TentLogiX: 5-bit Chaos-Driven S-Boxes for Lightweight Cryptographic Systems

Maha Allouzi
mallouzi@kent.edu
Rahaei, Arefeh
arahaei@kent.edu

Abstract

Cryptography is a crucial method for ensuring the security of communication and data transfers across networks. While it excels on devices with abundant resources, such as PCs, servers, and smartphones, it may encounter challenges when applied to resource-constrained Internet of Things (IoT) devices like Radio Frequency Identification (RFID) tags and sensors. To address this issue, a demand arises for a lightweight variant of cryptography known as lightweight cryptography (LWC).

In developing any cryptographic algorithm, the substitution box (S-box) is a fundamental component, providing nonlinear functionality between inputs and outputs. Researchers have TentLogiX diverse S-box designs tailored for various applications, but only a few manage to balance the trade-offs among cost, performance, and security, particularly in the context of resource-constrained IoT devices.

This paper delves into the realm of S-boxes employed in popular LWC algorithms, categorizing them by their input–output bit sizes and elucidating their strengths and limitations. The focus then shifts to a novel 5-bit S-box design, utilizing chaotic mapping theory to introduce a randomized behavior.

Subsequently, the paper proposed TentLogiX a 5-bit S-box, constructed based on compound chaotic system, tent-logistic systems, which has better chaotic performance and wider sequences and explores its security robustness through various cryptanalysis techniques, including bijective analysis, nonlinearity assessment, linearity evaluation, and differential cryptanalysis. The paper concludes by presenting a thorough comparison that underscores the superiority of the TentLogiX 5-bit S-box over its 5-bit counterparts.

Keywords: Lightweight Cryptography, S-Box, Cryptanalysis

1. Introduction

With the emergence of the Internet of Things (IoT), lightweight symmetric-key algorithms are increasingly necessary to ensure data protection while in rest or in transit on various resource-constrained devices. IoT devices often have limited computational power and memory, necessitating cryptographic solutions that are both secure and efficient.

In symmetric-key cryptography, substitution boxes (s-boxes) are a fundamental component used to introduce non-linearity into the encryption process. However, s-boxes tend to consume a significant amount of hardware and computing resources, which can be a concern for resource-limited environments typical of IoT devices. Therefore, the design of lightweight cryptographic algorithms often involves the use of smaller, more efficient s-boxes. Lately a variety of S-boxes have been studied and TentLogiX by various researchers to support different cryptographic applications, some of these are not suitable for lightweight applications due to their heavy structure and high demand for resources (such as the 6-bit and 8-bit S boxes). In contrast some suffer during cryptanalysis (such as the 3- and 4-bit S boxes). A common approach in lightweight cipher design is to utilize 4-bit s-boxes. These s-boxes are small enough to be implemented efficiently while still providing an adequate level of security. The reduced size of 4-bit s-boxes helps in minimizing the hardware footprint and computational overhead, making them suitable for IoT applications.

For instance, the Piccolo cipher is an example of a lightweight block cipher that employs 4-bit s-boxes [1]. Piccolo is designed specifically for environments where resource constraints are a significant concern. Its use of 4-bit s-boxes, combined with other efficient cryptographic techniques, enables it to offer robust security while maintaining a low resource profile, making it a practical choice for IoT devices.

The second most common s-box is 8-bit S box due to robust strength but requires large amount of resources to get an acceptable performance, these S boxes are variant of AES [2]. This a tradeoff between performance cost and security is missing and creates the demand for a balanced s-box.

2. Lightweight S-Boxes

In this section, we will review existing S-box designs and their pros and cons.

2.1. Popular S-boxes

Various researchers and scientists have TentLogiX different S-box designs over the years. Some of these designs exhibit strong resistance to various attacks but require significant resources, while others offer better performance but have weaker security against attacks.

Most of these S-boxes accept 3-bit, 4-bit, 5-bit, 6-bit, or 8-bit inputs and produce outputs of the same size or in a compressed format [3]. The most common sizes of S-boxes are 8-bit and 4-bit. Since a 4-bit S-box is usually much more compact in hardware than an 8-bit S-box, many lightweight block ciphers and hash functions use 4-bit S-boxes [4], such as PRESENT [5]. Vishal [6] presents an overview of S-boxes used by popular lightweight cryptography algorithms such as The PRINT [7], PRESENT [5], RECTANGLE [8], EPCBC [9] ,TWINE [10], LED [11], SKINNY [12] PICCOLO [13], DESL/DESXL [14], ASCON [15] PRIMATE [16], ICEPOLE [17] and SHAMASH [18],KLEIN [19], PUFFIN [20],LBlock (such 8 different 4 × 4 bit S-boxes) [21], SPONGENT [22].

3-bit S-box: SEA [23] cipher uses a 3-bit S-box, applied bitwise to groups of three data words, making it efficient for hardware and software. The S-box was chosen for its efficiency, meeting security criteria with a λ-parameter of 1/2 and δ-parameter of 1/4, requiring minimal operations per data word. However, the 3-bit design's limited configurations and the presence of two fixed points could pose security risks, potentially making it vulnerable to attacks. Despite this, SEAn,b resists linear and differential cryptanalysis, needing at least 3n/4 rounds for robust security, similar to AES.

Table 1 : 3-bit Sbox design

(x)	0	1	2	3	4	5	6	7
S(x)	0	5	6	7	4	3	1	2

4-bit S-box: PRESENT [5] cipher has a block length of 64 bits and supports two key lengths: 80 bits and 128 bits. Each of the 31 rounds includes an XOR operation to introduce a round key K_i for $i \leq i \leq 32$, with K_{32} used for post-whitening, a linear bitwise permutation, and a non-linear substitution layer. ("PRESENT: An Ultra-Lightweight Block Cipher - IACR") PRESENT uses a 4-bit S-box with 8064 possible configurations and is vulnerable to differential cryptanalysis. There are 4 measures to evaluate the security of a cipher against differential cryptanalysis. Define the differential-uniformity of S as:

$$Diff(S) = max_{\Delta I \neq 0, \Delta o}\, ND_s(\Delta I | \Delta o)$$

the smaller the value of *Diff* (S), the more secure the S-box against differential cryptanalysis [4].

RECTANGLE [8] incorporates a 4 × 4 S-box (Table 2) from PRESENT and reduces the number of rounds to 25 (compared to 31) to enhance software efficiency. It features an AES-like structure with the removal of certain functionalities (a slight modification in the SP Network) and introduces the bit-slice technique to boost performance and reduce costs. However, like PRESENT, it is also vulnerable to various cyber-attacks. Due to the asymmetric design of its permutation layer, RECTANGLE achieves an excellent security-performance tradeoff. EPCBC [9] is a lightweight cipher with a 96-bit key and 48-bit/96-bit block sizes, designed for Electronic Product Code (EPC) encryption using RFID-tags. EPCBC is based on PRESENT, with improvements to resist related-key differential attacks. It leverages PRESENT's security analyses and introduces new results for smaller block sizes. The cipher resists various attacks and is more efficient for EPC encryption than AES and PRESENT, avoiding the 33% overhead of 128-bit blocks. TWINE [10] is a 64-bit lightweight block cipher supporting 80 and 128-bit keys, designed for efficient hardware and software implementations. Unlike PRESENT, TWINE uses a Type-2 generalized Feistel structure with enhanced diffusion, omitting bit permutation and Galois-Field matrices. It relies on a single 4-bit S-box and simple XOR operations. Security analysis shows TWINE is vulnerable to impossible differential attacks, breaking 23-round TWINE-80 and 24-round TWINE-128. To ensure a sufficient security margin, TWINE uses 36 rounds for both key lengths, balancing security with implementation efficiency [10]. LED [11] is a 64-bit block cipher that follows AES-like design principles, allowing for straightforward calculations of the number of active S-boxes during encryption. It supports key sizes of 64 and 128 bits and arranges the cipher state in a 4x4 grid, where each nibble corresponds to an element from GF(2^4) with polynomial $x^4 + x + 1$ for field multiplication. SKINNY [12] is a tweakable block cipher family designed to rival NSA's SIMON in performance and offer stronger security against differential and linear attacks. It supports 64-bit and 128-bit blocks, with the internal state organized as a 4x4 array of nibbles or bytes. MANTIS, a low-latency variant of SKINNY, is optimized for memory encryption. Piccolo [13] is a 64-bit lightweight block cipher supporting 80 and 128-bit keys, designed for high security and compact hardware implementation. It uses a variant of the generalized Feistel network and achieves notable efficiency in energy consumption and gate equivalents (683 for 80-bit and 758 for 128-bit keys). The F-function in Piccolo operates on a 4x4 grid of nibbles, using two S-box layers separated by a diffusion matrix, similar to other lightweight ciphers like PRESENT and TWINE. It is resistant to known attacks, including related-key differential and meet-in-the-middle (MITM) attacks, making it suitable for highly constrained environments like RFID tags and sensor nodes [13].

The trend of using 4 × 4 S-box from PRESENT continues, KLEIN [19], Puffin [20], LBlock (such 8 different 4 × 4 bit S-boxes) [21]and SPONGENT (uses it for $b/4$ times parallelly, where b is the fixed number of bits of a state) [22].

Table 2 : 4-bit S-box design

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

5-bit S-box:

Ascon [15] employs a 5-bit S-box: $S : \{0, 1\}^5 \rightarrow \{0, 1\}^5$, for its substitution layer. The S-box is usually implemented in a bit sliced manner, allowing operations on entire 64-bit words.

