Indian Journal of Science and Technology, *Vol 9(30), DOI: 10.17485/ijst/2016/v9i30/99035, August 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Lightweight Security Algorithm For Wireless Node Connected with IoT

D. Aakash[1]* and P. Shanthi[2]

[1]Embedded Systems, School of Computing, SASTRA University, Thanjavur - 613401, Tamil Nadu, India; aakashd24@gmail.com
[2]School of Computing, SASTRA University, Thanjavur - 613401, Tamil Nadu, India; shanthip@cse.sastra.edu

Abstract

Objective: The enhancement of wireless technology and sensory device has brought to the introduction towards IoT (Internet of Things). Here security is taken as one of the major issues, as they operate in unlicensed bandwidth. Since most of the device is resource constrained the lightweight security is driven to these devices. Implementation of security is also tedious and selective as the conventional security algorithm degrades the network performance due to the high computational complexity and inherent delays that are incurred for execution of these algorithms**. Method**: In this work, a novel approach is made with new hybrid lightweight security algorithm named PRESENT-GRP which is derived in Intel Galileo Gen 2 board. This methodology helps to secure data during transmission and control process. **Findings**: This hybrid algorithm is an enhancement of the GRP permutation algorithm because it lags an S-box. This proposed algorithm is for low power devices which consumes relatively less memory. Thus, in Internet of Things (IoT) applications this algorithm may help to provide end to end privacy to share data securely. **Improvement:** In future works, this algorithm can be improved with still more security level by having a trade off between performance and memory usage. It can also be enhanced by implementing it in real time military related and health related applications for end to end security for smaller size datum**.**

Keywords: Hybrid Algorithm, IoT, Lightweight, Security, Wireless Network

1. Introduction

The advances in wireless technologies along with sensory devices have brought Wireless Sensor Network (WSN) as unitary of the universal networking support.

The end device and the coordinators are the fundamental components used in these networks, which can be either homogeneous or heterogeneous type deployed randomly or at necessary predetermined locations. Nevertheless, based on the application requirements the sensor nodes can be built to be nomadic and can be programmed to transmit via wireless medium. Aside from these merits, the nodes have a lot of demerits like a physical size limitation and because of which there is a restriction on on-board resources like processing capability, power, storage and bandwidth.

Principally, Aerospace, Military and Boarder security force surveillance were using WSN for their spotting and other monitoring activities, later due to the requirement for commercial market applications like medical monitoring, environs surveillance, industrial supervising and governing, traffic calming and supervising made this system offered to serve commercial products as well. The end device in these environs are brought up for occasional monitoring.

To exemplify, take up a scenario where sensory devices are utile for serving up for military related application, in that environs the data used must be essentially be authentic and secured. Therefore, leakage of datum leads to cracking of privacy concern in the environs. In addition, this may also lead to unauthorized datum injection as well as eavesdropping on the network. In such kind of situations, the securing functionality of the sensor network

Author for correspondence

had become a major concern. In resource constrained devices, complete implementation of a cryptographic algorithm is not possible due to its limitation on factors like power, memory and speed.

Because of its nature WSN possess assorted security issues and attacks, so it is always proposed to use, lightweight security schemes for discarding the technical overheads imposed and in turn this should not affect any of the overall preferred network performance. Secured datum resource, securing a channel route, intrusion detection and avoidance of attacks, key management, cryptography and firewall are some of the other methods which are taken in consideration of WSN.

For the improvement of security and privacy aspects of wireless device, computation with cryptographic techniques like encrypting the data, authenticating the messages received and to be transmitted must be performed[1]. Securing of the network lies in the selection of appropriate security mechanisms. As per standards the design of lightweight cryptography must be of range 1000–2000 gate equivalents[2]. Apart from this info the selection of cipher is also defined based on the following factors like key stream generation, efficiency of the network, selection of keys, modes of operation, operands size in computation and number of iterations processed.

Hence, determination of a Lightweight hybrid cryptographic algorithm for securing wireless powered nodes is concentrated on this paper.

1.1 Cryptography

The word cryptography is meant as "secret writing", the main purpose of it is to define security for data. On considering Wireless Sensor Network (WSN)[3], cryptography plays a vital role in enhancing the security channel sensor data communication.

From Kirchhoff's principle, the wide advertising of the works of a cryptographic algorithm should be possible in order to know about the weaknesses and the security that has to rest entirely with the secret key.

