Related-key Impossible Differential Analysis of Full Khudra

Qianqian Yang^{1,2,3}, Lei Hu^{1,2,**}, Siwei Sun^{1,2}, Ling Song^{1,2}

^{1}State Key Laboratory of Information Security, Institute of Information Engineering,
Chinese Academy of Sciences, Beijing 100093, China
^{2}Data Assurance and Communication Security Research Center,
Chinese Academy of Sciences, Beijing 100093, China
^{3}University of Chinese Academy of Sciences, Beijing 100049, China
{qqyang13,hu,swsun,lsong}@is.ac.cn

Abstract. Khudra is a 18-round lightweight block cipher proposed by Souvik Kolay and Debdeep Mukhopadhyay in the SPACE 2014 conference which is applicable to Field Programmable Gate Arrays (FPGAs). In this paper, we obtain 2^{16} 14-round related-key impossible differentials of Khudra, and based on these related-key impossible differentials for 32 related keys, we launch an attack on the full Khudra with data complexity of 2^{63} related-key chosen-plaintexts, time complexity of about $2^{68.46}$ encryptions and memory complexity of 2^{64}.

Keywords: Lightweight, Block Cipher, Khudra, Related-key, Impossible Differential Cryptanalysis

1 Introduction

Recently, lightweight block ciphers, which could be widely used in small embedded devices such as RFIDs and sensor networks, are becoming more and more popular. Due to the strong demand from industry, a lot of lightweight block ciphers are proposed in recent years, such as PRESENT[1], LED[2], LBlock[3], PRINCE[4], and two lightweight block ciphers SIMON and SPECK[5], designed by the U.S. National Security Agency.

Khudra[6] is a new lightweight block cipher which was recently proposed by Souvik Kolay and Debdeep Mukhopadhyay in the SPACE 2014 conference. While there are many lightweight block ciphers designed based on rationales and popular techniques which are suitable for implementation on Application Specific Integrated Circuits (ASICs), a few of block ciphers are applicable to Field Programmable Gate Arrays (FPGAs) due to the underlying FPGA architecture. Khudra was designed by using new methods and design criteria to enable it capable of operating on FPGAs. It is a 18-round generalized Feistel block cipher with 64-bit block size and 80-bit key size.

Security is crucially important for a cipher, and there are many different attacks on block ciphers. Thus a new cipher must be able to resist against all

[⋆⋆] The corresponding author.

know attacks. Differential cryptanalysis[7] and linear cryptanalysis[8] are two of the most basic and effective attacks on block ciphers. Based on differential cryptanalysis a bunch of variants of differential analysis have been developed, such as related-key attack[9], truncated differential attack[10], boomerang attack and impossible differential attack[11]. Those attacks are chosen plaintext attacks based on a differential distinguisher which uses pairs of plaintexts.

Impossible differential attack, which was independently proposed by Biham *et al.* [11] and Knudsen[12], is one of the well-know attacks on block ciphers. With its development, there are several approaches have been proposed to derive truncated impossible differentials of block ciphers/structures effectively such as the $\mathcal{U}$-method[13], UID-method[14] and the extended tool of generalized upon to the former two methods by Wu and Wang proposed in Indocrypt 2012[15]. Unlike traditional differential cryptanalysis, impossible differential attack starts with finding an input difference that result in an output difference with probability 0. It is an attack that aborts wrong key candidates by searching the plaintext-ciphertext pairs which meet the input and output differences of the impossible differential. Related-key attacks[9] allow a cryptanalyst to choose appropriate relation between keys and then to predict the encryptions under these keys. Related-key impossible differential attack[16] is a combination of the two above attacks.

Our Contributions. Based on the generalized Feistel structure of the block cipher Khudra, we obtain 2^{16} related-key impossible differential characteristics for the 14-round Khudra. By adding two rounds before and after the impossible differentials respectively, we propose a related-key impossible differential attack on full Khudra, with data complexity of 2^{63} chosen-plaintexts, time complexity of about $2^{68.46}$ encryptions and memory complexity of 2^{64}.

