Advances in Applied Mathematics 应用数学进展, 2022, 11(7), 4690-4697
Published Online July 2022 in Hans. http://www.hanspub.org/journal/aam
https://doi.org/10.12677/aam.2022.117495

Hans汉斯

基于SPN结构的分组密码算法ASD

谢　歆

西北师范大学，数学与统计学院，甘肃　兰州

收稿日期：2022年6月15日；录用日期：2022年7月12日；发布日期：2022年7月19日

摘　要

本文提出了一种轻量级分组密码算法ASD，该算法明文长度为64比特，密钥长度为80比特和128比特。算法整体采用SPN结构，混淆层采用16个并置的S盒运算，其中S盒为最优S盒；扩散层为PRESENT该部件的旋转。通过混合整数线性规划(MILP)寻找最小活跃S盒个数进行安全性分析，结果表明ASD具有足够的安全冗余。

关键词

轻量级分组密码，SPN结构，MILP，安全性分析

An Lightweight Block Cipher ASD Based on SPN Structure

Xin Xie

College of Mathematics and Statistics, Northwest Normal University, Lanzhou Gansu

Received: Jun. 15th, 2022; accepted: Jul. 12th, 2022; published: Jul. 19th, 2022

Abstract

This paper proposes a lightweight block cipher algorithm ASD, which has a plaintext length of 64 bits with key length of 80 bits and 128 bits. The algorithm adopts SPN structure as a whole, and the confusion layer adopts 16 concurrent S box operations, of which the S box is the optimal S box. The diffusion layer is present for the rotation of the part. Security analysis was performed by mixed integer linear programming (MILP) to find the minimum number of active S boxes, and the results showed that ASD had sufficient security margins.

文章引用: 谢歆. 基于 SPN 结构的分组密码算法 ASD [J]. 应用数学进展, 2022, 11(7): 4690-4697.
DOI: 10.12677/aam.2022.117495

Keywords

Lightweight Block Cipher, SPN Structure, MILP, Security Cryptanalysis

1. 引言

随着信息技术的飞速发展，各类安全问题引起广泛的关注，信息泄露等问题也愈演愈烈，加密成为重要的问题，分组加密算法是现代密码学的重要基础之一，在保护数据的安全和隐私方面发挥着不可替代的作用。5G 时代的到来和智能设备的普及，各类应用场景中资源受限的软硬件实现平台对密码算法的实现性质和表现提出了更加严苛的要求，也催生了众多轻量级密码算法设计的新思想。

目前分组密码所采用的整体结构主要为 Feistel 结构和 SPN 结构。加解密相似是 Feistel 结构的一个优点，但扩散速度比较慢。与 Feistel 结构相比，SPN 结构可以实现更快速的扩散。适合资源约束应用的密码需求越来越多，著名的轻量级分组密码算法有 CLEFIA [1]，PRESENT [2]，GIFT [3]，Midori [4]。这些密码都是专门针对资源受限的环境而设计的，如 RFID 标签和传感器网络。而这远远不够，仍需大量可靠的轻量级密码。安全性分析方面，Biham 和 Shamir 提出了差分密码分析[5]，Matsui 于 1993 年提出线性密码分析[6]，且广泛应用于各种分组密码。针对这两种分析方法，通常有两种方法评估算法的安全性：一种是计算差分或线性活跃 S 盒的最小个数，以获得最大概率或绝对线性偏差的上界；另一种是寻找一条好的差分路径或线性迹来计算最大概率或绝对线性偏差。孙等[7]提出的基于 MILP 的自动化搜索算法，可以得到活跃 S 盒的最小个数和最大概率，目前被广泛应用于各种分组密码算法。

