Weak Tweak-Key Analysis Of Blink Via Superbox

Shiyao Chen[1,2], Jian Guo[3] and Tianyu Zhang[3(✉)]

[1] School of Cyber Science and Technology, Shandong University, Qingdao, China
shiyao.chen@sdu.edu.cn
[2] State Key Laboratory of Cryptography and Digital Economy Security, Shandong University, Qingdao, China
[3] Nanyang Technological University, Singapore, Singapore
guojian@ntu.edu.sg,tianyu005@e.ntu.edu.sg

Abstract. This work presents the first third-party cryptanalysis of Blink, a recent tweakable block cipher built on the Three-Hash Framework with a long-key design. Leveraging the idea of Superbox, we develop a lightweight theoretical model capturing the value correlations, weak-key conditions, fixed-key probabilities and the local clustering behaviors inside a Blink Superbox when the value spaces are affine. This model is intended as a concrete, easy-to-follow specialization of existing generic frameworks adapted to the structure of Blink. We then build a simple pattern-based automatic tool to search for weak tweak-key differentials. The main results are as follows: For Blink-64 with 448-bit key size, we obtain a 10-round distinguisher with probability $2^{-50.42}$ for up to 2^{442} weak keys (equivalently 2^{-6} of the full key space); For Blink-128 with 1024-bit key size, we obtain a 10-round distinguisher with probability $2^{-68.83}$ for up to 2^{1010} weak keys (equivalently 2^{-14} of the full key space), which can be extended to a 12-round multiple differential distinguisher with probability $2^{-96.83}$. Based on the weak tweak-key distinguisher, we further mount a 13-round key recovery attack on Blink-64, recovering the full 448-bit master key with time complexity $2^{114.02}$ and 2^{56} chosen plaintexts. All these distinguishers and key recovery attacks work within the data and time bounds claimed by the designers. And our analysis remains consistent with the security of the full-round design of Blink, while offering additional insight into the edge-case behaviors arising from the tweak-key interaction in Blink, and could be potentially informative for future refinements of tweakable block cipher constructions.

Keywords: Blink · Superbox · Multiple Differential · Weak Tweak-Key Attack

1 Introduction

Low-latency cryptographic primitives have gained increasing attention in recent years, driven by the industrial demand for minimal delay in time-critical applications such as memory encryption, cache randomization, pointer authentication, and embedded code encryption. Several notable designs have been proposed in this direction, including PRINCE [BCG+12], PRINCEv2 [BEK+20], QARMA [Ava17], QARMAv2 [ABD+23], MANTIS [BJK+16], SCARF [CGL+23], SPEEDY [LMMR21], BipBip [BDD+23], and ChiLow [BDG+25a].

Most recently, Wang *et al.* [WHZ+25] proposed the Three-Hash Framework (THF) at ToSC 2025, which decouples tweak injection from the round permutation with the motivation of improving resistance to multiple-tweak attacks compared to the TWEAKEY framework [JNP14]. They apply THF in the design of Blink, a tweakable block cipher with the α-reflection structure used in PRINCE, MANTIS, and QARMA. The round function of Blink consists of an involutory 4-bit S-box, a MixKey layer with Midori MixColumn

matrix, and a fixed permutation layer. Moreover, Blink uses a long key, which avoids the use of a complex key schedule, ensures the independence between the subkeys used for hash generation and offers higher security.

Multiple Differential Attacks. Multiple differential cryptanalysis [BG11] clusters large sets of compatible differential trails that map the same input-difference pattern to the same output-difference pattern, and have become the dominant technique against ciphers with the α-reflection structure. Canteaut *et al.* applied this approach to PRINCE [CFG+14], achieving the first key recovery attacks on 10 and 11 rounds. Dobraunig *et al.* [DEKM16] successfully broke MANTIS_5 by constructing an optimal related-tweak truncated differential and expanding it into a practical multiple-differential cluster, enabling a verified 10-round key recovery. Eichlseder and Kales [EK18] further generalized clustering through semi-truncated characteristics that describe the full set of compatible related-tweak trails, allowing an attack on 12-round MANTIS_6. Later, several works [BDHBNP23, WNL+23] focused on the cryptanalysis of SPEEDY with multiple differentials, and finally broke SPEEDY-7-192 [BDG+25c, BN25]. This line of research shows that multiple differential attacks are particularly effective for low-latency ciphers, especially when it has the data limitation.

Theoretical Frameworks on Fixed-key Behaviors. There are several well-established frameworks that explain fixed-key differential phenomena. To list a few, commutative cryptanalysis [Wag04] models persistent truncated differentials using small affine commutants. Recent works [BFL+23, BBF+25] broadened this line of analysis by studying larger classes of affine or structural commutation phenomena and showing that such commuting structures can systematically generate exploitable differential patterns in a wider range of block-cipher constructions. Invariant-subspace methods [GJN+16] characterize cases where all valid transitions remain inside a linear invariant space, leading to affine weak-key sets whose dimension matches the dimension of the corresponding output-difference space. The theory was generalized by [Bey18], which interprets invariant subspaces as eigenvectors of correlation matrices. Quasi-differentials [BR22] give a general spectral treatment of fixed-key probabilities without relying on affine structure. Plateau characteristics [DR07, CF25] offer a general theory describing when fixed-key differential characteristics retain a constant probability over an affine region of the expanded key space, and how to cluster them into truncated differentials.

Our Contributions. It is well acknowledged that the common combination of small S-boxes and a simple MixColumn layer in lightweight designs often induces strong correlations, which can lead to weak key structures or even invalid trails, as seen from prior works [CLN+17, SWW18, LZS+20, BR22, PT22, CWWH25, BDG25b, NGJE25]. In contrast, Blink adopts a long-key design, making the weak-key behaviors easier to analyze. In this work, we present the first third-party cryptanalysis of Blink, with a focus on weak tweak-key distinguishers and key recovery attacks. The main results are shown in Table 1.

In our analysis, we leverage the idea of Superboxes [GP10, LMR+09] introduced to rebound attacks [MRST09] for hash function cryptanalysis, with the motivation that the Superbox structure naturally supports local clustering and integrates well with the automatic search. As such, we start by providing a simplified theoretic analysis on the clustering of differentials in Superbox and the associated weak-key conditions. Our theory targets only the most relevant case in our analysis where the value spaces can be represented as an affine subspace, tracks value correlations through the MixKey layer, and expresses weak-key spaces as intersections of affine cosets. While limited in coverage, it captures the accurate estimation of differential probability under weak keys, clustering of trails, and the resulting weak-key distributions.

Table 1: Summary of weak tweak-key distinguishers and attacks on Blink. The "Setting" column lists WT (weak tweak-key), ST (Single tweak-key) and MT (Multiple tweak-key).

Cipher (Data/Time Claim)	**Setting**	**Round**	**Data**	**Time**	**Prob.**	**Type**	**Source**
Blink-64 (D: 2^{56}/ T: 2^{128})	ST	10/14	$> 2^{56}$	–	2^{-100}	Differential†	[WHZ+25]
	WT*	10/14	$2^{50.42}$	–	$2^{-50.42}$	Differential	Section 4.2
	WT*	13/14	2^{56}	$2^{114.02}$	–	Key recovery	Section 5
Blink-128 (D: 2^{80}/ T: 2^{128})	ST	10/16	$> 2^{80}$	–	2^{-128}	Differential†	[WHZ+25]
	WT⋆	10/16	$2^{68.83}$	–	$2^{-68.83}$	Differential	Section 4.3
	WT⋆	12/16	$2^{69.83}$	–	$2^{-96.83}$	Multiple Diff.	Section 4.3
	WT⋆	13/16	$2^{85.52}$	$2^{119.43}$	–	Key recovery‡	Section A

† The designers of Blink evaluated the lower bounds of active S-boxes [WHZ+25, Table 3], based on which the designers claim [WHZ+25, Section 7.1] that under the data limitation 2^{56} and 2^{80}, it is highly unlikely to distinguish 10 rounds of both Blink-64 and Blink-128. The designers provide a comprehensive analysis of other attacks, including linear, impossible differential, integral, and invariant subspace attacks. Among which, differential and linear seem to be the most effective attacks.

‡ For Blink-64, we can mount key recovery attacks to recover the full 448-bit key while still within the claimed data and time complexity. However, for Blink-128, as the data complexity is slightly higher than 2^{80}, we only provide a simple key recovery attack to recover 72 bits out of the full 1024-bit key.

* The attacks work for up to $2^{448-6} = 2^{442}$ weak keys, *e.g.*, when $t = 0$.

⋆ The attacks works for up to $2^{1024-14} = 2^{1010}$ weak keys, *e.g.*, when $t = 0$.

Building upon our analysis on Blink Superbox, we introduce an intuitive pattern-based search model to identify weak tweak-key differential distinguishers for Blink. Under the weak tweak-key setting, we obtain a 10-round distinguisher for Blink-64 with probability $2^{-50.42}$ for up to $2^{448-6} = 2^{442}$ weak keys, and a 10-round distinguisher for Blink-128 with probability $2^{-68.83}$ for up to $2^{448-6} = 2^{442}$ weak keys, depending on the tweak value. If the adversary is capable of setting the tweak value, a convenient choice that maximizes the weak-key space is $t = 0$. We argue the distinguishers we identify are dominant, with experimentally verification of the key steps of the 10-round Blink-64 trail. For Blink-128, we further extend the 10-round trail into a valid 12-round multiple-differential distinguisher with probability $2^{-96.83}$. All these distinguishers operate within the data limits stated by the designers. We also mount a 13-round weak tweak-key key-recovery attack on Blink-64 that recovers the full 448-bit master key with time complexity $2^{114.02}$ and 2^{56} chosen plaintexts, which works within the designers' claimed bounds. Our attack leverages the 10-round Superbox-based differential trail by decomposing the key recovery into four phases and applying effective filtering conditions from the four Superboxes inside the distinguisher, which let us progressively reduce the key space until the entire 448-bit key is recovered.

These findings provide additional structural insights into the boundary behaviors of Blink's tweak-key interaction, while remaining consistent with the security of the full-round design. Due to our analysis, the designers of Blink have prohibited the use of the all-zero tweak in their final version [WHZ+25], which can prevent the structural collapse against weak tweak-key differential attacks and increase the security margin of Blink ciphers further. The observed phenomena under constrained tweak-key configurations may also guide future refinements of tweakable block cipher constructions.

Discussion with Related Works. Our theoretical analysis based on Superboxes can be viewed as a concrete and problem-driven specialization of the theoretical frameworks we reviewed above. Our goal is not to extend these frameworks, but simply to instantiate the theories in a concrete and specialized case of Superbox. This yields a deliberately

lightweight model, far more limited in scope than the generic theories, but offering a clear view of the correlations inside Superbox, weak key conditions, fixed-key probabilities, and the clustering behaviors that arise in the analysis of Blink. Providing such a practical description that is easy to follow and verify was our primary motivation.

Organization. The paper is organized as follows. Section 2 introduces the specifications of Blink and S-box properties. In Section 3, we provide a theoretical analysis of Superbox behaviors of Blink, followed by a summary of relevant active patterns and corresponding weak-key distribution, and a motivation of our searching strategy. Section 4 describes our search model and weak key distinguishers. Finally, in Section 5 and Section A, we mount key recovery attacks on Blink-64 and Blink-128.

2 Overview of Blink

2.1 Blink Construction

An internal state of Blink contains n-bits, and is organized as a $4 \times (n/16)$ array of nibbles. Blink follows the reflection structure, allowing for identical encryption and decryption circuits, which is as shown in Figure 1.

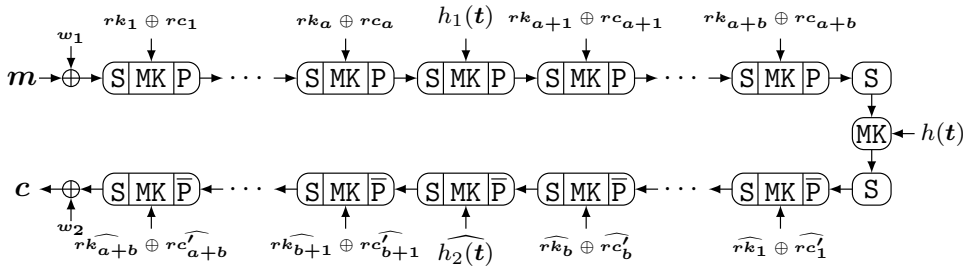


Figure 1: The Construction of Blink [WHZ+25].