Organization of this paper. In Section 2, we briefly describe the Khudra block cipher. In Section 3, we obtain 14-round related-key impossible differential characteristics and propose the related-key impossible differential attack on the full Khudra. Finally, Section 4 concludes this study.

2 Description of Block Cipher Khudra

2.1 Notation

The following notations are used in this paper:
P_i: the i-th 16-bit of 64-bit plaintext, $0 \leq i < 4$;
C_i: the i-th 16-bit of 64-bit ciphertext, $0 \leq i < 4$;
ΔX: the XOR difference of X and $X^{'}$;
ΔI_r^i: the XOR difference of the i-th 16-bit of 64-bit input of the r-th round, $1 \leq i \leq 4$;
ΔO_r^i: the XOR difference of the i-th 16-bit of 64-bit output of the r-th round, $1 \leq i \leq 4$;
$\oplus$: bitwise exclusive OR (XOR);
$x||y$: bit string concatenation of x and y;
ΔF_r^i: the XOR difference after the i-th F-function of the r-th round, $1 \leq i \leq 2$.

2.2 Description of Khudra

Khudra is a new lightweight block cipher using "Generalized type-2 transformations" of Feistel Structure (GFS) on 64-bit blocks and 80-bit keys. The cipher has 18 rounds that each round has two 16×16 F-Function. The structure of the cipher is depicted in Fig.1.

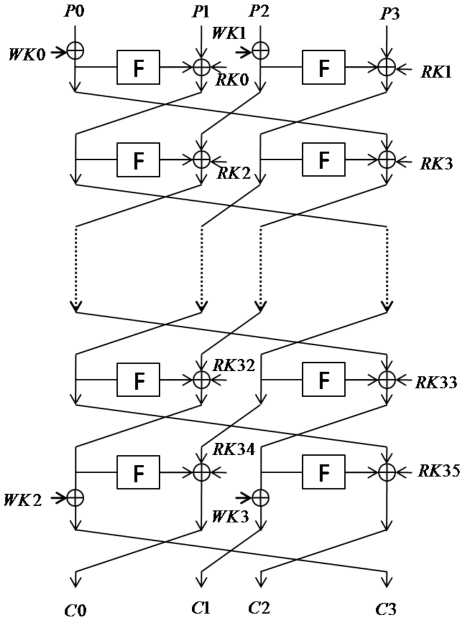

Fig. 1: Structure of Khudra

The F-function The F-Function has a similar structure used in Khudra, i.e., the same 4 branch type-2 generalized Feistel Structure. It uses 12 copies of a 4×4 S-box to provide non-linearity, see Fig.2.

The S-box Khudra uses the S-box of the block cipher PRESENT for its higher algebraic degree and low differential and linear probability[17]. The S-box is given in Table 1.

Table 1: S-box of Khudra

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

Key Schedule Function The 80-bit master key for the block cipher Khudra is $(k_0, k_1, k_2, k_3, k_4)$, the sizes of $k_i (i = 0, 1, 2, 3, 4)$ are 16-bit. The key scheduling

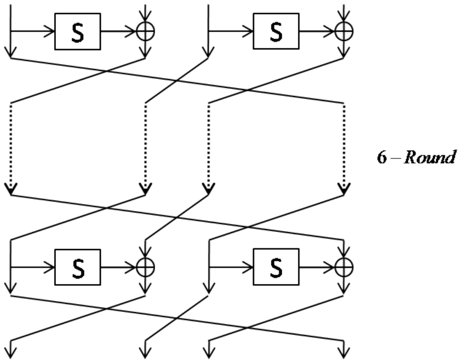

Fig. 2: F-function

part generates 16-bit round-keys $RKi(0 \leq i < 36)$ and 16-bit whitening keys $WKi(0 \leq i < 4)$, see Algorithm 1, where RC_i is the 16-bit round constant and $i_{(6)}$ is the 6-bit representation of the round counter i.

