Multidimensional Zero-Correlation Linear Cryptanalysis of Reduced Round SPARX-128

Mohamed Tolba, Ahmed Abdelkhalek, and Amr M. Youssef

Concordia Institute for Information Systems Engineering
Concordia University, Montréal, Québec, Canada

Abstract. SPARX is a family of ARX-based block ciphers proposed at Asiacrypt 2016. This family was designed with the aim of providing provable security against single-characteristic linear and differential cryptanalysis. SPARX-128/128 and SPARX-128/256 are two members of this family which operate on data blocks of length 128 bits and keys of length 128 and 256 bits, respectively. In this work, we show the existence of zero-correlation distinguishers for SPARX-128/128 and SPARX-128/256 with length longer than the best distinguishers reported by the SPARX's designers. In particular, we propose a zero-correlation distinguisher that covers 6 steps (24 rounds) for both variants of SPARX-128. Then, using specific linear masks and utilizing the properties of the linear layer and the S-box, we extended this distinguisher to 6.25 steps (25 rounds). By further utilizing the properties of the key schedule, we extended the 24-round distinguisher by 4 rounds to present a 28-round multidimensional zero-correlation attack against SPARX-128/256, i.e., 7 steps out of 10 steps. The 28-round attack is then extended to a 29-round (7.25 out of 10 steps) zero-correlation attack against SPARX-128/256 with the full code book by using the developed 25-round distinguisher. In addition, we extend the 25-round distinguisher by one round to launch a 26-round multidimensional zero-correlation attack against SPARX-128/128, i.e., 6.5 steps out of 8 steps. To the best of our knowledge, the presented attacks are the best currently known attacks against these two members of the SPARX family.
Keywords: Block Ciphers, Cryptanalysis, Multidimensional zero-correlation, SPARX.

1 Introduction

With the aim of developing block ciphers with provable security against single-characteristic linear and differential cryptanalysis, Dinu *et al.* [7] proposed a new ARX-based family of block ciphers at Asiacrypt 2016. This goal is achieved by proposing a strategy, named the long trail strategy, which is different from the wide trail strategy [6], that is widely used by many S-box based block ciphers, in that it uses a rather weak but larger S-boxes such as the ARX-based S-boxes in addition to employing a very light linear transformation layer. Adopting this strategy in the SPARX family allowed the designer to prove the security of the cipher against single-characteristic linear and differential cryptanalysis by

bounding the maximum linear and differential probabilities for any number of rounds.

SPARX-128/128 and SPARX-128/256 are two members of the SPARX family which employ a data block of length 128 bits using 128 and 256 key bits, respectively. SPARX-128/128 and SPARX-128/256 were analyzed against a wide range of attacks by the designers, and the best attacks on both of them were found using integral cryptanalysis based on Todo's division property [11]. More precisely, the best currently published attacks on SPARX-128/128 and SPARX-128/256 cover 22 and 24 rounds, respectively, which made the designers conclude that the security margin for all instances of SPARX is more than 30%.

Zero-correlation [4] is one of the recent techniques that is used to analyze symmetric-key primitives, where the attacker utilizes a linear approximation of probability exactly 1/2 over r_m rounds to act as a distinguisher. Then, this distinguisher can be utilized in a key recovery attack such that the keys which lead to this distinguisher are excluded. This technique proves its success against many of the recently proposed block ciphers as exemplified by the work done in [13,10,4,14,12].

In this paper, we present the longest published distinguisher against SPARX-128/128 and SPARX-128/256. This distinguisher is based on the zero-correlation cryptanalysis and covers 24 rounds. In addition, the properties of the linear layer and the S-box enable us to extend this distinguisher by one more round using a specific linear mask at the output of the distinguisher to achieve a 25-round zero-correlation distinguisher. Furthermore, from the properties of the S-box, we have a two rounds linear approximation which holds with a probability 1. Then, by exploiting the key schedule relations, we place this two-round linear approximation in a position that enables us to extend the 24-round distinguisher by 4 complete rounds, i.e., including the linear layer, to launch a 28-round key recovery attack against SPARX-128/256 using multidimensional zero-correlation attack. Moreover, we extended our 28-round attack by one more round using the 25-round distinguisher to launch a 29-round zero-correlation attack against SPARX-128/256 with the full code book. In addition, we extended the 25-round distinguisher to launch a 26-round attack against SPARX-128/128. The results of our paper are summarized in Table 1.

The remainder of the paper is organized as follows. The notations used throughout the paper and the specifications of SPARX-128/128 and SPARX-128/256 are presented in Section 2. Section 3 presents a brief introduction about zero-correlation and multidimensional zero-correlation attacks. In Section 4, we present our distinguisher for SPARX-128/128 and SPARX-128/256. Afterwards, in Section 5, we provide a detailed description of our multidimensional zero-correlation attacks against SPARX-128/128 and SPARX-128/256, and finally we conclude the paper in Section 6.

2 Description of SPARX-128/128 and SPARX-128/256

The following notations are used throughout the paper:

Table 1: Summary of the cryptanalysis results on SPARX-128/128 and SPARX-128/256

Attack	Version	# rounds	Time	Data	Reference
Integral	SPARX-128/256	24	2^{233}	2^{104} CP	[7]
Multidimensional zero-correlation	SPARX-128/256	28	$2^{233.5}$	$2^{123.25}$ KP	This work(Sec.5.1)
Zero-correlation	SPARX-128/256	29	$2^{227.2}$	†	This work(Sec. 5.1)
Integral	SPARX-128/128	22	2^{105}	2^{102} CP	[7]
Multidimensional zero-correlation	SPARX-128/128	26	$2^{117.25}$	$2^{116.2}$ KP	This work(Sec. 5.2)

†: Requires the full codebook, CP: Chosen Plaintext, KP: Known Plaintext.

- K: The master key.
- k_i: The i^{th} 16-bit of the key state, where $0 \leq i \leq 7$ for SPARX-128/128, and $0 \leq i \leq 15$ for SPARX-128/256.
- K_i: The i^{th} 32-bit of the key state, where $0 \leq i \leq 3$ for SPARX-128/128, and $0 \leq i \leq 7$ for SPARX-128/256.
- k_i^j: The i^{th} 16-bit of the key state after applying the key schedule permutation j times, where $0 \leq i \leq 7$, $0 \leq j \leq 32$ for SPARX-128/128, and $0 \leq i \leq 15$, $0 \leq j \leq 20$ for SPARX-128/256.
- K_i^j: The i^{th} 32-bit of the key state after applying the key schedule permutation j times, where $0 \leq i \leq 3$, $0 \leq j \leq 32$ for SPARX-128/128, and $0 \leq i \leq 7$, $0 \leq j \leq 20$ for SPARX-128/256.
- $RK_{(a,i)}$: The 32-bit round key used at branch a of round i where $0 \leq i \leq 32$ (resp. $0 \leq i \leq 40$) for SPARX-128/128 (resp. SPARX-128/256), and $0 \leq a \leq 3$, with $a = 0$ corresponding to the left branch.
- $X_{(a,i)}$ ($Y_{(a,i)}$): The left (right) 16-bit input at branch a of round i where $0 \leq i \leq 32$ (resp. $0 \leq i \leq 40$) for SPARX-128/128 (resp. SPARX-128/256), $0 \leq a \leq 3$, with $a = 0$ corresponding to the left branch, and the LSBs of both $X_{(a,i)}$ and $Y_{(a,i)}$ start from the right.
- $X_{(a,i)}[i, j, \cdots, k]$: The $i, j, \cdots, k$ bits of $X_{(a,i)}$.
- $X_{(a,i)}[i : j]$: The bits from i to j of $X_{(a,i)}$, where $i \leq j$.
- w: The number of 32-bit words, i.e., $w = 4$ for a 128-bit block and $w = 8$ for a 256-bit master key.
- R^4: The iteration of 4 rounds of SPECKEY [2,3] with their corresponding key additions.
- L_w: Linear mixing layer used in SPARX with w-word block size. Thus, L_4 represents the linear mixing layer used in SPARX-128/128 and SPARX-128/256.
- $\boxplus$: Addition mod 2^{16}.
- $\oplus$: Bitwise XOR.
- $\lll q$ ($\ggg q$): Rotation of a word by q bits to the left (right).
- $\|$: Concatenation of bits.

