On the Classification of 4 Bit S-boxes

G. Leander[1][*] and A. Poschmann[2]

[1] GRIM, University Toulon, France, `Gregor.Leander@rub.de`
[2] Horst-Görtz-Institute for IT-Security, Ruhr-University Bochum, Germany,
`poschmann@crypto.rub.de`

Abstract. In this paper we classify all optimal 4 bit S-boxes. Remarkably, up to affine equivalence, there are only 16 different optimal S-boxes. This observation can be used to efficiently generate optimal S-boxes fulfilling additional criteria. One result is that an S-box which is optimal against differential and linear attacks is always optimal with respect to algebraic attacks as well. We also classify all optimal S-boxes up to the so called CCZ equivalence. We furthermore generated all S-boxes fulfilling the conditions on nonlinearity and uniformity for S-boxes used in the block cipher Serpent. Up to a slightly modified notion of equivalence, there are only 14 different S-boxes. Due to this small number it is not surprising that some of the S-boxes of the Serpent cipher are linear equivalent. Another advantage of our characterization is that it eases the highly non-trivial task of choosing good S-boxes for hardware dedicated ciphers a lot.

Keywords. S-box, Vectorial Boolean function, Affine equivalence, Hardware Implementation.

1 Introduction

S-boxes play a fundamental role for the security of nearly all modern block ciphers. In the two major design strategies for block ciphers, Feistel networks and Substitution/Permutation networks, the S-boxes form the only non-linear part of a block cipher. Therefore, S-boxes have to be chosen carefully to make the cipher resistant against all kinds of attacks. In particular there are well studied criteria that a good S-box has to fulfill to make the cipher resistant against differential and linear cryptanalyses. There are mainly two ways of generating good S-boxes: (1) picking a random large S-box or (2) generating small S-boxes with good linear and differential properties. The main drawback of picking large S-boxes is, that these S-boxes are much more inefficient to implement, especially in hardware.

Many modern block ciphers use 4 or 8 bit S-boxes. In the AES, for example, an 8 bit S-box is used that provides very good resistance against linear and differential attacks. However, regarding the design of S-boxes there are still some fundamental questions unsolved. For example, it is not known if the AES S-box is really the optimal choice, it might be true that there exist S-boxes with

---

better resistance against linear and differential attacks. Hence, the AES S-box is a world record S-box, but it is still unclear if this is an optimal result. The problem to find optimal S-boxes is very hard due to the fact that the number of permutations mapping n-bits to n-bits is huge even for very small values of n. Therefore exhaustively checking all permutations to find good S-boxes is no option.

In this paper we focus on 4 bit S-boxes, as used for example in Serpent. One advantage in dimension 4 is that the optimal values for S-boxes with respect to linear and differential cryptanalyses are known. However, the number of 4-bit permutations is still huge: roughly 2^{44}. Furthermore a naive classification of good 4 bit S-boxes is still difficult. However, it is well known that the resistance of S-boxes against most attacks remains unchanged when an invertible affine transformation is applied before and after the S-box. This fairly standard technique allows us to easily classify all optimal 4 bit S-boxes. Surprisingly, up to equivalence, there are only 16 optimal S-boxes and we list them in this paper. This massive reduction enables us to exhaustively check all optimal S-boxes with respect to other criteria, such as algebraic degree or resistance against algebraic attacks and we list some of the results. Most notably an optimal S-box with respect to linear and differential properties is always optimal with respect to algebraic attacks. Furthermore we classify these optimal S-boxes also with respect to the more general CCZ equivalence.

Moreover, this classification simplifies the task to generate optimal S-boxes that:

- are uniformly distributed among all optimal S-boxes,
- are not linear equivalent,
- fulfill additional criteria.

In the second part of this paper we focus on Serpent-type S-boxes. The block cipher Serpent uses 8 S-boxes that were chosen to fulfill additional criteria that are, in general, not invariant under affine transformations. Still it is possible to develop a slightly modified notion of equivalence and again classify these S-boxes. It turns out that, up to equivalence, there are only 14 S-boxes fulfilling the Serpent criteria. Again, using this classification one can easily derive additional properties for these kind of S-boxes. For example we demonstrate that it is not possible to choose a Serpent-type S-box such that all component functions have maximal algebraic degree.

This reduction can also be used for the design of hardware optimized block ciphers. The highly non-trivial task of minimizing the area requirements of the circuit of an S-box in hardware is eased a lot, because only a very small set of S-boxes has to be synthesized.

In [2] it is observed that several of the Serpent-type S-boxes are linear equivalent, although they have been generated in a pseudo random way. Our classification shows that this can be explained as a consequence of the small number of equivalence classes. Instead of 8 different S-boxes our considerations show that Serpent uses only 4 (really) different S-boxes. In other words, it is possible to specify the Serpent cipher, by modifying the linear layer, in such a way that

only 4 S-boxes are used. We want to point out, that to the best of our knowledge this does not pose a threat to the security of Serpent. However, due to the small number of linear inequivalent S-boxes, even randomly generated optimal S-boxes might have unexpected and unwished relations. We feel that, when designing a block cipher, one should be aware of this fact and take it into account.

