



ARTICLE

Security Analysis of a Modulo- 2^{16} Implementation Variant of the MBRISI Lightweight ARX Block Cipher: Weak Keys, Deterministic Differentials, and Equivalent Keys

Yanjun Li^{1,2}, Yiping Lin^{2,*}, Yuting Ni², Lixian Zhang² and Shanshan Huo¹

¹Information Industry Information Security Evaluation Center, The 15th Research Institute of China Electronics Technology Group Corporation, Beijing, China

²Beijing Electronic Science and Technology Institute, Beijing, China

*Corresponding Author: Yiping Lin. Email: linyiping2225@163.com

Received: 11 June 2026; Accepted: 14 August 2026; Published: 22 October 2026

ABSTRACT: This paper presents an exact security evaluation of a closed modulo- 2^{16} implementation variant of the lightweight Add-Rotate-XOR (ARX) block cipher named MBRISI. We first resolve an arithmetic ambiguity in the original specification: modulo-65,537 addition on 16-bit words may produce the unrepresentable value 65,536, while representing its outputs by 16-bit overflow is non-injective. We construct distinct plaintext pairs that merge after the first round and consequently produce identical ciphertexts. In contrast, modulo- 2^{16} addition is closed and bijective over the 16-bit word space; therefore, all subsequent weak-key and equivalent-key results apply exclusively to this implementation variant. We prove that addition by a fixed round key is affine over $\text{GF}(2)^{16}$ if and only if the key belongs to $\{0x0000, 0x4000, 0x8000, 0xC000\}$, in which case every XOR difference propagates deterministically. Because this set is closed under the MBRISI round-key recurrence, weak initial subkeys make the complete ten-round encryption mapping affine over $\text{GF}(2)^{32}$. By modeling the rotation-XOR preprocessing as linear maps, we show that their images equal the even-parity subspace, their kernels have dimension one, and every image element has exactly two complementary preimages. These properties reduce exact weak-key counting and structural identification from 2^{32} half-key-pair tests to $\mathcal{O}(2^{15})$ word operations and yield exactly 2^{36} universal weak master keys, representing 2^{-28} of the key space. Such keys induce deterministic full-round differentials, while large equivalent-key fibers produce identical round-key sequences and encryption mappings. Beyond MBRISI, the analysis yields a reusable audit framework for clarifying arithmetic domains, characterizing exceptional addition constants and carry behavior, testing key-schedule invariants, and guiding the secure design and evaluation of lightweight ARX ciphers.

KEYWORDS: Lightweight block cipher; ARX cipher; IoT security; differential cryptanalysis; weak key; deterministic differential; equivalent key; key schedule

1 Introduction

The rapid growth of the Internet of Things (IoT) has led to the deployment of radio-frequency identification tags, smart cards, wireless sensors, and embedded controllers in open or otherwise untrusted environments. Such platforms often impose strict limits on memory, chip area, energy, and computational throughput. Established block ciphers such as the Advanced Encryption Standard (AES) and Camellia provide standard reference points for conventional block-cipher design [1,2]. In more constrained platforms, lightweight cryptography instead seeks a practical balance between security, implementation footprint, and performance.

Add–Rotate–XOR (ARX) constructions are especially attractive because their basic operations map efficiently to general-purpose processors and compact hardware. Representative ARX or ARX-like primitives include SPECK, the Lightweight Encryption Algorithm (LEA), Skein’s Threefish tweakable block cipher, and HIGHT (HIGH security and light weight) [3–6]. Their efficiency, however, does not remove the need for careful cryptanalysis. Modular addition is nonlinear with respect to XOR, and carry propagation creates dependencies across bit positions. When one operand is a round key, the differential behavior may depend strongly on the key pattern. Recent work has refined the analysis of ARX components and their interactions from differential, additive-differential, linear, differential–linear, rotational, and division-property perspectives [7–13]. Classical differential cryptanalysis provides the broader framework for assessing such propagation effects [14]. The classical Markov-cipher framework and more recent studies of its assumptions further show that key-dependent differential propagation should not be treated as negligible [15–18].

Weak diffusion in either the round function or the key schedule can reduce a cipher’s effective security margin. Differential attacks have been reported on lightweight designs such as SLIM, LCB, and SHADOW [19,20]. Differential cryptanalysis has also been reported for PICO [21], while TEA and XTEA provide classic examples of compact arithmetic/shift/XOR designs whose design and key-schedule evolution are documented in [22–24]. MILP and related constraint-based methods are widely used to search differential and related trails in bit-oriented and ARX primitives [25–28], while dedicated methods experimentally verify difference-based trails [29].

MBRISI, used as the cipher name in the original publication, is a lightweight block cipher with a 32-bit block and a 64-bit master key [30]. Its round function uses a key-dependent modular addition $X \boxplus K$ as its only nonlinear component, while all later round keys are generated recursively from the first two subkeys. This combination makes MBRISI a useful case study for examining how exceptional modular-addition constants interact with limited key-schedule diffusion.

The arithmetic domain must first be stated precisely. The original MBRISI description uses 16-bit words but specifies addition modulo 65,537. The result $65,536 = 0x10000$ cannot be represented in a 16-bit word without an additional convention. This paper therefore analyzes the natural closed 16-bit implementation in which addition is performed modulo 2^{16} . The round-function structure, key-schedule framework, block size, and key size are retained, but this modulo- 2^{16} implementation is a distinct variant rather than an algebraically equivalent form of the original modulo-65,537 construction. All weak-key and equivalent-key claims below concern this implementation variant; the original modulo-65,537 description is considered only in the arithmetic-domain clarification and the overflow-induced collision analysis of Section 6.1.

The main contributions are as follows.

1. Arithmetic-domain clarification and collision analysis. We compare mathematical modulo-65,537 addition, its 16-bit overflow representation, and modulo- 2^{16} addition. We show that modulo-65,537 arithmetic may produce an unrepresentable value, prove that 16-bit overflow handling is non-injective, construct distinct plaintext pairs that collide after the first round, and delimit all subsequent weak-key and equivalent-key analysis to the closed and bijective modulo- 2^{16} implementation variant.
2. Exact component-level characterization. We prove that

$$\mathcal{W} = \{0x0000, 0x4000, 0x8000, 0xC000\}$$

is exactly the set of 16-bit constants for which $x \mapsto x + K \bmod 2^{16}$ is affine over $\text{GF}(2)^{16}$. Consequently, every XOR difference propagates deterministically through modular addition for $K \in \mathcal{W}$.

