EL*i*F : An Extremely Lightweight & Flexible Block Cipher Family and Its Experimental Security

Adnan Baysal[1,2] and Ünal Kocabaş[1]

[1] TÜBİTAK BİLGEM, Gebze Kocaeli 41470, Turkey
{adnan.baysal,unal.kocabas}@tubitak.gov.tr
[2] Department of Computer Engineering, Kocaeli University,
Umuttepe Yerleşkesi Kocaeli 41380, Turkey

Abstract. In this paper, we analyzed an extreme case of lightweight block cipher design in terms of security and efficiency. To do this, we proposed EL*i*F block cipher family which has one of the smallest hardware area in a fully serial design. We also defined EL*i*F to be flexible and scalable so that it can be implemented for real life applications with different scenarios such as fixed key implementations. We also gave hardware implementation results for different implementation settings to show its efficiency and flexibility. Because of its flexible implementation properties, EL*i*F family of ciphers are suitable for systems with asymmetric computation powers such as RFID reader and tags. We made theoretical and experimental analysis for various block sizes. Using the results for small block lengths, we estimated minimum number of rounds that the cipher becomes secure depending on the block size.

Keywords: WSN, lightweight block cipher, DDT, LAT, algebraic attack, hardware, ASIC, GE

1 Introduction

Design and security analysis of lightweight block ciphers have become a popular research topic in recent years. Most of the lightweight ciphers designed in the last decade have very simple round functions. These ciphers are meant to be used in very constrained devices such as RFID tags and wireless sensor networks. Area is one of the most constrained resource in these devices. Many ciphers have been proposed with the aim of breaking previous implementation area records in terms of gate equivalent (in hardware) and/or code size (in software). Some examples of such ciphers are : PRESENT [9], KTANTAN and KATAN [11], PRINTcipher [16], Simon-Speck [7], PRINCE [10], PRIDE [5], RoadRunneR [6], and RECTANGLE [20].

Block length of a cipher puts a lower bound on the minimum implementation area since the state should be stored in memory, and memory costs area both in software and hardware. Low area lightweight block cipher implementations target to be close to this limit by using serialization techniques. Since serialization

increases total clock cycles to encrypt one block, there is a trade-off between area and speed of a cipher implementation.

In 2013, Bogdanov [2] depicted the round function of the extreme lightweight cipher in Albena Crypto Summer School. In this cipher, only a single non-linear two-input gate, a key XOR and a diffusion XOR are used over an N-bit state. He claimed that at least several thousands of rounds are necessary to provide security in that cipher. However, the picture was very general omitting implementation aspects and no bound depending on the block size was given.

In this paper, we propose an extremely lightweight block cipher family which have similar structure with Bogdanov's 'The Extreme Lightweight Cipher' in his presentation. We designed this family to have a simple description for each block length, and to have flexible implementation properties. We called this family as ELiF (**E**xtremely **L**$\boldsymbol{i}$ghtweight & **F**lexible) block cipher family[3]. We theoretically and experimentally analyzed its cryptanalytic properties to find minimum round number which makes the cipher secure.

The paper is organized as follows: ELiF family of block ciphers is defined in Section 2. Theoretical security against differential and linear cryptanalysis is examined in Section 3. We analyzed some instances of this cipher family experimentally in Section 4. Section 5 gave ASIC implementation results, and Section 6 concluded the paper.

2 ELiF Block Cipher Family

The ELiF family of block ciphers are parametrized by its block size b and round number r, and are denoted by ELiF$_{b,r}$. r is a function of b and one of the main objectives of this paper is to determine a lower bound on $r = f(b)$ from a security and efficient implementation perspective. Key schedule is omitted in the definition to keep flexible implementation properties. The basic serial round function of ELiF$_{b,r}$ is depicted in Fig. 1.

2.1 Serial round function

Definition 1. *Let $b > 2, r > 0$ be two integers and $x_j^{(i)} \in GF(2)$, $j = 0, 1, \ldots, b-1$ be input bits to the round i. Let $k^{(i)} \oplus c^{(i)}$ be the i^{th} round key and constant XOR. The i^{th} round function of ELiF$_{b,r}$ is defined algorithmically in Algorithm 1:*

2.2 n-latency parallel round function

The rotation direction of ELiF round function is selected as left so that $(b-2)$ round parallel implementations do not increase latency in hardware. So it is possible to increase the speed by utilizing extra gates without decreasing clock frequency. An example for $b = 8$ is given in Fig. 2.

[3] The original acronym was ELF, but this name is used recently in an IACR eprint paper, hence it is changed as ELiF.

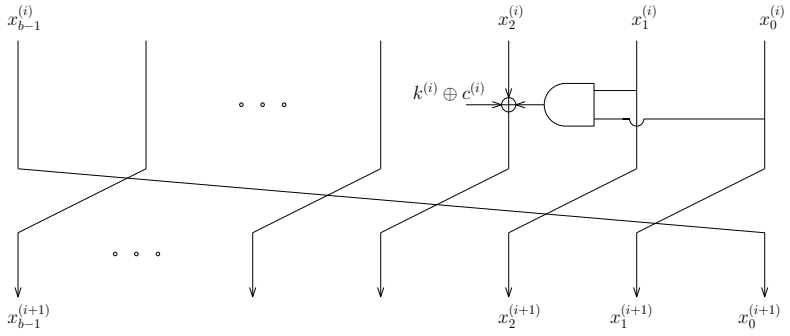


Fig. 1. ELiF$_{b,r}$ serial round function. (i) denotes round number.

