New Integral Distinguishers and Security Reassessment of LTLBC

Abhilash Kumar Das
Indian Institute of Technology Jammu, India

Keywords: Integral Cryptanalysis, Division Property, Lightweight Block Cipher, MILP, LTLBC, Cryptographic Security.

Abstract: This paper presents an in-depth study of integral distinguishers for the LTLBC block cipher, a 14-round 64-bit lightweight cryptographic scheme designed for low-latency applications in IoT environments. Leveraging the division property technique introduced by Yosuke Todo, we employ a Mixed Integer Linear Programming (MILP) approach to identify previously unpublished 6-round integral distinguishers for LTLBC. Additionally, we studied the `MixWord` permutation phase and showed that the cyclic intermixing tweak to the input word doesn't yield any significant improvement. Instead, it reduces to the original `MixWord` operation. This observation is rigorously justified through an algebraic proof and further followed by linear and differential cryptanalysis, leading to a revised active Sbox count for LTLBC. As a side contribution, we correct inaccuracies in the reported division property propagations for the FUTURE Sbox, initially presented at AFRICACRYPT 2022. Our findings provide a deeper understanding of LTLBC's security and offer valuable insights for the design of future lightweight block ciphers.

1 INTRODUCTION

Integral cryptanalysis, originally introduced as the Square attack by Daemen et al., has evolved into a powerful method for analyzing symmetric-key primitives through integral distinguishers (Todo, 2015). The division property, introduced by Todo (Todo, 2015), enhanced this technique by modeling the propagation of algebraic structures through cryptographic operations. Despite its strength, early applications faced computational limitations, particularly with large block ciphers.

To overcome these challenges, MILP-aided techniques were adopted. Xiang et al. (Xiang et al., 2016) demonstrated how MILP can automate the search for integral distinguishers, significantly improving efficiency. Later, Sun et al. (Sun et al., 2020) refined this for bit-level analysis, extending its use to ciphers with complex linear layers. Xu et al. (Xu et al., 2024) recently emphasized the need for accurate modeling in division property propagation, revealing flaws in prior analyses of the FUTURE cipher.

In the context of lightweight cryptography, especially for IoT, integral cryptanalysis remains a key evaluation tool. Mirzaie et al. (Mirzaie et al., 2023) exposed vulnerabilities in Shadow-32 using MILP-based analysis. Meanwhile, LTLBC (Sun et al., 2024), a cipher with low-latency IoT applications, has not yet been rigorously analyzed under integral cryptanalysis. Our study addresses this gap, uncovering potential weaknesses and proposing security enhancements to better protect IoT-based cloud systems (Yalli et al., 2025; Sasikumar and Nagarajan, 2024).

In this paper, we outline our key contributions.

1. We discovered new 6-round integral distinguishers for LTLBC using MILP-based division property analysis, giving details on best possible distinguishers.
2. We evaluate a structural property of `MixWord` architecture, showing a reduction of a cyclic word intermixing tweak to the original `MixWord` operation.
3. We revised some corrections in the proposed active Sboxes count of the original cipher via linear and differential cryptanalysis.
4. Correction of division property propagation for the FUTURE Sbox published in AFRICACRYPT 2022.

2 PRELIMINARIES

This section provides an overview of the key cryptographic concepts and techniques used in this paper, including the division property, MILP-based cryptanalysis, and integral distinguishers. These concepts

Das, A. K.
New Integral Distinguishers and Security Reassessment of LTLBC.
DOI: 10.5220/0013566600003979
In *Proceedings of the 22nd International Conference on Security and Cryptography (SECRYPT 2025)*, pages 717-722
ISBN: 978-989-758-760-3; ISSN: 2184-7711

form the foundation for our analysis of LTLBC.

2.1 Division Property

The division property, introduced by Todo (Todo, 2015), is a generalized integral property that facilitates integral cryptanalysis by exploiting the algebraic structure of cipher components. It extends traditional integral properties by leveraging propagation rules for basic operations such as XOR, AND, and n-bit Sboxes. The division property offers a systematic framework for analyzing the propagation of integral properties across cryptographic functions, particularly Sboxes and linear transformations. Its primary objective is to characterize how the division property evolves through various cipher components.

