RAGHAV

A new low power S-P network encryption design for resource constrained environment

Gaurav Bansod
Associate Professor
gaurav249@gmail.com
Pune Institute of Computer Technology(PICT), Pune
India

Abstract— **This paper proposes a new ultra lightweight cipher RAGHAV. RAGHAV is a Substitution-Permutation (SP) network, which operates on 64 bit plaintext and supports a 128/80 bit key scheduling. It needs only 994.25 GEs by using 0.13µm ASIC technology for a 128 bit key scheduling. It also needs less memory i.e. 2204 bytes of FLASH memory , which is less as compared to all existing S-P network lightweight ciphers. This paper presents a complete security analysis of RAGHAV, which includes basic attacks like linear cryptanalysis and differential cryptanalysis. This paper also covers advanced attack like zero correlation attack, Biclique attack, Algebraic attack, Avalanche effect, key collision attack and key schedule attack. In this cipher,use of block permutation helps the design to improve the throughput. RAGHAV cipher uses 8 bit permutations with S-Box which results in better diffusion mechanism. RAGHAV consumes very less power around 24mW which is less as compared to all existing lightweight ciphers. RAGHAV cipher scores on all design metrics and is best suited for applications like IoT.**

Index Terms— **Lightweight cipher, SP Network, Block cipher, Encryption standards, Embedded security, Ubiquitous computing, IoT (Internet of Things).**

I. INTRODUCTION

Lightweight cryptography is an emerging filed which represents the family of lightweight ciphers that are suitable for constrained environment. Many lightweight ciphers over a decade have been designed for applications where memory space, Gate Equivalents (GEs) is the major constraints. PRESENT[1], TWINE[4], PICCOLO[3], LED[6], MIDORI[29], PICO[22], ANU[26], BORON[27], SIMON and SPECK[28] are the popular lightweight cipher designs. All these ciphers have GEs ranging from 1000-2000. The most lightweight design in terms of Gate Equivalents is SIMON and SPECK, but there security is not guaranteed. PRESENT cipher is the most trusted and versatile lightweight design till date. No attacks are reported on PRESENT cipher. Most of these ciphers lack on one of the design metrics. In case of PRESENT cipher, it lacks throughput. PRESENT cipher has a very less throughput while BORON [27] which is latest S-P network and has highest throughput, but its power consumption is also high. There is need to design the block cipher which should score on all metrics as block ciphers are considered to be the workhorse in the cryptographic environment. The most ignored metric in design of a lightweight cipher is power dissipation which is a very crucial in the environments like IoT and Wireless Sensor Networks (WSN). In Wireless Sensor Network, most of the nodes are battery powered and there is a need to protect these nodes against external attacks. The versatile cipher like AES, Triple DES fails in such kind of environment as they need huge memory space as well as they dissipated more power. There is urgent need to secure these nodes without incurring more power to make the technologies like IoT feasible. This paper presents a cipher RAGHAV which has less GE's, needs less memory space, dissipates less power and have competitive throughput as compared to existing lightweight ciphers.

II. DESIGN CHOICES AND OUR CONTRIBUTION

First aim to design cipher is to reduce the Gate Counts so that the cipher should result in small hardware implementation. Moreover power consumption should also be less as the cipher design is aimed at providing security to battery powered nodes. Flash memory size of cipher should be very less as targeted processor would be of 8 bit, which generally has very less memory space. The designed cipher should perform efficiently both on hardware as well as on software platforms. By maintaining all these metrics, throughput of the cipher design also should be competitive.

Design Strategies:

- RAGHAV cipher has adopted 8-bit permutation. This result in minimum memory requirement as other S-P network ciphers operates on 64 bit and have 64 bit permutation layer. In RAGHAV cipher we made a successful attempt to minimize memory requirement by using a bit permutation with high diffusion mechanism. Bit permutation layer is designed in such a way that that bit distribution results in maximum number of active S-box in minimum number of rounds.
- RAGHAV cipher has used Robust S-box, which have $CAR_{LC} = 2$ & $CAR_{DC} = 2$ and it is highlighted in the Difference Distribution Table in section II. This property of S-box helps to improve diffusion layer with bit permutation layer and block permutation.

- RAGHAV cipher design has bit and block permutation with circular shifting that consumes very less GEs as shifting operators and bit permutation only needs wires in hardware implementation.
- RAGHAV cipher design also used 16-bit block permutation which results in good throughput as compared to SP network cipher tabulated in Table 13.
- RAGHAV mainly needs GE's only for S-Box, XOR gates, and fewer registers for storing plain text and key.
- In the cipher design at software level, care is taken to use minimum number of local and global variables which results in less memory requirement.
- Due to the use of a mesh kind of network in the cipher design and the reuse of limited registers in programming, the power dissipation of RAGHAV cipher is less.
- Different units of shift operators are used in the RAGHAV cipher design which results in more number of active S- boxes in minimum number of rounds.

RAGHAV cipher design aim is to provide rich encryption standards for those environments where power dissipation is critical.

III. THE BLOCK CIPHER RAGHAV

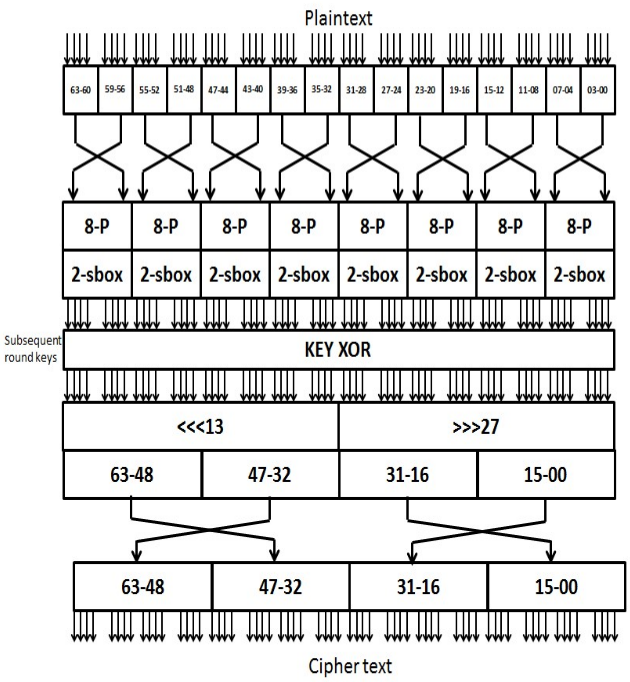

Figure 1: The Block Cipher RAGHAV

RAGHAV is a S-P network based cipher design which has the round function as shown in the Figure 1. This cipher design supports 128/80 bit key scheduling algorithm. In RAGHAV design 4bit block of data is grouped together and permuted as shown in Figure 1. In the design we have also used 8 bit permutation layer i.e. 8-P blocks. After permuting the bits, the non linear layer is applied i.e S- box. RAGHAV cipher has strong S-Box which act as a nonlinear layer which has CARLC = 2 and CARDC = 2. CARLC and CARDC represent cordiality property of linear and differential attacks which are discussed in next sections.