Algorithm 1 i^{th} round function of ELiF$_{b,r}$

$x_2^{(i)} \leftarrow k^{(i)} \oplus c^{(i)} \oplus x_2^{(i)} \oplus x_1^{(i)} x_0^{(i)}$
for j from $b-1$ down to 1 **do**
  $x_j^{(i+1)} \leftarrow x_{j-1}^{(i)}$
end for
$x_0^{(i+1)} \leftarrow x_{b-1}^{(i)}$

In Fig. 2, the total round number is divided by six by implementing 6 parallel rounds. In general, it is possible to implement $(b-2)$ parallel rounds without increasing the round latency. In Definition 2, we define a custom gate to be used in ELiF whose NAND gate area and latency depend on the implementation scenario.

Definition 2. *An ELiF gate is a 5 input Boolean function denoted by ELiF-G and defined as follows:*

$$\textit{ELiF-G}(x_0, x_1, x_2, k, c) = x_0 x_1 \oplus x_2 \oplus k \oplus c$$

The gate equivalent (GE) area of a single ELiF-G can be reduced in a fixed key and fixed constant implementation. This will be discussed in Section 5. It is possible to further decrease the round number by allowing more latency in round function (and of course by increasing the total area). For an n ELiF-G latency in the critical path of the cipher, it is possible to implement $n \times (b-2)$ rounds in parallel. This round function is called an n-latency parallel round function of ELiF block cipher. Here, latency puts an upper bound on the maximum achievable clock frequency since it may increase the critical path.

These trade-offs are beneficial for implementing the same algorithm both in lightweight and high throughput devices in the same system. This can be utilized for example in a scenario where there is a single master device which communicates with multiple resource constrained slaves simultaneously (e.g. RFID reader and tags in a WSN).

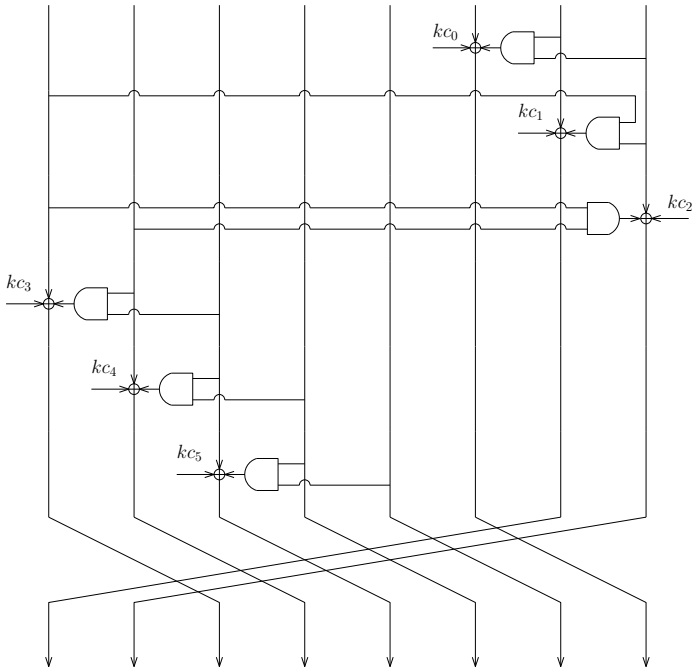


Fig. 2. ELiF$_{8,6}$ 1-latency parallel round function. $kc_i = k_i \oplus c_i$.

A note on inverse round function : The inverse of the ELiF round function is not efficient for parallelization techniques. This is because of the fact that output of the `XOR` gate enters into the `AND` gate in the next round in the reverse direction. Therefore, ELiF family of ciphers are more suitable for encryption only mode of operations when parallel implementation is necessary. Another affect of this situation is that diffusion is better in the decryption direction. This makes attacks in the decryption direction to be ineffficient in less rounds than that of encryption direction.

3 Theoretical Security Against Differential and Linear Cryptanalysis

Differential cryptanalysis [8] and linear cryptanalysis [18] are two successful attacks on block ciphers. They use statistical variations in non-linear layers (S-box layer) of block ciphers to find differential and linear trails with high probability/bias. These trails are used to determine high probability entries in the huge S-box determined by block cipher, which are (almost) independent of the secret key. All differential trails with the same input-output difference and different intermediate differences is called a *differential characteristic*. Similarly, linear trails

with the same input-output mask and different intermediate masks is called a *linear approximation* (or *linear hull*).

A block cipher is in fact a keyed permutation over its domain. So, for each key, it can be seen as a huge S-box. In order to be secure, this S-box should satisfy the property that distinguishing it from a random permutation is computationally infeasible. Moreover, there should be no short cut to determine the secret key from input-output pairs.

The only non-linear operation in EL*i*F family of block ciphers is the `AND` gate. Therefore, the difference distribution table (DDT) and linear approximation table (LAT) of the and gate can be used as a reference guide to calculate probabilities of differential trails and linear trais. DDT and LAT definitions used in this paper is given below ($\cdot$ is the dot product operator on Boolean vectors):

Definition 3. *Let $S : \mathbb{F}_2^b \to \mathbb{F}_2^c$ be a (possibly vectorial) Boolean function with b-bit input and c-bit output. DDT and LAT of S are matrices of size $2^b \times 2^c$ where i^{th} row j^{th} column entries $DDT_S(i,j)$ and $LAT_S(i,j)$ are defined as follows:*

$$DDT_S(i,j) = \#\{x | S(x) \oplus S(x \oplus i) = j\}$$

and

$$LAT_S(i,j) = |\#\{x|(i \cdot x) \oplus (j \cdot S(x)) = 1\} - \#\{x|(i \cdot x) \oplus (j \cdot S(x)) = 0\}|$$

Non-zero entries in each table can be used to generate differential and linear trails. Since we take the absolute value in LAT generation, small values are better in terms of security in each table. For DDT, zero input difference result in zero output difference since S is a function. Similarly, for LAT, zero output mask results in zero entries because input to S function takes all b-bit values in LAT entry calculation. In Tables 1 and 2, we give the DDT and LAT of two input `AND` gate considering the above observation.