We want to point out an important difference of this work and algorithms that consider the linear equivalence of two given functions (see for example [2]): the task of classifying all functions is not trivial, because the number of functions for which this algorithm has to be applied to might still be too huge.

It should be noted that the results and techniques given here clearly do not work for larger S-boxes. Even for dimension six, a complete classification of all good S-boxes seems elusive. To illustrate the huge amount of computation needed for this classification note that the number of linear inequivalent S-boxes in dimension five is already larger than 2^{61}, see [9].

2   Notation

In this section we introduce the notation used throughout the paper. For two vectors $a, b \in \mathbb{F}_2^n$, we denote by

$$\langle a, b \rangle = \sum_{i=0}^{n-1} a_i b_i$$

the inner product of a and b. The binary weight of a vector a is denoted by $\text{wt}(a)$. For a Boolean function in n variables

$$f : \mathbb{F}_2^n \to \mathbb{F}_2$$

and an element $a \in \mathbb{F}_2^n$ we denote the *Walsh Coefficient* of f at a by

$$f^{\mathcal{W}}(a) = \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x) + \langle a, x \rangle}. \tag{1}$$

The linearity of f is defined as

$$\text{Lin}(f) = \max_{a \in \mathbb{F}_2^n} |f^{\mathcal{W}}(a)|$$

The value $\text{Lin}(f)$ is large if and only if f is close to a linear or affine function, i.e. there exists a linear or affine function which is a good approximation for f. The maximal possible value for $\text{Lin}(f)$ is 2^n and is attained iff f is linear or affine. Moreover, due to Parsevals Equality

$$\sum_{a \in \mathbb{F}_2^n} \left(f^{\mathcal{W}}(a) \right)^2 = 2^{2n}$$

we see that $\text{Lin}(f) \geq 2^{n/2}$. Functions attaining this lower bound are called *bent functions*. Bent functions were introduced by Rothaus [13] and exist if and only if n is even.

This paper deals mainly with S-boxes, i.e. functions with values that are bit strings. Given an S-box mapping n bits to m bits

$$S : \mathbb{F}_2^n \to \mathbb{F}_2^m$$

we denote for any vector $b \in \mathbb{F}_2^m$ the corresponding *component function S*.

$$S_b : \mathbb{F}_2^n \to \mathbb{F}_2$$
$$x \mapsto \langle b, S(x) \rangle$$

The function S_b is the Boolean function derived from S by considering a fixed sum of the output bits determined by $b \in \mathbb{F}_2^m$. In particular, if b is the i-th vector in the canonical base, S_b corresponds to the i-th bit of S. We define the linearity of S as

$$\mathrm{Lin}(S) = \max_{a \in \mathbb{F}_2^n, b \in \mathbb{F}_2^m \setminus \{0\}} |S_b^{\mathcal{W}}(a)|$$

This linearity represents a measure for the resistance against linear cryptanalysis. For even dimension n the smallest known linearity for a permutation is $2^{n/2+1}$. It is a longstanding open problem to find S-boxes with smaller linearity, or to prove that such functions do not exist.

In linear cryptanalysis, introduced by Matsui in [10], one is interested in approximating S with a linear function.

The probability of a linear approximation of a combination of output bits S_b (determined by b) by a linear combination of input bits x (determined by a) can be written as

$$p = \frac{\#\{x | S_b(x) = \langle a, x \rangle\}}{2^n}. \qquad (2)$$

Combining equations (1) and (2) leads to

$$p = \frac{1}{2} - \frac{S_b^{\mathcal{W}}(a)}{2^{n+1}}.$$

The *linear probability bias* ε is a correlation measure for this deviation from probability $\frac{1}{2}$ for which it is entirely uncorrelated. We have

$$\varepsilon = \left| p - \frac{1}{2} \right| = \left| \frac{S_b^{\mathcal{W}}(a)}{2^{n+1}} \right|$$

and

$$\varepsilon \leq \left| \frac{\mathrm{Lin}(S)}{2^{n+1}} \right|.$$

Therefore, the smaller the linearity $\mathrm{Lin}(S)$ of a S-box is, the more secure the S-box is against linear cryptanalysis.

The idea of differential cryptanalysis (DC), invented by Biham and Shamir (see [1]), in a nutshell, is to trace how the difference of two encrypted messages m and $m+\delta$ propagates through the different rounds in a block cipher. Basically, if an attacker can guess the output differences with high probability, the cipher

will be vulnerable to a differential attack. Thus, a designer of a block cipher has to ensure that, given any nonzero input difference, no fixed output difference occurs with high probability. Since in nearly all block ciphers S-boxes represent the only nonlinear parts, it is particularly important to study the differential properties of these building blocks. To measure the resistance against differential cryptanalysis we define for $a \in \mathbb{F}_2^n$

$$\Delta_{S,a} : \mathbb{F}_2^n \to \mathbb{F}_2^m$$
$$x \mapsto S(x) + S(x + a)$$

and

$$\mathrm{Diff}(S) = \max_{a \neq 0, b \in \mathbb{F}_2^n} |\Delta_{S,a}^{-1}(b)|.$$

Clearly, the value $\mathrm{Diff}(S)$ is related to the maximal probability that any fixed nonzero input difference causes any fixed output difference after applying the S-box. Given an input difference a the value $|\Delta_{S,a}^{-1}(b)|$ is the number of message pairs $(x, x + a)$ with the output difference b.