Round structure. One forward round $\mathtt{R}$ and one inverse round of $\overline{\mathtt{R}}$ of Blink consists of the sequences respectively

$$\mathtt{R} = \mathtt{P} \circ \mathtt{MK} \circ \mathtt{S}, \qquad \overline{\mathtt{R}} = \overline{\mathtt{P}} \circ \mathtt{MK} \circ \mathtt{S}.$$

$\mathtt{S}$ denotes the S-box layer. The forward MixKey layer is $\mathtt{MK} = M(\cdot) + rk + rc$, and the inverse MixKey layer is $\mathtt{MK} = M(\cdot) + \widehat{rk} + \widehat{rc}$, with $\widehat{rk} = M(rk)$ and $\widehat{rc} = M(rc)$. $\mathtt{P}$ is a fixed permutation of the nibble positions, and $\overline{\mathtt{P}}$ is the inverse of $\mathtt{P}$.

The inverse MixKey layer can be equivalently expressed as $M(\cdot + rk + rc)$. Thus, we can equivalently write $\mathtt{R}$ and $\overline{\mathtt{R}}$ as:

$$\mathtt{R} = \mathtt{P} \circ \mathtt{ARC} \circ \mathtt{ARK} \circ \mathtt{MC} \circ \mathtt{S}, \qquad \overline{\mathtt{R}} = \mathtt{S} \circ \mathtt{MC} \circ \mathtt{ARK} \circ \mathtt{ARC} \circ \overline{\mathtt{P}},$$

where $\mathtt{ARC}$ and $\mathtt{ARK}$ refers to the XOR of round key and constant respectively.

S-box layer. The S-box layer $\mathtt{S}$ applies a 4-bit $Sbox$ for every nibble. $Sbox$ is involutory with differential uniformity 4.

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$Sbox(x)$	1	0	9	3	8	5	e	7	4	2	c	b	a	f	6	d

MixColumn layer. Blink uses the Midori Mixcolumn matrix $M = J - I$, where J is the all-one matrix and I is the identity matrix. Note that M is almost-MDS, involutory ($M = M^{-1}$), linear and symmetric ($M = M^{\intercal}$).

Permutation layer. P is a fixed nibble-wise permutation as follows.

$$\begin{aligned} n = 64: \ & P = [0, 5, 11, 10, 1, 6, 4, 13, 2, 12, 9, 15, 3, 7, 14, 8], \\ n = 128: \ & P = [5, 12, 4, 1, 17, 9, 10, 16, 28, 14, 21, 22, 11, 27, 8, 13, \\ & \quad 2, 25, 18, 3, 30, 6, 19, 20, 0, 23, 24, 31, 7, 15, 29, 26]. \end{aligned}$$

Toeplitz-based hash functions h_1 and h_2. Let τ be the tweak length. The hash function h_T is defined as $h_T(\boldsymbol{t}) = T \cdot \boldsymbol{t}$, where T is an $n \times \tau$ Toeplitz matrix and $\boldsymbol{t}$ represents the tweak as vector of length τ. If we write $\boldsymbol{k} = (k_0, \ldots, k_{n+\tau-2}) \in \{0,1\}^{n+\tau-1}$, the corresponding T has the form

$$T = \begin{pmatrix} k_{n-1} & k_n & \cdots & k_{n+\tau-2} \\ k_{n-2} & k_{n-1} & \cdots & k_{n+\tau-3} \\ \vdots & \vdots & \ddots & \vdots \\ k_0 & k_1 & \cdots & k_{\tau-1} \end{pmatrix}.$$

In Blink, two independent keys $\boldsymbol{k}_1, \boldsymbol{k}_2 \in \{0,1\}^{n+\tau-1}$ define two independent Toeplitz matrices T_1, T_2 and the three pair-wise independent hash functions are

$$h_1(\boldsymbol{t}) = T_1 \cdot \boldsymbol{t}, \quad h_2(\boldsymbol{t}) = T_2 \cdot \boldsymbol{t}, \quad h(\boldsymbol{t}) = (T_1 + T_2) \cdot \boldsymbol{t}.$$

Formation of Superboxes. Note that in Blink, Midori, MANTIS, we can always reorganize the P to form Superboxes [LMR+09, GP10]. A *Superbox* consists a sequence of S ∘ M ∘ S, where the operators and intermediate states are all vectors of the same length of a column, denoted by t.

$$\texttt{P} \circ \texttt{MK} \circ \texttt{S} \circ \texttt{P} \circ \texttt{MK} \circ \texttt{S} \iff \texttt{P} \circ \texttt{M} \circ \texttt{P} \circ \textit{Superbox}$$

2.2 Differential Properties of the Blink S-box

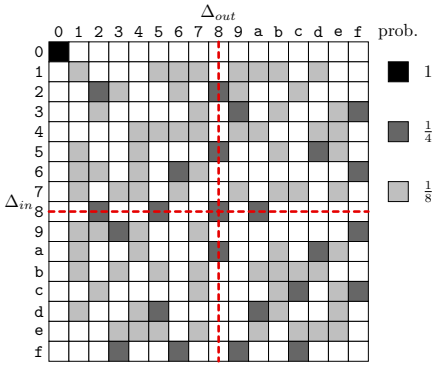


Figure 2: Differential distribution table of Blink S-box.

Blink uses an involutive 4-bit S-box, whose differential distribution table (DDT) is shown in Figure 2. In this paper, we focus on the differential properties of this S-box. The DDT shows that for 0x8 and 0xf, all four possible output differences occur with equal probability 2^{-2}. Moreover, for the difference 0x8, its input and output difference sets are

both the same set $\{\texttt{0x2}, \texttt{0x5}, \texttt{0x8}, \texttt{0xa}\}$, this is due to the involutive property and shown by the red dashed line in Figure 2. This property of the difference `0x8` makes it a desired choice when considering the propagation through Superbox, as it can include more possible differential transitions with higher probability. So, we will focus on the propagation of the difference `0x8` through Superbox and other operations of Blink in the rest of the paper.

3 Superbox Analysis of Blink

By default, + denote the addition over $\mathbb{F}_2^n$, *i.e.*, the XOR operation. In addition, we summarize [Rom10, Theroem 15.5, 15.6] as Theorem 1 and define difference vector (or equivalently as active mask) in Definition 1.

Theorem 1 (Intersection of Affine Subspaces)**.** *Let U and V be vector spaces over some arbitrary field, $a_0 \in U$ and $b_0 \in V$ be constants, $A = a_0 + U$ and $B = b_0 + V$ be affine subspaces. Then exactly one of the following holds: either $A \cap B = \varnothing$, or $A \cap B$ is an affine subspace with direction $U \cap V$ and*

$$\dim(A \cap B) \;=\; \dim(U \cap V) \;=\; \dim(U) + \dim(V) - \dim(U + V).$$

Definition 1 (Difference Vector [MWGP11])**.** Consider a vector Δ consisting of $n/4$ nibbles $\Delta = (\Delta_0, \Delta_1, ..., \Delta_{n/4-1})$. Then, the difference vector, also known as the active mask, of Δ is defined as

$$\delta \in \{0,1\}^{n/4} : \; \delta_i = \begin{cases} 0, & \text{if } \Delta_i = 0, \\ 1, & \text{otherwise.} \end{cases}$$

Example 1. Let the input difference state of Blink-64 (with 16 nibbles) be $\Delta = (\texttt{0x0}, \texttt{0x8}, \texttt{0x8}, \texttt{0x8}, \texttt{0x0}, \texttt{0x8}, \texttt{0x8}, \texttt{0x8}, \texttt{0x0}, \texttt{0x8}, \texttt{0x0}, \texttt{0x8}, \texttt{0x0}, \texttt{0x0}, \texttt{0x8}, \texttt{0x0})$, then its difference vector is $\delta = (0,1,1,1,0,1,1,1,0,1,0,1,0,0,1,0)$.

3.1 Theoretical Overview of Weak-key Behaviors in Superbox

Consider a *Superbox* layer with length t, key k, and the operations

$$x = Sbox(w), \quad y = Mx + k, \quad z = Sbox(y),$$

where M is an invertible matrix for the MixColumn operator, and $w, x, y, z, k \in (\mathbb{F}_2^n)^t$ are intermediate vectors for the *Superbox*. We additionally assume M is linear.

Now consider a differential trail Q for the *Superbox*. We denote the active masks of Q at x and y (equivalent to those at w and z respectively) by $a, b \in \{0,1\}^t$. We additionally denote the differences at w, x, y, z respectively by $\delta_w, \delta_x, \delta_y, \delta_z \in (\mathbb{F}_2^n)^t$. All constraints imposed by Q over the *Superbox* can be expressed equivalently on the space of (x, y, k):

$$x_i \in \mathcal{X}_i, \quad y_i = (Mx + k)_i \in \mathcal{Y}_i, \quad \forall\, i \in \{0, \ldots, t-1\}\,.$$

where $\mathcal{X}_i \subset \mathbb{F}_2^n$ denotes the output value space induced by the differential $\delta_w \to \delta_x$ if $a_i = 1$, and the full space $\mathbb{F}_2^n$ otherwise. Similarly, $\mathcal{Y}_i \subset \mathbb{F}_2^n$ denotes the input value space induced by $\delta_y \to \delta_z$ if $b_i = 1$, and the full space $\mathbb{F}_2^n$ otherwise. For convenience, we define:

$$\mathcal{X} := \prod_i \mathcal{X}_i, \qquad \mathcal{Y} := \prod_i \mathcal{Y}_i.$$

Weak Key Analysis with Affine Subspaces. We define the value space of x that is valid under Q for some key k:

$$\mathcal{X}(k) := \{\, x \in \mathcal{X} \mid Mx + k \in \mathcal{Y} \,\}.$$

We focus our analysis on the case where $\mathcal{X}_i$ and $\mathcal{Y}_i$ are either an affine subspace of $\mathbb{F}_2^n$ or the full space $\mathbb{F}_2^n$. Hence we may write $\mathcal{X}_i = u_i + U_i$ and $\mathcal{Y}_i = v_i + V_i$, where $U_i, V_i \subseteq \mathbb{F}_2^n$ are linear subspaces and $u_i, v_i \in \mathbb{F}_2^n$. Note that when $a_i = 0$, $U_i = \mathbb{F}_2^n$ and $u_i = 0$, and similarly when $b_i = 0$, $V_i = \mathbb{F}_2^n$ and $v_i = 0$. Thus the value spaces

$$\mathcal{X} = u + U, \qquad \mathcal{Y} = v + V,$$

are cosets of some subspaces $U, V \subseteq (\mathbb{F}_2^n)^t$ with offset vectors u, v. Thus, we have the value space $\mathcal{X}(k)$ expressed as an intersection of two sets:

$$\mathcal{X}(k) = (u + U) \cap M^{-1}(v + V + k).$$

Since M is linear,

$$M^{-1}(v + V + k) = M^{-1}(v + k) + M^{-1}V.$$

Hence $\mathcal{X}(k)$ is an intersection of two affine subspaces:

$$\mathcal{X}(k) = (u + U) \cap \big(M^{-1}(v + k) + M^{-1}V\big).$$

For readability, we additionally define:

$$W = U + M^{-1}V, \quad S = U \cap M^{-1}V, \quad L(k) := u + M^{-1}(v + k).$$

We have from Theorem 1

$$\mathcal{X}(k) = \begin{cases} \varnothing, & \text{if } L(k) \notin W, \\ x^* + S, & \text{if } L(k) \in W, \end{cases}$$

where x^* is any fixed element of the nonempty intersection $(u+U) \cap \big(M^{-1}(v+k) + M^{-1}V\big)$.