The output of non linear layer is XOR-ed with the subsequent sub keys. Cipher design also uses circular shifts which shift the LSB 32 bits by 27 and right shifts MSB 32 bits by 13. RAGHAV cipher uses this shifting to spread input data such that it will help the cipher design to increase number of active S-Box in minimum number of rounds. Circular shifted total 64 bits are grouped as 16 bits and cross permuted in the last layer before generation of cipher text. This round function runs 31 times to generate final cipher text. Key scheduling algorithm is used to generate subsequent sub keys for subsequent rounds.

A. S –Box

In the cipher design S – Box is only the component which has introduced the non linearity in the design. This has been helpful for increasing the complexity in any cipher. This layer substitutes the new value to its input value according to prefixed Table 1. RAGHAV cipher has used the same S - box which is given in the table 1 and satisfies all the criteria required for strong S – Box[1][5].

S-box S: $F_2^4 \rightarrow F_2^4$

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

Table 1: S-box of RAGHAV cipher

Design Criteria and properties of S – Box for RAGHAV cipher is the same which are used while designing PRESENT [1] and RECTANGLE [5]. But this S- box is having strong properties as compared to the S- box used in PRESENT and RECTANGLE.

B. Permutation P Layer (8 bit)

Permutation layer increase the strength of the round ciphers and introduces the more complexity. The main use of the Permutation is that it needs only wires for designing so which has not required GE's. RAGHAV also used the block shuffling and circular shifting which have used to increase the number of active S – Box. For designing the strongest permutation layer, the following criteria we have followed which is mentioned in paper [8]

1] At round r, the output of S-box is distributed in such a way that two of them affect the middle bits of S-box at round r+1 and other two affects the end bits.

2] The four bit output of each S – Box affect the next four different S – boxes.

i	0	1	2	3	4	5	6	7

P[*i*]	2	4	6	0	7	1	3	5

Table 2: Bit permutation table of RAGHAV cipher

C. Key Scheduling of RAGHAV - 128

The key scheduling of RAGHAV cipher is motivated by key scheduling of PRESENT cipher because till date no attacks are reported on the key scheduling of PRESENT cipher. The input key is stored into the KEY register which is given as K_{127} K_{126} $K_{125}\ldots K_2$ K_1 K_0 ,and from that for RAGHAV ,64 leftmost bits as key $K^i = K_{63}\ K_{62}\ldots K_2\ K_1\ K_0$ is applied to the initial round of cipher. For next round, the KEY register is updated as per the following key scheduling steps:

1) KEY register is circularly left shifted by 13
   KEY <<<13
2) The leftmost 8 bits i.e. $K_7\ K_6 \ldots\ K_1\ K_0$ are passed through the S – Box of RAGHAV.
   $[K_3\ K_2\ K_1\ K_0] \leftarrow S\ [K_3\ K_2\ K_1\ K_0]$
   $[K_7\ K_6\ K_5\ K_4] \leftarrow S\ [K_7\ K_6\ K_5\ K_4]$
3) Apply the round counter. i.e. XOR the round counter RC^i of respective round with the key bits $K_{63}\ K_{62}\ K_{61}\ K_{60}\ K_{59}$.
   $[K_{63}\ K_{62}\ K_{61}\ K_{60}\ K_{59}] \leftarrow [K_{63}\ K_{62}\ K_{61}\ K_{60}\ K_{59}] \oplus RC^i$

D. Encryption Algorithm

Input-
Plain text: $A_{64} \rightarrow a^{63}\ a^{62}\ a^{61}\ a^{60} \ldots a^3\ a^2\ a^1 a^0$, *S-Box [16], P [8],*
Output-
Cipher text: C_{64}

For i = 0 to 31 do
$P_i^L \rightarrow a^{63}\ a^{62}\ a^{61}\ a^{60} \ldots a^{35}\ a^{34}\ a^{33} a^{32}$
$P_i^R \rightarrow a^{31}\ a^{30}\ a^{29}\ a^{28} \ldots a^3\ a^2\ a^1 a^0$

Pt1 ← *((*P_i^L *&0xf0f0f0f0) >> 4) | ((* P_i^L *&0x0f0f0f0f) << 4)*
Pt2 ← P [Pt1]
Pt3 ← S-Box [Pt2]

Pt1 ← *((*P_i^R *&0xf0f0f0f0) >> 4) | ((* P_i^R *&0x0f0f0f0f) << 4)*
Pt2 ← P [Pt1]
Pt4 ← S-Box [Pt2]

Pt5 ← [Pt3 ⊕ (RKi & 0x00000000ffffffff)]
Pt6 ← [Pt4 ⊕ (RKi & 0xffffffff00000000)] >> 32

Pt7 ← RCS (Pt5, 27)
Pt8 ← LCS (Pt6, 13)

Pt9← RCS ((Pt7 &0xffff0000), 16) | LCS ((Pt7&0x0000ffff), 16)

Pt10← RCS ((Pt8 &0xffff0000), 16) | LCS ((Pt8&0x0000ffff), 16)

$A_{64} \rightarrow\ P_{i+1}^L \,||\, P_{i+1}^R$
i = i+1

End
$C_{64} \rightarrow\ A_{64} \rightarrow P_{31}^L \,||\, P_{31}^R$

IV. Security Analysis

Security analysis consists of different cryptanalysis technique that every cipher should resist [26]. Cryptanalysis is the scientific way to test the strength of cipher. This paper shows the result of basic cryptanalysis attacks like linear and differential cryptanalysis and also covers advanced attack like zero correlation and Biclique attack. Non-linear layer i.e. S-box in the RAGHAV cipher design plays a very important role in security analysis.

A. Linear and Differential Cryptanalysis

The Linear and Differential Cryptanalysis are the basic cryptanalysis techniques [10]. The maximum number of active trails in minimum number of round shows the good resistance against linear and differential attacks for any cipher. These trails can be found out from the minimum number of active S-Boxes with the help of Difference Distribution Table (DDT) and Linear Approximation Table (LAT). The Linear cryptanalysis depends on the high probability of occurrence of linear expression where as Differential cryptanalysis depends on the occurrence of every difference pair with high probability in DDT.