	01	10	11
0	2	2	2
1	2	2	2

Table 1. DDT of AND gate.

	00	01	10	11
1	2	2	2	2

Table 2. LAT of AND gate.

In Tables 1 and 2, columns and rows represent input output difference (mask) respectively. Observing the tables, it can be seen that an active `AND` gate has difference and correlation probability 2^{-1}. Therefore at least b active `AND` gates

are necessary for any trail to be unuseful in differential and linear cryptanalysis. Moreover, since there may be other trails with the same input/output difference (mask) and different intermediate differences (masks), the clustering of trails should also be considered while determining the minimal round number.

Tables 1 and 2 also shows that if an AND gate is differentially (resp. linearly) active, that is the input difference (resp. output mask) is non-zero, then the output difference (resp. input mask) can be selected any value with the same probability. The following theorem gives the minimum number of differentially active AND gates in b consecutive serial ELiF rounds with non-zero input difference.

Theorem 1. *For any positive integer $b > 3$ and any non-zero input difference, there are at least 2 active* AND *gates in ELiF$_{b,b}$.*

Proof. Any input bit enters into an AND gate in at most $b-1$ rounds because of the bit permutation. Hence, there is at least one active AND gate. There are three cases:

Case 1 : The first AND gate is active. Then if right hand-side input (cf. Fig. 1) is active (has difference), then the second AND gate is also active. So assume that only the left hand-side input of the first AND gate is active and the second AND gate is passive. Since the second AND gate is passive, this difference cannot be cancelled in the second round, and enters into the last round's AND gate, making it active.

Case 2 : The last AND gate is active. Similar to the case 1, if left hand-side input to this and gate is active, then the previous AND gate is also active. Hence assume that left hand-side input is passive and right hand-side input is active. This active bit can be made passive only if the second round's AND gate is active. If it is kept active, then the first round's AND gate is active. In either case, there are at lest two active AND gate.

Case 3 : An intermediate round's AND gate is active. Then, if left hand-side input is active, then the previous round's AND gate is also active. Otherwise, the next rounds AND gate is active. □

For each 1-bit input difference to ELiF round function, there is an iterative b-round characteristic with two active AND gate. Therefore, the result in Thm. 1 can be extended to any multiple of b rounds. Since an active AND gate has probability 2^{-1}, any differential trail in $\frac{b^2}{2}$ rounds has probability at most 2^{-b}. Clustering of differential trails increases the probability. Therefore more than $\frac{b^2}{2}$ rounds is necessary for the security against classical differential cryptanalysis.

For the linear cryptanalysis, there is a worse bound for b consecutive rounds:

Theorem 2. *For any non-zero input mask to b-bit ELiF there is at least one active* AND *gate in b consecutive rounds.*

Proof. Any active input mask bit enters into an XOR gate in at most b rounds because of the bit permutation layer. When this happens, then the corresponding AND gate is active. Therefore there is at least one active AND gate in b rounds.

Any input linear mask with a single active bit can be used to generate b-round iterative characteristic with a single active `AND` gate. To do this, whenever the `AND` gate is active (i.e. output mask is 1), input mask to this `AND` gate is selected as `00` (this is possible by Table 2). Hence, any b^2-round linear trail has correlation of at most 2^{-b}.

Diffusion of bits : Diffusion of bits in ELiF encryption is slower than decryption, which makes parallel encryption round function efficient. We experimentally analyze the diffusion of each single bit for both direction and find the maximal rounds that a single bit affects all output bits. For even b values ($b \leq 128$), we get the following formula:

Direction	Bit	Max. round
Encryption	2	$\frac{b^2}{2} - b + 2$
Decryption	0	$2b - 3$

Table 3. Maximal rounds for a single bit diffusing to all state bits for even b.

In Table 3, decryption round function starts with the inverse bit permutation. Column 2 gives the bit position satisfying maximal rounds for diffusion. In all tests, these positions did not changed. This can be seen easily by observing the path each bit follows over rounds. When the selected bit enters into an `AND` gate, that bit diffuses to the third bit by the `XOR` gate, which is the only diffusion operation. Hence, a simple algorithm replacing the `AND` gate and `XOR` gate with `OR` gate gives the minimal round that the target bit affects all output bits in linear time complexity.

Observations in this section show that more than b^2 rounds are necessary to make ELiF secure against both classical differential and linear cryptanalysis. In the next section, we experimentally analyze ELiF family for small block lengths to estimate secure round number.

4 Experimental Security Analysis

ELiF family of block ciphers has very similar (and simple) structure for each block length. Hence it is reasonable to guess the behavior of the cipher with higher block sizes using the experimental data on small block lengths. Because of this rationale, we analyzed the cryptographic properties of various ELiF instances for small block lengths. We applied experimental tests to find the minimum round number. For differential and linear cryptanalysis, we chose 8 up to 16 bits block sizes and considered the resulting ciphers as keyed S-boxes. Then we generated the DDT and LAT of these S-boxes for many keys. We used the information gained from this analysis to estimate secure round numbers for higher

block sizes. We also practically applied algebraic cryptanalysis for block sizes less than 15.