受 PRESENT 启发，本文提出了一种轻量级的分组密码算法 ASD。采用 SPN 结构来构造 ASD 算法，算法版本是 ASD-64-80 和 ASD-64-128，支持 64 比特长度的明文分组以及 80 比特和 128 比特的密钥。在整体结构上，ASD 算法采用 SPN 结构设计，S 盒采用与 PRESENT 算法等价的 S 盒，其各项密码学性质达到最优，为最轻的 S 盒之一，扩散层采用 PRESENT 的旋转操作，具有很好的密码学性质。在安全性方面，利用混合整数线性规划的自动化分析方法寻找最少活跃盒个数进行差分分析和线性分析。实验结果表明，ASD 算法可以抵抗上述两种攻击。

本文结构安排如下：第二节给出算法的结构，第三节给出算法的设计准则，第四节进行算法分析，最后，总结全文。

2. ASD 分组密码算法

ASD 是一个轻量级分组密码算法，分组长度支持 64 比特，密钥长度支持 80 比特和 128 比特，分别记作 ASD-64-80 和 ASD-64-128，迭代轮数为 31 轮。

2.1. 符号

X：n 比特明文

Y：n 比特密文

R_i：第 i 轮的轮常数

K_i：第 i 轮的轮密钥

$\oplus$：异或

S：4 比特 S 盒

2.2. ASD 算法描述

ASD 算法整体采用 SPN 结构，轮函数包含三个步骤：ASD 算法整体采用 SPN 结构，轮函数 F 包含三个步骤：轮密钥加(AddRoundKey)、S 盒替换(SubNibble)、P 置换(Permutation)操作。算法结构见图 1，其中 K_i 为轮密钥。

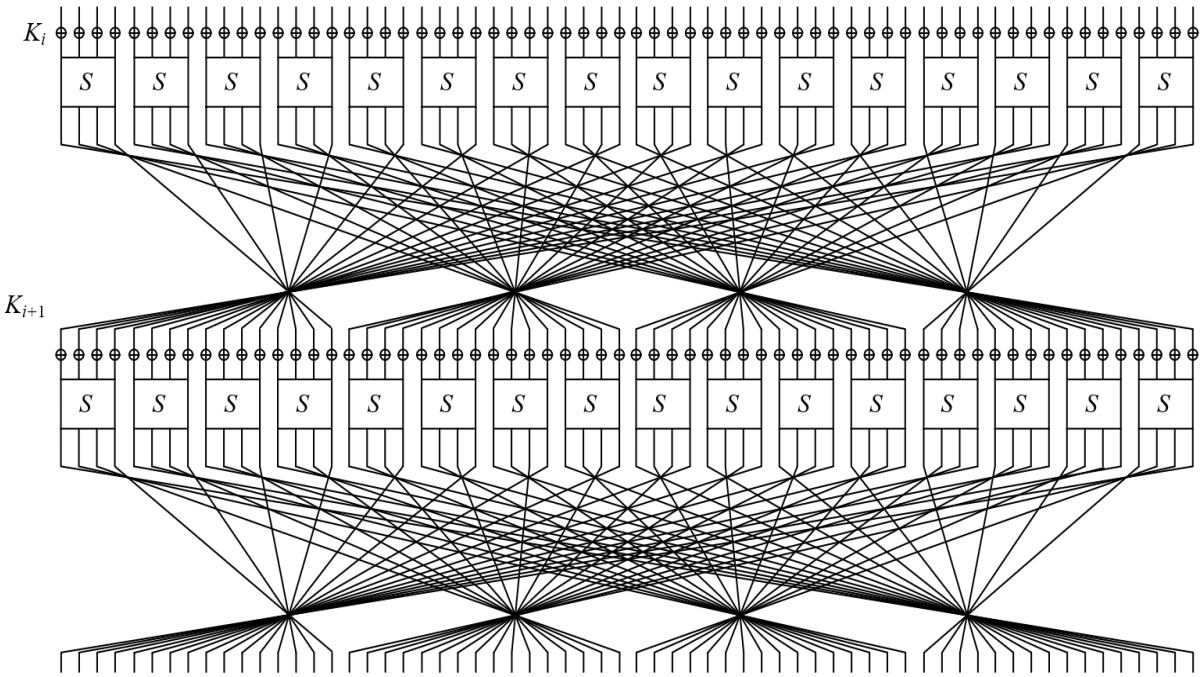

Figure 1. Diagram of ASD algorithm structure
图 1. ASD 算法结构图

2.2.1. 轮密钥加(Add Round Key)

