International Journal of Security and Its Applications
Vol. 10, No. 12 (2016), pp.111-128
http://dx.doi.org/10.14257/ijsia.2016.10.12.10

A New Ultra Lightweight Encryption Design for Security at Node Level

Dr. Gaurav Bansod

Pune Institute of Computer Technology
gaurav249@gmail.com

Abstract

This paper proposes a new lightweight cipher VAYU. VAYU has a balanced Feistel structure. VAYU cipher supports 64 bit plaintext and 128/80 bit key length and it has a total of 31 rounds. It needs only 1290 GEs for 128 bit key length. It also results in minimal memory size as compared to all other existing lightweight ciphers. This paper discusses the security analysis of VAYU cipher design which is adequate against linear and differential cryptanalysis, Biclique attack, zero correlation attack, algebraic attack. VAYU cipher design will be best suitable for applications like IoT, smart Wireless Sensor networks. VAYU cipher uses two F-functions with substitution box, which results in a high diffusion mechanism.

Keywords: *Lightweight cryptography, Feistel cipher, Block cipher, IoT, Encryption, Embedded security, ubiquitous computing*

1. Introduction

Lightweight ciphers are popular for applications like RFID, IoT, smart wireless sensor network, wireless body area network. In applications like IoT, security is essential for protecting sensitive data. RFID needs nearly 10000 (Gate Equivalents) GEs for its hardware implementation. For providing security in constrained devices GEs should not be more than 2000-2200. AES and DES are the encryption standards which have used nearly 2400-3500 GEs. This led to the emergence of the field of lightweight cryptography. There are many existing ciphers which are lightweight in terms of memory requirements. Most of the ciphers are block ciphers because they allow a high diffusion mechanism. Block ciphers are divided into Feistel networks and SP networks. PRESENT [1], PRIDE [2], PRINCE [3], Rectangle [4], LED [5] are the SP networks based block cipher. CLEFIA [6], HIGHT [7], L-Block [8], XTEA [9] are the Feistel based block ciphers. Feistel network can be classified into two categories, Classical Feistel structure and Generalized Feistel structure. VAYU cipher which is proposed in this paper uses Classical Feistel structure, where block size is divided into two equal halves. Classical Feistel structure has many advantages as compared to Generalized Feistel structure because the Generalized Feistel structure requires more number of round functions for providing adequate amount of security. PRESENT, SIMON [10], SPECK [10], and RECTANGLE ciphers have lesser GEs which is a dominant requirement in lightweight cipher design. PRESENT cipher has a strong permutation layer while RECTANGLE cipher is designed with the strong cryptanalysis properties. SIMON and SPECK ciphers are the most compact ciphers and it has better software and hardware performance as compared to other existing lightweight cipher designs.

In this paper, we have proposed an ultra-lightweight cipher which has lesser GEs and less memory size. VAYU cipher has data complexity 260 and generates maximum number of active S-boxes in minimum rounds. It also has a robust Permutation layer. All these metrics will make the VAYU cipher relevant to small scale embedded devices where memory and power consumption are constraints.

ISSN: 1738-9976 IJSIA
Copyright ⓒ 2016 SERSC

For VAYU cipher, we have used the following notations

PT	64-bit input plaintext block
CT	64-bit output cipher text block
RKi	32-bit Round sub key for round i
F	Function
⊕	Bitwise exclusive-OR operation
<<<n	Left cyclic shift by n bits
>>>n	Right cyclic shift by n bits
RCi	Round counter i
||	Concatenation of two strings
!	Bitwise NOT operation
64 bits	Maximum length of plain text
MSB	Most significant bits
LSB	Least significant bits
←	Assigner
●	Multiplication operator

2. The Block Cipher Vayu

VAYU is a 31 round Feistel based network. It supports 80/128 bits key and has a block length of 64-bits. Figure 1 shows the round function of VAYU cipher.

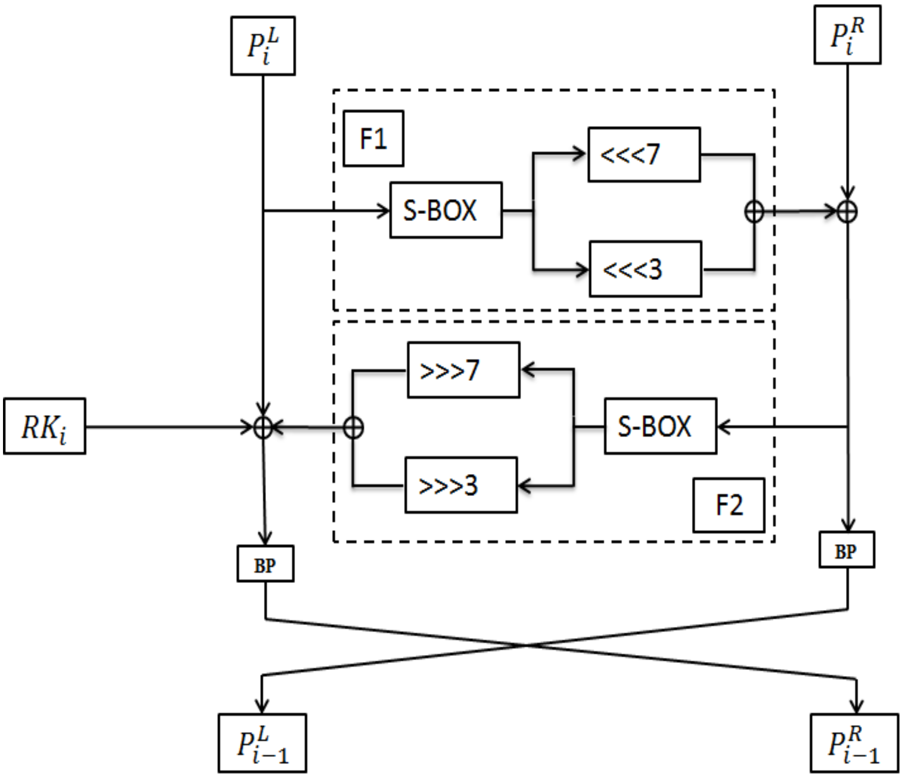


Figure 1. Round Function of VAYU Cipher

The block length of 64-bit is divided into two equal halves each of 32-bit length. It has two F-functions (F1 & F2) which have substitution box and shift operators included in them. The S-Box shown in above Figure 1 consists of 8 S-Box's which can process 32-bits input plaintext to F1 F-function simultaneously where single S-Box has 4 bit input and generates 4 bit output. The results are EX-ORed with another 32-bit block as shown

Copyright © 2016 SERSC

in Figure 1. Another F-function has the same substitution box and shift operators to generate 32-bit block which is EX-ORed with the round key. Key scheduling algorithm is used to generate the round key for each round function.