3. Efficient weak-key identification and exact counting. We model the rotation–XOR preprocessing as linear maps over $\text{GF}(2)$, characterize their images, kernels, and preimage multiplicities, reduce exact

- counting and structural identification from 2^{32} half-key-pair tests to $O(2^{15})$ word operations, and prove that the universal weak-key class under Definition 1 contains exactly $2^{36} = 68,719,476,736$ master keys.
4. Full-round and key-equivalence consequences. We prove that every key in the identified class makes the complete ten-round encryption mapping affine and quantify large equivalent-key fibers whose members induce identical round-key sequences and encryption mappings.
 5. Transferable ARX-like audit and design guidance. We distill the MBRISI analysis into a reusable workflow: clarify the implemented arithmetic domain, characterize exceptional addition constants and carry behavior, linearize the surrounding preprocessing, test whether the key schedule preserves exceptional sets, and inspect full-round differential behavior and equivalent-key fibers. The resulting methodology also emphasizes the importance of operation placement, round constants, nonlinear schedule updates, and multiword key mixing. This is a cipher-audit framework, not a claim that established designs such as SPECK, TEA/extended TEA (XTEA), HIGHT, or LEA admit the same weakness.

The remainder of the paper is organized as follows. [Section 2](#) specifies the modulo- 2^{16} MBRISI implementation variant. [Section 3](#) clarifies the arithmetic domain and reports an exhaustive comparison of the alternative modulus interpretations. [Section 4](#) analyzes deterministic propagation through the key-dependent modular addition. [Section 5](#) gives the exact weak-key characterization, linear-map structure, and master-key count. [Section 6](#) analyzes overflow-induced plaintext collisions and presents full-round differential validation, equivalent-key experiments, and complexity results. [Section 7](#) compares representative ARX-like designs and formulates transferable audit and design guidelines. [Section 8](#) concludes the paper.

2 Specification of the Modulo- 2^{16} MBRISI Implementation Variant

MBRISI was proposed as a lightweight block cipher with a 32-bit block, a 64-bit master key, and ten rounds [30]. This section fixes the notation and describes the modulo- 2^{16} implementation analyzed in the remainder of the paper.

2.1 Notation

[Table 1](#) summarizes the notation. Bit positions are numbered from the least significant bit, which has index 0.

Table 1: Notation used for the modulo- 2^{16} MBRISI implementation variant.

Symbol	Description
P, C	32-bit plaintext and ciphertext
L_i, R_i	16-bit left and right branches after round i
K	64-bit master key
SK_i	16-bit subkey used in round i , $1 \leq i \leq 10$
$\parallel$	Concatenation
$\oplus$	Bitwise XOR
ROTL_s	Left rotation by s bit positions
$\boxplus$	Addition modulo 2^{16}
ΔX	XOR difference between two values of X

2.2 Encryption

The plaintext is parsed as $P = L_0 \parallel R_0$, where $L_0, R_0 \in \{0, 1\}^{16}$. For $0 \leq i \leq 9$, round $i + 1$ applies

$$\begin{aligned} L_{i+1} &= (\text{ROTL}_1(L_i) \oplus \text{ROTL}_8(L_i) \oplus \text{ROTL}_2(L_i) \oplus R_i) \boxplus SK_{i+1}, \\ R_{i+1} &= L_i. \end{aligned} \quad (1)$$

The ciphertext is $C = L_{10} \parallel R_{10}$. Fig. 1 illustrates the ten-round Feistel-like structure. Rotations and XORs are linear over $\text{GF}(2)$; the addition by SK_{i+1} is the only nonlinear operation with respect to XOR differences.

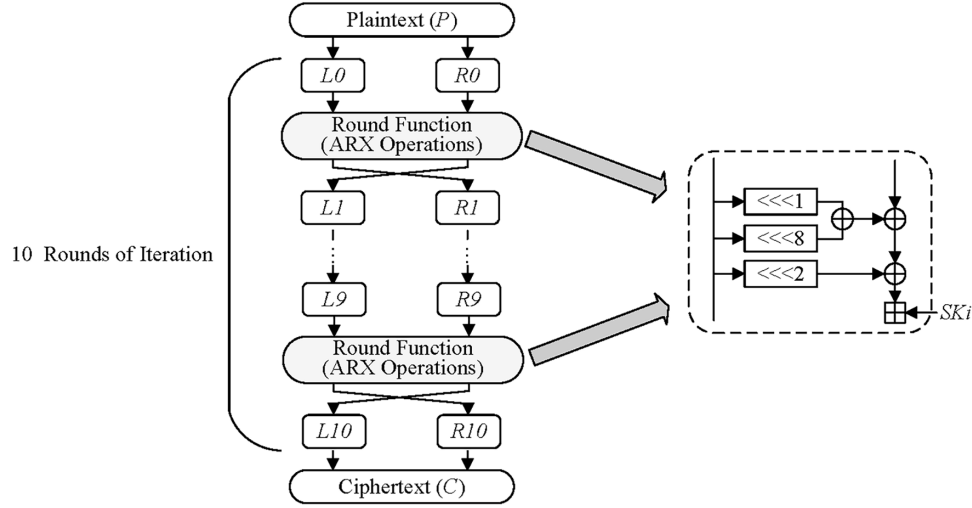


Figure 1: Encryption process of MBRISL.

2.3 Key Schedule

The 64-bit master key is rearranged into four 16-bit words. For a slice $K_{[i:j]}$, indices decrease from i to j :

$$\begin{aligned} kc_1 &= K_{[3:0]} \parallel K_{[19:16]} \parallel K_{[35:32]} \parallel K_{[51:48]}, \\ kc_2 &= K_{[7:4]} \parallel K_{[23:20]} \parallel K_{[39:36]} \parallel K_{[55:52]}, \\ kc_3 &= K_{[11:8]} \parallel K_{[27:24]} \parallel K_{[43:40]} \parallel K_{[59:56]}, \\ kc_4 &= K_{[15:12]} \parallel K_{[31:28]} \parallel K_{[47:44]} \parallel K_{[63:60]}. \end{aligned} \quad (2)$$

The preprocessing maps are

$$\begin{aligned} q_1 &= kc_1 \oplus \text{ROTL}_5(kc_1), & q_2 &= kc_2 \oplus \text{ROTL}_3(kc_2), \\ q_3 &= kc_3 \oplus \text{ROTL}_5(kc_3), & q_4 &= kc_4 \oplus \text{ROTL}_3(kc_4). \end{aligned} \quad (3)$$

The first two round keys are

$$SK_1 = q_1 \boxplus q_2, \quad SK_2 = q_3 \boxplus q_4, \quad (4)$$

and the remaining subkeys satisfy

$$SK_i = SK_{i-1} \boxplus SK_{i-2}, \quad 3 \leq i \leq 10. \quad (5)$$

Thus, the entire ten-round key sequence is determined by the 32-bit pair (SK_1, SK_2) . Fig. 2 summarizes the key schedule.