In other words, the condition $L(k) \in W$ marks a weak key space. If the condition is not met, the differential Q is impossible. If the condition is met, or k is a weak key for Q, the probability for Q under k is

$$\Pr(Q \mid k) = \frac{|\mathcal{X}(k)|}{2^{nt}} = 2^{\dim(S) - nt}.$$

Additionally from Theorem 1, $\dim(S) = \dim(U) + \dim(V) - \dim(W)$. Thus the probability of Q depends only on U and V (or equivalently $\mathcal{X}$ and $\mathcal{Y}$), and is independent of the specific value of k.

Weak Key Distribution for Clustering Differentials. Building on the fact that a weak-key differential has constant probability for all weak keys, we now consider the clustering of differentials and the resulting key distribution. Let the cluster of differentials be $\{Q^{(j)}\}_{j \in C}$, where each $Q^{(j)}$ induces its own affine constraints: $\mathcal{X}^{(j)} = u^{(j)} + U^{(j)}$, $\mathcal{Y}^{(j)} = v^{(j)} + V^{(j)}$. Each differential has its own weak key space $\mathcal{K}^{(j)}$, and the global weak-key set of the cluster is simply the union of all weak key spaces:

$$\mathcal{K}^{(j)} := \{\, k \mid L^{(j)}(k) \in W^{(j)} \,\}, \qquad \mathcal{K}_{\text{cluster}} := \bigcup_{j \in C} \mathcal{K}^{(j)}.$$

where $W^{(j)} = U^{(j)} + M^{-1}V^{(j)}$ and $L^{(j)}(k) = u^{(j)} + M^{-1}(v^{(j)} + k)$. Each $\mathcal{K}^{(j)}$ is either empty or an affine subspace of the key space, thus the cluster forms a union of affine

subspaces. These subspaces together determines a *weak key distribution*, *i.e.*, for different weak keys, the number of valid differentials under this key differs, thus the probability for the clustered differentials differs as well. For each key k, the clustered probability can be expressed as:

$$\Pr(Q_{\text{cluster}} \mid k) = 2^{\dim(S)-nt} \cdot \left|\{j \in C \mid k \in \mathcal{K}^{(j)}\}\right|.$$

A concrete case study for Blink will be given in Section 3.2.

3.2 Concrete Examples in Blink

We identify a special type of differential for a Blink S-box: $\texttt{0x8} \to \alpha$ with probability 2^{-2}, where $\alpha \in \{\texttt{0x2}, \texttt{0x5}, \texttt{0x8}, \texttt{0xa}\}$. To simplify the notations, in this subsection, we use V_α to denote the output value space $\texttt{0x8} \to \alpha$. We have that V_α is an affine subspace of D_α with offset c_α. The details are given in Table 2. We noticed that there are only two possible D_α, thus we define $D_1 = D_{\texttt{0x2}} = D_{\texttt{0xa}} = \{\texttt{0x0}, \texttt{0x2}, \texttt{0x8}, \texttt{0xa}\}$ and $D_2 = D_{\texttt{0x5}} = D_{\texttt{0x8}} = \{\texttt{0x0}, \texttt{0x8}, \texttt{0x5}, \texttt{0xd}\}$. We note that $D_1 + D_2 = \{\texttt{0x0}, \texttt{0x2}, \texttt{0x5}, \texttt{0x7}, \texttt{0x8}, \texttt{0xa}, \texttt{0xd}, \texttt{0xf}\}$ and $D_1 \cap D_2 = \{\texttt{0x0}, \texttt{0x8}\}$. We now give example analyses regarding relevant active patterns for *Superbox* differentials with fixed input and output difference $\texttt{0x8} \cdot a \to \texttt{0x8} \cdot b$, where $a, b \in \{0,1\}^4$ are the active masks as defined above.

Table 2: Selected $\delta_{in} \to \delta_{out}$ and corresponding V_{out}, D_{out}, c_α for Blink

δ_{in}	α	p	V_α	D_α	c_α
0x8	0x2	2^{-2}	{0x0, 0x8, 0x2, 0xa}	{0x0, 0x8, 0x2, 0xa}	0x0
0x8	0x5	2^{-2}	{0x1, 0x9, 0x4, 0xc}	{0x0, 0x8, 0xd, 0x5}	0x1
0x8	0x8	2^{-2}	{0x3, 0xe, 0xb, 0x6}	{0x0, 0x8, 0x5, 0xd}	0x3
0x8	0xa	2^{-2}	{0x5, 0x7, 0xf, 0xd}	{0x0, 0x8, 0x2, 0xa}	0x5

Example 1: 1-to-3 active. In this case, without loss of generality, let $a = \{0,0,1,0\}$ and $b = \{1,1,0,1\}$. The differential pattern must satisfy

$$\{\texttt{0x0}, \texttt{0x0}, \texttt{0x8}, \texttt{0x0}\} \to \{\texttt{0x0}, \texttt{0x0}, \alpha, \texttt{0x0}\} \to \{\alpha, \alpha, \texttt{0x0}, \alpha\} \to \{\texttt{0x8}, \texttt{0x8}, \texttt{0x0}, \texttt{0x8}\},$$

where $\alpha \in \{\texttt{0x2}, \texttt{0x5}, \texttt{0x8}, \texttt{0xa}\}$. Thus

$$\mathcal{X} = \mathbb{F}_2^n \times V_\alpha \times \mathbb{F}_2^n \times \mathbb{F}_2^n, \qquad \mathcal{Y} = V_\alpha \times \mathbb{F}_2^n \times V_\alpha \times V_\alpha$$

are cosets of

$$U = \mathbb{F}_2^n \times \mathbb{F}_2^n \times D_\alpha \times \mathbb{F}_2^n, \qquad V = D_\alpha \times D_\alpha \times \mathbb{F}_2^n \times D_\alpha,$$

with offsets

$$u = [0, 0, c_\alpha, 0], \qquad v = [c_\alpha, c_\alpha, 0, c_\alpha].$$

As $M = M^{-1}$ and $u = Mv$, shifting away the offsets reduces the weak key condition to

$$L(k) = Mk \in W, \qquad W := U + MV.$$

To analyze W rigorously, we work in the dual. We have

$$U^\perp = \{0\} \times \{0\} \times D_\alpha^\perp \times \{0\}, \qquad V^\perp = D_\alpha^\perp \times D_\alpha^\perp \times \{0\} \times D_\alpha^\perp,$$

and since M is invertible and symmetric in Blink, we have

$$W^\perp = U^\perp \cap (MV)^\perp = U^\perp \cap MV^\perp = U^\perp.$$

Thus, we have $\dim(W^\perp) = \dim(U^\perp) = 2$ and $\dim(W) = 4n - \dim(W^\perp) = 16 - 2 = 14$. We have the weak key condition simply as

$$(Mk)_2 = k_0 + k_1 + k_3 \in D_\alpha.$$

We then proceed to compute the probability of Q holding given a weak key, $\Pr(Q \mid k)$. As $\dim(S) = \dim(U) + \dim(V) - \dim(W)$, and we have $\dim(V) = 10$, $\dim(U) = \dim(W) = 14$, we have

$$\Pr(Q \mid k) = 2^{14+10-14-4\cdot 4} = 2^{-6}.$$

From the simplified weak key condition $k_0 + k_1 + k_3 \in D_\alpha$, we see that when $k_0 + k_1 + k_3 \in D_1 \cap D_2 = \{\texttt{0x0}, \texttt{0x8}\}$, all 4 differentials are valid. Otherwise, when $k_0 + k_1 + k_3 \in (D_1 \cup D_2) \setminus (D_1 \cap D_2) = \{\texttt{0x2}, \texttt{0xa}, \texttt{0x5}, \texttt{0xd}\}$, exactly 2 differentials are valid. Thus, we have a weak key distribution:

$$\Pr(Q \mid k) = \begin{cases} 2^{-4} & k_0 + k_1 + k_3 \in \{\texttt{0x0}, \texttt{0x8}\}, \\ 2^{-5} & k_0 + k_1 + k_3 \in \{\texttt{0x2}, \texttt{0xa}, \texttt{0x5}, \texttt{0xd}\}, \\ 0 & \text{otherwise.} \end{cases}$$

Example 2: 3-to-3 active. Let $a = \{1, 1, 1, 0\}$ and $b = \{1, 1, 1, 0\}$ without loss of generality. The differential pattern must satisfy

$$\{\texttt{0x8}, \texttt{0x8}, \texttt{0x8}, \texttt{0x0}\} \to \{\alpha, \beta, \gamma, \texttt{0x0}\} \to \{\alpha, \beta, \gamma, \texttt{0x0}\} \to \{\texttt{0x8}, \texttt{0x8}, \texttt{0x8}, \texttt{0x0}\},$$

where $\alpha \neq \beta \neq \gamma \in \{\texttt{0x2}, \texttt{0x8}, \texttt{0xa}\}$. Similar to above, we have U, V, u, v defined as

$$U = V = D_\alpha \times D_\beta \times D_\gamma \times \mathbb{F}_2^n, \qquad u = v = [c_\alpha, c_\beta, c_\gamma, 0].$$

As in Blink we have $M = M^{-1}$ and $M = J - I$, where J is the all-one matrix. Shifting away the offsets again yields the weak key condition

$$L(k) = Ju + Mk \in W, \qquad W := U + MV.$$

We now derive W via its dual. We have

$$U^\perp = D_\alpha^\perp \times D_\beta^\perp \times D_\gamma^\perp \times \{0\} = V^\perp.$$

We can express $U^\perp$ and $(MV)^\perp = MV^\perp$ as follows

$$\begin{aligned} U^\perp &= \left\{(p, q, r, 0) : p \in D_\alpha^\perp,\ q \in D_\beta^\perp,\ r \in D_\gamma^\perp.\right\} \\ MV^\perp &= \left\{(q + r,\ p + r,\ p + q,\ p + q + r) : p \in D_\alpha^\perp,\ q \in D_\beta^\perp,\ r \in D_\gamma^\perp\right\}. \end{aligned}$$

It is easy to see $W^\perp = U^\perp \cap (MV)^\perp$ can be expressed as

$$W^\perp = \left\{(p,\ q,\ p + q,\ 0)\ :\ p \in D_\alpha^\perp,\ q \in D_\beta^\perp,\ p + q \in D_\gamma^\perp\right\}.$$

As two of $D_\alpha, D_\beta, D_\gamma$ equal D_1 and one equals D_2, we can easily check $\dim(W^\perp) = 3$ and $\dim(W) = 4n - \dim(W^\perp) = 16 - 3 = 13$.

We notice that $w \in W$ if and only if $w_0 + w_1 \in D_\alpha + D_\beta$ and $w_0 + w_2 \in D_\alpha + D_\gamma$. Substituting $w = L(k) = Ju + Mk$, as we have $(Ju + Mk)_0 + (Ju + Mk)_1 = k_0 + k_1$ and $(Ju + Mk)_0 + (Ju + Mk)_2 = k_0 + k_2$, we obtain straightforward weak key conditions as:

$$k_0 + k_1 \in D_\alpha + D_\beta, \qquad k_0 + k_2 \in D_\alpha + D_\gamma.$$

As two of D_α, D_α, D_γ equals D_1 and one equals D_2, $w_0 + w_1 \in D_\alpha + D_\beta$ and $w_0 + w_2 \in D_\alpha + D_\gamma$ results in $\dim(W) = 13$. Under a weak key, the differential holds with probability

$$\Pr(Q \mid k) = 2^{10+10-13-4\cdot 4} = 2^{-9}.$$

We next discuss the clustering of differentials. We see that when $k_0 + k_1,\ k_0 + k_2 \in D_1$, all 6 differentials are valid. When $(k_0 + k_1,\ k_0 + k_2) \in (D_1, \texttt{0x5} + D_1)$ or $(\texttt{0x5} + D_1, D_1)$ or $(\texttt{0x5} + D_1, \texttt{0x5} + D_1)$, exactly 2 differentials are valid. Thus, we have a weak key distribution:

$$\Pr(Q \mid k) = \begin{cases} 3 \cdot 2^{-8} & (k_{i_0} + k_{i_1}, k_{i_0} + k_{i_2}) \in (D_1)^2 \\ 2^{-8} & (k_{i_0}, k_{i_1}) \in (D_1 + D_2)^2 \setminus (D_1)^2 \\ 0 & \text{otherwise} \end{cases}$$

Summary of Clusters and Respective Weak Key Distribution. We have summarized the clustering effect for selected active pattern in forward direction and the corresponding weak key distribution in Table 3.

