On the Bit-Slice representations of some nonlinear bijective transformations

Oliver Coy Puente, Rene Fernández Leal and
Reynier Antonio de la Cruz Jiménez

Institute of Cryptography, Havana University, Cuba.

2023
CTCRYPT

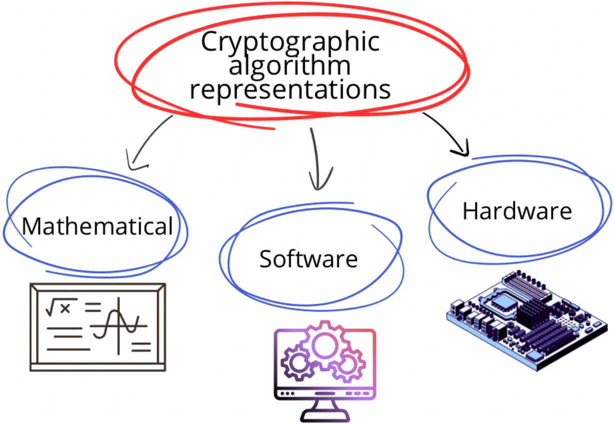
Summary

CTCRYPT 2023

Motivation

Cryptography is the field of theoretical and applied research and practical activities related to the development and application of cryptographic information protection methods.

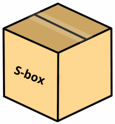
Cryptographic algorithm representations



Motivation: S-Boxes – a main cryptographic primitive

(S)ubstitution-Boxes

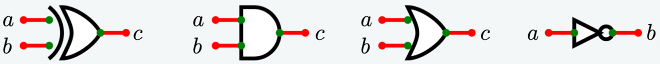
In wild of the symmetric Cryptography, S-Boxes are one of the main crypto primitives for building suitable strong cryptographic products.



Motivation: BitSlicing – a simulation of hardware in software

The basic concept of Bitslicing[1] is to simulate a hardware implementation in software.

In the Bit-Slice implementation context, S-Boxes are computed by using binary logical operations



$\mathtt{XOR} : c = a \oplus b$ $\quad$ $\mathtt{AND} : c = a \wedge b$ $\quad$ $\mathtt{OR} : c = a \vee b$ $\quad$ $\mathtt{NOT} : b = \neg a$

Figure 1: Logical Gates of operations XOR, AND, OR, NOT.

[1] Eli Biham. "A fast new DES implementation in software". In: *Fast Software Encryption: 4th International Workshop, FSE'97 Haifa, Israel, January 20–22 1997 Proceedings 4*. Springer. 1997, pp. 260–272.

CTCRYPT 2023

Almost Optimal Permutations???

CTCRYPT 2023

An 8-bit nonlinear bijective transformation without fixed points is called *almost optimal permutation* if it has:

- (algebraic) minimum degree equal to 7;
- (graph) algebraic immunity 3 and 441 equations;
- differential uniformity under 8;
- nonlinearity over 100.

Mission

Our Target

To obtain the Bit-Slice representation of some specific almost optimal permutations

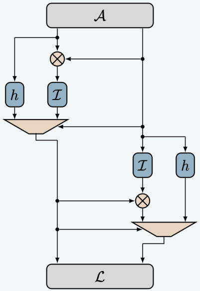


Figure 2: High level of view of $\pi'_{h,\mathcal{I}}$.

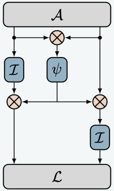


Figure 4: High level of view of $\hat{\pi}_{\psi,\mathcal{I}}$.

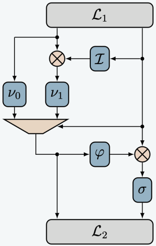


Figure 5: High level of view of $\tilde{\pi}_{\mathsf{Kuz}}$.

Preliminaries

CTCRYPT 2023

We use the notions of Bit-Slice and Gate Equivalent Complexities as implementation criteria.

Definition (Bit-Slice Gate Complexity - BGC[2])

The smallest number of operations in `XOR`, `AND`, `OR`, `NOT` required to implement an S-Box.

[2] Bao et al., "PEIGEN–a Platform for Evaluation, Implementation, and Generation of S-boxes".