2.1 Specifications of SPARX-128/128 and SPARX-128/256

SPARX [7,8] is a family of ARX-based Substitution-Permutation Network (SPN) block ciphers. It follows the SPN design construction while using ARX-based S-boxes instead of S-boxes based on look-up tables. The ARX-based S-boxes form a specific category of S-boxes that rely solely on addition, rotation and XOR operations to provide both non-linearity and diffusion. The SPARX family adopts the 32-bit SPECKEY ARX-based S-box (S), shown in Fig. 1, which resembles one round of SPECK-32 [2,3] with only one difference, that is, the key is added to the whole 32-bit state instead of just half the state as in SPECK-32.

For a given member of the SPARX family whose block size is n bits, the plaintext is divided into $w = n/32$ words of 32 bits each. Then, the SPECKEY S-box (S), is applied to w words in parallel, and iterated r times interleaved by the addition of independent subkeys. Then, a linear mixing layer (L_w) is applied to ensure diffusion between the words. As depicted in Fig. 1, the structure made of a key addition followed by S is called a round while the structure made of r rounds followed by L_w is called a step. Thus, the ciphertext corresponding to a given plaintext is generated by iterating such steps. The number of steps and the number of rounds in each step depend on both the block size and the key length of the cipher.

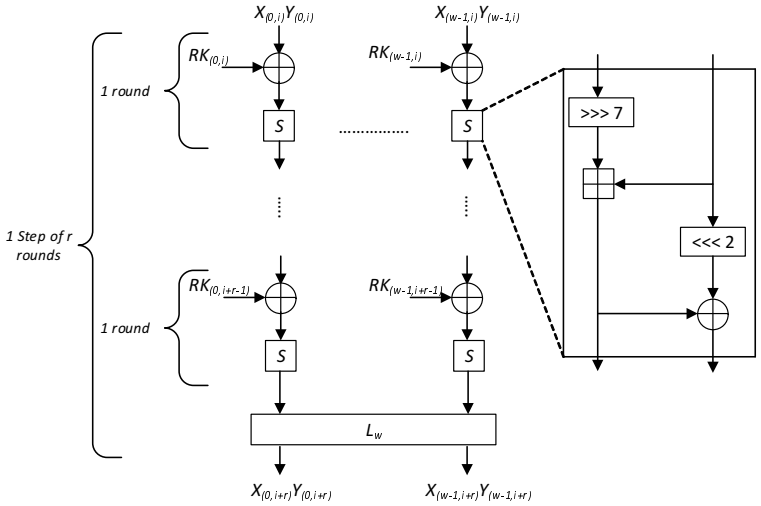


Fig. 1: SPARX structure

SPARX-128/128 and SPARX-128/256 are two members of the SPARX family which operate on 128-bit blocks using 128-bit and 256-bit keys, respectively. Both variants use 4 rounds in each step and iterate over 8 and 10 steps, i.e., the total number of rounds is 32 and 40, respectively. More precisely, in SPARX-128/128

and SPARX-128/256, 4 SPECKEY S-boxes (S) are iterated simultaneously for 4 times, while being interleaved by the addition of the round keys and then a linear mixing layer (L_4) is applied, as shown in Fig. 2. The structure of L_4 is depicted in Fig. 2.

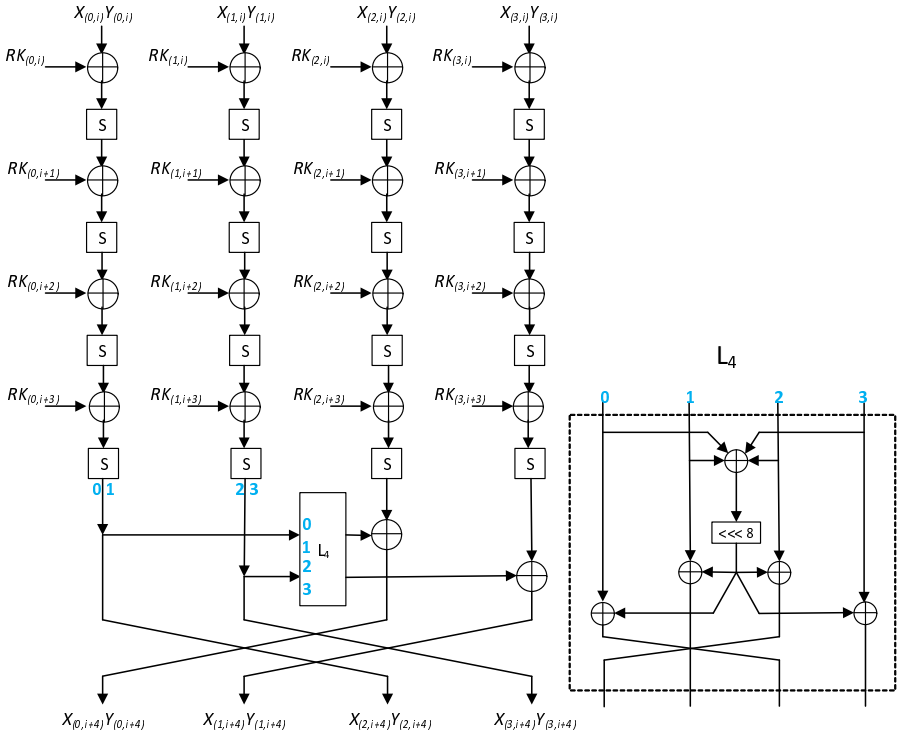


Fig. 2: SPARX-128/128 and SPARX-128/256 step structure

SPARX-128/128 key schedule. The 128-bit master key instantiates the key state, denoted by $k_0^0 \| k_1^0 \| k_2^0 \| k_3^0 \| k_4^0 \| k_5^0 \| k_6^0 \| k_7^0$. Then, the 4×32-bit round keys used in branch number 0 of the first step are extracted. Afterwards, the permutation illustrated in Fig. 3 is applied and then the 4×32-bit round keys used in branch number 1 of the first step are extracted. The application of the permutation and the extraction of the keys are interleaved until all the round keys encompassing the post-whitening ones are generated. This means that the round keys of a given branch in step j are generated first and then the key state is updated.

