PICO: An Ultra lightweight and Low power encryption design for pervasive computing

Gaurav Bansod, Narayan Pisharoty, and Abhijit Patil

***Abstract*— In this paper we are proposing an ultra lightweight, a very compact block cipher 'PICO'. PICO is a substitution and permutation based network, which operates on a 64 bit plain text and supports a key length of 128 bits. It has a VERY compact structure that requires GEs for a 128 bit key length. The PICO cipher uses strong bit permutation layer which only needs wires for implementation this reduces overall gate count. Its unique design helps to generate a large number of active S - boxes in fewer rounds which thwart the linear and differential attacks on the cipher. PICO shows good performance on both the hardware and the software platforms. PICO consumes only 2504 bytes of Flash memory which is less than ultra lightweight cipher PRESENT. PICO has a very strong Substitution layer(S- box) which not only makes the design robust but also introduces a great avalanche effect. PICO has strong and compact key scheduling which is motivated from latest NSA designed cipher SPECK. PICO consumes 28mW of dynamic power which is less than the PRESENT cipher (31mW). In this paper we have presented the security analysis of PICO and its performance as an ultra lightweight compact cipher. PICO resists linear, differential, biclique, zero correlation, meet in the middle and key related attacks.**

***Index Terms*— Lightweight Cryptography, SP Network, Block cipher, IoT, Encryption, Embedded security**

I. Introduction

In recent years, many lightweight ciphers are introduced which has less footprint area, low power consumption and less gate counts. Lightweight Ciphers like PRESENT [1], PICCOLO [2], TWINE [3], SIMON and SPECK [4] have the robust design and needs less than 2200 Gate Equivalents (GEs) for implementation. Recently NSA launched the SIMON and SPECK cipher which are considered to be the most ultra lightweight ciphers. SIMON and SPECK has robust design and interesting key scheduling which makes the design robust and can thwart all possible types of attacks. PRESENT being the S-P network also has compact design and its hardware implementation needs around 1800-2000 GEs. Cipher like PRESENT is design by keeping goal as compact hardware implementation. But PRESENT has very weak substitution layer and its permutation layer is also vulnerable to attacks like statistical Saturational attack [5]. PRESENT has bit permutations as its P- layer which only requires wires for its hardware implementation [1]. In this paper we aimed at design compact and robust S P network cipher which not only needs less footprint area but also take care of the other factors like power consumption, GEs and all possible types of attacks. In this paper we have presented a compact cipher called PICO which is S-P network that needs less GEs, less footprint area and low power consumption as compared to the PRESENT cipher. We also aimed at providing stronger substitution layer that makes the design more robust. PICO shows good resistance against linear and differential attacks. Pico also shows good resistance against biclique attacks, zero correlation attack and key related attack. We believe that the PICO is the most ultra lightweight S-P network cipher till date which has compact structure, robust design and needs less GEs for its hardware implementation.

GAURAV BANSOD, NARAYAN PISHAROTY and ABHIJIT PATIL are with the Department of Electronics and Telecommunication, Symbiosis Institute of Technology, Symbiosis International University, Lavale, Pune, 412115, Maharashtra, INDIA

(E-mail: gauravb@sitpune.edu.in, narayanp@sitpune.edu.in, abhijit.patil@sitpune.edu.in).

We have used the following notification in this paper for PICO cipher

- P_j Input plaintext block of j = 64bits
- C_j Output cipher text block of j = 64 bits
- K^i 64-bit subkey for round i
- $\oplus$ Bitwise exclusive-OR operation
- $LCS(P_j,n)$ Left circular shift by n bits
- $RCS(P_j,n)$ Right circular by n bits
- $\|$ Concatenation of two strings
- ! Bitwise NOT operation
- & Bitwise AND operation
- | Bitwise OR operation

II. The PICO Block Cipher

The design of PICO cipher is based on a substitution permutation network [6]. It has total 32 rounds. PICO cipher supports 64 bit plaintext and 128 bit key length. Fig. 1 shows the block diagram of PICO cipher. Plaintext bits/ cipher text bits are arranged in 4×16 array format as shown in Fig. 2(a).

Let $P = p^{63}\|\cdots\|p^1\|p^0$ is the 64-bit plaintext, then row 0 contains the first 16 bits of plaintext $p^{15}\|\cdots\|p^1\|p^0$, row 1 contains the next 16 bits $p^{31}\|\cdots\|p^{17}\|p^{16}$, and so on, as shown in Fig. 2(a).

Fig. 2(b) represents the 'Two-dimensional Representation' of 4×16 array. Each round consist the following 3 steps:

AddRoundkey,
SubColumn &
Bit_Shuffle

After the last round, there is an AddRoundKey as shown in Fig 1.

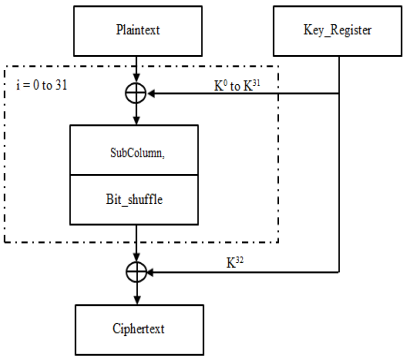

Fig. 1. Block Diagram of PICO cipher

S-box is a nonlinear element in cipher design which is followed by a bit permutation layer. 33 different subkeys each of 64 bits is generated from the 128 bit user defined key bits and out of these subkeys K^0 to K^{31} are applied at 0 to 31 rounds of the PICO cipher. K^{32} is the subkey which is used as post whitening key.

Pseudo code for PICO cipher is given as
P = P^{63} ...P^0
RoundKeys()
for i = 0 to 31 do
Add_round_key (P, K^i)
SubColumn (A)
Bit_Shuffle (A)
End for
Add_round_key (A , K_{32})
C → A

$$\begin{bmatrix} p^{15} & p^{14}\dots & \dots p^{1} & p^{0} \\ p^{31} & p^{30}\dots & \dots P^{17} & p^{16} \\ p^{47} & p^{46}\dots & \dots p^{33} & p^{32} \\ p^{63} & p^{62}\dots & \dots p^{49} & p^{48} \end{bmatrix} \qquad \begin{bmatrix} p^{0,15} & p^{0,14}\dots & \dots p^{0,1} & p^{0,0} \\ p^{1,15} & p^{1,14}\dots & \dots p^{1,1} & p^{1,0} \\ p^{2,15} & p^{2,14}\dots & \dots p^{2,1} & p^{2,0} \\ p^{3,15} & p^{3,14}\dots & \dots p^{3,1} & p^{3,0} \end{bmatrix}$$