Algorithm 1 Key Scheduling $(k_0, k_1, k_2, k_3, k_4)$

```
$WK0 \leftarrow k_0, WK1 \leftarrow k_1, WK2 \leftarrow k_3, WK3 \leftarrow k_4,$;
for $i = 1$ to 35 do
    $RCi \leftarrow \{0||i_{(6)}||00||i_{(6)}||0\}$;
    $RKi \leftarrow k_{i\ mod\ 5} \oplus RCi$;
end for
```

3 Related-key Impossible Differential Attack on Full Khudra

In this section, we describe a related-key impossible differential attack on the full Khudra. We first give the related-key impossible differential characteristics, and then give the attack on the full Khudra in detail.

3.1 Related-key Impossible Differential Characteristic of Block Cipher Khudra

According to the key schedule of Khudra, the difference values of round-keys are cyclic, that is, $\Delta RKi = \Delta RKi\ mod\ 5$. Assuming that one of the 5 16-bit keys have a difference Δ (in our attack, the difference just is Δk_0), we obtain the 14-round related-key impossible differentials, the details are given in Fig.3.

With the input difference $\Delta input = (0, 0, 0, \Delta k_0)$ and the related keys difference $\Delta K = (\Delta k_0, 0, 0, 0, 0)$, we deduce that $\Delta I_7^4 = \Delta' \neq 0$ and $\Delta I_7^3 = \Delta I_8^2 = 0$.

By the same taken, with the output difference $\Delta output = (0, \Delta k_0, 0, 0)$, we deduce that $\Delta O_7^4 = 0$, $\Delta O_8^2 = 0$ and $\Delta I_8^1 = \Delta \neq 0$. Thus there are two contradictions in the second F-function of the 7th round and the first F-function of the 8th round, respectively. As the value of Δk_0 is arbitrary, there are 2^{16} values of Δk_0. For one Δk_0, there is one 14-round impossible differential characteristic.

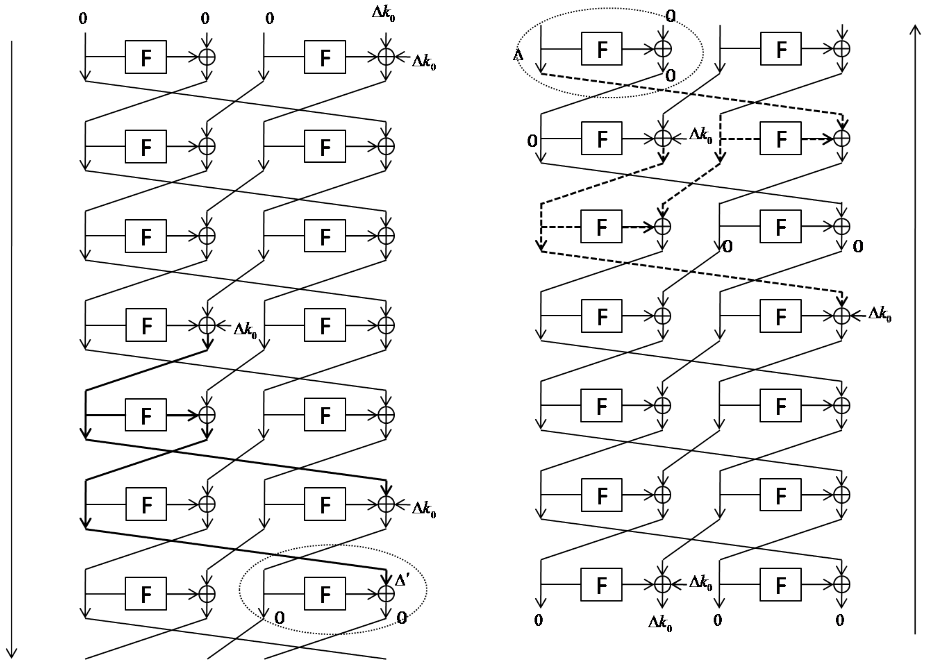

Fig. 3: Related-key impossible differentials for 14-round Khudra. (No difference in fine line. Differences in the overstriking line and dotted line are non-zero.)