S-boxes with differential factors can be vulnerable to differential-linear attacks. However, the Shamash [18] 5 bits S-box (Table 3) lacks differential factors, making it immune to such attacks.

The Shamash S-box is resistant to linear-differential attacks due to its unique structure. While it contains 31 linear structures, its inverse has none, which prevents attackers from exploiting linear patterns in the inverse S-box. In contrast, the Ascon S-box has 91 linear structures, and its inverse includes two undisturbed bits, making it more vulnerable to attacks in specific rounds. The absence of linear structures in Shamash's inverse S-box means that attackers face significant challenges in finding exploitable patterns, enhancing the security of the Shamash cipher.

Table 3 : 5-bit S-Box Design.

(x)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S(X)	16	14	13	2	11	17	21	30	7	24	18	28	26	1	12	6
	31	25	0	23	20	22	8	27	4	3	19	5	9	10	29	15

DESL [24] (DES Lightweight), a new block cipher derived from the traditional DES (Data Encryption Standard). Unlike DES, DESL utilizes a single S-box, which is used eight times. The S-box used in DESL is a 6-to-4 bit S-box, $S : \{0, 1\}^6 \rightarrow \{0, 1\}^4$ (Table 4). This means that it takes a 6-bit input and produces a 4-bit output, similar to the original DES S-boxes.

Table 4 : 6-bit S-box design

X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S(X)	14	5	7	2	11	8	1	15	0	10	9	4	6	13	12	3
	5	0	8	15	14	3	2	12	11	7	6	9	13	4	1	10
	4	9	2	14	8	7	13	0	10	12	15	1	5	11	3	6
	9	6	15	5	3	8	4	11	7	1	12	2	0	14	10	13

8-bit S-box: The ICEBERG [25] algorithm employs an 8 × 8 S-box , denoted as S: $\{0,1\}^8 \rightarrow \{0,1\}^8$ (Table 5) inspired by the AES algorithm. This S-box is spread over three stages, S0, S1,S0, using 4 × 4 S-boxes (Table 6, 7) in parallel to achieve the substitution. While many cryptographic algorithms use 8-bit S-boxes due to their robustness, they are often costly and resource-intensive for implementation on constrained IoT devices.

Table 5 : 8-bit S-box design

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	24	c1	38	30	e7	57	df	20	3e	99	1a	34	ca	d6	52	fd
10	40	6c	d3	3d	4a	59	f8	77	fb	61	0a	56	b9	d2	fc	f1
20	07	f5	93	cd	00	b6	62	a7	63	fe	44	bd	5f	92	6b	68
30	03	4e	a2	97	0b	60	83	a3	02	e5	45	67	f4	13	08	8b
40	10	ce	be	b4	2a	3a	96	84	c8	9f	14	c0	c4	6f	31	d9
50	ab	ae	0e	64	7c	da	1b	05	a8	15	a5	90	94	85	71	2c
60	35	19	26	28	53	e2	7f	3b	2f	a9	cc	2e	11	76	ed	4d
70	87	5e	c2	c7	80	b0	6d	17	b2	ff	e4	b7	54	9d	b8	66
80	74	9c	db	36	47	5d	de	70	d5	91	aa	3f	c9	d8	f3	f2
90	5b	89	2d	22	5c	e1	46	33	e6	09	bc	e8	81	7d	e9	49
a0	e0	b1	32	37	ea	5a	f6	27	58	69	8a	50	ba	dd	51	f9
b0	75	a1	78	d0	43	f7	25	7b	7e	1c	ac	d4	9a	2b	42	e3
c0	4b	01	72	d7	4c	fa	eb	73	48	8c	0c	f0	6a	23	41	ec
d0	b3	ef	1d	12	bb	88	0d	c3	8d	4f	55	82	ee	ad	86	06
e0	a0	95	65	bf	7a	39	98	04	9b	9e	a4	c6	cf	6e	dc	d1
f0	cb	1f	8f	8e	3c	21	a6	b5	16	af	c5	18	1e	0f	29	79

Table 6:S_0 4X4 s-box

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
d	7	3	2	9	a	c	1	f	4	5	e	6	0	b	8

Table 7:S_1 4X4 S-box

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
4	a	f	c	0	d	9	b	e	6	1	7	3	5	8	2

2.2. S-box Facts

Table 8 lists the key facts and design issues related to existing S-boxes, focusing on the number of bits, security, and implementation cost. It highlights how smaller S-boxes, such as 3-bit and 4-bit designs, are easy to implement with minimal computational overhead but offer insufficient security due to their limited number of possible values, making them vulnerable to brute-force and differential attacks. Larger S-boxes, such as 8-bit or higher, provide enhanced security by increasing the complexity and possible output values, but they come at a higher implementation cost in terms of memory and processing power. The table emphasizes the trade-offs between the size of the S-box, the level of security achieved, and the resources required for efficient implementation in different cryptographic systems.

Table 8: Existing S- boxes and related concerns

S-box type	Facts
3-bit	The implementation cost is quite minimal, but with just 8 potential values, it may be easily broken. A sufficient degree of security could not be achieved even with an increase in the number of rounds.
4-bit	Low resource needs and low security (with just 16 potential options) Increasing the number of rounds might fix this problem, however it has a negative impact on execution time.
5-bit	A slight increase in resource requirements compared to a 4-bit S-box, A moderate level of security with 32 values.
6-bit	Requires slightly more memory (to store 64 possible values) and somewhat higher processing power (to derive and process 64 possible values) compared to a 4-bit S-box.

	These increased demands result in higher energy consumption relative to a 4-bit S-box. Additionally, these factors may lead to an increased requirement for physical area (gate equivalents).
8-bit	Requires substantial memory to store 256 values and significantly higher processing power to derive and process them. These demands result in higher energy consumption compared to 4-bit and 6-bit S-boxes. Additionally, these factors could increase the requirement for physical area (gate equivalents).

3. Background

Over the past decade, there has been significant interest in studying the behavior of chaotic systems, which are characterized by their sensitive dependence on initial conditions and resemblance to random behavior. Chaos has potential applications in various components of digital communication systems, including compression, encryption, and modulation. The discovery of self-synchronization in chaotic oscillations [26] has led to a surge of research on the application of chaos in cryptology.

3.1. Chaotic S-boxes

Chaotic maps have many advantages in applications to cryptography because of their simple structure. The general mathematical model of chaotic mapping can be expresses as:

$$x(n+1) = f[x(n)] \qquad (1)$$

Where F[x] denotes functions regarding x. x(0) is the initial state value of the map and $\{x(1), x(2) \ldots\}$ is the output sequence values. For discrete maps, Lyapunov exponent is define as:

$$\lambda = \lim_{n \to \infty} \frac{1}{n} \left(\sum_{i=1}^{n} \log |f'[x_i]| \right) \qquad (2)$$

Where $f'[x_i]$ is the derivative of $f[x_i]$. if $\lambda > 0$, then the chaotic behaviors exist in the system. In this section , we first discuss the two maps used in our TentLogiX S-box; logistic map and the tent map. Then we TentLogiX *TentLogix* a new discrete compound 5- bit chaotic S-box, which has better chaotic performance and wider chaotic range than both logistic and tent maps.

3.2. Logistic chaotic map

The logistic map is one of the famous chaotic maps, which has a simple mathematical structure but yet complex chaotic behavior. The mathematical model of Logistic map is [26]:

$$x(n+1) = \mu * x(n) * (1 - x(n)), \qquad (3)$$

Where μ is the system parameter in the range of [0,4]. In order to determine the range of parameters corresponding to its chaotic phenomena, we calculate the Lyapunov exponent under different parameters μ and found the chaotic range of logistic map was μ ∈ [3.57,4]. The birfurcation diagram of logistic map is shown in figure 1-a and the distribution under μ = 3.78 is shown in figure 1-b

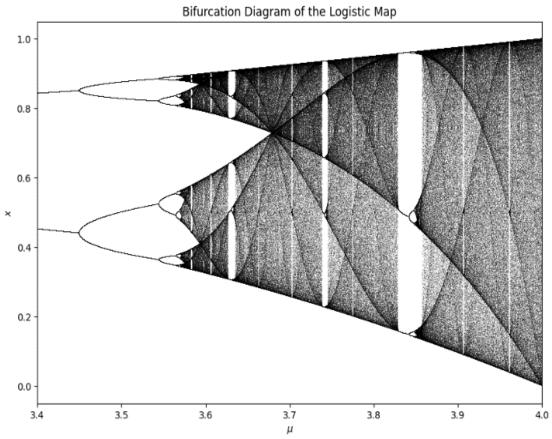


Figure 1-a bifurcation diagram of logistic map

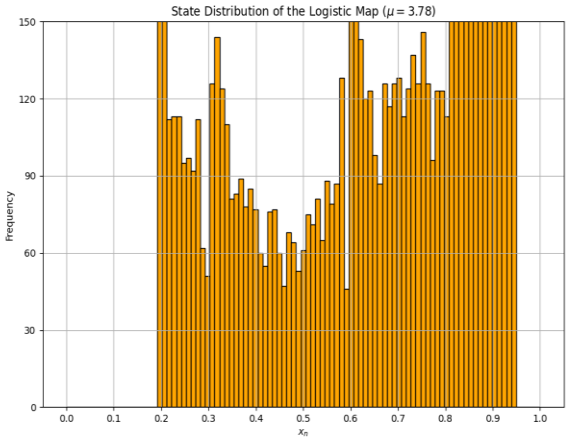


figure 1-b state distribution under μ=3.78

The logistic map has three primary limitations. First, its chaotic behavior is confined to a narrow range of the parameter μ∈ [3.57,4]. Even within this range, certain values of μ can result in non-chaotic behavior. Second, the distribution of state values within the interval [0,1] is non-uniform. For instance, as noted in [27], the logistic map at μ=3.9 exhibits aperiodic behavior rather than chaotic. Finally, if μ is fixed rather than varied within the range, it leads to a reduced key space. These issues limit the logistic map's effectiveness and application potential [28].