Clearly it holds for any S-box that $\mathrm{Diff}(S) \geq 2$. Functions attaining this lower bound are called APN functions. However it is unknown if APN permutations exist in even dimension.

3   Optimal 4 bit S-boxes

As explained in the introduction, a natural requirement for 4 bit S-boxes is an optimal resistance against linear and differential cryptanalyses. Unlike for higher dimensions the optimal values for $\mathrm{Lin}(S)$ and $\mathrm{Diff}(S)$ are known for dimension $n = 4$. More precisely, for any bijective mapping $S : \mathbb{F}_2^4 \to \mathbb{F}_2^4$ we have $\mathrm{Lin}(S) \geq 8$ and $\mathrm{Diff}(S) \geq 4$. To see that $\mathrm{Lin}(S) \geq 8$ note that, if S is a bijection then all its component functions S_b have even weight and therefore all Walsh coefficients are divisible by 4. Furthermore we must have $\mathrm{Lin}(S) > 4$ since there are no vectorial bent functions from $\mathbb{F}_2^n \to \mathbb{F}_2^n$ for any n as proven by Nyberg in [11]. Knudsen [7] showed that there is no APN Permutation on $\mathbb{F}_2^4$, i.e. no S-boxes with $\mathrm{Diff}(S) = 2$. Therefore, as $\mathrm{Diff}(S)$ is always even, we must have $\mathrm{Diff}(S) \geq 4$. With respect to these observations we call S-boxes attaining these minima *optimal*.

Definition 1. *Let* $S : \mathbb{F}_2^4 \to \mathbb{F}_2^4$ *be an S-box. If S fulfills the following conditions we call S an* optimal S-box

1. *S is a bijection.*
2. *$\mathrm{Lin}(S) = 8$.*
3. *$\mathrm{Diff}(S) = 4$.*

An example for an optimal S-box is the inverse function, where one identifies the vector space $\mathbb{F}_2^n$ with $\mathbb{F}_{2^n}$, the finite field with 2^n elements, and considers the mapping

$$I : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$$
$$I(x) = x^{2^n - 2}$$

This mapping fulfils $\mathrm{Lin}(S) = 2^{n/2+1}$ when n is even, as was proven for example in [8]. The proof that $\mathrm{Diff}(S) = 4$ when n is even and $\mathrm{Diff}(S) = 2$ when n is odd is trivial. This type of mapping is also used in the Advanced Encryption Standard (AES). However, in AES we have $n = 8$ and it is not clear that this S-box is optimal in this dimension. It can only be viewed as the world record S-box in a sense that no bijection is known with better resistance against linear and differential cryptanalyses.

When designing a block cipher it is important to know the set of S-boxes to choose from in order to get an optimal resistance against known attacks. Since the number of all permutations on $\mathbb{F}_2^n$ is $2^n!$ and thus huge even for small dimensions, it is crucial to be able to reduce the number of S-boxes which have to be considered. A well known and well suited tool is the notion of linear equivalence.

3.1 Linear Equivalence

It is well known (see for example [4] and [12]) that the values of $\mathrm{Diff}(S)$ and $\mathrm{Lin}(S)$ remain unchanged if we apply affine transformations in the domain or co-domain of S. In particular if we take an optimal S-box in the above sense and transform it in an affine way, we get another optimal S-box. Using such a transformed S-box can also be viewed as changing the linear layer of a block cipher.

This is formalized in the following theorem.

Theorem 1. *Let $A, B \in \mathrm{GL}(4, \mathbb{F}_2)$ be two invertible 4×4 matrices and $a, b \in \mathbb{F}_2^4$. Let $S : \mathbb{F}_2^4 \to \mathbb{F}_2^4$ be an optimal S-box. Then the S-box S' with*

$$S'(x) = B(S(A(x) + a)) + b$$

is an optimal S-box as well.

This observation can be used to define an equivalence relation on the set of all optimal S-boxes. We call two S-boxes S_1, S_2 equivalent if there exist bijective linear mappings A, B and constants $a, b \in \mathbb{F}_2^4$ such that

$$S'(x) = B(S(A(x) + a)) + b.$$

If two S-boxes S_1 and S_2 are equivalent in the above sense we denote this by $S_1 \sim S_2$. A natural question that arises is in how many equivalence classes the set of all optimal S-boxes is split. As we have already pointed out this reduction to equivalence classes is also important for the practical design of block ciphers, because it simplifies the choice of good S-boxes. We computed the number of equivalence classes using the observations presented in Section 6 and it turns out that this number is very small. There are only 16 different, i.e. non-equivalent, classes for each of them we list a representative in Table 6 and their polynomial representation in Table 7. Each row in Table 6 contains one representative, where we identify vectors $\mathbb{F}_2^n$ with integers in $\{0, \ldots, 15\}$. We list the images of the values in integer ordering, i.e. the first integer represents the image of 0, the second the image of 1, and so on. We summarize this result in the following fact.