There are two types of cryptographic methods used. They are:

- Symmetric Key Cryptography.

This methodology uses like key for encoding and decoding.

- Public Key Cryptography.

This methodology uses, unlike key for encoding and decoding.

In a Wireless Sensor Network, we mostly prefer on symmetric key cryptography because the nodes used are resource constrained type so on consideration towards the factors like power, processing time, performance and cost. This method is opted as the relevant one due to its fast processing in both software as well as the hardware platform.

The Symmetric Key Cryptography is further subdivided as in Table 1.

Most of the modern ciphers are block cipher and it varies of block size from 64 to 128-bits. And also block cipher is more efficient in hardware than stream cipher.

1.2 Lightweight Cryptography

With the development of new technologies, reduced hardware components and fabrication costs[4,5], new applications have emerged which demand to establish communication between severely resource limited devices in order to achieve specific design goals. While designing all these systems, imparting security for devices and privacy of the intended users have to be given high priority[6]. Traditional security algorithms may not be suitable for these applications as they are designed for providing high levels of security without giving much emphasis for optimized utilization of resources.

In order to make these existing algorithms work with these resources constrained devices special design practices have to be considered which should take into account the memory footprint, execution time, level of security guaranteed and resource utilization. Algorithms that are designed and implemented based on the above factors form a specific group of cryptographic algorithm known as a Lightweight cryptography algorithm[7].

1.3 Different types of Lightweight Cryptography

Security enhancement for data protection is considered to be the major concern for embedded devices and its

Table 1. Basic types of cipher

Types	Description
Block Cipher	In this it encrypts a plain value into a block of bits
Stream Cipher	In this it encrypts plain value completely to cipher symbol values.

applications[8]. Ensuring the data integrity, authentication varying from small systems for critical applications, cryptography is taken as the reliable measure. Lightweight cryptography is highly secure with minimal resource requirements. One such effective, lightweight block cipher namely PRESENT is used in resource constrained environments. The variations in the PRESENT algorithm in terms of making it more complicated for the attacker to break through are dealt exclusively in the paper. PRESENT is an extremely-lightweight block cipher especially for sensor based applications under resource constrained environments[9]. It is a Substitution- Permutation Network (SPN) with size of keys varying in 80 and 128 bits[10]. Though it is lightweight on consideration on GE count its permutation layer requires more memory area. So, a complex bit permutation algorithm is chosen and the algorithm is modified totally. On accounting, all these as the key concept, the PRESENT algorithm suited for providing the security countermeasures for sensor data[11]. The data from sensors is made secure with the PRESENT algorithm where it is encrypted and the key-dependent S-box makes it highly secure[12]. Finally, the information is made secure owing to the fact it may be used for sensitive or confidential application and have to expedite its use for critical applications.

2. Description about GRP Algorithm

GRP algorithm is a complex bit permutation instruction. This factor makes it suitable for cryptographic environment. In this algorithm the permutation takes place in log 2 (n) steps, while other permutation uses O (n) steps.

GRP is analyzed in combination with RC5 block cipher and the result determines that the security level of RC5 algorithm is enhanced with lesser number of cycles[13]. GRP is scaled effectively for 2 (n) bits system by using various instruction IA-64, Reduced Instruction Set (RISC) and Shift Right Pair instruction (SHRP) processors[14].

The Lookup table is one of the change for bit instruction, but it lags speed in processing. Table Lookup is also having drawbacks in the fields of memory requirement and processing performance.

This bit instruction can create various keys at each round from the defined sequence of integers. The generation of the key is derived with an example in Figure 1[15].

GRP 64-bit transposition is developed and a novelty is obtained in 32-bit Intel Quark device. Here the input is given from the sensor and the Group transposition is performed the GRP bit encoding is shown in Figure 2.

Figure 2 also derives the usage of various logic gates for transposition[16]. In Figure 3, the key register derives the key, according to Group transposition, where the sequence of integers is used as a major factor on encoding in each round respectively.

In order to transpose 64-bits, it hardly requires six stages or rounds.

