INTEGRATION, the VLSI journal 94 (2024) 102090

Contents lists available at ScienceDirect

Integration, the VLSI Journal

journal homepage: www.elsevier.com/locate/vlsi

HDLBC: A lightweight block cipher with high diffusion☆

Yongchao Li, Jingya Feng, Qi Zhao, Yongzhuang Wei *

Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin, 541004, China

ARTICLE INFO

Keywords:
Lightweight block cipher
Generalized Feistel structure
AND-Rotation-XOR
Internet of Things
SIMON

ABSTRACT

Both the diffusion property and the area consumption are two important evaluation criteria in the design and implementation of symmetric encryption algorithms. Many AND-Rotation-XOR (AND-RX) block ciphers are usually designed by reducing the diffusion property to minimize the area consumption. On the other hand, these AND-RX block ciphers use multiple round function operations to achieve the enough diffusion property, which always induce more area consumption in their hardware implementation. How to trade off the diffusion property and the area consumption appears to be an interesting task in the design of block cipher. In this paper, HDLBC as a new family of lightweight block cipher (with 64-bit plaintext and 64-bit/128-bit key) for the Internet of Things (IoT) is proposed. More specifically, the HDLBC is designed by using only two F-functions (RA_1 and RA_2), where the non-linear layer of the F-functions is constructed by the NAND operation that consumes the least area among the non-linear logic operations. To the best of our knowledge, HDLBC cipher requires the minimum number of F-functions to provide the diffusion property, where the F-functions require fewer implementation resources than the F-functions of existing similar encryption algorithms. It illustrates the hardware implementation of HDLBC cipher on SMIC 0.18 μm requires only 1248 Gate Equivalents (GEs), its throughput rate is 256 Kbps at 100 KHz. Compared with other encryption algorithms, the implementation performance of HDLBC cipher achieves well-balanced in both the area consumption and diffusion property. Moreover, the security analysis shows that HDLBC cipher has enough security margin against various known attacks, such as differential cryptanalysis, linear cryptanalysis, impossible differential cryptanalysis, zero correlation cryptanalysis, etc.

1. Introduction

IoT has more and more application scenarios with the advent of the 5G (fifth generation) seamless communication network technology. For instance, smart logistics, smart healthcare, smart home, etc. Notice that many IoT devices are usual resource-constrained. Therefore, lightweight encryption schemes are urgently required to provide the security for IoT data.

During the past decade, many lightweight block ciphers were proposed, for instance, SCENERY [1], QTL [2], RECTANGLE [3], SIMON and SPECK [4], SIMECK [5], GIFT [6], CRAFT [7], PRESENT [8], etc. Actually, there are many indicators used to guide the design of lightweight block ciphers, such as low power consumption, low latency, low area, high throughput, high diffusion, and so on. For example, PRESENT cipher was proposed at CHES 2007, which requires only 2018 GEs for hardware implementation (encryption and decryption) on SMIC 0.18 μm. Currently, it is listed as an encryption algorithm of ISO/IEC standard. Moreover, GIFT cipher was presented at CHES 2017, which selects smart S-box to reduce the area consumption, and it also further optimizes the PLayer to improve the diffusion property. The Midori [9] cipher was designed to achieve superior energy and efficiency in hardware, PRINCE [10] cipher was designed to meet low latency properties, and RECTANGLE cipher uses bit-slice technology in order to be suitable for multi-platform implementations. On the other hand, the block cipher SIMON was published by the National Security Agency (NSA), which uses AND-RX/ARX operations instead of traditional S-boxes [4]. More precisely, the hardware implementations of SIMON cipher for encryption and decryption require only 1751 GEs on SMIC 0.13 μm, which is even smaller than the area of the implementation for PRESENT cipher. SIMON-like ciphers are more suitable for resource-constrained devices because of the very simple logic operations (Addition, AND, Rotation, and XOR), (these operations allow for excellent hardware performance).

Remarkably, SIMON-like ciphers have relative slow diffusion property due to the use of Feistel structure. In this case, SIMON-like ciphers

☆ The work was supported in part by the National Natural Science Foundation of China (Grant No. 61872103), in part by the Guangxi Natural Science Foundation, China (2019GXNSFGA245004), and in part by the Innovation Project of Guangxi Graduate Education, China (YCSW2023304, YCBZ2023132).

* Corresponding author.
E-mail addresses: supermelyc@vip.qq.com (Y. Li), fengjyk@126.com (J. Feng), wakeupzq@163.com (Q. Zhao), walker_wyz@guet.edu.cn (Y. Wei).

https://doi.org/10.1016/j.vlsi.2023.102090
Received 6 December 2022; Received in revised form 6 July 2023; Accepted 14 September 2023
Available online 24 September 2023

Table 1
Parameters for HDLBC-64 and HDLBC-128.

Cipher	Block size $4n$	Branch size n	Key size m	Round number Nr
HDLBC-64	64	16	64	25
HDLBC-128	64	16	128	32

usually need more round operations to achieve sufficient diffusion property, which always induces more area consumption in hardware implementation. For instance, SHADOW cipher [11] was proposed by GUO et al. in 2021, which makes use of a combination of generalized Feistel structure (GFS) and AND-RX operations. Although four F-functions of SHADOW are cleverly selected to solve the problem of relative slow diffusion property, it still needs relative high area consumption in hardware implementation. Thus, how to trade off the diffusion property and the area consumption appears to be an interesting task in the design of block cipher.

In this work, HDLBC as a new family of lightweight block cipher (with 64-bit plaintext and 64-bit/128-bit user secret key) for the Internet of Things (IoT) is proposed. The main contributions are given as follows:

- In order to make a better trade off between diffusion properties and area consumption, HDLBC uses a new GFS with NAND-RX operations. More precisely, the GFS requires only two F-functions based on NAND-RX operations to achieve the goal of diffusion property.
- In order to address the problem of slow diffusion after the initial plaintext and key injection of full 0/F in the SIMON-like ciphers, a key schedule is constructed by based on the NAND-RX structure. Our experimental result shows that this key schedule achieves pretty good diffusion without significantly increasing the hardware area.
- In the hardware implementation, HDLBC cipher is compared with the current classical SIMON-like ciphers, e.g., SIMON, SPECK, and SIMECK, under different CMOS technologies. The experimental results show that it requires only 1436 GEs on SMIC 0.13 μm, which is smaller than SIMECK. Moreover, its throughput rate is 256 Kbps at a frequency of 100 KHz, compared to SIMON, which is a performance improvement by 76.01%. Furthermore, the security of HDLBC cipher is checked by using various attacks. In particular, we used Mixed Integer Linear Program (MILP) to find the optimal characteristic probability for the HDLBC cipher. The results show that the HDLBC cipher has enough ability against impossible differential attacks, differential attacks, integral attacks, and so on.

The rest of this paper is organized as follows. The specification of the HDLBC cipher is given in Section 2. The design rationale for the linear and non-linear layers of the HDLBC cipher is described in Section 3. In Section 4, the security evaluation for the HDLBC cipher is provided. The hardware performance evaluation of this cipher is investigated in Section 5. Finally, we conclude the paper in Section 6.

2. Specification of HDLBC

HDLBC is a family of two NAND-RX block ciphers with GFS, i.e. HDLBC-64 and HDLBC-128. Both of them accept 64-bit plaintext and have different key sizes m, where m stands for the size of the user secret key (m = 64 for HDLBC-64 and m = 128 for HDLBC-128). The basic parameters of HDLBC-64 and HDLBC-128 are listed in Table 1.