2.1. Flow of Encryption

Block length of 64-bits is divided into two 32-bits plaintextP_i^L andP_i^R. P_i^L consist of MSB-32 bit and P_i^R consist of LSB-32 bit.

$$\text{PT} \leftarrow P_i^L \| P_i^R$$

1. Apply F-function on P_i^L

$$\text{P} \leftarrow \text{F1}\ (P_i^L)$$

2. EXOR with P_i^R

$$P_{i1} \leftarrow \text{P} \oplus P_i^R$$

3. Apply another F-function on P_{i1}

$$P_{i2} \leftarrow \text{F2}\ (P_{i1})$$

4. EXOR P_{i2} with round key RK_i and P_i^L

$$P_{i3} \leftarrow P_{i2} \oplus P_i^L \oplus RK_i$$

5. Apply permutation layer to P_{i1} and P_{i3}

$$P_{i+1}^R = \text{BP}[P_{i3}]$$
$$P_{i+1}^L = \text{BP}[P_{i1}]$$

After 31 rounds we will get cipher text as

$$\text{CT} \leftarrow P_{31}^L \parallel P_{31}^R$$

2.2. F-function

VAYU cipher has two identical F-functions except direction of shifting. In F1 F-function, P_i^L is input to the S-box and the output of the S-box undergoes left circular shift by 7 and left circular shift by 3 on P_i^L. Output of the shifting operation undergoes EX-OR. This output is fed to the S-box of F2 F-function as shown in Figure 1. The output of S-box is again fed to shift operators and is EX-ORed with the key. The operation of both F-Functions is depicted in Figure 1.

2.3. S-box

The S-box used in VAYU cipher design is of 4-bit to 4-bit S-box S: $F_2^4 \rightarrow F_2^4$ *i.e.*, F is function which have performed the substitution of 4 input bits and ‘2’ shows it binary input on single trail. Table 1 represents the hexadecimal values for the Substitution layer,

Table 1. S-box of VAYU Cipher

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

2.4. Bit Permutation (BP)

Bit permutation layer of VAYU performs the swapping of 32 bits of the PiL. The operation of the bit permutation is depicted in Table 2 where ‘*i*’ shows the original bit position and BP [i] represents new position to which bit will get shift. BP[*i*] stands for permutation of i. The combination of circular shift operation along with the bit permutation helps to increase the active S-box count. The S-box with non-zero input or non-zero output is referred as “Active S-box”. Diffusion level of cipher gets improved with this active S-box count. We have followed the following criteria for the design of bit permutation layer as illustrated in paper [12]. This bit permutation layer shuffles the 32 bits such that minimum number of active S-box will get increased, which results in high diffusion. Bit permutation helps to make the robust round function

- 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.
- All output trails from each S-box at round ‘r’ affect the four different S-boxes in ‘r+1’.

Table 2. Bit Permutation Table For VAYU

i	00	01	02	03	04	05	06	07
BP[*i*]	24	08	00	19	23	28	12	04
i	08	09	10	11	12	13	14	15
BP[*i*]	09	25	18	01	13	05	22	29
i	16	17	18	19	20	21	22	23
BP[*i*]	17	02	10	26	06	14	30	21
i	24	25	26	27	28	29	30	31
BP[*i*]	03	16	27	11	31	20	07	15

2.5. Encryption Algorithm

```
Input: Plaintext: $A_{64} \leftarrow a^{63} a^{62} a^{61} a^{60} ... a^{3} a^{2} a^{1}a^{0}$,
S [16], BP [32],RKi[32], Pt1, Pt2, Pt3, Pt4, Pt5, Pt

Output: Cipher text: $C_{64}$

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

$Pt1 \leftarrow Sbox[P_i^L]$
$Pt2 \leftarrow [LCS(Pt1,7) \oplus LCS(Pt1,3)]$
$Pt3 \leftarrow Pt2 \oplus P_i^R$
$Pt4 \leftarrow Sbox [Pt3]$
$Pt5 \leftarrow [RCS(Pt4,7) \oplus RCS(Pt4,3)]$

    $Pt \leftarrow Pt5 \oplus P_i^L \oplus RKi$
    $P_{i+1}^R \leftarrow P[P_i^L]$
    $P_{i+1}^L \leftarrow P[Pt5]$

$A_{64} \leftarrow P_{i+1}^L \,||\, P_{i+1}^R$
```

```
i = i+1
End

$C_{64} \leftarrow A_{64} \leftarrow P_{31}^{L} || P_{31}^{R}$
```

2.6. Key Schedule of 80-bit and 128-bit Key Length

The VAYU cipher's key scheduling is motivated from the PRESENT cipher key scheduling design. No attacks till date are reported on the PRESENT cipher key scheduling. In the VAYU cipher, key scheduling algorithm generates total of 31 sub-keys each of size 32 bits[31].

1. 128-bit key scheduling

A user defined 128-bit key is stored in the register KEY, 64-bit LSB's from KEY register is extracted as follows

$$RKi = K_{31}\, K_{30} \ldots K_2\, K_1\, K_0$$

$$KEY = K_{127}\, K_{126}\, K_{125} \ldots K_2\, K_1\, K_0$$

After extracting key of 64-bits, register KEY is updated in the following manner

- KEY <<< 13 (KEY left circular shifted by 13).
- $[K_3\, K_2\, K_1\, K_0] \leftarrow S\, [K_3\, K_2\, K_1\, K_0]$
- $[K_7\, K_6\, K_5\, K_4] \leftarrow S\, [K_7\, K_6\, K_5\, K_4]$
- $[K_{63}\, K_{62}\, K_{61}\, K_{60}\, K_{59}] \leftarrow [K_{63}\, K_{62}\, K_{61}\, K_{60}\, K_{59}] \oplus RC^{i}$

For 0 to 24 rounds, 5-bits of round counter 'i' is XOR-ed with the 5-bits of key register KEY *i.e.*, from $K_{59}$to K_{63}.