Table 3: Weak key distributions for the 1-to-3, 3-to-1, 2-to-2, 3-to-3, and 4-to-2 active patterns. Here i_* denote the active indices in the output, j_* denote the active indices in the input. The subspaces are: $D_1 = \{\texttt{0x0}, \texttt{0x2}, \texttt{0x8}, \texttt{0xa}\}$, $D_1 + D_2 = \{\texttt{0x0}, \texttt{0x2}, \texttt{0x5}, \texttt{0x7}, \texttt{0x8}, \texttt{0xa}, \texttt{0xd}, \texttt{0xf}\}$, $(D_1 + D_2)^2 \setminus (D_1)^2$ stands for the set of tuples whose elements are both in $D_1 + D_2$ but not both in D_1.

Pattern	$\Pr(k)$	$\Pr(Q \mid k)$	Key condition on k
1-to-3	1/8	2^{-4}	$k_{i_0} + k_{i_1} + k_{i_2} \in \{\texttt{0x0}, \texttt{0x8}\}$
	1/4	2^{-5}	$k_{i_0} + k_{i_1} + k_{i_2} \in \{\texttt{0x2}, \texttt{0x5}, \texttt{0xd}, \texttt{0xa}\}$
	5/8	0	otherwise
3-to-1	1/8	2^{-4}	$k_{i_0} \in \{\texttt{0x0}, \texttt{0x8}\}$
	1/4	2^{-5}	$k_{i_0} \in \{\texttt{0x2}, \texttt{0x5}, \texttt{0xd}, \texttt{0xa}\}$
	5/8	0	otherwise
2-to-2	1/8	2^{-4}	$k_{i_0} + k_{i_1} \in \{\texttt{0x0}, \texttt{0x8}\}$
	1/4	2^{-5}	$k_{i_0} + k_{i_1} \in \{\texttt{0x2}, \texttt{0x5}, \texttt{0xd}, \texttt{0xa}\}$
	5/8	0	otherwise
3-to-3	1/16	$3 \cdot 2^{-8}$	$(k_{i_0} + k_{i_1},\ k_{i_0} + k_{i_2}) \in (D_1)^2$
	3/16	2^{-8}	$(k_{i_0} + k_{i_1},\ k_{i_0} + k_{i_2}) \in (D_1 + D_2)^2 \setminus (D_1)^2$
	3/4	0	otherwise
4-to-2	1/16	$3 \cdot 2^{-8}$	$(k_{i_0},\ k_{i_1}) \in (D_1)^2$
	3/16	2^{-8}	$(k_{i_0},\ k_{i_1}) \in (D_1 + D_2)^2 \setminus (D_1)^2$
	3/4	0	otherwise
2-to-4	1/16	$3 \cdot 2^{-8}$	$((Mk)_{j_0},\ (Mk)_{j_1}) \in (D_1)^2$
	3/16	2^{-8}	$((Mk)_{j_0},\ (Mk)_{j_1}) \in (D_1 + D_2)^2 \setminus (D_1)^2$
	3/4	0	otherwise

3.3 Superbox Patterns and Chain-like Trails for Blink

As mentioned in Section 2.2, we focus on Superboxes whose active input and output cells are all fixed to the difference `0x8`, however, we also tested the other three candidates $\{\texttt{0x2}, \texttt{0x5}, \texttt{0xa}\}$, but they both have lower probabilities, and the gap will widen rapidly as the trail propagates through additional rounds, making them non-competitive compared to the `0x8` based patterns. This observation is further confirmed by our later experiments on the Blink-64 distinguisher, which also show that `0x8` based patterns dominate the multi-round differential behavior. Thanks to the swap between the `S` and `P` operations, each Superbox can be analyzed column-wise and independently, and therefore we only

need to consider the propagation patterns listed in Table 3. One exception is the 4-to-4 pattern: we exclude it because P and $\overline{\mathtt{P}}$ split the four active cells of an input column into four different output columns, thereby activating more columns and significantly reducing the overall trail probability. Now, considering the maximum probabilities of the Superbox patterns in Table 3, for each active S-box, it has an average probability 2^{-1}, *e.g.*, 1-to-3, 3-to-1, and 2-to-2 patterns all with probability 2^{-4} for 4 active S-boxes; 3-to-3, 4-to-2, and 2-to-4 patterns all with probability $3 \cdot 2^{-8}$ for 6 active S-boxes.

With the above Superbox analysis, the core non-linear behavior of Blink can be well captured. We then naturally connect these Superboxes through the surrounding linear layer: P (or $\overline{\mathtt{P}}$) first splits the four cells of each column into four different columns, after which M mixes the nibbles within each column, and finally another P (or $\overline{\mathtt{P}}$) is applied. This structure ensures strong diffusion between the input and output of adjacent Superboxes. Moreover, since we restrict ourselves to Superbox patterns whose active input and output cells are always fixed to the difference 0x8, the propagation through the linear layer becomes *deterministic*, which greatly simplifies the search model. As a result, the overall multi-round differential trail constructed from these Superboxes forms a clean chain-like structure, each Superbox accounts for the non-linear transitions and weak-key conditions, while the linear layers deterministically connect their differences across rounds.

4 Weak Tweak-Key Differential Distinguishers of Blink

In this section, with the analysis of Superbox propagation in the previous section, we will simply provide the Superbox based automatic model to search the weak tweak-key differential distinguisher for Blink ciphers. With this, for Blink-64, we find a 10-round weak tweak-key differential trail with a maximum probability of $2^{-50.42}$ by setting 6-bit conditions on key space. For Blink-128, we find a 10-round weak tweak-key differential trail with a maximum probability of $2^{-68.48}$, which can be extended two more rounds to a 12-round multiple differential distinguisher with probability $2^{-96.84}$ (the corresponding random probability are 2^{-100}). As the designers of Blink [WHZ+25, Section 7.1] claim, "we conclude that when the data does not exceed 2^{56}, it is highly unlikely to distinguish 10 rounds of Blink-64", our two identified 10-round distinguishers remain within the data limitations claimed by the designers, *i.e.*, 2^{56} for Blink-64 and 2^{80} for Blink-128, and even a 12-round multiple differential distinguisher is mounted for Blink-128.

4.1 Simple Superbox Pattern based Differential Search Model

According to analyses in Section 3.2 and Section 3.3, it can be observed that there are not too many choices when considering the Superbox based propagation of the differential trail for Blink. Thus, we can simply model these Superbox patterns into the search model by using the mature MILP or SAT/SMT automatic search methods. For example, we can directly follow Mouha *et al.* 's [MWGP11] automatic search model of the differential pattern, *i.e.*, we consider truncated differences with every nibble in our search model, which has either a zero or a non-zero difference as described in Definition 1. Considering that how to search for the distinguisher is not a difficult task for our cryptanalysis on Blink ciphers, in the following, we just briefly provide the difference vector propagations through different operations to capture this pattern-based searching. To avoid distractions, the detailed MILP or SAT/SMT model constraints will not be discussed here.

Difference Vector Propagation through Superbox. Let $\delta x = (\delta x_0, \delta x_1, \delta x_2, \delta x_3)$ and $\delta y = (\delta y_0, \delta y_1, \delta y_2, \delta y_3)$ be one-column difference vectors of input and output of the Superbox respectively. Then, according to Table 3, we only have several possible patterns to propagate through the Superbox, which are given in Table 4 below. Based on these

patterns of Superbox, we can model the corresponding propagation rules by using the convex hull representation for MILP [SHW+14] or Product-of-Sum representation for both MILP and SAT/SMT [AST+17].

Table 4: Propagation of difference vectors through Superbox.

Pattern	δx	δy	Pattern	δx	δy
0-to-0	(0,0,0,0)	(0,0,0,0)			
1-to-3	(0,0,0,1) (0,0,1,0) (0,1,0,0) (1,0,0,0)	(1,1,1,0) (1,1,0,1) (1,0,1,1) (0,1,1,1)	3-to-1	(1,1,1,0) (1,1,0,1) (1,0,1,1) (0,1,1,1)	(0,0,0,1) (0,0,1,0) (0,1,0,0) (1,0,0,0)
2-to-2	(0,0,1,1) (0,1,1,0) (1,1,0,0)	(0,0,1,1) (0,1,1,0) (1,1,0,0)	2-to-2	(0,1,0,1) (1,0,1,0) (1,0,0,1)	(0,1,0,1) (1,0,1,0) (1,0,0,1)
3-to-3	(1,1,1,0)	(1,1,1,0) (1,1,0,1) (1,0,1,1) (0,1,1,1)	3-to-3	(1,1,0,1)	(1,1,1,0) (1,1,0,1) (1,0,1,1) (0,1,1,1)
3-to-3	(1,0,1,1)	(1,1,1,0) (1,1,0,1) (1,0,1,1) (0,1,1,1)	3-to-3	(0,1,1,1)	(1,1,1,0) (1,1,0,1) (1,0,1,1) (0,1,1,1)
2-to-4	(0,0,1,1) (0,1,1,0) (1,1,0,0) (1,0,1,0) (0,1,0,1) (1,0,0,1)	(1,1,1,1)	4-to-2	(1,1,1,1)	(0,0,1,1) (0,1,1,0) (1,1,0,0) (1,0,1,0) (0,1,0,1) (1,0,0,1)

Difference Vector Propagation through MixColumn. As we have discussed in Section 2.2, we fix the difference to `0x8` for each active nibble of the input and output states of the Superbox, that is, the propagations of difference vectors through MixColumn will be deterministic, which is given in Table 5. Thus, the propagation rules of MixColumn can be easily modeled by the XOR operation using its MILP or SAT/SMT constraints. For nibble-based permutation layers of Blink, the propagation of difference vectors through permutations `P` and $\overline{\mathtt{P}}$ can be directly modeled by wiring the corresponding variables.

Objective Function. With the above differential patterns through different operations of Blink and the discussion in Section 3.3, the objective function of this Superbox pattern based search model can be approximately evaluated by summing the Hamming weights of the difference vectors of the input and output for all Superboxes, that is, count the number of active S-boxes as the objective function.

Next, we will present distinguishers of Blink-64 and Blink-128, which are obtained by simply applying our search model provided above.

Remark. As mentioned in Section 2.2 and Section 3.3, we have tested three other candidates `{0x2, 0x5, 0xa}`, they both have lower probabilities ($2^{-4.68}$, *e.g.*, `0x0002` $\rightarrow$ `0x2220`

Table 5: Propagation of difference vectors through MixColumn.

Pattern	δx	δy	Pattern	δx	δy
0-to-0	(0,0,0,0)	(0,0,0,0)	4-to-4	(1,1,1,1)	(1,1,1,1)
1-to-3	(0,0,0,1) (0,0,1,0) (0,1,0,0) (1,0,0,0)	(1,1,1,0) (1,1,0,1) (1,0,1,1) (0,1,1,1)	3-to-1	(1,1,1,0) (1,1,0,1) (1,0,1,1) (0,1,1,1)	(0,0,0,1) (0,0,1,0) (0,1,0,0) (1,0,0,0)
2-to-2	(0,0,1,1) (0,1,1,0) (1,1,0,0)	(0,0,1,1) (0,1,1,0) (1,1,0,0)	2-to-2	(0,1,0,1) (1,0,1,0) (1,0,0,1)	(0,1,0,1) (1,0,1,0) (1,0,0,1)

and `0x0220` → `0x0220`), and probabilities will drop faster across chained Superboxes. Besides, `{0x2, 0x5, 0xa}` do not allow 3-to-3, 4-to-2 and 2-to-4 patterns, which makes it difficult to follow the optimal truncated differential. Hence, other potential differences are less competitive compared to the `0x8`-based patterns. This could be partially confirmed by our experiments in Section 4.2, where we fixed the input/output differences of two tested 4-round trails (both with 2 Superboxes), the experiment results show that the differential probabilities of those trails are dominated by the `0x8`-based ones.