2.1. Notations

In the specification of HDLBC, we use the following notations in Table 2.

Table 2
Notations of HDLBC.

Notations	Descriptions
P	64-bit plaintext
C	64-bit ciphertext
K	64/128-bit master key or round key
L^i, R^i	16-bit L and R in the i round
SK^i	16-bit round subkey SK in the i round
F	Function F
RA	RoundfunctionA
Nr	Round number
|	Concatenation of two binary strings
⊕	Bitwise exclusive-OR operation
<<<	Left circular shift operation

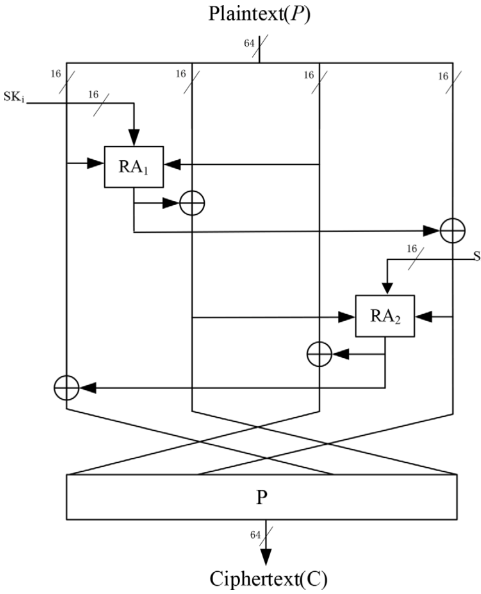

Fig. 1. The round function of HDLBC.

Table 3
The encryption routine of HDLBC-64.

```
Algorithm 1: HDLBC-64 Encryption Routine
Input: $P_{(64)}$, $K_{(64)}$
Output: $C_{(64)}$
1: $P_{(64)} \rightarrow P^1_{0(16)} \mid P^1_{1(16)} \mid P^1_{2(16)} \mid P^1_{3(16)}$
2: $Generatekey(K_{(64)},SK)$
3: for i=0 to Nr-1 do the following
       $RA_1\ (P^i_{0(16)},\ P^i_{2(16)}\ ,SK^i) \oplus\ P^i_{1(16)}\ \rightarrow\ P^{i+1}_{3(16)}$,
       $RA_1\ (P^i_{0(16)},\ P^i_{2(16)}\ ,SK^i) \oplus\ P^i_{3(16)}\ \rightarrow\ P^{i+1}_{1(16)}$,
       $RA_2\ (P^{i+1}_{1(16)},\ P^{i+1}_{3(16)}\ ,SK^i) \oplus\ P^i_{2(16)}\ \rightarrow\ P^{i+1}_{0(16)}$,
       $RA_2\ (P^{i+1}_{1(16)},\ P^{i+1}_{3(16)}\ ,SK^i) \oplus\ P^i_{0(16)} \rightarrow\ P^{i+1}_{2(16)}$,
       $PLayer\ (P^{i+1}_{0(16)},\ P^{i+1}_{1(16)},\ P^{i+1}_{2(16)},\ P^{i+1}_{3(16)})$
4: end for
5: $P^{Nr}_{0(16)}|P^{Nr}_{1(16)}|P^{Nr}_{2(16)}|P^{Nr}_{3(16)}\ \rightarrow\ P_{(64)}$
6: $C_{(64)} \leftarrow P_{(64)}$
```

2.2. Encryption process

The encryption of HDLBC has Nr iterative round operations by using the GFS. In particular, the round function of HDLBC-64 is described in Fig. 1, where the input 64-bit plaintext block is denoted by $P \in \{0,1\}^{64}$, i.e., $P = P_{0(16)} \parallel P_{1(16)} \parallel P_{2(16)} \parallel P_{3(16)}$. Similarly, the 64-bit key is denoted by $K \in \{0,1\}^{64}$, the output 64-bit ciphertext is denoted by $C \in \{0,1\}^{64}$. The encryption procedure is given in Table 3.

The key components used in HDLBC are described below.

Round Function: The round function consists of the following components, F-functions (RA_1 and RA_2), branch XOR, and PLayer. In the

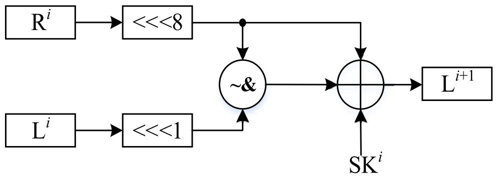

Fig. 2. The RA of HDLBC.

Table 4
PLayer form HDLBC.

i	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*
0	57	8	59	16	61	24	63	32	56	40	58	48	60	56	62
1	49	9	51	17	53	25	55	33	48	41	50	49	52	57	54
2	41	10	43	18	45	26	47	34	40	42	42	50	44	58	46
3	33	11	35	19	37	27	39	35	32	43	34	51	36	59	38
4	25	12	27	20	29	28	31	36	24	44	26	52	28	60	30
5	17	13	19	21	21	29	23	37	16	45	18	53	20	61	22
6	9	14	11	22	13	30	15	38	8	46	10	54	12	62	14
7	1	15	3	23	5	31	7	39	0	47	2	55	4	63	6

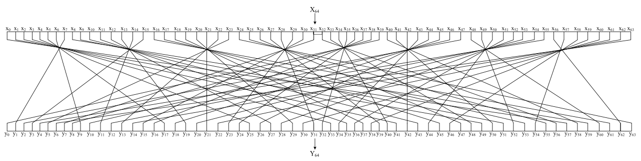

Fig. 3. The process of PLayer.

first place, the inputs to the function RA_1 consist of P_0 and P_2, and the output of RA_1 is XORed with P_1 and P_3, respectively (denoted by P'_1 and P'_3). Similarly, the inputs of the function RA_2 consist of P'_1 and P'_3. Moreover, the output of RA_2 is XORed with P_0 and P_2, respectively. Finally, the 64-bit result is performed as a PLayer permutation. F-functions and the PLayer permutation are detailed as follows:

The F-functions (RA_1 and RA_2): The F-functions of HDLBC is shown in Fig. 2. In each round, L^i and R^i are first cyclically shifted left by 1 and 8 bits, respectively. The results executed by NAND operation. After completing the NAND operation, their result is XORed with the value of R^i cyclically shifted left by 8 bits, then XORed with the round key SK^i. The result will be saved to L^{i+1}. The RA functions of HDLBC is described in equations (1, 2).

$$T = R^i \lll 8 \tag{1}$$

$$L^{i+1} \leftarrow \left(\sim \left(T \& \left(L^i \lll 1\right)\right) \oplus T \oplus SK^i\right) \tag{2}$$

PLayer: The bit permutation in HDLBC-64 is given in Table 4. As shown in Fig. 3, the *i*th bit of the state is moved to position P(*i*)th bit. The PLayer of HDLBC is described in Eq. (3).

$$i = p(i), 0 \leq i \leq 63 \tag{3}$$

2.3. Decryption process

The decryption process for HDLBC is the same as the encryption process. However, the order of the subkeys in the decryption algorithm is the reverse of the encryption process. For example, the subkey in the first round of decryption is the subkey in the last round of the encryption process, and the PLayer is different from the encryption process. The FP-box in the decryption process is shown in Table 5.