Fact 1 *There exist exactly* 16 *non equivalent optimal S-boxes. Any optimal S-box is equivalent to exactly one S-box given in Table 6.*

Note that G_3 is equivalent to the invers mapping $x \mapsto x^{-1}$.

This massive reduction allows us to exhaustively check all optimal S-boxes up to equivalence with respect to other criteria as explained in the next section.

3.2 Other Criteria

With this reduction to 16 equivalence classes it is now easy to study additional criteria, such as algebraic immunity, improved resistance against linear and differential cryptanalyses, which are again invariant under this equivalence.

Algebraic Degree Another important criterion for an S-box is to have high algebraic degree. It is well known that every Boolean function $f : \mathbb{F}_2^n \to \mathbb{F}_2$ can be uniquely represented by a multivariate polynomial of degree at most 4, i.e. there exist coefficients $\alpha_u \in \mathbb{F}_2$ such that

$$f(x) = f(x_1, \ldots, x_n) = \sum_{u \in \mathbb{F}_2^n} \alpha_u x_1^{u_1} \cdots x_n^{u_n}$$

The (algebraic) degree of f is defined to be the maximal weight of u such that $\alpha_u \neq 0$. For an S-box we define the algebraic degree as

$$\deg(S) = \max_{b \in \mathbb{F}_2^n, b \neq 0} \deg(S_b)$$

Clearly the degree is invariant under linear equivalence. Moreover it is easy to see that also the multiset

$$\{\deg(S_b) \quad | \quad b \in \mathbb{F}_2^n\}$$

is invariant under linear equivalence. It is known that any bijection must have degree smaller than 3. We computed the degree of all 16 representatives and we list the results in Table 1.

$S-box$	G_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}	G_{13}	G_{14}	G_{15}
$\deg(S_b) = 2$	3	3	3	0	0	0	0	0	3	1	1	0	0	0	1	1
$\deg(S_b) = 3$	12	12	12	15	15	15	15	15	12	14	14	15	15	15	14	14

Table 1. Number of $b \in \mathbb{F}_2^4 \setminus \{0\}$ such that $\deg(S_b) = 2, 3$

It should be noted that a frequently used criterion for good S-boxes is to have high algebraic degree. From this perspective the following fact is of interest

Fact 2 *There exist* 8 *non linear equivalent optimal S-boxes* $S : \mathbb{F}_2^n \to \mathbb{F}_2^n$ *such that* $\deg(S_b) = 3$ *for all* $b \in \mathbb{F}_2^n, b \neq 0$.

Again, one example of such an S-box is the "inverse" S-box, which indeed is equivalent to G_3.

Linearity Besides minimizing the linearity $\mathrm{Lin}(S)$ it might also be important to study which Walsh coefficients occur and how often they occur. In particular a reasonable design goal would be to minimize the number of Walsh coefficients with the highest maximal value, as this minimizes the number of linear approximation with maximal probability. We computed the Walsh spectra for all of the 16 S-boxes. In Table 9 we list for each S-box the number of times a certain Walsh coefficient is attained.

Algebraic Relations Algebraic attacks, invented by Courtois and Pieprzyk (see [5]), are another type of attack which have recently attracted a lot of attention. It still seems to be unclear which conditions exactly enable this attack. However, the main criterion to successfully mount an algebraic attack is the number of linear independent low degree equations that are fulfilled by the input and output values of the S-box, i.e equations of the form $P(x, S(x)) = 0$ for all x. While for large S-boxes or for a huge number of small S-boxes computing the number of such equations needs an enormous computational effort, it is very easy to compute the number of equations for all 16 representative given above. Following [6] we computed the number of quadratic equations, i.e. all equations of the form

$$\sum_{i,j} \alpha_{ij} x_i y_i + \sum_{i \neq j} \beta_{ij} x_i x_j + \sum_{i \neq y} \gamma_{ij} y_i y_j + \sum_i \delta_i x_i + \sum_i \epsilon_i y_j + \nu$$

where $y = S(x)$. Remarkably all 16 representatives, and therefore all optimal S-boxes fulfill exactly 21 linear independent quadratic equations. As explained by Courtois in [6] this is the minimal number of quadratic equations for any mapping from $\mathbb{F}_2^4$ to $\mathbb{F}_2^4$. In this sense the optimal S-boxes are also optimal with respect to algebraic attacks. We summarize these observation in the following fact.