Linear Cryptanalysis

The linear cryptanalysis is also known as known plain text attack [27], which is in the form of linear expressions having the plaintext, cipher text and key bits. The maximum bias can be calculated by using the linear approximation. In case of the RAGHAV cipher, the maximum bias is 2^{-2} based on S- box used in the cipher design. Matsui`s Piling-up lemma [Howard] is used to calculate the probability bias for 'n' rounds. Consider the example, for the round 2, the following figure 2 shows the minimum number of active S – Boxes in RAGHAV cipher for linear trails. Computer generated algorithms are used to find out minimum number of active S- boxes for 'n' rounds.

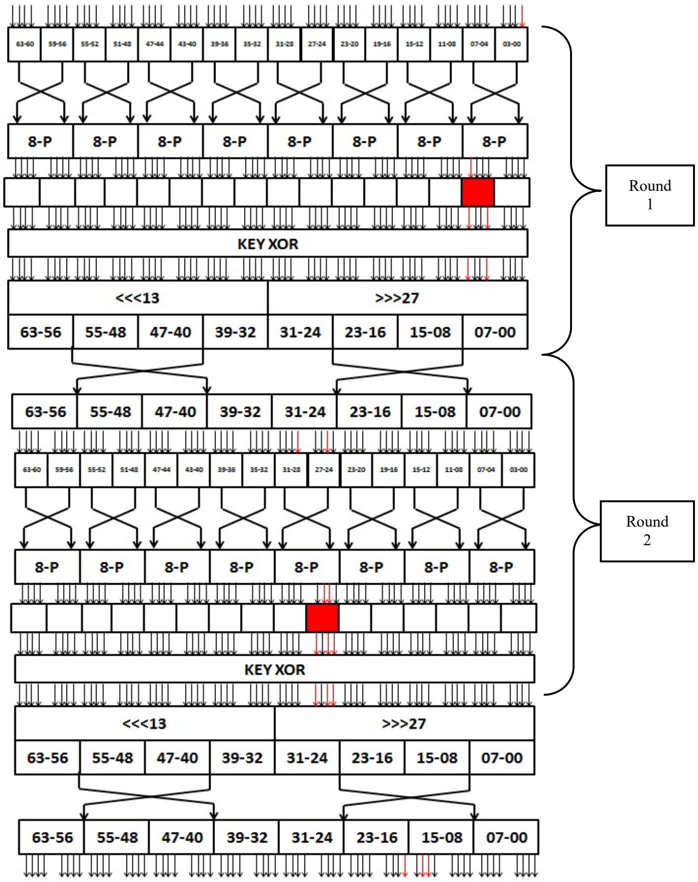

Figure 2: Linear Trails

Red arrow indicates the active trails in the respective round. The total numbers of red boxes are 2, which depicts minimum two active S – Boxes. Table 3 shows the minimum number of active S – Boxes for linear trials.

#Round	# Min. active S-boxes
1	1
2	2
3	4
4	6
5	12

Table 3: Minimum number of active S-box from Linear Trail

For round 5, the total number of minimum active S – boxes are 12. Maximum bias for RAGHAV S – Box is 2^{-2}. So as per the piling up lemma principle the total bias for round 5 is[26]:

$$= 2^{(\text{No. of active S – Box - 1})} \text{ x } (\text{Max. Bias})^{(\text{No. of Active S - Box})}$$

$$= 2^{(12 - 1)} \text{ x } (2^{-2})^{(12)}$$

$$= 2^{-13}$$

Hence to find the total number of minimum active S – Boxes for RAGHAV cipher, one have to consider 25 rounds and calculated the minimum number of active S –Boxes. This can be given by piling up lemma principle:

For 25 rounds the bias will be

$$\varepsilon = 2^4 \times (2^{-13})^5 = 2^{-61}$$

The complexity of linear attack can be given by formula $N_L = 1/(\varepsilon)^2$, So for RAGHAV, the linear complexity is given as below:

$$N_L = 1/(\varepsilon)^2 = 1/(2^{-61})^2$$
$$N_L = 2^{122}$$

Thus one can conclude that the 25 rounds of RAGHAV, gives the sufficient security against linear cryptanalysis.

Differential cryptanalysis:

Differential cryptanalysis [12] [13] is applied successfully by Biham and Shamir on DES (1990). The minimum number of active trails can be found out by using difference distribution table (DDT). In this attack, one has to consider the input and output difference with high probability occurrence from DDT. The S – box which has non-zero input differences and non-zero output differences are to be considered as active S – Box [26]. Consider the example, for the round 2, the following figure 3 shows the minimum number of active S – Boxes in RAGHAV cipher for differential trails.

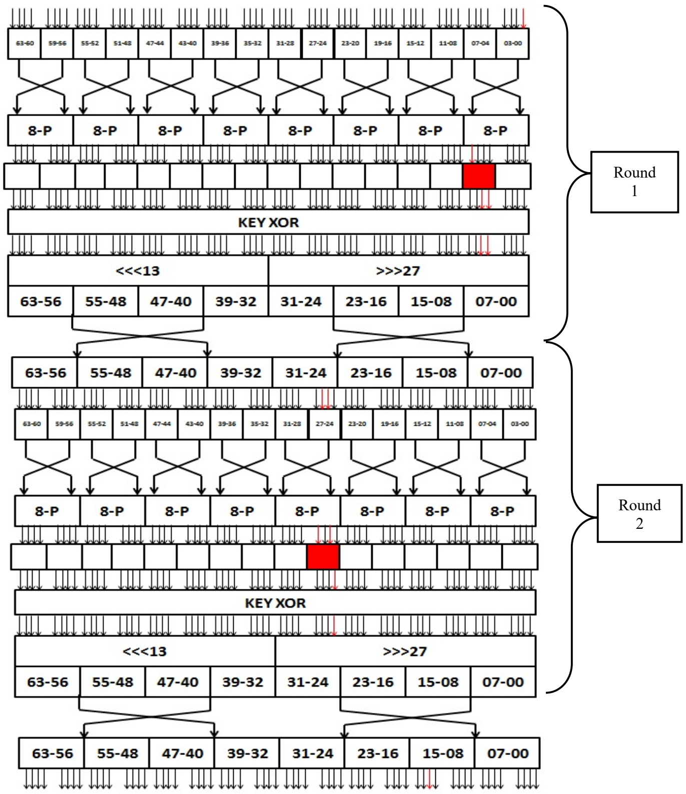

Figure 3: Differential Trails

Red arrow indicates the active trails in the respective round. The total numbers of red boxes are 2, so there are minimum two active S – Boxes. Table 4 shows the minimum number of active S –Box for differential analysis.