Table 5
FP-box for HDLBC.

i	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*
0	39	8	38	16	37	24	36	32	35	40	34	48	33	56	32
1	7	9	6	17	5	25	4	33	3	41	2	49	1	57	0
2	47	10	46	18	45	26	44	34	43	42	42	50	41	58	40
3	15	11	14	19	13	27	12	35	11	43	10	51	9	59	8
4	55	12	54	20	53	28	52	36	51	44	50	52	49	60	48
5	23	13	22	21	21	29	20	37	19	45	18	53	17	61	16
6	63	14	62	22	61	30	60	38	59	46	58	54	57	62	56
7	31	15	30	23	29	31	28	39	27	47	26	55	25	63	24

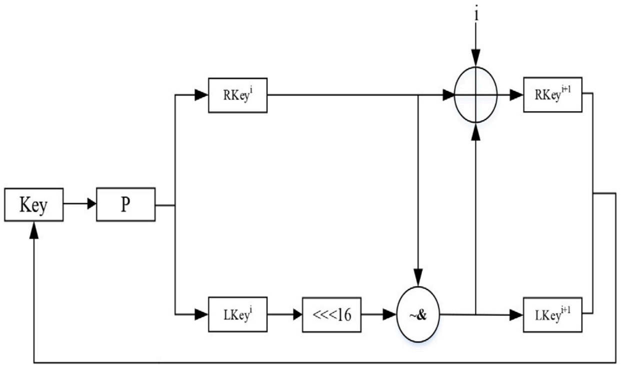

Fig. 4. Key schedule of HDLBC-64.

2.4. Key schedule

2.4.1. Key schedule of HDLBC-64

The key schedule of HDLBC-64 is illustrated in Fig. 4. First, the 64-bit initial key performed PLayer (same as Table 4). The result of the operation is then divided into 32-bit $LKey^i$ and $RKey^i$. $LKey^i$ first left cyclic shifts 16-bit to the left and does a NANDed operation with $RKey^i$. The result of the operation is $LKey^{i+1}$ for the next round. Then $LKey^{i+1}$, $RKey^i$ and i (round number) perform XOR operation to get $RKey^{i+1}$ in the next round. The first 16-bit of the 32-bit $RKey^{i+1}$ are used in RA_1, while the next 16-bit are used in RA_2.

2.4.2. Key schedule of HDLBC-128

The key schedule of HDLBC-128 is similar to Fig. 4. First, the 128-bit initial key performed PLayer (the PLayer used by HDLBC-128 is in Table 6). The result of the operation is then divided into 64-bit $LKey^i$ and $RKey^i$. $LKey^i$ first left cyclic shifts 32-bit to the left and does a NANDed operation with $RKey^i$. The result of the operation is $LKey^{i+1}$ for the next round. Then $LKey^{i+1}$, $RKey^i$ and i (round number) perform XOR operation to get $RKey^{i+1}$ in the next round. The first 16-bit of its lowest significant bit of the 32-bit are used in RA_1, while the next 16-bit are used in RA_2.

3. Design rationale

We would like to share the design motivation of the HDLBC before introducing the design rationale. Moreover, we will compare it with the similar designs to highlight the advantages of our design.

Motivations. In the first place, the SIMON family has quite good advantages in software and hardware performance. Later, the lightweight block cipher SIMECK was proposed by Yang et al. [5], which achieves a good trade-off between security and efficiency. Notice that the round function of SIMECK shares almost the same as SIMON, except for the different parameter selection of rotation operations. Recently, SHADOW was proposed by GUO et al. [11], which is a combination of GFS and AND-RX operation. Although four F-functions of SHADOW are cleverly selected to solve the problem of relative slow diffusion

Table 6
PLayer for HDLBC-128.

i	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*	*i*	*P(i)*
0	18	16	2	32	50	48	34	64	82	80	66	96	114	112	98
1	79	17	63	33	111	49	95	65	15	81	127	97	47	113	31
2	8	18	120	34	40	50	24	66	72	82	56	98	104	114	88
3	13	19	125	35	45	51	29	67	77	83	61	99	109	115	93
4	22	20	6	36	54	52	38	68	86	84	70	100	118	116	102
5	83	21	67	37	115	53	99	69	19	85	3	101	51	117	35
6	12	22	124	38	44	54	28	70	76	86	60	102	108	118	92
7	17	23	1	39	49	55	33	71	81	87	65	103	113	119	97
8	26	24	10	40	58	56	42	72	90	88	74	104	122	120	106
9	87	25	71	41	119	57	103	73	23	89	7	105	55	121	39
10	16	26	0	42	48	58	32	74	80	90	64	106	112	122	96
11	21	27	5	43	53	59	37	75	85	91	69	107	117	123	101
12	30	28	14	44	62	60	46	76	94	92	78	108	126	124	110
13	91	29	75	45	123	61	107	77	27	93	11	109	59	125	43
14	20	30	4	46	52	62	36	78	84	94	68	110	116	126	100
15	25	31	9	47	57	63	41	79	89	95	73	111	121	127	105

Table 7
Comparison of hardware resources and full diffusion rounds for the F-function of the other ciphers based on the SMIC 0.18 μm.

Cipher	F-function hardware footprint(GEs)	Full diffusion rounds	Ref.
SIMECK-64	298.88	11	[5]
SIMON-64	298.88	9	[4]
HDLBC-64	**181.44**	**3**	New

property, they still need relatively high area consumption in hardware implementation. Actually, the Feistel structure is compatible with irreversible round functions, but it perhaps induces slow diffusion property. In order to maintain a fast diffusion speed via the Feistel structure, we consider to use two F-functions (RA_1 and RA_2) in the design. The core idea is to look for an appropriate non-linear operation that consumes fewer gates than the equivalent of an AND operation. However, this non-linear operation can further improve the diffusivity.

3.1. Design of round function

As mentioned in Section 2, the round functions of HDLBC are composed of three parts: F-functions, branch XOR operations, and PLayer operations. The initial plaintext is divided into four branches to allow for confusion and diffusion in parallel. At the same time, the input of branch XOR operation comes from the values of the F-functions, it naturally brings up the issue of correlation. Therefore, how to skillfully build the round function structure of the HDLBC in these simple operations directly determines the accuracy of our later analysis of its security. On the other hand, from the perspective of the diffusivity of the SIMON, the diffusing speed is relatively slow due to fewer XOR operations. We can appropriately increase the number of XOR operations in the HDLBC round function. In fact, HDLBC has better hardware implementation advantage since it spends a relatively lower F-function hardware footprint compared to the implementations consumption of SIMON and SIMECK (see Table 7). Moreover, the number of full diffusion rounds of HDLBC-64 is only 3. However, the full diffusion rounds of SIMON-64 (or SIMECK-64) is at least 9. Therefore, we attribute the diffusion speed advantage to the well-designed round function.

3.2. Key schedule

We aim to design a hardware-efficient and fast diffusion cryptographic algorithm. The following three basic requirements should be met by the key schedule:

(1) The hardware area should be consumed as small as possible;

(2) Each subkey should be affected by master key as much as possible;

(3) Latency in the key schedule should be as low as possible.

Table 8
Results of dependency and avalanche effect on HDLBC.