All these tests were done experimentally for randomly chosen $2b$-bit keys. For all block sizes, we did not utilize any key schedule and used the keys in a circulant order. We also used an LFSR based constant schedule with a period of $2b^2$-bits.

4.1 Differential analysis and linear analysis

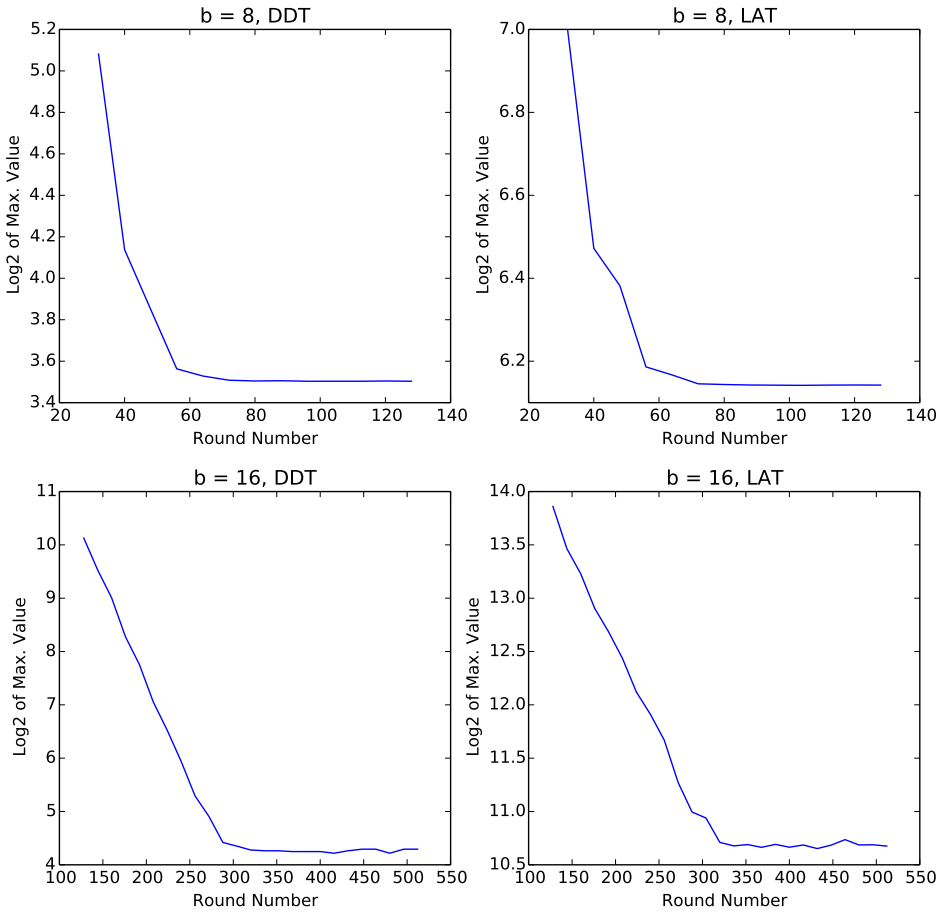
Maximum values in DDT and LAT of a full block cipher corresponds to the maximal probability (resp. correlation) differential characteristics (resp. linear approximation). Differential and linear trails found in Section 3 do not consider the clustering, i.e. trails with the same input-output difference (mask) and different intermediate differences. These trails and falseness of independence assumptions may change overall probability, but it is hard to compute exact probability considering all possible trails. Hence, we analyzed small ELiF variants experimentally to find real characteristic probabilities as a function of b.

Since the calculation of the whole DDT and LAT have complexity $O(2^{2b})$ and $O(b2^{2b})$ respectively (latter is by using fast Walsh transform), it is not practical to find maximal values of DDT and LAT for a random permutation over more than $b = 16$-bit vectors. Therefore, in our experiments, we tried block sizes $b \in \{4, 6, \ldots, 16\}$ (only even b). We considered the maximum probability and maximum absolute correlation entries in the tables for increasing round numbers. If the distribution of became indistinguishable from that of a random permutation, then the cipher is assumed to be secure against differential cryptanalysis and linear cryptanalysis. We searched for the minimum round number where resulting S-boxes behave like a random permutation in its DDT and LAT. Moreover we did the analysis for multiple keys and saved the maximums of DDT and LAT for each key in a histogram. To reduce the size of the tested rounds, we started from $\frac{b^2}{2}$ up to $2b^2$ with steps of b rounds. Line graphs for base 2 logarithm of average DDT and LAT maximum values depending on the round number for block sizes $b = 8$ and $b = 16$ are given in Fig. 3.

The graph of maximum values depending on round number for other b values were very similar to that of in Fig. 3. In all experiments, DDT and LAT maximum values saturated to a b dependent point after around b^2 rounds. Maximum values do not change much after that point. When we analyzed the distribution of maximum values in DDT and LAT for different keys, the variations in maximum values of DDT and LAT maximums were relatively small for round numbers after b^2. DDT and LAT maximum value histogram plots are given in Fig. 4 for $b = 14$ and three different rounds.

It can be seen from Fig. 4 that the distribution of the maximum values in each table become better after b^2 rounds, but the change is marginal after $\frac{3b^2}{2}$ rounds. Similar distributions obtained for other b values. According to a result in [15], DP values of a random S-box of size n-bits should be in the interval $[2B_n, 2n]$ with high probability, where $N = 2^n - 1$ and $B_n = \ln(N)^2/\ln(\ln(N))^2$.