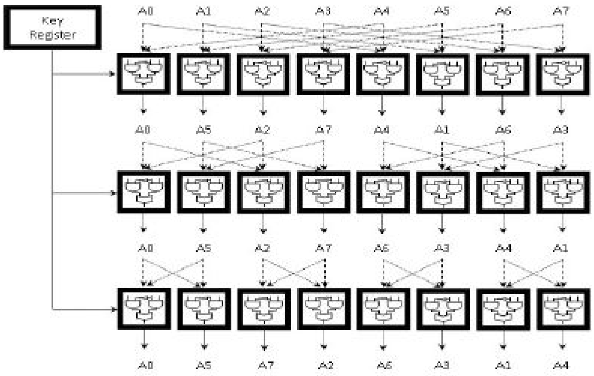


Figure 1. 8-Bit GRP permutation.

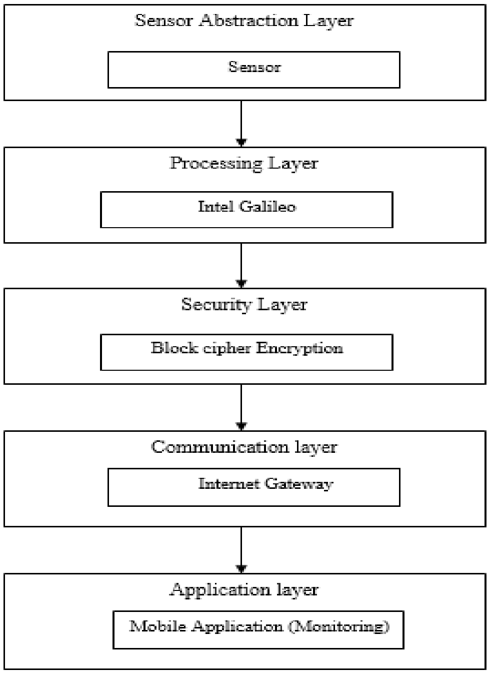
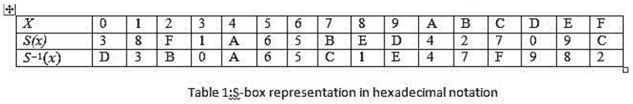


Figure 2. Enhancing security for wireless node with lightweight cryptography algorithm.

$2^6 = 64$-bits

Group transposition requires only two operands and it can do permutation for higher order bits also, it never required bit repetition methodology. Hence this algorithm is also termed for its speed processing.

It is also feasible for permutation as multi-bit sub word and various audio video processing. This algorithm is also utile on accelerating the speed ten times more on sorting out mini-value sets. In Table 2 different permutation algorithm are compared with GRP.



X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(x)	3	8	F	1	A	6	5	B	E	D	4	2	7	0	9	C
$S^{-1}(x)$	D	3	B	0	A	6	5	C	1	E	4	7	F	9	8	2

Table 1:S-box representation in hexadecimal notation

Figure 3. S-box representation in hexadecimal notation.

Table 2. Comparison of Bit-Permutation Algorithms

Permutation	Merit	Demerit
GRP	Uses lesser memory and less GE Count Better cryptographic properties	Number of stages varies with number of bits Latency is increasing with bits size increases
OMFLIP (*Omega-Flip*)	Operate in single cycle and uses Fixed number of stages for permuting Less Latency	Easily reversible
DDR (*Data-Dependent rotation*)	Easy to implement only for small fraction of bitwise operation	Latency is more and require more cycles
PPERM3R	In this the control bit is easy to generate support repetition and result in the middle of permutation can be obtained	Latency is more Initialization process takes time 2n permutation is harder from n permutation
Table Lookup	This method is faster And easy to implement in Hardware	It consumes more memory, latency is more and require more number of instructions

3. Design Flow and Block Diagram

The design flow consists of various modules that are expressed in Figure 2, which helps to distinguish the whole methodology of the proposed work.

4. Hybrid Algorithm

The Group permutation algorithm always uses up one pace in advance of other bit transposition instruction mainly for its exercise in speed cryptographic implementation and lesser memory usage in hardware devices. The only fault found on diverse research is that it lags a Substitution box, which is one the major essence in defining a secured crypt.

Thus, for selecting the S - box we may use a PRESENT lightweight algorithm through analysis taken from various papers. Though, the PRESENT algorithm S box is of non-linear type, it could only provide adequate level of protection. As the methodology used in it is same as PRESENT algorithm, it also supports only 80 and 128-bit key length, with a block size up to 128-bits.