Round	6	8	25
ds_a	0.999515	0.999488	0.999721
d_a	0.999679	0.999937	0.999571
d_c	1.000000	1.000000	1.000000
w_a	31.956219	32.000625	32.007312
pr	0.493828∼0.503406	0.496234∼0.505516	0.495844∼0.503734
pr_a	0.499316	0.500010	0.500114

pr_a: Indicates the average probability of the output changing.

In order to strike a balance between the three aspects above, the primary idea is to use a linear key schedule. Within the linear key schedule, its update function can make use of a round-based Feistel structure. This design sacrifices some latency to save area consumption, but guarantees good diffusion of the entire key. We generalize this design principle to fit HDLBC and supply priority to area consumption and diffusion property. From the experimental results, the PLayer and rotation parameters are selected (mainly considering differential and linear analysis). It is eventually found that these components improve the diffusivity. Meanwhile, HDLBC introduces the round number as the round constant to avoid possible circular shift symmetries and slide attacks. Compared with key schedule of SIMON, HDLBC adopted less resource consumption to achieve better diffusivity. With all these efforts, we attain the final key schedule strategies.

4. Security evaluation

4.1. Dependency and avalanche effect

The avalanche effect is an important criterion for cryptographic algorithms. Kam and Davida [12] and H. Feistel [13] first proposed the concepts of dependency and the avalanche effect. Thereafter, Webster and Tavares [14] introduced the strict avalanche criterion. For HDLBC, we tested its strict avalanche effect (d_{sa}), avalanche effect (d_a), and completeness (d_c) by studying [15] and Eq. (4). Assume a random input sample of $\#X$. The following equation is given:

$$\begin{aligned} d_c &= 1 - \#\left\{(i,j) \mid a_{ij} = 0, i = 1,\dots,n; j = 1,\dots,m\right\} / (nm), \\ d_a &= 1 - \sum_{i=1}^{n} \mid \sum_{j=1}^{m} 2jb_{ij}/\#X - m \mid / (nm), \\ d_{\mathrm{sa}} &= 1 - \sum_{i=1}^{n} \sum_{j=1}^{m} \mid 2a_{ij}/\#X - 1 \mid / (nm). \end{aligned} \tag{4}$$

In order to determine whether a cipher has good dependency, it should satisfy $d_c = 1$, $d_a \approx 1$, and $d_{sa} \approx 1$.

Moreover, the avalanche effect means even 1-bit change of the plaintext would cause many bits change in ciphertext. If the change affects at least half of the bits, which means 50% of the ciphertext, then it will be treated as a good avalanche effect. The avalanche effect can be observed by following Eq. (5).

$$w_{\mathrm{a}} = \frac{\text{Number of changed bit in ciphertext}}{\text{Number of bits in ciphertext}} \times 100\% \tag{5}$$

We selected 1000 specific plaintext blocks and randomly changed them by 1-bit to form test cases. According to the Table 8 and Fig. 5, we know that HDLBC reaches full dependency after 6 rounds; thus, we believed full rounds of HDLBC should meet the criteria for an avalanche effect.

4.2. Differential and linear cryptanalysis

Differential cryptanalysis (DC) [16] and linear cryptanalysis (LC) [17] are the most classical analysis methods for block ciphers. In general, the ability of an encryption algorithm against differential

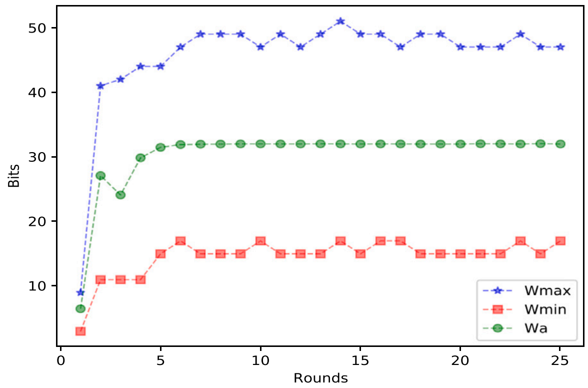

Fig. 5. Analysis of avalanche effect of HDLBC.

and linear cryptanalysis can be reflected by the best differential and linear characteristic probability corresponding to different rounds, as shown in equations (6, 7). We adopted the MILP automated search technique [18] to obtain the best differential and linear characteristic probability for HDLBC.

$$\mathrm{DP}(\alpha,\beta)=\underset{X,K}{Prob}\{F(X,K)\oplus F(X\oplus\alpha,K)=\beta\}. \tag{6}$$

$$\mathrm{LP}(\alpha,\beta)=\left(2\cdot\underset{X,K}{Prob}\{\alpha\cdot X=\beta\cdot F(X,K)\}-1\right)^2. \tag{7}$$

The MILP model used in the analysis included two key components: the objective function and the constraint conditions. The objective function directs the model's solution, while the constraint conditions characterize the differential (or linear) class propagation properties of the model. The specific MILP model of HDLBC is described as follows:

Objective function: In order to evaluate the differential (or linear) class security of the algorithm, the objective function can be set to maximize the probability of differential (or linear) characteristics. For the HDLBC cipher, each round consists of 32 NAND operations, generating a differential (or linear) probability of p. When solving for r-round differential (or linear) characteristics, the corresponding objective function can be defined as follows:

$$\text{minimize}\sum_{i=0}^{32r-1}p_i$$

Constraint conditions: The HDLBC cipher primarily involves operations such as XOR, NAND, PLayer. The propagation properties of these operations can be characterized as follows:

(1) XOR operation: The XOR operation can be represented as $c=a\oplus b$, where (a,b,c) can take on eight possible combinations, including four potential propagation patterns: (0, 0, 0), (0, 1, 1), (1, 0, 1), and (1, 1, 0). These patterns can be characterized using the following inequalities, where d is a dummy variable that can take on the values $\{0,1\}$.

$$\begin{cases} a+b+c\geq 2d\\ a+b+c\leq 2\\ d\geq a, d\geq b, d\geq c\end{cases}$$

(2) NAND operation: The NAND operation can be represented as $c=\sim(a\&b)$. Among all eight possible input combinations, only (0, 0, 1) is an impossible propagation pattern. The possible propagation patterns and their probabilities are shown in Table 9.

According to the Table 9, the possible propagation probabilities for the NAND operation are only 1 and 2^{-1}. Introducing an additional binary variable p to represent the probability of possible propagation patterns, $p=1$ indicates a probability of 2^{-1}, while $p=0$ indicates a probability of 1 (2^0). Furthermore, all possible propagation patterns (a,b,c) can be extended to the form (a,b,c,p), including: (0, 0, 0, 0), (0, 1, 0, 1), (0, 1, 1, 1), (1, 0, 0, 1), (1, 0, 1, 1), (1, 1, 0, 1), and (1, 1, 1, 1). These patterns can be characterized using the following inequalities:

$$\begin{cases} -a+p\geq 0\\ -b+p\geq 0\\ -c+p\geq 0\\ a+b-p\geq 0\end{cases}$$

Table 9
Possible propagation patterns and their probabilities for the NAND operation.

Δa	Δb	Δc	Probability
0	0	0	1
0	1	0	2^{-1}
0	1	1	2^{-1}
1	0	0	2^{-1}
1	0	1	2^{-1}
1	1	0	2^{-1}
1	1	1	2^{-1}

Table 10
The best differential characteristic probability of HDLBC-64.