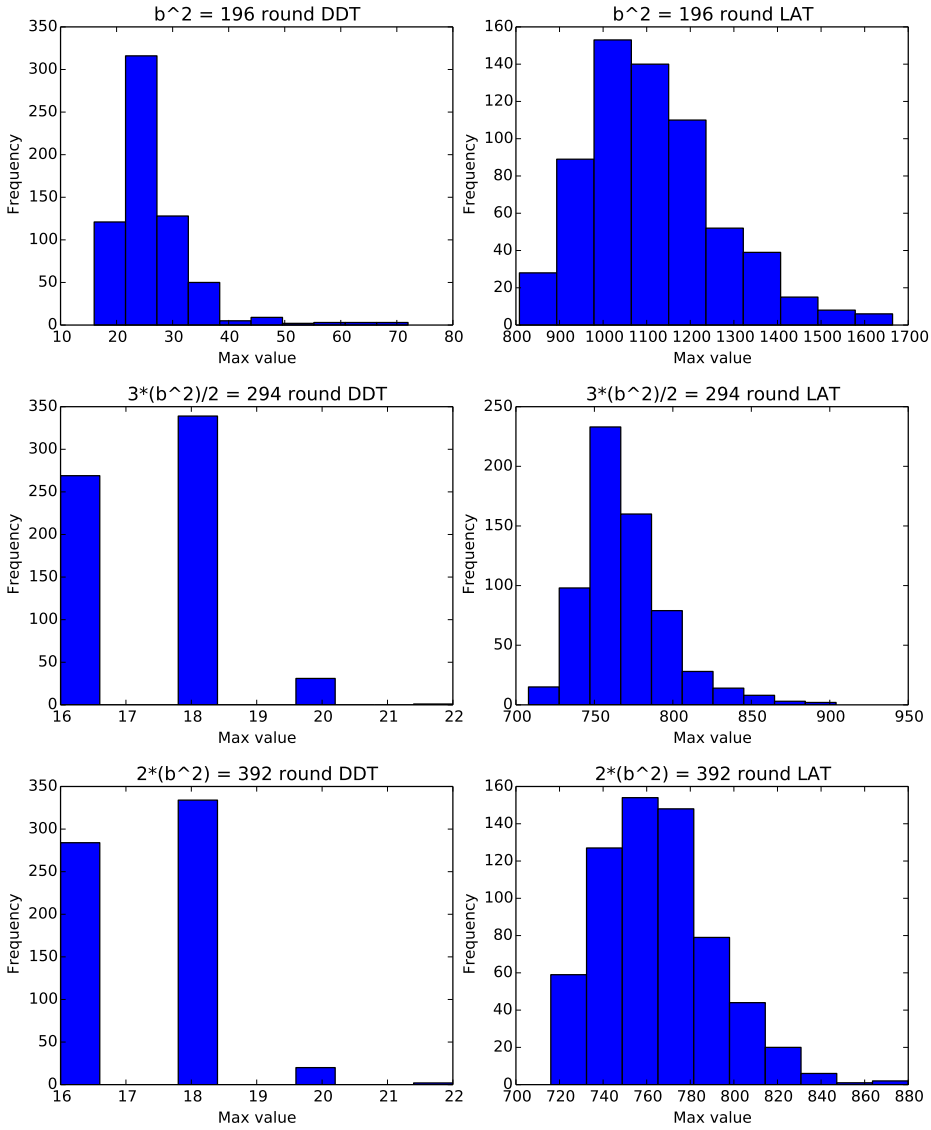
Fig. 3. log_2 of average maximum values in DDT and LAT for $b = 8, 16$, $2b$-bit key, and varying round numbers. 32 random keys are used for $b = 16$, all keys are used for $b = 8$.



Histogram of DDT maximum values satisfied this property after $b^2 + tb$ for some small positive integer t in all experiments.

On the other hand, to compare the distribution of LAT maximum values of ELiF variants with random S-boxes, we experimentally generated 8-bit random S-boxes and analyzed the distribution of LAT maximum values. Distributions for 10000 random S-boxes were very similar to that of 8-bit ELiF variant slightly more than $b^2 = 64$ rounds. Moreover, as seen in Fig. 4 (and for other b parameters in our search space), distributions are similar to that found in [14]. The actual numbers in our graphs are different because of our definition of LAT, but the distribution is almost the same.

Fig. 4. Distribution af DDT and LAT max values for 640 random 28-bit key in b =14-bit ELiF, for the round numbers b^2, $\frac{3b^2}{2}$, and $2b^2$.



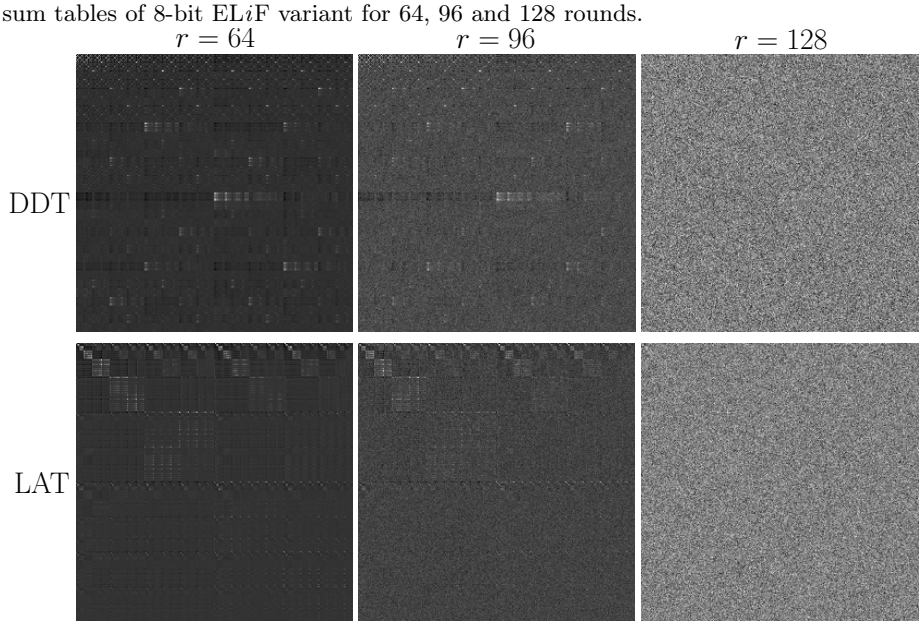
Distribution of characteristics with high probability : Besides analyzing maximum values in DDT and LAT, we also analyzed the placement of input-