(a) (b)

Fig. 2 (a) represents 4×16 array format and its Two-dimensional representation is in (b)

Each round of PICO cipher consists of the following operations:

A. *Add_round_key*

Add_round_key performs an XOR operation with 64 bit plaintext and 64 bit sub key. Sub keys are denoted by K^i where i range from 0 to 32 and the current state output P → p^{63}…..p^0 is given as,

$$P \rightarrow P \oplus K^i.$$

B. *SubColumn*

The S-box used in PICO cipher is of 4×4 S-box where S-box: $F_2^4 \rightarrow F_2^4$. Table 1 represents the hexadecimal values for the Substitution layer,

TABLE 1
S-BOX OF PICO CIPHER

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

SubColumn operation is based on the simultaneous application of S-boxes to the 4 bits in the same column. Operation of SubColumn is illustrated in Fig. 3. The input of a S-box is Column(i) = $p^{3,i} \| p^{2,i} \| p^{1,i} \| p^{0,i}$ where i ranges from $0 \leq i \leq 15$ and $p^{0,i}$ is LSB bit and $p^{3,i}$ is the MSB bit of the 4 bit nibble. The output is S-box(Column(i)) = $q^{3,i} \| q^{2,i} \| q^{1,i} \| q^{0,i}$. S- box is applied column wise in our cipher design.

$$\begin{matrix} p^{0,15} & p^{0,14}\dots & \dots p^{0,1} & p^{0,0} \\ p^{1,15} & p^{1,14}\dots & \dots p^{1,1} & p^{1,0} \\ p^{2,15} & p^{2,14}\dots & \dots p^{2,1} & p^{2,0} \\ p^{3,15} & p^{3,14}\dots & \dots p^{3,1} & p^{3,0} \end{matrix}$$

S-box

$$\begin{matrix} q^{0,15} & q^{0,14}\dots & \dots q^{0,1} & q^{0,0} \\ q^{1,15} & q^{1,14}\dots & \dots q^{1,1} & q^{1,0} \\ q^{2,15} & q^{2,14}\dots & \dots q^{2,1} & q^{2,0} \\ q^{3,15} & q^{3,14}\dots & \dots q^{3,1} & q^{3,0} \end{matrix}$$

Fig. 3. SubColumn Operation

Let X = ABCD be the input to the S-box and Y = y3y2y1y0 is the output. For example X= ABCD =0000 then Y = 1110. The 4 × 4 S-box of PICO is described by the following equations, these equations are formed by using the K-map.

y0 = (!B&C&D) | (B&C&!D) | (A&B&!D) | (A&!B&D) | (!A&!B&!C&!D) | (!A&B&!C&D);

y1 = (!A&!C&D) | (A&!B&!D) | (A&!B&C) | (!A&B&!D) | (A&B&C&!D);

y2 = (!A&B&!C) | (!A&!B&C) | (A&!C&D) | (A&C&!D);

y3 = (!A&C&D) | (!A&B&C) | (A&!B&!D) | (B&!C&D) | (A&B&!C&!D);

C. *Permutaion_Layer*

Permutation layer of PICO cipher is based on bit permutation and it is a linear permutation operation. The bit permutation used in PICO is given in the Table 2. The bit $p^{i,j}$ present in i^{th} row and j^{th} column is shifted to the new bit position as described in the Table 2. Bit_Shuffle($p^{i,j}$) i.e. Bit_Shuffle($p^{0,0}$) → $p^{0,10}$. It means after permutation operation , the bit will be shifted to 0^{th} row and 10^{th} column.

TABLE 2
P-BOX OF PICO CIPHER

j → / i ↓	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0,10	1,5	1,12	2,6	2,12	3,0	3,11	0,1	3,3	0,15	2,9	0,2	3,12	2,2	1,8	1,4
1	3,8	0,6	1,1	1,15	2,4	3,5	0,12	2,14	1,14	3,4	0,11	0,4	1,7	2,3	2,8	3,15
2	0,8	2,7	0,3	2,11	3,9	3,1	1,0	1,9	2,5	2,10	3,13	3,2	0,0	0,9	1,2	1,10
3	3,10	3,7	0,7	1,3	1,13	0,14	2,15	2,0	2,1	0,5	3,14	2,13	0,13	3,6	1,6	1,11

D. Key Schedule of 128-bit key length

Key schedule of PICO cipher is motivated from the SPECK cipher key scheduling design [4]. SPECK key scheduling is compact in memory size requirement and no attacks till date are reported on it.

In PICO cipher total 33 subkeys are used each of size 64 bits are extracted from 128 bit key scheduling algorithm which is mentioned in the next subsection.

1) 128-bit key scheduling

User defined 128 bit key is stored in the register Key, subkey K^0 and L^1 can be given as

$$\text{Key} = k^{127}\, k^{126}\, k^{125} \ldots k^2\, k^1\, k^0$$
$$K^0 = k^{63}\, k^{62} \ldots k^1\, k^0$$
$$L^1 = k^{127}\, k^{126} \ldots k^{66}\, k^{65}\, k^{64}$$

After extracting key as K^0 and L^1 each of 64 bits, the subkeys K^1 to K^{32} are generated as follows

For j = 0 to 31 do
$L^2{}_{64} = ((K^j{}_{64}) \oplus \text{RCS}(L^1{}_{64},3)) \oplus (L^1{}_{64})$;
$K^{j+1}{}_{64} = ((L^2{}_{64}) \oplus \text{LCS}(K^j{}_{64},7)) \oplus j$;
$L^1{}_{64} = L^2{}_{64}$;
End for

Where LCS represents left circular shift by 7 bits and RCS represents right Circular shift by 3 bits.

Subkeys are arranged in 4×16 array format as shown in Fig. 4 to perform AddRoundKey operation.

$$\begin{bmatrix} k^{0,15} & k^{0,14} \ldots & \ldots k^{0,1} & k^{0,0} \\ k^{1,15} & k^{1,14} \ldots & \ldots k^{1,1} & k^{1,0} \\ k^{2,15} & k^{2,14} \ldots & \ldots k^{2,1} & k^{2,0} \\ k^{3,15} & k^{3,14} \ldots & \ldots k^{3,1} & k^{3,0} \end{bmatrix}$$

Fig. 4. 4×16 two dimensional representations of Subkey bits

III. SECURITY ANALYSIS OF PICO

Various cryptanalysis techniques are applied on a cipher to find robustness of cipher design against all possible types of attacks. In paper we have focused on attacks like the differential attack, the linear attack, the algebraic attack, the key scheduled attack, the key collision attack, the zero-correlation attack and the biclique attack. S-box is a nonlinear layer in cipher design and it plays a very important role to provide security against well-known attacks like linear attack and differential attack.