3.2 Related-key Impossible Differential Analysis on Block Cipher Khudra

With these 14-round related-key impossible differentials on rounds 3-16, we can attack full Khudra. This is clarified in Fig.4.

Choosing $N_r = 32$ related keys by some strategies, we can obtain $N_\Delta = C_{N_r}^2 \approx (N_r)^2/2 \approx 2^9$ different values of $\Delta K = (\Delta k_0, 0, 0, 0, 0)$ i.e. difference of related-key. We call the 32 related keys as $K^{(i)}, 0 < i \leq 32$.

-Data Collection Phase. For a fixed difference of key, by expanding two rounds before and after the 14-round impossible differential path respectively, we deduce that the plaintext difference is $\Delta P = (\Delta P_0,\ \Delta P_1, \Delta P_2, \Delta P_3) = (\Delta k_0, 0, *, *)$

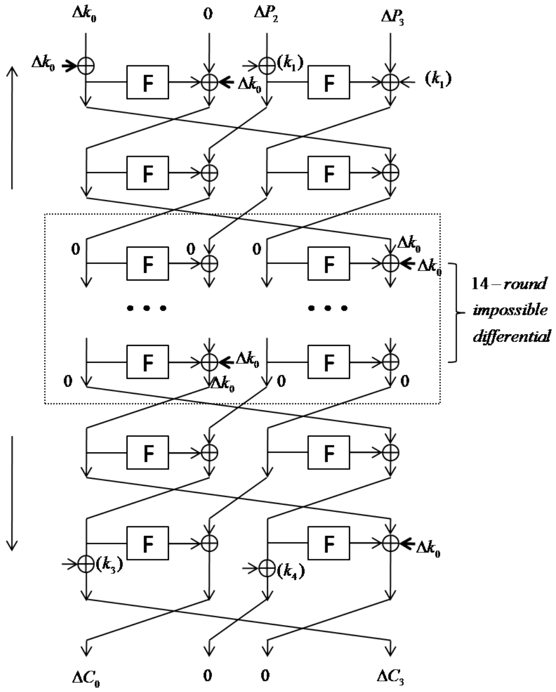

Fig. 4: Related-key impossible differential attack against full Khudra

where Δk_0 is the corresponding value and the ciphertext difference is $\Delta C = (\Delta C_0, \Delta C_1, \Delta C_2, \Delta C_3) = (*, 0, 0, *)$.

For each $K^{(i)}$, we construct corresponding N_s structures of plaintexts. In each structure the 32 bits in (P_0, P_1) are fixed and other 32 bits are traversed. Thus we get $N_r \times N_s \times 2^{32}$ chosen-plaintexts.

-Key Recovery Phase. In this attack, we should guess 64-bit values of $k_0||k_1||k_3||k_4$. Thus we build a table D storing 2^{64} values of $k_0||k_1||k_3||k_4$. Besides, we build a table F storing $\Delta F = F(F(x) \oplus y) \oplus F(F(x) \oplus y')$ and another table F' storing $\Delta F' = F(x) \oplus F(x')$.

Supposing K is the right key, two related keys are K^i and K^j, $K^i = K \oplus \Delta K_i$, $K^j = K \oplus \Delta K_j$, $K^i \oplus K^j = \Delta K_{ij} = (\Delta k_0^{ij}, 0, 0, 0, 0)$, the corresponding plaintexts satisfying $\Delta P = P^i \oplus P^j = (\Delta k_0^{ij}, 0, *, *)$. For each one of $N_\Delta = 2^9$ related-key differences, we do Step 1 to Step 5 as follows.

-Step 1. For each structure, the values of (P_0^i, P_1) and (P_0^j, P_1) are fixed. Thus for every k_0 we do as following:

(a) Calculate the value of $\Delta F_2^1 = F(F(P_0^i \oplus k_0^i) \oplus k_0^i \oplus P_1) \oplus F(F(P_0^j \oplus k_0^j) \oplus k_0^j \oplus P_1)$. Constructing (P_2^i, P_2^j) which satisfies $\Delta P_2^{ij} = \Delta F_2^1$, there are 2^{16} pairs of plaintexts of (P_2^i, P_2^j).