2.1.1 Definitions

Definition 2.1 ($\boldsymbol{u} \succcurlyeq \boldsymbol{v}$,(Xiang et al., 2016))**.** *We say $\boldsymbol{u} \succcurlyeq \boldsymbol{v}$ if and only if $\forall i, u_i \geq v_i$. Otherwise, $\boldsymbol{u} \not\succcurlyeq \boldsymbol{v}$ i.e. at least one element in $\boldsymbol{u}$ is strictly smaller than $\boldsymbol{v}$ coordinate wise.*

Definition 2.2 (Division Property (Todo, 2015))**.** *Let $\mathbb{X}$ be a multiset whose elements belong to $\mathbb{F}_2^n$. The multiset $\mathbb{X}$ is said to have the division property $\mathcal{D}_{\boldsymbol{k}}^n$ if, for all $\boldsymbol{u} \in \mathbb{F}_2^n$ satisfying $\boldsymbol{u} \not\succcurlyeq \boldsymbol{k}$, the parity of the sum*

$$\bigoplus_{\boldsymbol{x} \in \mathbb{X}} \left(\prod_{i=0}^{n-1} x_i^{u_i} \right)$$

over all $\boldsymbol{x} \in \mathbb{X}$ is always even. Otherwise, the parity of the sum may or may not be even. Hence, we call it unknown.

The $\boldsymbol{k}$ has the same dimension as that of $\boldsymbol{u}$. The notation of the division property depends on the instance element, i.e. $\mathcal{D}_3^4$ can be thought of as $\mathcal{D}_{\mathbb{K}}^{1,4}$ with $\boldsymbol{k}_i \in \mathbb{K}$ where $\boldsymbol{k}_i$ can take one or more vectors from

$$\mathbb{K} = \{(0,1,1,1),(1,0,1,1),(1,1,0,1),(1,1,1,0)\}$$

However, they are not the same because of the different representations. The former is the finite field representation, and the latter is the vectorial representation. It is easy to notice that the notation $\mathcal{D}_{\mathbb{K}}^{n,m}$ is justified using $n = 1$, $m = 4$ with the element take value of $\mathbb{F}_2^{n,m}$. We give a suitable example to make a distinction. Note that we define the hamming weight $hw(\cdot)$ as the number of non-zero elements in a vector v e.g. let $v = (1,0,1,0)$ then $hw(v) = 2$.

Example 2.1. *Consider a multiset $\mathbb{X}$ of 4-bit elements:*

$$\mathbb{X} = \mathtt{0x}\{\mathtt{9,\ 6,\ 7,\ 4,\ 4,\ a,\ 8,\ 4,\ 5,\ b}\}.$$

We analyze the division property by evaluating the parity of the sum for all possible values of $\boldsymbol{u}$. Notably, the parity remains zero whenever hamming weight $\mathtt{hw}(\boldsymbol{u}) < 3$, indicating that the division property of $\mathbb{X}$ is $\mathcal{D}_3^4$.

However, we identify two specific vectors, $\boldsymbol{k}_0$ and $\boldsymbol{k}_1$, given by $\{(0111),(1011)\}$, that yield a parity of 1. This implies the existence of a minimal set $\mathbb{K} = \{\boldsymbol{k}_0, \boldsymbol{k}_1\}$, where for any $\boldsymbol{u} \not\succcurlyeq \boldsymbol{k}_0, \boldsymbol{k}_1$, the parity over the multiset remains zero. To illustrate, consider $\boldsymbol{u} = (1011)$. It is evident that $\boldsymbol{u} \not\succcurlyeq \boldsymbol{k}_0$ but $\boldsymbol{u} \succcurlyeq \boldsymbol{k}_1$, which confirms its inclusion in $\mathbb{K}$. If $\boldsymbol{u}$ were not included, we would observe a case where $\boldsymbol{u} \not\succcurlyeq \boldsymbol{k}_0$ while still producing a parity of 1 over $\mathbb{X}$, contradicting the case. This analysis establishes a clear criterion for defining the division property of the given multiset.