2. 80 bit Key scheduling

A user defined 80-bit key is stored in key register KEY and LSB bits from it are used as round subkeys.

$$RKi = K_{31}\, K_{30} \ldots K_2\, K_1\, K_0$$

$$KEY = K_{79}\, K_{78}\, K_{77} \ldots K_2\, K_1\, K_0$$

After extracting 64-bit key, register KEY is updated as follows

- KEY<<< 13(KEY left circular shifted by 13).
- $[K_3\, K_2\, K_1\, K_0] \leftarrow S\, [K_3\, K_2\, K_1\, K_0]$.
- $[K_{63}\, K_{62}\, K_{61}\, K_{60}\, K_{59}] \leftarrow [K_{63}\, K_{62}\, K_{61}\, K_{60}\, K_{59}] \oplus RC^{i}$

2.6. Design Criterion of S-box

Compactness and resistance against linear and differential attacks are the two parameters we have considered while designing the S-box. VAYU S-box is a 4-bit to 4-bit *i.e.*, S: $F_2^4 \leftarrow F_2^4$, It means that it takes a 4-bit input and produces a 4-bit output. Properties essential for a good S-box design are mentioned below. These properties are previously mentioned in RECTANGLE, PRESENT and DES cipher[31].

A complete design criterion of the VAYU's S-box is as follows:

1. For any nonzero input and output differences $\triangle A$, $\triangle B \in F_2^4$ respectively, we have

$$DC\ (\triangle A, \triangle B) = \#\ \{A \in F_2^4\ |S\ (x) \oplus S\ (x \oplus \triangle A) = \triangle B\} \leq 4$$

Proof: This property shows that maximum value in Difference Distribution Table (DDT) have to be less than or equal to 4 and its differential probability given by $4/16 = 2^{-2}$. Table 4 represents the DDT of VAYU's cipher S-Box, the highlighted box shows the maximum DDT value in the table 4. Value in DDT should have to be less for non-zero input and output difference to provide better security against differential attack. For non-zero input and output difference ideal S-Box should have differential probability equal to 1/16.

2. For any nonzero input and output differences $\triangle A$, $\triangle B \in F_2^4$ respectively we have such that $Hw(\triangle A) = Hw(\triangle B) = 1$, where Hw(x) denotes the Hamming weight of x, we have

$$SetDC = DC\ (\triangle A, \triangle B) = \#\ \{A \in F_2^4\ |S\ (x) \oplus S\ (x \oplus \triangle A) = \triangle B\} = 0$$

Cardinality of SetDC can be given as CarDC, we have CarDC= 2.

Proof: This property indicates that the place in DDT table where hamming weight of input difference and hamming weight of output difference equal to one (i.e. 1,2,4,8), should have zero value ideally. To achieve all value equal to zero is difficult. The total number of non-zero value is called as cardinality of set_{DC} value represent by "CAR_{DC}". These values are highlighted in table 4. We have achieved the value equal to 2 by software algorithm.

3. For any nonzero input sum and output sum such that A, $B \in F_2^4$ so we have LC(A, B)

$$LC\ (A, B) = \#\{x \in F_2^4|A \bullet x = B \bullet S(x)\} - 8| \leq 4$$

Proof: This property shows that the maximum value in Linear approximation table (LAT) should be equal to 4 and probability bias is given as $4/16=2^{-2}$.

4. For any nonzero input sum and output sum such that A, $B \in F_2^4$, such that Hw(a) = Hw(b) = 1, we have

$$SetLC = LC\ (A, B) = \#\{x \in F_2^4|A \bullet x = B \bullet S(x)\} - 8| \neq 0$$

Cardinality of SetLC can be given as CarLC, we have CarLC=2.

Proof: This property indicates that the place in LAT table where hamming weight of input and output mask is equal to 1, should be zero ideally. The input mask and output mask values where hamming weight is zero are 1,2,4,8. To achieve all values at this location in table equal to zero is difficult. We have obtained a value as minimum as possible, which is equal to 2. This parameter is called as cardinality of Set_{LC} represent by CAR_{LC}

5. Bijective *i.e.,* S (a) ≠ S(b) for all values of a ≠ b.

Proof: This property indicates that no value in S-box should repeat as substitution. For *e.g.,* a = 9 & b = 4 and suppose S(9) = A & S(4) = 5 hence S(9) ≠S(4).

6. No static point *i.e.*, S (a) ≠ a for all values of $a \in F_2^4$.

Proof: This property shows that the substitution should not be linear *i.e.*, for any value "a" should not substitute for itself S (a).

3. Security Analysis of VAYU

Cryptanalysis techniques show the strength of the cipher against attacks. This paper focuses on the cryptanalysis attacks like linear cryptanalysis, differential cryptanalysis, algebraic cryptanalysis, Biclique cryptanalysis and zero correlation attack. In VAYU cipher, S-box is the only non-linear element. Selection of S-box is very important in cipher design to make the design robust. Using principal of pilling up lemmas, it is able to find out minimal number of active S-box[31].

3.1. Linear Cryptanalysis

Linear cryptanalysis [13] [14] is most significant and sturdy attacks implemented on block ciphers. It is essential that cipher is required to resist such attacks. It is also called as known plain text attack. It uses the high probability occurrences of linear expression containing plaintext bits, cipher text bits and sub key bits. This expression is used for mounting a linear attack on a cipher. To mount a linear attack, the attacker needs to have the knowledge about a subset of the plaintext and its corresponding cipher text. The attacker will be able to find out trails, ultimately minimum number of active S-box with the help of LAT (Linear Approximation Table). If P_L is the linear probability then its bias can be given as $|P_L-1/2|$, bias (ε) for the VAYU cipher S-box is 2^{-2}. Matsui`s Piling-up lemma [15] is used to calculate probability bias for 'n' rounds[31].

The best way to resist against linear cryptanalysis is,

- Optimizing the bias in the LAT. For ideal S-box bias, values should be 1/8 which is practically not possible to achieve.
- Increase the number of active S-boxes in the cipher structure.