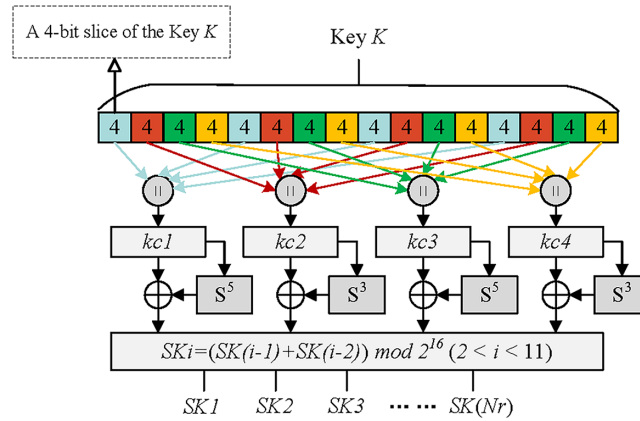


Figure 2: MBRISI key-schedule structure.

Algorithm 1 summarizes the recursive generation of $SK_3, \dots, SK_{10}$ from SK_1 and SK_2 .

Algorithm 1: Generation of $SK_3, \dots, SK_{10}$

Require: SK_1, SK_2

Ensure: $SK_3, \dots, SK_{10}$

- 1: **for** $i = 3$ to 10 **do**
 - 2: $SK_i \leftarrow (SK_{i-1} + SK_{i-2}) \bmod 2^{16}$
 - 3: **end for**
 - 4: **return** $SK_3, \dots, SK_{10}$
-

3 Modulus Clarification and Experimental Comparison

The original MBRISI specification uses modulus 65,537 in a 16-bit word-oriented structure. For 16-bit operands X and K , the result $(X + K) \bmod 65,537$ may equal 65,536 (0x10000), which exceeds the 16-bit representation range. Handling this overflow by 16-bit wraparound is also problematic. For $K = 0x0002$, the inputs $X = 0xFFFFE$ and $X = 0xFFFF$ produce 65,536 and 0, respectively, under modulo-65,537 addition; both are represented as 0 after 16-bit overflow handling, so the resulting mapping is not bijective.

We therefore compare three interpretations: the mathematical operation $M_{65537} = (X + K) \bmod 65,537$, its 16-bit overflow representation M_{ov} , and $M_{16} = (X + K) \bmod 2^{16}$. For each representative key in Table 2, all 2^{16} input words were exhaustively tested. Here, N_{65536} counts results exceeding the 16-bit range, and N_{diff} counts differences between M_{65537} and M_{16} .

Table 2: Exhaustive comparison of the three modulus interpretations.

K	N_{65536}	N_{diff}	$ \text{Im}(M_{ov}) $	$ \text{Im}(M_{16}) $
0x0000	0	0	65,536	65,536
0x0001	1	1	65,536	65,536
0x0002	1	2	65,535	65,536
0x8000	1	32,768	65,535	65,536
0xFFFF	1	65,535	65,535	65,536

The experiment shows that modulo-65,537 is not closed over 16-bit words, whereas overflow-induced wraparound may introduce collisions. In contrast, modulo- 2^{16} always gives 2^{16} distinct outputs for a fixed key and is therefore a well-defined 16-bit ARX operation. Moreover, for uniformly selected X and K , the two mathematical operations differ with probability $65,535/131,072 \approx 0.499992$. Hence, modulo- 2^{16} is necessary for the closed and bijective 16-bit implementation analyzed here. It defines a distinct implementation variant rather than an equivalent form of the original modulo-65,537 construction. Except for the overflow-induced collision analysis in [Section 6.1](#), all subsequent theoretical and experimental results are limited to the modulo- 2^{16} implementation variant.

4 Deterministic Differential Properties of $X \boxplus K$

For a fixed n -bit constant K , define

$$F_K(x) = (x + K) \bmod 2^n$$

and the XOR derivative in direction Δ by

$$D_\Delta F_K(x) = F_K(x) \oplus F_K(x \oplus \Delta). \quad (6)$$

A differential $\Delta \rightarrow \Gamma$ has probability one if $D_\Delta F_K(x) = \Gamma$ for every x . If this holds for every Δ , then every derivative is independent of x , which is equivalent to F_K being affine over $\text{GF}(2)^n$.

[Fig. 3](#) shows the component analyzed in this section. In MBRISI, the data-dependent input X is the XOR of rotated state words and the opposite branch, while K is the round key.

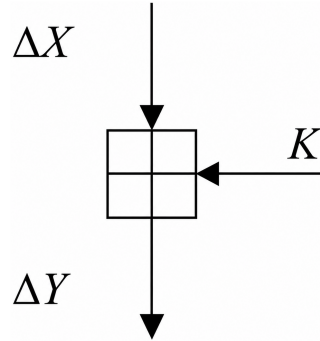


Figure 3: Differential propagation through $X \boxplus K$.

For $n \geq 2$, define

$$\mathcal{W}_n = \{t 2^{n-2} \bmod 2^n : t \in \mathbb{Z}_4\}. \quad (7)$$

For $n = 16$, this set is

$$\mathcal{W} = \mathcal{W}_{16} = \{0x0000, 0x4000, 0x8000, 0xC000\}.$$

Theorem 1: For every $K \in \mathcal{W}_n$, the mapping $F_K(x) = (x + K) \bmod 2^n$ is affine over $\text{GF}(2)^n$. Consequently, for every input XOR difference Δ , the output difference $D_\Delta F_K(x)$ is independent of x and occurs with probability one.

Proof: Write $K = t 2^{n-2}$ with $t \in \{0, 1, 2, 3\}$. Addition by K leaves the lower $n - 2$ bits unchanged and adds t modulo 4 to the two most significant bits. If these bits are denoted by (x_{n-1}, x_{n-2}) , the four possible maps are

listed in Table 3. Each coordinate is an affine Boolean function of the input bits. The remaining coordinates are identity maps, so F_K is affine on all n bits. The XOR derivative of an affine mapping depends only on Δ , not on x . $\square$

Table 3: Affine action of the four candidate addition constants.

t	$K = t2^{n-2}$	Action on (x_{n-1}, x_{n-2})	Derivative on $(\delta_{n-1}, \delta_{n-2})$
0	$00\ 0^{n-2}$	(x_{n-1}, x_{n-2})	$(\delta_{n-1}, \delta_{n-2})$
1	$01\ 0^{n-2}$	$(x_{n-1} \oplus x_{n-2}, x_{n-2} \oplus 1)$	$(\delta_{n-1} \oplus \delta_{n-2}, \delta_{n-2})$
2	$10\ 0^{n-2}$	$(x_{n-1} \oplus 1, x_{n-2})$	$(\delta_{n-1}, \delta_{n-2})$
3	$11\ 0^{n-2}$	$(x_{n-1} \oplus x_{n-2} \oplus 1, x_{n-2} \oplus 1)$	$(\delta_{n-1} \oplus \delta_{n-2}, \delta_{n-2})$

Theorem 1 establishes that the four constants in $\mathcal{W}$ are sufficient for deterministic propagation of every XOR difference. Section 5.1 proves the converse: no other 16-bit constant has this universal property. The component-level result is therefore exact rather than a collection of selected probability-one characteristics.