Now, we explain the effect of division property in the output multiset $\mathbb{Y}$ when passed through non-linear transformation such as the Sbox S of lower algebraic degree. According to the proposition given by Yosuke Todo (Todo, 2015) for an Sbox with algebraic degree d,

Proposition 2.1 (Propagation characteristic of division property)**.** *If an input multiset $\mathbb{X}$ has the division property $\mathcal{D}_k^n$, then the output multiset $\mathbb{Y}$ satisfies the division property*

$$\mathcal{D}_k^n \xrightarrow{S} \mathcal{D}_{\lceil k/d \rceil}^n.$$

Furthermore, if S is a bijective Sbox, then the preservation of the maximum degree implies

$$\mathcal{D}_n^n \xrightarrow{S} \mathcal{D}_n^n.$$

So, the division property is preserved for the linear layer since the algebraic degree is 1. It is clear that the division property $\mathcal{D}_0^4$ would be invalid if $|\mathbb{X}|$ is odd.

3 SPECIFICATION OF LTLBC

LTLBC is a lightweight block cipher designed for low-latency applications, particularly in IoT environments. It follows a 14-round Substitution-Permutation Network (SPN) structure, operating on a 64-bit plaintext and using a 128-bit master key.

3.1 Notations

The notations used in LTLBC can be referred from Table 1.

3.2 Encryption Algorithm

The encryption process of LTLBC consists of an initial key addition followed by 14 rounds of transformations. Each round includes `PermuteBits` (Pb),

Table 1: Notations used in LTLBC.

Notation	Description
P	64-bit plaintext
K	128-bit master key
C	64-bit ciphertext
$T^{(i)}$	Intermediate state after ith round
R_i	ith round number out of 14
RK_i	64-bit subkey for round i
RC_i	64-bit round constant for round i
$X[i]$	ith bit of X
$X[L:H]$	$(H-L)$-bit slice of X taken from Lth bit to $(H-1)$th bit
$X\|\|Y$	concatenation of X and Y
$Op(X\|\|Y)$	$Op(X)\|\|Op(Y)$ if a valid operation
$X^{\lll t}$	t left cyclic shifts of X

`MixWord` (Mw), `SubCell` (Sb), `AddRoundKey` (Ak), and `AddRoundCon` (Ac).

The final round excludes the `MixWord` operation, and the resulting state $T^{(14)}$ is output as the ciphertext C. The round function $f_R^{(i)}$ of ith round can be expressed as (also shown in Figure 1)

$$T^{(r)} = \begin{cases} Ac \circ Ak \circ Sb \circ Pb(X), & \text{if } r = 14, \\ Ac \circ Ak \circ Sb \circ Mw \circ Pb(X), & \text{else.} \end{cases}$$

Finally, we express ciphertext C as

$$C = f_R^{(14)} \bigcirc_{i=1}^{13} f_R^{(i)} \circ (X \circ K_0)$$

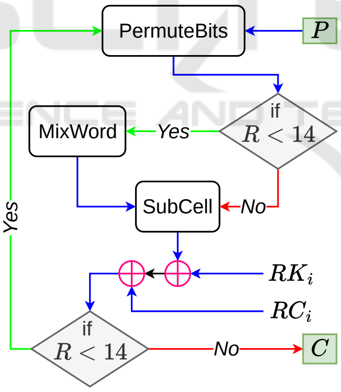

Figure 1: Structure of the LTLBC block cipher.

This section provides a complete overview of LTLBC, including its round function and key schedule, forming the foundation for further cryptanalysis and security evaluation. The reader can refer to (Sun et al., 2024) for more details.

4 SEARCHING THE INTEGRAL DISTINGUISHERS

First, we provide all the known tools required to model the cipher and then apply the searching technique to the LTLBC cipher.

4.1 MILP-Based Tool and Basic Modelling Strategies

The LTLBC implements Sbox and XOR operation in the entire encryption. The following MILP-based bit-division property modelling can be used in our case. To efficiently search for integral distinguishers in block ciphers, the propagation of the division property through fundamental operations like XOR and Sboxes must be accurately modeled using MILP.