Preliminaries

Definition (Gate Equivalent complexity - GEC[3])

The smallest number of Gate Equivalents (GEs) required to implement an S-Box, given the cost of atomic operations. (see, Table 1)

Techniques	NAND	XNOR	XOR	AND OR	NOT
UMC 180nm	1.00	3.00	3.00	1.33	0.67
TSMC 65nm	1.00	3.00	3.00	1.50	0.50
Software	-	-	1.00	1.00	1.00

Table 1: Cost of atomic operations under various techniques

[3] Bao et al., "PEIGEN–a Platform for Evaluation, Implementation, and Generation of S-boxes".

Some nonlinear bijective transformations

CTCRYPT 2023

Let see some 8-bit instances belonging to two classes of nonlinear bijective transformations[4][5] and we revisit the TU-decomposition of the S-Box used in the Russian cryptographic standard GOST R 34.12-2015 "Kuznyechik"[6].

---

[4] Reynier Antonio de la Cruz Jiménez. "Generation of 8-bit s-boxes having almost optimal cryptographic properties using smaller 4-bit s-boxes and finite field multiplication". In: *Progress in Cryptology–LATINCRYPT 2017: 5th International Conference on Cryptology and Information Security in Latin America, Havana, Cuba, September 20–22, 2017, Revised Selected Papers 5*. Springer. 2019, pp. 191–206.

[5] Reynier Antonio De La Cruz Jiménez. "Constructing 8-bit permutations, 8-bit involutions and 8-bit orthomorphisms with almost optimal cryptographic parameters". In: *Mathematical Aspects of Cryptography* 12.3 (2021), pp. 89–124.

[6] Alex Biryukov, Léo Perrin, and Aleksei Udovenko. "Reverse-engineering the S-box of Streebog, Kuznyechik and STRIBOBr1". In: *Advances in Cryptology–EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I 35*. Springer. 2016, pp. 372–402.

An instance of the permutation $\pi'_{h_1,h_2,\mathcal{P}_d}$

Let $\mathbb{F}_{2^4} = \mathbb{F}_2[\xi]/\xi^4 \oplus \xi \oplus 1$ and $\pi'_{h,\mathcal{I}}$ be an instance of the class of nonlinear bijective transformation $\pi'_{h_1,h_2,\mathcal{P}_d}$[7] by choosing:

- $\mathcal{A} \in \mathrm{GA}_8(\mathbb{F}_2)$ and $\mathcal{L} \in \mathrm{GL}_8(\mathbb{F}_2)$;
- The inversion function $\mathcal{I}$ over $\mathbb{F}_{2^4}$ defined by

$$\mathcal{I}(X) = \mathcal{P}_{14}(X) = X^{14}; \tag{1}$$

- A random permutation $h = h_1 = h_2 \in S(\mathbb{F}_{2^4})$.

[7] Cruz Jiménez, "Generation of 8-bit s-boxes having almost optimal cryptographic properties using smaller 4-bit s-boxes and finite field multiplication".

An instance of the permutation $\pi'_{h_1,h_2,\mathcal{P}_d}$

Cryptographic properties of $\pi'_{h,\mathcal{I}}$

- Nonlinearity — 108
- Algebraic Degree — 7
- Differential Uniformity — 6
- Graph Algebraic Immunity — 3(441)

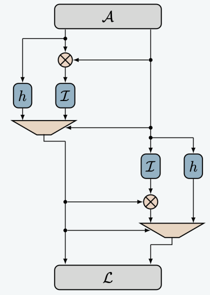


Figure 2: High level of view of $\pi'_{h,\mathcal{I}}$.

A variant of $\pi'_{h,\mathcal{I}}$

The following instance was obtained as a result of a oriented search on the structural elements used in one of the possible modification of $\pi'_{h_1,h_2,\mathcal{P}_d}$ that offer the best implementation cost (achieved in this paper) of the resulting almost optimal permutation.

The nonlinear bijective transformation $\dot{\pi}_{\lambda,\tau}$ employ the following components:

- $\mathcal{A} \in \mathrm{GA}_8(\mathbb{F}_2)$, $\mathcal{L} \in \mathrm{GL}_8(\mathbb{F}_2)$ and $\lambda \in \mathrm{GL}_4(\mathbb{F}_2)$;
- The 4-bit nonlinear bijective transformation τ defined over $\mathbb{F}_{2^4}$.