5 Exact Universal Weak-Key Characterization

A weak-key definition based only on one probability-one differential would be too broad. Indeed, for every K ,

$$(x \oplus 2^{15}) + K \equiv (x + K) + 2^{15} \pmod{2^{16}},$$

so $\Delta X = 2^{15}$ always propagates deterministically. We therefore use the following stronger definition.

Definition 1 (universally deterministic round key): A 16-bit round key K is universally deterministic if, for every XOR difference ΔX ,

$$F_K(x) \oplus F_K(x \oplus \Delta X), \quad F_K(x) = (x + K) \bmod 2^{16},$$

is independent of x . Equivalently, F_K is affine over $\text{GF}(2)^{16}$. A 64-bit master key is a universal weak master key if all its round keys satisfy this property. The count below is exact under this definition, but does not cover broader characteristic-dependent weak-key classes.

5.1 Exact Characterization of Universally Deterministic Round Keys

Define

$$\mathcal{W} = \{t2^{14} \bmod 2^{16} : t \in \mathbb{Z}_4\} = \{0x0000, 0x4000, 0x8000, 0xC000\}. \quad (8)$$

Theorem 2: The mapping $F_K(x) = (x + K) \bmod 2^{16}$ is affine over $\text{GF}(2)^{16}$ if and only if $K \in \mathcal{W}$.

Proof: Sufficiency follows from Theorem 1. For necessity, let $K \notin \mathcal{W}$ and write $K = 2^r u$, where u is odd. Then $r = v_2(K) \leq 13$. Set

$$m = 16 - r, \quad \Delta = 2^r, \quad y_1 = 1 - u, \quad y_2 = 3 - u$$

modulo 2^m , and let $x_i = 2^r y_i$. Since y_1, y_2 are even, toggling bit r adds 2^r , and direct calculation gives

$$F_K(x_1) \oplus F_K(x_1 \oplus \Delta) = 3 \cdot 2^r, \quad F_K(x_2) \oplus F_K(x_2 \oplus \Delta) = 7 \cdot 2^r.$$

Thus the derivative for the same nonzero Δ depends on x , so F_K is not affine. $\square$

Hence $\mathcal{W}$ is exactly the set of 16-bit constants for which all XOR differences propagate deterministically through modular addition.

5.2 Closure under the Round-Key Recurrence

Theorem 3: If $SK_1, SK_2 \in \mathcal{W}$ and

$$SK_i = (SK_{i-1} + SK_{i-2}) \bmod 2^{16}, \quad 3 \leq i \leq 10,$$

then $SK_i \in \mathcal{W}$ for all $1 \leq i \leq 10$.

Proof: For $x = a2^{14}$ and $y = b2^{14}$, with $a, b \in \mathbb{Z}_4$,

$$(x + y) \bmod 2^{16} = ((a + b) \bmod 4)2^{14} \in \mathcal{W}.$$

Thus $\mathcal{W}$ is an additive subgroup of $\mathbb{Z}/2^{16}\mathbb{Z}$, and the claim follows by induction. $\square$

Therefore, a master key is universally weak if and only if $SK_1, SK_2 \in \mathcal{W}$.

Theorem 4: For every universal weak master key, the complete ten-round encryption mapping is affine over $\text{GF}(2)^{32}$. Consequently,

$$E_K(P) \oplus E_K(P \oplus \Delta P)$$

is independent of P for every input difference ΔP .

Proof: Rotations, XORs, and the branch swap are linear over $\text{GF}(2)$, while addition by each round key is affine by Theorem 2. Hence every round, and therefore their ten-round composition, is affine. $\square$

5.3 Image of the Rotation-XOR Transformations over $\text{GF}(2)$

For $s \in \{3, 5\}$, define

$$T_s : \text{GF}(2)^{16} \rightarrow \text{GF}(2)^{16}, \quad T_s(x) = x \oplus \text{ROTL}_s(x).$$

Thus,

$$q_1 = T_5(kc_1), \quad q_2 = T_3(kc_2), \quad q_3 = T_5(kc_3), \quad q_4 = T_3(kc_4).$$

Let

$$\mathcal{E} = \{q \in \text{GF}(2)^{16} : q_{15} \oplus q_{14} \oplus \cdots \oplus q_0 = 0\} \tag{9}$$

be the even-parity subspace.

Theorem 5: For $s \in \{3, 5\}$,

$$\ker(T_s) = \{0^{16}, 1^{16}\}, \quad \text{rank}(T_s) = 15, \quad \text{Im}(T_s) = \mathcal{E}.$$

Every $q \in \mathcal{E}$ has exactly two complementary preimages, whereas no odd-parity word has a preimage.

Proof: The condition $T_s(x) = 0$ is equivalent to $x = \text{ROTL}_s(x)$. Since $\gcd(16, 3) = \gcd(16, 5) = 1$, each rotation has one orbit on all 16 coordinates, so x is either 0^{16} or 1^{16} . Rank-nullity therefore gives rank 15.

Each input bit occurs twice in $x \oplus \text{ROTL}_s(x)$, so every output has even parity and $\text{Im}(T_s) \subseteq \mathcal{E}$. Both spaces have dimension 15, hence they are equal. Each image element has $|\ker(T_s)| = 2$ complementary preimages. $\square$

5.4 Exact Counting of Admissible q -Pairs

For $w \in \mathcal{W}$, define

$$\mathcal{Q}_w = \{(q_1, q_2) \in \mathcal{E}^2 : (q_1 + q_2) \bmod 2^{16} = w\}, \quad \mathcal{Q} = \bigcup_{w \in \mathcal{W}} \mathcal{Q}_w.$$

Theorem 6: *The exact number of admissible pairs is*

$$|\mathcal{Q}| = 2^{16} = 65536.$$

Proof: Fix $q_1 \in \mathcal{E}$. For $w_t = t2^{14} \in \mathcal{W}$, define

$$q_2(t) = (w_t - q_1) \bmod 2^{16}, \quad t \in \{0, 1, 2, 3\}.$$

These four words have the same lower 14 bits, while their two most significant bits run through 00, 01, 10, 11. Exactly two therefore have even parity. Since $|\mathcal{E}| = 2^{15}$, and the sets $\mathcal{Q}_w$ are disjoint,

$$|\mathcal{Q}| = 2^{15} \cdot 2 = 2^{16}.$$

$\square$

5.5 Exact Number of Universal Weak Master Keys

By Theorem 5, each admissible q_1 and q_2 has two preimages, so every pair in $\mathcal{Q}$ lifts to four distinct (kc_1, kc_2) pairs. Hence

$$N_{\text{half}} = 4|\mathcal{Q}| = 2^{18} = 262,144.$$

The pairs (kc_1, kc_2) and (kc_3, kc_4) independently determine SK_1 and SK_2 , respectively. Since

$$K \mapsto (kc_1, kc_2, kc_3, kc_4)$$

is a permutation of the 64 master-key bits, no key is counted twice. Therefore,

$$N_{\text{univ}} = N_{\text{half}}^2 = (2^{18})^2 = 2^{36} = 68,719,476,736. \quad (10)$$

The corresponding fraction of the 64-bit key space is

$$\frac{N_{\text{univ}}}{2^{64}} = 2^{-28}.$$

6 Experimental Validation and Complexity

[Section 6.1](#) examines collisions caused by the 16-bit overflow representation of modulo-65,537 outputs. The remaining subsections validate the weak-key and equivalent-key properties of the closed modulo- 2^{16} implementation variant.