将明文 X 逐比特异或轮密钥 K_i，获得输出状态：

$$V = X \oplus K_i \tag{1}$$

其中 $X = (x_{63}, x_{62}, \cdots, x_0)$ 表示 64 比特明文，$V = (v_{63}, v_{62}, \cdots, v_0)$ 表示轮密钥加以后的状态。

2.2.2. S 盒替换(Sub Nibble)

替换是基于半字节的非线性替换，将 64 比特状态 V 划分为 16 个 4 比特块，进行 S 盒操作，获得输出状态：

$$W = S(V) \tag{2}$$

其中 $W \triangleq (w_{63}, w_{62}, \cdots, w_0)$ 表示 S 盒替换完成后的状态，具体见表 1。

2.2.3. 扩散层 P (Permutation)

将 64 的状态进行 P 置换，ASD 算法的 P 置换。输入的第 i 比特对应输出的第 $P(i)$比特，例如，输入的第 0 比特对应输出的第 63 比特，输入的第 15 比特对应输出的第 12 比特，具体见表 2。

Table 1. S box truth table of ASD algorithm
表 1. ASD 算法的 S 盒真值表

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

Table 2. P Permutation of ASD algorithm
表 2. ASD 算法的 P 置换

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

2.2.4. 密钥扩展

将 80 比特种子密钥 $K=k_{79}k_{78}\cdots k_1k_0$ 放置在寄存器中，每轮提取左边 64 比特作为轮密钥 $K_i=k_{63}k_{62}\cdots k_1k_0=k_{79}k_{78}\cdots k_{17}k_{16}$。对 K 做如下更新：

1) $\left[k_{79}k_{78}\cdots k_1k_0\right]=\left[k_{18}k_{17}\cdots k_{20}k_{19}\right]$；
2) $\left[k_{79}k_{78}k_{77}k_{76}\right]=S\left[k_{79}k_{78}k_{77}k_{76}\right]$；
3) $\left[k_{19}k_{18}k_{17}k_{16}k_{15}\right]=\left[k_{19}k_{18}k_{17}k_{16}k_{15}\right]\oplus R_i, i=1,2,\cdots,32$。

首先，将种子密钥 K 循环右移 19 比特；然后，将种子密钥最左端的 4 比特进行 S 盒操作，为了提高运行效率，密钥扩展算法的 S 盒采用加密算法中的 S 盒；最后，将 $k_{19}k_{18}k_{17}k_{16}k_{15}$ 这 5 比特与 R_i 进行异或操作，R_i 取轮密钥最右边的 5 比特 $k_4k_3k_2k_1k_0$。K 更新完成。

将 128 比特种子密钥 $K=k_{127}k_{126}\cdots k_1k_0$ 放置在寄存器中，每轮提取左边 64 比特作为轮密钥 $K_i=k_{63}k_{62}\cdots k_1k_0=k_{127}k_{126}\cdots k_{65}k_{64}$。对 K 做如下更新：

1) $\left[k_{79}k_{78}\cdots k_1k_0\right]=\left[k_{62}k_{61}\cdots k_{64}k_{63}\right]$；
2) $\left[k_{119}k_{118}k_{117}k_{116}\right]=S\left[k_{119}k_{118}k_{117}k_{116}\right]$；
3) $\left[k_{99}k_{98}k_{97}k_{96}\right]=S\left[k_{99}k_{98}k_{97}k_{96}\right]$；
4) $\left[k_{79}k_{78}k_{77}k_{76}\right]=S\left[k_{79}k_{78}k_{77}k_{76}\right]$；
5) $\left[k_{57}k_{56}k_{55}k_{54}k_{53}\right]=\left[k_{57}k_{56}k_{55}k_{54}k_{53}\right]\oplus R_i, i=1,2,\cdots,32$。