A variant of $\pi'_{h,\mathcal{I}}$

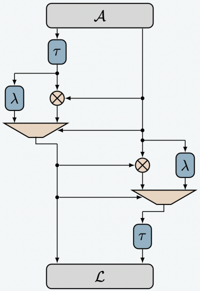


Figure 3: High level of view of $\dot{\pi}_{\lambda,\tau}$.

Cryptographic properties of $\dot{\pi}_{\lambda,\tau}$

- Nonlinearity — 108
- Algebraic Degree — 7
- Differential Uniformity — 6
- Graph Algebraic Immunity — 3(441)

An instance of the permutation $\hat{\pi}_{\psi,h}$

Let $\mathbb{F}_{2^4} = \mathbb{F}_2[\xi]/\xi^4 \oplus \xi \oplus 1$ and $\hat{\pi}_{\psi,\mathcal{I}}$ be an instance of the class of nonlinear bijective transformation $\hat{\pi}_{\psi,h}$ by choosing:

- $\mathcal{A} \in \mathrm{GA}_8(\mathbb{F}_2)$ and $\mathcal{L} \in \mathrm{GL}_8(\mathbb{F}_2)$;
- The inversion function $\mathcal{I}$ over $\mathbb{F}_{2^4}$ defined by (1);
- A non-bijective 4-bit function ψ, which have not preimage for 0.

Cryptographic properties of $\hat{\pi}_{\psi,\mathcal{I}}$

- Nonlinearity — 104
- Algebraic Degree — 7
- Differential Uniformity — 6
- Graph Algebraic Immunity — 3(441)

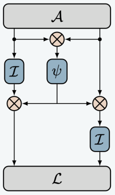


Figure 4: High level of view of $\hat{\pi}_{\psi,\mathcal{I}}$.

The S-Box of the block cipher Kuznyechik

For the Kuznyechik S-Box is suggested[8] it TU-decomposition. All 4-bit operations/transformations are described over $\mathbb{F}_{2^4} = \mathbb{F}_2[\xi]/\xi^4 \oplus \xi^3 \oplus 1$. The S-Box $\tilde{\pi}_{\mathrm{Kuz}}$ employ:

- $\mathcal{L}_i \in \mathrm{GL}_8(\mathbb{F}_2), i = 1, 2$;
- The inversion function $\mathcal{I}$ defined by (1);
- The 4-bit nonlinear transformations ν_0, ν_1, φ and σ.

[8] Alex Biryukov, Léo Perrin, and Aleksei Udovenko. "Reverse-engineering the S-box of Streebog, Kuznyechik and STRIBOBr1". In: *Advances in Cryptology–EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I 35*. Springer. 2016, pp. 372–402.

The S-Box of the block cipher Kuznyechik

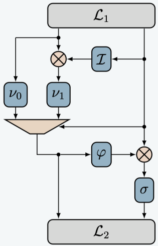


Figure 5: High level of view of $\tilde{\pi}_{\mathsf{Kuz}}$.

Cryptographic properties of $\tilde{\pi}_{\mathsf{Kuz}}$

- Nonlinearity — 100
- Algebraic Degree — 7
- Differential Uniformity — 8
- Graph Algebraic Immunity — 3(441)

Bit-Slice representations of $\pi'_{h,\mathcal{I}}$, $\dot{\pi}_{\lambda,\tau}$, $\hat{\pi}_{\psi,\mathcal{I}}$ and $\tilde{\pi}_{\mathrm{Kuz}}$

The current section is devoted to the problem of finding low gate count logic circuit representations for the studied S-Boxes, combining analytical methods with the open source tool `sboxgates`[9].