#Round	# Min. active S-boxes
1	1
2	2
3	4
4	6
5	9

Table 4: Minimum number of active S-boxes from Differential Trail

The maximum bias can be calculated by using the differential distribution table and in case of the RAGHAV cipher; the maximum bias is 2^{-2}. The differential probability is given by P_d:

$$P_d = (2^{-2})^{\text{No. of active S – Box}}$$

In case of RAGHAV cipher, for 5 rounds, 9 minimum number of active S – Boxes are present, so for 25 rounds there will be minimum of 45 active S – Boxes. The total differential probability P_d is given as:

$$P_d = (2^{-2})^{45} = 2^{-90}$$

The complexity of the differential attack can be calculated as $N_d = C / P_d$. Where C = 1 and $P_d = 2^{-90}$.

$$N_d = ½^{-90} = 2^{90}.$$

Thus, we conclude that the differential complexity is greater than the defined limit i.e. 2^{64}, hence RAGHAV cipher gives the sufficient security against differential cryptanalysis.

B. Biclique Attack

The complexity of the block cipher can be improved with the help of Biclique attack. Biclique attack is an extension of Meet In The Middle (MITM) attack [15]. It is a theoretical attack. The RAGHAV-128 cipher successfully resists against the Biclique attack and gives the maximum data complexity i.e. 2^{40} which is comparatively greater than other existing ciphers. The attack is mounted on rounds 28 ~ 31 with 4 – dimensional Biclique. The selection of the key is very important in the biclique and MITM attacks. These keys are selected from key scheduling algorithm and the position of the keys is given below:

Round 28 = K_{83}, K_{82}………K_{20}
Round 29 = K_{70}, K_{69}…… ….. K_7
Round 30= K_{57}, K_{56},…K_0,K_{127} …. K_{122}
Round 31 = K_{44}, K_{43},…K_0,K_{127} ….. K_{109}

From above keys position, by selecting the specific keys (K_{34}, K_{35}, K_{36}, K_{37}) and (K_{47}, K_{48}, K_{49}, K_{50}) ,gives the highest data and computational complexity for RAGHAV-128. Where K_{34}, K_{35}, K_{36}, K_{37} are the Δi-deferential i.e. forward path key selection which is denoted by red color and K_{47}, K_{48}, K_{49}, K_{50} are the ∇j-deferential i.e. reverse path key selection which is denoted by blue color. The forward and backward path for Biclique is shown in the figure 4. Care should be taken that the both red and blue keys should not activate the same bit or same S- box. This will led to attack failure.

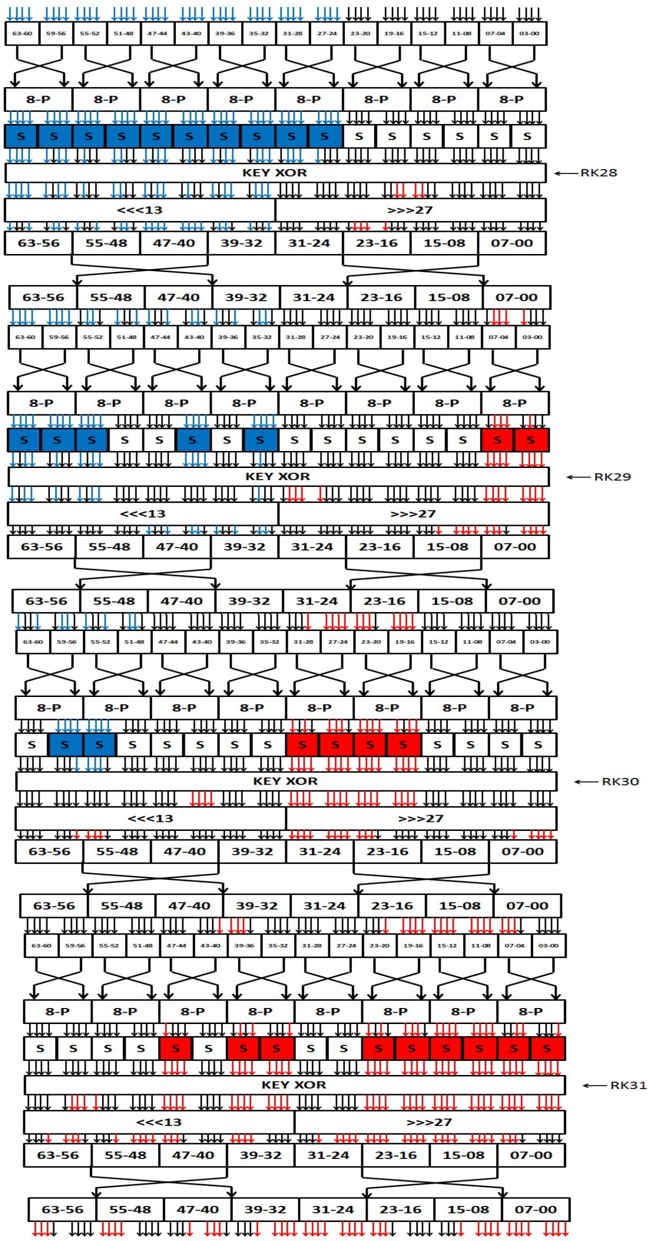

Figure 4: Biclique attack on RAGHAV-128 (4-Dimensional)

Meet In The Middle attack

Meet in the Middle attack helps to analyze complete computational complexity of the block cipher. The key selection is also important for the MITM and this is in totally coordination with the Biclique attack [26]. There are some crucial rules that should follow in case of selecting the 4 dimensional keys while mounting the Biclique attack. These selection of keys in Biclique helps in MITM also which are given below:

1) Red key selected in Biclique should be same as the red key required for backward computation in MITM.
2) Blue key selected in Biclique should be same as the blue key required for forward computation in MITM.
3) While mounting the Biclique or MITM the red trails and blue trials should not be mixed as stated in Biclique.

The total computational complexity can be calculated by using the following formula [15]:

$$C_{Total} = 2^{K-2d} \left(C_{biclique} + C_{precomp} + C_{recomp} + C_{falsepos} \right.$$

Where, K is the length of key, in this case it is 128. d is Biclique dimension, in this case its 4.

$C_{biclique}$ is Biclique computational complexity can be calculated as follows:

$$C_{Biclique} = 2^{d+1} \times \frac{No.\ of\ rounds\ in\ Biclique}{Total\ No.\ of\ Rounds\ in\ cipher}$$

$C_{precomp}$ is Pre – computational complexity can be calculated as follows:

$$C_{precomp} = 2^{d} \times \frac{No.\ of\ rounds\ in\ Precomp}{Total\ No.\ of\ Rounds\ in\ cipher}$$

C_{recomp} is Re – computational complexity can be calculated as follows:

$$C_{recomp} = 2^{2d} \times \frac{No.\ of\ active\ S-Box\ in\ precomp}{Total\ No.\ of\ S-Box}$$

$C_{falsepos}$ is falsepos computational complexity can be calculated as follows:

$$C_{faslepos} = 2^{2d - No.of\ matching\ bits}$$

The computational complexity of the RAGHAV – 128 is $C_{total} = 2^{127.028}$. Table 5 shows the data and computational complexity comparison between RAGHAV-128 and other existing lightweight ciphers.