output difference and mask values satisfying maximal DP and LP in these tables. To visualize this easily, we applied the following approach:

For a fixed b, and for some set of keys (all keys if computation time is reasonable, or some number of random keys) we generated all DDT's and LAT's of ELiF$_{b,r}$ for changing r. Each r corresponds to a keyed S-box as mentioned previously. For all keyed S-boxes, we calculated two sum matrices for DDT and LAT, which adds all values in indices of DDT over all keys. Higher number positions in these sum tables show higher key independence of the corresponding differential characteristics or linear approximations.

After generating the sum matrices, we excluded first rows and columns since these correspond to zero input-output difference/mask. Then we normalize the values in table by taking the minimum value to zero and maximum value to 255. This data is then shown as a gray-scale graphics where each value shows a gray-scale value. 0 is black and 255 is white in this table. Other values give shades of gray where smaller numbers are darker and higher numbers are lighter. An example graph for $b = 8, r = 64, r = 96$ and $r = 128$ is given in Fig. 5.

Fig. 5. Graphical illustration of normalized DDT sum (over different keys) and LAT sum tables of 8-bit ELiF variant for 64, 96 and 128 rounds.



From Fig. 5, it can be seen that the placement of higher probability characteristics (and approximations) follows an interesting pattern even in $\frac{3b^2}{2} = 96$ rounds of ELiF. This pattern disappears in $2b^2 = 128$ rounds. As usual, graphs were similar for other b values.

Patterns in $\frac{3b^2}{2}$ rounds may be used to apply differential (and linear) attacks with a single input difference (mask) and multiple output difference (mask). In this attack, if output difference (mask) obeys the pattern, a counter may be increased. Depending on the probabilities and number of output differences (masks), total differential (linear) probability may be increased. Therefore, data complexity may be significantly lower for $\frac{3b^2}{2}$ rounds. Hence, these patterns should be avoided which is the case with $2b^2$ rounds.

4.2 Algebraic analysis

In algebraic cryptanalysis [12], a cipher is represented as a system of multivariate polynomial equations, and this system is tried to be solved by an appropriate solver using the known plaintext-ciphertext pairs. If applicable, a few pairs is enough for the attack to work. To reduce the degree of resulting polynomials, new variable names are assigned and corresponding equations are added to the system at each round. To be able to find a finite (and small) set of solutions, the system is better if it is over-defined, i.e. there should be more equations than variables.

ELiF family of ciphers have very simple round function which consist of a single AND operation. So generating equations is straightforward. In this analysis, we assigned a new variable name for each ELiF-G output, and insert the resulting equation to the system. Hence the degree of the equations are at most quadratic. We used $2b$-bit keys in a circulant manner and used an LFSR based constant schedule.

SAGE computer algebra system [3] is used for both generating and solving the equation system.We assigned each key bit and each plaintext bit a variable name. If the serial round number is r, then there are precisely $3b + r$ variables and $2b + r$ equations (r from ELiF-G's, $2b$ from input-output) for any single encryption. Since equations are insufficient, another plaintext with the first bit flipped is also used to generate more equations (and necessarily variables). Since key and plaintext variables are the same, second encryption results in r variables (for ELiF-G outputs) and $r + b$ equations (b equations for ciphertext). Hence there are $3b + r$ variables and equations.

We used `cryptominisat` tool for SAGE to solve the equations. We collected statistics for different b and r values for a number of random key and known plaintext values. We tested $b \in \{4, 5, \ldots, 14\}$, and used two plain-cipher pairs for each test. Tests were applied on a single core of standard PC (2.7GHz CPU) running CentOS 7 and SAGE 6.9.

In our experiments, we saw that the time complexity depends linearly on the round number r, and exponentially on the block size b. We also observed there were a few solutions for each test. In Fig. 6, we showed base 2 logarithm of average time to generate the solutions for b from 4 up to 14.

Fig. 6 shows an exponential dependence on block length for the time to compute the solution. For large b, solution time is expected to be close to 2^{2b} seconds in our computation environment. These numbers may change in a different hardware-software setting, but we believe that exponential nature will be

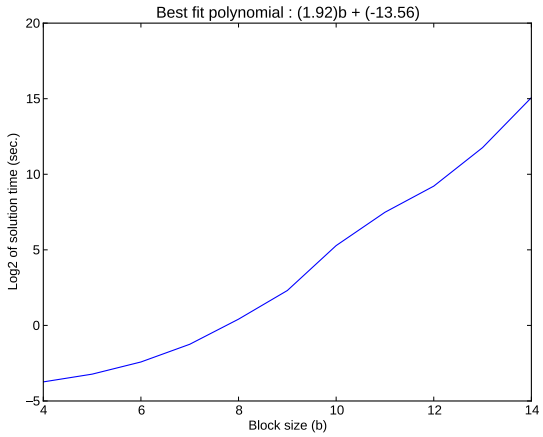


Fig. 6. $\log_2$ graph of average time to compute solutions using cryptominisat for various b values. Best-fit polynomial for the experimental data is calculated by the Python [4] numpy.polyfit function.

preserved. Hence we believe that ELiF family of ciphers are as strong as other 'classical' ciphers.