(b) By traversing (P_3^i, P_3^j), we can obtain $2^{16} \times 2^{32} = 2^{48}$ pairs of plaintexts.

(c) As the ciphertext difference is $\Delta C = (\Delta C_0, \Delta C_1, \Delta C_2, \Delta C_3) = (*, 0, 0, *)$, the number of remaining expected pairs is $2^{48} \times 2^{-32} = 2^{16}$. In other words, that is for every k_0 the number of remaining expected pairs for a structure is 2^{16}.

-Step 2. For each pair, the values of ΔP_3 and (P_2^i, P_2^j) are known. Thus by looking up the table, we get the value of k_1 satisfying the difference value $\Delta F_1^2 = F(P_2^i \oplus k_1) \oplus F(P_2^j \oplus k_1) = \Delta P_3$. For each pair, there is one expected solution of k_1, thus for every $k_0||k_1$ the number of remaining expected pairs for a structure is $2^{16} \times 2^{-16} = 1$.

-Step 3. For remaining pairs, the values of ΔC_3^{ij}, (C_1, C_2) and (k_0^i, k_0^j) are known. Similar to Step 2, by looking up the table, we get the value of k_1 satisfying the difference $\Delta F_{17}^1 = F(F(C_1 \oplus k_4) \oplus k_0^i \oplus C_2) \oplus F(F(C_1 \oplus k_4) \oplus k_0^j \oplus C_2) = \Delta C_3^{ij}$. Similarly, there is one expected solution of k_4 for each pair, thus for every $k_0||k_1||k_4$ the number of remaining expected pairs for a structure is $1 \times 2^{-16} = 2^{-16}$.

-Step 4. For remaining pairs, the values of ΔC_0^{ij} and (C_3^i, C_3^j) are known. Similar to Step 2, by looking up the table, we get the value of k_3 satisfying the difference $\Delta C_0^{ij} = \Delta F_{18}^1 = F(C_3^i \oplus k_3) \oplus F(C_3^j \oplus k_3)$. Similar to Step 2, by looking up the table we get k_3. Similarly, there is one expected solution of k_3 for each pair, thus for each $k_0||k_1||k_4||k_3$ the number of remaining expected pairs for a structure is $2^{-16} \times 2^{-16} = 2^{-32}$.

-Step 5. If we find a right pair for the guessing value $k_0||k_1||k_4||k_3$, we delete the corresponding subkey $k_0||k_1||k_4||k_3$ in table D.

-Step 6. For other differences of related-key Δk_0, we repeat step 1 to step 5. Ultimately, for each survived candidate in Table D, we compute the seed key by doing an exhaustive search for other 16 bits.

Complexity Analysis. Recently, Boura et al.[18] proposed a generic vision of impossible differential attacks on block ciphers. In their method, they split the cipher in three parts: $E = E_3 \circ E_2 \circ E_1$, in E_2 there is an impossible differential($\Delta_X \nrightarrow \Delta_Y$), see Fig.5. And Δ_X (resp. Δ_Y) is propagated through E_1^{-1} (resp. E_3) with probability 1 to obtain Δ_{in} (resp. Δ_{out}). Thus the differential $(\Delta_X \leftarrow \Delta_{in})$ (resp. $\Delta_Y \leftarrow \Delta_{out}$) is verified with probability $1/2^{c_{in}}$ (resp. $1/2^{c_{out}}$), where c_{in} (resp. c_{out}) is the number of bit-conditions that have to be verified to obtain Δ_X from Δ_{in} (resp. Δ_Y from Δ_{out}).