Cipher Name	Rounds	Data Complexity	Computational Complexity	Reference
RAGHAV-128	31	2^{40}	$2^{127.028}$	This Paper
PRESENT-80	31	2^{23}	$2^{79.54}$	[Jeong, 12]
PRESENT-128	31	2^{19}	$2^{127.42}$	[Jeong, 12]
PICCOLO-80	25	2^{48}	$2^{79.13}$	[Jeong, 12]
PICCOLO-128	31	2^{24}	$2^{127.35}$	[Jeong, 12]
LED-64	48	2^{64}	$2^{63.58}$	[Jeong, 12]
LED-80	48	2^{64}	$2^{79.37}$	[Jeong, 12]
LED-96	48	2^{64}	$2^{95.37}$	[Jeong, 12]
LED-128	48	2^{64}	$2^{127.37}$	[Jeong, 12]

Table 5: Biclique Attack Comparison

C. Zero Correlation Attack

Zero – correlation technique is mounted on the block ciphers and which is the extended part of linear approximation and impossible differential cryptanalysis [14]. Zero – Correlation is technique to find the correlation with value zero and for mounting the attack, in this paper, matrix method has been applied. There are some principles one should follow while mounting the zero – correlation which has been explained by the following figures 5 [14].

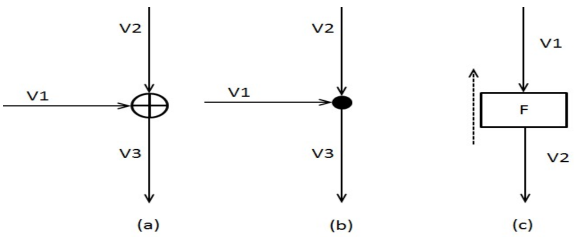

Figure 5: (a) XOR operation (b) Branching Operation (c) F – function

In figure 2 (a) $V_3 = V_1 = V_2$, it shows that whenever there is an XOR operation, then always all the values should be equal. In figure 2 (b) $V_3 = V_1 + V_2$, it shows the summing point gives the actual addition of remaining two branches as per the operation given in table no. --. In figure 2 (c) it shows that whenever there is F – function then it should be always operate at reverse direction but in RAGHAV cipher deign we have not used the F – function .There are different rules for arithmetic addition in the matrix method for that some specific notations are used like, 0 denotes zero mask,   denotes an arbitrary non zero mask, a denotes non zero mask with fixed value,   denotes an arbitrary non zero mask, * denotes any other mask. Table 6 shows the arithmetic rules for matrix addition for zero – correlation.

+	**0**	**$\bar{0}$**	**a**	**$\bar{a}$**	*
0	0	$\bar{0}$	a	$\bar{a}$	*
$\bar{0}$	$\bar{0}$	*	$\bar{a}$	*	*
b	b	$\bar{b}$	a+b	$\overline{a+b}$	*
$\bar{b}$	$\bar{b}$	*	$\overline{a+b}$	*	*
*	*	*	*	*	*

Table 6: Zero correlation arithmetic

By applying (0000 0000 0000 0000 0000 0000 0000 aaaa) from initial round 0 and stepping further in forward direction according to the RAGHAV cipher design, one should get following trails. By applying (bbbb 0000 0000 0000 0000 0000 0000 0000) from round 5 in backward direction, following trails can be derived as shown in the table . The trails for zero correlation attack for RAGHAV cipher is shown in the table 7 and contradiction is achieved at round number 3 represented in terms of 0 and $\bar{0}$.

#Rounds	Trails
1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 a aaa
2	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 $\bar{0}$ 0 $\bar{0}$ 0 $\bar{0}$ 0 $\bar{0}$ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
3	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 $\bar{0}$ 0 0 0 $\bar{0}$ 0 0 0 0 0 0 0 0 0 0 0 0 0 $\overline{0\ 0}$ 0 0 0 0 0 0 0
3	0 0 0 0 0 0 0 $\bar{0}$ 0 0 0 0 0 0 0 0
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
4	0 0 0 0 0 0 0 0 0 0 0 $\bar{0}$ 0 0 0 0 0 0 0 0 0 0 0 0 $\bar{0}$ 0 0 $\bar{0}$ 0 0 $\bar{0}$ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
5	b bbb 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Table 7: Trails for zero-correlation for RAGHAV cipher

D. Key Schedule Attack

For designing the key scheduling algorithm there is no set of rules or guidelines available, hence there are many versions of key scheduling algorithms present in cipher designing. Such types of algorithms are tested by using the key related attacks. Related key attack [17] and slide key attack [18] are the two attacks which focus on weakness of cipher's key scheduling. Related key attack is successfully applied on AES-256[19].

In this paper, key scheduling algorithm which is used in the RAGHAV cipher design is motivated by the PRESENT cipher because there is no key related attack mounted on PRESENT cipher till date. The key scheduling used here is having the nonlinear element S – Box which gives the high diffusion property and XOR operation of 5 – bits from the key register with round constant RC_i to introduce more complexity in the algorithm.

E. Avalanche Attack

When a single bit of plaintext or key bits is changed, then after completion of all rounds, cipher text bits should be changed by more than 50 % with +/- negligible deviation, this phenomenon is known as Avalanche effect. The block cipher is known to have poor randomization and cryptanalytic properties, if cipher does not follows the Avalanche effect. Those ciphers who has poor Avalanche effect, the plain text of such ciphers are easily predictable and cipher can be cracked. For testing the Avalanche effect the help of computer generated algorithm found to be fruitful. In RAGHAV cipher design, almost all cipher text have changed more than 50% with negligible deviation for one bit in plaintext as well as in key. Table 8 shows the some of the results of Avalanche effects of RAGHAV-128.