Modelling XOR (Xiang et al., 2016). The XOR operation is widely used in cryptographic structures, particularly in key addition and linear layers. When two input bits x_0, x_1 undergo XOR, the resulting output bit y satisfies the relation

$$y = x_0 \oplus x_1$$

If the input multiset $\mathbb{X} = \{(x_0, x_1) \in \mathbb{F}_2^2\}$ has a division property $\mathcal{D}^{1,2}_{(k_0,k_1)}$ with vector $\boldsymbol{k} = (k_0, k_1)$ and $k_0, k_1 \in \mathbb{F}_2$, the output multiset follows $\mathcal{D}^{1,1}_{(k_0+k_1)}$. There can be four division trails possible, namely,

$$\begin{aligned} &\mathcal{D}^{1,2}_{(0,0)} \to \mathcal{D}^{1,1}_{(0)} && \text{(valid)} \\ &\mathcal{D}^{1,2}_{(0,1)} \to \mathcal{D}^{1,1}_{(1)} && \text{(valid)} \\ &\mathcal{D}^{1,2}_{(1,0)} \to \mathcal{D}^{1,1}_{(1)} && \text{(valid)} \\ &\mathcal{D}^{1,2}_{(1,1)} \to \mathcal{D}^{1,1}_{(2)} && \text{(invalid)} \\ &\mathcal{D}^{1,2}_{(k_0,k_1)} \to \mathcal{D}^{1,2}_{k_0} \text{ or } \mathcal{D}^{1,2}_{k_1} && \text{(invalid)} \end{aligned}$$

We give two examples of invalid division trails.

Example 4.1. *Consider*

$$\mathbb{X} = \{(0,0),(0,1),(0,1),(1,1),(0,0),(1,1)\}$$

with $\mathcal{D}^{1,2}_{(1,1)}$ *and apply* XOR $y_i = (x_0 \oplus x_1)_i$ *for all* i *giving* $\mathbb{Y} = \{0,1,1,0,0,0\}$ *with* $\mathcal{D}^{1,1}_{2}$ *that is invalid.*

Example 4.2. *Consider*

$$\mathbb{X} = \{(0,0),(0,1),(0,1),(1,1),(1,1),(1,1)\}$$

with $\mathcal{D}^{1,2}_{(0,1)(1,0)}$ *and apply* XOR $y_i = (x_0 \oplus x_1)_i$ *for all* i *giving same* $\mathbb{Y} = \{0,1,1,0,0,0\}$ *with invalid* $\mathcal{D}^{1,1}_{2}$.

The linear layer always propagates the division property, preserving the hamming weight of the input division.

Modelling Sbox (Xiang et al., 2016). Due to the nonlinearity of Sbox, we propagate input unknowns to all the possible output unknowns. Suppose given the input division property of n-bit Sbox is $\mathcal{D}^{1,n}_{\boldsymbol{k}}$. This implies that the parity of the sum is unknown when $\boldsymbol{u} \succcurlyeq \boldsymbol{k}$. We propagate to the output division $\boldsymbol{v}$ if any $\boldsymbol{v} \in \{\boldsymbol{u} | \boldsymbol{u} \succcurlyeq \boldsymbol{k}\}$.

4.2 Integral Distinguishers

We report the best integral distinguishes in Table 2 with the least active bits in the input division property across 1- to 6-rounds, giving all 64 balanced bits. Additionally, we give a bound in the number of best distinguishers.

Table 2: Input division $\mathcal{D}_{\boldsymbol{k}}^{1,64}$ across different rounds.

round	input division $\boldsymbol{k}$	active	$\log_2$(bound)
1	`0x0000000000000003`	2	10.97
2	`0x00000000000003ff`	10	37.14
3	`0x0000000003ffffff`	26	51.58
4	`0x00001fffffffffff`	45	47.18
5	`0x0fffffffffffffff`	60	63.35
6	`0x7fffffffffffffff`	63	22.86

We investigated the presence of balanced bits in a half-round implementation of LTLBC but found none. This suggests that LTLBC retains resistance beyond half its rounds against integral cryptanalysis based on the division property.

5 TWEAK TO `MIXWORD` OPERATION

As discussed in Section 3.2, the `MixWord` layer takes 8-bytes input, out of which 4-bytes are given to a symmetrically looking circuit with 5 and 9 left cyclic shifts independently. However, We analyze the security impact of interlinking all eight bytes in the circuit at the input stage.