Design of a cipher is an important factor to increase large number of active S-boxes. Structure that provides large number of active S-boxes considered having robust architecture which can thwart all possible types of attacks. Computer based techniques are used in this paper for selection of good S-box and also to calculate the minimum number of active S-boxes.

A. Design Criteria of the S-box

Gate count is increased by using separate S-box in each round of cipher and similarly this does not provide sensible amount of improvement in the resistance against known attacks [7]. We have chosen 4×4 S-box in our design of PICO cipher. S- box used in the PICO cipher is robust and prevents clustering of linear and differential trails. One of the most important aspect in our cipher design is the nonlinear robust layer i.e. S-box.

4×4 S-box provides compactness and selection of proper S-box provide resistance against linear and differential attack. These two parameters we have considered while designing the S-box.

PICO S-box is S: $F_2^4 \leftarrow F_2^{4,}$ it means that it takes a 4 bit input and produces a 4 bit output. Important properties for a good S-box design are mentioned below which we have considered for S- box selection.

Property 1: Linear Property
$a \in F_2^4$ is input to the S-box. A is the input mask and B is the output mask such that $A \in F_2^4$, $B \in F_2^4$ so LC (A, B) is defined as:

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

This property is used to form the Linear Approximation Table (LAT), where LC represents Linear Cryptanalysis [8].The • denotes mask operation on F_2^4 and # indicate the number of matches in LAT for the input mask A and B minus 8.

Property 2: Differential Property
$a \in F_2^4$ is input to the S-box and ΔA, ΔB are the input and output differences such that ΔA, $\Delta B \in F_2^4$ so that DC (ΔA, ΔB) is defined as:

$$\text{DC}(\Delta A, \Delta B) = \#\{a \in F_2^4 | S(a) \oplus S(a \oplus \Delta A) = \Delta B\}$$

This property used to form Difference Distribution Table (DDT) and DC represents differential Cryptanalysis [9].

Complete design criteria of the S-box which we have used in designing of the PICO cipher is given below,

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

$$DC(\Delta A, \Delta B) = \#\{a \in F_2^4 \mid S(a) \oplus S(a \oplus \Delta A) = \Delta B\} \leq 4$$

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

$$SetDC = DC(\Delta A, \Delta B) = \#\{a \in F_2^4 \mid S(a) \oplus S(a \oplus \Delta A) = \Delta B\} = 0$$

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

This is the most important property in designing S- box. We have achieved Cardinality of 2 in both linear and differential table for the given S- box. This property indicates the strength and robustness of S-box.

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

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

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

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

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

5. Bijective i.e. $S(a) \neq S(b)$ for all values of $a \neq b$.

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

Strength of the S-box depends on cardinality, For PICO cipher, S-box has $Car_{DC} = 2$ and $Car_{LC} = 2$.

In the case of PRESENT cipher, S-box has CarDC = 0 and CarLC = 8 [10], while in case of RECTANGLE cipher, S-box has CarDC = 2 and CarLC = 2 [10]. This shows that PRESENT has a weak S-box, while RECTANGLE S-box shows good strength. As our designed S-box shows the good cardinality values, it shows that our S-box is good in design as compare to the PRESENT S-box.

B. PICO cipher S-Box Selection

Selection of S-box in PICO cipher is driven by two definitions

1) Permutation-then-XOR equivalence [10] [11]

If there exist a 4 X 4 permutation matrices m0 , m1 and constant A, B which belongs to F_2^4 for two S-boxes S'(x) and S(m0(x) + A), such that S′(x)= m1(S (m0(x) + A)) + B, then the equivalence is called PE equivalence.

When S-box satisfies the criteria 1 to 5 from section A then its PE equivalent S-box also satisfies criteria 1 to 5.

2) Affine equivalence [10] [11]

If there exist a bijective linear mapping A, B and constant a, b belongs to F_2^4 for two S-boxes such that S′ (x) = B(S (A (x) + a)) + b, then the equivalence called affine equivalence.

When S-boxes satisfies criteria 1,3 and 5 from section A then the affine equivalent S-box also satisfies criteria 1, 3 and 5.

These two definitions are considered while designing the S-box for the PICO cipher.

C. Linear cryptanalysis

In cryptanalysis technique 'Liner Cryptanalysis' [8] is the most significant attack which is applicable on the symmetric-key block ciphers. This attack is also referred as the known plaintext attack. High probability occurrences of linear expression containing plaintext bits, cipher text bits and subkey bits are used for mounting the linear attack on cipher. Linear attack is mounted by having the knowledge about a subset of plaintext and its corresponding ciphertext. Attacker will find the correlation between them. S-box is examined by forming Linear Approximation Table (LAT) as mentioned in Table 3. Bias (ε) can be given as $|P_L - 1/2|$ where P_L represents the linear probability. Maximum bias value for PICO cipher is 2^{-2}. Matsui`s Piling-up lemma [8] is used to calculate the probability bias for 'n' rounds. The best way to resist against linear attack is mentioned below,

1] Optimizing the bias in LAT. For ideal S-box bias, values should be 1/8 which is practically not possible to achieve.

2] Increase the number of active S-boxes in cipher structure.

Lemma 1: Matsui`s Pilling up Lemma [8]

For 'n' independent random binary variables X_1 , $X_2 \ldots X_n$, the equation is,

$$\varepsilon = 2^{n-1} \prod_{i=1}^{n} \varepsilon_i$$

Where ε represents the total bias of $X_1 \oplus \ldots \oplus Xn = 0$ and n represents number of active S-box.

TABLE 3
LINEAR APPROXIMATION TABLE FOR PICO CIPHER

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

Table 4 represents the minimum number of active S-boxes from linear trails.

TABLE 4
MINIMUM NUMBERS OF ACTIVE S-BOXES FROM LINEAR TRAILS

#Round	# Min. active S-boxes
1	1
2	2
3	4
4	6
5	8
6	11

Theorem1:

For 24 rounds of PICO it has total 44 active S-boxes and total bias for 24 rounds is 2^{-45}.