Plaintext	0000 0000 0000 0000	# Bits changes
Key **Cipher text**	0000 0000 0000 0000 0000 0000 0000 0000 0f7d8e1d5184d11a	-
Key **Cipher text**	0000 0000 0800 0000 0000 0000 0000 0000 730281ffc81ce9b6	32
Key **Cipher text**	0000 0000 0004 0000 0000 0000 0000 0000 ee9476d76673591b	33

Table 8: Avalanche Effect For RAGHAV-128

V. Hardware and Software Performance of RAGHAV

A. GE's

The area requirement is the main concern in hardware performance for any cipher design for application like WSN and IoT. The RAGHAV Cipher has very less number of requirements of Gate Equivalents and also less flash memory size which results in good hardware as well as software performance. In this design for computation of Gate Equivalent we used standard library of IBM 8RF with 0.13 µm technology. The areas of some basic gates in this library are: NOT 0.75, AND 1.25, OR 1.25, XOR 2.00, 2-1 MUX 2.25, D flip-flop 4.25. Figure 6 shows the data path for RAGHAV-128 which is based on the serial ASIC implementation [21]. The key point of the serial ASIC implementation is that it uses only one S – Box repeatedly for complete round so it saves the most of the GE's [26]. In RAGHAV-128 the use of permutation block and bit – bit permutations are more so it results in only use of wires which has not required any GE's. So the RAGHAV-128 has very less number of GE's i.e. 1226.5 compared to other existing ciphers. Figure 6 shows the Serial ASIC data path of the RAGHAV-128. In the design use of circle A, B, A', B' are the connectors.

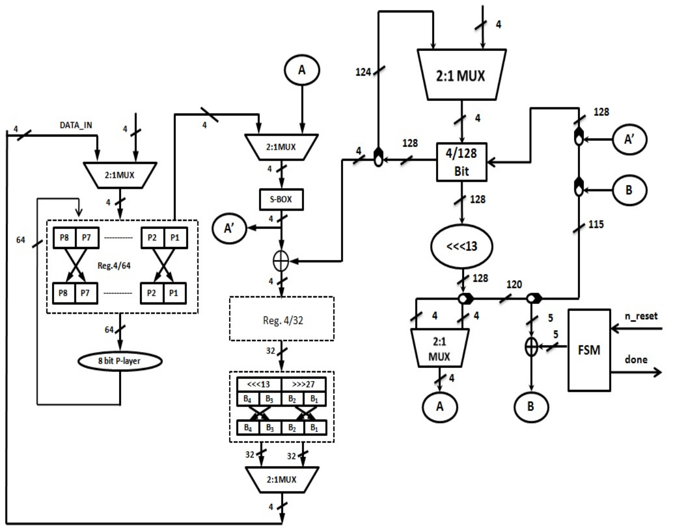

Figure 6: Data path for RAGHAV-128

According to the IBM 0.13 µm library, we have calculated the GEs for RAGHAV-128 [22]. Table no 9 gives the complete required GE's for each section i.e. data section and key section of RAGHAV-128.

Data Layer	GE's	Key Layer	GE's
D Reg.	96×4.25	D Reg.	128×4.25
2:1 MUX	40×2.25	2:1 MUX	2×2.25
XOR	4×2	XOR	5×2
SBOX	28	Shift Operator	0
Shift Operator	0	FSM	134
Total	534	Total	692.5
Total No. of gates required for 128 bit key = 1226.5			

Table 9: Calculation of GEs for RAGHAV-128

The comparison of the GE's of RAGHAV and other existing ciphers are given in the table 10. Figure 7 shows the graphical comparison of the GE's for RAGHAV-128 and other existing ciphers. Table 11 shows that the RAGHAV cipher is 8.36% compact than PRESENT, 48.87 % compact than AES and so on.

Algorithm	Block size [Bit]	Key size [Bit]	Type	Technology used in µm	Serialized architecture based GEs	Round based GEs
SEA	64	128	Feistel	0.13	-	3758
HIGHT	64	128	GFN	0.35	-	3048
AES	128	128	SPN	0.13	2400	-
DESX	64	128	Feistel	0.18	2629	-
CLEFIA	64	128	GFN	0.13	2488	-
TEA	64	128	Feistel	-	3872	-
DESXL	64	128	Feistel	0.18	2168	-
PRESENT	64	128	SPN	0.13	1339	1884
SIMON	64	128	Feistel	0.13	954	-
SPECK	64	128	Feistel	0.13	996	-
LED	64	128	SPN	0.18	1265	-
RECTANGLE	64	128	SPN	0.13	1787	-
PICCOLO	64	128	GFN	0.13	818	1362
RAGHAV	64	128	SPN	0.13	1227	1458

Table 10: Comparison of GEs with existing ciphers

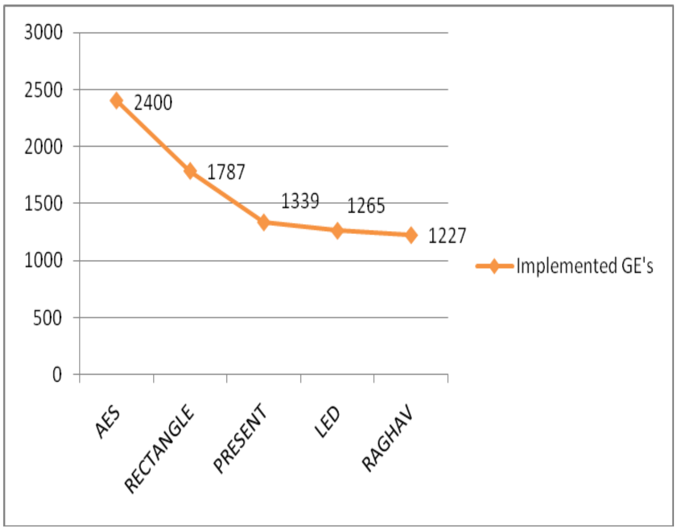

FIGURE 7: GEs COMPARISON WITH EXISTING LIGHTWEIGHT CIPHER

	AES	LED	RECTANGULAR	PRESENT
RAGHAV	-48.87%	-3%	-31.33%	-8.36%

Table 11: GEs Comparison of RAGHAV

From above comparison and analysis it is concluded that the RAGHAV cipher achieves the best results in requirement of Gate Equivalent compared to other existing ciphers.

B. Memory Requirement

The RAGHAV cipher results in the less FLASH memory size which results in the better software performance. In this design for software performance RAGHAV has been tested on ARM 7 - LPC2129 processor. So the same processor has been considered for other existing ciphers also to calculate their memory size. Figure 8 shows the graphical comparison of the RAGHAV with other existing ciphers in terms of flash memory and RAM.