首先，将种子密钥 K 循环右移 63 比特；然后，将种子密钥中间的 12 比特进行 S 盒操作，为了提高运行效率，密钥扩展算法的 S 盒采用加密算法中的 S 盒；最后，将 $k_{57}k_{56}k_{55}k_{54}k_{53}$ 这 5 比特与 R_i 进行异或操作，R_i 取为轮密钥的中间 5 比特 $k_{75}k_{74}k_{73}k_{72}k_{71}$。$K$ 更新完成。

3. 设计准则

3.1. S 盒的设计

3.1.1. 相关定义及定理

定义 1 (差分均匀度)令 S 表示一个 4 × 4 的 S 盒。对任意非零输入差分和输出差分 $\alpha,\beta\in F_2^n$，定义集合 $D_S(\alpha\to\beta)=\{x\in F_2^n \mid S(x\oplus\alpha)\oplus S(x)=\beta\}$，集合 $D_S(\alpha\to\beta)$ 中元素的个数为 $\delta_S(\alpha,\beta)$，则函数 S 的差分均匀度为 $\delta(S)=\max_{\alpha\neq 0,\beta}\delta_S(\alpha,\beta)$。

定义 2 (线性度)令 S 表示一个 4 × 4 的 S 盒。对任意非零输入掩码和输出掩码 $\alpha,\beta\in F_2^n$，令 $Imb_S(\alpha,\beta)=\left|\#\{x\in F_2^n \mid \alpha\cdot x=\beta\cdot S(x)\}-8\right|$，其中，“·”为内积运算，则函数 S 的线性度为 $\lambda(S)=\max_{\alpha,\beta\in F_2^n,\beta\neq 0}2Imb_S(\alpha,\beta)$。

对于任意 4 × 4 的双射 S 盒，均有 $\delta(S)\geq 4$，$\lambda(S)\geq 8$，使得这两个值都达到最小值的 S 盒被称为最优 S 盒。

定义 3 (最优 S 盒[8])令 S 表示一个 4 × 4 的 S 盒，若满足 1) S 是双射；2) $\delta(S)=4$；3) $\lambda(S)=8$ 这三个条件，称其为最优 S 盒。

Eslice 选取最优 S 盒的规则如下：

1) S 是双射，即对任意 $x\neq x^*$，有 $S(x)\neq S(x^*)$。

2) 由差分分布表(表 3)可知，对任意非零输入差分和输出差分 $\alpha,\beta\in F_2^n$，有 $\delta_S(\alpha,\beta)\leq 4$，由定义 1 知，$\delta(S)=4$。

3) 由线性逼近表(表 4)可知，对任意非零输入掩码和输出掩码 $\alpha,\beta\in F_2^n$，有 $Imb_S(\alpha,\beta)\leq 4$，由定义 2 知，$\lambda(S)=8$。

4) S 没有不动点，即对任意的 $x\in F_2^4$，有 $S(x)\neq x$。

定义 4 (差分活跃 S 盒[9])在一条 i 轮差分特征 $\Omega=(\beta_0,\beta_1,\cdots,\beta_i)$ 中，若第 j 轮($j\leq i$)的输入差分 β_{i-1}，导致该轮某个 S 盒的输入差分非零，则称这条差分特征导致该 S 盒活跃，简称该 S 盒是差分活跃 S 盒。

定义 5 (线性活跃 S 盒[9])在一条 i 轮线性特征 $\Omega=(\beta_0,\beta_1,\cdots,\beta_i)$ 中，若第 j 轮($j\leq i$)的输出掩码 β_{i-1}，导致该轮某个 S 盒的输出掩码非零，则称这条线性特征导致该 S 盒活跃，简称该 S 盒是线性活跃 S 盒。