Table 3 represents the minimum number of active S-BOX from differential trail

Table 3. Minimum Number Of Active S-box from Linear Trail

#Round	# Min. active S-boxes
1	0
2	3
3	7
4	18

Theorem1:
For 24 rounds of VAYU, it has total 108 active S-boxes and the total bias for 24 rounds is 2^{-73}.

Proof:
It was found that for 4 rounds VAYU cipher, it has minimum 18 active S-boxes.

Maximum bias for the VAYU cipher S-box is 2^{-2} by using Matsui`s Pilling up Lemma for 4 rounds of VAYU cipher the total bias can be given as,

$$2^{17}\times (2^{-2})^{18}= 2^{-19}$$

By applying the same lemma for 24 rounds the total bias (ε) can be given as,

$$\varepsilon^{=} 2^{5} \times (2^{-19})^{6} = 2^{-109}$$

By calculating required number of known plaintext / cipher text, complexity of linear attack can be computed and can be given as,

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

For 18 rounds of the VAYU cipher, the required number of known plaintext / cipher text can be given as

$$N_L=1/ (\varepsilon)^{2}= 1/ (2^{-109})^{2}$$

$$N_L = 2^{218}$$

The required number of known plaintext / cipher text is 2^{218} which is greater than the available limit *i.e.*, 2^{64}. Hence, the complete round of the VAYU cipher shows a good resistance against a Linear Attack.

3.2. Differential Cryptanalysis

Differential attack [16] [17] is another of the most significant attacks applied by Biham and Shamir on DES in 1990. To mount the differential attack for a specific number of rounds in an encryption system, pairs of high probability input and output occurrences are used to recover the round keys. Nonlinear layer is examined by DDT (Difference Distribution table). If S-box has non zero input difference and non-zero output difference, such S-box referred as active S-box in differential cryptanalysis. Differential probability of VAYU cipher S-box is 2^{-2}.

Table 4 represents the Difference Distribution Table for the VAYU Cipher S-box. There are two approaches for providing security against differential cryptanalysis

- By minimizing the differential probability, for an ideal S-box, this probability is 1/16.
- It is necessary to find a structure that maximizes the minimum number of active S-boxes.

Table 5. Minimum Number of Active S-boxes from Differential Trail

#Round	# Min. active S-boxes
1	0
2	4
3	16
4	27

Table 5 represents minimum number of active S-Boxes obtained from differential trail using Difference Distribution Table. For 4 rounds of the VAYU cipher, there are minimum of 27 active S-boxes. So, for 24 rounds, there will be a minimum of 162 active S-boxes. Total differential probability P_d is given as $(2^{-2})^{162}= 2^{-324}$.

The complexity of the differential attack can be computed by calculating the required number of chosen plaintext / cipher text and is given as,

$$N_d= C/P_d$$

Where C = 1 and $P_d= 2^{-324}$, so the required number of chosen plaintext / cipher text are $N_d=1/2^{-324}= 2^{324}$. The required numbers of chosen plaintext / cipher text is 2^{324} which are greater than available limit *i.e.*, 2^{64}; hence complete rounds of the VAYU cipher provide resistance against a Differential Attack.

Table 4. Difference Distribution for the VAYU Cipher

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	2	2	2	0	2	0	0	2	0	0	4	0	2
2	0	0	0	4	0	0	4	0	2	2	0	0	2	2	0	0
3	0	0	0	2	2	2	0	2	0	4	0	2	0	0	2	0
4	0	0	0	0	0	0	0	0	0	0	4	4	0	0	4	4
5	0	2	4	0	2	0	0	0	0	2	0	0	2	2	2	2
6	0	0	0	0	0	0	4	4	2	2	0	0	2	2	0	0
7	0	2	4	0	2	0	0	0	0	2	2	2	0	2	0	0
8	0	0	0	4	0	0	4	0	0	0	2	2	0	0	2	2
9	0	0	0	2	2	2	0	2	0	0	0	2	4	0	2	0
A	0	0	0	0	0	0	0	0	2	2	2	2	2	2	2	2
B	0	0	0	2	2	2	0	2	4	0	2	0	0	0	0	2
C	0	4	0	0	0	4	0	0	2	2	0	0	2	2	0	0
D	0	2	4	0	2	0	0	0	2	0	0	0	2	0	2	2
E	0	4	0	0	0	4	4	4	0	0	0	0	0	0	0	0
F	0	2	4	0	2	0	0	0	2	0	2	2	2	0	0	0

3.3. Zero-Correlation Attack [18]

Cipher should resist a zero correlation attack. It is an extension of linear cryptanalysis. In this paper matrix method [19] is demonstrated as it is an automatic tool to find zero correlation approximations. To mount zero correlation attack, we have put an arbitrary non-zero value at the input side of round function shown in Figure 1 denoted here by ‘a’ and continued all the operations till 3 rounds.

While going through the round function we have come across the situation where summing point or XOR operation occurred in midway. We have followed the lemma principle and arithmetic table given in section 4.1 of [19] for obtaining data on trails connected with summing and XOR branching operation.

For (0000 0000 0000 000a 0000 0000 0000 0000) → (0000 0000 0b00 0000 0000 0000 0000 0000) has correlation exactly zero for which the values a and b are non-zero. Trails for zero correlation attack are shown in table 6 and the contradiction was found at round 3 for the VAYU cipher. 3^{rd} round of VAYU cipher design shows contradiction.