6.1 Overflow-Induced Plaintext Collisions

Consider the 16-bit overflow representation

$$M_{\text{ov}}(x, k) = ((x + k) \bmod 65,537) \bmod 2^{16}.$$

For any first-round subkey $2 \leq k \leq 2^{16} - 1$, define

$$x_0 = 2^{16} - k, \quad x_1 = 2^{16} + 1 - k.$$

Although $x_0 \neq x_1$,

$$M_{\text{ov}}(x_0, k) = M_{\text{ov}}(x_1, k) = 0,$$

because $x_0 + k = 65,536$ overflows to zero in a 16-bit word, whereas $x_1 + k = 65,537 \equiv 0 \pmod{65,537}$.

Let

$$A(L) = \text{ROTL}_1(L) \oplus \text{ROTL}_8(L) \oplus \text{ROTL}_2(L).$$

For any L_0 , construct

$$P_j = L_0 \parallel (A(L_0) \oplus x_j), \quad j \in \{0, 1\}.$$

Both plaintexts enter the state $(0, L_0)$ after the first round; hence

$$P_0 \neq P_1, \quad E_K(P_0) = E_K(P_1),$$

independently of the remaining rounds and subkeys. Given either member of the pair and SK_1 , its colliding counterpart is obtained by replacing x_0 with x_1 , or vice versa, in the first-round input.

The experiment used $SK_1 = 0x0002$, $SK_2 = 0x1357$, and ten rounds, so $x_0 = 0xFFFFE$ and $x_1 = 0xFFFF$. All ten constructed pairs produced identical first-round states and ten-round ciphertexts. Five representative cases are shown in [Table 4](#).

Table 4: Representative overflow-induced plaintext collisions.

L_0	P_0	P_1	Common Ciphertext
0x1234	0x1234A754	0x1234A755	0xC08FC9CE
0x2345	0x23457143	0x23457142	0x4CD5C029
0x3456	0x3456103E	0x3456103F	0x32BC5B57
0x4567	0x456707E8	0x456707E9	0x50473FCE
0x5678	0x567872B9	0x567872B8	0xBDA1C149

Thus, overflow handling destroys injectivity: once two plaintexts merge after the first round, no subsequent round can distinguish them. In contrast, addition modulo 2^{16} is bijective for every fixed subkey and does not admit this component-level collision.

6.2 Full-Round Differential Validation

Theorem 4 proves that every universal weak master key induces an affine ten-round mapping. The following experiment serves as an implementation cross-check and comparison with ordinary keys.

We selected 20 weak master keys and 20 ordinary random master keys. For each key, $N = 2^{16}$ plaintext pairs satisfying $P' = P \oplus \Delta P$, with $\Delta P = 0x00000001$, were encrypted for $r = 1, \dots, 10$ rounds. We measured

$$\widehat{p}_r = \max_{\Delta C} \frac{\#\{P : E_K(P) \oplus E_K(P \oplus \Delta P) = \Delta C\}}{N} \quad (11)$$

and the number N_Δ of distinct output differences. Table 5 reports averages over the 20 keys.

Table 5: Empirical differential propagation from one to ten rounds.

Rounds	Weak-Key $\widehat{p}_r$	Random-Key $\widehat{p}_r$	Random-Key N_Δ
1	1	6.748×10^{-1}	10.00
2	1	1.930×10^{-1}	663.50
3	1	7.383×10^{-2}	10,625.60
4	1	3.986×10^{-3}	38,210.70
5	1	8.362×10^{-4}	58,414.30
6	1	1.694×10^{-4}	64,349.75
7	1	5.188×10^{-5}	65,435.35
8	1	3.204×10^{-5}	65,529.70
9	1	2.213×10^{-5}	65,535.20
10	1	2.213×10^{-5}	65,535.45

For every tested weak key, $\widehat{p}_r = 1$ and $N_\Delta = 1$ for $1 \leq r \leq 10$, confirming the affine-composition result. In contrast, the random-key maximum frequency decreases rapidly, and the ten-round experiment produces on average 65,535.45 differences, close to the 65,535.5 expected for uniformly distributed 32-bit differences.

6.3 Equivalent-Key Collisions

The key schedule also admits equivalent master keys. Since $SK_3, \dots, SK_{10}$ are determined only by SK_1 and SK_2 ,

$$(SK_1, SK_2) = (SK'_1, SK'_2) \implies E_K(P) = E_{K'}(P) \quad \text{for all } P. \quad (12)$$

Moreover, Theorem 5 gives

$$T_s(\bar{x}) = \bar{x} \oplus \text{ROTL}_s(\bar{x}) = T_s(x).$$

Thus, complementing any subset of kc_1, kc_2, kc_3, kc_4 leaves every round key unchanged and gives at least $2^4 = 16$ master-key representations of each reachable sequence.

For a fixed weak initial pair $(SK_1, SK_2) = (w_a, w_b) \in \mathcal{W}^2$, let $H(w) = 4|\mathcal{Q}_w|$. The exact number of master keys generating the same complete round-key sequence is

$$N_{\text{fiber}}(w_a, w_b) = 16|\mathcal{Q}_{w_a}||\mathcal{Q}_{w_b}| = H(w_a)H(w_b). \quad (13)$$

Across the 16 weak initial pairs, the fiber size ranges from 1,908,990,864 to 7,635,264,400, with average 2^{32} . More generally, the encryption mapping is determined by the 32-bit pair (SK_1, SK_2) , so the nominal 64-bit key space induces at most 2^{32} distinct mappings.

For example, $K = 0x0123456789ABCDEF$ and $K' = 0x012C456889A4CDE0$ both generate $(SK_1, SK_2) = (0x4E81, 0x7DB0)$.

No ciphertext mismatch was observed over 2^{20} randomly selected plaintexts, confirming that the two master keys implement the same encryption mapping.