[9] Marcus Dansarie. "sboxgates: A program for finding low gate count implementations of S-boxes". In: *Journal of Open Source Software* 6.62 (2021), p. 2946

Bit-Slice representations of $\pi'_{h,\mathcal{I}}$

From the definition of $\pi'_{h,\mathcal{I}}$ is evident that

$$\mathtt{BGC}\left(\pi'_{h,\mathcal{I}}\right) = \mathtt{BGC}\left(\mathcal{A}\right) + 2 \cdot \mathtt{BGC}\left(\otimes\right) + \mathtt{BGC}\left(\mathcal{F}_1\right) + \\ + \mathtt{BGC}\left(\mathcal{F}_2\right) + 2 \cdot \mathtt{BGC}\left(\mathcal{I}\right) + 2 \cdot \mathtt{BGC}\left(h\right) + \mathtt{BGC}\left(\mathcal{L}\right), \tag{2}$$

where by $\mathcal{F}_i$, $i \in \{1, 2\}$, are denoted the left and right branches containing the conditionals *if*.

Using analytical methods was obtained that $\mathtt{BGC}\left(\mathcal{A}\right) = 1$, $\mathtt{BGC}\left(\mathcal{L}\right) = 0$, $\mathtt{BGC}\left(\otimes\right) = 31$ and $\mathtt{BGC}\left(\mathcal{F}_1\right) = \mathtt{BGC}\left(\mathcal{F}_2\right) = 12$.

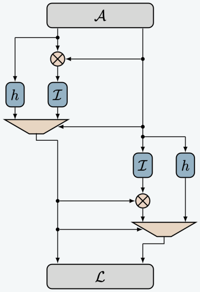


Figure 2: High level of view of $\pi'_{h,\mathcal{I}}$.

With the help of the open source tool `sboxgates` was calculated that $\mathtt{BGC}\left(h\right) = 19$ and $\mathtt{BGC}\left(\mathcal{I}\right) = 17$.

Finally, from (2) was obtained that $\mathtt{BGC}\left(\pi'_{h,\mathcal{I}}\right) = 159$.

Combinatorial circuit of $\pi'_{h,\mathcal{I}}$

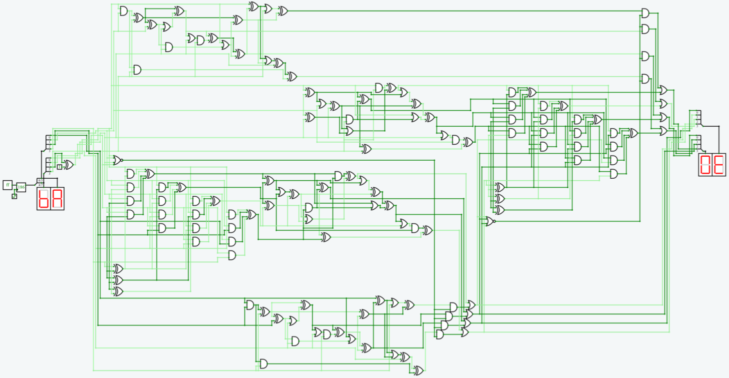


Figure 9: Combinatorial circuit of the S-Box $\pi'_{h,\mathcal{I}}$, where for the input value BA, the corresponding output value is 0E.

Bit-Slice representations of $\dot{\pi}_{\lambda,\tau}$

From the definition of $\dot{\pi}_{\lambda,\tau}$ is evident that

$$\mathtt{BGC}(\dot{\pi}_{\lambda,\tau}) = \mathtt{BGC}(\mathcal{A}) + 2\cdot\mathtt{BGC}(\otimes) + \mathtt{BGC}(\mathcal{F}_1) + \\ + \mathtt{BGC}(\mathcal{F}_2) + 2\cdot\mathtt{BGC}(\tau) + 2\cdot\mathtt{BGC}(\lambda) + \mathtt{BGC}(\mathcal{L})\,. \tag{3}$$

Using analytical methods was obtained that $\mathtt{BGC}(\mathcal{A}) = 1$, $\mathtt{BGC}(\mathcal{L}) = \mathtt{BGC}(\lambda) = 0$, $\mathtt{BGC}(\otimes) = 31$ and $\mathtt{BGC}(\mathcal{F}_1) = \mathtt{BGC}(\mathcal{F}_2) = 12$.

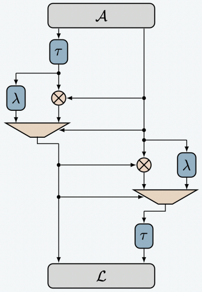


Figure 3: High level of view of $\dot{\pi}_{\lambda,\tau}$.

With the help of the open source tool `sboxgates` was calculated that $\mathtt{BGC}(\tau) = 16$.