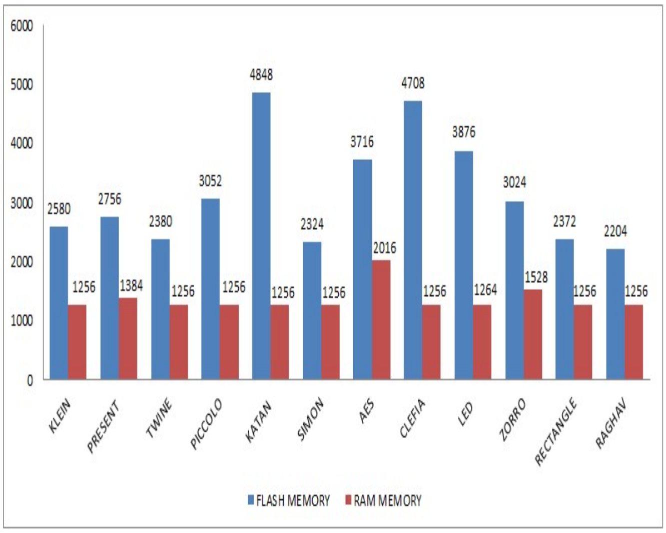

Figure 8: Flash memory and RAM memory Comparison of Standard algorithms with RAGHAV Cipher implemented on LPC2129

In terms of memory requirement the RAGHAV has 20.03 % superior than PRESENT, 43.13 % superior than LED and so on. Table 12 shows comparison of requirement of a flash memory of RAGHAV cipher with other existing ciphers.

	PRESENT	LED	SIMON	TWINE	CLEFIA
RAGHAV	-20.03%	-43.13%	-5.16%	-7.39%	-53.18%

Table 12: A Memory Requirement Comparison Of RAGHAV Cipher With Existing Ciphers

C. Throughput

Throughput decides the speed of the execution of the algorithm as the speed increases the throughput is also more. Here the execution time taken as the time required to execute complete 24 round of RAGHAV cipher. As the execution time increase the throughput decreases. In the RAGHAV cipher the execution time required for 25 round is 1043.77 usec and the throughput is 61.31 Kbps so this is the highest throughput in all other existing ciphers. Table 13 shows the comparison of the throughput with SP network.

Ciphers	**Block Size**	**Key Size**	**Execution Time (In uSec)**	**Throughput (In Kbps)**	**No. of Cycles**
SP NETWORK					
LED	64	128	7092.86	9	425572
KLEIN	64	96	887.51	72	10650.12
RAGHAV	**64**	**128**	**1043.77**	**61.31**	**12525.24**
HUMMINGBIRD-2	16	128	316.51	51	3798.12
PRESENT	64	128	2648.65	24.16	31783.8

Table 13: Calculation of throughput for RAGHAV-128

D. Power Consuption

We have calculated the power consumption by using X-power analyzer tool available in ISE design suit 14.2. Power is calculated with 10MHz frequency and on VIRTEX VI family. Figure 9 represents dynamic power consumption of standard ciphers with comparison of RAGHAV; RAGHAV Cipher consumes 24mw power which is lesser than other lightweight ciphers.

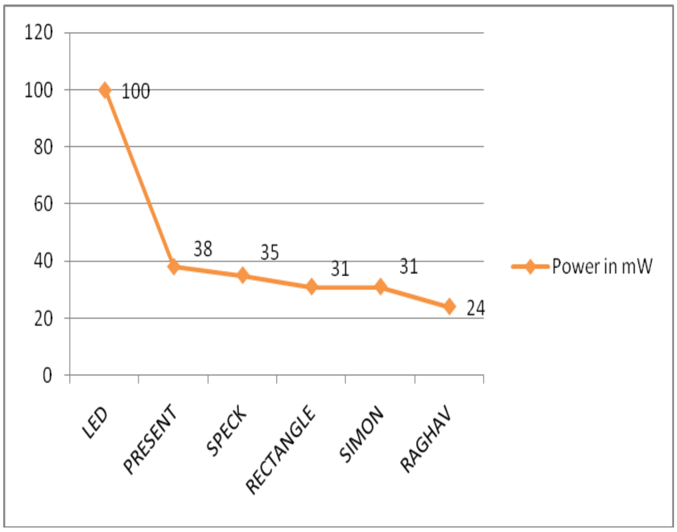

Figure 9: Comparison of power consumption for RAGHAV-128 with existing lightweight ciphers.

Table 14 shows the comparison of the power consumption for RAGHAV-128 with existing lightweight ciphers. Table 15 shows the comparison in percentage. In terms of percentage RAGHAV-128 is 76 % superior than LED, 36.84% superior than PRESENT and so on. So it is conclude that RAGHAV cipher has very less power consumption as compared to other existing lightweight ciphers.

Ciphers	Block Size	Key Size	Power Consumption (In mW)
LED	64	128	100
PRESENT	64	128	38
SPECK	64	128	35
RECTANGLE	64	128	31
SIMON	64	128	31
RAGHAV	**64**	**128**	**24**

Table 14: Calculation of power consumption for RAGHAV-128

	LED	PRESENT	SPECK	RECTANGLE	SIMON
RAGHAV	-76 %	-36.84 %	-31.42 %	-22.58 %	-22.58%

Table 15: GEs comparison of RAGHAV in percentage

VI. Conclusion

In this paper, we have proposed the robust S-P network cipher named as "RAGHAV" which results in good linear and differential complexity, and also results in more number of active S-Box for minimum number of rounds. RAGHAV cipher needs only 1227 GEs for 128 bit key scheduling which is less, as compared to most of the existing lightweight ciphers. We believe that RAGHAV is the smallest S-P design till date in terms of Gate Equivalent, Memory requirement and power consumption. In addition to constrained metrics like GEs, Memory and Power, RAGHAV also proves its strength by resisting basic as well as advanced attacks. This design will prove to be a crusader in making the technologies like IoT feasible and will have a positive impact in the field of lightweight cryptography.

Test Vector of LiCi with 128 bit key

Plaintext	Key	Cipher text
00000000 00000000	00000000 00000000 00000000 00000000	0f7d8e1d5184d11a
00000000 00000000	FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF	4cf9eeaf5bbad078

Acknowledgments

I would like to thank Swapnil Sutar,JRF,CR Rao Institute, Hyderabad for his great assistance and efforts for completing this paper in time. Also, I would also like to thank Abhijit Patil, IBM,Chennai, Jagdish Patil and Dr. Narayan Pisharoty for valuable suggestions and the help they gave time to time.

References

[1] A. Bogdanov, G. Leander, L.R. Knudsen, C. Paar, A. Poschmann, M.J.B. Robshaw, Y. Seurin, and C. Vikkelsoe, "PRESENT - An Ultra-Lightweight Block Cipher," In P. Paillier and I. Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems — CHES 2007*, Vol. 4727 in *LNCS*, pp. 450-466, Springer Berlin Heidelberg, 2007.

[2] F. Abed, E. List, S. Lucks, and J. Wenzel. Cryptanalysis of the speck family of block ciphers. Cryptology ePrint Archive, Report 2013/568, 2013. http://eprint.iacr.org/.

[3] KyojiShibutani, TakanoriIsobe, HarunagaHiwatari, Atsushi Mitsuda, Toru Akishita, and TaizoShirai,"Piccolo: An Ultra-Lightweight Blockcipher", pp. 342-357, Volume-6917 Springer Berlin Heidelberg, 2011.