Proof:

For 6 rounds PICO has minimum 11 active S-boxes. Maximum bias for the PICO cipher S-box is 2^{-2} by using Matsui`s Pilling up Lemma for 6 rounds of PICO cipher the total Bias can be given as

$$2^{10} \times (2^{-2})^{11} = 2^{-12}$$

For 24 rounds the total bias (ε) can be given as

$$\varepsilon = 2^{3} \times (2^{-12})^{4} = 2^{-45}$$

By calculating required number of known plaintext / ciphertext, we can compute complexity of linear attack and can be given as

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

For 18 rounds of PICO cipher the required number of known plaintext / ciphertext can be given as

$$N_L = 1/ (\varepsilon)^2 = 1/ (2^{-45})^2$$
$$N_L = 2^{90}$$

Available limit is 2^{64} and required no. of known plaintext is 2^{90}, hence complete rounds of PICO cipher shows good resistance against a linear attack.

Table 5 shows the linear trails for PICO cipher.

TABLE 5
LINEAR TRAILS FOR PICO CIPHER

#Round	Input to S-box	Output from S-box
1	0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000	0000 0000 0000 0000 0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
2	0000 0000 0000 0000 0000 0000 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000	0000 0000 1000 0000 0000 0000 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000
3	0000 0000 0000 0010 0000 0000 0000 0000 0100 0000 0000 0000 0000 0000 0000 0000	0100 0000 0000 0000 0100 0000 0000 0010 0000 0000 0000 0000 0000 0000 0000 0000
4	0000 0000 0100 0000 0000 0001 0000 0000 0000 0001 0000 0000 0000 0000 0000 0000	0000 0000 0000 0000 0000 0000 0100 0000 0000 0001 0000 0000 0000 0000 0000 0000
5	0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0010 0000 0000 0000 0000 0000	0000 0000 0010 0000 0001 0000 0010 0000 0000 0000 0000 0000 0000 0000 0000 0000
6	0000 0000 0000 0000 0000 0000 1000 0000 0000 0000 0000 0000 0000 0000 0010 0001	0000 0000 1010 0001 0000 0000 1000 0000 0000 0000 0000 0000 0000 0000 0010 0001

D. Differential cryptanalysis

In cryptanalysis technique "Differential Cryptanalysis" [9] [12] is one of the most significant attack applicable to symmetric key block cipher. Differential Cryptanalysis firstly applied on DES by Biham and Shamir in 1990. Pair of high probability input and output occurrences are used to mount this attack. Substitution layer is a nonlinear layer in our design, which is examined by forming difference distribution table (DDT) as mentioned in Table 6.

Differential trails are formed by considering high probability input and output difference for each round, S-box that has non-zero input difference or non-zero output difference is referred as an active S-box.

P_D represents differential probability, P_D value for PICO S-box is $4/16 = 1/4 = 2^{-2}$.

TABLE 6
DIFFERENCE DISTRIBUTION TABLE FOR PICO S-BOX

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

Security against this attack can be provided by using following two approaches,

1. Minimize the differential probability (P_D), For ideal S-box P_D=1/16 .
2. Build a cipher design such that it maximizes the minimum number of active S-boxes.

Table 7 represents the differential trails for PICO cipher. Nonzero input difference to S-box or non-zero output difference from S-box is referred as an active S-box.

TABLE 7
DIFFERENTIAL TRAIL FOR PICO CIPHER

#Round	**Input to S-box**	**Output from S-box**
1	0000 0000 0000 0000 0000 0000 0100 0000 0000 0000 0000 0000 0000 0000 0000 0000	0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0100 0000 0000 0000 0000 0000
2	0000 0000 0000 0000 0000 0000 0000 0001 0000 0000 0000 0000 0000 0000 0000 0000	0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0001 0000 0000 0000 0000
3	0000 0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000	0000 0001 0000 0000 0000 0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
4	0000 0000 0000 0000 0100 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 1000	0000 0000 0000 1000 0000 0000 0000 1000 0100 0000 0000 0000 0000 0000 0000 0000
5	0000 0000 0000 0000 1000 0000 0000 0100 0000 0000 0100 0000 0000 0000 0000 0000	0000 0000 0100 0000 0000 0000 0100 0000 1000 0000 0000 0100 0000 0000 0000 0000
6	0001 0000 0000 1000 0000 0100 0000 0000 0000 0000 0000 0000 0000 1000 0000 0000	0001 1000 0000 1000 0001 1000 0000 1000 0000 0100 0000 0000 0000 0000 0000 0000

Table 8 represents the minimum number of active S-boxes from differential trails.

TABLE 8
MINIMUM NUMBERS OF ACTIVE S-BOXES FROM DIFFERENTIAL TRAIL

#Round	# Min. active S-boxes
1	1
2	2
3	3
4	5
5	8
6	12

There are 12 active S-boxes for 6 rounds of PICO cipher so for 24 rounds there are 48 active S-boxes. Differential probability for complete cipher is given as $(2^{-2})^{48} = 2^{-96}$.

Total number of chosen plaintext required to mount this attack can be given as,

$$N_d = C/P_d$$

Where C = 1 and $P_d = 2^{-96}$, so the required number of chosen plaintext / ciphertext are,

$$N_d = 1/2^{-96} = 2^{96}.$$

E. Zero-Correlation attack

Zero-correlation attack [13] [14] is the extension of Linear Cryptanalysis. The block ciphers should resist zero correlation attack. Zero-correlation Attack is based on linear approximations with a correlation value of zero. Zero-correlation attack is considered as a counter part of impossible differential cryptanalysis in domain of linear cryptanalysis. We have applied matrix method [13] to mount Zero-correlation attack which is explained below. Following three Lemmas are used to find contradictions.

Lemma 2: XOR approximation

Either the three linear selection patterns at an XOR $\oplus$ are equal or the correlation over $\oplus$ is exactly zero.

Lemma 3: Branching approximation

Either the three linear selection patterns at a branching point • sum up to 0 or the correlation over • is exactly zero.

Lemma 4: Permutation approximation

Over a permutation φ, if the input and output selection patterns are neither both zero nor both nonzero, the correlation over φ is exactly zero.

1) The Matrix Method [13]

Impossible differential characteristic can be constructed by using Miss-in-the-middle approach. Contradiction in middle is formed by two differential paths with probability.

Matrix method is used to find linear approximation with correlation zero and mentioned below,

The linear masks applied to the words can be of the following five types:

1. Zero mask denoted by 0,
2. An arbitrary non-zero mask denoted by $\bar{0}$,
3. Non-zero mask with a fixed value a,
4. The exclusive-or of a fixed non-zero mask a and an arbitrary non-zero mask, denoted by $\bar{a}$,
5. Any other mask is denoted by $*$.

TABLE 9
ARITHMETIC RULES MULTIPLICATION BY 0, 1 AND 1F.

