8: Comparative results of lightweight block ciphers in terms of time, throughput and energy.

The time and energy axis rank best block cipher which takes less encryption timing and energy consumption. Similarly, the throughput axis ranks best block cipher which gives high throughput (bits/sec.).

TABLE 3: CONCISE OF BLOCK CIPHER STUDIES ON SOFTWARE AND HARDWARE PLATFORM

Algorithm	Summary
HIGHT & LBlock	Fourth in encryption timing consumption. Second position in throughput producing. Fourth position in energy consumption.
ITUBee	Second in encryption timing consumption. Fifth and last in throughput producing. Second position in energy consumption.
Improved Lilliput	Third in encryption timing consumption. Third in position in throughput producing. Third in position in energy consumption.
LED	Fifth and last in encryption timing consumption. First in throughput producing. Fifth and last in energy consumption.
KLEIN	First in encryption timing consumption. Fourth in throughput producing. First in energy consumption.

VII. CONCLUSION

As IoT is emerging, it is being applied in wide variety of applications. The major issue in the IoT environment is security and privacy. The IoT devices are built with resource restriction constraints such as limited power source, computational capabilities, and storage capacity. The lightweight and ultra-lightweight block cipher algorithm are most suitable to provide data security.

We discuss that a precise comparative study must be conducted on the platform to measure the performance metrics. The performance metrics is also classified as software and hardware performance Metrics. These Metrics are implemented on ASIC and FPGA model. It consistently measures the encryption time, throughput and energy consumption per block. The same constant values are measured with these models. The result can be analyzed as the encryption time in FPGA is very less as compared to the ASIC model. Both models give the same throughput and the energy consumption is also better in FPGA.