Modified S-Box Layer:

The algorithm makes use of 4*4-bit S-box [16]typically represented by the function $S: F_2^4 \rightarrow F_2^4$. The following S-box is modified to enhance the security and performance for the intended application. The modifications made are stated below:

- The usage of key dependent confusion property is done in two steps overcoming fixed one.
- Some of the major concern of this modification is that it is highly secure in terms of its efficiency proven against linear and differential cryptanalysis.
- The first step in this modified algorithm is choosing the best S-box which has the equivalent characteristics of the PRESENT.
- The linear and differential cryptanalysis is the best tools for checking the security standards of the algorithm where the linear analysis is done with the Walsh Transform which describes the closeness of the Boolean function f to whether it is affine or linear function.
- The differential cryptanalysis explores the strength of the algorithm by checking its non-random behavior by giving certain differences in the input and output to check it is highly probable that is a highly secure block cipher.

- The above criteria were examined and the good S-boxes were found to be Serpent S-boxes. The hexadecimal notation of the S-box is given below Figure 3.

Therefore, the hybrid algorithm is defined as per the block diagram designed below Figure 4.

5. Hardware Implementation Description

Quark, described by Intel at IDF 2013, as a device with reduced usage of power, smaller form factor and also inexpensive; it's also ideal for "wearable" and the Internet of Things.

The overall hardware prototype of the proposed hybrid cryptographic algorithm is Modified PRESENT-GRP which was implemented into IoT environ with Intel Galileo acting as node as well as gateway is derived. The software platform that was used in this application is Intel Arduino 1.6.0, which is one of the rapid prototyping platforms used for Arduino family.

The implementation of the Modified PRESENT-GRP algorithm, in Intel Galileo is set up as the end node and the Android mobile as the base. Thereby the output of the algorithm passes through a cloud by connecting Galileo through the gateway script to the internet and the decrypted output is exposed on the main screen of the application.

The sensor value sensed by the end node is encrypted with the help of a hybrid cryptographic algorithm and is transferred to the cloud using gateway scripts. Then the output from cloud is parsed into the mobile application where decryption process is carried out. At the base, it receives the encoded value and the decoded result is shown in the main screen of the application.

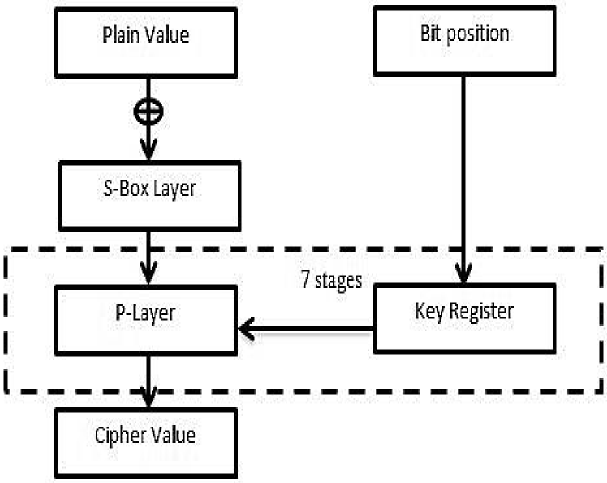


Figure 4. 128-bits modified PRESENT-GRP Permutation block diagram.

Along with the monitoring application designed the remote control off switch is also derived from enabling alert on critical thresholds.

The snapshot of the proposed work is shown in the Figure 5 given below, along with it the hardware description is also derived in Table 3.

In this work the free cloud (www.parse.com) is used to send and receive data. Thus, in this work, only data security is carried out, but in future we may also add cloud security as one of the vast field of research. (Figure 6).

In Figure 6, the execution time taken for encoding and decoding process is given as 0.0519 seconds and 0.0656 seconds respectively. Therefore, the algorithm speed is determined based on the execution time of the two modules. (Figure 7).