	0	**1**	**1F**
0	0	0	0
$\bar{\mathbf{0}}$	0	$\bar{0}$	$\bar{0}$
a	0	a	$\bar{0}$
$\bar{\mathbf{a}}$	0	$\bar{a}$	*
*****	0	*	*

The matrix shows that how a linear mask of each output word is affected by the linear mask of an input word. Table 9 and 10 illustrate Arithmetic rules for multiplication and addition.

TABLE 10
ARITHMETIC RULES ADDITION BETWEEN TWO MASK

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

2) Zero-Correlation for 4 rounds of PICO

For (0a00000000000000) → (000000000b000000) has correlation exactly zero for which the values a and b are non-zero. Trails for zero correlation attack are shown in Table 11 and we found contradiction at round 2 for PICO cipher. Red bits in table 11 shows contradiction.

TABLE 11
TRAILS FOR ZERO-CORRELATION FOR PICO CIPHER

#Round	**Trails**
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 a a a 0 0 0 0 0 0 0 0
1	0 0 0 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 $\bar{0}$ 0 0 0 0 0 0 0 0 0 0 0 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
2	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 $\bar{0}$ 0 0 0 0 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 0 0 0 0 0 0 0 0 0 0 0 $\bar{0}$
2	0 0 0 0 0 0 0 0 0 0 0 0 0 0 $\bar{0}$ 0 0 0 0 0 0 0 0 0 0 0 $\bar{0}$ 0 0 0 0 0 0 0 0 0 0 $\bar{0}$ 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
3	0 0 0 0 $\bar{0}$ 0 0 0 0 0 0 0 0 $\bar{0}$ 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 $\bar{0}$ 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 0 0 0 0 0 0 0 0 0 b b b b 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

F. Biclique attack

Biclique attack [15] [16] is an extension of meet-in-the-middle attack.

We have applied biclique cryptanalysis technique on PICO-128. From this result we have compare the result of PICO with other standard cipher algorithms like PRESENT, PICCOLO and LED.

We have constructed a 4-dimensional biclique for round 29 ~ 32 of PICO-128. For these rounds the partial keys used is (K^{29}, K^{30}, K^{31}, K^{32}) which is described as follows,

$K^{29} = k_{52}, k_{51}, \ldots k_0, k_{63} \ldots . k_{53}$
$K^{30} = k_{45}, k_{44}, \ldots k_0, k_{63} \ldots . k_{46}$
$K^{31} = k_{38}, k_{37}, \ldots k_0, k_{63} \ldots . k_{39}$
$K^{32} = k_{31}, k_{30}, \ldots k_0, k_{63} \ldots . k_{32}$

From above equation we found that by varying following sub keys (k_{30}, k_{14}, k_{62}, k_{46}) and (k_{15}, k_{33}, k_{44}, k_{27}) gives bicliques for the attack on the full PICO-128.

To construct the Δi-differential we consider sub keys (k_{30}, k_{14}, k_{62}, k_{46}) and for the ∇j-differential, we consider sub keys (k_{15}, k_{33}, k_{44}, k_{27}) Let f be a sub-cipher from round 29 to round 32. Since the Δi-differential affects the 48 bits of the ciphertext as illustrated from Fig. 5. As a result, the data complexity does not exceed 2^{48}. Red lines from the round in Fig 5 shows the data complexity.

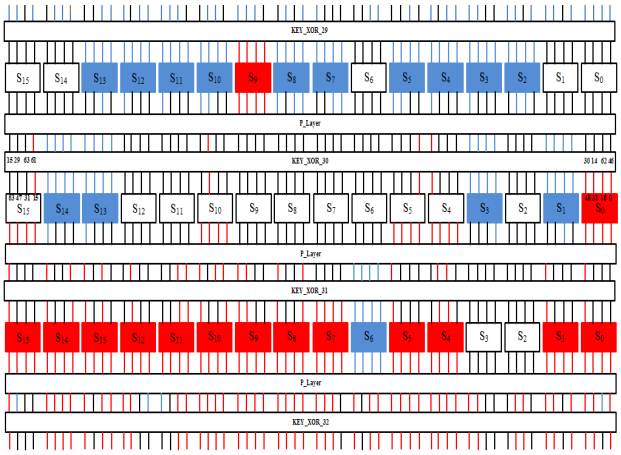

Fig. 5. Four Dimensional Biclique for PICO-128

Fig. 6 and 7 represents the re-computation in forward and backward direction. The total computational complexity of PICO-128 is computed as follows.

$$C_{total} = 2^{k-2d}\,(C_{biclique} + C_{precomp} + C_{recomp} + C_{falsepos}).$$
$$C_{total} = 2^{127.717}.$$