Fact 3 *Let $S : \mathbb{F}_2^4 \to \mathbb{F}_2^4$ be an optimal S-box. Then S fulfills 21 quadratic equations, which is the minimum number for S-boxes in dimension 4.*

4 Serpent-Type S-boxes

For the resistance of ciphers against linear and differential cryptanalyses not only the linearity of an S-box might be important, but also where this maximum occurs. Depending on the design strategy, for resistance against DC it might be important, that any one bit input difference causes an output difference of at least two bits (see for example DES or Serpent). Such a condition can be used to increase the minimal number of active S-boxes in two consecutive rounds. A similar requirement for LC is that the probability of a linear approximation using only one input and one output bit is especially low. We formalize these two conditions using the following notation

$$\mathrm{Lin}_1(S) = \max\{|S_b^{\mathcal{W}}(a)| \mid \mathrm{wt}(a) = \mathrm{wt}(b) = 1\}.$$

and

$$\mathrm{Diff}_1(S) = \max_{a \neq 0, b \in \mathbb{F}_2^n} \{|\Delta_{S,a}^{-1}(b)| \ \mathrm{wt}(a) = \mathrm{wt}(b) = 1\}.$$

The S-boxes in the Serpent cipher fulfill the following conditions

Definition 2. *Let $S : \mathbb{F}_2^4 \to \mathbb{F}_2^4$ be a S-box. If S fulfills the following conditions we call S a* Serpent-type S-box

1. *S is optimal*
2. *$\mathrm{Diff}_1(S) = 0$, i.e. any one bit input difference causes at least two bits output difference.*

We generated all S-boxes having 4-bit input and 4-bit output value fulfilling these conditions. The total number of these S-boxes is $2,211,840$ but again this can be reduced using, a slightly modified, notion of linear equivalence.

4.1 Equivalence of Serpent-type S-boxes

It is easy to see that the condition $\mathrm{Diff}_1(S) = 0$ is, in general, not invariant under the above defined equivalence relation. However, when we restrict to bit permutations in the domain and the co-domain of a mapping S, instead of allowing arbitrary linear bijections, this gives us a similar tool as before. We have the following Theorem, which is a modified version of Theorem 1.

Theorem 2. *Let $P_0, P_1 \in \mathrm{GL}(4, \mathbb{F}_2)$ be two 4×4 permutation matrices and $a, b \in \mathbb{F}_2^4$. Let $S : \mathbb{F}_2^4 \to \mathbb{F}_2^4$ be a Serpent-type S-box. Then the S-boxes S' with*

$$S'(x) = P_1(S(P_0(x) + a)) + b$$

is a Serpent-type S-box as well.

Proof. trival.

Clearly this again defines an equivalence relation of the set of S-boxes. If two S-boxes S_1, S_2 are equivalent in the above sense we denote this by

$$S_1 \sim_S S_2.$$

As this notion of equivalence is a special type of the equivalence used in Section 3 we have the implication

$$S_1 \sim_S S_2 \Rightarrow S_1 \sim S_2$$

Again, one might be interested in how many equivalence classes the set of all Serpent-type S-boxes is split. Like before, this number is surprisingly small. There are only 20 different classes and for each class we list a representative in Table 8.

Fact 4 *There exist exactly 20 non equivalent Serpent-Type S-boxes. Any Serpent-Type S-box is equivalent to exactly one S-box given in Table 8.*

4.2 Other Criteria

Again this reduction to only 20 representatives allows us to exhaustively check all Serpent-type S-boxes with respect to other criteria. Since Serpent-type S-boxes are in particular optimal S-boxes, Fact 3 immediately applies to Serpent-type S-boxes as well. On the other hand due to the computations given in Table 2, it is impossible to choose a Serpent-type S-box such that all linear combinations of coordinate functions have maximal degree.

$S - box$	R_0	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9
$\deg(S_b) = 2$	3	3	3	1	1	1	3	3	1	3
$\deg(S_b) = 3$	12	12	12	14	14	14	12	12	14	12

$S - box$	R_{10}	R_{11}	R_{12}	R_{13}	R_{14}	R_{15}	R_{16}	R_{17}	R_{18}	R_{19}
$\deg(S_b) = 2$	3	3	3	1	3	1	3	3	3	3
$\deg(S_b) = 3$	12	12	12	14	12	14	12	12	12	12

Table 2. Number of $b \in \mathbb{F}_2^4$ such that $\deg(S_b) = 2, 3$

Fact 5 *For any Serpent-type S-box S there exists an element $b \in \mathbb{F}_2^n$ such that S_b is a quadratic Boolean function.*

In particular this implies that the "inverse" Function is not linear equivalent to a Serpent-type S-box.

Linearity We computed the Walsh spectra for all of the 20 S-boxes. In Table 10 we list for each S-box the number of times a certain Walsh coefficient is attained.

As already mentioned before, not only the linearity of an S-box is important for the resistance of a cipher against linear cryptanalysis, but also where this maximum occurs. In particular it might be important, depending on the linear layer, that the Walsh coefficients $S_b^{\mathcal{W}}(a)$ with $\mathrm{wt}(a) = \mathrm{wt}(b) = 1$ are especially small. We list the values in Table 3. In particular we see that there exist no

$S - box$	R_0	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}	R_{11}	R_{12}	R_{13}	R_{14}	R_{15}	R_{16}	R_{17}	R_{18}	R_{19}
$\mathrm{Lin}_1(S)$	4	4	8	4	4	4	4	4	4	8	8	8	8	4	4	4	4	8	4	4

Table 3. Linearity of all 20 Classes of Serpent-type S-boxes

Serpent-type S-box such that $\mathrm{Lin}_1(S) = 0$.