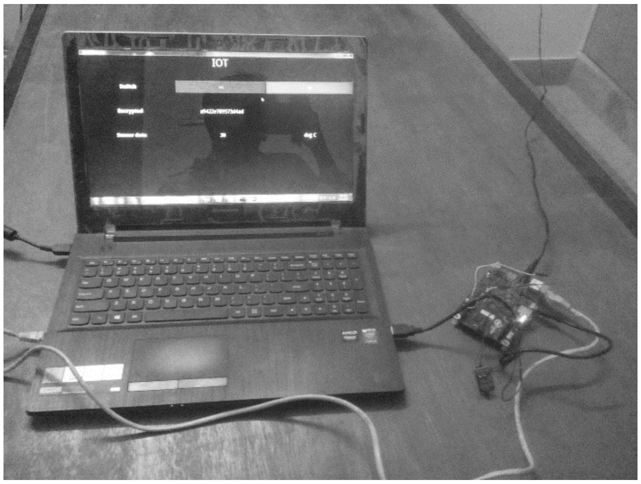
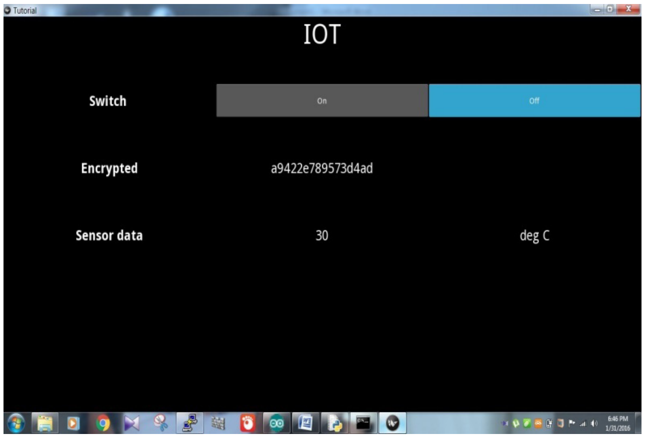


Figure 5. Hardware setup of Intel Galileo Gen 2.

Table 3. Hardware description

WIRELESS NODE		
MCU *Microcontroller Unit*	Quark SoC	
	Device Name	Intel Galileo Gen 2
	Type	32-bit
	CPU Speed	400 MHz
	SRAM	512 kb
	Model Type	Intel® Quark™ SoC X1000



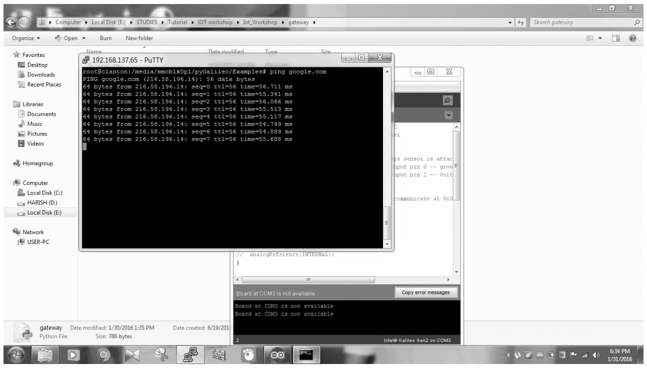
IOT

Switch

Encrypted a9422e789573d4ad

Sensor data 30 deg C

Figure 6. Android application showing output of encrypted and decrypted of hybrid algorithm.



```
root@clanton:/media/mmcblk0p1/pyGalileo/Examples# ping google.com
PING google.com (216.58.196.14): 56 data bytes
64 bytes from 216.58.196.14: seq=0 ttl=56 time=56.711 ms
64 bytes from 216.58.196.14: seq=1 ttl=56 time=55.341 ms
64 bytes from 216.58.196.14: seq=2 ttl=56 time=56.066 ms
64 bytes from 216.58.196.14: seq=3 ttl=56 time=55.513 ms
64 bytes from 216.58.196.14: seq=4 ttl=56 time=55.117 ms
64 bytes from 216.58.196.14: seq=5 ttl=56 time=54.749 ms
64 bytes from 216.58.196.14: seq=6 ttl=56 time=54.889 ms
64 bytes from 216.58.196.14: seq=7 ttl=56 time=55.688 ms
```

```
//  analogReference(INTERNAL);
```

```
Board at COM3 is not available
Board at COM3 is not available
```

Figure 7. Internet gateway connected from Galileo to cloud.

6. Security Analysis

Various cryptanalysis techniques are used on a cipher to find robustness of cipher design against all possible types of attacks. One of the well-known attacks is brute force attack which decodes the value by using all possible ways. This attack can be prevented to a certain extent by enhancing the size of keys, thereby leading to more combination and making it tedious to compute. Thus, these methodologies consume more time for processing its robustness. So, we may move onto two famous cryptanalysis methods that are useful for computing robustness.