Finally, from (3) was obtained that $\mathtt{BGC}(\dot{\pi}_{\lambda,\tau}) = 119$.

Combinatorial circuit of $\dot{\pi}_{\lambda,\tau}$

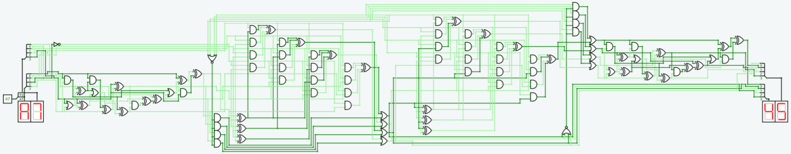


Figure 10: Combinatorial circuit of the S-Box $\dot{\pi}_{\lambda,\tau}$, where for the input value A7, the corresponding output value is 45.

Bit-Slice representations of $\hat{\pi}_{\psi,\mathcal{I}}$

From the definition of $\hat{\pi}_{\psi,\mathcal{I}}$ is evident that

$$\begin{aligned}\mathrm{BGC}\left(\hat{\pi}_{\psi,\mathcal{I}}\right) = \mathrm{BGC}\left(\mathcal{A}\right) + 3 \cdot \mathrm{BGC}\left(\otimes\right) + \\ +2 \cdot \mathrm{BGC}\left(\mathcal{I}\right) + \mathrm{BGC}\left(\psi\right) + \mathrm{BGC}\left(\mathcal{L}\right).\end{aligned} \tag{4}$$

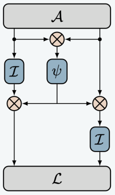


Figure 4: High level of view of $\hat{\pi}_{\psi,\mathcal{I}}$.

Using analytical methods was obtained that $\mathrm{BGC}\left(\mathcal{A}\right) = 1$, $\mathrm{BGC}\left(\mathcal{L}\right) = 0$ and $\mathrm{BGC}\left(\otimes\right) = 31$.

With the help of the open source tool `sboxgates` was calculated that $\mathrm{BGC}\left(\psi\right) = 21$ and $\mathrm{BGC}\left(\mathcal{I}\right) = 17$.

Finally, from (4) was obtained that $\mathrm{BGC}\left(\hat{\pi}_{\psi,\mathcal{I}}\right) = 149$.

Combinatorial circuit of $\hat{\pi}_{\psi,\mathcal{I}}$

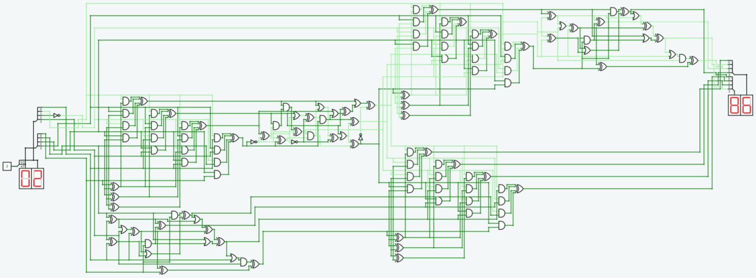


Figure 12: Combinatorial circuit of the S-Box $\hat{\pi}_{\psi,\mathcal{I}}$, where for the input value 02, the corresponding output value is 86.

A more compact Bit-Slice representation of the Kuznyechik S-Box

Motivation

1. In 2016 was proposed a method[10], requiring 681 Boolean operations to find a Bit-Slice representation of the Kuznyechik S-Box,
2. Considering the TU-decomposition previously described in the Figure 5, in 2021 was proposed a new method[11], which requires 226 Boolean (logical) operations; i.e., 455 logical operations less than the previously known method.