3.1.2. S 盒的选取

ASD 算法 S 盒在设计时主要遵循下列准则：

1) 单比特输入差分能引起单比特输出差分的差分特征个数为 0；

2) S 盒是最优 S 盒；

3) S 盒没有不动点，即对于任意 $v\in F_2^4$，有 $S(v)\neq v$；

4) 单比特输入掩码能引起单比特输出掩码对应偏差非零的线性特征个数为 4。

基于上述准则选择了 ASD 算法的 S 盒，S 盒的选择对于 SPN 结构算法的安全性具有极大的影响。相较于 8 比特 S 盒，之所以选择 4 比特 S 盒，是因为 8 比特 S 盒不适合轻量级环境。对于选择 4 比特 S 盒作为扩散层的分组密码算法，以 Leander 等人[8]的仿射等价类研究为基础，选择了一个与 PRESENT 算法的 S 盒仿射等价的 S 盒作为 ASD 算法的 S 盒。该 S 盒具有 PRESENT 算法的 S 盒所有的优点。

3.2. 扩散层设计

扩散层作为分组密码的核心部分，其目的是提高扩散和混淆程度来实现雪崩效应，这有助于抵抗差分分析、线性分析以及一些未知分析方法的攻击。它的设计不仅影响算法的安全性，还影响分组密码在

软硬件中的实现效率,在实际应用中迫切需求具有轻量级的扩散层，采用按比特进行拉线操作的扩散层极大地减少硬件面积以及更好地防御侧信道攻击，因此本文采用比特级拉线操作构造扩散层，且该扩散层具有较高的扩散性能。

3.3. ASD 的密钥扩展算法设计

密钥扩展算法设计准则如下：

1) 密钥扩展算法采用与加密算法相同的 S 盒，以此减少实现代价；
2) 使用轮常数消除对称性；
3) 采用简单的逻辑运算，易于实现。

4. 安全性分析

实验平台的硬件环境为处理器：AMD Ryzen 5 5600U，内存 16GB，操作系统：Windows 10。采用 MILP 搜索算法进行安全性分析。

4.1. 差分分析

差分和线性密码分析是分组密码最强大的技术之一。要使用差分密码分析(DC)攻击 n 比特分组密码，需要找到一条概率大于 2^{1-n} 的差分传播，差分传播由一组微分特征组成，其概率是具有指定输入差分特征和输出差分特征的概率之和。应用 MILP 搜索模型，搜索差分活跃 S 盒个数的方法评估算法抵抗攻击的能力，得到了 16 轮活跃 S 盒个数。差分活跃 S 盒个数的结果见表 3：

Table 3. The minimum number of active S box of ASD algorithm
表 3. ASD 算法活跃 S 盒的最小个数

轮数	#{AS}	轮数	#{AS}	轮数	#{AS}	轮数	#{AS}
1	1	5	10	9	18	13	26
2	2	6	12	10	20	14	28
3	4	7	14	11	22	15	30
4	6	8	16	12	24	16	32

通过差分活跃 S 盒个数的下界评估最大差分概率的上界。本文通过 MILP 模型搜索了 ASD-64 的结果，16 轮最小差分活跃 S 盒的个数为 32，由 S 盒的差分分布表(见表 4)可知该算法 S 盒的最大差分概率为 2^{-2}，因此，估算 16 轮差分特征的概率约为：$DP_{\max} < 2^{-2\times 32} = 2^{-64} \le 2^{-64}$，根据安全性分析，ASD-64 的绝对安全冗余为 16 轮。实验结果也表明，16 轮差分特征概率为 $2^{-70} \le 2^{-64}$，因此，认为该算法能够抵抗差分攻击。

4.2. 线性分析

运用类似差分分析的方法，进一步评估算法抵抗线性攻击的能力。通过估算线性特征中活跃 S 盒个数的下界，评估线性特征最大偏差概率的上界。采用 MILP 自动化搜索算法搜索 ASD-64 的结果，线性活跃 S 盒个数的结果同差分结果一致，由 S 盒的线性分布表(见表 5)可知该算法 S 盒的最大线性概率为 2^{-2}，估算 16 轮的线性偏差为：$LP_{\max} \le 2^{32-1}\cdot 2^{-2\times 32} = 2^{-33}$；考虑到迭代轮数为 31，因此，我们认为该算法能够抵抗线性攻击。