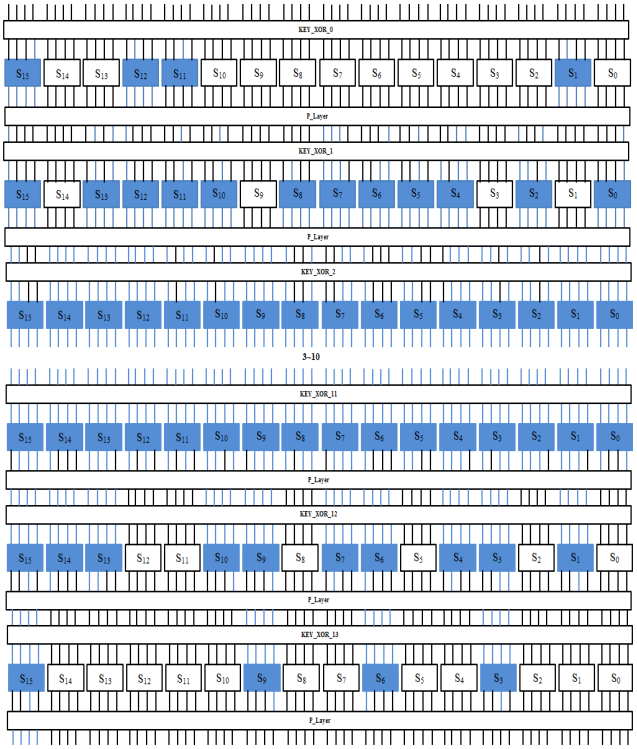

Fig. 6. Re-computations in Forward directions for PICO-128

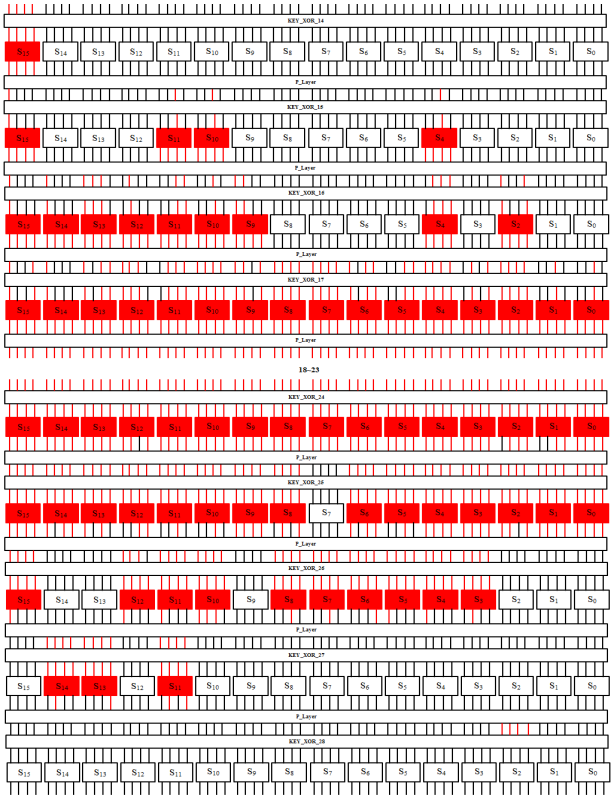

Fig. 7. Re-computations in backward directions for PICO-128

G. *Algebraic attack*

Attacker applies algebraic attack [17] more usually on stream cipher because it is more successful on it rather than applying on block cipher. The 4 × 4 bit S-box is be described

by minimum 21 equations in 8 input or output variables. x = a × 21 quadratic equation in y = a × 8 variables are used to examine complete cipher. Where 'a' represents number of S-boxes used in encryption algorithm and key scheduling algorithm. In our cipher design for single round of encryption there are total 16 S-boxes that are used and for 128-bit key scheduling there are 48 S-boxes used. For 32 rounds of cipher, there are a total of 32 × 16 = 512 S-boxes in the encryption system.

The number of quadratic equations that can be formed given as,

$$a = (512) \times 21 = 10752$$

and the number of variables can be given as

$$b = (512) \times 8 = 4096.$$

The 10752 number of quadratic equations in 4096 variables provide good resistance against the algebraic attack.

H. Key schedule attacks

There is no specific instructions are provided to design key scheduling algorithm. So large number of multiple key scheduling algorithm can be formed and vide variety of key related attack can be mounted. Related key attack [18] and slide attack [19] are the two important attack mounted on key scheduling to check wheatear key scheduling is vulnerable or not. Related key attack is also known as chosen key attack and is successfully applied on reduced round AES-256 [20].

In this paper, we aimed at providing efficient and compact key scheduling which resist against the possibility of related key attack.

Recently, NSA has designed the lightweight ciphers named SIMON and SPECK which is optimized for hardware and software implementations [4]. There is no kind of key related attack found on SPECK key scheduling. In this paper, our key scheduling design is motivated from SPECK key scheduling.

I. Key Collision Attack

This attack depends on the key length regardless of the key scheduling algorithm. Key collision attack [21] forms message with complexity $2^{k/2}$, where k denotes the length of key size. In our cipher design k =128, so the complexity of created message is given as $2^{128/2} = 2^{64}$.

J. Avalanche Effect [22]

When a single bit change in the input changes the output significantly, this results in an avalanche effect. For example by flipping a single bit in the input or in a key could change the half of the bits in cipher text. Cipher with good avalanche effect has higher probability to resist all possible types of attacks.

In case of robust design of block ciphers, drastic change in the **cipher text** is visible when a small change in the **key** or the **plaintext** takes place. The poor randomization occurs when a block cipher does not show the avalanche effect to a significant degree.

In our design we have applied a single bit change in input plaintext / Key bits and observed the output. We found that in the case of the PICO cipher that any single bit change in key results in changes in more than half of the bits of cipher text. Table 12 summarizes the Avalanche Effect.

TABLE 12
AVALANCHE EFFECT FOR PICO-128

Plaintext	0000 0000 0000 0000	No. of bits Change
Key	0000 0000 0000 0000 0000 0000 0000 0000	--
Ciphertext	fda7e7de58c913f4	
Key	0800 0000 0000 0000 0000 0000 0000 0000	40
Ciphertext	72f4081fae46ef5d	
Key	0400 0000 0000 0000 0000 0000 0000 0000	36
Ciphertext	f95da221c75cbeb7	

IV. SECURITY COMPARISON WITH STANDARD ALGORITHM

In this section we have compared the security analysis of PICO with the other standard algorithms. The comparison is represented in Table 13 and 14. Table 13 compares the linear complexity and differential complexity by considering the number of active S-boxes for particular rounds.

TABLE 13
LINEAR AND DIFFERENTIAL ATTACK COMPARISON

Cipher Name	**PICO**	**PRESENT**	**L-Block**	**FEW**	**PICCOLO**
#Rounds	24	25	15	27	30
# Active S-box From Linear Trails	45	50	32	45	30
# Active S-box From Differential Trails	48	50	32	45	30
#Known Plaintext	2^{90}	2^{102}	2^{66}	2^{90}	2^{120}
#Chosen Plaintext	2^{96}	2^{100}	2^{64}	2^{90}	2^{120}
Reference	This Paper	[1]	[26]	[27]	[3]

Table 14 compares the data complexity and computational complexity of PICO with the other ciphers.

TABLE 14
BICLIQUE ATTACK COMPARISON

Cipher Name	Rounds	Data Complexity	Computational Complexity	Reference
PICO-128	Full(32)	2^{48}	$2^{127.717}$	This Paper
PRESENT-80	Full(31)	2^{23}	$2^{79.54}$	[16]
PRESENT-128	Full(31)	2^{19}	$2^{127.42}$	[16]
PICCOLO-80	Full(25)	2^{48}	$2^{79.13}$	[16]
`PICCOLO-128	Full(31)	2^{24}	$2^{127.35}$	[16]
LED-64	Full(48)	2^{64}	$2^{63.58}$	[16]
LED-80	Full(48)	2^{64}	$2^{79.37}$	[16]
LED-96	Full(48)	2^{64}	$2^{95.37}$	[16]
LED-128	Full(48)	2^{64}	$2^{127.37}$	[15]

Table 15 compares the S-box design consideration comparison with the lightweight ciphers. PICO S-box have CAR_{DC} = 2 and CAR_{LC} = 2 which illustrate that PICO cipher S-box is robust in design and provides good security than other lightweight ciphers.