REFERENCES

1. Atzori, L., Iera, A., and Morabito, G., The internet of things: A survey. Computer networks, 2010, 54, pp. 2787-2805.
2. Hou, J., Qu, L., and Shi, W., A survey on internet of things security from data perspectives. Computer Networks, 2019, 148, pp. 295-306.
3. Sethi, P., & Sarangi, S. R., Internet of things: architectures, protocols, and applications. Journal of Electrical and Computer Engineering, 2017.
4. Afzal, B., Umair, M., Shah, G. A., and Ahmed, E., Enabling IoT platforms for social IoT applications: vision, feature mapping, and challenges. Future Generation Computer Systems, 2019, 92, pp. 718-731.
5. Lu, X., Qu, Z., Li, Q., and Hui, P., Privacy information security classification for internet of things based on internet data. International Journal of Distributed Sensor Networks, 2015, 11(8), pp. 932-941.
6. Wu, M., Lu, T. J., Ling, F. Y., Sun, J., and Du, H. Y., Research on the architecture of Internet of Things. In 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), August. 2010, Vol. 5, pp. V5-484).
7. Westin, A. F., Privacy and freedom. 25 Washington and Lee Law Review, 1968, 166.
8. Abomhara, M., Køien, G.M., Security and privacy in the Internet of Things: current status and open issues. In: International Conference on Privacy and Security in Mobile Systems (PRISMS). IEEE, 2014.
9. Katagi, M., & Moriai, S., Lightweight cryptography for the internet of things. Sony Corporation, 2008, pp. 7-10.
10. Ebrahim, M., Khan, S., and Khalid, U. B., Symmetric algorithm survey: a comparative analysis. arXiv preprint arXiv:1405.0398, 2014.
11. Mohd, B. J., and Hayajneh, T., Lightweight Block Ciphers for IoT: Energy Optimization and Survivability Techniques. IEEE Access, 2018, 6, pp. 35966-35978.
12. Hong, D., Sung, J., Hong, S., Lim, J., Lee, S., Koo, B. S. and Kim, H.. HIGHT: A new block cipher suitable for low-resource device. In International Workshop on Cryptographic Hardware and Embedded Systems, october, 2006, pp. 46-59, Springer, Berlin, Heidelberg.
13. Lucks, S., The saturation attack—a bait for Twofish. In International Workshop on Fast Software Encryption, April, 2001, pp. 1-15. Springer, Berlin, Heidelberg.
14. D. Wagner, The Boomerang Attack, FSE'99, LNCS 1636, Springer-Verlag, 1999. pp. 156-170.
15. Jakobsen, T., and Knudsen, L. R., The interpolation attack on block ciphers. In International Workshop on Fast Software Encryption, January 1997, pp. 28-40, Springer, Berlin, Heidelberg.
16. Knudsen, L. R., Truncated and higher order differentials. In International Workshop on Fast Software Encryption, December, 1994, pp. 196-211, Springer, Berlin, Heidelberg.
17. Wu, W., and Zhang, L., LBlock: a lightweight block cipher. In International Conference on Applied Cryptography and Network Security, June, 2011, pp. 327-344, Springer, Berlin, Heidelberg.
18. Biham, E., Biryukov, A. and Shamir, A.: Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials. EUROCRYPT 1999. LNCS, 1999, vol. 3027, pp. 12-23. Springer, Heidelberg.
19. Kim, J., Hong, S., Sung, J., Lee, C. and Lee, S.: Impossible differential cryptanalysis for block cipher structure. In: Johansson, T., Maitra, S. (eds.) INDOCRYPT 2003. LNCS, 2003, vol. 2904, pp. 82-96, Springer, Heidelberg.
20. Biham, E., New types of cryptanalytic attacks using related keys. Journal of Cryptology, 1994, 7(4), pp. 229-246.
21. Karakoç, F., Demirci, H., and Harmancı, A. E., ITUbee: a software oriented lightweight block cipher. In International Workshop on Lightweight Cryptography for Security and Privacy, May, 2013, pp. 16-27. Springer, Berlin, Heidelberg.
22. Ali, M. P., and George, G. T., Optimised Design of Light Weight Block Cipher Lilliput with Extended Generalised Feistal Network (EGFN). International Journal of Innovative Research in Science, Engineering and Technology, 2017, 6(4).
23. Guo, J., Peyrin, T., Poschmann, A., and Robshaw, M., The LED block cipher. In International Workshop on Cryptographic Hardware and Embedded Systems, September, 2011, pp. 326-341, Springer, Berlin, Heidelberg.
24. Biryukov, A., and Wagner, D., Slide attacks. InL. R. Knudsen, editor, FastSoftware Encryption. In SixthInternational Workshop, Rome, Italy, March 1999, LNCS. Springer Verlag.
25. F. Mendel, C. Rechberger, M. Schl□a_er and S. S. Thomsen. The Rebound Attack: Cryptanalysis of Reduced Whirlpool and Gr_stl. In Orr Dunkelman, editor, Fast Software Encryption, volume 5665 of LNCS, Springer, 2009, pg. 260-276.
26. Gilbert, H., & Peyrin, T., Super-Sbox cryptanalysis: improved attacks for AES-like permutations. In International Workshop on Fast Software Encryption, February, 2010, pp. 365-383, Springer, Berlin, Heidelberg.
27. L.R. Knudsen and V. Rijmen. Known-Key Distinguishers for Some Block Ciphers. In Kaoru Kurosawa, editor, ASIACRYPT, volume 4833 of LNCS, 2007, pg. 315-324. Springer.
28. Gong, Z., Nikova, S., and Law, Y. W.. KLEIN: a new family of lightweight blocks ciphers. In International Workshop on Radio Frequency Identification: Security and Privacy Issues, 2011, June, pp. 1-18, Springer, Berlin, Heidelberg.
29. Tarhuni, M. A., Ng, S. H., Samsudin, A. and Ng, W. P., Enhanced counter mode. In 9th Asia-Pacific Conference on Communications (IEEE Cat. No. 03EX732), September, 2003 Vol. 2, pp. 701-705. IEEE.
30. Kerckhof, S., Durvaux, F., Hocquet, C., Bol, D., and Standaert, F. X., Towards green cryptography: a comparison of lightweight ciphers from the energy viewpoint. In International Workshop on Cryptographic Hardware and Embedded Systems, September, 2012, pp. 390-407, Springer, Berlin, Heidelberg.

Retrieval Number: B3742078219/19©BEIESP
DOI:10.35940/ijrte.B3742.098319
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

31. Mohd, B. J., Hayajneh, T., Yousef, K. M. A., Khalaf, Z. A., and Bhuiyan, M. Z. A. Hardware design and modeling of lightweight block ciphers for secure communications. Future Generation Computer Systems, 2018, 83, pp. 510-521.

AUTHORS PROFILE

Prabhat Kumar is pursuing his Ph.D. in the Department of Computer Science, Institute of Science, Banaras Hindu University, Varanasi, India. He did his MCA from Dr. Hari Singh Gour Vishwavidyalaya (A Central University), Sagar, Madhya Pradesh, India in 2017. His research areas are Internet of Things, Machine Learning, Network security, Wireless Sensor Networks, Security and Privacy in IOT.

S. Suresh is an Assistant Professor in the Department of Computer Science, Institute of Science, Banaras Hindu University, Varanasi, India. He obtained Ph.D. from National Institute of Technology, Tiruchirappalli, India. His areas of interest include Theoretical Computer Science, Machine Learning and IoT, Big Data Analytics, Distributed and Cloud Computing. He has served as session chair and program committee several of conferences and workshops.