Table 4. Differential Distribution Table of S-box
表 4. S 盒差分分布表

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	2	0	0	4	2	0	0	0	2	0	0	4	2
2	0	0	0	0	0	0	2	2	2	0	2	0	2	4	0	2
3	0	0	0	2	0	0	2	0	2	4	2	2	2	0	0	0
4	0	0	0	4	0	0	0	4	0	0	0	4	0	0	0	4
5	0	2	0	0	4	2	0	0	4	2	0	0	0	2	0	0
6	0	2	4	0	2	0	0	0	0	0	0	2	2	2	0	2
7	0	0	4	0	2	2	0	0	0	2	0	2	2	0	0	2
8	0	2	0	2	0	2	0	2	0	2	0	2	0	2	0	2
9	0	2	0	0	0	2	4	0	0	2	0	0	0	2	4	0
A	0	0	0	0	0	4	2	2	2	0	2	0	2	0	0	2
B	0	4	0	2	0	0	2	0	2	0	2	2	2	0	0	0
C	0	0	0	0	4	0	0	0	4	0	4	0	0	0	4	0
D	0	2	0	0	0	2	0	0	0	2	4	0	0	2	4	0
E	0	0	4	2	2	2	0	2	0	2	0	0	2	0	0	0
F	0	2	4	2	2	0	0	2	0	0	0	0	2	2	0	0

Table 5. Linear Approximation Table of S-box
表 5. S 盒线性逼近表

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	4	0	−4	0	0	2	−2	−2	−2	−2	−2	2	−2
2	0	0	0	0	0	0	4	4	0	0	4	−4	0	0	0	0
3	0	0	0	−4	4	0	0	0	−2	2	−2	−2	−2	−2	2	−2
4	0	0	0	0	0	0	−4	4	0	0	0	0	0	0	4	4
5	0	0	−4	0	0	−4	0	0	−2	2	−2	−2	2	2	−2	2
6	0	0	0	0	0	0	0	0	4	4	0	0	−4	4	0	0
7	0	0	−4	0	−4	0	0	0	−2	2	2	2	−2	−2	2	−2
8	0	0	0	−4	−2	−2	2	−2	0	−4	0	0	−2	2	2	2
9	0	0	0	0	−2	2	2	−2	2	2	−2	−2	4	0	4	0
A	0	0	0	−4	−2	−2	−2	2	4	0	0	0	2	−2	−2	−2
B	0	0	0	0	2	−2	2	−2	2	2	2	2	0	−4	0	4
C	0	4	0	0	−2	2	−2	−2	0	0	0	−4	−2	−2	−2	2
D	0	4	4	0	−2	−2	2	2	−2	2	−2	2	0	0	0	0
E	0	−4	0	0	−2	2	2	2	0	0	−4	0	−2	−2	−2	2
F	0	4	−4	0	2	2	2	2	2	−2	−2	2	0	0	0	0

4.3. SPN 结构分析

SPN 结构中的 S 是指替换(Substitution)，P 是指置换或更广泛的线性变换(Permutation)。SPN 结构是目前广泛使用的一种分组密码整体结构，PRESENT，GIFT 和 Midori 等分组密码都采用该结构。SPN 结构的原理是在这种密码结构的每一轮中，首先每一轮的输入经过一个可逆函数 S 作用，其中可逆函数 S 由子密钥控制，然后再作用到一个置换 P。SPN 结构清晰，S 层一般被称为混淆层，主要起混淆作用。P 一般被称为扩散层，主要起扩散的作用。直观来看，先经过混淆层，再经过扩散层，就很接近 Shannon 所提出的混淆原则和扩散原则，而现代分组密码将混淆层和扩散层通过整体结构迭代多次，将会增强密码的混淆性和扩散性，使得密码的输入和输出之间的依赖关系更为复杂。和 Feistel 结构相比，SPN 结构优势在于可以得到更快的扩散。综上所述，该结构具有足够的安全性。