6.4 Independent Exact Verification

The following checks independently confirm the analytical results.

1. All 2^{16} inputs were tested for each $K \in \mathcal{W}$; for every other constant, the derivative witness in Theorem 2 was verified.
2. The matrices and exhaustive enumerations of T_3 and T_5 both give rank 15, the even-parity image, and two complementary preimages.
3. Admissible q -pairs were counted both directly in $\mathcal{E}$ and through canonical kc_1 representatives, with identical results for every w .
4. All $262,144 = 2^{18}$ valid (kc_1, kc_2) pairs were explicitly constructed and checked for uniqueness.
5. The master-key rearrangement was verified to be a permutation of all 64 bit positions.
6. All 16 initial pairs in $\mathcal{W}^2$ were verified to remain in $\mathcal{W}$ throughout ten rounds.

The exact counts for the four possible values of SK_1 are summarized in [Table 6](#).

Table 6: Independent exact counts for the four values of SK_1 .

$SK_1 = w$	$ \mathcal{Q}_w $	Number of (kc_1, kc_2) Pairs
0x0000	10,923	43,692
0x4000	16,383	65,532
0x8000	21,845	87,380
0xC000	16,385	65,540
Total	65,536	262,144

The counts satisfy

$$10,923 + 16,383 + 21,845 + 16,385 = 2^{16}, \quad 4 \cdot 2^{16} = 2^{18}, \quad (2^{18})^2 = 2^{36}.$$

All checks are exact under the universal deterministic definition; broader characteristic-dependent weak-key classes are outside the present count.

6.5 Search Complexity

A direct search over all (kc_1, kc_2) pairs requires 2^{32} tests. The image-space method scans 2^{15} elements of $\mathcal{E}$ and tests four values in $\mathcal{W}$, giving $O(2^{15})$ word operations for exact counting and structural identification. Each admissible (q_1, q_2) then lifts to four (kc_1, kc_2) pairs.

The corresponding complexity comparison is summarized in [Table 7](#).

The complexity bound concerns counting and structural identification. Explicitly outputting the 2^{18} half keys or 2^{36} full weak keys necessarily requires time proportional to the output size.

Table 7: Complexity comparison for one half of the key schedule.

Method	Domain	Complexity	Purpose
Naive pair search	All (kc_1, kc_2)	2^{32}	Test every half-key pair
Image-space method	$q_1 \in \mathcal{E}, w \in \mathcal{W}$	$O(2^{15})$	Exact counting and identification
Preimage verification	All 16-bit words	$O(2^{16})$	Confirm images and multiplicities

7 Discussion

7.1 Security Implications for MBRISI

The results show that MBRISI's security is strongly affected by the interaction between its key-dependent modular addition and its key schedule. For round keys in $\mathcal{W} = \{0x0000, 0x4000, 0x8000, 0xC000\}$, addition by the round key is affine over $\text{GF}(2)^{16}$. Because the remaining round operations are linear and the recurrence preserves $\mathcal{W}$, the expected nonlinear effect disappears throughout all ten rounds.

The universal weak-key class under Definition 1 contains exactly $2^{36} = 68,719,476,736$ keys, or 2^{-28} of the nominal 64-bit key space. This fraction is small under ideal uniform key generation, but it remains relevant for constrained deployments in which keys may be derived from low-entropy seeds, counters, serial numbers, or incompletely mixed device-specific values.

The exact count is definition-dependent. It is complete for master keys whose every round addition is affine for every XOR difference. It is not an exhaustive count of all keys that might support a selected high-probability or deterministic trail. In particular, the difference 2^{15} propagates deterministically through addition by every 16-bit constant, so a definition based only on the existence of one deterministic difference would be too broad to isolate a meaningful weak-key class.

The equivalent-key result exposes a separate and more general limitation. The complete encryption mapping is determined by the 32-bit pair (SK_1, SK_2) , so there are at most 2^{32} distinct mappings despite the nominal 64-bit key length. Complement symmetry gives at least 16 representations of every reachable round-key sequence. For weak initial subkey pairs, the exact fibers are much larger: about 1.91 to 7.64 billion master keys, with average 2^{32} . The general complement symmetry is independent of membership in $\mathcal{W}$.

7.2 Comparison with Representative ARX-Like Ciphers

Table 8 places MBRISI beside representative lightweight ARX or ARX-like designs. The comparison focuses on structural features relevant to the present analysis rather than on implementation performance or a complete security ranking.

Table 8: Structural comparison with representative ARX-like ciphers.

Cipher	Round-Structure Relation	Key-Schedule Feature Relevant Here	Interpretation for This Work
MBRISI	32-bit Feistel-like structure; the only XOR-nonlinear operation is addition by a subkey	Four rotation–XOR words produce SK_1, SK_2 ; all later subkeys follow a two-word modular-addition recurrence without round constants	Exceptional addition constants can be preserved across all rounds; this is the target analyzed in the paper

(Continued)

Table 8 (continued)

Cipher	Round-Structure Relation	Key-Schedule Feature Relevant Here	Interpretation for This Work
SPECK32/64 [3]	Same 32-bit block and 64-bit key profile; ARX round function with addition, rotation, and XOR	Recursive word mixing includes the round index, introducing round-dependent asymmetry	A close structural reference point showing why the weakness is not caused by ARX operations alone
TEA/XTEA [22–24]	Feistel-based arithmetic/shift/XOR designs with deliberately simple implementations	TEA's simple key schedule is known to admit equivalent keys; XTEA revised the key-selection rule and round-dependent indexing	A historical caution that lightweight simplicity must be accompanied by careful key-schedule analysis; the present MBRISI attack is not asserted to apply to TEA or XTEA
HIGHT [6]	Generalized Feistel design using byte-oriented addition, XOR, and rotations	Whitening keys and subkeys are expanded with round-dependent constants	A lightweight comparison in which explicit constants help differentiate round-key material
LEA [4]	Word-oriented ARX design using modular addition, rotations, and XOR	The key expansion uses round constants and rotation-dependent updates of multiple words	A reference design with substantially more round-dependent variation than MBRISI's two-subkey recurrence

Note: ARX, Add–Rotate–XOR; HIGHT, HIGH security and light weight; LEA, Lightweight Encryption Algorithm; TEA, Tiny Encryption Algorithm; XTEA, extended Tiny Encryption Algorithm; XOR, exclusive OR. MBRISI and SPECK are cipher names used without expanded forms in the cited publications.

The comparison supports two conclusions. First, MBRISI's weakness should not be attributed to ARX operations in general. The decisive issue is the combination of a key-dependent addition with a schedule that maps a 64-bit master key to only two initial round keys and then preserves certain algebraic patterns. Second, round constants, round-index injection, and multiword mixing do not by themselves prove security, but they provide round-dependent asymmetry that is absent from the MBRISI recurrence.