Table 6. Trails for Zero-correlation of VAYU Cipher

	P_L^i	P_R^i
0	0000 0000 0000 0000 0000 000a 0000 0000	0000 0000 0000 0000 0000 0000 0000 0000
1	$\overline{0}000\ 0\overline{0}00\ 0000\ 0000$ 0000 0000 0000 0000	0000 0000 0a00 0000 0000 0000 0000 0000
2	$000\overline{0}\ 00\overline{0}0\ 0000\ 000\overline{0}$ $0000\ 000\overline{0}\ 0{*}00\ \overline{0}0\overline{0}0$	$0000\ \overline{0}000\ 0000\ 0000$ $0000\ 0000\ \overline{0}000\ 00\overline{00}$

3	$\overline{00}$*0 0*$\overline{0}$* 0*$\overline{0}$* *00* 0000 000$\overline{0}$ $\overline{0}$0*0 0*0*	0000 0000 $\overline{00}$*0 $\overline{0}$0$\overline{00}$ 0$\overline{0}$*$\overline{0}$ $\overline{0}$0$\overline{00}$ 000$\overline{0}$ 00$\overline{0}$0
3	$\overline{00}$0* **0* **0$\overline{0}$ **0$\overline{0}$ *00* **$\overline{0}$* **$\overline{00}$ *$\overline{0}$00	***0 0**0 $\overline{0}$*00 0*00 *$\overline{0}$*$\overline{0}$ $\overline{0}$*$\overline{0}$0 0000 0$\overline{00}$0
4	**00 $\overline{00}$00 $\overline{0}$000 $\overline{0}$000 0000 0000 000* *$\overline{0}$0*	0000 0$\overline{0}$00 0000 0000 0000 0000 **0$\overline{0}$ 00$\overline{00}$
5	00*0 00$\overline{0}$0 0000 0000 0000 0000 0000 00$\overline{0}$0	0000 0000 0000 0000 0000 0000 000$\overline{0}$ 000$\overline{0}$
6	0000 0000 0000 0000 0000 0000 0000 0000	0000 0000 0000 0000 0000 000b 0000 0000

3.4. Biclique Attack [20]

This attack is an extension of Meet-In-The-Middle attack(MITM).In this paper, Biclique attack is mounted on VAYU-128. A 3-dimensional Biclique was constructed for round 28 ~ 31 of VAYU-128. For these rounds, the partial keys used are (K^{29}, K^{30}, K^{31}) which are described as follows:

$K^{29} = K_{38}, K_{37}\ldots\ldots\ \ldots.. K_{7}$
$K^{30} = K_{25}, K_{24},\ldots K_{0},K_{127}\ \ldots.\ K_{122}$
$K^{31} = K_{12}, K_{11},\ldots K_{0,}K_{127}\ \ldots..\ K_{109}$

From the above equations, it was found that by varying the following sub keys (K_{38},K_{37},K_{36}) and (K_{17},K_{16},K_{15}), it gives Biclique on the full VAYU-80. To construct the Δi-differential, sub keys (K_{38},K_{37},K_{36}) have been considered and denoted by red key while for the ∇j-differential, sub keys (K_{17},K_{16},K_{15}) have been considered and denoted by blue key. Let, f is a sub-cipher from round 28 to round 31[31]. The Δi-differential affects the 64-bits of the cipher text as shown in Figure 2. The data complexity does not exceed 2^{64}. Red color arrows at the 31st round in Figure 2 represent the data complexity. The total computational complexity of VAYU is computed as follows.

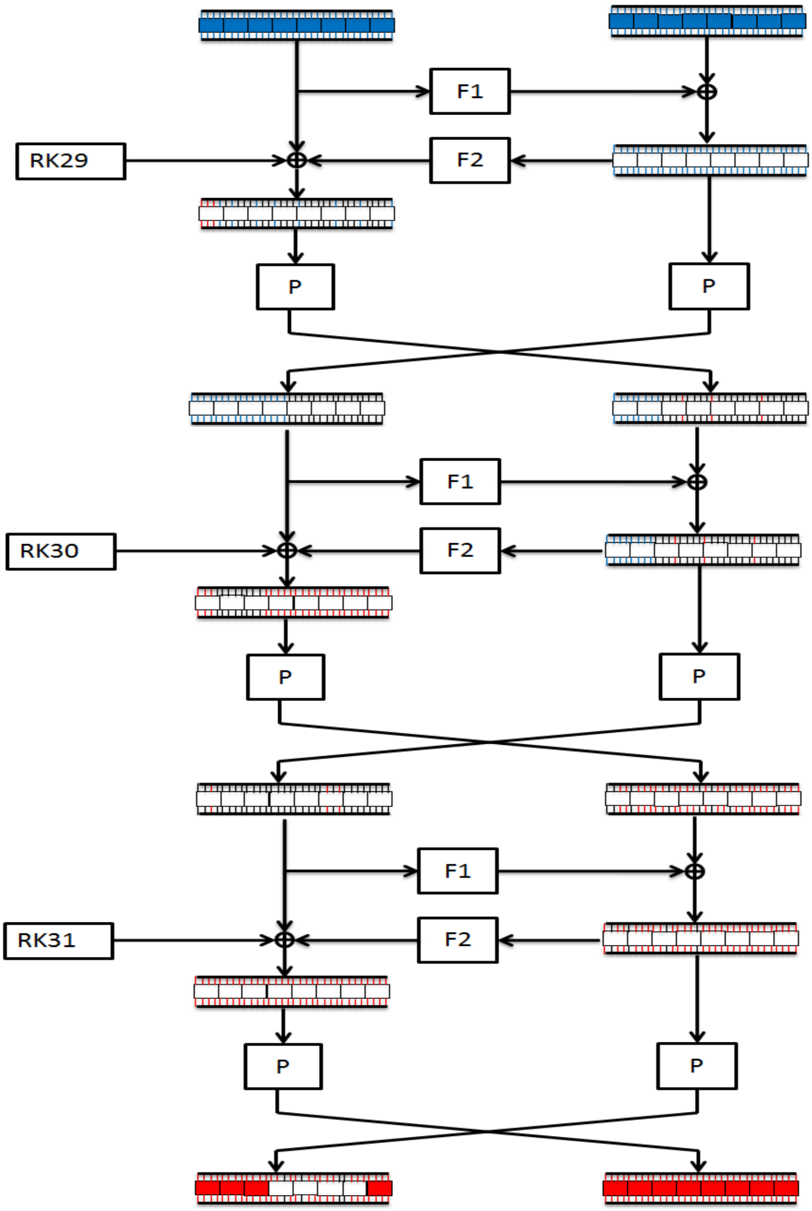


Figure 2. 3 - Dimensional Biclique for VAYU Cipher

$C_{total} = 2^{k-2d} (C_{Biclique} + C_{precomp} + C_{recomp} + C_{falsepos}) = 2^{127.95}$

Where,

C_{total} = total computational complexity,

$C_{Biclique}$ = Biclique computational complexity,

$C_{precomp}$= Pre-computational complexity,

C_{recomp} = Re-computational complexity,

$C_{falsepos}$ = Falsepos complexity,

k = number of key bits used for key scheduling,

Copyright ⓒ 2016 SERSC

d = dimension of key selected.