SPARX-128/256 key schedule. The 256-bit master key instantiates the key state, denoted by $k_0^0 \| k_1^0 \| k_2^0 \| k_3^0 \| k_4^0 \| k_5^0 \| k_6^0 \| k_7^0 \| k_8^0 \| k_9^0 \ \| k_{10}^0 \| k_{11}^0 \| k_{12}^0 \| k_{13}^0 \| k_{14}^0 \| k_{15}^0$. First, the 4×32-bit round keys used in branch number 0 of the first step are

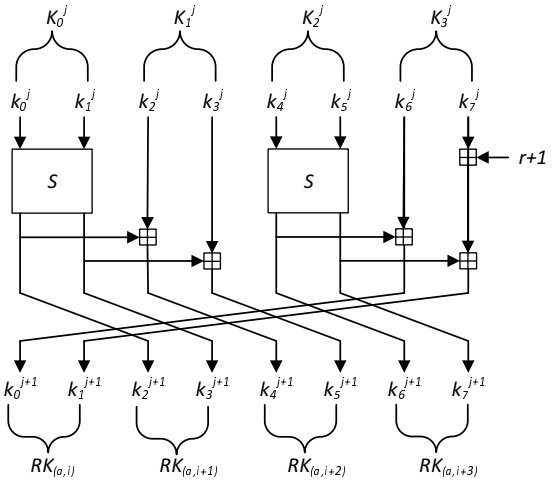


Fig. 3: SPARX-128/128 key schedule permutation, where the counter r is initialized to 0

extracted. Then, the 4×32-bit round keys used in branch number 1 of the first step are extracted. Afterwards, the permutation illustrated in Fig. 4 is applied and then the 4×32-bit round keys used in branch number 2 and 3 of the first step are extracted. The application of the permutation and the extraction of the keys are interleaved until all the round keys encompassing the post-whitening ones are generated.

3 Multidimensional Zero-Correlation Linear Cryptanalysis

In the traditional linear cryptanalysis [9], the attacker tries to find a linear relation between an input x and an output y of an n-bit block cipher function f that has the following form:

$$\Gamma_x \circ x \oplus \Gamma_y \circ y = 0,$$

where $\circ$ is a bitwise dot product operation. This linear relation has a probability p, and in this type of attack it should be far from $1/2$ or equivalently its correlation $C = 2 \times p - 1$ is not zero. Later on, Bogdanov and Rijmen [4] proposed a new technique called zero-correlation cryptanalysis which, in contrast to the linear cryptanalysis, exploits linear relations with correlation exactly zero to exclude wrong keys which lead to this linear approximation. The following lemmas are used to specify the propagation of linear masks through the different operations (XOR, branch, and S-box) that are used in the round function.

Lemma 1 *(XOR operation [4,12]): Either the three linear masks at an XOR $\oplus$ are equal or the correlation over $\oplus$ is exactly zero.*

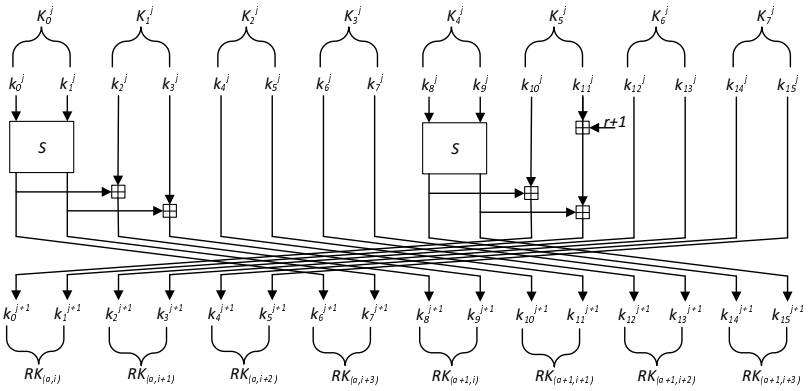


Fig. 4: SPARX-128/256 key schedule permutation, where the counter r is initialized to 0

Lemma 2 *(Branching operation [4,12]): Either the three linear masks at a branching point • sum up to 0 or the correlation over • is exactly zero.*

Lemma 3 *(S-box permutation [4,12]): Over an S-box S, if the input and output masks are neither both zero nor both nonzero, the correlation over S is exactly zero.*

To remove the burden of the high data complexity of the zero-correlation attack and the statistical independence for multiple zero-correlation linear approximations, Bogdanov *et al.* [5] proposed the multidimensional zero-correlation attack. In this technique, we have m different linear approximations with zero-correlation, where all the $l = 2^m - 1$ non-zero linear approximations involved in the spanned linear space of these m linear approximations should have zero-correlation. The zero-correlation linear approximation over r_m rounds can act as a distinguisher, then the attacker can preappend/append additional rounds called analysis rounds. The attack proceeds by gathering N plaintext/ciphertext pairs and creating array of counters $V[z]$, where $|z| = m$ bits, and initializing it to zero. Then, for each plaintext/ciphertext pair and key guess, the attacker computes the corresponding bits needed to apply the m linear approximations to compute z and increments the corresponding counter by one. Afterwards, the attacker computes the statistics T [5]:

$$T = \sum_{z=0}^{2^m-1} \frac{(V[z] - N2^{-m})^2}{N2^{-m}(1-2^{-m})} = \frac{N2^m}{(1-2^{-m})} \sum_{z=0}^{2^m-1} \left(\frac{V[z]}{N} - \frac{1}{2^m} \right)^2. \quad (1)$$

The right key has T that follows χ^2-distribution with mean $\mu_0 = l\frac{2^n-N}{2^n-1}$, and variance $\sigma_0^2 = 2l(\frac{2^n-N}{2^n-1})^2$, while the statistics for the wrong key guess follows χ^2-distribution with mean $\mu_1 = l$ and variance $\sigma_1^2 = 2l$ [5]. The number of known

plaintexts required by the attack can be estimated as follows:

$$N = \frac{2^n(Z_{1-\gamma} + Z_{1-\zeta})}{\sqrt{l/2} - Z_{1-\zeta}}, \tag{2}$$

where γ (resp. ζ) denotes the probability to incorrectly discard the right key (resp. the probability to incorrectly accept a random key as the right key) and $Z_p = \phi^{-1}(p)$ $(0 < P < 1)$, ϕ is the cumulative function of the standard normal distribution. According to the required success probabilities, the decision threshold is set to $\tau = \mu_0 + \sigma_0 Z_{1-\gamma} = \mu_1 - \sigma_1 Z_{1-\zeta}$.