7.3 Transferable Analysis Framework

The MBRISI case study suggests the following audit workflow for ARX-like block ciphers with key-dependent modular additions.

1. Locate key-dependent nonlinear components. Identify operations of the form $X \boxplus SK_i$, or related addition–XOR structures, whose differential behavior may depend on the subkey.
2. Characterize exceptional constants. For each fixed addition constant, study the XOR derivative $D_{\Delta}F_K(x)$. Determine whether selected differences, or all differences, propagate independently of x . Affine constants and probability-one derivatives should be characterized with necessity and sufficiency whenever possible.
3. Linearize the surrounding preprocessing. Express XORs, rotations, and bit permutations as linear maps over $\text{GF}(2)$. Compute image, kernel, rank, and preimage multiplicities rather than enumerating the full master-key space.

4. Test preservation by the key schedule. Determine whether the exceptional subkey set is closed under the recurrence or otherwise invariant across rounds. Round constants and nonlinear updates must be included in this step.
5. Lift component properties to the full cipher. If the exceptional pattern survives, derive multi-round or full-round consequences, including deterministic differentials and characteristic-dependent weak-key classes.
6. Audit the master-key-to-round-key map. Measure the number of master keys mapping to the same initial or complete round-key sequence. Nontrivial fibers reveal equivalent-key behavior even when no deterministic differential is present.

This framework is informative for MBRISI-like designs, but its conclusions remain cipher-specific. SPECK, TEA/XTEA, HIGHT, and LEA are included as reference points for structure and key-schedule design; the paper does not claim that they contain the same weak-key class or exhibit the same weaknesses.

7.4 Design Recommendations

Future lightweight designs should avoid allowing a small, algebraically structured set of subkeys to remain invariant across multiple rounds. More generally, the security impact of key-dependent modular-addition placement and the interaction between carry-induced dependencies and adjacent rotations, XORs, and branch updates deserve further systematic study in lightweight-cipher analysis. By identifying the exceptional round keys and proving that the MBRISI key schedule preserves them, this work exposes a concrete structural weakness and contributes to addressing this analytical gap. Designers of similar constructions should therefore exercise caution and consider incorporating round constants, stronger multiword key mixing, and nonlinear key-schedule updates, together with explicit analysis of carry-induced dependencies and equivalent-key classes.

8 Conclusion

This paper evaluated the arithmetic-domain and weak-key security of MBRISI. For the original modulo-65,537 description, addition on 16-bit words may produce the unrepresentable value 65,536, and its 16-bit overflow representation is non-injective. For any first-round subkey $2 \leq SK_1 \leq 2^{16} - 1$, distinct plaintexts can therefore be constructed that merge after the first round and yield the same ciphertext. In contrast, modulo- 2^{16} addition is closed and bijective; all weak-key results in this paper concern only this implementation variant.

Under Definition 1, we proved that $\mathcal{W} = \{0x0000, 0x4000, 0x8000, 0xC000\}$ is exactly the set of constants for which fixed-key addition is affine over $\text{GF}(2)^{16}$. Since $\mathcal{W}$ is closed under the subkey recurrence, weak initial subkeys make the ten-round encryption mapping affine and force deterministic propagation of every XOR difference. The rotation-XOR maps have rank 15, image equal to the even-parity subspace, and two complementary preimages per image word. Hence each key-schedule half has 2^{18} valid choices, giving exactly $2^{36} = 68,719,476,736$ universal weak master keys, or 2^{-28} of the key space. The image-space method reduces exact counting from 2^{32} half-key-pair tests to $O(2^{15})$ word operations.

Independent exhaustive checks confirm the analytical results. A fixed weak initial-subkey pair has between 1,908,990,864 and 7,635,264,400 equivalent master keys, averaging 2^{32} , so the key schedule exhibits both weak-key and equivalent-key deficiencies.

More broadly, the analysis suggests a reusable ARX-like audit strategy: clarify the arithmetic domain, characterize addition derivatives and carry propagation, model surrounding linear operations, test key-schedule closure, and inspect equivalent-key fibers. The results are cipher-specific, but they show that

operation placement, round constants, multiword mixing, nonlinear schedule updates, and equivalent-key resistance should be assessed together.

Acknowledgement: Not applicable.

Funding Statement: This work is supported by Key Laboratory of Equipment Data Security and Guarantee Technology, Ministry of Education under Grant No. 2024010102.

Author Contributions: Conceptualization and methodology, Yanjun Li and Yiping Lin; software, validation, and formal analysis, Yiping Lin, Yuting Ni and Lixian Zhang; investigation and resources, Yanjun Li and Shanshan Huo; writing—original draft, Yiping Lin and Yuting Ni; writing—review and editing, all authors; supervision and funding acquisition, Yanjun Li. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The datasets generated and analysed during the current study are not publicly available but are available from the corresponding author on reasonable request.

Ethics Approval: Not applicable. This study does not involve human or animal subjects.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. National Institute of Standards and Technology (NIST). Advanced encryption standard (AES). In: FIPS PUB 197. Gaithersburg, MD, USA: NIST; 2001. updated 2023. doi:10.6028/NIST.FIPS.197-upd1.
2. Aoki K, Ichikawa T, Kanda M, Matsui M, Moriai S, Nakajima J, et al. Specification of Camellia: a 128-bit block cipher. Tokyo, Japan: NTT Corporation and Mitsubishi Electric Corporation; 2000 [cited 2026 Aug 10]. Available from: <https://info.isl.ntt.co.jp/crypt/eng/camellia/dl/cryptrec/call-3e.pdf>.
3. Beaulieu R, Shors D, Smith J, Treatman-Clark S, Weeks B, Wingers L. The SIMON and SPECK lightweight block ciphers. In: Proceedings of the 52nd Annual Design Automation Conference DAC '15; 2015 Jun 7–11; San Francisco, CA, USA. New York, NY, USA: Association for Computing Machinery; 2015. p. 1–6. doi:10.1145/2744769.2747946.
4. Hong D, Lee JK, Kim DC, Kwon D, Ryu KH, Lee DG. LEA: a 128-bit block cipher for fast encryption on common processors. In: Kim Y, Lee H, Perrig A, editors. Information security applications. Vol. 8267 of Lecture Notes in Computer Science. Cham, Switzerland: Springer; 2014. p. 3–27. doi:10.1007/978-3-319-05149-9_1.
5. Ferguson N, Lucks S, Schneier B, Whiting D, Bellare M, Kohno T, et al. The Skein hash function family. Version 1.3. 2010 [cited 2026 Aug 10]. Available from: <https://www.schneier.com/wp-content/uploads/2015/01/skein.pdf>.
6. Hong D, Sung J, Hong S, Lim J, Lee S, Koo BS, et al. HIGHT: a new block cipher suitable for low-resource device. In: Goubin L, Matsui M, editors. Cryptographic hardware and embedded systems—CHES 2006. Vol. 4249 of Lecture Notes in Computer Science. Berlin/Heidelberg, Germany: Springer; 2006. p. 46–59. doi:10.1007/11894063_4.
7. Machado AW. Differential probability of modular addition with a constant operand; 2001. Cryptology ePrint Archive, Paper 2001/052. [cited 2026 Aug 10]. Available from: <https://eprint.iacr.org/2001/052>.
8. Zheng B, Guan J. Differential characteristic probability of added key on modulo 2^n operation. J Electron Inf Technol. 2009;31(11):2708–12. doi:10.3724/SP.J.1146.2008.01448.
9. Mouha N, Kolomeec N, Akhtiamov D, Sutormin I, Panferov M, Titova K, et al. Maximums of the additive differential probability of exclusive-or. IACR Trans Symmetric Cryptol. 2021;2021(2):292–313. doi:10.46586/tosc.v2021.i2.292-313.
10. Niu Z, Sun S, Liu Y, Li C. Rotational differential-linear distinguishers of ARX ciphers with arbitrary output linear masks; 2022. Cryptology ePrint Archive, Paper 2022/765. [cited 2026 Aug 10]. Available from: <https://eprint.iacr.org/2022/765>.
11. Xu Z, Xu H, Tan L, Qi W. New linear approximation of modular addition and improved differential-linear cryptanalysis of SPARX-64/128. Cryptogr Commun. 2024;16(5):921–47. doi:10.1007/s12095-024-00708-z.