Let Δ_X (resp. Δ_Y) denote the input differences of the impossible differential, Δ_{in} (resp. Δ_{out}) denote the set of all possible input (resp. output) differences of the cipher, r_{in} (resp. r_{out}) denote the number of rounds of the differential path (Δ_X, Δ_{in}) (resp. (Δ_Y, Δ_{out})), and r_Δ denote the number of rounds of the impossible differential.

According to the method of calculating complexity in[18], we get the results of data complexity, time complexity and memory complexity as follows.

-Data Complexity. In our work, $\Delta_{in} = \Delta P$, $\Delta_{out} = \Delta C$ and $c_{in} = c_{out} = 32$. It follows that for a given key $k_0||k_1||k_4||k_3$, a pair of plaintext-ciphertext already satisfying ΔP and ΔC, the probability that it is the right pair is $2^{-c_{in}} \times$

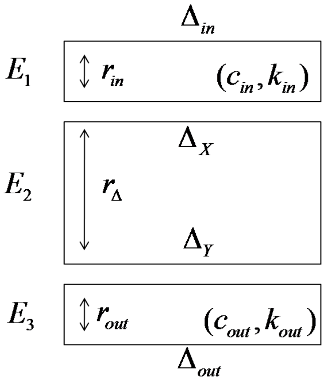

Fig. 5: Generic Vision of Impossible Differential Attack

$2^{-c_{out}} = 2^{-64}$. Therefore, for N pairs of plaintext-ciphertexts satisfying ΔP and ΔC, the probability that the given key is remaining in the table D is $P = (1 - 2^{-64})^N$.

Because of $\Delta P = (\Delta k_0, 0, *, *)$, $\Delta C = (*, 0, 0, *)$, there are $2^{32} \times 2^{32} \times 2^{-32} = 2^{32}$ pairs of plaintext-ciphertext satisfying ΔP and ΔC for one structure. Taking $N_s = 2^{26}$ such that $N = N_\Delta \times N_s \times 2^{32} = 2^9 \times 2^{26} \times 1 = 2^{67}$, there is $2^{64} \times (1 - 2^{-64})^N = 2^{64} \times (1 - 2^{-64})^{67} = 2^{52.46}$ values of $k_0||k_1||k_3||k_4$ remaining in table D.

The data complexity is $N_r \times N_s \times 2^{32} = 2^5 \times 2^{26} \times 2^{32} = 2^{63}$ chosen-plaintexts.

-Time Complexity. To analyze the time complexity, we will analyze the time complexity in each step. In the data collection phase, the time complexity is 2^{63} full-round encryptions.

In the key recovery phase, building the table F and F' the time complexity is $2^{48} \times 1/18 \times 2 + 2^{32} \times 1/18 \approx 2^{44.83}$. In Step 1. the time complexity is $2^{16} \times N_\Delta \times N_s \times 1/18 \times 2 = 2^{16} \times 2^9 \times 2^{26} \times 1/18 \times 2 \approx 2^{47.83}$. In Step 2, the time complexity is $2^{16} \times 2^{9+26+16} \times 1/18 \times 1/24 \approx 2^{58.25}$. In Step 3, the time complexity is $2^{32} \times 2^{9+26} \times 1/18 \times 1/24 \approx 2^{58.25}$. In Step 4, the time complexity is $2^{48} \times 2^{9+26-16} \times 1/18 \times 1/24 \approx 2^{58.25}$. In Step 5 for the exhaustive searching, the time complexity is $2^{52.46} \times 2^{16} \approx 2^{68.46}$.

Therefore, the total time complexity is $2^{68.46}$ full-round encryptions.

-Memory Complexity. For storing the key table D, F and F', the memory complexity is $2^{64}, 2^{48}$ and 2^{32}, respectively. Thus the memory complexity is 2^{64}.

In summary, we propose an attack on full Khudra with the data, time and memory complexities are 2^{63}, $2^{68.46}$ and 2^{64}, respectively. Using the same method, we could launch an attack on full-round Khudra without whitening keys with a data complexity of 2^{63} chosen plaintexts, a time complexity of $2^{68.46}$ full-round encryptions and a memory complexity of 2^{63}.