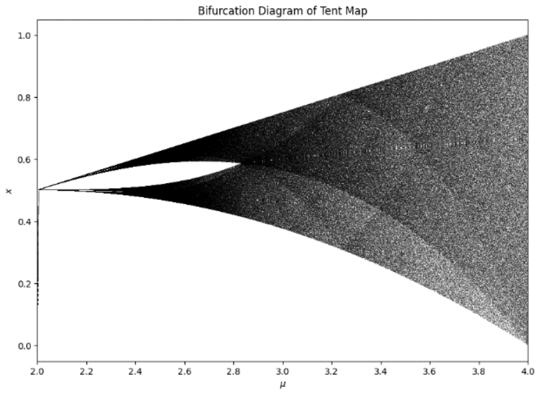
3.3. Tent chaotic map

The tent map is another chaotic system, distinguished by its tent-like shape in the bifurcation diagram. The mathematical model of the tent map is described as follows [29]:

$$x(n+1) = \begin{cases} f_1\,[x(n)] = \frac{\mu}{2} * x(n) \\ f_2[x(n)] = \frac{\mu}{2} * (1 - x(n)) \end{cases} \qquad (4)$$

The parameter μ for the tent map ranges from [0, 4]. According to Equation (4), the Lyapunov exponent λ of the tent map can be calculated as λ=log(μ/2). Thus, when μ>2, λ becomes positive, and when μ=4 ,λ reaches its maximum value, $\lambda_{\max} = \log(2) = 0.6931$. The chaotic behavior of the tent map is illustrated in the bifurcation analysis shown in Figure 2a, which indicates that the chaotic range extends from μ∈[2,4].

The state distribution for μ=3.78 is depicted in Figure 2b. However, like the logistic map, the tent map also suffers from similar issues: a limited chaotic range and a non-uniform distribution of output state values.



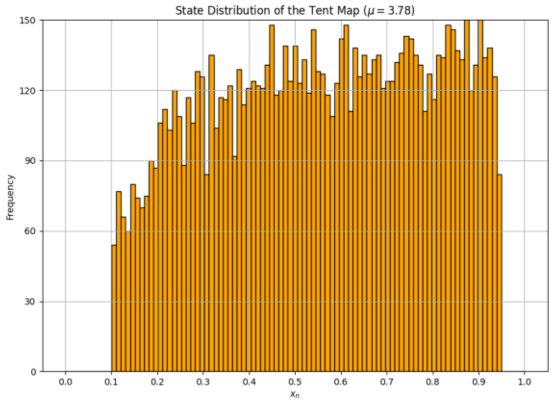


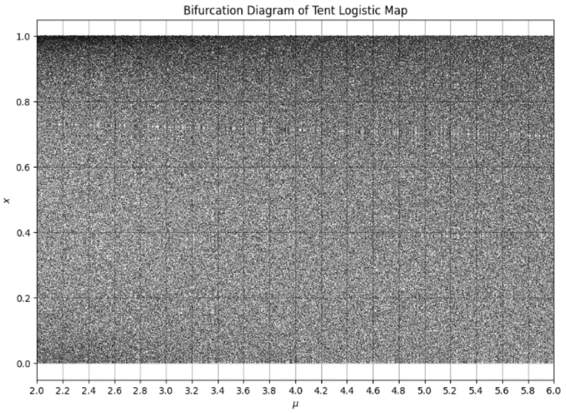
Figure 2-a bifurcation diagram of Tent map

Figure 2 b state Distribution of Tent map (μ=3.78)

3.4. Tent-Logistic Chaotic map

To address the issues, present in the logistic and tent maps, we employed a compound approach combining the logistic and tent maps [6]. The mathematical model of this compound system is as follows:

$$x(n+1) = \begin{cases} f_1\,[x(n)] = \frac{4(9-\mu)}{9*x(n)} * x(n) * \big(1-x(n)\big) + \frac{2\mu}{9} * x(n) \\ f_2[x(n)] = \frac{4(9-\mu)}{9*x(n)} * x(n) * \big(1-x(n)\big) + \frac{2\mu}{9} * (1-x(n) \end{cases} \quad (5)$$



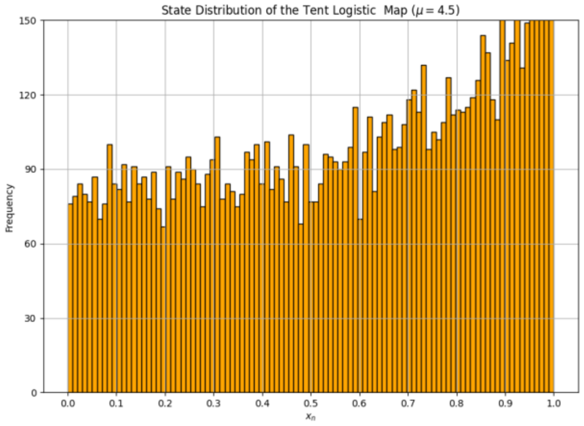


Figure 3 a- bifurcation diagram of Tent-Logistic map

Figure3 b state Distribution of Ten-Logistict map (μ=3.78)

Where μ is the system parameter within the range [0, 9], when μ=0, equation (5) simplifies to the optimal chaotic logistic map, and when μ=9, equation (5) reduces to the optimal chaotic tent map. Thus, both the optimal chaotic logistic and tent maps can be seen as specific instances of equation (5).

In Figure 3a, it is evident that the chaotic range covered the entire interval of μ∈ [0, 9], which is significantly broader than that of the logistic or tent maps. Additionally, the output sequences are uniformly distributed within the [0, 1] range, as shown in Figure 3b. Consequently, the Tent Logistic System exhibits superior chaotic performance compared to the logistic and tent maps.

3.5. TentLogiX: The TentLogiX S-box

This section gives an inclusive view of the design criteria of *TentLogiX* the TentLogiX 5-bit chaos-based S-box for lightweight cryptography algorithms. The following are the steps for generating the 5-bit chaos S-box:

1. Initialize Variables:
   a. Set the initial value x to a chosen starting value between 0 and 1.
   b. Define a parameter μ that controls chaotic behavior (e.g., μ = 4.5).
2. Define Logistic-Tent Map Function:
   a. For each input x ($where\ 0 \leq x \leq 1$):
      i. If $x < 0.5$ apply the formula:

$$x(n+1) = \begin{cases} f_1\,[x(n)] = \dfrac{4(9-\mu)}{9 * x(n)} * x(n) * \big(1 - x(n)\big) + \dfrac{2\mu}{9} * x(n) & , x < 0.5 \\ f_2[x(n)] = \dfrac{4(9-\mu)}{9 * x(n)} * x(n) * \big(1 - x(n)\big) + \dfrac{2\mu}{9} * \big(1 - x(n)\big) & , otherwise \end{cases}$$

3. Generate the S-box:
   a. Create an empty list S-Box
   b. For each iteration (repeat 32 times):
      i. Computer the next value of x using logistic-tent map function:
      ii. Scale the compound value x to an integer between 0 and 31:

$$value = \lfloor x * 32 \rfloor$$

   c. Ensure the value is unique by checking if it is already in the S-Box:
      i. If it is, computer the next value of x and scale it again until a unique value is found.
   d. Add the unique value to the S-box list.

4. Return the s-box, which contains 32 unique values, each representing a 5-bit number (from 0 to 31) .

Table 9 *TentLogiX* S-box with x= 0.66

x		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S (x)	0	25	17	30	4	12	27	28	10	3	8	21	24	19	11	26	15
	1	31	2	20	7	14	5	13	9	23	16	18	29	22	0	1	6

3.6. Design criteria of 5-bit S-box

To build a simple but robust 5-bit S-box, $S : F25 \rightarrow F25$, that could be easily implemented on resource-constrained IoT device, the following simple but security efficient rules need to apply:

1. S-box, S, must have distinct 32 elements (0–31) spread over 16 columns and 2 rows that satisfies bijective property (Section 4.1).
2. Generate the complex chaotic sequence of the elements in 5- bit S-box using tent- logistic map equation as defined follows:

$$x(n+1) = \begin{cases} f_1\,[x(n)] = \dfrac{4(9-\mu)}{9 * x(n)} * x(n) * \big(1 - x(n)\big) + \dfrac{2\mu}{9} * x(n) \\ f_2[x(n)] = \dfrac{4(9-\mu)}{9 * x(n)} * x(n) * \big(1 - x(n)\big) + \dfrac{2\mu}{9} * (1 - x(n) \end{cases}$$

Where μ is the system parameter, and $\mu \in [0,4]$.