12. Hu K, Yap T. Perfect monomial prediction for modular addition. *IACR Trans Symmetric Cryptol.* 2024;2024(3):177–99. doi:10.46586/tosc.v2024.i3.177-199.
13. Gong X, Wang Q, Hao Y, Jiao L, Hu X. Persistence of hourglass(-like) structure: improved differential-linear distinguishers for several ARX ciphers; 2025. *Cryptology ePrint Archive*, Paper 2025/1667. [cited 2026 Aug 10]. Available from: <https://eprint.iacr.org/2025/1667>.
14. Biham E, Shamir A. Differential cryptanalysis of DES-like cryptosystems. *J Cryptol.* 1991;4(1):3–72. doi:10.1007/BF00630563.
15. Lai X, Massey JL, Murphy S. Markov ciphers and differential cryptanalysis. In: Davies DW, editor. *Advances in Cryptology—EUROCRYPT '91*. Vol. 547 of *Lecture Notes in Computer Science*. Berlin/Heidelberg, Germany: Springer; 1991. p. 17–38. doi:10.1007/3-540-46416-6_2.
16. Peyrin T, Tan QQ. Mind your path: on (key) dependencies in differential characteristics. *IACR Trans Symmetric Cryptol.* 2022;2022(4):179–207. doi:10.46586/tosc.v2022.i4.179-207.
17. Roh D. Validity of differential characteristics of ARX block ciphers. *IEEE Access.* 2023;11:100672–82. doi:10.1109/ACCESS.2023.3311898.
18. Xu Z, Li Y, Jiao L, Wang M, Meier W. Do NOT misuse the Markov cipher assumption: automatic search for differential and impossible differential characteristics in ARX ciphers; 2022. *Cryptology ePrint Archive*, Paper 2022/135. [cited 2026 Aug 10]. Available from: <https://eprint.iacr.org/2022/135>.
19. Chan YY, Khor CY, Teh JS, Teng WJ, Jamil N. Differential cryptanalysis of lightweight block ciphers SLIM and LCB. In: Chen J, He D, Lu R, editors. *Emerging information security and applications*. Vol. 1641 of *Communications in Computer and Information Science*. Cham, Switzerland: Springer; 2022. p. 55–67. doi:10.1007/978-3-031-23098-1_4.
20. Li Y, Lin H, Bi X, Huo S, Han Y. MILP-based differential cryptanalysis on full-round SHADOW. *J Inf Secur Appl.* 2024;81(1):103696. doi:10.1016/j.jisa.2023.103696.
21. Wang C, Zhang Z, Hu L. Differential cryptanalysis on ultra lightweight block cipher PICO. *J Cryptologic Res.* 2023;10(4):685–701. doi:10.13868/j.cnki.jcr.000619.
22. Wheeler DJ, Needham RM. TEA, a tiny encryption algorithm. In: Preneel B, editor. *Fast software encryption*. Vol. 1008 of *Lecture Notes in Computer Science*. Berlin/Heidelberg, Germany: Springer; 1995. p. 363–66. doi:10.1007/3-540-60590-8_29.
23. Needham RM, Wheeler DJ. TEA extensions. Cambridge, UK: Computer Laboratory, University of Cambridge; 1997 [cited 2026 Aug 10]. Available from: <https://www.cl.cam.ac.uk/ftp/users/djw3/xtea.ps>.
24. Shepherd SJ. The tiny encryption algorithm. *Cryptologia.* 2007;31(3):233–45. doi:10.1080/01611190601090606.
25. Fu K, Wang M, Guo Y, Sun S, Hu L. MILP-based automatic search algorithms for differential and linear trails for SPECK. In: Peyrin T, editor. *Fast software encryption*. Vol. 9783 of *Lecture Notes in Computer Science*. Berlin/Heidelberg, Germany: Springer; 2016. p. 268–88. doi:10.1007/978-3-662-52993-5_14.
26. Bagherzadeh E, Ahmadian Z. MILP-based automatic differential search for LEA and HIGHT block ciphers. *IET Inf Secur.* 2020;14(5):595–603. doi:10.1049/iet-ifs.2018.5539.
27. Zhou C, Zhang W, Ding T, Xiang Z. Improving the MILP-based security evaluation algorithm against differential/linear cryptanalysis using a divide-and-conquer approach. *IACR Trans Symmetric Cryptol.* 2019;2019(4):438–69. doi:10.13154/tosc.v2019.i4.438-469.
28. Chakraborty D, Hadipour H, Nguyen PH, Eichlseder M. Finding complete impossible differential attacks on AndRX ciphers and efficient distinguishers for ARX designs. *IACR Trans Symmetric Cryptol.* 2024;2024(3):84–176. doi:10.46586/tosc.v2024.i3.84-176.
29. Sadeghi S, Rijmen V, Bagheri N. Proposing an MILP-based method for the experimental verification of difference-based trails: application to SPECK, SIMECK. *Des Codes Cryptogr.* 2021;89(9):2113–55. doi:10.1007/s10623-021-00904-5.
30. Poojary A, Kiran Kumar VG, Nagesh HR. FPGA implementation novel lightweight MBRISI cipher. *J Ambient Intell Humaniz Comput.* 2023;14(9):11625–37. doi:10.1007/s12652-022-03726-y.