Rounds	9	10	11	12	13	14	15	16
DC	2^{-19}	2^{-24}	2^{-31}	2^{-34}	2^{-39}	2^{-46}	2^{-53}	2^{-65}

Table 11
The best linear characteristic probability of HDLBC-64.

Rounds	9	10	11	12	13	14	15	16
LC	2^{-18}	2^{-24}	2^{-31}	2^{-34}	2^{-39}	2^{-44}	2^{-53}	2^{-66}

(3) PLayer operation: The HDLBC cipher includes linear transformations such as the PLayer operation, which shuffles the differential values of specified bit positions. Suppose the length of the plaintext is n, and the differential values before and after the linear transformation are $(x_0,x_1,\dots,x_{n-1})$ and $(y_0,y_1,\dots,y_{n-1})$, respectively. If the differential value of the ith bit (0 ≤ i ≤ n-1) is shifted to the p_ith bit, n equations can be established as follows:

$$\begin{cases} y_0=x_{p_0}\\ \vdots\\ y_{n-1}=x_{p_{n-1}}\end{cases}$$

(4) Additional constraint conditions: To obtain appropriate solutions, it is necessary not only to characterize the cryptographic properties of each component of the cipher but also to control the input and output differentials. Suppose the block length of the HDLBC cipher is n, and the input and output differentials are $(x_0,x_1,\dots,x_{n-1})$ and $(y_0,y_1,\dots,y_{n-1})$, respectively. To ensure that the input differential is not all zeros during the search for differential characteristics, the following constraint must be set:

$$\sum_{i=0}^{n-1}x_i\geq 1$$

By formulating the MILP model of HDLBC into mathematical inequalities, we have successfully entered the automated search process and obtained security results as shown in Tables 10 and 11. According to the results, we can obtain that the best differential and linear characteristic probability of HDLBC-64 after 16 rounds satisfies $2^{-65} < 2^{-64}$. Thus, we consider that HDLBC-64 for 25 rounds is sufficient to resist differential and linear cryptanalysis under the single-key scenario.

Remark. The bound of 2^{-64} for HDLBC-64 is reached after 16 rounds. As for SIMON, the single trail of SIMON-64 attains the bound 2^{-64} with 19 rounds [19].

4.3. Impossible differential cryptanalysis and zero-correlation cryptanalysis

Impossible differential cryptanalysis (IDC) [20] makes use of a pair of differences (Δin, Δout), in encryption algorithms where Δin does not propagate to Δout. A similar idea was developed in linear cryptanalysis as zero-correlation cryptanalysis (ZC) [21]. Resistance to both attacks is generally explored by the longest distinguisher we can find.

In this subsection, we make use of the same technical approach as in Section 4.2. Although IDC and ZC belong to differential and linear classes of analysis, respectively, the objective functions and constraints in MILP are also different in some places. More specifically, IDC and ZC do not require the specification of an objective function and require the specification of the model's input and output differential values in the constraints. Suppose the input and output differential values are $(a_0, a_1, \dots, a_{n-1})$ and $(b_0, b_1, \dots, b_{n-1})$, respectively. In this case, the following $2n$ constraint conditions need to be added:

$$\begin{cases} x_0 = a_0 \\ \vdots \\ x_{n-1} = a_{n-1} \\ y_0 = b_0 \\ \vdots \\ y_{n-1} = b_{n-1} \end{cases}$$

By transforming the operations in the HDLBC round function into the constraints in MILP. Furthermore, we solved the transformed MILP model using Python coding and obtained the security results. The longest impossible differential and zero-correlation distinguishers determined with the model for HDLBC-64 both achieve 8 rounds, and we selected only two of them to list as follows:

$$\begin{pmatrix} 0000 & 0000 & 0000 & 0000 \\ 0000 & 0000 & 0000 & 0000 \\ 0000 & 0001 & 0000 & 0000 \\ 0000 & 0001 & 0000 & 0000 \end{pmatrix} \overset{8r-IDC}{\rightarrow} \begin{pmatrix} 0000 & 0000 & 0000 & 0000 \\ 0000 & 0000 & 0000 & 0000 \\ 0000 & 0000 & 0000 & 0000 \\ 0000 & 0001 & 0000 & 0000 \end{pmatrix}$$

$$\begin{pmatrix} 0000 & 0000 & 0000 & 0000 \\ 0000 & 0000 & 0000 & 0000 \\ 0000 & 0000 & 0000 & 0000 \\ 0000 & 0001 & 0000 & 0000 \end{pmatrix} \overset{8r-ZC}{\rightarrow} \begin{pmatrix} 0000 & 0000 & 0000 & 0000 \\ 0000 & 0000 & 0000 & 0000 \\ 0000 & 0001 & 0000 & 0000 \\ 0000 & 0001 & 0000 & 0000 \end{pmatrix}$$

For the results, the 25 rounds of HDLBC-64 can resist impossible differential and zero-correlation cryptanalysis.

4.4. Integral cryptanalysis

It is a popular method to evaluate the resistance of an algorithm to integral analysis [22] by its division property [23]. Therefore, we used the bit-based division property [24] to evaluate the degree of algebraicity. In the following, we introduced the definitions and propagation rules related to integral analysis, as well as the use of $\mathbb{F}_2^n$ to denote an n-dimensional vector space over $\mathbb{F}_2$.

Definition 1 (*Bit Product Function [25]*)**.** Given any $u \in \mathbb{F}_2^n$, $\pi_{\boldsymbol{u}}(\cdot)$ is a function from $\mathbb{F}_2^n$ to $\mathbb{F}_2$. For any $\boldsymbol{x} \in \mathbb{F}_2^n$

$$\pi_{\boldsymbol{u}}(\boldsymbol{x}) = \prod_{i=0}^{n} x_i^{u_i}$$

where u_i and x_i are the ith components of $\boldsymbol{u}$ and $\boldsymbol{x}$, respectively.

Definition 2 (*Division Property [24]*)**.** Let $\mathbb{X}$ be a multiset whose elements take a value of $(\mathbb{F}_2^n)^m$, and $\boldsymbol{k}$ be an m-dimensional vector whose coordinates take values between 0 and n. When the multiset $\mathbb{X}$ has the division property $\mathcal{D}^{n,m}_{k^{(0)},k^{(1)},\dots,k^{(q-1)}}$, it fulfills the following conditions: The parity of $\pi_u(\boldsymbol{x})$ over all $\boldsymbol{x} \in \mathbb{X}$ is always even when

$$\boldsymbol{u} \in \left\{ \left(u_0, u_1, \dots, u_{m-1}\right) \in \left(\mathbb{F}_2^n\right)^m \mid W(\boldsymbol{u}) \not\succeq k^{(\mathbf{0})}, \dots, W(\boldsymbol{u}) \not\succeq k^{(q-1)} \right\}$$

Definition 3 (*Division Trail [25]*)**.** Let f_r denote the round function of an iterated block cipher, and set the input multiset with division property $\mathcal{D}^{n,m}_{k^{(0)},k^{(1)},\dots,k^{(q-1)}}$ be the initial division property of the input, the division property after i rounds of f_r propagation is noted as $\mathcal{D}^{n,m}_{\mathbb{K}_i}$, then the following propagation chain can be obtained:

$$\{\boldsymbol{k}\} \overset{def}{=} \mathbb{K}_0 \xrightarrow{f_r} \mathbb{K}_1 \xrightarrow{f_r} \mathbb{K}_2 \xrightarrow{f_r} \cdots \xrightarrow{f_r} \mathbb{K}_r$$

Therefore, for all $(k_0, k_1, \dots, k_r) \in \mathbb{K}_0 \times \mathbb{K}_1 \times \cdots \times \mathbb{K}_r$, and $i \in \{1, 2, \dots, r\}$ are able to propagate the rule to k_{i-1} through the division property, the claim is that $(k_0, k_1, \dots, k_r)$ is a division trail of r rounds.