3. Any value, x$_i$, in S, must be different from its column index, Ci, to avoid a fixed point, i.e.,

$$x_i \neq \begin{cases} c_i & if\ x_i \in R_0 \\ c_{(15+i)} & if\ x_i \in R_1 \end{cases} \tag{6}$$

where, R_0 is the 0th row and R_1 is the 1st row, $R_0 0 \subset S$ and $R_1 \subset S$

4. An input value, I_{ni}, and its corresponding value, x_i, in S must have bit variation of n bit(s), 0 < n ≤ 5, to meet overall Strict Avalanche Criteria (SAC), i.e.,

$$Bit\ var(x_i, I_{ni}\ ) \geq n, \forall\ x_i \in S \tag{7}$$

where , I_{ni}: $f(Ri, Ci)$, $Ri \rightarrow \{0, 1\}$, $Ci \rightarrow \{0, 1\}^4$ and , $xI_i \rightarrow \{0, 1\}^5$

Here, in design criteria (2), The Tent-Logistic map combines the advantages of the logistic and tent maps, creating a broader chaotic range and a more uniform distribution of state values. This hybridization helps mitigate the fixed-point problem. The paper's analysis shows that the chaotic behavior of the Tent-Logistic map is spread over a wider range of parameters, and the output sequences are more uniformly distributed, which minimizes the likelihood of fixed points occurring. By expanding the parameter space and enhancing randomness, the Tent-Logistic system ensures better performance for cryptographic purposes, particularly in eliminating vulnerabilities like fixed points that attackers could exploit.

By implementing the above-defined set of rules, the total number of possible random 5-bit S-boxes is 31! $\approx 8.22 \times 10^{33}$, which is vastly larger than the number of possible 4-bit S-boxes, which is 15! $\approx 1.3 \times 10^{12}$. This exponential increase in the number of possible configurations enhances the security of the S-box significantly. The larger key space makes brute-force attacks far more difficult, while the increased complexity of the S-box design adds robustness against cryptanalytic attacks such as linear and differential cryptanalysis. As a result, the TentLogiX 5-bit S-box offers not only a broader range of randomness but also an increased resistance to cryptographic vulnerabilities, making it a more secure option for lightweight cryptographic systems.

4. Security Analysis:

In this section, we present a comprehensive security analysis of the chaos-based 5-bit S-box derived from the tent-logistic map, addressing several critical cryptographic properties. First, bijectivity is a fundamental property that ensures each input value maps uniquely to an output value, thereby preventing information loss during encryption. We evaluate the TentLogiX S-box for bijectivity to confirm its suitability for cryptographic applications. Nonlinearity and linearity are also crucial aspects of S-box design. Nonlinearity measures the deviation from an ideal linear transformation, while linearity assesses the S-box's resistance to linear cryptanalysis. Our analysis provides a detailed examination of these properties to evaluate the S-box's cryptographic strength.

Furthermore, we investigate the TentLogiX S-box's resistance to differential cryptanalysis by computing its Differential Approximation Probability (DAP). A low DAP value indicates higher resistance to differential attacks. Additionally, we assess the S-box's cryptographic robustness by measuring its Strict Avalanche Criterion (SAC) and Bit-Independence Criterion (BIC-SAC). The SAC evaluates how changes in a single input bit propagate through the S-box, ensuring strong avalanche effects. BIC-SAC focuses on bit-independence, which is crucial for resisting certain advanced attacks. It also provides a comparative analysis of the cryptanalysis of the newly TentLogiX 5-bit S-box with 5-bit S-boxes found in VISHAL [29], ASCON [30], PRIMATE [16], ICEPOLE [17] and SHAMASH [18].

The improved performance of the TentLogiX S-box is clearly demonstrated by the data presented in Table 10 and Figure 4.

Table 10 Cryptoanalysis of various 5-bit boxes

S-box 5(bit)	Linear Probability	Nonlinearity (H_d)	DAP	SAC	BIC-SAC
TentLogiX	0.25	2.75	0.25	0.5	0.51
VISHAL [29]	0.25	2.625	0.25	0.51	0.53
ASCON [30]	0.25	2.5	0.25	0.57	0.58
PRIMATE	0.375	2.5	0.0625	0.52	0.54
ICEPOLE	0.25	1.531	0.25	0.43	0.44
SHAMASH	0.375	2.5	0.0625	0.56	0.57

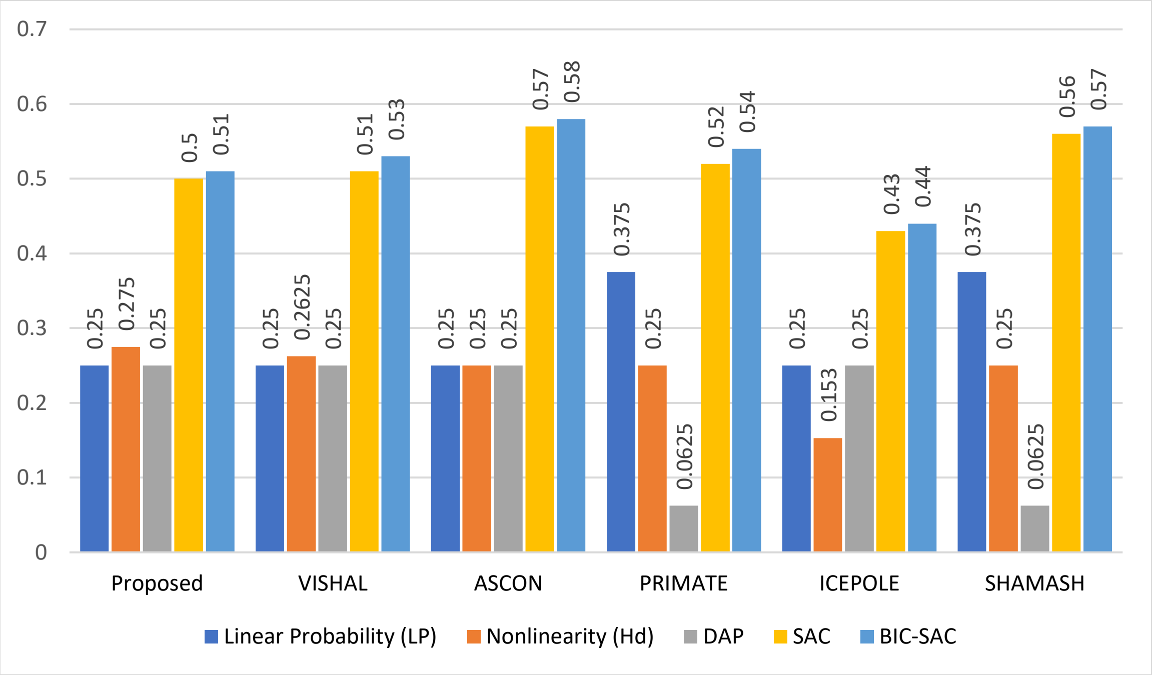


Figure 4 Cryptanalysis of various 5-bit S-box

4.1. Bijective property

The concept of bijectivity in the design of an S-box, particularly for an $m \times m$ S-box (Here m=5), is a fundamental characteristic that ensures each input maps to a unique output, thereby creating a one-to-one correspondence between the input and output sets. This property is crucial for achieving strong cryptographic functions, as it guarantees that the S-box is both injective (no two inputs map to the same output) and surjective (every possible output is mapped by some input). The bijective nature of an S-box enhances its resistance to cryptographic attacks by ensuring that the distribution of the output is uniformly spread across the range of possible values, which, for an m-bit S-box, is from 0 to 2^{m-1}. To verify the bijectivity of an S-box, one typically employs methods such as the calculation of the Hamming weight, which assesses the number of 1's present in the binary representation of each possible output. And it is defined as:

$$H_{wt}\left(\sum_{i=1}^{m} b_i f_i\right) = 2^{m-1}$$

where $b_i \in \{0, 1\}$ and $(b_1, b_2, \ldots, b_m) \neq (0, 0, \ldots, 0)$ for each Boolean function, $f_i(1 \leq i \leq m)$. Here, f_i fulfils the bijective property by balancing 0 and 1. Also, TentLogiX has all distinct values from 0 to 31, and thus manifests the bijective property.

The design process involves careful selection and testing of the S-box transformation to confirm that it maintains bijectivity, thereby contributing to the strength and security of the encryption algorithm it is used in.

4.2. Nonlinearity

The design of S-box functions aims to introduce nonlinearity into cryptographic algorithms, making them secure against straightforward mathematical attacks. It's imperative that these algorithms resist attempts at decryption through systems of equations that could predict S-box behavior. The method of selection for S-box values is based on a dynamic, unpredictable system utilizing chaotic sequences, rendering the task of finding a deterministic relationship between input values and corresponding outputs impossible. This is further explored in Section 3.6, which elucidates how a 5-bit input is arbitrarily transformed, with nonlinearity assessment methods such as Hamming distance or the Walsh spectrum being applied to gauge this attribute.

The strategy involves examining all pairs of input and output, calculating the Hamming distance H_d – the count of different corresponding bits. For the newly formulated 5-bit S-box, the Hamming distance ranges from a minimum of 1 to a maximum of 5, as shown in Table 11 below, The average Hamming distance H_d is a critical metric, with the newly developed S-box achieving an Hd of 2.75, an indicator of its robust nonlinearity when compared to similar 5-bit S-boxes, as depicted in Figure 5. An S-box with a higher Hd is preferred since it indicates stronger nonlinearity. Consequently, the TentLogiX S-box design fulfills the criteria for nonlinearity, a desirable attribute for cryptographic systems.