4 Zero-Correlation Distinguisher of SPARX-128/128 and SPARX-128/256

In this section, we present a 24-round zero-correlation distinguisher for SPARX-128/128 and SPARX-128/256, which will be exploited later in our attacks against 26 rounds (6.5 steps out of 8) of SPARX-128/128; and 28, 29 rounds (7, 7.25 steps out of 10) for SPARX-128/256. As depicted in Fig. 5, this distinguisher begins with only branch 0 containing a linear mask α_0 at round i. Then, by propagating this linear mask 3 steps forward, and by utilizing Lemma 1 and Lemma 2, we have a linear mask α_9 applied on $X_{(3,i+12)}Y_{(3,i+12)}$. From the other side, at round $i + 24$, branch 0 has a linear mask β_0, branch 1 has no linear mask, and branch 2 and 3 have linear masks β_1 and β_2, respectively. The linear masks β_1 and β_2 are chosen such that $L_4(\beta_1, \beta_2) = (\beta_0, 0)$. This choice enables us to pass one round backward with only one word having a linear mask β_3 at branch 2. Then, following Lemma 1 and Lemma 2, we can propagate the linear masks backward for one additional round and a linear layer to end with branch 3 having a zero linear mask before applying the inverse of R^4 to obtain $X_{(3,i+12)}Y_{(3,i+12)}$. Here, R^4 can be considered as one big S-box, and hence, from Lemma 3, this linear approximation has a zero-correlation.

5 Multidimensional Zero-Correlation Cryptanalysis of SPARX-128/128 and SPARX-128/256

The following observations, which stem from the structure of SPARX-128/128 and SPARX-128/256, are exploited in our attacks.

Observation 1 *As depicted in Fig. 6a, there is a 2-round linear approximation that holds with probability 1 (*$0x0080\ 0x4001 \rightarrow 0x0004\ 0x0004$*).*

Observation 2 *As illustrated in Fig. 6b, the linear mask* $0\beta\beta0$*, where 0 and* β *denote* $0x0000$ *and 16-bit non-zero linear mask, respectively, propagates through the linear layer* L_4 *as* $\beta\beta00$*, i.e.,* $L_4(0\beta\beta0) = \beta\beta00$*.*

Observation 3 *From Observation 2 and the specification of the S-box, the 24-round distinguisher can be extended to 25-round distinguisher, as shown in Fig. 6c.*

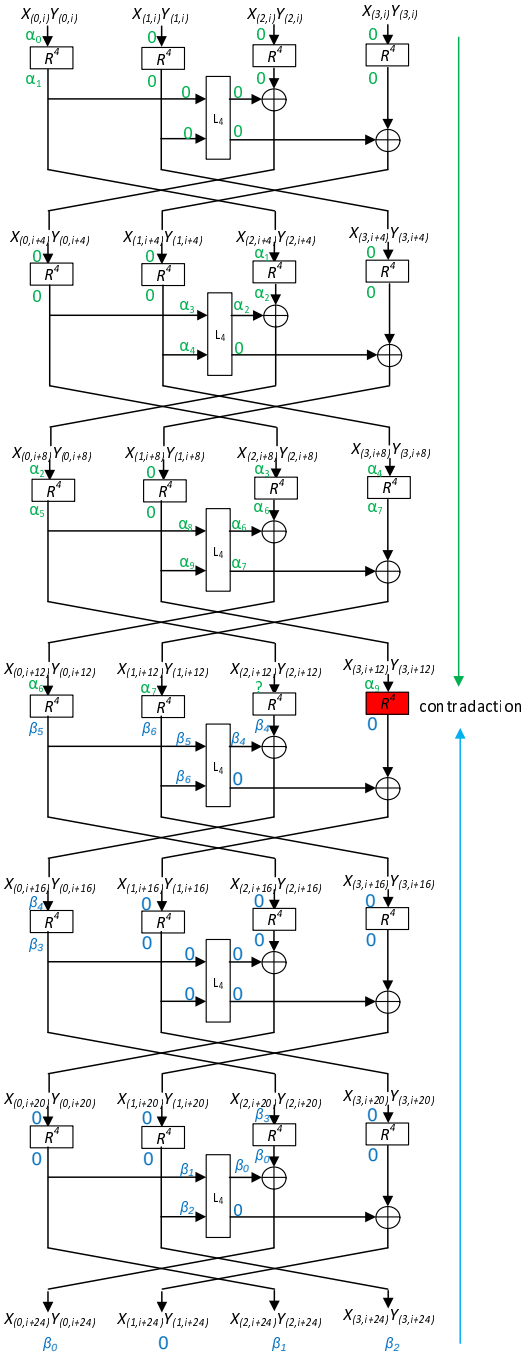
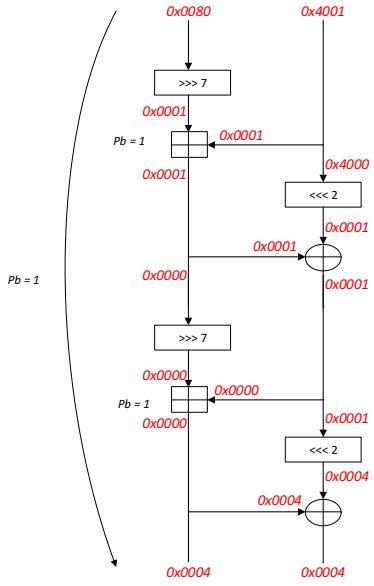
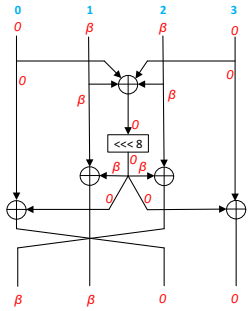


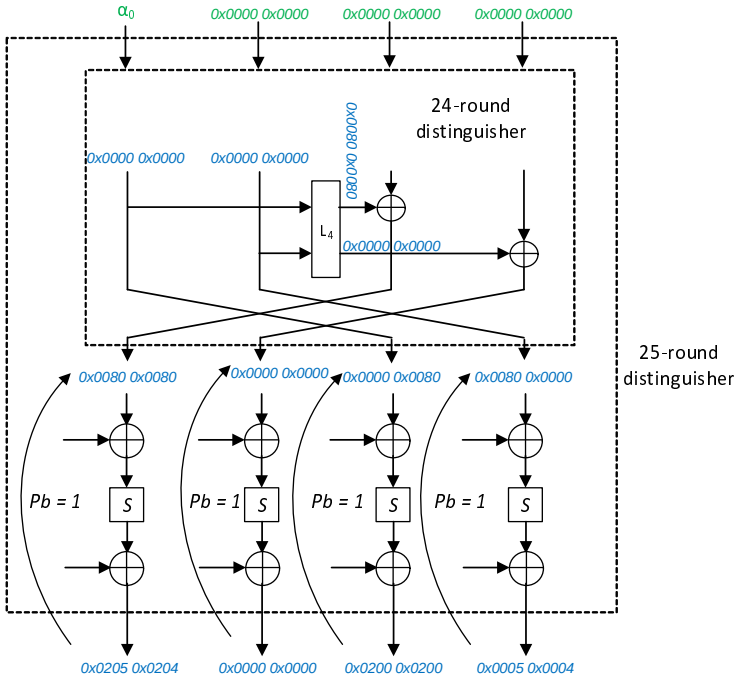
Fig. 5: A 24-round zero-correlation distinguisher of SPARX-128/128 and SPARX-128/256, where α_i, β_j are 32-bit non-zero linear masks and **0** denotes $0x0000\ 0x0000$ linear mask



(a) A 2-round linear approximation which holds with probability 1 for SPARX family



(b) The propagation of the linear mask $0\beta\beta0$ through the linear layer L_4



(c) A 25-round zero-correlation distinguisher, where α_0 is 32-bit non-zero linear mask

Fig. 6: Illustrations of Observations 1,2 and 3.