5.1 `MixWord` Modification

We modify the `MixWord` operation as under.

$$\begin{aligned}
\mathtt{tmp}_0 &\leftarrow (X[0:16] \oplus X[16:32])^{|<t|},\\
\mathtt{tmp}_1 &\leftarrow (X[16:32] \oplus X[32:48])^{|<t|},\\
\mathtt{tmp}_2 &\leftarrow (X[32:48] \oplus X[48:64])^{|<t|},\\
\mathtt{tmp}_3 &\leftarrow (X[48:64] \oplus X[0:16])^{|<t|},\\
Y[0:16] &\leftarrow X[0:16] \oplus \mathtt{tmp}_0,\\
Y[16:32] &\leftarrow X[16:32] \oplus \mathtt{tmp}_1,\\
Y[32:48] &\leftarrow X[32:48] \oplus \mathtt{tmp}_2,\\
Y[48:64] &\leftarrow X[48:64] \oplus \mathtt{tmp}_3.
\end{aligned} \tag{1}$$

We observe the above equations are not involutary, as proved in Lemma 5.1. Therefore, we propose a different set of equations for both the encryption and decryption, capturing the effect of all four words from input. Here, two bytes is equal to one word.

Lemma 5.1. *Consider a function* $f : (\mathbb{F}_2^{16})^4 \to (\mathbb{F}_2^{16})^4$ *and*

$$y_0, y_1, y_2, y_3 = f(x_0, x_1, x_2, x_3)$$

associated with the `MixWord` *layer with t left cyclic shift in the circuit, then*

$$\begin{aligned}
y_0 &= x_0 \oplus (x_0 \oplus x_1)^{|<t|}\\
y_1 &= x_1 \oplus (x_1 \oplus x_2)^{|<t|}\\
y_2 &= x_2 \oplus (x_2 \oplus x_3)^{|<t|}\\
y_3 &= x_3 \oplus (x_3 \oplus x_0)^{|<t|}
\end{aligned}$$

if and only if

$$\begin{aligned}
y'_0 &= x_0 \oplus (x_0 \oplus x_2)^{|<2t|}\\
y'_1 &= x_1 \oplus (x_1 \oplus x_3)^{|<2t|}\\
y'_2 &= x_2 \oplus (x_2 \oplus x_0)^{|<2t|}\\
y'_3 &= x_3 \oplus (x_3 \oplus x_1)^{|<2t|}
\end{aligned}$$

where $y'_0, y'_1, y'_2, y'_3 = f^2(x_0, x_1, x_2, x_3)$ *with twice the original left cyclic shifts.*

Proof. We use commutativity as a distribution rule $(a \oplus b)^{|<t|} = a^{|<t|} \oplus b^{|<t|}$ and rewrite the expression $y'_0, y'_1, y'_2, y'_3 = f \circ f(x_0, x_1, x_2, x_3)$. Now,

$$\begin{aligned}
y'_0 &= y_0 \oplus (y_0 \oplus y_1)^{|<t|}\\
&= x_0 \oplus (x_0 \oplus x_1)^{|<t|} \oplus \{x_0 \oplus (x_0 \oplus x_1)^{|<t|}\\
&\oplus x_1 \oplus (x_0 \oplus x_1)^{|<t|}\}^{|<t|}\\
&= x_0 \oplus x_0^{|<2t|} \oplus x_2^{|<2t|}\\
&= x_0 \oplus (x_0 \oplus x_2)^{|<2t|}
\end{aligned}$$

Similarly,

$$\begin{aligned}
y'_1 &= x_1 \oplus (x_1 \oplus x_3)^{|<2t|}\\
y'_2 &= x_2 \oplus (x_2 \oplus x_0)^{|<2t|}\\
y'_3 &= x_3 \oplus (x_3 \oplus x_1)^{|<2t|}
\end{aligned}$$

This completes the proof. □

The well-established set of equations 2 has an involutary property with less number of gate counts compared to the equation set 1.

$$\begin{aligned}
\mathtt{tmp}_0 &\leftarrow (X[0:16] \oplus X[32:48])^{|<2t|},\\
\mathtt{tmp}_1 &\leftarrow (X[16:32] \oplus X[48:64])^{|<2t|},\\
Y[0:16] &\leftarrow X[0:16] \oplus \mathtt{tmp}_0,\\
Y[16:32] &\leftarrow X[16:32] \oplus \mathtt{tmp}_1,\\
Y[32:48] &\leftarrow X[32:48] \oplus \mathtt{tmp}_0,\\
Y[48:64] &\leftarrow X[48:64] \oplus \mathtt{tmp}_1.
\end{aligned} \tag{2}$$

The modified cyclic word input reduces to the original `MixWord` operation, except with twice the previous left cyclic shifts t. This observation implies that the diffusion achieved by the `MixWord` transformation remains unchanged regardless of the permutation of word inputs. The same is experimented with using linear and differential cryptanalysis in Section 6.