5 Implementations of ELiF Family

In this section, we examine several implementation methods of ELiF in hardware and software, and give performance estimations.

5.1 Hardware implementations

In hardware, there are two generic methods to implement an iterated block cipher : serial and parallel implementations. In a serial implementation, area is reduced for repeated applications of some primitives by using multiple clock cycles. In a parallel implementation, on the other hand, speed is increased by utilizing more area.

ELiF family of block ciphers can be implemented in a serialized or parallel manner. In the extreme serial case where only a single ELiF-G can be utilized, the cipher will have one of the least area among b-bit block ciphers except for the control logic. For an acceptable latency (i.e. minimum clock frequency) of n ELiF-G's, the speed can be increased by a factor of $n \times (b-2)$.

Moreover, parallel implementations of ELiF round function allows the use of fixed key such as in the ciphers PRINTcipher [16] and KTANTAN [11]. The main advantage in such a scenario is the reduction of area by using NAND gates when the key bit is 1, AND gates otherwise. Such a scenario is beneficial for reducing

the area and power/energy consumption of RFID tag like devices. In a similar way, round constant can also be fixed, but this requires the implementation of at least 2-latency parallel round function since $(b-2)$-bit key is insufficient to provide security in $2b^2$ rounds.

To calculate estimated hardware area of the ELiF round functions for the aforementioned scenarios, we need to calculate the area of a single ELiF-G for fixed and varying key and constant cases. Table 4 gives GE values for some logic gates and ELiF-G variants.

AND	NAND	XOR2	XOR3	Flip-Flop	ELiF-G	ELiF-G fixed key	ELiF-G Fixed key&cnst
1.25GE	1.00GE	2.25GE	4.00GE	6.00GE	7.5GE	5.125GE	3.375GE

Table 4. GE values for logic gates. Last two colums shows the average area over two possible fixed value.

Using Table 4, we estimated the state bits (including $(b-2)$-bit key and $\log_2(b)$-bit constant LFSR if any of them is not fixed) and round function area, standard gate (AND, NAND, XOR) latency and total clock cycles for $2b(b-2) = 2b^2 - 4b$ rounds (selected as a multiple of $(b-2)$ to compare with fixed key implementations). Fixed key and constant version of ELiF-G have two standard gate latency, whereas other variants have three.

We gave estimation results in Table 5. For fixed key implementations, we assumed that half of the key bits are 0. If an LFSR is implemented we added one extra gate in the total area for the feedback function. We omit other logic in total area since we did not implement any of them. We also gave throughput at 100KHz clock frequency.

These estimations are based on GE values for logic gates. For a real implementation, some control logic would be necessary. In low area implementations, these costs are usually negligible (cf. [19, 9]). Table 5 shows that ELiF family of ciphers are in fact flexible to fit most of the implementation needs. Fastest implementations have reasonable area overhead. On the other hand, it is one of the least area block cipher in serial implementation for varying key scenarios. For comparison purposes, we gave results for lightweight implementations of NIST standard AES [13] and ISO standard PRESENT [9] cipher in Table 6.

The implementations in Table 6 are fully functional cores, whereas we gave estimation for ELiF family members. So it is not directly possible to compare area values, but the area numbers of ELiF gives close approximation to real implementation values because of the relatively small effect of the control logic.

5.2 Software implementations

A single b-bit block encryption in software can be implemented in word of at least b-bits using bit-wise and, xor and shift operations for each serial round. This implementation would require too many cycles. On the other hand, since

b	Fixed input	Par. Level	Latency (gates)	# of cycles	Area (RF)	GE Area (state)	GE Area (total)	T'put @100KHz
64	No	Serial	3	7936	**7.5**	792	800.5	0.8 Kbps
64	No	1-latency	3	128	465	792	1258	50 Kbps
64	No	2-latency	6	64	930	792	1723	100 Kbps
64	No	4-latency	12	32	1860	792	2653	200 Kbps
64	Key	1-latency	**2**	128	317.75	420	**738.75**	50 Kbps
64	Key	2-latency	4	64	635.5	420	1056.5	100 Kbps
64	Key	4-latency	8	32	1271	420	1692	200 Kbps
64	Key&const	2-latency	4	64	418.5	**384**	802.5	100 Kbps
64	Key&const	4-latency	8	32	837	**384**	1221	200 Kbps
64	Key&const	8-latency	16	**16**	1674	**384**	2058	400 Kbps
128	No	Serial	3	32256	**7.5**	1566	1574.5	0.4 Kbps
128	No	1-latency	3	256	945	1566	2512	50 Kbps
128	No	2-latency	6	128	1890	1566	3457	100 Kbps
128	No	4-latency	12	64	3780	1566	5347	200 Kbps
128	Key	1-latency	2	256	645.75	810	1456.75	50 Kbps
128	Key	2-latency	4	128	1291.5	810	2102.5	100 Kbps
128	Key	4-latency	8	64	2583	810	3394	200 Kbps
128	Key&const	2-latency	4	128	850.5	768	1618.5	100 Kbps
128	Key&const	4-latency	8	64	1701	768	2469	200 Kbps
128	Key&const	8-latency	16	32	3042	768	4170	400 Kbps

Table 5. Estimated ASIC performance of ELiF$_{64,7936}$ and ELiF$_{128,32256}$. The area of control logic is omitted in total area. 'Par.' stands for parallel.