5.1 28-round Multidimensional Zero-Correlation Attack on SPARX-128/256

In this attack, and in order to maximize the number of attacked rounds, we have chosen to place the 24-round distinguisher at the bottom, and add 4 analysis rounds at the top to launch a 28-round attack against SPARX-128/256. Taking into account the key schedule relations, the top 4 analysis rounds involve all the master key bits, and in order to be able to extend 4 rounds above the distinguisher, we utilize Observation 1. In particular, we choose a specific linear mask at branch 0 at the beginning of our 24-round zero-correlation distinguisher. This specific linear mask, after propagating it backward through the linear layer L_4, enables us to bypass 2 rounds of branch 0 with probability 1 by exploiting Observation 1 to have a new extended distinguisher (the dotted one in Fig. 7).

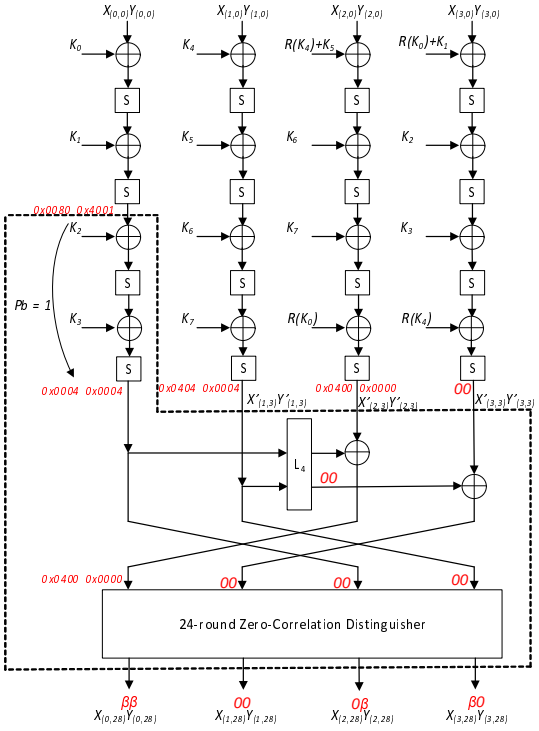


Fig. 7: A 28-round multidimensional zero-correlation linear cryptanalysis of SPARX-128/256, where 0 and β denotes $0x0000$ and 16-bit non-zero linear mask, respectively

Key Recovery. The attack proceeds by gathering enough plaintext/ciphertext

pairs. Then, we guess the round keys involved in the analysis rounds to estimate the statistics T. However, this techniques will make the time complexity of the attack exceeds the exhaustive search. Therefore, we use the partial compression technique in order to reduce the time complexity of the attack as follows:

Step 1. Allocate an array of counters $N_1[X_1]$ and initialize it to zeros, where $X_1 = X_{(0,0)}Y_{(0,0)}||X_{(1,0)}Y_{(1,0)}\ ||X_{(2,0)}Y_{(2,0)}||(X_{(0,28)} \oplus Y_{(0,28)} \oplus Y_{(2,28)} \oplus X_{(3,28)})$, i.e., $|X_1| = 112$ bits. Then, from the gathered plaintext/ciphertext pairs compute X_1 and increment the corresponding counter. Since all the non-zero 16-bit linear masks in the ciphertext are equal β, then, we can store the XOR of $(X_{(0,28)} \oplus Y_{(0,28)} \oplus Y_{(2,28)} \oplus X_{(3,28)})$ instead of storing each one separately to apply the linear mask β.

Step 2. Allocate an array of counters $N_2[X_2]$ and initialize it to zeros, where $X_2 = X_{(0,0)}Y_{(0,0)}||X_{(1,3)}[0,1,7:15]Y_{(1,3)}[0:10]\ ||X_{(2,0)}Y_{(2,0)}||(X_{(0,28)} \oplus Y_{(0,28)} \oplus Y_{(2,28)} \oplus X_{(3,28)})$, i.e., $|X_2| = 102$ bits. Then, guess K_4, K_5, K_6 and partially encrypt X_1 to compute X_2 and add the corresponding counter $N_1[X_1]$ to $N_2[X_2]$.

Step 3. Allocate an array of counters $N_3[X_3]$ and initialize it to zeros, where $X_3 = X_{(0,0)}Y_{(0,0)}||X'_{(1,3)}[2,10]Y'_{(1,3)}[2]\ ||X_{(2,0)}Y_{(2,0)}||(X_{(0,28)} \oplus Y_{(0,28)} \oplus Y_{(2,28)} \oplus X_{(3,28)})$, i.e., $|X_3| = 83$ bits. Then, guess 22 bits of K_7 ($K_7[0:10,16,17,23:31] \equiv k_{14}[0,1,7:15], k_{15}[0:10]$) and partially encrypt X_2 to compute X_3 and add the corresponding counter $N_2[X_2]$ to $N_3[X_3]$. Since the linear mask on $X'_{(1,3)}Y'_{(1,3)}$ is $0x0404\ 0x0004$, i.e., we need to compute only 3 bits of $X'_{(1,3)}Y'_{(1,3)}$, and we need only to know 22 bits of $X_{(1,3)}[0,1,7:15]Y_{(1,3)}[0:10]$ and 22 bits of K_7 to compute this linear mask.

Step 4. Allocate an array of counters $N_4[X_4]$ and initialize it to zeros, where $X_4 = X_{(0,0)}Y_{(0,0)}||X'_{(1,3)}[2,10]Y'_{(1,3)}[2]\ ||X_{(2,3)}[0,1,7:15]Y_{(2,3)}[0:10]||(X_{(0,28)} \oplus Y_{(0,28)} \oplus Y_{(2,28)} \oplus X_{(3,28)})$, i.e., $|X_4| = 73$ bits. Then, guess the remaining 10 bits of K_7 and partially encrypt X_3 to compute X_4 and add the corresponding counter $N_3[X_3]$ to $N_4[X_4]$.

Step 5. Allocate an array of counters $N_5[X_5]$ and initialize it to zeros, where $X_5 = X_{(0,0)}Y_{(0,0)}||X'_{(1,3)}[2,10]Y'_{(1,3)}[2]\ ||X'_{(2,3)}[10]||(X_{(0,28)} \oplus Y_{(0,28)} \oplus Y_{(2,28)} \oplus X_{(3,28)})$, i.e., $|X_5| = 52$ bits. Then, guess 22 bits of $R(K_0)$ ($R(K_0)[0:10,16,17,23:31]$) and partially encrypt X_4 to compute X_5 and add the corresponding counter $N_4[X_4]$ to $N_5[X_5]$.