[10] Borisenko N.P., “Method of forming S-blocks with minimum number of logic elements”

[11] Avraamova et al., “A compact bit-sliced representation of Kuznyechik S-box”

A more compact Bit-Slice representation of the Kuznyechik S-Box

From the definition of $\tilde{\pi}_{\mathsf{Kuz}}$ is evident that

$$\begin{gathered}\mathtt{BGC}\left(\tilde{\pi}_{\mathsf{Kuz}}\right) = \mathtt{BGC}\left(\mathcal{L}_1\right) + 2\cdot\mathtt{BGC}\left(\otimes\right) + \\ +\mathtt{BGC}\left(\mathcal{I}\right) + \mathtt{BGC}\left(\nu_0\right) + \mathtt{BGC}\left(\nu_1\right) + \mathtt{BGC}\left(\mathcal{F}\right) + \\ +\mathtt{BGC}\left(\varphi\right) + \mathtt{BGC}\left(\sigma\right) + \mathtt{BGC}\left(\mathcal{L}_2\right).\end{gathered} \tag{5}$$

Using analytical methods was obtained that $\mathtt{BGC}\left(\mathcal{L}_1\right) = 9$, $\mathtt{BGC}\left(\mathcal{L}_2\right) = 5$, $\mathtt{BGC}\left(\otimes\right) = 31$ and $\mathtt{BGC}\left(\mathcal{F}\right) = 15$.

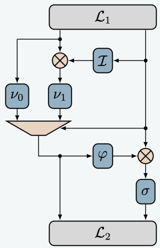


Figure 5: High level of view of $\tilde{\pi}_{\mathsf{Kuz}}$.

With the help of the open source tool `sboxgates` was calculated that $\mathtt{BGC}\left(\mathcal{I}\right) = 20$, $\mathtt{BGC}\left(\nu_0\right) = 19$, $\mathtt{BGC}\left(\nu_1\right) = 12$, $\mathtt{BGC}\left(\varphi\right) = 18$ and $\mathtt{BGC}\left(\sigma\right) = 19$.

Finally, from (5) was obtained that $\mathtt{BGC}\left(\tilde{\pi}_{\mathsf{Kuz}}\right) = 179$.

Combinatorial circuit of the Kuznyechik S-Box

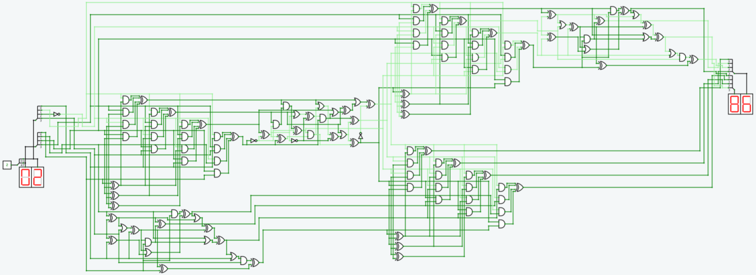


Figure 12: Combinatorial circuit of the S-Box $\hat{\pi}_{\psi,\mathcal{I}}$, where for the input value 02, the corresponding output value is 86.

Comparing robustness and implementation cost of some S-Boxes

S-Boxes	Logical Operations				BGC $(\cdot)$	GEC $(\cdot)$		Basic Cryptographic Parameters			
	XOR	AND	OR	NOT		UMC/180nm	TSMC/65nm	NL	AD	DU	AI
π_{Whp}	58	15	21	7	101	226.57	231.50	100	6	8	3(441)
π_{AES}	83	32	0	0	115	291.56	297.00	112	7	4	2(39)
$\dot{\pi}_{\lambda,\tau}$	49	48	20	2	119	238.78	250.00	108	7	6	3(441)
$\hat{\pi}_{\psi,\mathcal{I}}$	73	58	15	3	149	318.10	330.00	104	7	6	3(441)
$\pi'_{h,\mathcal{I}}$	69	54	34	2	159	325.38	340.00	108	7	6	3(441)
$\tilde{\pi}_{\text{Kuz}}$	94	54	26	5	179	391.75	404.50	100	7	8	3(441)
π_{Kal}	NR				361	NR		104	7	8	3(441)

Table 10: A comparison between Bit-Slice Gate/Gate Equivalent Complexities and the cryptographic parameters of some 8-bit S-Boxes (NR means "not reported"). The basic cryptographic parameters: nonlinearity, (algebraic) minimum degree, differential uniformity and (graph) algebraic immunity are denoted by NL, AD, DU and AI.

On the Bit-Slice representations of some nonlinear bijective transformations

Oliver Coy Puente, Rene Fernández Leal and
Reynier Antonio de la Cruz Jiménez

Institute of Cryptography, Havana University, Cuba.

CTCRYPT 2023