TABLE 15
S-BOX DESIGN CONSIDERATION

Cipher Name	Max. Val. in DDT	Max. Val. in LAT	CAR_{DC}	CAR_{LC}
PICO	4	4	2	2
PRESENT	4	4	0	8
RECTANGLE	4	4	2	2
TWINE	4	4	5	7

V. HARDWARE AND SOFTWARE PERFORMANCE OF PICO CIPHER

Design of a PICO cipher provides optimum performance in software platform. We have considered the 32 bit ARM 7 LPC2129 [23] processor for analyzing software performance of PICO cipher.

Footprint area i.e. GEs are computed with standard cell library based on UMCL 180 0.18µ logic process (UMCL18G212T3) [24]. Memory size required for PICO cipher on 32 bit processor is 2504 bytes as Flash memory and 1256 as RAM memory. All other ciphers are written in embedded C and implemented on a 32 bit processor for comparison with the PICO cipher. Fig. 8 represents the memory comparison of the existing lightweight ciphers with PICO cipher. PICO needs less memory as compared to the other S-P network light weight ciphers.

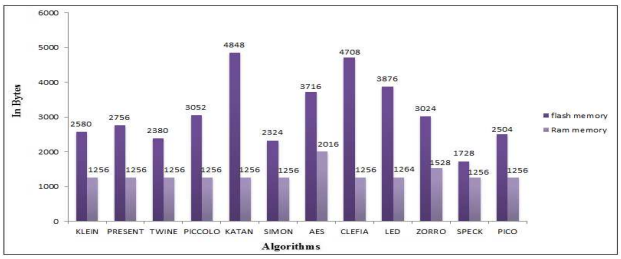

Fig. 8. Flash memory and RAM memory Comparison of Standard algorithms with PICO Cipher implemented on LPC2129

Data path for PICO cipher is shown in Fig. 9.

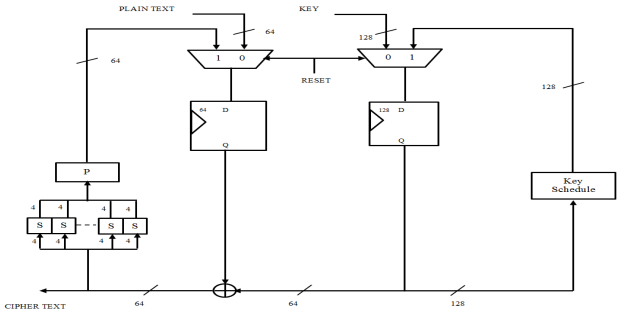

Fig. 9. Data path for PICO cipher for 64-bit plaintext and 128-bit key

Gate equivalent (GEs) is computed with UMCL180 standard cell library represented in Table 16.

TABLE 16
GATE COUNT OF UMCL18G212T3 LIBRARY

Standard Cell	Process	GE
NOT	0.18µm	*0.67*
AND	0.18µm	*1.33*
XOR	0.18 µm	*2.67*
D F.F.	0.18 µm	*6*

GEs calculation for PICO cipher is represented in Table 17. GEs are counted as per the encryption algorithm mentioned in in Section I. For 128 bit key scheduling complete cipher need 1877 GEs .

TABLE 17
CALCULATION OF GES FOR PICO CIPHER

DATA Layer	GEs	%	KEY Layer	GEs	%
D Reg.	*384*	*20.44*	K Reg.	*768*	*40.89*
S-Box	*384*	*2044*	Shift Operator	*0*	*0*
P-Layer	*0*	*0*	S-box	*0*	*0*
XOR	*170.88*	*9.09*	XOR	*170.88*	*9.09*
Total	*938.88*	*43.89*	Total	*938.88*	*56.11*
Total number of gates required for 128-bit key = *1877.76 =1878*					

Fig. 10. Shows the GEs [23] comparison of other existing ciphers with PICO cipher. PICO results in consuming less GEs as comapred to all the other existing lightweight ciphers.

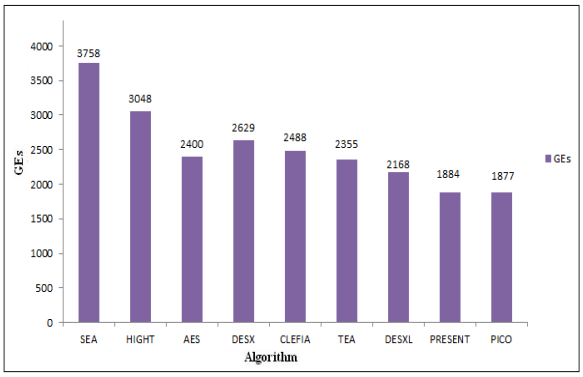

Fig. 10. GEs Comparison of Standard algorithms with PICO cipher

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. Table 18 represents dynamic power consumption of standard ciphers; PICO consumes 28mw power which is lesser than other lightweight ciphers.

TABLE 18
POWER CONSUMPTION OF LIGHTWEIGHT CIPHER

Standard Cell	Dynamic Power Consumption in mW
PICO	*28*
PRESENT	*31*
LED	*61*

Table 19 shows the comparison of lightweight ciphers with PICO based on parameters like execution time, throughput and number of cycles required to convert plain text to cipher text. Throughput is computed on software platform at 12MHz.

TABLE 19
COMPARISON WITH RESPECT TO THROUGHPUT, EXECUTION TIME AND NUMBER OF CYCLES

Ciphers	Block Size	Key Size	Execution Time (In uSec)	Throughput (In Kbps)	No. of Cycles
SP NETWORK					
ZORRO	128	128	913.21	140	10958.52
KLEIN	64	96	887.51	72	10650.12
PICO	**64**	**128**	**4134.23**	**15.48**	**49610.76**
PRESENT	64	128	2648.65	24.16	31783.8
FEISTEL STRUCUTRE					
SPECK	64	128	49.02	1305	588.24
SIMON	64	128	105.67	605	1268.04
PICCOLO	64	128	227.68	281	2732.16
CLEFIA	128	128	1048.01	122	12576.12
TWINE	64	128	592.87	108	7114.44

VI. CONCLUSION

In this paper we have presented a PICO, an ultra-lightweight and low power cipher. PICO has a compact design which results in lesser foot print area and lower power consumption. PICO performs efficiently both on the hardware and the software platform. PICO achieves a great speed while encrypting the text, as it is based on S-P network as compared to the Feistel based ciphers. We have shown the resistance of PICO cipher against all types of possible attacks. During the cipher design we have done extensive computer based searches for good S-boxes, minimum number of active S-boxes and calculation of hamming weight for specific entries in the LAT and in the DDT. PICO cipher has a very strong S-box and a robust permutation layer which prevents the cipher design from undergoing the clustering of linear and differential trails. In designing PICO, we have achieved a very small gate count so that it can be implemented for security in any small scale embedded system. Through PICO cipher, we have tried to design the smallest block cipher of S-P network type and it is definitely better than any other lightweight S-P network cipher designed so far. PICO needs less GEs, less footprint area and less power consumption as compared to PRESENT cipher. Moreover PICO shows good resistance against attacks as compared to the PRESENT cipher. PICO has a very strong S- box and P-layer which makes the cipher design more robust as compared to PRESENT cipher. For applications like RFID tags, Wireless sensor nodes, where small GEs and power consumption play a crucial role, we believe PICO is the best suited design.