Table 11 Nonlinearity measure through Hamming distance (Hd)

Input	output	Hamming distance (H_d)	Input	output	Hamming distance (H_d)
0 (00000)	25(11001)	3	16(10000)	31(11111)	4
1(00001)	17(10001)	1	17(10001)	2(00010)	3
2(00010)	30(11110)	3	18(10010)	20(10100)	2

3(00011)	4(00100)	3	19(10011)	7(00111)	2
4(00100)	12(01100)	1	20(10100)	14(01110)	3
5(00101)	27(11011)	4	21(10101)	5(00101)	1
6(00110)	28(11100)	3	22(10110)	13(01101)	4
7(00111)	10(01010)	3	23(10111)	9(01001)	4
8(01000)	3(00011)	3	24(11000)	23(10111)	4
9(01001)	8(01000)	1	25(11001)	16(10000)	2
10(01010)	21(10101)	5	26(11010)	18(10010)	1
11(01011)	24(11000)	3	27(11011)	29(11101)	2
12(01100)	19(10011)	5	28(11100)	22(10110)	2
13(01101)	11(01011)	2	29(11101)	0 (00000)	4
14(01110)	26(11010)	2	30(11110)	1(00001)	5
15(01111)	15(01111)	0	31(11111)	6(00110)	3

Additionally, enhancing the nonlinearity could involve optimizing the chaotic map used for S-box generation or introducing more complex algebraic constructions that further obscure the relationship between the S-box inputs and outputs. These measures would strengthen the encryption against cryptanalytic attacks and ensure a higher level of security.

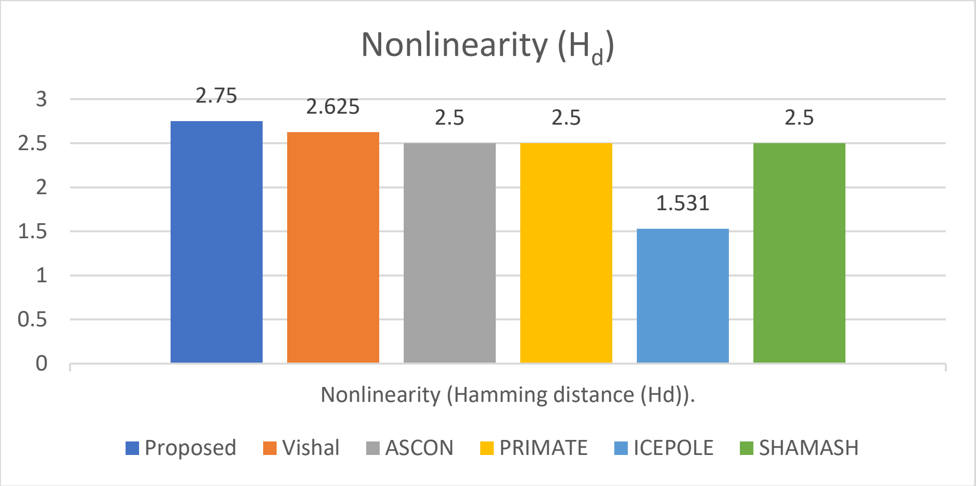


Figure 5 Nonlinearity (Hamming Distance)

4.3. Linear approximation probability (LP)

The Linear Approximation Probability (LP) metric, a concept introduced by Matsui in 1993, is utilized to gauge the maximum discrepancy in the correlation of input-output pairings within an S-box. This metric considers the input and output differentials, denoted as Δx and Δy, to quantify the deviation from expected behavior. The LP for an S-box is calculated by identifying the maximum deviation from uniform distribution, represented by the equation:

$$LP = max_{\Delta x, \Delta y \neq 0} \left| \frac{\#\{x \in X | x.\Delta x = S(x).\Delta y\}}{2^n} - \frac{1}{2} \right|$$

Table 12 TentLogiX Linear Approximation Table

		Output Sum																														
Input Sum		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	0	16	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
	1	0	0	-2	-2	-4	-4	-2	-2	0	-4	-2	2	4	0	-2	2	-6	-2	4	0	2	-2	0	4	-2	-2	0	0	6	-2	4	-4
	2	0	-2	-2	4	4	-2	2	4	2	0	0	-2	2	4	0	2	0	6	2	0	0	2	-6	4	-2	-4	0	-2	2	4	-4	-2
	3	0	-2	-4	2	-4	-2	4	-2	-2	0	-6	4	-2	-4	-2	4	2	4	-2	0	-2	-4	-2	-4	0	-2	4	2	0	2	0	2
	4	0	0	2	2	0	4	2	-2	4	0	-2	2	0	0	-6	-6	-6	2	-4	4	2	-2	-4	0	2	-2	-4	0	-2	-2	0	0
	5	0	-4	-4	0	0	0	0	0	0	4	4	0	-4	4	-4	4	0	0	0	0	4	0	0	4	0	0	0	0	0	-4	4	8
	6	0	-2	0	-2	4	2	4	-6	-2	0	-2	0	2	-4	2	4	2	0	-2	-4	2	0	-2	4	0	2	-4	-2	0	-6	-4	-2
	7	0	-6	2	4	0	2	-2	0	-2	0	0	2	-2	0	4	-2	0	-2	2	0	0	-2	-2	4	10	0	4	2	2	0	0	-2
	8	0	0	2	-2	-4	-4	-2	-6	-4	0	2	2	0	4	-2	-2	2	2	0	-4	-2	6	-4	0	2	-2	-4	4	-2	2	0	0
	9	0	4	4	4	-4	0	4	4	-8	4	0	0	4	0	0	0	0	0	-4	0	0	0	0	4	0	0	0	-4	0	0	4	0
	10	0	-2	0	-6	0	2	0	-2	-2	0	2	-8	2	0	-2	0	-2	4	-2	0	-6	-4	2	0	4	-2	0	-2	4	2	0	2
	11	0	2	2	0	4	2	2	-4	-2	0	4	2	-2	4	0	2	4	2	2	4	0	-6	2	0	-2	-4	0	2	-2	0	4	-6
	12	0	0	-4	0	0	-4	4	-4	4	4	4	0	0	-4	0	0	0	0	0	4	0	4	0	0	4	4	0	-4	0	4	4	-4
	13	0	0	2	-2	4	0	2	2	-4	0	2	2	-4	-4	-2	2	-6	-2	4	-4	2	2	0	-4	2	-6	0	-4	-2	2	0	0
	14	0	-2	2	-4	4	2	-2	0	-2	4	-4	-2	2	0	0	2	-4	2	-2	0	4	2	-2	0	-2	4	4	6	-2	4	4	-2
	15	0	-2	0	2	-4	2	0	2	2	4	-2	-4	-2	0	-2	4	-2	0	2	0	-6	4	2	0	0	-2	0	2	-4	-6	0	-6

	16	0	0	0	4	4	-4	-4	0	-6	-6	2	-2	-2	-2	-2	2	-2	2	-2	6	-2	2	-2	-2	0	4	0	0	0	-4	0	0
	17	0	0	2	-2	0	0	-2	2	2	-2	-4	4	2	6	0	8	0	0	-2	2	0	0	2	-2	6	2	-4	-4	-2	2	0	0
	18	0	-6	2	0	4	-6	-2	0	0	2	-6	0	0	-2	2	-4	2	0	0	2	-2	0	4	2	-2	-4	-4	-2	-2	0	4	2
	19	0	2	4	2	4	2	-4	-2	4	2	0	2	4	-2	-4	2	4	-2	0	-2	-4	2	-4	-2	0	-2	4	-2	4	-2	4	2
	20	0	0	6	-6	0	-4	6	6	2	-2	0	0	-2	-2	-4	0	4	0	2	2	0	0	-2	2	2	2	0	4	2	-2	0	0
	21	0	-4	-4	-4	0	0	0	4	-2	2	2	2	2	2	2	-2	2	-2	-6	2	2	2	-2	-6	0	-4	0	0	4	-4	0	-4
	22	0	2	0	-2	0	2	0	-2	0	-2	0	2	0	-2	0	2	0	-2	-4	6	0	6	4	6	0	-6	4	2	0	2	-4	2
	23	0	6	-2	0	4	-6	-2	0	0	6	-2	0	-4	2	-2	0	-2	-4	-4	-2	-2	-4	0	2	2	0	0	2	2	0	-4	-2
	24	0	4	2	-2	0	-4	2	-2	2	2	0	0	2	2	8	0	-4	4	2	2	0	0	-2	-2	2	-2	4	0	-2	-6	0	4
	25	0	0	0	0	0	0	4	-4	-2	-2	-6	-6	-2	6	-2	-2	2	-6	2	2	2	2	-2	-2	0	0	4	-4	0	0	0	0
	26	0	-2	4	2	0	2	4	-2	0	2	0	2	-4	2	4	2	-4	-2	0	2	-4	2	0	-2	-4	2	-4	2	8	2	0	2
	27	0	2	2	-4	-4	2	-6	0	0	2	-2	0	-8	-2	2	0	2	4	0	2	2	0	-4	2	-2	0	0	-6	2	0	0	-2
	28	0	-4	0	-4	0	0	0	0	-2	2	2	6	2	2	-2	-2	-2	-2	2	2	-6	-2	-2	2	-4	4	4	-4	-4	0	-4	0
	29	0	4	-6	-2	4	4	2	2	-2	-2	-4	4	-2	2	0	-4	0	4	2	-2	-4	4	2	2	2	2	0	0	2	-2	4	0
	30	0	-2	2	0	0	-2	2	0	4	-6	2	0	-4	2	2	0	-2	0	-8	-6	-2	0	0	2	-2	0	4	-2	-2	0	4	-2
	31	0	-2	4	2	0	-2	0	-2	0	2	0	2	0	2	-4	-2	0	6	0	-2	4	2	8	-2	0	2	4	-2	4	-2	-4	-2