4 Conclusion

In this paper, we obtain 2^{16} 14-round related-key impossible differential characteristics. Using 32 related keys, we proposed a related-key impossible differential attack on the full Khudra by respectively extending 2 rounds backward and forward, with a data complexity of 2^{63} chosen plaintexts, a time complexity of $2^{68.46}$ full-round encryptions and a memory complexity of 2^{64}.

References

1. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: Cryptographic Hardware and Embedded Systems - CHES 2007. pp. 450-466. Springer (2007)
2. Guo, J., Peyrin, T., Poschmann, A., Robshaw, M.: The LED Block Cipher. In: Cryptographic Hardware and Embedded Systems - CHES 2011. pp. 326-341. Springer (2011)
3. Wu, W., Zhang, L.: LBlock: A Lightweight Block Cipher. In: Applied Cryptography and Network Security - ACNS 2011. pp. 327-344. Springer (2011)
4. Borghoff, J., Canteaut, A., Gneysu, T., Kavun, E.B., Knezevic, M., Knudsen, L.R., Leander, G., Nikov, V., Paar, C., Rechberger, C., Rombouts, P., Thomsen, S.S., n, T.Y.: PRINCE - A Low-Latency Block Cipher for Pervasive Computing Applications. In: Advances in Cryptology - ASIACRYPT 2012. pp. 208-225. Springer (2012)
5. Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK Families of Lightweight Block Ciphers. Cryptology ePrint Archive (2013), https://eprint.iacr.org/2013/404
6. Kolay, S., Mukhopadhyay, D.: Khudra: A New Lightweight Block Cipher for FPGAs. In: Security Privacy and Applied Cryptography Engineering - SPACE 2014. pp. 126-145 (2014)
7. Biham, E., Shamir, A.: Differential Cryptanalysis of DES-like Cryptosystems. Journal of Cryptology 4(1), 3-72 (1991)
8. Matsui, M.: Linear Cryptanalysis Method for DES Cipher. In: Advances in Cryptology - EUROCRYPT93, ser. Lecture Notes in Computer Science, vol. 765, pp. 386-397. Springer, Berlin (1994)
9. Biham, E.: New Types of Cryptanalytic Attacks Using Related Keys. Journal of Cryptology 7(4), 229-246 (1994)
10. Knudsen, L.: Truncated and Higher Order Differentials. In: FSE 1995, ser. Lecture Notes in Computer Science, vol. 1008, pp. 196-211. Springer, Berlin (1995)
11. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials. In: Journal of Cryptology. pp. 12-23. Springer (1999)
12. L.R. Knudsen. DEAL - A 128-bit Block Cipher. Department of Informatics, University of Bergen, Norway. Technical report, 1998.
13. Kim, J., Hong, S., Sung, J., Lee, S., Lim, J., Sung, S.: Impossible Differential Cryptanalysis for Block Cipher Structures. In: Progress in Cryptology - INDOCRYPT 2003. pp. 82-96. Springer (2003)

14. Luo, Y., Wu, Z., Lai, X., Gong, G.: A Unified Method for Finding Impossible Differentials of Block Cipher Structures. Cryptology ePrint Archive (2009), http://eprint.iacr.org/2009/627
15. Wu, S., Wang, M.: Automatic Search of Truncated Impossible Differentials for Word-Oriented Block Ciphers. In: Progress in Cryptology - INDOCRYPT 2012. pp. 283-302. Springer (2012)
16. Jakimoski, G., Desmedt, Y.: Related-Key Differential Cryptanalysis of 192-bit Key AES Variants. In: Selected Areas in Cryptography. pp. 208-221 (2004)
17. Bogdanov, A.A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 450-466. Springer, Heidelberg (2007)
18. Boura, C., Naya-Plasencia, M., Suder, V.: Scrutinizing and Improving Impossible Differential Attacks: Applications to CLEFIA, Camellia, LBlock and Simon. In: Advances in Cryptology - ASIACRYPT 2014. pp. 179-199. Springer (2014)