TEST VECTORS(FOR 128 BIT KEY)

Plain Text:	0000 0000 0000 0000
Key:	0000 0000 0000 0000 0000 0000 0000 0000
Cipher Text:	fda7e7de58c913f4
Plain Text:	0123 4567 89ab cdef
Key:	0000 0000 0000 0000 0000 0000 0000 0000
Cipher Text:	8ebcf6ffd7289163

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, pages 450-466, Springer Berlin Heidelberg, 2007.

[2] Kyoji Shibutani, Takanori Isobe, Harunaga Hiwatari, Atsushi Mitsuda, Toru Akishita, and Taizo Shirai,“Piccolo: An Ultra-Lightweight Blockcipher”, pages 342-357, Volume-6917 Springer Berlin Heidelberg, 2011.

[3] T. Suzaki, K. Minematsu, S. Morioka, and E. Kobayashi, “TWINE: A Lightweight, Versatile Block Cipher” Cryptology ePrint Archive. Available at

[4] www.nec.co.jp/rd/media/ code/research/images/twine_LC11.pdf

[5] Beaulieu, R., Shors, D., Smith, J., Clark, S.T., Weeks, B., Wingers, L, “The SIMON and SPECK Families of Lightweight Block Ciphers”

Cryptology ePrint Archive, Report 2013/404, Available at http://eprint.iacr.org
[6] Collard, B., Standaert, F.-X. “A Statistical Saturation Attack against the Block Cipher PRESENT” In: Fischlin, M. eds. (2009) CT-RSA 2009. Springer, Heidelberg, pages. 195-211
[7] A. Menezes, P.C. van Oorschot, and S. Vanstone. The Handbook of Applied Cryptography. CRC Press, 1996.
[8] National Institute of Standards and Technology (NIST), “Advanced Encryption Standard (AES),” Federal Information Processing Standards Publication 197, November 26, 2001. Available at http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.
[9] Howard M. Heys, “A Tutorial on Linear and Differential Cryptanalysis” ,http://citeseer.nj.nec.com/443539.html
[10] Biham, E., Shamir, A. “Differential Cryptanalysis of DES-like Cryptosystems” Journal of Cryptology, vol. 4, no. 1, pp. 372, 1991
[11] 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. Available at https://eprint.iacr.org/2014/084.pdf
[12] Leander, G., Poschmann, A, “On the Classification of 4 bit S-boxes” In: Carlet, C., Sunar, B. (eds.) WAIFI 2007. LNCS, vol. 4547, pp. 159–176. Springer, Heidelberg (2007)
[13] M. Wang. “Di erential cryptanalysis of reduced-round PRESENT”. AFRICACRYPT 2008. LNCS. vol. 5023, pp. 40-49. Springer, Heidelberg. 2008
[14] Bogdanov, A., Rijmen, V.: “Zero Correlation Linear Cryptanalysis of Block Ciphers” IACR Eprint Archive Report 2011/123 (March 2011)
[15] Hadi Soleimany and Kaisa Nyberg. “Zero-correlation linear cryptanalysis of reduced-round lblock” Cryptology ePrint Archive, Report 2012/570, 2012. http://eprint.iacr.org/.
[16] 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.
[17] A. Bogdanov, D. Khovratovich and C. Rechberger,: “Biclique Cryptanalysis of the Full AES”, ASIACRYPT 2011, LNCS 7073, pp. 344–371, IACR, 2011.
[18] M. Albrecht, C. Cid. “Algebraic techniques in differential cryptanalysis” FSE 2009, LNCS, vol. 5665, pp. 193-208. Springer, Heidelberg. 2009
[19] E. Biham. “New Types of Cryptanalytic Attacks Using Related Keys”. Proceedings of Eurocrypt 93, LNCS. vol. 765, pp 398-409, Springer-Verlag. 1994
[20] A. Biryukov and D. Wagner, “Advanced Slide Attacks”, Proceedings of Eurocrypt 2000, LNCS. vol. 1807, pp. 589-606, Springer-Verlag. 2000
[21] 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
[22] 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
[23] 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), pages 138-148, July 2000.
[24] Gaurav Bansod, Nishchal Raval, 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.
[25] A. Poschmann. "Lightweight cryptography: cryptographic engineering for a pervasive world," In PhD Thesis, Faculty of Electrical Engineering and Information Technology, Ruhr-University Bochum, Germany, February 2009.
[26] 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
[27] M Kumar, SK Pal and A Panigrahi. “FeW: A Lightweight Block Cipher” Scientific Analysis Group, DRDO, Delhi, INDIA, Department of Mathematics, University of Delhi, INDIA 2014.
[28] F. X. Standaert, G. Piret, G. Rouvroy, J. J. Quisquater, and J. D. Legat, “ICEBERG: an involutional Cipher Efficient for Block Encryption in Reconfigurable Hardware,” In B. Roy and W. Meier, editors, Fast Software Encryption — FSE 2004, pages 279–298. Springer-Verlag, 2004.
[29] D. Hong, J. Sung, S. Hong, J. Lim, S. Lee, B. S. Koo, C. Lee, D. Chang, J. Lee, K. Jeong, H. Kim, J. Kim, and S. Chee, “HIGHT: A New Block Cipher Suitable for Low-Resource Device,” In L. Goubin and M. Matsui, editors, Cryptographic Hardware and Embedded Systems — CHES 2006, Vol. 4249 in LNCS, pages 46–59, Springer Berlin Heidelberg, 2006.
[30] T. Shirai, K. Shibutani, T. Akishita, S. Moriai, and T. Iwata, “The 128-bit blockcipher CLEFIA," In Fast Software Encryption- FSE'07, Vol. 4593 in LNCS, pages 181-195, Springer Berlin Heidelberg, 2007.