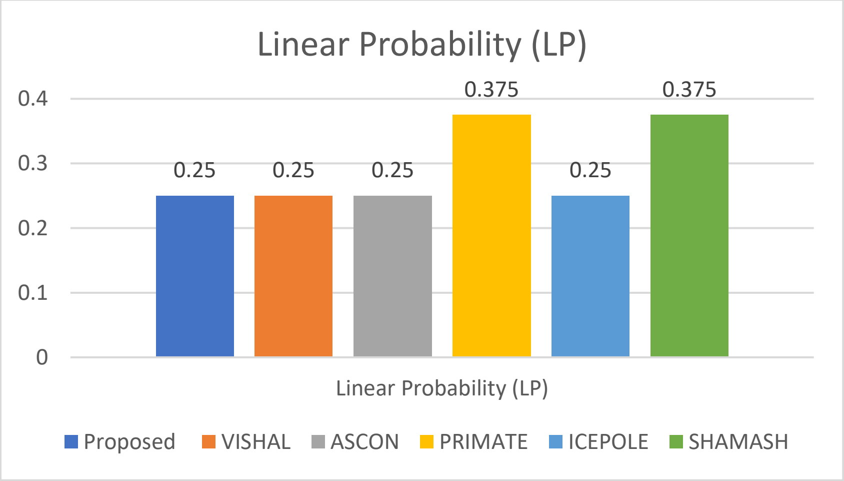


Figure 6 Linear Probability

For TentLogiX, the most frequent occurrence of the input differential x. Δx equaling the S-box's output differential S(x).Δy was recorded eight times in table 12. This yields a maximum LP of 0.25. An LP value approaching zero is indicative of a more secure S-box, as it suggests a lower predictability of linear relationships. As illustrated in Figure 6 security level of TentLogiX, in terms of linearity, is on par with or superior to existing 5-bit S-boxes.

Further enhancements to the LP metric could involve refining the design of the S-box to reduce the LP value, thereby increasing resistance to linear cryptanalysis. This can be achieved by integrating more complex transformations that distribute input-output correlations more evenly, ensuring that the LP value is minimized, and the S-box's security is maximized.

4.4. High resistance to Differential Cryptanalysis (DAP)

Differential Cryptanalysis is an analytical technique aimed at deciphering the reliability of S-boxes by examining their Differential Distribution Table (DDT). This method assesses the variation in the output of an S-box when there are slight changes in the input, which is crucial for safeguarding against Differential Cryptanalysis. The pivotal measure in this context is the Differential Approximation Probability (DAP), which quantifies the S-box's differential uniformity of input-output mappings. It can be mathematically expressed as:

$$DAP = max_{\Delta x \neq 0, \Delta y} \left(\frac{\#\{x \in X | S(x + (S(x + \Delta x) = \Delta y)\}}{2^n}\right)$$

where X denotes the complete set of potential input values, and n represents the number of input bits. The DAP represents the highest likelihood of a specific output differential Δy given an input differential Δx. For each input x, with possible differentials ranging from 0 to 31, TentLogiX has a recorded DAP of 0.25, equivalent to $8/2^5$ as shown in Table 13.

Figure 7 illustrates the DAP for different 5-bit s-boxes used for comparison, this S-box's DAP suggests a commendable level of security when compared to other 5-bit S-boxes.

An ideal S-box would have a DAP of $1/2^n$, which implies an extremely low probability of differential correlation that would be virtually impossible to exploit for cryptanalysis. Hence, the smaller the DAP value, the stronger the nonlinearity of the S-box, enhancing its resistance to differential attacks. The 5-bit S-box TentLogiX here demonstrates robust defense mechanisms, maintaining high resistance to differential cryptanalysis even with its limited size.

Improving upon this measure might include developing an S-box with an even more uniform distribution over the DDT, further reducing the maximum DAP and thereby strengthening the cryptographic system's resilience against differential attacks. This could potentially be achieved by introducing more sophisticated algorithms for generating S-boxes or by employing more intricate mathematical functions to define the S-box mappings.

Table 13 TentLogiX DAP

	Δout	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Δin	0	32	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Δin	1	-	-	-	-	2	-	-	4	2	-	-	4	-	2	-	2	-	-	-	2	-	2	4	2	2	-	2	-	-	2	-	-
Δin	2	-	-	-	2	2	4	2	2	-	2	-	2	2	2	-	-	4	2	-	-	-	2	2	2	-	-	-	-	-	-	-	-
Δin	3	-	2	2	-	-	-	2	4	2	-	2	-	-	-	-	2	2	2	-	-	-	-	2	-	2	-	-	2	2	4	-	-
Δin	4	-	2	2	2	-	-	-	2	-	-	2	-	-	-	4	2	4	2	-	2	-	2	-	2	-	2	-	2	-	-	-	-
Δin	5	-	-	4	-	-	-	2	-	2	-	2	-	2	-	-	-	-	-	-	-	4	-	-	2	2	-	4	2	2	4	-	-
Δin	6	-	-	2	-	2	2	2	2	-	-	-	2	-	-	-	-	-	-	4	2	-	-	4	-	-	2	2	2	-	2	-	2
Δin	7	-	-	-	-	-	2	-	-	2	2	-	4	2	2	-	2	-	6	4	2	-	-	2	-	-	-	-	-	-	-	2	-
Δin	8	-	-	-	-	-	4	4	-	2	-	-	2	2	-	-	2	2	-	2	-	-	-	-	-	2	2	4	-	2	-	-	2
Δin	9	-	-	-	-	-	-	2	2	4	2	-	2	-	-	2	2	2	4	2	4	-	4	-	-	-	-	-	-	-	-	-	-
Δin	10	-	2	-	4	-	-	-	-	-	4	-	-	4	2	-	4	-	-	-	-	2	-	2	2	-	-	-	2	-	2	-	2

Δin	11	-	4	2	2	6	-	-	2	2	-	-	-	-	2	-	-	4	-	-	-	-	-	2	2	-	2	-	-	-	-	-	2
Δin	12	-	2	2	-	2	-	-	-	-	4	2	2	-	-	-	2	-	-	2	2	2	4	-	-	-	2	2	-	-	-	-	2
Δin	13	-	-	2	-	4	-	2	-	-	-	-	-	-	-	-	-	2	2	6	-	2	-	-	-	2	-	-	2	-	-	4	4
Δin	14	-	-	4	4	2	-	-	2	-	-	-	-	-	2	-	2	-	-	-	-	-	-	-	-	2	4	2	-	2	-	4	2
Δin	15	-	-	-	2	-	-	-	-	-	2	-	2	-	-	2	-	-	2	-	2	6	2	2	4	-	2	-	-	-	2	2	-
Δin	16	-	-	2	4	-	4	2	2	-	2	2	2	-	-	-	-	-	2	-	2	2	-	-	-	2	-	-	2	-	-	2	-
Δin	17	-	-	-	-	-	-	-	2	2	2	2	-	-	-	4	-	2	-	-	4	-	4	-	-	-	2	-	2	2	2	-	2
Δin	18	-	4	2	-	-	-	2	-	2	-	-	-	2	4	-	4	-	2	6	-	-	2	2	-	-	-	-	-	-	-	-	-
Δin	19	-	-	-	-	2	6	-	-	-	-	2	-	-	-	-	2	-	-	-	-	-	2	2	-	-	4	4	2	2	-	4	-
Δin	20	-	-	-	-	2	-	-	-	6	-	-	-	-	4	-	-	-	-	2	4	4	2	-	2	-	2	-	2	-	-	2	-
Δin	21	-	-	-	4	2	-	-	2	-	2	-	-	-	-	2	-	-	-	-	2	-	-	-	2	-	2	-	2	4	2	4	2
Δin	22	-	4	2	4	-	-	-	-	2	-	-	-	-	2	2	-	2	-	-	-	2	-	2	-	6	-	-	-	2	-	-	2
Δin	23	-	-	-	-	-	2	-	-	-	2	4	2	-	-	4	2	2	-	-	-	-	4	-	-	2	2	-	2	2	-	2	-
Δin	24	-	4	-	-	-	-	2	-	-	-	2	-	4	-	4	-	-	-	-	-	-	-	-	2	-	2	2	2	2	4	-	2
Δin	25	-	2	2	2	-	2	2	-	-	2	-	2	4	2	-	-	-	-	2	2	-	-	4	2	-	-	2	-	-	-	-	-

Δin	26	-	-	2	-	-	-	-	-	-	2	8	2	2	2	-	2	-	-	-	-	4	-	-	2	-	-	2	-	-	2	2	-
Δin	27	-	2	-	2	4	-	2	2	-	-	2	-	-	-	2	-	-	-	-	2	-	-	-	2	-	-	4	-	6	-	-	2
Δin	28	-	-	2	-	-	-	-	-	2	2	-	2	2	4	4	2	-	4	-	-	-	-	-	2	2	-	-	2	-	-	-	2
Δin	29	-	2	-	-	-	2	4	2	-	-	-	-	2	-	-	-	-	2	-	-	2	2	-	-	4	2	-	2	4	2	-	-
Δin	30	-	2	-	-	2	2	2	2	2	-	-	2	2	2	2	-	-	-	-	-	-	-	-	2	2	-	2	2	-	2	2	-
Δin	31	-	-	-	-	-	2	-	-	-	2	2	-	2	-	-	-	6	2	2	-	2	-	2	-	2	-	-	-	-	2	2	4