[4] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. TWINE: A Lightweight Block Cipher for Multiple Platforms. In Selected Areas in Cryptography (SAC), volume 7707 of LNCS, pages 339–354. Springer, 2012.

[5] Zhang, W., Bao, Z., Lin, D., Rijmen, V., Yang, B., Verbauwhede, "RECTANGLE: A bit-slice ultra-lightweight block cipher suitable for multiple Platforms" *Cryptology ePrint Archive, Report 2014/084, 2014.* Available at https://eprint.iacr.org/2014/084.pdf

[6] J. Guo, T. Peyrin, A. Poschmann, and M. Robshaw, "The LED Block Cipher," In *Cryptographic Hardware and Embedded Systems CHES 2011, LNCS*, Vol. 6917/2011, pp. 326-341, Springer, 2011.

[7] L. Yang , M. Wang , S. Qiao , Side Channel Cube Attack on PRESENT, in: Pro-ceeding of Cryptography and Network Security- CANS 2009, Springer, 2009, pp. 379–391.

[8] D Coppersmith, "The Data Encryption Standard (DES) and its strength against attacks", IBM Thomas J Watson Research Center technical report RC 18613 (81421), 22 December 1992
[9] M. Matsui, “Linear Cryptanalysis Method for DES Cipher,” Advances in Cryptology, Proc. Eurocrypt’93, LNCS 765, T. Helleseth, Ed., Springer-Verlag, 1994, pp. 386–397.
[10] F. Chabaud and S. Vaudenay, Links between differential and linear cryptanalysis, Advances in Cryptology, EUROCRYPT’94, Lecture Notes in Computer Science, Springer-Verlag, New York, 950 (1995) pp. 356-365.
[11] Howard M. Heys, “A Tutorial on Linear and Differential Cryptanalysis” ,http://citeseer.nj.nec.com/443539.html.
[12] E. Biham, A. Shamir, Differential Cryptanalysis of the Data Encryption Standard, Springer-Verlag, 1993.
[13] E. Biham, A. Shamir, “Differential Cryptanalysis of DES-like Cryptosystems,” Journal of Cryptology, Vol. 4, No. 1, 1991, pp. 3–72.
[14] Bogdanov, A., Rijmen, V.: “Zero Correlation Linear Cryptanalysis of Block Ciphers” *IACR Eprint Archive Report 2011/123 (March 2011).*
[15] Jeong, K., Kang, H., Lee, C., Sung, J., Hong, S.: “Biclique Cryptanalysis of Lightweight Block Ciphers PRESENT, Piccolo and LED”, *Cryptology ePrint Archive, Report 2012/621*.
[16] M. Albrecht, C. Cid. “Algebraic techniques in differential cryptanalysis” *FSE 2009.* LNCS, vol. 5665, pp. 193-208. Springer, Heidelberg. 2009.
[17] E. Biham. “New Types of Cryptanalytic Attacks Using Related Keys”. *Proceedings of Eurocrypt 93.*LNCS.vol. 765, pp 398-409, Springer-Verlag. 1994.
[18] A. Biryukov and D. Wagner. “Advanced Slide Attacks”.*Proceedings of Eurocrypt 2000.*LNCS.vol. 1807, pp. 589-606, Springer-Verlag. 2000. [19] A. Biryukov, D. Khovratovich and I. Nikoli´c. “Distinguisher and Related-Key Attack on the Full AES-256”. http://eprint.iacr.org/2009/241. 2009.
[19] Z. Shi and R. B. Lee, “Bit permutation instructions for accelerating software cryptography,” In Proceedings of the IEEE International Conference on Application Specific Systems, Architectures and Processors (ASAP 2000), pp. 138-148, July 2000.
[20] Gaurav Bansod., NishchalRaval., Narayan Pisharoty.: “Implementation of a New Lightweight Encryption Design for Embedded Security”, IEEE Transactions on Information Forensics and Security, Issue 1,Vol 10,Jan 2015.
[21] A. Poschmann. Lightweight Cryptography - Cryptographic Engineering for a Pervasive World. Number 8 in IT Security. Europ¨aischerUniversit¨atsverlag, 2009.Published: Ph.D. Thesis, Ruhr University Bochum.
[22] Bansod, Gaurav, Narayan Pisharoty, and Abhijit Patil."PICO: An Ultra Lightweight and Low Power Encryption Design for Ubiquitous Computing."*Defence Science Journal* 66.3 (2016): 259-265.
[23] Wu, W., Zhang, L. “L-Block: A Lightweight Block Cipher”. In: Lopez, J., Tsudik, G. eds. (2011) *Applied Cryptography and Network Security. Springer, Heidelberg, pp. 327-344.*
[24] M Kumar, SK Pal and APanigrahi. “FeW: A Lightweight Block Cipher”. Scientific Analysis Group, DRDO, Delhi, INDIA, Department of Mathematics, University of Delhi, INDIA2014.
[25] KyojiShibutani, TakanoriIsobe, HarunagaHiwatari, Atsushi Mitsuda, Toru Akishita, and TaizoShirai,“Piccolo: An Ultra-Lightweight Blockcipher”, pp. 342-357, Volume-6917 Springer Berlin Heidelberg, 2011.
[26] G Bansod, A Patil, S Sutar, N Pisharoty “*An Ultra Lightweight Encryption Design for Security in Pervasive Computing*”, Big Data Security on Cloud (Big Data Security), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016.
[27] G BANSOD, N PISHAROTY, A PATIL, “BORON: an ultra lightweight and low power encryption design for pervasive computing”, Frontiers of Information Technology and Electronic Engineering, 2016,**DOI:** 10.1631/FITEE.1500415 .
[28] R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, L. Wingers. The SIMON and SPECK Families of Lightweight Block Ciphers. In IACR eprint archive. Available at https://eprint.iacr.org/2013/404.pdf.
[29] Subhadeep Banik, Andrey Bogdanov, Takanori Isobe, Kyoji Shibutani, Harunaga Hiwatari, Toru Akishita, and Francesco Regazzoni, “Midori: A Block Cipher for Low Energy (Extended Version)”, In IACR eprint archive. Available at https://eprint.iacr.org/2013/404.pdf.

Gaurav Bansod received the P.hD degree in Embedded security from Symbiosis International University, Pune, India. He also received M.Tech. Degree in Embedded Sytems from Jawaharlal Nehru Technological University, Hyderabad, India in 2008. His supervisor is Dr. Narayan pisharoty who has received P.hD degree from Carnegie Mellon University, USA. across India. He has publications in reputed IEEE Transactions on Information Forensics and Security , Springer and Wiley. He has published other papers in SCI indexed journals. His research area includes low power cryptographic design, embedded system and hardware and software design.