Step 6. Allocate an array of counters $N_6[X_6]$ and initialize it to zeros, where $X_6 = X_{(0,1)}[0:5,7:15]Y_{(0,1)}[0:14]||X'_{(1,3)}[2,10]Y'_{(1,3)}[2]\ ||X'_{(2,3)}[10]||(X_{(0,28)} \oplus Y_{(0,28)} \oplus Y_{(2,28)} \oplus X_{(3,28)})$, i.e., $|X_6| = 50$ bits. Then, guess the remaining 10 bits of $R(K_0)$ and partially encrypt X_5 to compute X_6 and add the corresponding counter $N_5[X_5]$ to $N_6[X_6]$.

Step 7. Allocate an array of counters $N_7[X_7]$ and initialize it to zeros, where $X_7 = X_{(0,2)}[7]Y_{(0,2)}[0,14]||X'_{(1,3)}[2,10]Y'_{(1,3)}[2]\ ||X'_{(2,3)}[10]||(X_{(0,28)} \oplus Y_{(0,28)} \oplus Y_{(2,28)} \oplus X_{(3,28)})$, i.e., $|X_7| = 23$ bits. Then, guess 30 bits of K_1 ($k_2[0:5,7:15], k_3[0:14]$) and partially encrypt X_6 to compute X_7 and add the corresponding counter $N_6[X_6]$ to $N_7[X_7]$.

The steps of the key recovery phase are summarized in Table 2, whereas the

second column gives the keys to be guessed in each step. The third column presents the saved state in each step after the partial encryption, the fourth column is the counter size for each obtained state in the corresponding step, and the fifth column quantifies the time complexity of each step measured in 28-round encryption by considering the number of S-box accesses.

Table 2: Key recovery process of the attack on 28-round SPARX-128/256

Step	Guessed keys	Obtained state	Size	Time complexity
1	†	X_1	112	‡
2	K_4, K_5, K_6	X_2	102	$2^{112} \times 2^{3\times 32} \times \frac{3}{28\times 4} \approx 2^{202.8}$
3	$K_7[0:10, 16, 17, 23:31]$	X_3	83	$2^{102} \times 2^{96+22} \times \frac{1}{28\times 4} \approx 2^{213.2}$
4	$K_7[11:15, 18:22]$	X_4	73	$2^{83} \times 2^{118+10} \times \frac{3}{28\times 4} \approx 2^{205.8}$
5	$R(K_0)[0:10, 16, 17, 23:31]$	X_5	52	$2^{73} \times 2^{128+22} \times \frac{1}{28\times 4} \approx 2^{216.2}$
6	$R(K_0)[11:15, 18:22]$	X_6	50	$2^{52} \times 2^{150+10} \times \frac{1}{28\times 4} \approx 2^{205.2}$
7	$K_1[0:14, 16:21, 23:31]$	X_7	23	$2^{50} \times 2^{160+30} \times \frac{1}{28\times 4} \approx 2^{233.2}$

†: No additional key guesses needed, ‡: Negligible complexity.

After Step 7, we have guessed 190 key bits gK from the master key and evaluated X_7, that contains all the 23 bits involved in computing the zero-correlation masks. Therefore, to recover the master key, the following steps are performed:

1. Allocate an array of counters $V[z]$, where $|z| = 16$ bits.
2. For 2^{23} values of X_7
   (a) Evaluate all 16 basis zero-correlation masks on X_7 and calculate z.
   (b) Update the counter $V[z]$ by $V[z] = V[z] + N_7[X_7]$.
3. For each guessed key gK, compute $T_{gK} = \frac{N \times 2^{16}}{1-2^{-16}} \sum_{z=0}^{2^{16}-1} \left(\frac{V[z]}{N} - \frac{1}{2^{16}} \right)^2$.
4. If $T_k < \tau$, then the guessed values of gK are key candidates.
5. Exhaustively search all the remaining key candidates with 2^{66} values for the 66 bits of the key that are not retrieved by the above steps of the attack using 2 plaintext/ciphertext pairs.

Attack complexity. Since the beginning of distinguisher has a specific linear mask and the end of the distinguisher has a variable 16-bit linear mask β, then

$m = 16$, and hence $l = 2^{16} - 1$. Here, we set $\gamma = 2^{-2.7}$ and $\zeta = 2^{-26}$ and hence we have $z_{1-\gamma} \approx 1$ and $z_{1-\zeta} \approx 5.54$. According to Equation (2), the data complexity is about $2^{123.25}$ known plaintexts. The time complexity of the attack is dominated by two parts. The first part is the time required to reduce the key search space which can be computed from Table 2. The second part is the time required to retrieve the whole master key by exhaustively searching the remaining $2^{190} \times 2^{-26} = 2^{164}$ key candidates with the 2^{66} key bits not involved in the attack using 2 plaintext/ciphertext pairs. Therefore, the time complexity of the attack is $2^{233.2} + 2 \times 2^{164} \times 2^{66} \approx 2^{233.5}$ 28-round encryptions.

29-round Zero-Correlation Attack on SPARX-128/256. The above attack can be extended one more round to launch a key recovery attack against 29-round of SPARX-128/256 with the full code book. This extra round can be obtained by selecting the linear masks at the end of the distinguisher as in Observation 3 to convert the 24-round distinguisher to 25-round distinguisher. However, at this time we will use only one zero-correlation linear approximation. Therefore, we require the full code book. The time complexity of the attack is dominated by Step 7, and it will be $2^{227.2}$ instead of $2^{233.2}$ because we store only 10 bits instead of 16 bits at the end of the distinguisher.

5.2 26-round Multidimensional Zero-Correlation Attack on SPARX-128/128

As depicted in Fig. 8, in this attack we use the 25-round zero-correlation distinguisher obtained by utilizing Observation 3. Then, we append an additional round at the bottom of the distinguisher. In the previous attack, the analysis rounds were placed above the distinguisher, therefore, the relation of the round keys to the master key was straightforward and we use the master key relations in the attack from the beginning. On the other hand, in here, we place the analysis rounds at the bottom of the distinguisher, and hence the relation of the round keys to the master key is not trivial. Therefore, we will perform the attack on the round keys. Then, we will explain how to recover the master key from the recovered round keys. Here, in order to balance between the time and data complexity, we choose α_0 having linear masks in the first 30-bit only.

Key Recovery. Similar to the previous attack, we first gather N plaintext/ciphertext pairs, and then proceed as follows:

Step 1. Allocate an array of counters $N_1[X_1]$ and initialize it to zeros, where $X_1 = X_{(0,0)}[0:13]Y_{(0,0)}[0:15]||X_{(0,26)}[0:13]Y_{(0,26)}[2:13]\ ||X_{(2,26)}[0:4,11]$ $Y_{(2,26)}[2:4,11]||X_{(3,26)}[0:13]Y_{(3,26)}[2:13]$, i.e., $|X_1| = 92$ bits. Then, from the N plaintext/ciphertext pairs compute X_1 and increment the corresponding counter.