Similarly to Section 4.2, when applying an integral analysis to HDLBC using MILP, we need to set up the objective function and constraints.

Objective function: Suppose that the multiset $\mathbb{X}$ has the division property $\mathcal{D}^{n,m}_{k^{(0)},k^{(1)},\dots,k^{(q-1)}}$ and if $\exists k \in \mathbb{K}$ such that $u \succeq k$, then $\mathbb{X}$ no longer has the integral property. At this point, the full unit vector contained by $\mathbb{K}$. The objective function can then be set as follows:

$$Obj : Min \left\{ a_0^r + a_1^r + \cdots + a_{n-1}^r \right\}$$

where $\left\{ a_0^r + a_1^r + \cdots + a_{n-1}^r \right\}$ is a r-round division trail.

Constraint conditions: The HDLBC cipher primarily involves operations such as XOR, NAND, PLayer. The propagation properties of these operations can be characterized as follows:

(1) Copy operation: Let $(a) \overset{Copy}{\rightarrow} (b_0,\ b_1)$ be a division trail of the copy operation. Then the propagation of the division property in the copy operation is modeled as follows.

$$\begin{cases} a - b_0 - b_1 = 0 \\ a, b_0, b_1 \in \mathbb{F}_2 \end{cases}$$

(2) XOR operation: Let $(a_0, a_1) \overset{XOR}{\rightarrow} (b)$ be a division trail of the XOR operation. Then the propagation of the division property in the XOR operation is modeled as follows.

$$\begin{cases} a_0 + a_1 - b = 0 \\ a_0, a_1, b \in \mathbb{F}_2 \end{cases}$$

(3) NAND operation: Let $(a_0, a_1) \overset{NAND}{\rightarrow} (b)$ be a division trail of the NAND operation. Notice that since the NAND operation is equivalent to the AND operation XOR 1, it does not change the division property of variables and constants. Therefore, we model the NAND operation as an AND operation. Then the propagation of the division property in the NAND operation is modeled as follows.

$$\begin{cases} b - a_0 \geq 0 \\ b - a_1 \geq 0 \\ b - a_0 - a_1 \geq 0 \\ a_0, a_1, b \in \mathbb{F}_2 \end{cases}$$

For the PLayer operation, because it is a linear permutation operation, the value does not change in the bit-based division property but only follows the bit position. By transforming the operations in the HDLBC round function into the objective function and constraints in MILP. Furthermore, we solved the transformed MILP model using Python after coding and obtained the security results. As a result, we found that the longest integral distinguisher for HDLBC-64 reaches 11 rounds, which is

$$\left(\mathrm{C}^1\ \mathrm{A}^{31},\ \mathrm{A}^{32}\right) \xrightarrow{11\mathrm{r}} \left(\mathrm{U}^{31}, \mathrm{U}^1\ \mathrm{B}^1 \mathrm{U}^{31}\right),$$

where 'A^i', 'B^i', 'C^i', 'U^i' respectively stand for i consecutive active, zero-sum, constant and unknown bits. Therefore full round HDLBC is secure against integral cryptanalysis.

Table 12
Area requirement of HDLBC-64.

Module(round function)	GEs	Proportion
Data Register	384	30.78%
Branch XOR	170.88	13.69%
RA_1 and RA_2	181.44	14.54%
KS: Key Register	384	30.78%
KS: NAND	16	1.28%
KS: Constants XOR	10.56	0.85%
KS: XOR	74.72	5.98%
Control and others	26	2.10%
Total	1247.6	100%

RA: Fig. 2 shows that the RA operation has a three-input XOR, and it is known from [28] that a three-input XOR at 0.18 μm takes only 4.67 GEs.

4.5. Meet-in-the-middle attack and slide attack

The Meet-in-the-Middle (MITM) attack has been applied to the security analysis of many block ciphers [26]. The resistance of HDLBC to MITM attack is evaluated by referring to the approach adopted in [9]. Specifically, the number of rounds that can be attacked using the MITM method can be calculated by considering three concepts [9]: partial matching, initial structure, and splice-and-cut. For partial-matching, the number of rounds in both forward and backward directions cannot reach to the full diffusion rounds, implying that the bound for partial-matching is (2*R_f-2), where R_f denotes the number of full diffusion rounds for the target cipher. The length of the initiate structure can also be determined in terms of R_f and the number of rounds in which the key difference affects all non-linear operations, thereby limiting the length of initial structure to be at most R_f. The number of rounds that splice-and-cut can attack is also limited and obtained simply by subtracting one round from the full diffusion round.

Overall, using that the R_f for HDLBC is 3 rounds, the upper bound on the number of rounds that can be attacked using MITM is (2 × 3-2) + 3 + 3 = 10. Considering the total rounds for HDLBC, we believe that all versions of HDLBC have good resistance to MITM attacks. The round constant addition can effectively prevent the slide attack [27].

5. Hardware performance

In this section, we have tested the performance of the ASIC and FPGA for HDLBC, which was implemented using Verilog-HDL. In the FPGA hardware implementation, we used the Xilinx Virtex-5VLX50T and ISE 14.6 to test the performance of the HDLBC. It consumed FF, LUTS, and Slices with 133, 186, and 53 respectively. The clock frequency is 431.267 MHz and the throughput is 1104 Mbps. In the ASIC hardware implementation, we used Synopsys Design Compiler version B-2008.09 to test its performance and synthesized a standard cell library based on SMIC 0.18 μm and SMIC 0.13 μm CMOS technology. In general, the hardware implementation has two architectures: a round-based architecture and a serialized architecture. The round-based architecture is the reuse of the round function, which is a direct embodiment of the HDLBC structure. It is a good trade off between area and speed. In the following, we will introduce the encryption mode and encryption and decryption mode under the round-based architecture in detail.

The round-based architecture (encryption): HDLBC-64 uses 64-bit data paths for encryption. The calculation of one round function can be completed within one clock cycle. The data path of encryption consists of one RA_1, one RA_2, four 16-bit branch XORs, one PLayer, and two registers for storing internal state and key. In addition, our design architecture needs 5-bit round constants XOR for updating the subkey. The ciphertext is obtained after 25 clock cycles. Fig. 6 illustrates the design diagram of HDLBC-64. The cost calculation of HDLBC-64 is shown in Table 12.

Table 13
Comparison between the hardware resources of the HDLBC and other ciphers based on the SMIC 0.18 μm.

Ciphers	Structure	Latency	Block size	Key size	Area(GEs)	Ref.
KLEIN	SPN	12	64	64	2032	[29]
MIBS	Feistel	32	64	64	1396	[30]
HDLBC	GFS	25	64	64	1248	New
PUFFIN	SPN	32	64	128	2577	[31]
CRAFT	SPN	32	64	128	2072	[7]
PRESENT	SPN	32	64	128	1886	[8]
LED	SPN	48	64	128	1872	[32]
PICCOLO[a]	GFS	31	64	128	1773	[33]
SIMON	Feistel	44	64	128	1751	[4]
SKINNY	SPN	36	64	128	1696	[34]
SHADOW	GFS	32	64	128	1688	[11]
HDLBC	GFS	32	64	128	1631	New

[a] This number includes 576 GEs for key storage that is not considered in the original work.