4.2 10-round Differential Trail of Blink-64

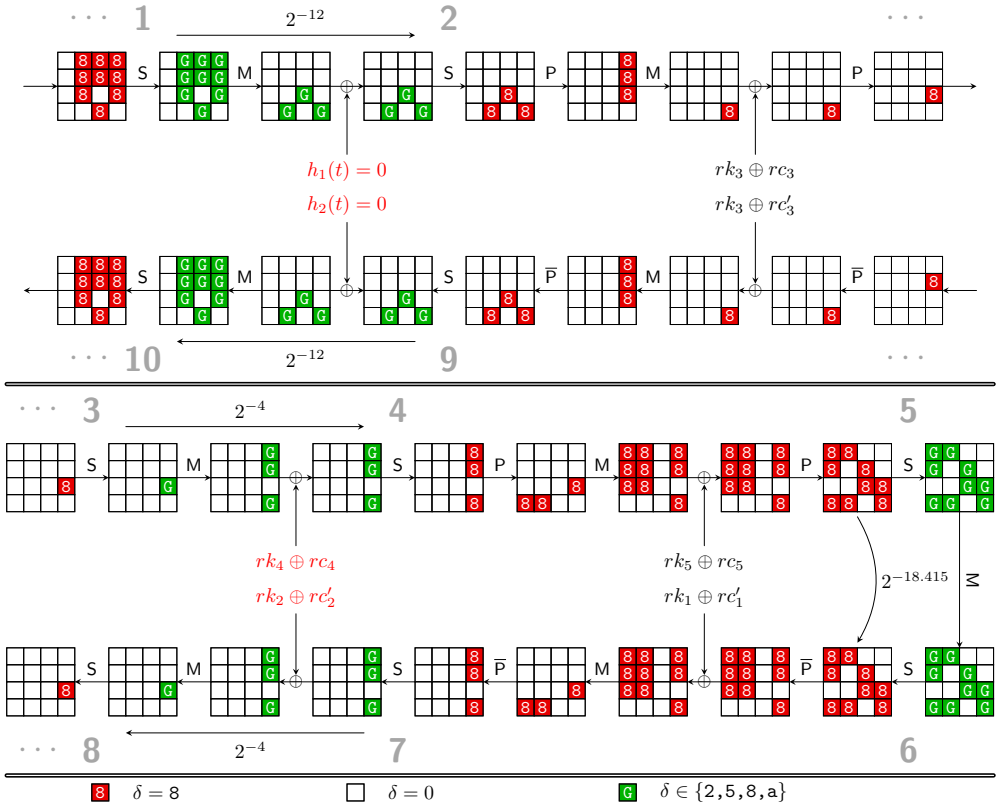


Figure 3: 10-round differential trails of Blink-64.

As given in Figure 3, the obtained 10-round differential distinguisher of Blink-64 can be divided into five Superboxes, *i.e.*, Round 1-2, Round 3-4, Round 5-6, Round 7-8 and Round 9-10. To reach the maximum probability of this distinguisher, we set tweak $t = 0$ and let rk_4, rk_2 in Superboxes of Round 3-4 and Round 7-8 to satisfy the key conditions of 3-to-1 and 1-to-3 patterns both with probability 2^{-4} in Table 3, then this 10-round distinguisher has a probability of $2^{-(12+4+18.415+4+12)} \approx 2^{-50.42}$ with 6-bit key conditions on the key space. Thus, under the weak tweak-key setting, we can significantly increase the probability of the 10-round distinguisher from 2^{-100} (the optimal characteristic evaluated by Blink designers [WHZ+25, Section 7.1]) to $2^{-50.42}$. Some experiments are also performed on this 10-round weak tweak-key differential trail as below, which could partially verify our analysis on this weak tweak-key distinguisher.

Verifications and Experiments. We firstly perform experiments on two Superboxes of Round 3-4 and Round 5-6, which are connected by $\mathtt{P} \circ \mathtt{MK} \circ \mathtt{P}$. For each of 512 randomly chosen keys, we then evaluate the differential from Round 3 to Round 6 with 2^{29} pairs, among which 329 keys (ratio $329/512 \approx 0.643$) are with zero probability and the rest keys are distributed in two clusters as shown in Figure 4, which corresponds to two kinds of key conditions with probabilities 2^{-4} and 2^{-5} for the 1-to-3 Superbox pattern of Round 3-4, *i.e.*, expecting $2^{29-18.415-4} \approx 96$ and $2^{30-18.415-5} \approx 48$ right pairs respectively.

Secondly, we perform experiments on two Superboxes of Round 1-2 and Round 3-4, which are also connected by $\mathtt{P} \circ \mathtt{MK} \circ \mathtt{P}$. For each of 512 randomly chosen keys, we similarly evaluate the differential from Round 1 to Round 4 with 2^{23} pairs, among which 319 keys (ratio $319/512 \approx 0.623$) are with zero probability and the rest keys are distributed in two clusters as shown in Figure 5, which also corresponds to two kinds of key conditions with probabilities 2^{-4} and 2^{-5} for the 1-to-3 Superbox pattern from Round 3-4, *i.e.*, expecting $2^{23-12-4} = 128$ and $2^{23-12-5} = 64$ right pairs respectively.

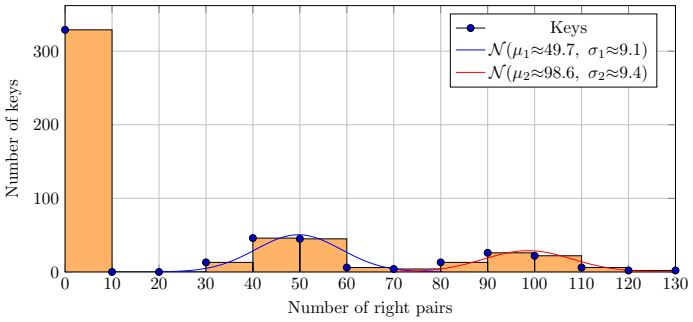


Figure 4: Experiment on the disinguisher in Figure 3 from Round 3 to Round 6.

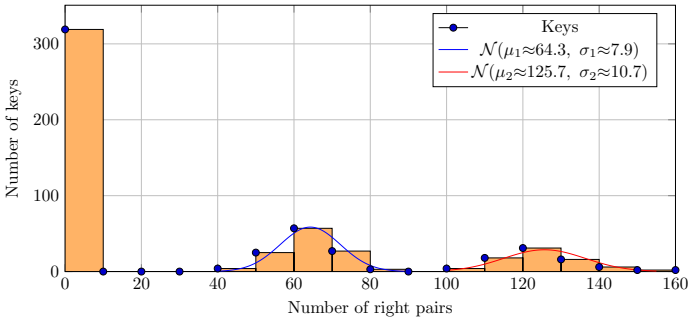


Figure 5: Experiment on the distinguisher in Figure 3 from Round 1 to Round 4.

4.3 10-round Differential Trail of Blink-128 and its Extension to 12-round Multiple Differential Distinguisher

As given in Figure 7, similarly, to reach the maximum probability, we set tweak $t = 0$, and let rk_5, rk_2 in Superboxes of Round 4-5 and Round 8-9 satisfy the key conditions of 2-to-2, 4-to-2, and 2-to-4 patterns with probabilities 2^{-4}, $2^{-6.415}$ and $2^{-6.415}$ in Table 3 respectively. Then, this 10-round distinguisher has a probability of $2^{-68.83}$ with 14-bit (3+3+4+4=14) conditions on the key space. Thus, under the weak tweak-key setting, we can significantly increase the probability of the 10-round distinguisher from 2^{-128} (the optimal characteristic evaluated by Blink designers [WHZ+25, Section 7.1]) to $2^{-68.83}$.

Extension to 12-round Multiple Differential Distinguisher Blink-128. As shown in Figure 6, we can append one round before and after, which leads to a 12-round multiple differential distinguisher with probability $2^{-68.83-28} = 2^{-96.83}$. According to the output difference set in Round 12, there are 14 green ■ active cells in the difference set $\{\texttt{0x2}, \texttt{0x5}, \texttt{0x8}, \texttt{0xa}\}$ and 18 white □ non-active cells, which corresponds to the random probability of $2^{-(2\times14+4\times18)} = 2^{-100}$ for a random input pair to reach this output difference set. So, this 12-round multiple differential distinguisher is valid[1].

As the designers claim that the data complexity for Blink-128 is 2^{80}, we evaluate the data complexity of this 12-round multiple differential distinguisher following the method of constructing the initial structure in [BG11], which will also be adopted later in our key recovery attacks on Blink. Thus, as depicted in Figure 6, the input differences in Round 1 cover $2^{2\times14} = 2^{28}$ differences, which have a linear span[2] of $2^{3\times14} = 2^{42}$. Then, for one initial structure, it can generate $2^{28+42-1} = 2^{69}$ satisfied pairs with 2^{42} chosen-plaintext queries, and one needs $2^{96.83-69} = 2^{27.83}$ such initial structures, which has a data complexity of $2^{27.83+42} = 2^{69.83}$. Thus, under the weak tweak-key setting, a 12-round multiple differential distinguisher is mounted with the probability $2^{-96.83}$ and the data complexity $2^{69.83}$.

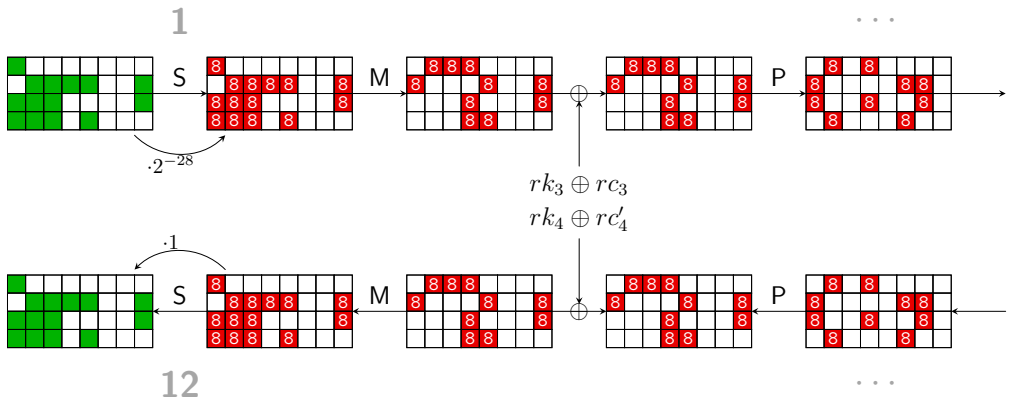


Figure 6: Appending two rounds to the 10-round differential trail of Blink-128 (cf. Figure 7) to mount a 12-round multiple differential distinguisher.

[1] We also tried to extend the 10-round differential trail of Blink-64, however, it has a probability of $2^{-(50.42+14)} = 2^{-64.42}$ after appending one round before, which is lower than the random probability 2^{-64}.

[2] As $\{\texttt{0x2}, \texttt{0x5}, \texttt{0x8}, \texttt{0xa}\}$ has a span of 3-bit dimensional space $\{\texttt{0x0}, \texttt{0x2}, \texttt{0x5}, \texttt{0x7}, \texttt{0x8}, \texttt{0xa}, \texttt{0xd}, \texttt{0xf}\}$.