Step 2. Allocate an array of counters $N_2[X_2]$ and initialize it to zeros, where $X_2 = X_{(0,0)}[0:13]Y_{(0,0)}[0:15]||X_{(0,26)}[0:13]Y_{(0,26)}[2:13]\ ||X_{(2,25)}[9]Y_{(2,25)}[9]$

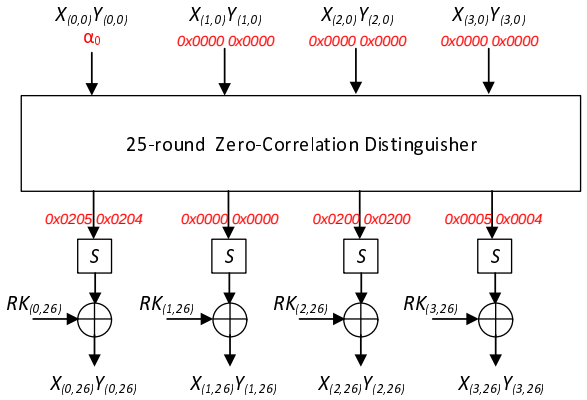


Fig. 8: A 26-round multidimensional zero-correlation linear cryptanalysis of SPARX-128/128

$||X_{(3,26)}[0:13]Y_{(3,26)}[2:13]$, i.e., $|X_2| = 84$ bits. Then, guess $RK_{(2,26)}[2:4,11,16:20,27]$ and partially decrypt X_1 to compute X_2 and add the corresponding counter $N_1[X_1]$ to $N_2[X_2]$.

Step 3. Allocate an array of counters $N_3[X_3]$ and initialize it to zeros, where $X_3 = X_{(0,0)}[0:13]Y_{(0,0)}[0:15]||X_{(0,26)}[0:13]Y_{(0,26)}[2:13]\ ||X_{(2,25)}[9]Y_{(2,25)}[9]$ $||X_{(3,25)}[0,2]Y_{(3,25)}[2]$, i.e., $|X_3| = 61$ bits. Then, guess $RK_{(3,26)}[2:13,16:29]$ and partially decrypt X_2 to compute X_3 and add the corresponding counter $N_2[X_2]$ to $N_3[X_3]$.

Step 4. Allocate an array of counters $N_4[X_4]$ and initialize it to zeros, where $X_4 = X_{(0,0)}[0:13]Y_{(0,0)}[0:15]||X_{(0,25)}[0,2,9]Y_{(0,25)}[2,9]\ ||X_{(2,25)}[9]Y_{(2,25)}[9]$ $||X_{(3,25)}[0,2]Y_{(3,25)}[2]$, i.e., $|X_4| = 40$ bits. Then, guess $RK_{(0,26)}[2:13,16:29]$ and partially decrypt X_3 to compute X_4 and add the corresponding counter $N_3[X_3]$ to $N_4[X_4]$.

To determine the surviving round key candidates, we proceeds as in the previous attack in Section 5.1 with $m = 30$, and hence $|z| = 30$ bits. Moreover, instead of using X_7, we use X_4. The number of surviving round key candidates is $2^{62} \times 2^{-\zeta}$. To retrieve the master key, we will show how to retrieve the 128-bit key after applying the key permutation 24 times, i.e., $K_0^{24}||K_1^{24}||K_2^{24}||K_3^{24}$. Then, we can revert the key schedule permutation 24 times to retrieve the master key. We have retrieved $RK_{(0,26)}[2:13,16:29]$ which allows us to deduce $K_2^{24}[2:13,16:29]$, see Fig. 9. Retrieving the remaining 102 bits of $K_0^{24}||K_1^{24}||K_2^{24}||K_3^{24}$ can be done as follows:

1. We guess K_0^{24}, K_3^{24} and the remaining 6 bits of K_2^{24} to compute $RK_{(1,25)}$, $RK_{(1,27)}$, $RK_{(2,25)}$, $RK_{(2,26)}$. Hence in total we have $2^{62-\zeta+32+32+6-10=122-\zeta}$ remaining key candidates for $K_0^{24}, K_2^{24}, K_3^{24}, RK_{(3,26)}[2:13,16:29], RK_{(1,25)}$, $RK_{(1,27)}$, $RK_{(2,25)}$, because we have 10-bit filter on $RK_{(2,26)}[2:4,11,16:20,27]$.

2. We guess the remaining 6 bits of $RK_{(3,26)}$ to compute $RK_{(2,24)}, RK_{(1,26)}, K_1^{24}$. Therefore, in total we have $2^{122-\zeta+6}$ key candidates for $K_0^{24}, K_1^{24}, K_2^{24}, K_3^{24}$.
3. We apply the inverse of the key permutation 24 times to retrieve $2^{122-\zeta+6}$ key candidates for K.
4. We test the remaining key candidates using one plaintext/ciphertext pairs to identify the correct key.

Attack complexity. Here, we set $m = 30$ (and hence $l = 2^{30} - 1$), $\gamma = 2^{-2.7}$, and $\zeta = 2^{-26}$. Thus $z_{1-\gamma} \approx 1$ and $z_{1-\zeta} \approx 5.54$. The data complexity is $2^{116.2}$ known plaintexts, which can be computed from equation (2). In this case, the time complexity of the attack is dominated by three parts. The first part is the time required to reduce the key search space which is dominated by Step 4 and equals $2^{61} \times 2^{10+26+26} \times \frac{1}{26\times 4} \approx 2^{116.3}$. The second part is the time required to retrieve the whole master key and equals $2^{62-26+32+32+6} \times \frac{3}{26\times 4} + 2^{122-26+6} \times \frac{2}{26\times 4} + 2^{122-26+6} \times \frac{24\times 2}{26\times 4} + 2^{102} \approx 2^{103}$. The third part is the time required by the data collection phase which is equal to $2^{116.2}$. Therefore, the time complexity of the attack is $2^{116.3} + 2^{103} + 2^{116.2} \approx 2^{117.25}$ 26-round encryptions.

Remark: It is worth noting that the above zero correlation attacks are also applicable to 15 rounds of SPARX-64/128 using the zero correlation distniguisher shown in Fig. 10. The details of this attack are omitted from this version of the paper because of the space limitations and also since attacking 15 rounds of SPARX-64/128 does not violate the designer's security claims for this version of the cipher where the designers already reported a 15-round attack against it, see also [1].

6 Conclusion

The best attacks against SPARX-128/128 and SPARX-128/256, which are reported by the designers of the SPARX family in [7], are integral attacks which cover 22 and 24 rounds. Therefore, the designers concluded that the security margin is more than 30%. In this paper, we presented 24- and 25-round zero-correlation distinguishers that are used to launch key recovery attacks against 28, 29 rounds (7, 7.25 out of 10 steps) of SPARX-128/256 and 26 rounds (6.5 out of 8 steps) of SPARX-128/128. To the best of our knowledge these are the first third party attacks against SPARX-128/128 and SPARX-128/256. While these attacks contradict the security margin claim of the designers, they still may not impact the practical security of the ciphers.