Table 14
Comparison between the hardware resources of the HDLBC and other ciphers based on the SMIC 0.13 μm.

Ciphers	Structure	Latency	Block size	Key size	Area(GEs)	Ref.
KLEIN	SPN	12	64	64	2760	[29]
HDLBC	GFS	25	64	64	1084	New
mCRYPTON	SPN	13	64	128	3473	[35]
LED	SPN	48	64	128	3194	[32]
PRINCE	SPN	12	64	128	2953	[10]
XTEA	Feistel	32	64	128	2521	[36]
MIDORI	SPN	16	64	128	2033	[9]
SPECK	ARX	27	64	128	2014	[4]
RECTANGLE	SPN	26	64	128	1787	[3]
GIFT	SPN	29	64	128	1587	[6]
SIMECK	Feistel	44	64	128	1484	[5]
HDLBC	GFS	32	64	128	1436	New

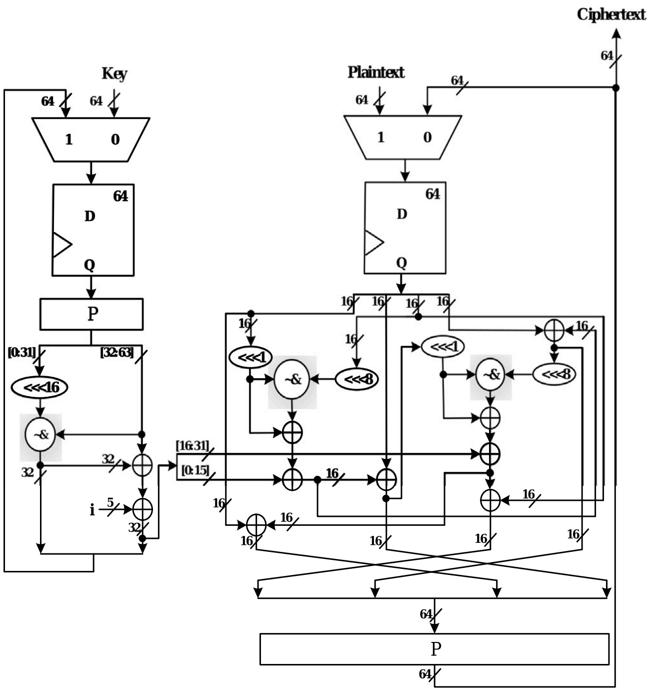

Fig. 6. The datapath of the round-based HDLBC-64.

For the HDLBC-128 version, the state data path is the same as the HDLBC-64 version. The area of HDLBC-64 is 1248 GEs and the area of HDLBC-128 is 1631 GEs. Fig. 7 shows that HDLBC has a good throughput rate compared with other ciphers at 100 KHz frequency.

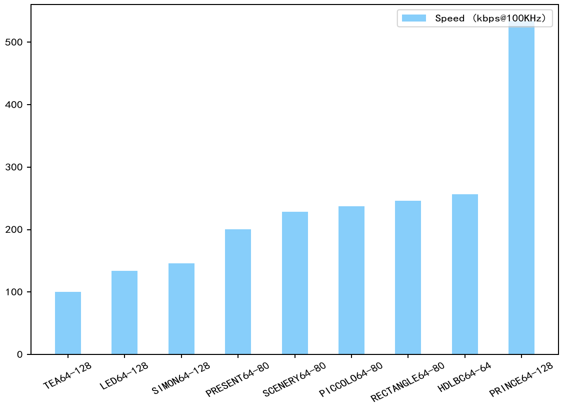

Fig. 7. Comparison between the throughput of HDLBC and other ciphers.

Table 15
The test vectors of HDLBC.

Plaintext	Key	Ciphertext
0000-0000-0000-0000	0000-0000-0000-0000	F074-0EEB-19D6-B2B9
0000-0000-0000-0000	FFFF-FFFF-FFFF-FFFF	3773-EE79-34B0-3643
FFFF-FFFF-FFFF-FFFF	0000-0000-0000-0000	E4CA-6277-1706-0E7D
FFFF-FFFF-FFFF-FFFF	FFFF-FFFF-FFFF-FFFF	92CC-91E7-4E1D-34B5
0123-4567-89AB-CDEF	0123-4567-89AB-CDEF	20B4-ACD6-393C-2242

The round-based architecture (encryption and decryption): Decryption can reuse the encryption components since HDLBC is a GFS. However, the order of the round function operations is reversed. Tables 13 and 14 show the comparison of cipher hardware implementation results under different process libraries. The result illustrates that HDLBC has more advantages in resource-constrained RFID devices. In addition, we have given the test vectors of HDLBC as shown in Table 15. The data in the table is expressed in hexadecimal.

6. Conclusion

In this paper, a new HDLBC lightweight block cipher family is proposed based on the generalized Feistel structure of NAND-RX. HDLBC overcomes the difficulty of changing only half of the plaintext after one iteration of the traditional Feistel structure. Benefiting from the use of NAND operations in the non-linear layer, HDLBC achieves a trade-off between high diffusivity and hardware resources. Compared to the current efficient lightweight block ciphers, HDLBC also achieves competitive hardware performance (in particular, comparable with SIMON). We proved that HDLBC can resist differential cryptanalysis, linear cryptanalysis, impossible differential cryptanalysis, zero-correlation cryptanalysis, integral cryptanalysis, etc. Therefore, HDLBC not only has good performance in hardware implementation but also has sufficient security levels.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] J. Feng, L. Li, SCENERY: a lightweight block cipher based on Feistel structure, Front. Comput. Sci. 16 (3) (2022) 1–10.

[2] L. Li, B. Liu, H. Wang, QTL: a new ultra-lightweight block cipher, Microprocess. Microsyst. 45 (2016) 45–55.

[3] W. Zhang, Z. Bao, D. Lin, V. Rijmen, B. Yang, I. Verbauwhede, RECTANGLE: a bit-slice lightweight block cipher suitable for multiple platforms, Sci. China Inf. Sci. 58 (12) (2015) 1–15.

[4] R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, L. Wingers, The SIMON and SPECK families of lightweight block ciphers, Cryptol. Eprint Arch. (2013).

[5] G. Yang, B. Zhu, V. Suder, M.D. Aagaard, G. Gong, The simeck family of lightweight block ciphers, in: International Workshop on Cryptographic Hardware and Embedded Systems, Springer, 2015, pp. 307–329.

[6] S. Banik, S.K. Pandey, T. Peyrin, Y. Sasaki, S.M. Sim, Y. Todo, GIFT: a small present, in: International Conference on Cryptographic Hardware and Embedded Systems, Springer, 2017, pp. 321–345.

[7] C. Beierle, G. Leander, A. Moradi, S. Rasoolzadeh, CRAFT: lightweight tweakable block cipher with efficient protection against DFA attacks, IACR Trans. Symmetric Cryptol. 2019 (1) (2019) 5–45.