Fact 6 *Let $S : \mathbb{F}_2^4 \to \mathbb{F}_2^4$ be a Serpent-type S-box. Then $\mathrm{Lin}_1(S) \in \{4, 8\}$.*

This fact demonstrates that the choice made for the block cipher Serpent is indeed optimal with regard to this criterion.

5 Relation between the Representatives and Inverses

The number of representatives can be further reduced when also the inverses of the S-boxes are considered. This is due to the next Theorem, which is obvious to proof.

Theorem 3. *Let $S : \mathbb{F}_2^4 \to \mathbb{F}_2^4$ be an optimal (resp. a Serpent-type) S-box then its inverse S^{-1} is an optimal (resp. a Serpent-type) S-box as well.*

We have the following relations between the representatives of the optimal S-boxes and their inverses.

$$
\begin{aligned}
G_0 &\sim G_2^{-1}, \;\; G_1 \sim G_1^{-1}, \;\; G_2 \sim G_0^{-1} \\
G_3 &\sim G_3^{-1}, \;\; G_4 \sim G_4^{-1}, \;\; G_5 \sim G_5^{-1} \\
G_6 &\sim G_6^{-1}, \;\; G_7 \sim G_7^{-1}, \;\; G_8 \sim G_8^{-1} \\
G_9 &\sim G_9^{-1}, \; G_{10} \sim G_{10}^{-1}, G_{11} \sim G_{11}^{-1} \\
G_{12} &\sim G_{12}^{-1}, G_{13} \sim G_{13}^{-1}, G_{14} \sim G_{15}^{-1} \\
G_{15} &\sim G_{14}^{-1}
\end{aligned}
$$

Remarkably, all optimal S-boxes except for G_0, G_2 and G_{14}, G_{15} are linear equivalent to their inverses.

For the Serpent-type S-boxes we have the following relations

$$
\begin{aligned}
R_0 &\sim_S R_{18}^{-1}, \;\; R_1 \sim_S R_6^{-1}, \;\; R_2 \sim_S R_{17}^{-1} \\
R_3 &\sim_S R_5^{-1}, \;\; R_4 \sim_S R_{13}^{-1}, \;\; R_5 \sim_S R_3^{-1} \\
R_6 &\sim_S R_1^{-1}, \;\; R_7 \sim_S R_{16}^{-1}, \;\; R_8 \sim_S R_{15}^{-1} \\
R_9 &\sim_S R_{10}^{-1}, R_{10} \sim_S R_9^{-1}, R_{11} \sim_S R_{12}^{-1} \\
R_{12} &\sim_S R_{11}^{-1}, R_{13} \sim_S R_4^{-1}, R_{14} \sim_S R_{19}^{-1} \\
R_{15} &\sim_S R_8^{-1}, R_{16} \sim_S R_7^{-1}, R_{17} \sim_S R_2^{-1} \\
R_{18} &\sim_S R_0^{-1}, R_{19} \sim_S R_{14}^{-1}
\end{aligned}
$$

From these relations we see the following Fact.

Fact 7 *No Serpent-type S-box is self-equivalent in a sense that*

$$
S \sim_S S^{-1}
$$

and in particular no Serpent-type S-box is an involution.

Clearly, as mentioned before, Serpent-type S-boxes are optimal S-boxes and therefore each of the Serpent-type S-boxes R_i must be equivalent to one of the optimal S-boxes G_j. For the sake of completeness we list these relations below.

$$
\begin{aligned}
R_0 &\sim G_1, \;\; R_1 \sim G_1, \;\; R_2 \sim G_1 \\
R_3 &\sim G_{10}, \;\; R_4 \sim G_9, \; R_5 \sim G_{10} \\
R_6 &\sim G_1, \;\; R_7 \sim G_2, \; R_8 \sim G_{15} \\
R_9 &\sim G_0, \;\; R_{10} \sim G_2, R_{11} \sim G_0 \\
R_{12} &\sim G_2, \; R_{13} \sim G_9, R_{14} \sim G_2 \\
R_{15} &\sim G_{14}, R_{16} \sim G_0, R_{17} \sim G_1 \\
R_{18} &\sim G_1, \;\; R_{19} \sim G_0
\end{aligned}
$$