To mount the Biclique attack the first step is the proper selection of keys. If 3 keys are selected for Biclique mounting then it's known as 3 dimensional Biclique.. There are following key selection criteria that one has to follow to do the successfully mounting of Biclique attack:

- Red key selected for round in Biclique is same as the red key selected for calculation of backward computation in MITM.
- Blue key selected for round in Biclique is same as the blue key selected for calculation of forward computation in MITM
- The care must be taken the blue and red part should not mix up or overlap with each other means both the keys should not activate the same S-box or a bit.

3.5. Avalanche Effect

If output changes considerably with changes in single bit at input side, resulting in avalanche effect. Most of the robust design resulting in drastic change in cipher text with single bit changes in input. Poor randomization occurs when a block cipher does not show the avalanche effect to a significant degree.

The output obtained by applying a single bit change in the input plaintext / Key bits was observed. It was found that in the case of the VAYU cipher any single bit change in key or plaintext results in 50% cipher text change *i.e.*, more than half of the bits of cipher text. Table 7 summarizes the Avalanche Effect. There will be tolerance of +/- negligible amount. We have change each bit of VAYU by algorithm and test the avalanche effect which results in 50% bits change with negligible deviation[31].

Table 7. Avalanche Effect for VAYU-128

Plaintext	0000 0000 0000 0000	# of bits changes
Key Cipher text	0000 0000 0000 0000 0000 0000 0000 0000 8ae0 563b 5d25 1cbf	
Key Cipher text	0000 0001 0000 0000 0000 0000 0000 0000 480f 8c18 c6fd 8e8a	34

Keeping key constant and changing a bit in plaintext

Plaintext	0000 0000 0000 0000	# of bits changes
Key	0000 0000 0000 0000 0000 0000 0000 0000	
Plaintext Cipher text	0000 0000 0000 0000 8ae0 563b 5d25 1cbf	
Plaintext Cipher text	0000 0000 0040 0000 5b7f 05d4 bbf1 ec8b	36

4. Security Comparison with Standard Algorithms

In this section, the VAYU cipher has been compared with the other existing lightweight ciphers. Table 8 compares the linear complexity and differential complexity by considering the minimum number of active S-boxes for rounds[31].

Table 8. Linear & Differential Attack Comparison

Cipher Name	VAYU	PRESENT	L-Block	FEW	PICCOLO
#Rounds	24	25	15	27	30
#Known Plaintext	2^{218}	2^{102}	2^{66}	2^{90}	2^{120}
#Chosen Plaintext	2^{324}	2^{100}	2^{64}	2^{90}	2^{120}
Reference	This paper	[1]	[27]	[28]	[29]

Table 9 compares the data complexity and computational complexity of VAYU with other ciphers.

Table 9. Biclique Attack Comparison

Cipher Name	Rounds	Data Complexity	Computational Complexity	Reference
VAYU-128	Full(31)	2^{60}	$2^{127.95}$	This Paper
PRESENT-80	Full(31)	2^{23}	$2^{79.54}$	[20]
PRESENT-128	Full(31)	2^{19}	$2^{127.42}$	[20]
PICCOLO-80	Full(25)	2^{48}	$2^{79.13}$	[20]
PICCOLO-128	Full(31)	2^{24}	$2^{127.35}$	[20]
LED-64	Full(48)	2^{64}	$2^{63.58}$	[20]
LED-80	Full(48)	2^{64}	$2^{79.37}$	[20]
LED-96	Full(48)	2^{64}	$2^{95.37}$	[20]
LED-128	Full(48)	2^{64}	$2^{127.37}$	[20]

Copyright ⓒ 2016 SERSC

5. Hardware and Software Performance of VAYU Cipher

Design of VAYU cipher is constructed in such a way that it results in optimum hardware and software performance. The compact structure of the VAYU cipher results in a small footprint area and requires lesser memory size in software. The 32-bit ARM 7 LPC2129 processor was considered for analyzing the software performance of the VAYU cipher.

Footprint size(GEs) is computed with the ARM standard cell library for the IBM 8RF (0.13 micron). The area 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.All other ciphers are written in Embedded C and implemented on a 32-bit processor for comparison with the VAYU cipher. Figure 3 represents the memory comparison of the existing lightweight ciphers with the VAYU cipher.

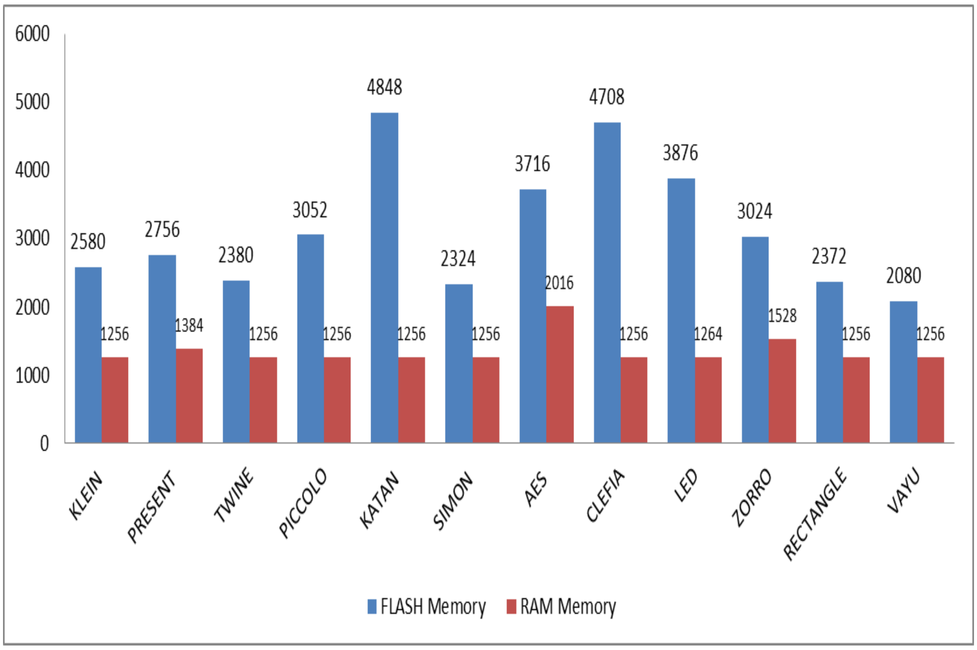


Figure 3. Flash memory and RAM memory Comparison of existing cipher with VAYU Cipher implemented on LPC2129

Table 10 shows the comparison of existing ciphers with VAYU [29] [30], which shows that VAYU's memory decreases minimum by 10.5% as compared to SIMON.