6 LINEAR/DIFFERENTIAL CHARACTERISTICS IN TWO VERSIONS

We model the original LTLBC and compared it with the tweaked version. We keep every element of the model details as the original one and just replace the `MixWord` architecture with new tweaked shift-pairs $(11,5)$ instead of $(5,9)$ and input bytes

$$(x_0x_1x_2x_3) \xrightarrow{\text{modified to}} (x_0x_2x_1x_3).$$

We analyze the linear and differential characteristics of the cipher under the original and tweaked versions and present the number of active Sboxes for 1- to 9-rounds in Table 3. According to the security claim of LTLBC, resisting linear and differential cryptanalysis requires more than 31 active S-boxes. Here, $n_A(\cdot)$ and $n_A^*(\cdot)$ represent the proposed and the revised active Sbox count respectively. The argument l or d in $n_A(\cdot)$ represents linear or differential cryptanalysis. The $\boxed{\text{red}}$ marked figures in the table were corrected, which we believe to be inaccurate according to the MILP model for original `MixWord` architecture using the $(5,9)$ shifts. We use İlter and Selçuk's modelling strategy for `MixWord` linear masking. Unlike linear cryptanalysis, we adopt a straightforward modeling approach to map the input differential to the output differential directly.

Table 3: Active Sboxes in Linear and Differential Cryptanalysis for both $(5,9)$ and $(11,5)$ `MixWord` architecture.

shift (5,9)									
R	1	2	3	4	5	6	7	8	9
$n_A(l)$	1	4	7	13	16	$\boxed{21}$	⊥	⊥	⊥
$n_A^*(l)$	1	4	7	13	16	19	22	26	31
$n_A(d)$	1	4	7	13	$\boxed{16}$	$\boxed{22}$	⊥	⊥	⊥
$n_A^*(d)$	1	4	7	13	17	19	26	30	36
shift (11,5)									
R	1	2	3	4	5	6	7	8	9
$n_A^*(l)$	1	4	7	13	16	18	21	26	31
$n_A^*(d)$	1	4	7	13	17	21*	24	29	36

⊥ - not recorded, * - improved

We provide the best linear and differential trails returned by Gurobi and a trail diagram for 3-round linear trails in Figure 2 with `A`, `B`, and `C` denoting the input trail to the `PermuteBits`, `MixWord`, and `SubCell`. Moreover, we attach 6-round linear and differential trails showing the number of active Sboxes in the Table 4 with bolded hexadecimal digits and `A`$^+$ denoting the input trail to the 7th round. We compute the probability bias ε_A and propagation ratio p_A of all the active Sboxes n_A using the Linear Approximation Table (LAT) and Differential Distribution Table (DDT) The notion of MILP modelling for linear cryptanalysis can be found in (Das, 2024; İlter and Selçuk, 2022) and differential cryptanalysis in (Mouha et al., 2012).