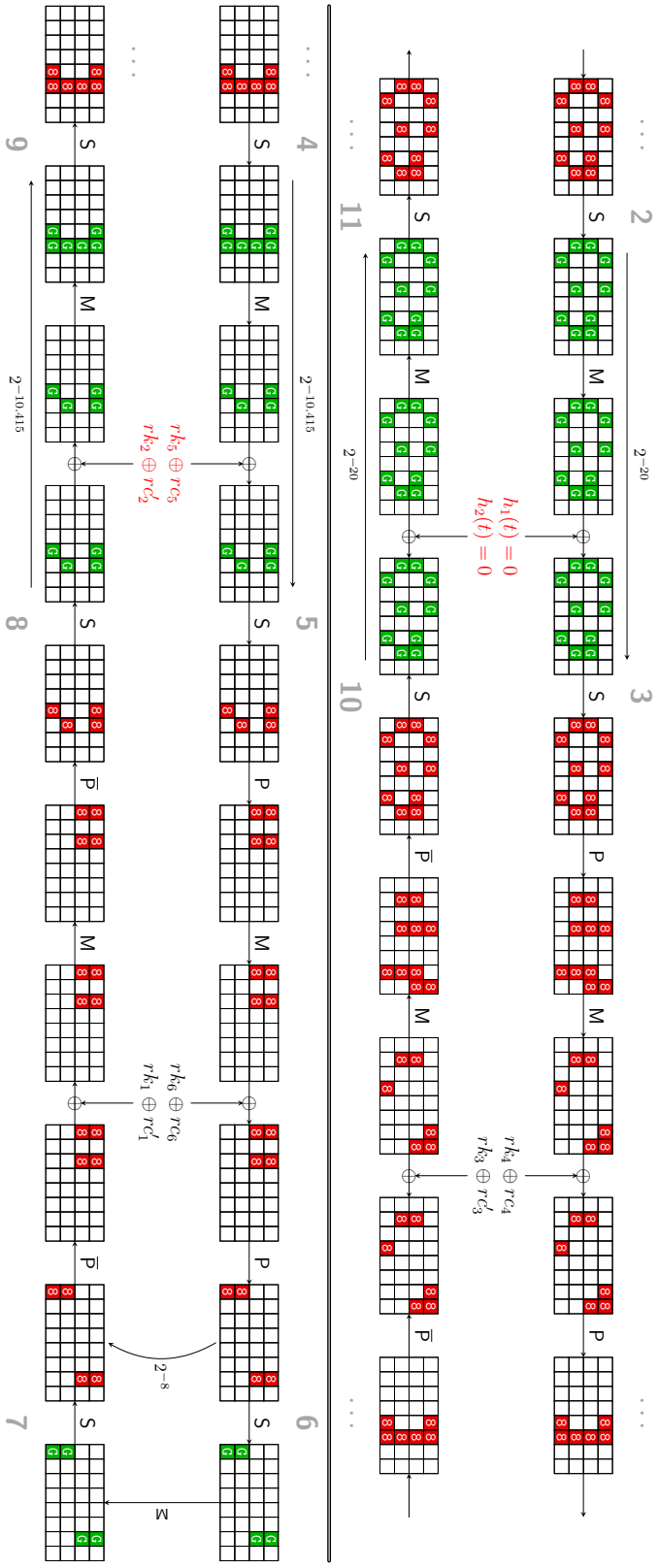


Figure 7: 10-round differential trails of Blink-128 (indexing from Round 2 to Round 11).

5 Key Recovery Attack on Blink-64

Based on the distinguishers presented above, we now provide a key recovery attack on 13-round Blink-64, in which we can recover all 448-bit key information with the overall time complexity of $2^{114.02}$ encryptions of 13-round Blink-64 and the data complexity of 2^{56} chosen plaintexts. Key recovery attacks on 13-round Blink-128 are also derived, however, the time complexity is slightly higher than the designers' claimed data complexity 2^{80}, so we just provide a simple 72-bit key recovery attack on 13-round Blink-128 in Section A.

5.1 Data Collection Phase

For the 13-round key recovery attack on Blink-64 as depicted in Figure 8, one needs to generate about $1/(2^{-64.42} \cdot 2^{-22.1-13.54}) = 2^{100.06}$ message pairs to expect 1 valid pair to follow this 11-round weak tweak-key distinguisher when tweak $t = 0$. Considering the success probability and multiple key recovery phases introduced later, we want an expected number of $N_V = 8$ valid pairs following the distinguisher, which needs to generate about $N_M = N_V \cdot 2^{100.06} \approx 2^{103.10}$ message pairs[3]. Using the similar way to construct the initial structure in [DEKM16, EK18], we can generate the satisfied pairs for the multiple differentials given in Figure 8, the input differences of the corresponding initial structure cover $|\Delta^M| = 13^{13} \approx 2^{48.10}$ differences, which have a linear span of $|\langle\Delta^M\rangle| = 16^{13} = 2^{52}$. From one initial structure, $2^{52+48.10-1} = 2^{99.10}$ pairs can then be obtained with $|\langle\Delta^M\rangle| = 2^{52}$ data queries. Finally, we need a number of $N_{IS} = N_M/2^{99.10} = 2^4$ initial structures, and this has the data complexity of $D = N_{IS} \cdot |\langle\Delta^M\rangle| = 2^{56}$ chosen-plaintext queries, which corresponds to the security claim of data complexity for Blink-64.

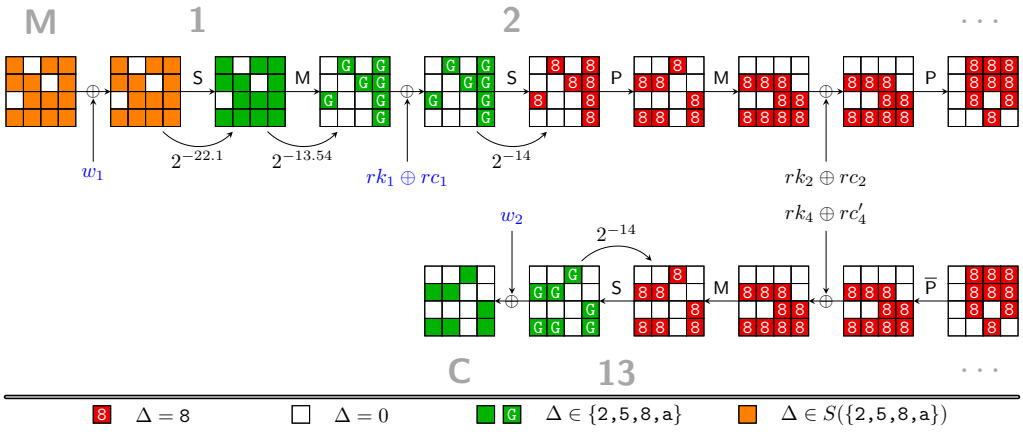


Figure 8: Phase I of key recovery attacks on 13-round Blink-64 (Round 1, 2 and 13 are appended to the distinguisher in Figure 3).

Filtering Conditions on Ciphertext before Key Guessing. As shown in Figure 8, the output differences of a valid ciphertext pair will have two types of filtering conditions: 9 white cells of zero differences and 7 green cells of $\{\texttt{0x2}, \texttt{0x5}, \texttt{0x8}, \texttt{0xa}\}$, which has a filtering probability of $1/|\Delta^C| = (2^{-4})^9 \cdot (2^{-2})^7 = 2^{-50}$. By using filtering conditions on the ciphertext differences, the relevant pairs for later key recovery phases can be reduced from $N_M = 2^{103.10}$ to $N_F = N_M/|\Delta^C| = 2^{53.10}$, which contains N_V expected valid pairs.

[3]Considering the initial structure, we roughly let the number of required message pairs be $N_M \approx 2^{103.10}$.

5.2 Key Recovery Phases

We now make use of $N_F = 2^{53.10}$ filtered pairs from the data collection phase above to narrow down the keyspace, which has four phases detailed below.

Table 6: Key conditions to check for key recovery Phase I.

Round	Input	Output	Operation	Condition	Guessed Key bits
1	■ †	■	S	$2^{-1.7\times 13} = 2^{-21}$	$13 \times 4 = 54$
	(■,■,□,■)	(□,□,■,□)	M	2^{-4}	-
	(□,■,■,■)	(■,□,□,□)	M	2^{-4}	-
	(■,□,■,■)	(□,■,□,□)	M	2^{-4}	-
	(■,■,■,■)	(■,■,■,■) ‡	M	$2^{-1.54}$	-
2,13	G	8	S	$2^{-2\times 7} = 2^{-14}$	$7 \times 4 = 28$

† According to DDT of Blink, for the output difference set {0x2, 0x5, 0x8, 0xa}, it has 13 input differences {0x1, 0x2, 0x3, 0x4, 0x5, 0x6, 0x8, 0x9, 0xa, 0xb, 0xc, 0xd, 0xe}, which will propagate to {0x2, 0x5, 0x8, 0xa} with probability $4/13 \approx 2^{-1.7}$.

‡ For all four cells with difference set {0x2, 0x5, 0x8, 0xa} in a column, it has 88 cases for its output column to keep this form after M operation, thus with probability $88/4^4 \approx 2^{1.54}$.

Phase I: Recovering 108 bits of key material from N_F filtered pairs. We check the key guesses on several conditions in Round 1, 2 and 13 as shown in Figure 8, which are listed in Table 6 in detail.

- **Round 1.** Guessing 52 bits (corresponding to 13 active orange cells) of w_1 allows us to compute forward to output differences of S and M in Round 1. Then, for each filtered pair, the probabilities of these active cells to hold for constrained differences are $2^{-22.1} \cdot 2^{-13.54} = 2^{-35.64}$. So, it has the time complexity with about $2^{52+53.10} = 2^{105.1}$ encryptions of 1-round Blink-64, which will expect $N_F \cdot 2^{-35.64} = 2^{17.46}$ pairs left for each guessed key.
- **Round 2.** Guessing 28 bits (corresponding to 7 active cells) of rk_1 allows us to compute forward to the output of S in Round 2, which has a probability with 2^{-14} and expects $N_F \cdot 2^{-35.64} \cdot 2^{-14} = 2^{3.46}$ pairs to be left. The time complexity of this step is about $2^{52+28+17.46} = 2^{97.46}$ encryptions of 1-round Blink-64.
- **Round 13.** Similarly, guessing 28 bits of w_2 allows us to compute backward to the input of S in Round 13, which also has a probability with 2^{-14} and expects $N_F \cdot 2^{-35.64} \cdot 2^{-14} \cdot 2^{-14} = 2^{-10.54}$ pairs to be finally left. The time complexity of this step is about $2^{52+28+28+3.46} = 2^{111.46}$ encryptions of 1-round Blink-64.

Thus, for a wrong key guess of 108-bit key material, there will be about $2^{-10.54}$ suggested message pairs, and the right key guess will expect $N_V = 8$ valid pairs, so the signal-to-noise ratio is $S_N = 8/2^{-10.54} = 2^{13.54}$. According to the formula in [Sel08, Section 3] as below

$$P_S = \Phi\left(\frac{\sqrt{\mu S_N} - \Phi^{-1}(1-2^{-a})}{\sqrt{S_N+1}}\right),$$

where Φ is the normal distribution, μ is the number of right pairs ($N_V = 8$ here), and a is the advantage level. Since only the top suggested key candidate (*i.e.*, $2^{108-a} = 1$) is kept, it thus delivers a 108-bit advantage. So, the success probability with the parameters above to recover the right 108-bit key is $P_S = 99.67\%$, and the time complexity of Phase I is $T_1 = 1/13 \cdot (2^{105.1} + 2^{97.46} + 2^{111.46}) \approx 2^{107.78}$ encryptions of 13-round Blink-64.

Remark. It should be noted that directly modeling the key counters using the binomial distribution may lead to a more accurate analysis of the success probability and the complexity. This was also mentioned in [SB02, Section 3.2.1], where the author discussed that replacing normal approximation with binomial (or Poisson) distributions can provide a (slightly) refined estimate in certain cases. However, as shown in [SB02, Section 3.4], there appears to be no significant difference between the results obtained using the normal approximation and those derived from the original binomial distribution, even for relatively small values of μ_0. Besides, considering relatively large S_N and number of right pairs, the formula of normal distribution is used here, as N_M is very large and the binomial distribution can be approximated by the normal distribution.

Phase II: Recovering 156 bits of key material from N_V valid pairs. With the recovered 108 bits of key information and $N_V = 8$ valid pairs from Phase I, we now check conditions inside the distinguisher to narrow down the key space further. As shown in Figure 9, once guessing 3 left cells of w_1, 9 left cells of rk_1 and 9 cells of rk_2, we can check the Superbox from Round 3 to Round 4 with a filtering probability of 2^{-12}. Considering N_V valid pairs, it has a combined filter power of $2^{-12\cdot 8} = 2^{-96}$ on the guessed 84-bit key of w_1, rk_1 and rk_2. Similarly, by guessing 9 left cells of w_2 and 9 cells of rk_4, we can check the Superbox from Round 12 to Round 11, which also has a filter power of 2^{-96} on the guessed 72-bit key of w_2 and rk_4. Thus, the time complexity of Phase II is $T_2 = 8 \times (4/13 \cdot 2^{84} + 3/13 \cdot 2^{72}) \approx 2^{85.30}$ encryptions of 13-round Blink-64.