Table 10. A Memory Requirement Comparison Of VAYU Cipher With Existing Ciphers

	PRESENT	LED	SIMON	TWINE	CLEFIA
VAYU	-24.52%	-46.33%	-10.49%	-12.60%	-55.81%

Serialized architecture data path for VAYU cipher based on serial ASIC architecture where single S-Box can use many times which is mentioned in 5.1.1 and depicted in figure 5.3 of [30]. Data path for VAYU cipher is shown in Figure 4.Where A, A', B, K are the connector used to connect the data path[31].

Copyright ⓒ 2016 SERSC

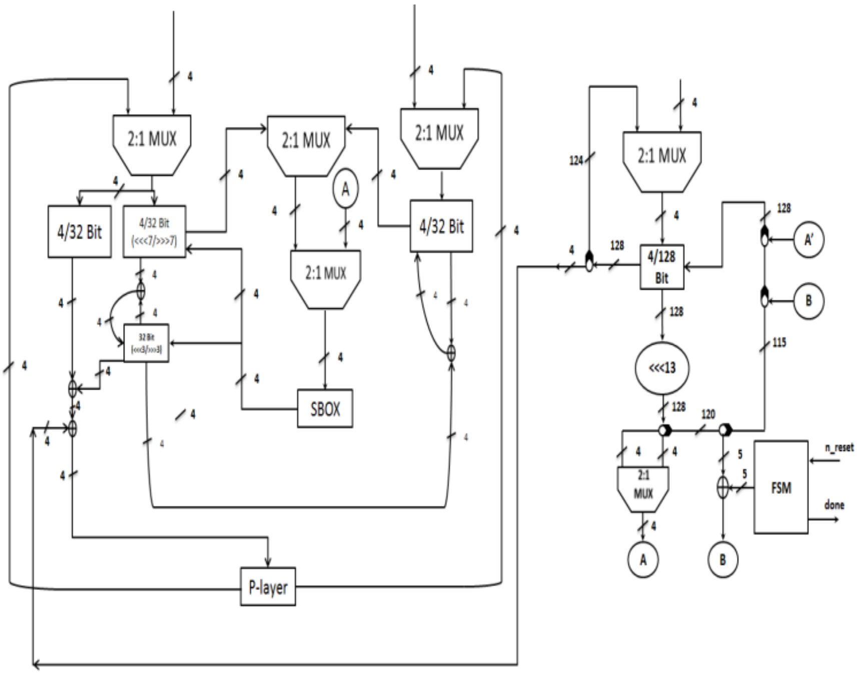


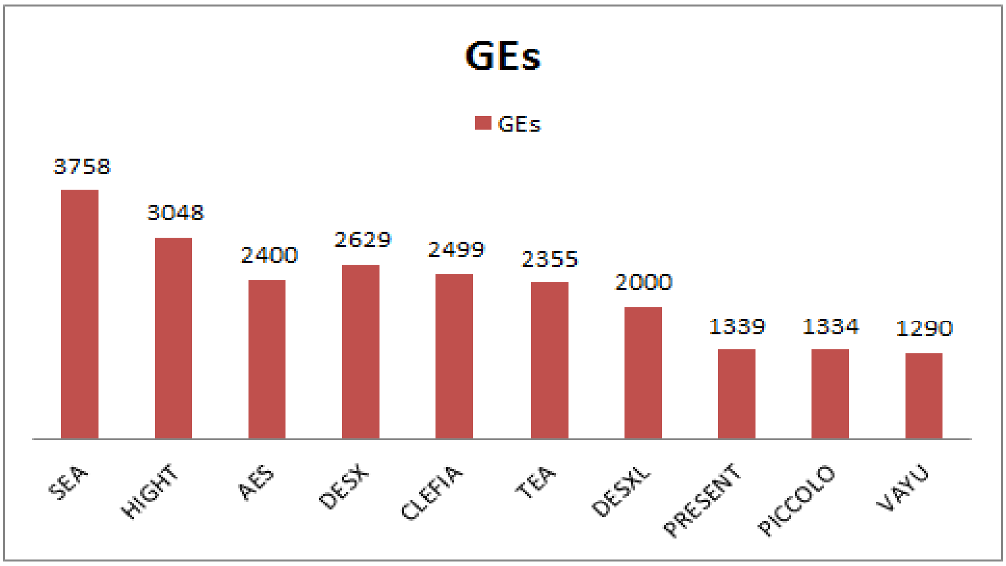
Figure 4. Data path for VAYU cipher for 128-bit key scheduling

Gate equivalent (GEs) is computed with above mentioned library and GEs calculation for the VAYU cipher is represented in Table 11[31].

Table 11. Calculation Of GEs For VAYU-128

Data Layer	GE’s	Key Layer	GE’s
D Reg.	544	D Reg.	544
2:1 MUX	9	2:1 MUX	4.5
XOR	32	XOR	10
SBOX	24	FSM	122
Shift Operator	0	Shift Operator	0
Total	609	Total	680.5
Total No. of gates required for 128 bit key = 1289.5			

Figure 5 shows GEs comparison of other existing ciphers with VAYU cipher.



GEs Comparison of Standard algorithms with VAYU cipher

Table 12 shows the gate equivalent comparison of VAYU cipher to existing ciphers. From Table 12, we can conclude that VAYU achieved best result in GEs reduction as compared to existing lightweight ciphers. The VAYU cipher has 1085.5 GEs for 80 bit key.