Thus the two attacks or properties which are most important to do cryptanalysis are:

- Linear cryptanalysis.
- Differential cryptanalysis.

6.1 Linear Analysis

In this, Walsh transform is considered as one of the best tool for analyzing linear factors as well as it acts as a key perspective for Boolean functionality. Thereby, the below equation from Figure 8 can be used for analyzing the linearity of confusion property:

On measuring with the Equation (1), the modified substitution box is considered to be robust along linear attacks.

For permutation layer we have the Figure 9, which consist of two Types L and M.

From Figure 9, Type L shows how far the transposition moves the bits around and deriving the probability of moving of As to Ot is p, then the possibility of As = Ot is 0.5 + p/2. So the bias for triplet (es, 0, ET) is p/2. On considering for Group algorithm, the highest bias was obtained when all the factors are zero. In Type M it defines the importance of the control bits in order to compute its probability path. Thus, the analysis of two types defines the robustness of the permutation layer. Finally, based on the analysis the hybrid algorithm is determined as effective to linear analysis.

6.2 Differential Analysis

This analysis is also taken as one of the major methods for analyzing block cipher. It helps to determine the secret key with the help of difference of input and output computed in non-random behavior. Hence, for a well-defined block cipher design the probability should be eminent with non-zero input and output divergence. Thus, to analyze for the cryptanalysis, the following Equation (2) from Figure 10 is used to calibrate the differential resistance of confusion property.

$$S_b^W(a) = \Sigma_{x\in F_2^4}(-1)^{b.S(x)\oplus a.x}$$

Where $a \in F_2^4$ and $b \in F_2^4$. (1)

Figure 8. Linear analysis.

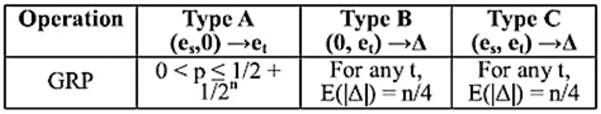
Operation	Type L $(e_s, 0, e_t)$	Type M (e_s, e_u, e_t)
GRP	$b \leq 1/4 + 1/2^{n+1}$ Maximum with s = t = 0	$b \leq 1/4 - 1/2^{n+1}$ Maximum with s = u = t = 0

Figure 9. Linear properties of permutation.

$$\mathrm{Diff}(S) = \max_{x \varepsilon F_2^n} \#\{S(x) + S(x+\Delta_1) = +\Delta_o\}$$

$$\text{Where } \Delta_1 \dot{\varepsilon}\, F_2^n \text{ and } \Delta_o\, \dot{\varepsilon} F_2^m \qquad (2)$$

Figure 10. Differential analysis.



Operation	Type A $(e_s,0) \rightarrow e_t$	Type B $(0, e_t) \rightarrow \Delta$	Type C $(e_s, e_t) \rightarrow \Delta$
GRP	$0 < p \leq 1/2 + 1/2^n$	For any t, $E(\lvert\Delta\rvert) = n/4$	For any t, $E(\lvert\Delta\rvert) = n/4$

Figure 11. Differential properties of permutation.

Hence, on various analysis Serpent S-box are considered to be highly resistant one of these properties.

Similarly, on consideration of the permutation layer Figure 11 denotes certain properties.

Type A denotes the possibility of moving from s bit to t bit is defined when its control bits are derived in a random manner. It also defines the possibility of GRP[16] which is totally dependent on s and t that comes under the possibilities between 0 and 0.5. Thereby as per type A, the characteristic of p is always ½ n for any value of s and t and if it's much lesser or it can be removed.

Type B and C, is used to define the dissemination effect by comparison and on reckoning the hamming weight of difference in output. If its size is huge then it symbolizes the avalanche effect which shows good cryptanalysis properties. E (Δ) represents expected value of difference when the input sequence is taken at random. These types also determine the properties for GRP to bring n/4 variant output bits. Finally, type B characteristics are based on hamming weight and type C is on variation of one bit and it has a lesser effect on the deviation.

Finally, on concluding the above analysis factors the hybrid algorithm shows better robustness towards linear and differential properties.