Phase III: Recovering 72 bits of key material from N_V valid pairs. With the recovered 264 bits of key information and $N_V = 8$ valid pairs from Phase I and II, as depicted in Figure 9, we can further guess the left 7 cells of rk_2, 4 cells of rk_3 and 2 cells of rk_4 (2 cells of rk_4 are guessed in Phase II), then the Superbox from Round 5 to Round 6 can be checked with the probability 2^{-4}. To check the Superbox from Round 10 to Round 9, we only need to guess the left 5 cells of rk_4 as the required key material is already guessed in rk_2, rk_3, rk_4 above. In total, this has a combined filter power of $2^{(-4-4)\times 8} = 2^{-64}$ on all 72-bit guessed key (2^{66} key space due to two 3-bit conditions on rk_2 and rk_4) after repeating with $N_V = 8$ valid pairs. Thus, the time complexity of Phase III is about $T_3 = 8 \times 2^{66} = 2^{69}$, which expects to leave $2^{66-64} = 4$ key candidates.

Phase IV: Exhausting the remaining 112 bits of key material. After three phases above, we will obtain $108 + 156 + 72 = 336$ bits of key material, and the remaining 112 bits of key material (48 bits of rk_3 and 64 bits of rk_5) can be recovered by exhaustive search with $T_4 = 4 \times 2^{112} = 2^{114}$ encryptions due to 4 left key candidates in Phase III. So, based on our weak tweak-key differential attack on 13-round Blink-64, the total time complexity to recover all 448-bit key information is $T = (T_1 + T_2 + T_3 + T_4) = (2^{107.78} + 2^{85.30} + 2^{69} + 2^{114}) \approx 2^{114.02}$ encryptions of 13-round Blink-64.

6 Conclusion

We have presented the first third-party cryptanalysis of Blink, showing that it admits a structured Superbox-based analysis. By developing a lightweight theoretical model tailored to the affine value spaces inside a Blink Superbox, we characterized its weak-key conditions, fixed-key differential probabilities, and local clustering, and integrated this with a simple pattern-based automatic search model. Our results include 10-round weak tweak-key distinguishers for Blink ciphers, a 12-round multiple differential distinguisher for Blink-128, and a 13-round key-recovery attack on Blink-64 that recovers the full 448-bit key in time $2^{114.02}$ using 2^{56} chosen plaintexts, all of which are within the designers' claimed data and time complexities. While our findings remain consistent with the designers' security

claims, which highlight subtle edge-case behaviors arising from simple cipher components and their interaction with tweak and key, and could be potentially informative for future refinements of tweakable block cipher constructions in general.

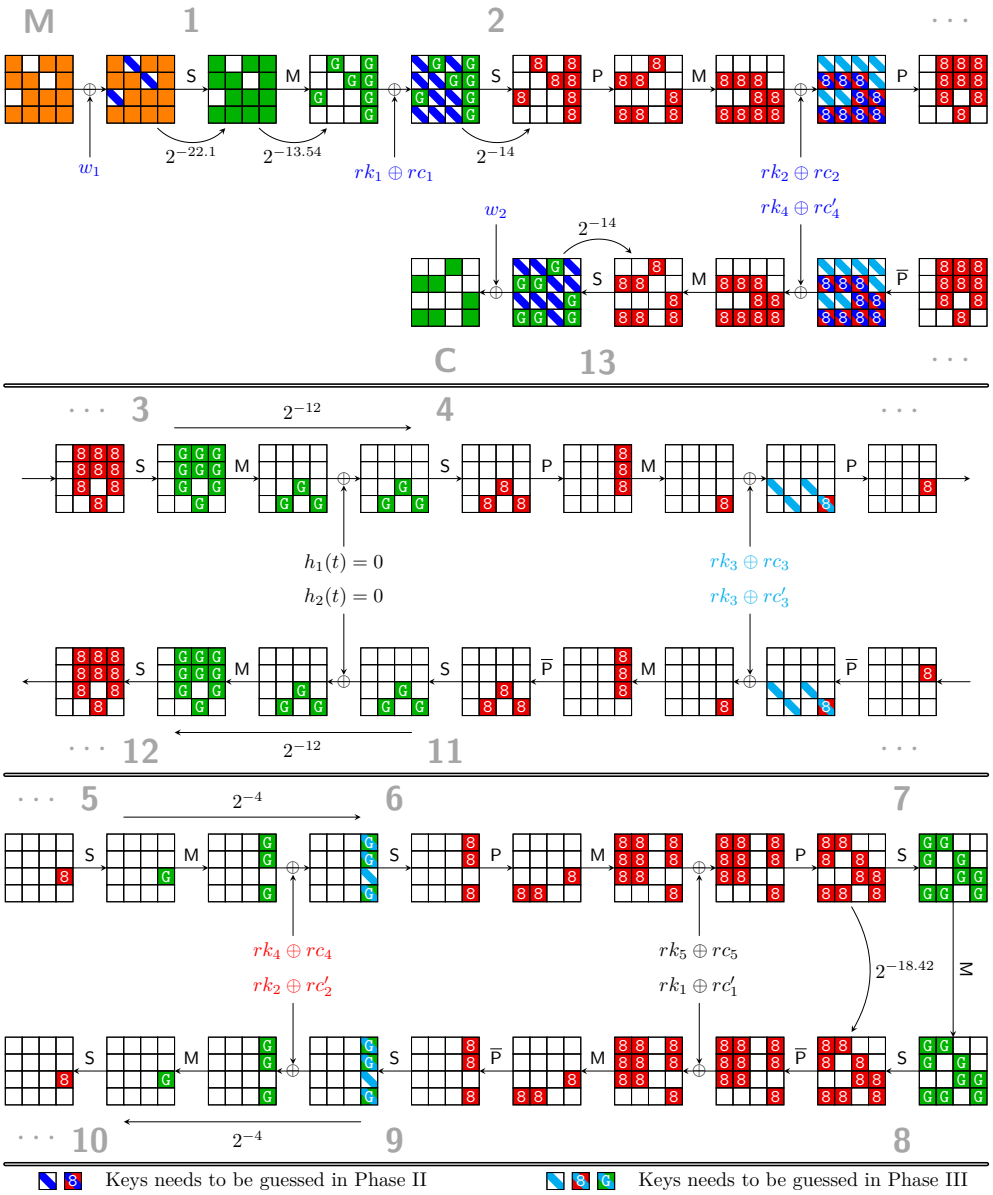


Figure 9: Phase II and III of key recovery attacks on 13-round Blink-64.

Acknowledgements. We would like to thank all anonymous reviewers for their fruitful and helpful comments. We also appreciate the feedbacks from the designers of Blink, which can help them refine the security analysis and strengthen the cipher further. This research is supported by the National Key R&D Program of China (Grant No. 2024YFA1013000), the National Cryptologic Science Fund of China (Grant No. 2025NCSF01013), the National Natural Science Foundation of China (Grant No. U2336207, 62032014), Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (JYB2025XDXM114), and the Ministry of Education in Singapore under Grant RG102/24.

References

[ABD+23] Roberto Avanzi, Subhadeep Banik, Orr Dunkelman, Maria Eichlseder, Shibam Ghosh, Marcel Nageler, and Francesco Regazzoni. The qarmav2 family of tweakable block ciphers. *IACR Trans. Symmetric Cryptol.*, 2023(3):25–73, 2023.

[AST+17] Ahmed Abdelkhalek, Yu Sasaki, Yosuke Todo, Mohamed Tolba, and Amr M. Youssef. MILP modeling for (large) s-boxes to optimize probability of differential characteristics. *IACR Trans. Symmetric Cryptol.*, 2017(4):99–129, 2017.

[Ava17] Roberto Avanzi. The QARMA Block Cipher Family. Almost MDS Matrices Over Rings With Zero Divisors, Nearly Symmetric Even-Mansour Constructions With Non-Involutory Central Rounds, and Search Heuristics for Low-Latency S-Boxes. *IACR Trans. Symmetric Cryptol.*, 2017(1):4–44, 2017.

[BBF+25] Jules Baudrin, Christof Beierle, Patrick Felke, Gregor Leander, Patrick Neumann, Léo Perrin, and Lukas Stennes. Commutative cryptanalysis as a generalization of differential cryptanalysis. *Des. Codes Cryptogr.*, 93(8):3243–3281, 2025.

[BCG+12] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalçin. PRINCE - A Low-Latency Block Cipher for Pervasive Computing Applications - Extended Abstract. In *ASIACRYPT 2012*, pages 208–225, 2012.

[BDD+23] Yanis Belkheyar, Joan Daemen, Christoph Dobraunig, Santosh Ghosh, and Shahram Rasoolzadeh. Bipbip: A low-latency tweakable block cipher with small dimensions. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2023(1):326–368, 2023.

[BDG+25a] Yanis Belkheyar, Patrick Derbez, Shibam Ghosh, Gregor Leander, Silvia Mella, Léo Perrin, Shahram Rasoolzadeh, Lukas Stennes, Siwei Sun, Gilles Van Assche, and Damian Vizár. Chilow and chichi: New constructions for code encryption. In *EUROCRYPT (1)*, volume 15601 of *Lecture Notes in Computer Science*, pages 212–243. Springer, 2025.

[BDG25b] Christina Boura, Patrick Derbez, and Baptiste Germon. Extending the quasidifferential framework: From fixed-key to expected differential probability. *IACR Trans. Symmetric Cryptol.*, 2025(1):515–541, 2025.

[BDG+25c] Christina Boura, Patrick Derbez, Baptiste Germon, Rachelle Heim Boissier, and María Naya-Plasencia. SPEEDY: Caught at last. *Cryptology ePrint Archive*, 2025. To appear in Asiacrypt 2025.

[BDHBNP23] Christina Boura, Nicolas David, Rachelle Heim Boissier, and María Naya-Plasencia. Better steady than speedy: Full break of speedy-7-192. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 36–66. Springer, 2023.

[BEK+20] Dusan Bozilov, Maria Eichlseder, Miroslav Knezevic, Baptiste Lambin, Gregor Leander, Thorben Moos, Ventzislav Nikov, Shahram Rasoolzadeh,

Yosuke Todo, and Friedrich Wiemer. Princev2 - more security for (almost) no overhead. In Orr Dunkelman, Michael J. Jacobson Jr., and Colin O'Flynn, editors, *Selected Areas in Cryptography - SAC 2020 - 27th International Conference, Halifax, NS, Canada (Virtual Event), October 21-23, 2020, Revised Selected Papers*, volume 12804 of *Lecture Notes in Computer Science*, pages 483–511. Springer, 2020.

[Bey18] Tim Beyne. Block cipher invariants as eigenvectors of correlation matrices. In *ASIACRYPT (1)*, volume 11272 of *Lecture Notes in Computer Science*, pages 3–31. Springer, 2018.

[BFL+23] Jules Baudrin, Patrick Felke, Gregor Leander, Patrick Neumann, Léo Perrin, and Lukas Stennes. Commutative cryptanalysis made practical. *IACR Trans. Symmetric Cryptol.*, 2023(4):299–329, 2023.

[BG11] Céline Blondeau and Benoît Gérard. Multiple differential cryptanalysis: Theory and practice. In Antoine Joux, editor, *Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers*, volume 6733 of *Lecture Notes in Computer Science*, pages 35–54. Springer, 2011.

[BJK+16] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS. In *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, volume 9815, pages 123–153. Springer, 2016.

[BN25] Tim Beyne and Addie Neyt. Improved differential cryptanalysis of SPEEDY. *IACR Trans. Symmetric Cryptol.*, 2025(3):444–474, 2025.

[BR22] Tim Beyne and Vincent Rijmen. Differential cryptanalysis in the fixed-key model. In *CRYPTO (3)*, volume 13509 of *Lecture Notes in Computer Science*, pages 687–716. Springer, 2022.