5.1 CCZ Equivalence

The linear equivalence defined above, is a special case of a more general equivalence (see [4]), called Carlet-Charpin-Zinoviev equivalence (CCZ for short). Two functions $F, G : \mathbb{F}_2^n \to \mathbb{F}_2^n$ are called CCZ equivalent if there exist a linear permutation on $\mathbb{F}_2^n \times \mathbb{F}_2^n$ such that the graph of F, i.e. the set $\{(x, F(x)) \mid x \in \mathbb{F}_2^n\}$ is mapped to the graph of G. Note that a permutation is always CCZ equivalent to its inverse (cf. Section 5). CCZ equivalence is in particular interesting, as many cryptographic properties of S-boxes are CCZ invariant. This includes the Walsh spectra, the uniformity and the number of algebraic relations of any degree. Using the algorithms presented in [3] we classified the optimal S-boxes up to CCZ equivalence. There are 6 non CCZ equivalent classes and we list all CCZ equivalence relations between the optimal S-boxes G_i below.

$$G_0 \sim_{\text{ccz}} G_1 \sim_{\text{ccz}} G_2 \sim_{\text{ccz}} G_8 \qquad G_3 \sim_{\text{ccz}} G_5$$
$$G_4 \sim_{\text{ccz}} G_6 \qquad\qquad G_7 \sim_{\text{ccz}} G_{11} \sim_{\text{ccz}} G_{12}$$
$$G_9 \sim_{\text{ccz}} G_{10} \qquad\qquad G_{14} \sim_{\text{ccz}} G_{15}$$

5.2 The block cipher Serpent

As an application of our observations we study the S-boxes in the Serpent cipher, as specified in the AES submission. The 8 different S-boxes in Serpent have been generated in a pseudo random manner from the set of all Serpent-type S-boxes with the additional criterion that $\text{Lin}_1(S) = 4$. We list the S-boxes used in Serpent in Table 4. Up to $\sim_S$-equivalence these S-boxes have been actually

S_0	$3, 8, 15, 1, 10, 6, 5, 11, 14, 13, 4, 2, 7, 0, 9, 12$
S_1	$15, 12, 2, 7, 9, 0, 5, 10, 1, 11, 14, 8, 6, 13, 3, 4$
S_2	$8, 6, 7, 9, 3, 12, 10, 15, 13, 1, 14, 4, 0, 11, 5, 2$
S_3	$0, 15, 11, 8, 12, 9, 6, 3, 13, 1, 2, 4, 10, 7, 5, 14$
S_4	$1, 15, 8, 3, 12, 0, 11, 6, 2, 5, 4, 10, 9, 14, 7, 13$
S_5	$15, 5, 2, 11, 4, 10, 9, 12, 0, 3, 14, 8, 13, 6, 7, 1$
S_6	$7, 2, 12, 5, 8, 4, 6, 11, 14, 9, 1, 15, 13, 3, 10, 0$
S_7	$1, 13, 15, 0, 14, 8, 2, 11, 7, 4, 12, 10, 9, 3, 5, 6$

Table 4. The S-boxes used in the cipher Serpent

chosen from a set of only 14 S-boxes, as 6 of the 20 representatives do not fulfill $\text{Lin}_1(R_i) = 4$. It is therefore no surprise that two of the S-boxes in Serpent are linear equivalent, namely we have

$$S_4 \sim_S S_5$$

Furthermore, if also the more general equivalence and inverses are considered it turns out that the following relations hold

$$S_0 \sim S_1^{-1} \sim G_2 \qquad S_2 \sim S_6 \sim G_1$$
$$S_3 \sim S_7 \sim G_9 \qquad S_4 \sim S_5 \sim G_{14}$$

Hence, even though all Serpent S-boxes have been randomly generated, Serpent uses only 4 different S-boxes with respect to linear equivalence and inverses. This can also be viewed as follows: There exists a different specification of the Serpent cipher, which uses a different linear layer, that uses only 4 S-boxes.

6   Implementation Details

In this section we explain some of the shortcuts that have been used to generate the set of representatives G_j for optimal S-boxes. Using these ideas all our computations could be done within a few minutes on a regular PC. The most important speedup was due to the following Lemma.

Lemma 1. *Let* $S : \mathbb{F}_2^n \to \mathbb{F}_2^n$ *be a bijection. Then there exist bases* B, B' *of* $\mathbb{F}_2^n$ *such that*
$$S(B) = B'$$

Proof. We give a proof by induction. For any subset $B \subset \mathbb{F}_2^n$ we denote by $\langle B \rangle$ the linear span of B. Assume that we already constructed two sets $B_i, B_i' \subset \mathbb{F}_2^n$ each consisting of $i < n$ linear independent elements such that $S(B_i) = B_i'$. We have to find an element $x \in \mathbb{F}_2^n \setminus \langle B_i \rangle$ that is mapped into the set $\mathbb{F}_2^n \setminus \langle B_i' \rangle$. There are $2^n - 2^i$ elements in $\mathbb{F}_2^n \setminus \langle B_i \rangle$ but, as S is a permutation, only $2^i - i$ possible images in $\langle B_i' \rangle$. Furthermore, for $i < n$ it holds that $2^n - 2^i > 2^i - i$. Thus, as S is a bijection, at least one element in $\mathbb{F}_2^n \setminus \langle B_i \rangle$ gets mapped to $\mathbb{F}_2^n \setminus \langle B_i' \rangle$.   □

Using this lemma, we can speedup the search for optimal S-boxes. We can restrict to optimal S-boxes fulfilling

$$S(i) = i \quad \text{for } i \in \{0, 1, 2, 4, 8\}$$

as, due to the above lemma, any optimal S-box is equivalent to such an S-box. This observation reduced the search space from $16! \approx 2^{44}$ to only $11! \approx 2^{25}$ permutations that have to be created. We generated all those permutations and tested if they fulfill $\text{Diff}(S) = 4$ and $\text{Lin}(S) = 8$. This resulted in 1396032 optimal S-boxes.