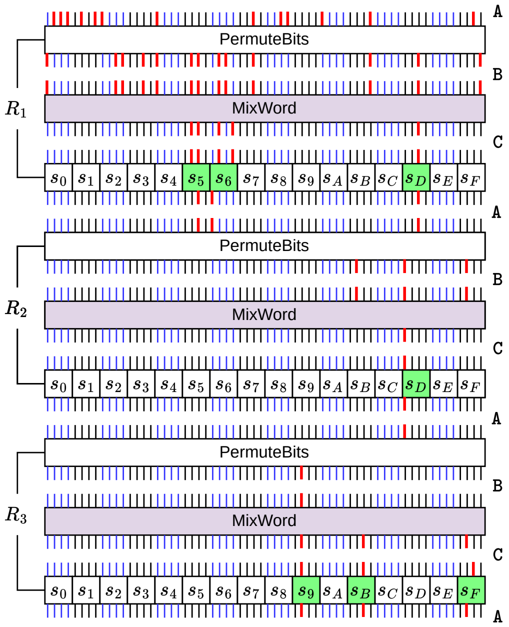
A
PermuteBits
B
R_1
MixWord
C
s_0 s_1 s_2 s_3 s_4 s_5 s_6 s_7 s_8 s_9 s_A s_B s_C s_D s_E s_F
A
PermuteBits
B
R_2
MixWord
C
s_0 s_1 s_2 s_3 s_4 s_5 s_6 s_7 s_8 s_9 s_A s_B s_C s_D s_E s_F
A
PermuteBits
B
R_3
MixWord
C
s_0 s_1 s_2 s_3 s_4 s_5 s_6 s_7 s_8 s_9 s_A s_B s_C s_D s_E s_F
A

Figure 2: 3-Round linear trail of LTLBC showing 7 active Sboxes with red input-output mask.

Table 4: Linear and differential trail for 6-round LTLBC.

R	I/O	Linear Trail	$\pm\varepsilon_A$	Differential Trail	p_A
1	`A`	0x7580800230810002	2^{-7}	0x0000002100000001	2^{-3}
	`B`	0x8032866200010201		0x0000000010001008	
	`C`	0x0000**65**000000**2**00		0x000000000000000**8**	
2	`A`	0x0000**28**000000**2**00	2^{-8}	0x000000000000000**1**	2^{-12}
	`B`	0x0000000000040804		0x0000000010000000	
	`C`	0x0000000000000**8**00		0x00000000**1**0**2**000**2**0	
3	`A`	0x0000000000000**8**00	2^{-12}	0x00000000**1**0**8**000**8**0	2^{-35}
	`B`	0x0000000004000000		0x8020000008000000	
	`C`	0x0000000000**4**0**2**000**2**		0x**843**00**41**00**81**000**1**0	
4	`A`	0x0000000000**4**0**2**000**4**	2^{-25}	0x**c22**00**82**00**a2**000**2**0	2^{-43}
	`B`	0x0401000000100000		0x400c220c00080408	
	`C`	0x0**c21**0**82**00**81**00**8**00		0x000**62**000000**4**00	
5	`A`	0x0**4c2**0**82**00**44**00**8**00	2^{-31}	0x000**14**000000**4**00	2^{-46}
	`B`	0x0c008c0104000408		0x0000000000402040	
	`C`	0x000**8**00**1**0000000**8**		0x000000000000**2**000	
6	`A`	0x000**1**00**4**0000000**4**	2^{-32}	0x000000000000**8**000	2^{-55}
	`B`	0x0000000000102010		0x0000000002000000	
	`C`	0x000000000000**2**000		0x0000000000**2**0**4**000**4**	
	`A`$^+$	0x000000000000**4**000		0x0000000000**2**0**8**000**4**	

*I/O represents the input-output trail.

7 FUTURE SBOX: REVISED PROPAGATION

We take the Algebraic Normal Form (ANF) form of FUTURE Sbox , displaying all the division propa-

Table 5: All possible propagations $\boldsymbol{v}$'s for every possible input division property $\boldsymbol{u}$ for FUTURE Sbox.

$\boldsymbol{u}$	all output propagations $\boldsymbol{v}$'s
0000	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
0001	1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
0010	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
0011	1, 3, $\boxed{5}$, 6, 7, 8, 9, 10, $\boxed{11}$, 12, 13, 14, 15
0100	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
0101	1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
0110	2, 3, 5, 6, 7, 9, 10, 11, 13, 14, 15
0111	7, 9, 10, $\boxed{11}$, 13, 14, 15
1000	1, 2, 3, 4, 5, 6, $\boxed{7}$, 8, 9, 10, 11, 12, 13, 14, 15
1001	1, 2, 3, 5, 6, $\boxed{7}$, 8, 9, 10, 11, 12, 13, $\boxed{14}$, 15
1010	1, 3, 4, $\boxed{5}$, 6, $\boxed{7}$, 8, 9, 10, 11, 12, 13, 14, 15
1011	1, 3, $\boxed{5}$, 6, $\boxed{7}$, 8, 9, 10, $\boxed{11}$, 12, $\boxed{13}$, $\boxed{14}$, 15
1100	2, 3, 5, 6, $\boxed{7}$, 10, 11, 12, 13, 14, 15
1101	2, 3, 5, 6, $\boxed{7}$, 10, 11, 13, $\boxed{14}$, 15
1110	10, 11, 13, 14, 15
1111	15