[CF25] Anne Canteaut and Merlin Fruchon. Understanding unexpected fixed-key differential behaviours: How to avoid major weaknesses in lightweight designs (extended version). Cryptology ePrint Archive, Paper 2025/1553, 2025.

[CFG+14] Anne Canteaut, Thomas Fuhr, Henri Gilbert, María Naya-Plasencia, and Jean-René Reinhard. Multiple differential cryptanalysis of round-reduced PRINCE. In *FSE*, volume 8540 of *Lecture Notes in Computer Science*, pages 591–610. Springer, 2014.

[CGL+23] Federico Canale, Tim Güneysu, Gregor Leander, Jan Philipp Thoma, Yosuke Todo, and Rei Ueno. SCARF - A low-latency block cipher for secure cache-randomization. In Joseph A. Calandrino and Carmela Troncoso, editors, *32nd USENIX Security Symposium, USENIX Security 2023, Anaheim, CA, USA, August 9-11, 2023*, pages 1937–1954. USENIX Association, 2023.

[CLN+17] Anne Canteaut, Eran Lambooij, Samuel Neves, Shahram Rasoolzadeh, Yu Sasaki, and Marc Stevens. Refined Probability of Differential Characteristics Including Dependency Between Multiple Rounds. *IACR Trans. Symmetric Cryptol.*, 2017(2):203–227, 2017.

[CWWH25] Chengcheng Chang, Meiqin Wang, Wei Wang, and Kai Hu. Quasidifferential saves infeasible differential - improved weak-key key-recovery attacks on round-reduced GIFT. In *Topics in Cryptology - CT-RSA 2025 - Cryptographers' Track at the RSA Conference 2025, San Francisco, CA, USA, April 28-May 1, 2025, Proceedings*, pages 27–50, 2025.

[DEKM16] Christoph Dobraunig, Maria Eichlseder, Daniel Kales, and Florian Mendel. Practical key-recovery attack on MANTIS5. *IACR Trans. Symmetric Cryptol.*, 2016(2):248–260, 2016.

[DR07] Joan Daemen and Vincent Rijmen. Plateau characteristics. *IET Inf. Secur.*, 1(1):11–17, 2007.

[EK18] Maria Eichlseder and Daniel Kales. Clustering related-tweak characteristics: Application to MANTIS-6. *IACR Trans. Symmetric Cryptol.*, 2018(2):111–132, 2018.

[GJN+16] Jian Guo, Jérémy Jean, Ivica Nikolic, Kexin Qiao, Yu Sasaki, and Siang Meng Sim. Invariant subspace attack against midori64 and the resistance criteria for s-box designs. *IACR Trans. Symmetric Cryptol.*, 2016(1):33–56, 2016.

[GP10] Henri Gilbert and Thomas Peyrin. Super-Sbox Cryptanalysis: Improved Attacks for AES-Like Permutations. In *FSE*, volume 6147, pages 365–383, 2010.

[JNP14] Jérémy Jean, Ivica Nikolic, and Thomas Peyrin. Tweaks and Keys for Block Ciphers: The TWEAKEY Framework. In *ASIACRYPT 2014, Proceedings, Part II*, volume 8874, pages 274–288, 2014.

[LMMR21] Gregor Leander, Thorben Moos, Amir Moradi, and Shahram Rasoolzadeh. The SPEEDY family of block ciphers engineering an ultra low-latency cipher from gate level for secure processor architectures. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2021(4):510–545, 2021.

[LMR+09] Mario Lamberger, Florian Mendel, Christian Rechberger, Vincent Rijmen, and Martin Schläffer. Rebound Distinguishers: Results on the Full Whirlpool Compression Function. In *ASIACRYPT*, pages 126–143, 2009.

[LZS+20] Yunwen Liu, Wenying Zhang, Bing Sun, Vincent Rijmen, Guoqiang Liu, Chao Li, Shaojing Fu, and Meichun Cao. The phantom of differential characteristics. *Des. Codes Cryptogr.*, 88(11):2289–2311, 2020.

[MRST09] Florian Mendel, Christian Rechberger, Martin Schläffer, and Søren S. Thomsen. The Rebound Attack: Cryptanalysis of Reduced Whirlpool and Grøstl. In *FSE*, volume 5665, pages 260–276, 2009.

[MWGP11] Nicky Mouha, Qingju Wang, Dawu Gu, and Bart Preneel. Differential and linear cryptanalysis using mixed-integer linear programming. In *International Conference on Information Security and Cryptology*, volume 7537, pages 57–76. Springer, 2011.

[NGJE25] Marcel Nageler, Shibam Ghosh, Marlene Jüttler, and Maria Eichlseder. Autodiver: Automatically verifying differential characteristics and learning key conditions. *IACR Trans. Symmetric Cryptol.*, 2025(1):471–514, 2025.

[PT22] Thomas Peyrin and Quan Quan Tan. Mind your path: On (key) dependencies in differential characteristics. *IACR Trans. Symmetric Cryptol.*, 2022(4):179–207, 2022.

[Rom10] S. Roman. *Advanced Linear Algebra.* Graduate Texts in Mathematics. Springer New York, 2010.

[SB02] Ali Aydin Selçuk and Ali Biçak. On probability of success in linear and differential cryptanalysis. In Stelvio Cimato, Clemente Galdi, and Giuseppe Persiano, editors, *Security in Communication Networks, Third International Conference, SCN 2002, Amalfi, Italy, September 11-13, 2002. Revised Papers*, volume 2576 of *Lecture Notes in Computer Science*, pages 174–185. Springer, 2002.

[Sel08] Ali Aydin Selçuk. On Probability of Success in Linear and Differential Cryptanalysis. *J. Cryptology*, 21(1):131–147, 2008.

[SHW+14] Siwei Sun, Lei Hu, Peng Wang, Kexin Qiao, Xiaoshuang Ma, and Ling Song. Automatic Security Evaluation and (Related-key) Differential Characteristic Search: Application to SIMON, PRESENT, LBlock, DES(L) and Other Bit-Oriented Block Ciphers. In *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, volume 8873, pages 158–178, 2014.

[SWW18] Ling Sun, Wei Wang, and Meiqin Wang. More accurate differential properties of LED64 and midori64. *IACR Trans. Symmetric Cryptol.*, 2018(3):93–123, 2018.

[Wag04] David A. Wagner. Towards a unifying view of block cipher cryptanalysis. In *FSE*, volume 3017 of *Lecture Notes in Computer Science*, pages 16–33. Springer, 2004.

[WHZ+25] Jianhua Wang, Tao Huang, Guang Zeng, Tianyou Ding, Shuang Wu, and Siwei Sun. THF: designing low-latency tweakable block ciphers. *IACR Trans. Symmetric Cryptol.*, 2025(4):125–166, 2025.

[WNL+23] Jinliang Wang, Chao Niu, Qun Liu, Muzhou Li, Bart Preneel, and Meiqin Wang. Cryptanalysis of speedy. In *Australasian Conference on Information Security and Privacy*, pages 124–156. Springer, 2023.

Appendix A Key Recovery Attack on Blink-128

Based on the 10-round distinguisher presented in Section 4.3, we now provide a key recovery attack on 13-round Blink-128, in which we just recover 72-bit key information with the overall time complexity of $2^{119.43}$ encryptions of 13-round Blink-128 and the data complexity of $2^{85.52}$ chosen plaintexts. As the data complexity is slightly over the designers' claim 2^{80}, so we did not try to recover more key bits from the distinguisher inside. In the following, we just provide how to recover 72-bit key information of w_1.

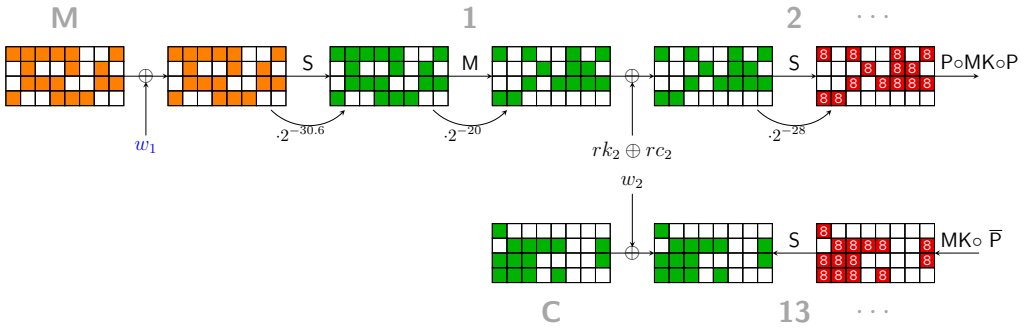


Figure 10: Key recovery attack on 13-round Blink-64 (Round 1 is appended to the distinguisher in Figure 6).

Data Collection Phase. As depicted in Figure 10, similar to key recovery attacks on Blink-64 in Section 5, one needs to generate about $1/(2^{-96.83} \cdot 2^{-20-30.6}) = 2^{-147.43}$ message pairs to expect 1 valid pair to follow the 12-round weak tweak-key distinguisher (cf. Section 4.3) when tweak $t = 0$. Considering the success probability, we want an expected number of $N_V = 13$ valid pairs following the distinguisher, which needs to generate about $N_M = N_V \cdot 2^{147.43} \approx 2^{151.13}$ message pairs. The input differences of the corresponding initial structure cover $|\Delta^M| = 13^{18} \approx 2^{66.61}$ differences, which have a linear span of $|\langle \Delta^M \rangle| = 16^{18} = 2^{72}$. From one initial structure, $2^{66.61+72-1} = 2^{137.61}$ pairs can then be obtained with $|\langle \Delta^M \rangle| = 2^{72}$ data queries. Finally, we need a number of $N_{IS} = N_M/2^{137.61} = 2^{13.52}$ initial structures, and this has a data complexity of $D = N_{IS} \cdot |\langle \Delta^M \rangle| = 2^{85.52}$ chosen-plaintext queries.

Filtering Conditions on Ciphertext before Key Guessing. As shown in Figure 10, the output differences of a valid ciphertext pair will have two types of filtering conditions: 18 white cells of zero differences and 14 green cells of $\Delta = \{\texttt{0x2}, \texttt{0x5}, \texttt{0x8}, \texttt{0xa}\}$, which has a filtering probability of $1/|\Delta^C| = (2^{-4})^{18} \cdot (2^{-2})^{14} = 2^{-100}$. By using the filtering conditions on the ciphertext differences, the relevant pairs for later key recovery phases can be reduced from $N_M = 2^{151.13}$ to $N_F = N_M/|\Delta^C| = 2^{51.13}$, which contains the expected $N_V = 13$ valid pairs.

Key Recovery Phase. We now make use of $N_F = 2^{51.13}$ filtered pairs from the data collection phase above to narrow down the keyspace, and we only need to check the key guesses on conditions in Round 1, which are similar to that in Table 6.

- **Round 1.** Guessing 72 bits (corresponding to 18 active orange cells) of w_1 allows us to compute forward to output differences of `S` and `M` in Round 1. Then, for each filtered pair, the probabilities of these active cells to hold for constrained differences are $2^{-30.6} \cdot 2^{-20} = 2^{-50.6}$. So, it has the time complexity with about $2^{72+51.13} = 2^{123.13}$

encryptions of 1-round Blink-128, which will expect $N_F \cdot 2^{-50.6} = 2^{0.53}$ pairs left for each guessed key.

Thus, for a wrong key guess of 72-bit key material, there will be about $2^{0.53}$ suggested message pairs, and the right key guess will expect $N_V = 13$ valid pairs, so the signal-to-noise ratio is $S_N = 13/2^{0.53} = 2^{3.17}$. According to the formula of success probability in [Sel08, Section 3], the number of right pairs μ is 13, and we keep only the top suggested key candidate (*i.e.*, $2^{72-a} = 1$) during the ranking process, it thus delivers an a-bit advantage with $a = 72$. So, the success probability with the parameters above to recover the right 72-bit key is $P_S = 64.21\%$, and the time complexity is $T = 1/13 \times 2^{123.13} \approx 2^{119.43}$ encryptions of 13-round Blink-128.