[8] A. Bogdanov, L.R. Knudsen, G. Leander, C. Paar, A. Poschmann, M.J. Robshaw, Y. Seurin, C. Vikkelsoe, PRESENT: An ultra-lightweight block cipher, in: International Workshop on Cryptographic Hardware and Embedded Systems, Springer, 2007, pp. 450–466.

[9] S. Banik, A. Bogdanov, T. Isobe, K. Shibutani, H. Hiwatari, T. Akishita, F. Regazzoni, Midori: A block cipher for low energy, in: International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2015, pp. 411–436.

[10] J. Borghoff, A. Canteaut, T. Güneysu, E.B. Kavun, M. Knezevic, L.R. Knudsen, G. Leander, V. Nikov, C. Paar, C. Rechberger, et al., PRINCE–a low-latency block cipher for pervasive computing applications, in: International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2012, pp. 208–225.

[11] Y. Guo, L. Li, B. Liu, Shadow: A lightweight block cipher for IoT nodes, IEEE Internet Things J. 8 (16) (2021) 13014–13023.

[12] J.B. Kam, G.I. Davida, Structured design of substitution-permutation encryption networks, IEEE Trans. Comput. 28 (10) (1979) 747–753.

[13] H. Feistel, Cryptography and computer privacy, Sci. Am. 228 (5) (1973) 15–23.

[14] A. Webster, S.E. Tavares, On the design of S-boxes, in: Conference on the Theory and Application of Cryptographic Techniques, Springer, 1985, pp. 523–534.

[15] Y.M. Motara, B. Irwin, Sha-1 and the strict avalanche criterion, in: 2016 Information Security for South Africa, ISSA, IEEE, 2016, pp. 35–40.

[16] A. Baksi, Machine learning-assisted differential distinguishers for lightweight ciphers, in: Classical and Physical Security of Symmetric Key Cryptographic Algorithms, Springer, 2022, pp. 141–162.

[17] M. Matsui, Linear cryptanalysis method for DES cipher, in: Workshop on the Theory and Application of of Cryptographic Techniques, Springer, 1993, pp. 386–397.

[18] S. Sun, L. Hu, P. Wang, K. Qiao, X. Ma, L. Song, Automatic security evaluation and (related-key) differential characteristic search: application to SIMON, PRESENT, LBlock, DES (L) and other bit-oriented block ciphers, in: International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2014, pp. 158–178.

[19] S. Kölbl, G. Leander, T. Tiessen, Observations on the SIMON block cipher family, in: Annual Cryptology Conference, Springer, 2015, pp. 161–185.

[20] E. Biham, A. Biryukov, A. Shamir, Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials, in: Advances in Cryptology—EUROCRYPT'99: International Conference on the Theory and Application of Cryptographic Techniques Prague, Czech Republic, May 2–6, 1999 Proceedings 18, Springer, 1999, pp. 12–23.

[21] A. Bogdanov, V. Rijmen, Linear hulls with correlation zero and linear cryptanalysis of block ciphers, Des. Codes Cryptogr. 70 (3) (2014) 369–383.

[22] L. Knudsen, D. Wagner, Integral cryptanalysis, in: International Workshop on Fast Software Encryption, Springer, 2002, pp. 112–127.

[23] Y. Todo, Structural evaluation by generalized integral property, in: Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 2015, pp. 287–314.

[24] Y. Todo, M. Morii, Bit-based division property and application to Simon family, in: International Conference on Fast Software Encryption, Springer, 2016, pp. 357–377.

[25] Z. Xiang, W. Zhang, Z. Bao, D. Lin, Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers, in: International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2016, pp. 648–678.

[26] W. Diffie, M. Hellman, Special feature exhaustive cryptanalysis of the NBS data encryption standard, Computer 6 (10) (1977) 74–84.

[27] S. Roy, S. Roy, A. Biswas, K.L. Baishnab, LCB: Light cipher block an ultrafast lightweight block cipher for resource constrained IOT security applications, KSII Trans. Internet Inf. Syst. (TIIS) 15 (11) (2021) 4122–4144.

[28] Z. Lu, W. Wang, K. Hu, Y. Fan, L. Wu, M. Wang, Pushing the limits: Searching for implementations with the smallest area for lightweight s-boxes, in: International Conference on Cryptology in India, Springer, 2021, pp. 159–178.

[29] Z. Gong, S. Nikova, Y.W. Law, KLEIN: A new family of lightweight block ciphers, RFIDSec 7055 (2011) 1–18.

[30] M. Izadi, B. Sadeghiyan, S.S. Sadeghian, H.A. Khanooki, MIBS: A new lightweight block cipher, in: Cryptology and Network Security: 8th International Conference, CANS 2009, Kanazawa, Japan, December 12-14, 2009. Proceedings 8, 2009.

[31] H. Cheng, H.M. Heys, C. Wang, Puffin: A novel compact block cipher targeted to embedded digital systems, in: 2008 11th EUROMICRO Conference on Digital System Design Architectures, Methods and Tools, IEEE, 2008, pp. 383–390.

[32] J. Guo, T. Peyrin, A. Poschmann, M. Robshaw, The LED block cipher, in: Cryptographic Hardware and Embedded Systems–CHES 2011: 13th International Workshop, Nara, Japan, September 28–October 1, 2011. Proceedings 13, Springer, 2011, pp. 326–341.

[33] K. Shibutani, T. Isobe, H. Hiwatari, A. Mitsuda, T. Akishita, T. Shirai, Piccolo: An ultra-lightweight blockcipher, in: CHES, vol. 6917, Springer, 2011, pp. 342–357.

[34] C. Beierle, J. Jean, S. Kölbl, G. Leander, A. Moradi, T. Peyrin, Y. Sasaki, P. Sasdrich, S.M. 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 36, Springer, 2016, pp. 123–153.

[35] C.H. Lim, T. Korkishko, mCrypton–a lightweight block cipher for security of low-cost RFID tags and sensors, in: International Workshop on Information Security Applications, Springer, 2005, pp. 243–258.

[36] R.M. Needham, D.J. Wheeler, Tea extensions, Report, Cambridge University, Cambridge, UK, 1997, 1997.

Yongchao Li received the B.S. degree from Jishou University, China in 2020. He is currently working toward a Master's degree in Guilin University of Electronic Technology, China since 2021. His current research interests include the design and analysis of symmetric encryption algorithms.

Jingya Feng received the B.S. degree from Hebei University of Science and Technology, China in 2015, and received the M.S. from Hunan Normal University, China in 2021. Since 2021, she has been a Ph.D. student of Guilin University of Electronic Technology, China. Her research interests include the design and optimization implementation of symmetric encryption algorithms.

Qi Zhao received the B.S. degree from Taiyuan University of Technology, China in 2019. He is currently working toward a Master's degree in Guilin University of Electronic and Technology, China since 2020. His current research interests include the differential analysis of symmetric encryption algorithms.

Yongzhuang Wei received the M.S. and the Ph.D. degrees in cryptology from Xidian University, Xi'an, China, in 2004 and 2009, respectively. Since July 2011, he has been doing research with the State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences, Beijing, China. Since September 2014, he joined the Guangxi Key Laboratory of Cryptography and Information Security at Guilin University of Electronic Technology, where he is currently employed as a full professor. He is now a member of Chinese Association for Cryptologic Research (CACR). His current research interests include Boolean functions, stream ciphers, block ciphers, and hash functions.