Table 12. GEs Comparison Of VAYU

	PRESENT	AES	CLEFIA	PICCOLO
VAYU	-3.29%	-46.25%	-48.37%	-3.29%

6. Conclusion

In this paper, we suggest a balanced Feistel based cipher “VAYU”, which has maximal data complexity i.e. 2^{60} and results in maximum number of active S-boxes in a fewer rounds. VAYU cipher needs only987 GEs for 128 bit key length which is very less as compared to all existing lightweight ciphers. VAYU cipher design is robust and is best suited for applications where small footprints are the major constraints. We believe VAYU cipher is the smallest cipher design till date, in terms of Gate Equivalents. With this cipher design we have achieved less GEs and competitive memory space.

VAYU cipher not only resists basic attacks but also it resists advance attacks like MITM, Zero correlation and Biclique. VAYU cipher design will have a positive impact in the field of lightweight cryptography, specifically this kind of designs will prove to be a crusader in making applications like IoT feasible.

Test Vectors (For 128 bit key)

Plain text	Key	Cipher text
0000000000000000	00000000000000000000000000000000	ad8d0baeabed93a3
123456789abcdef0	00000000000000000000000000000000	a9fd236a42111466

Copyright ⓒ 2016 SERSC

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 and Dr. Narayan Pisharoty for valuable suggestions time to time.

References

[1] 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] Zhao, J., Wang, X., Wang, M., Dong, X.: Differential Analysis on Block Cipher PRIDE. Cryptology ePrint Archive (2014), http://eprint.iacr.org/2014/525

[3] J. Borghoff, A. Canteaut, T. G¨uneysu, E. B. Kavun, M. Knezevic, L. R. Knudsen, G. Leander, V. Nikov, C. Paar, C. Rechberger, P. Rombouts, S. S. Thomsen, and T. Yal¸cin. PRINCE - A Low-Latency Block Cipher for Pervasive Computing Applications - Extended Abstract. In X. Wang and K. Sako, editors, ASIACRYPT, volume 7658 of LNCS, pp. 208–225. Springer, 2012.

[4] 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

[5] 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.

[6] “The 128 bit blockcipher“ CLEFIA: Algorithm specification." On-linedocument, 2007. Sony Corporation.

[7] Hong, D., Sung, J., Hong, S.H., Lim, J.-I., Lee, S.-J., Koo, B.-S., Lee, C.-H., Chang, D., Lee, J., Jeong, K., Kim, H., Kim, J.-S., Chee, S.: HIGHT: A New Block Cipher Suitable for Low-Resource Device. In: Goubin, L., Matsui, M. (eds.) CHES 2006. LNCS, vol. 4249, pp. 46–59. Springer, Heidelberg (2006)

[8] Wu, W., Zhang, L.: LBlock: A Lightweight Block Cipher. In: Lopez, J., Tsudik, G. (eds.) ACNS. Lecture Notes in Computer Science, vol. 6715, pp. 327–344 (2011)

[9] S. Hong, D. Hong, Y. Ko, D. Chang, W. Lee, S. Lee: Differential Cryptanalysis of TEA and XTEA. In: ICISC’03, LNCS, vol. 2971, pp. 402–417, Springer-Verlag, 2004.

[10] 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/.

[11] K. Nyberg. Generalized Feistel networks. In Kwangjo Kim and Tsutomu Matsumoto, editors, Advances in Cryptology - ASIACRYPT’96, LNCS 1163, pp. 91–104. Springer Verlag, 1996.

[12] 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

[13] 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.

[14] 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.

[15] Howard M. Heys, “A Tutorial on Linear and Differential Cryptanalysis” ,http://citeseer.nj.nec.com/443539.html

[16] E. Biham, A. Shamir, Differential Cryptanalysis of the Data Encryption Standard, Springer-Verlag, 1993

[17] E. Biham, A. Shamir, “Differential Cryptanalysis of DES-like Cryptosystems,” Journal of Cryptology, Vol. 4, No. 1, 1991, pp. 3–72

[18] Bogdanov, A., Rijmen, V.: “Zero Correlation Linear Cryptanalysis of Block Ciphers” IACR Eprint Archive Report 2011/123 (March 2011)

[19] HadiSoleimany and Kaisa Nyberg. “Zero-correlation linear cryptanalysis of reduced-round lblock” Cryptology ePrint Archive, Report 2012/570, 2012. http://eprint.iacr.org/

[20] 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.

[21] M. Albrecht, C. Cid. “Algebraic techniques in differential cryptanalysis” FSE 2009. LNCS, vol. 5665, pp. 193-208. Springer, Heidelberg. 2009

[22] E. Biham. “New Types of Cryptanalytic Attacks Using Related Keys”. Proceedings of Eurocrypt 93. LNCS. vol. 765, pp 398-409, Springer-Verlag. 1994

[23] A. Biryukov and D. Wagner. “Advanced Slide Attacks”. Proceedings of Eurocrypt 2000. LNCS. vol. 1807, pp. 589-606, Springer-Verlag. 2000

[24] 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

[25] R. Anderson, E. Biham and L. Knudsen, "Serpent: a proposal for the advanced encryption standard," NIST AES proposal 174, June 1998. available at ftp://dijkstra.urgu.org/crypto/Serpent/v1/res/serpent.pdf

[26] 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.

[27] 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

[28] M Kumar, SK Pal and APanigrahi. "FeW: A Lightweight Block Cipher". Scientific Analysis Group, DRDO, Delhi, INDIA, Department of Mathematics, University of Delhi, INDIA2014.

[29] G. Bansod, A. Patil, S. Sutar, N. Pisharoty " ANU: an ultra lightweight cipher design for security in IoT", SCN-15-0848.R1, Security and Communication Networks, DO - 10.1002/sec.169, http://dx.doi.org/10.1002/sec.1692

[30] G.Bansod, N. Pisharoty, A.Patil "PICO: An Ultra lightweight and Low power encryption design for pervasivecomputing", Defence Science Journal, 66 : 259-265.

[31] G.Bansod,A Patil.S Sutar, N Pisharoty "An ultralightweight encryption design for security in pervasive computing", 2nd IEEE International Confernce on BigDataSecurity on Cloud, April 2016. Columbia University, NewYork,USA ,Pages 79-84.

Copyright ⓒ 2016 SERSC