	Area (GE)	Throughput @100KHz
AES [1]	3100	80 Kbps
PRESENT-80 [1]	1570	200 Kbps
PRESENT-80 [1]	1000	11.4 Kbps

Table 6. Lightweight implementations of AES and PRESENT.

ELiF family of ciphers consists of operations on bits, its bitslice implementation is trivial. By this approach, w blocks can be encrypted simultaneously using an array of size b, where w is the word length of the CPU. For a 64-bit CPU, this speeds up the encryption by a factor about 64. Using SSE instructions, this can be further improved.

6 Conclusion

In this paper, we analyzed one of the extreme cases of lightweight block ciphers. For this aim, we proposed a flexible cipher family ELiF and studied on its experimental analysis for some block lengths. We applied S-box analysis tools to small block size variants. We generalized the results for real block lengths to find the required minimum number of rounds for security. We saw that b-bit ELiF instance should have round number around $2b^2$ for no key schedule versions. We saw that if a key schedule is not used, constant schedule becomes vital for the security.

Hardware implementation estimations show that ELiF family is indeed very flexible for most of the implementation scenarios. It can be implemented in a very small sized, high speed, or in between settings. In software, ELiF family of ciphers becomes more efficient if implemented in a bitslice manner.

References

1. Ecrypt Lightweight Block Ciphers. www.ecrypt.eu.org/lightweight/index.php/Block_ciphers, Visited on November 19, 2015.
2. Lightweight Block Cipher Design. www.cosic.esat.kuleuven.be/summer_school_albena/slides/Andrey_lightweight-bc.pdf, Visited on November 19, 2015.
3. SageMath - Open-Source Mathematical Software System. http://www.sagemath.org/, Visited on November 19, 2015.
4. Welcome to Python.org. http://www.python.org, Visited on November 19, 2015.
5. Martin R. Albrecht, Benedikt Driessen, Elif Bilge Kavun, Gregor Leander, Christof Paar, and Tolga Yalçin. Block ciphers - focus on the linear layer (feat. PRIDE). In Juan A. Garay and Rosario Gennaro, editors, *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I*, volume 8616 of *Lecture Notes in Computer Science*, pages 57–76. Springer, 2014.
6. Adnan Baysal and Suhap Sahin. Roadrunner: A small and fast bitslice block cipher for low cost 8-bit processors. *IACR Cryptology ePrint Archive*, 2015:906, 2015.
7. Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. The SIMON and SPECK families of lightweight block ciphers. *IACR Cryptology ePrint Archive*, 2013:404, 2013.
8. Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. In Alfred Menezes and Scott A. Vanstone, editors, *Advances in Cryptology - CRYPTO '90, 10th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1990, Proceedings*, volume 537 of *Lecture Notes in Computer Science*, pages 2–21. Springer, 1990.

9. Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: an ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466. Springer, 2007.
10. Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalçin. PRINCE - A low-latency block cipher for pervasive computing applications - extended abstract. In 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*, volume 7658 of *Lecture Notes in Computer Science*, pages 208–225. Springer, 2012.
11. Christophe De Cannière, Orr Dunkelman, and Miroslav Knezevic. KATAN and KTANTAN - A family of small and efficient hardware-oriented block ciphers. In Christophe Clavier and Kris Gaj, editors, *Cryptographic Hardware and Embedded Systems - CHES 2009, 11th International Workshop, Lausanne, Switzerland, September 6-9, 2009, Proceedings*, volume 5747 of *Lecture Notes in Computer Science*, pages 272–288. Springer, 2009.
12. Nicolas Courtois and Josef Pieprzyk. Cryptanalysis of block ciphers with overdefined systems of equations. In Yuliang Zheng, editor, *Advances in Cryptology - ASIACRYPT 2002, 8th International Conference on the Theory and Application of Cryptology and Information Security, Queenstown, New Zealand, December 1-5, 2002, Proceedings*, volume 2501 of *Lecture Notes in Computer Science*, pages 267–287. Springer, 2002.
13. Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Information Security and Cryptography. Springer, 2002.
14. Joan Daemen and Vincent Rijmen. Probability distributions of correlation and differentials in block ciphers. *J. Mathematical Cryptology*, 1(3):221–242, 2007.
15. Philip Hawkes, Luke O'Connor, and Philip Hawkes. Asymptotic bounds on differential probabilities, 1998.
16. Lars R. Knudsen, Gregor Leander, Axel Poschmann, and Matthew J. B. Robshaw. Printcipher: A block cipher for ic-printing. In Mangard and Standaert [17], pages 16–32.
17. Stefan Mangard and François-Xavier Standaert, editors. *Cryptographic Hardware and Embedded Systems, CHES 2010, 12th International Workshop, Santa Barbara, CA, USA, August 17-20, 2010. Proceedings*, volume 6225 of *Lecture Notes in Computer Science*. Springer, 2010.
18. Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer, 1993.
19. Axel Poschmann, San Ling, and Huaxiong Wang. 256 bit standardized crypto for 650 GE - GOST revisited. In Mangard and Standaert [17], pages 219–233.
20. Wentao Zhang, Zhenzhen Bao, Dongdai Lin, Vincent Rijmen, Bohan Yang, and Ingrid Verbauwhede. RECTANGLE: a bit-slice lightweight block cipher suitable for multiple platforms. *SCIENCE CHINA Information Sciences*, 58(12):1–15, 2015.