References

1. Ahmed Abdelkhalek, Mohamed Tolba, and Amr M. Youssef. Impossible Differential Attack on Reduced Round SPARX-64/128. In Marc Joye and Abderrahmane Nitaj, editors, *AFRICACRYPT 2017: 9th International Conference on Cryptology in Africa, Dakar, Senegal, May 24-26, 2017, Proceedings*, pages 135–146. Springer International Publishing, Cham, 2017.
2. Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. The SIMON and SPECK Families of Lightweight Block Ciphers. Cryptology ePrint Archive, Report 2013/404, 2013. http://eprint.iacr.org/2013/404.
3. Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. SIMON and SPECK: Block Ciphers for the Internet of Things. Cryptology ePrint Archive, Report 2015/585, 2015. http://eprint.iacr.org/2015/585.
4. Andrey Bogdanov, Huizheng Geng, Meiqin Wang, Long Wen, and Baudoin Collard. Zero-Correlation Linear Cryptanalysis with FFT and Improved Attacks on ISO Standards Camellia and CLEFIA. In Tanja Lange, Kristin Lauter, and Petr Lisoněk, editors, *Selected Areas in Cryptography – SAC 2013: 20th International Conference, Burnaby, BC, Canada, August 14-16, 2013, Revised Selected Papers*, pages 306–323. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.
5. Andrey Bogdanov, Gregor Leander, Kaisa Nyberg, and Meiqin Wang. Integral and Multidimensional Linear Distinguishers with Correlation Zero. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology – ASIACRYPT 2012: 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, pages 244–261. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
6. Joan Daemen and Vincent Rijmen. The Wide Trail Design Strategy. In Bahram Honary, editor, *Cryptography and Coding: 8th IMA International Conference Cirencester, UK, December 17–19, 2001 Proceedings*, pages 222–238. Springer Berlin Heidelberg, 2001.
7. Daniel Dinu, Léo Perrin, Aleksei Udovenko, Vesselin Velichkov, Johann Großschädl, and Alex Biryukov. Design Strategies for ARX with Provable Bounds: SPARX and LAX. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *Advances in Cryptology – ASIACRYPT 2016: 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, pages 484–513. Springer Berlin Heidelberg, 2016.
8. Daniel Dinu, Lo Perrin, Aleksei Udovenko, Vesselin Velichkov, Johann Groschdl, and Alex Biryukov. Design Strategies for ARX with Provable Bounds: SPARX and LAX (Full Version). Cryptology ePrint Archive, Report 2016/984, 2016. http://eprint.iacr.org/2016/984.
9. Mitsuru Matsui and Atsuhiro Yamagishi. A New Method for Known Plaintext Attack of FEAL Cipher. In RainerA. Rueppel, editor, *Advances in Cryptology – EUROCRYPT 92*, volume 658 of *Lecture Notes in Computer Science*, pages 81–91. Springer Berlin Heidelberg, 1993.
10. Ling Sun, Kai Fu, and Meiqin Wang. Improved Zero-Correlation Cryptanalysis on SIMON. In Dongdai Lin, XiaoFeng Wang, and Moti Yung, editors, *Information Security and Cryptology: 11th International Conference, Inscrypt 2015, Beijing, China, November 1-3, 2015, Revised Selected Papers*, pages 125–143. Springer International Publishing, Cham, 2016.

11. Yosuke Todo. Structural Evaluation by Generalized Integral Property. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology – EUROCRYPT 2015: 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, pages 287–314. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.
12. Yanfeng Wang and Wenling Wu. Improved Multidimensional Zero-Correlation Linear Cryptanalysis and Applications to LBlock and TWINE. In Willy Susilo and Yi Mu, editors, *Information Security and Privacy: 19th Australasian Conference, ACISP 2014, Wollongong, NSW, Australia, July 7-9, 2014. Proceedings*, pages 1–16. Springer International Publishing, Cham, 2014.
13. Long Wen, Meiqin Wang, Andrey Bogdanov, and Huaifeng Chen. Multidimensional zero-correlation attacks on lightweight block cipher HIGHT: Improved cryptanalysis of an ISO standard. *Information Processing Letters*, 114(6):322 – 330, 2014.
14. Hong Xu, Ping Jia, Geshi Huang, and Xuejia Lai. Multidimensional Zero-Correlation Linear Cryptanalysis on 23-Round LBlock-s. In Sihan Qing, Eiji Okamoto, Kwangjo Kim, and Dongmei Liu, editors, *Information and Communications Security: 17th International Conference, ICICS 2015, Beijing, China, December 9-11, 2015, Revised Selected Papers*, pages 97–108. Springer International Publishing, Cham, 2016.

A Key Schedule Relations for SPARX-128/128

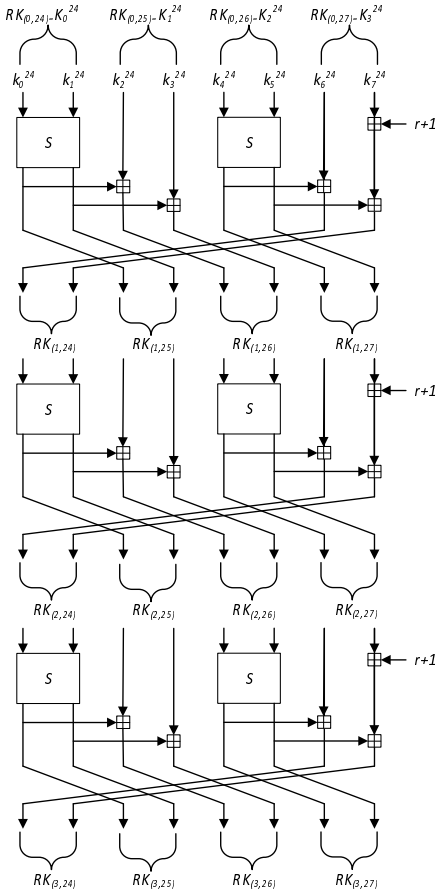


Fig. 9: Key secluded relations of SPARX-128/128

B Zero-Correlation Distinguisher for SPARX-64/128

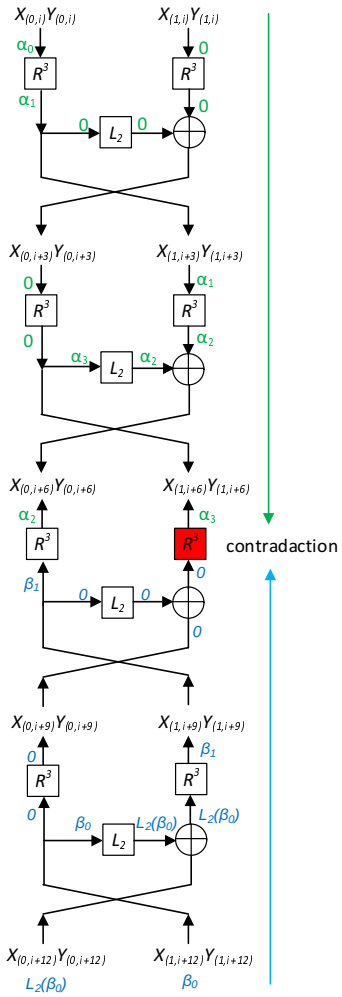


Fig. 10: A 12-round zero-correlation distinguisher of SPARX-64/128, where α_i, β_j are 32-bit non-zero linear masks and **0** denotes $0x0000\ 0x0000$ linear mask