5. 总结

本算法采用分组密码算法常用的 SPN 结构，该结构优点在于适用于资源受限的环境，通过 MILP 估计了活跃 S 盒的个数进行计算，以此计算差分特征的概率和线性偏差，实验表明该算法该可以抵抗差分分析和线性分析等安全密码分析，具有绝对安全冗余 16 轮。

参考文献

[1] Shirai, T., Shibutani, K., Akishita, T., Moriai, S. and Iwata, T. (2007) The 128-Bit Blockcipher CLEFIA. *International Workshop on Selected Areas in Cryptography*, Berlin, Heidelberg, 28 March 2007, 181-195. https://linkspringer.53yu.com/chapter/10.1007/978-3-540-74619-5_12 https://doi.org/10.1007/978-3-540-74619-5_12

[2] Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J. and Vikkelsoe, C. (2007) PRESENT: An Ultra-Lightweight Block Cipher. *International Workshop on Cryptographic Hardware and Embedded Systems*, Berlin, Heidelberg, 10 September 2007, 450-466. https://linkspringer.53yu.com/chapter/10.1007/978-3-540-74735-2_31 https://doi.org/10.1007/978-3-540-74735-2_31

[3] Banik, S., Pandey, S.K., Peyrin, T., Sasaki, Y., Sim, S.M. and Todo, Y. (2017) GIFT: A Small Present. *International Conference on Cryptographic Hardware and Embedded Systems*, Cham, 25 August 2017, 321-345. https://linkspringer.53yu.com/chapter/10.1007/978-3-319-66787-4_16 https://doi.org/10.1007/978-3-319-66787-4_16

[4] Banik, S., Bogdanov, A., Isobe, T., Shibutani, K., Hiwatari, H., Akishita, T. and Regazzoni, F. (2015) Midori: A Block Cipher for Low Energy. *International Conference on the Theory and Application of Cryptology and Information Security*, Berlin, Heidelberg, 30 December 2015, 411-436. https://linkspringer.53yu.com/chapter/10.1007/978-3-662-48800-3_17 https://doi.org/10.1007/978-3-662-48800-3_17

[5] Biham, E. and Shamir, A. (1992). Differential Cryptanalysis of the Full 16-Round DES. *Annual International Cryptology Conference*, Berlin, Heidelberg, 16 August 1992, 487-496. https://linkspringer.53yu.com/chapter/10.1007/3-540-48071-4_34 https://doi.org/10.1007/3-540-48071-4_34

[6] Matsui, M. (1993) Linear Cryptanalysis Method for DES Cipher. *Workshop on the Theory and Application of Cryptographic Techniques*, Berlin, Heidelberg, 27 May 1993, 386-397. https://linkspringer.53yu.com/chapter/10.1007/3-540-48285-7_33 https://doi.org/10.1007/3-540-48285-7_33

[7] Fu, K., Wang, M., Guo, Y., Sun, S. and Hu, L. (2016) MILP-Based Automatic Search Algorithms for Differential and Linear Trails for Speck. *International Conference on Fast Software Encryption*, Berlin, Heidelberg, 20 July 2016, 268-288. https://linkspringer.53yu.com/chapter/10.1007/978-3-662-52993-5_14 https://doi.org/10.1007/978-3-662-52993-5_14

[8] Leander, G. and Poschmann, A. (2007) On the Classification of 4 Bit S-Boxes. *International Workshop on the Arithmetic of Finite Fields*, Berlin, Heidelberg, 21 September 2007, 159-176. https://linkspringer.53yu.com/chapter/10.1007/978-3-540-73074-3_13 https://doi.org/10.1007/978-3-540-73074-3_13

[9] 李超. 分组密码的攻击方法与实例分析[M]//孙兵, 李瑞林. 北京: 科学出版社, 2010: 77-107.