7. Conclusion and Future Work

In this work, the proposed algorithm uses of key dependent confusion property for deriving a new Modified PRESENT-GRP lightweight algorithm. Here the objective of this implementing key dependent confusion property is to use various substitution boxes for each round. Then the algorithms, major cryptanalysis is also considered and proved as it is robust against those attacks and finally concluded as its improved security level than PRESENT algorithm.

Hence, based on the proposed algorithm the implementation on the targeting hardware device to secure sensor reading of Intel Galileo Gen 2 when operated in IoT (Internet of Things) scenario is prototyped.

The extension of the project can be done for IoT based application with multiple nodes. In addition, to it we will also inject some more attacks on the proposed algorithm to further analyze its security level with the perspective of various resources constrained factors. Thus the algorithm constraints like power, cost and performance are also maintained under tradeoff to recoup its lightweight characteristic. It may be also used in future as an involute cipher with other block cipher like PRINCE to improve its security level.

8. Acknowledgment

The authors would like to thank SASTRA University, Thanjavur for providing support for this research work under the "Fund for Improvement of S&T Infrastructure in Universities and Higher Educational Institutions (FIST) Program–2014" Grant no: SR/FSC/ETI- 371/2014 Also, we are thankful to Dr. A. Umamakeshwari, Associate Dean of CSE department, SASTRA University for her constant guidance throughout the work.

9. References

1. Bogdanov A, Knudsen LR, Leander G. PRESENT: An ultra-lightweight block cipher in Cryptographic Hardware and Embedded Systems (Lecture Notes in Computer Science). Berlin, Germany: Springer-Verlag; 2007. p. 450–66.
2. Zaba MR, Jamil N, Rusli ME, Jamaludin MA, MohdYasir AA. I-PRESENT: An involutive lightweight block cipher. Journal of Information Security. 2014 Jul; 5(3):114–22.
3. Kiruthika B, Ezhilarasie R, Umamakeswari A. Implementation of modified RC4 algorithm for Wireless Sensor Networks on CC 2431. Indian Journal of Science and Technology. 2015 May; 8(S9):198–206.
4. Ashok J, Thirumoorthy P. Design considerations for implementing an optimal battery management system of a wireless sensor node. Indian Journal of Science and Technology. 2014 Sep; 7(9):1255–9.
5. Shi Z, Lee RB. Bit permutation instructions for accelerating software cryptography. Proc IEEE Int Conf Appl Specific Syst. Archit Process. (ASAP); 2000 Jul. p. 138–48.

6. Leander PG, Schramm K, Paar C. New light-weight crypto algorithms for RFID. Proc IEEE Int Symp Circuits Syst. (ISCAS); 2007 May. p. 1843–6.
7. Eisenbarth T, Kumar S. A survey of lightweight-cryptography implementations. IEEE Des Test Comput. 2007 Nov–Dec; 24(6):522–33.
8. Data Encryption Standard (DES). 1995. Available from: www.http.csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf
9. Yang X, Lee RB. Fast subword permutation instructions using omega and flip network stages. Proc Int Conf Comput Design; 2000 Sep. p. 15–22.
10. Hernandez-Castro JC, Peris-Lopez P. On the key schedule strength of PRESENT. SPRINGER; 2011 Sep. p. 253–63.
11. Mihajloska H, Gligoroski D. A new approach into constructing S-boxes for Lightweight Block ciphers. 8th Conference on Informatics and Information Technology with International Participation (CIIT); 2011. p. 1–5.
12. Poschmann A. Lightweight cryptography: Cryptographic engineering for a pervasive world. *Doktor-Ingenieur* Faculty of Electrical Engineering and Information Technology. Germany: Ruhr-University Bochum; 2009 Feb.
13. Borghoff J. PRINCE - A low-latency block cipher for pervasive computing applications. Advances in Cryptology. Berlin, Germany: Springer-Verlag; 2012 Dec. p. 208–25.
14. Kaliski BS, Yin YL. On the security of the RC5 encryption algorithm. RSA Technical Report; 1998 Sep.
15. Kaliski BS, Yin YL. On differential and linear cryptanalysis of RC5 encryption algorithm. Advances in Cryptology (Lecture Notes in Computer Science); 1995 Aug. p. 171–84.
16. Lee R, Mahon M, Morris D. Pathlength reduction features in the PA-RISC architecture. 37th IEEE Comput Soc Int Conf. Dig Papers; 1992 Feb. p. 129–35.