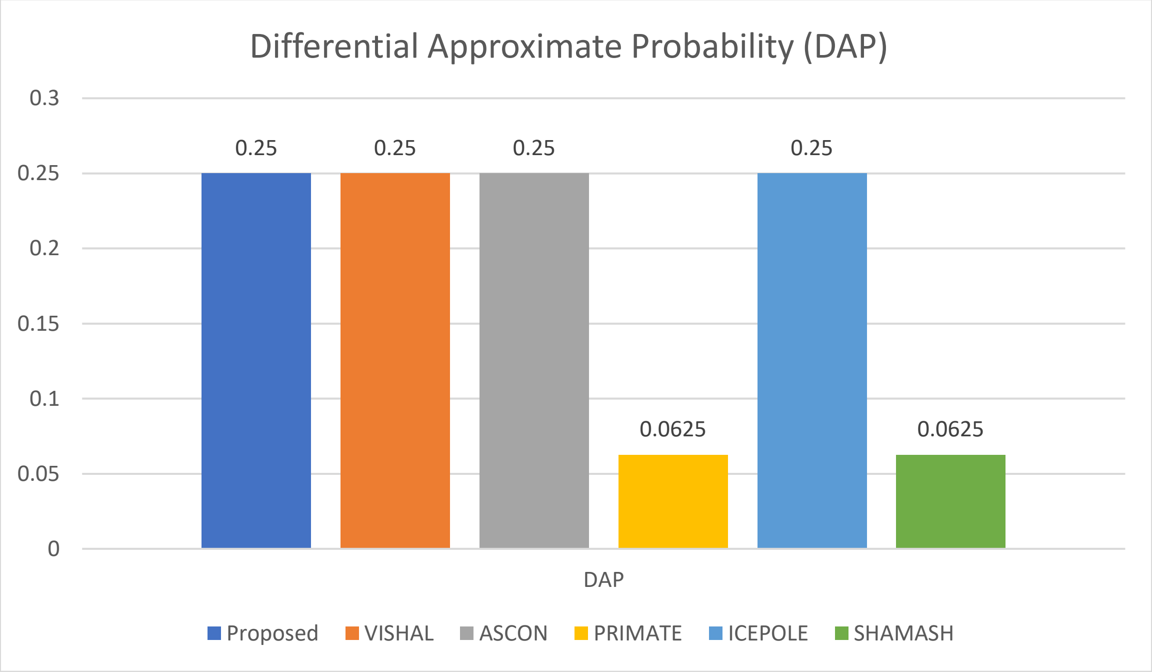


Figure 7 Differential Approximate Probability

4.5. High degree of avalanche effect

The avalanche effect refers to the phenomenon where a slight change in the input, such as flipping a single bit, causes a significant and widespread change in the output. Achieving a high degree of avalanche effect is crucial in cryptographic systems to ensure unpredictability and security. An S-box that produces a strong avalanche effect contributes to the overall robustness of the cipher by making it more resistant to various types of cryptanalytic attacks, including differential cryptanalysis. An S-box designed to meet the strict avalanche criterion (SAC) ensures that, for any n bit input, at least $n/2$ output bits will change when one input bit is modified. This property is essential for distributing the effects of small input changes across the entire output, thereby enhancing the diffusion and making the cipher significantly harder to break.

As introduced by Webster and Tavares [32], the SAC property of an S-box can be confirmed by considering a 5-bit input X and a set of input vectors, $\mathrm{X}_1,\ \mathrm{X}_2, \mathrm{X}_{3,}\, \mathrm{X}_{4_{,}}\, \mathrm{X}_5$ derived by altering only the jth bit of X. The corresponding 5-bit output vectors, $\mathrm{Y}_1, \mathrm{Y}_2, \mathrm{Y}_{3,}\ \mathrm{Y}_{4_{,}}\, \mathrm{Y}_5$, can be computed using a substitution function, where $Y_j = S(X_j)$ An avalanche vector, $v_1,\ v_2, v_{3,}\ v_{4_{,}}\, v_5$ is then calculated by XORing the original output vector Y with each Y_j, i.e V_j=Y$\oplus$ Y_j.

To further analyze, a 5×5 dependency matrix A is generated by adding the ith bit of Vj to $a_{i,j}$, where ai,j is the ith element of matrix A. These steps are repeated multiple times for different input vectors X, and each element of matrix A is then divided by 2^n (where n is the number of input/output bits) to compute the SAC matrix.

The average avalanche effect of TentLogiX is 0.50 (50.00%) (Table 15). Which made the TentLogiX S-box satisfy the SAC property. Thus, the TentLogiX S-box meets the SAC requirement. Figure 8 compares the SAC value of the TentLogiX S-box with other existing 5-bit S-boxes, showing that the average SAC value of the TentLogiX S-box is closest to the ideal value (0.5), making it superior to the alternatives.

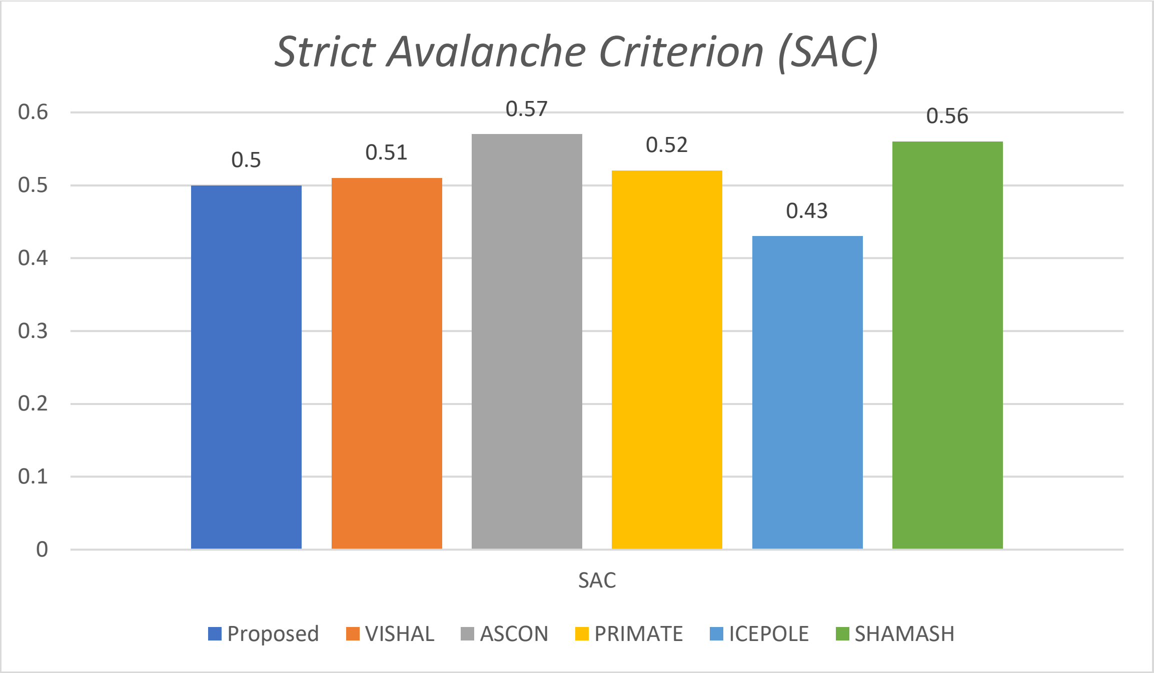


Figure 8 Strict Avalanche Criterion (SAC)

4.6. Bit Independence Criterion

The Bit Independence Criterion (BIC) is an important property of cryptographic functions, particularly S-boxes, ensuring that the output bits are statistically independent of each other when individual input bits are changed. For the TentLogiX 5x5 S-box, we evaluate the BIC by examining the correlation between pairs of output bits when a single input bit is toggled.

To meet the BIC, the TentLogiX S-box ensures that:

- Changing one bit in the input affects approximately half of the output bits.
- The influence on each output bit is independent of the others, minimizing correlation between output bits and preventing patterns that could be exploited by differential cryptanalysis.

The independence of output bits was tested through statistical measures, and it was found to be 0.51, confirming that the TentLogiX S-box exhibits low correlation between output bit pairs, thus adhering to the Bit Independence Criterion and contributing to its robustness against cryptanalytic attacks. Figure 9 provides a comparative analysis of the Bit Independence Criterion (BIC) for multiple 5-bit S-boxes.