gations. We correct the inaccuracies found in the 'Security Analysis' section of the integral attack. If the propagations marked $\boxed{(.)}$ are removed, then the MILP-based search will struggle to find pathways to the unit vector, giving an unbalanced bit with less probability. Thereafter, the security against the integral attack would be in doubt. According to the current literature, the 6th and 7th rounds of FUTURE yield all 64-balanced bits (Xu et al., 2024).

8 DISCUSSION AND CONCLUSION

This study underscores the critical role of robust diffusion and strong S-box design in enhancing cipher security. Through detailed MILP-based analysis, we identified new 6-round integral distinguishers for LTLBC by modeling its S-box and `MixWord` layer, and refined active S-box estimates for linear and differential attacks. While LTLBC shows resilience against integral attacks up to 6 rounds, it remains vulnerable to other attacks by the 8th round. These insights are valuable for strengthening lightweight encryption in resource-constrained environments like IoT and distributed systems.

Future Research. We will analyze LTLBC's resistance against quantum cryptanalysis, particularly under Grover's search and algebraic attacks and consider hybrid approaches integrating LTLBC with post-quantum cryptographic schemes.

REFERENCES

Das, A. (2024). Bit-Based MILP Modelling of Non-Bit-Permutation Linear Layers for Linear Cryptanalysis. In *2024 19th Asia Joint Conference on Information Security (AsiaJCIS)*, pages 1–8. IEEE.

İlter, M. B. and Selçuk, A. A. (2022). Milp-aided cryptanalysis of the future block cipher. In *International Conference on Information Technology and Communications Security*, pages 153–167. Springer.

Mirzaie, A., Ahmadi, S., and Aref, M. R. (2023). Integral cryptanalysis of round-reduced shadow-32 for IoT nodes. *IEEE Internet of Things Journal*, 11(6):10592–10599.

Mouha, N., Wang, Q., Gu, D., and Preneel, B. (2012). Differential and linear cryptanalysis using mixed-integer linear programming. In *Information Security and Cryptology: 7th International Conference, Inscrypt 2011, Beijing, China, November 30–December 3, 2011. Revised Selected Papers 7*, pages 57–76. Springer.

Sasikumar, K. and Nagarajan, S. (2024). Comprehensive review and analysis of cryptography techniques in cloud computing. *IEEE Access*.

Sun, L., Wang, W., and Wang, M. Q. (2020). MILP-aided bit-based division property for primitives with non-bit-permutation linear layers. *IET Information Security*, 14(1):12–20.

Sun, W., Li, L., and Huang, X. (2024). LTLBC: a low-latency lightweight block cipher for internet of things. *Cluster Computing*, 27(7):9783–9794.

Todo, Y. (2015). Structural Evaluation by Generalized Integral Property. In Oswald, E. and Fischlin, M., editors, *Advances in Cryptology – EUROCRYPT 2015*, pages 287–314, Berlin, Heidelberg. Springer Berlin Heidelberg.

Xiang, Z., Zhang, W., Bao, Z., and Lin, D. (2016). Applying MILP Method to Searching Integral Distinguishers Based on Division Property for 6 Lightweight Block Ciphers. In Cheon, J. H. and Takagi, T., editors, *Advances in Cryptology – ASIACRYPT 2016*, pages 648–678, Berlin, Heidelberg. Springer Berlin Heidelberg.

Xu, Z., Cui, J., Hu, K., and Wang, M. (2024). Integral attack on the full future block cipher. *Tsinghua Science and Technology*.

Yalli, J. S., Hasan, M. H., Jung, L. T., Yerima, A. I., Aliyu, D. A., Maiwada, U. D., Al-Selwi, S. M., and Shaikh, M. U. (2025). A Systematic Review For Evaluating IoT Security: A Focus On Authentication, Protocols and Enabling Technologies. *IEEE Internet of Things Journal*.