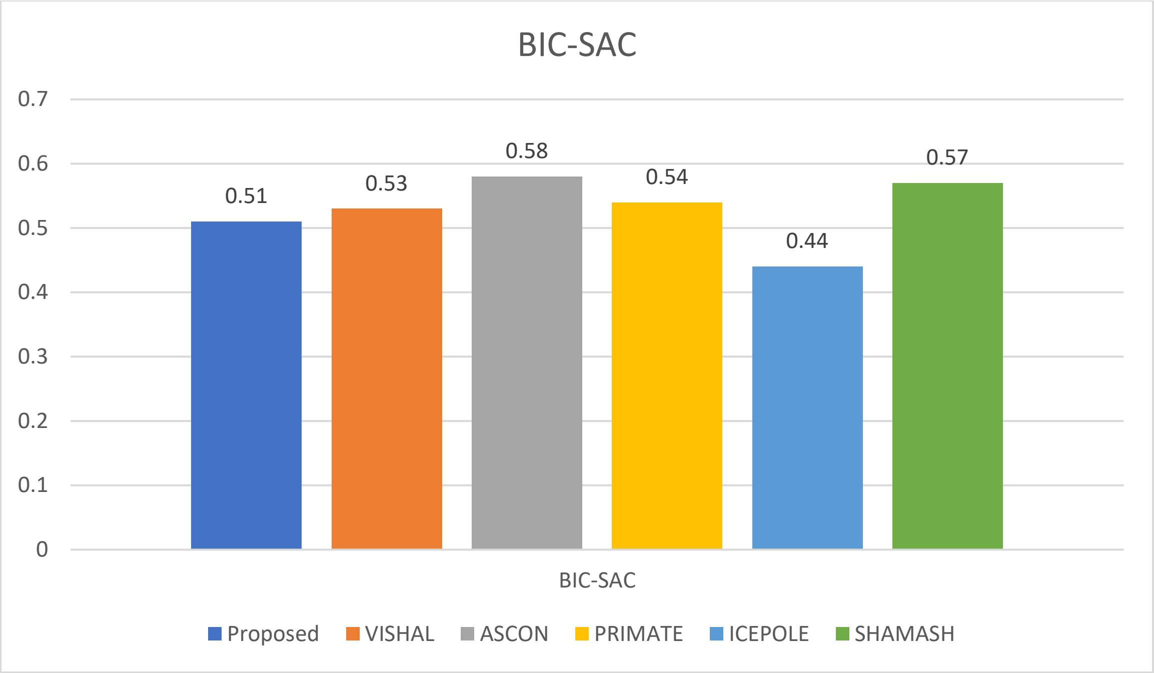


Figure 9 Strict Avalanche Criterion (SAC)

5. Conclusion

In conclusion, this paper presents a novel 5-bit S-box design, TentLogiX, based on a compound chaotic system combining logistic and tent maps. This innovative design addresses the limitations of existing chaotic maps by expanding the chaotic range and improving uniformity in output distribution. The TentLogiX S-box demonstrates superior security attributes, as evidenced by its performance in cryptanalysis. With strong resistance to differential cryptanalysis, enhanced nonlinearity, and low linear approximation probability, the TentLogiX S-box achieves a notable balance between cryptographic security and efficiency for lightweight cryptography systems. It outperforms several 5-bit counterparts in key security metrics, making it highly suitable for resource-constrained IoT environments. Future research may focus on further optimizing the chaotic map and exploring its integration into broader cryptographic frameworks.

References

[1] K. Shibutani, I. Takanori , H. Harunaga and M. Atsushi , "Piccolo: An Ultra-Lightweight Blockcipher," *International Conference on Cryptographic Hardware and Embedded Systems,* no. 13, pp. 342-357, Sep 2011.

[2] M. J. Dworkin, E. Barker, J. R. Nechvatal and J. F. Law, "Advanced Encryption Standard (AES)," 2001.

[3] L. De Meyer and S. Vaudenay, DES S-box generator. Cryptologia, vol. 41, 2017, pp. 153-171.

[4] W. Zhang, Z. Bao, V. Rijmen and M. Liu, A New Classification of 4-bit Optimal S-boxes and its Application to PRESENT, RECTANGLE and SPONGENT, vol. 9054, Springer, 2015, p. 494–515.

[5] A. Bogdanov and et al., An Ultra-Lightweight Block Cipher, Berlin, Heidelberg: Springer, 2007, p. 450–466.

[6] V. A. Thakor, M. A. Razzaque, A. D. Darji and A. R. Patel, A novel 5-bit S-box design for lightweight cryptography algorithms, vol. 73, Elsevier, 2023.

[7] L. Knudsen, G. Leander , A. Poschmann and . M. Robshaw, PRINTcipher: A block cipher for IC-printing, Springer, 2010, p. 16–32.

[8] w. Zhang,, Z. Bao, D. Lin and et al. , RECTANGLE: A Bit-slice Lightweight Block Cipher Suitable for Multiple Platforms, vol. 58, Sci. China Inf. Sci, 2015, p. 1–15.

[9] H. Yap, K. Khoo, A. Poschmann and M. Henricksen, EPCBC - A Block Cipher Suitable for Electronic Product Code Encryption, vol. vol, Springer, Berlin, Heidelberg, 2011.

[10] T. Suzaki, K. Minematsu, S. Morioka and E. Kobayash, TWINE : A Lightweight , Versatile Block Cipher., 2011.

[11] J. Guo, T. Peyrin, A. Poschmann and M. Robshaw, The LED Block Cipher, vol. 6917, Springer, Berlin, Heidelberg, 2011.

[12] G. Beierle, J. Jean and S. Kölbl, The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS, vol. 9815, Springer, Berlin, Heidelberg, 2016, p. 123–153.

[13] K. Shibutani, T. Isobe and H. Hiwatari, Piccolo: An Ultra-Lightweight Blockcipher, vol. 6917, Springer, Berlin, Heidelberg, 2011.

[14] G. Leander, C. Paar, A. Poschmann and K. Schramm, "New Lightweight DES Variants," in *Lecture Notes in Computer Science*, 2007.

[15] C. Dobraunig, M. Eichlseder, F. Mendel and M. Schläffer , Ascon v1.1, Inst Appl Inf Proc Commun Graz, 2016.

[16] E. Andreeva, B. Bilgin , A. Bogdanov, A. Luykx, F. Mendel and B. Mennink, PRIMATEs v1, CAESAR Competition, 2014.

[17] P. Morawiecki, K. Gaj , E. Homsirikamol, K. Matusiewicz, J. Pieprzyk, M. Rogawski and a. et, ICEPOLE: High-Speed, Hardware-Oriented Authenticated Encryption, Berlin, Heidelberg: Springer, 2014.

[18] D. Penazzi and S. Montes M., "Shamash (and shamashash)(version 1). Lightweight Crypography standared," 2019.

[19] Z. Gong, S. Nikova and Y. Law, "KLEIN: A New Family of Lightweight Block Ciphers," *Springer*, 2012.

[20] H. Cheng, H. M. Heys and C. Wang, "PUFFIN: A Novel Compact Block Cipher Targeted to Embedded Digital Systems," in *11th EUROMICRO conference on digital system design architectures, methods and tools*, 2008.

[21] W. Wu and L. Zhang, "LBlock: A Lightweight Block Cipher," in *Applied Cryptography and Network Security.*, Berlin, Heidelberg, 2011.

[22] A. Bogdanov, M. Knežević, G. Leander and D. Toz, "Spongent: A Lightweight Hash Function," in *Preneel, B., Takagi, T. (Eds.), Cryptographic Hardware and Embedded Systems – CHES 2011*, Berlin, Heidelberg, 2011.

[23] F. Standaert, G. Piret, N. Gershenfeld and J. Quisquater, "SEA: A Scalable Encryption Algorithm for Small Embedded Applications," *SEA: A Scalable Encryption Algorithm for Small Embedded Applications,* vol. 3928, 2006.

[24] G. Leander, C. Paar, A. Poschmann and K. Schramm, New Lightweight DES Variants, Berlin, Heidelberg: Springer, 2007, p. 196–210.

[25] F. Standaert, G. Piret, G. Rouvroy, J. Quisquater and J. Legat, ICEBERG : An Involutional Cipher Efficient for Block Encryption in Reconfigurable Hardware, vol. 3017, Berlin, Heidelberg: Springer, 2004, p. 279–98.

[26] R. May, "mathematical models with very complicated dynamics," *Nature,* vol. 261, p. 459–467 , 1976.

[27] J. Ahmad and H. S.O., "Chaos-based diffusion for highly autocorrelated data in encryption algorithms," *Nonlinear Dynamic,* vol. 82, pp. 1839-1850, July 2015.

[28] Q. Lu, C. Zhu and G. Wang, "A Novel S-Box Design Algorithm Based on a New Compound Chaotic System," *entropy,* vol. 21, 2019.

[29] V. A. Thakor, M. A. Razzaque, . A. D. Darji and R. Aksh , "A novel 5-bit S-box design for lightweight cryptography algorithms," *Journal of Information Security and Applications,* vol. 73, 2023.

[30] C. E. M. M. F. e. a. Dobraunig, "ASCON v1.2: Lightweight Authenticated Encryption and Hashing," *Journal of Cryptology,* vol. 34, no. 33, 2021.

[31] K. Jeong, Y. Lee,, J. Sung and S. Hong, Improved differential fault analysis on PRESENT-80/128., vol. 90, International Journal of Computer Mathematics, 2013, p. 2553–2563.

[32] O. Toshihiko, Lightweight Cryptography Applicable to Various IoT Devices, 2017.

[33] F. Standaert, G. Piret, N. Gershenfeld and J. Quisquater, "SEA: A Scalable Encryption Algorithm for Small Embedded Applications," in *Smart Card Research and Advanced Applications*, 2006.

[34] D. Penazzi and M. M. Shamash, "Shamash (and Shamashash) (version 1). Lightweight Cryptogr Standard Proc Round," 2019.