Given this set of optimal S-boxes we created a set of representatives as follows. We started by the first S-box in the set and generated all equivalent S-boxes. Whenever one of these equivalent S-boxes was present in the set, we removed this S-box from the set. Note that there are approximately 2^{15} invertible 4×4 matrices, thus running naively through all invertible matrices A, B and all constants

$c, d \in \mathbb{F}_2^4$ to generate all equivalent rows results in generating approximately 2^{38} S-boxes. However, as all S-boxes in the set are chosen such that

$$S(i) = i \quad \text{for} \ \ i \in \{0, 1, 2, 4, 8\} \tag{3}$$

then, if we fix A and d, the values for B and c are completely determined. Namely if we have

$$A(S(B(x) + c)) + d = S'(x)$$

for two S-boxes fulfilling (3) it must hold that

$$c = S^{-1}\left(A^{-1}(d)\right)$$

and

$$B(i) = S^{-1}\left(A^{-1}(i + d)\right) + c \quad \text{for } i \in 0, 1, 2, 4, 8$$

Using this observation we only had to generate approximately 2^{19} S-boxes.

7   Hardware Implementation

During our investigations we automatically synthesized thousands of S-boxes. Since hardware designers try to avoid using look-up tables, S-boxes are usually realized in combinatorial logic. Therefore, we fed the S-boxes in a combinatorial description into the synthesis tool. We compiled them in an area-efficient way, i.e. we instructed the synthesis tool to minimize the area requirements. Unfortunately logic synthesis tools like Synopsys *Design Compiler* use heuristic algorithms to map VHDL-code to standard-cells. Hence, it is never guaranteed that the resulting gate-level netlist is the smallest possible for a given VHDL-code.

To illustrate these suboptimal results we generated for all representatives R_i of Serpent-type S-boxes all equivalent S-boxes and synthesized them for the AMIS MTC45000 CMOS 0.35μm standard-cell library. The results are given in Table 5. Table 5 lists for all representatives of Serpent-type S-boxes the minimal and maximal area requirements of equivalent S-boxes.

Repr.	R_0	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9
min GE	27.4	25.0	28.0	25.3	27.3	28.0	26.4	25.3	23.7	24.0
max GE	38.7	35.3	37.3	33.3	35.3	35.3	37.3	37	33.3	33.3

Repr.	R_{10}	R_{11}	R_{12}	R_{13}	R_{14}	R_{15}	R_{16}	R_{17}	R_{18}	R_{19}
min GE	26.3	24.4	24.3	28.7	21.3	25.3	24.7	27.0	23.4	25.7
max GE	37	38.7	29.7	36.3	31	34.7	34	39	30.3	39.3

Table 5. Area Requirement for all 20 Classes of Serpent-type S-boxes

A two bit input `XOR`-gate typically needs 10 transistors or 2.5 gate equivalences (GE), respectively. If one input bit is fixed, the `XOR`-gate is either superfluent ($inputbit_a \oplus 0 = inputbit_a$) or can be replaced by an inverter ($inputbit_a \oplus 1 = \neg inputbit_a$), which costs two transistors. Bit-permutations do not need any transistors in hardware. They are realized by wiring, which means that they come for a negligible amount of additional area or even for free. Hence, any two S-boxes which differ only by a permutation of the input bits and an permutation of the output bits should be compiled to the same combinatorial "*core*". Adding a constant before and after the core can be realized with not more than 16 transistors (= 4 GE). Therefore two equivalent Serpent-type S-boxes can be implemented with a difference of at most 4 GE. Hence, S-boxes in the same class should have been compiled to the same core and their size should not differ by more than 4 GEs. However, our figures clearly show that this is unfortunately not the case. We believe that this discrepancy is caused by a suboptimal synthesis due to the heuristic mapping algorithms.

As one can see, the biggest minimal representatives of an S-box class require 28.7 GE, whereas the smallest representatives only require 21.3 GE. Furthermore, the overall biggest representatives require 39.3 GE, which is 84 % more than the smallest we found. This implies an area saving of 46 % when optimal S-boxes are carefully chosen compared to a (worst-case) random selection approach.

One possible next step is to manually synthesize all good candidate S-box with the aim to gain the minimal result. Since this is a cumbersome work, it is impossible for a lot of S-boxes. Our classification of S-boxes into 20 classes greatly reduces the work and helps to find the most area-efficient S-box.

Acknowledgement

References

1. Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. In Alfred Menezes and Scott A. Vanstone, editors, *CRYPTO*, volume 537 of *Lecture Notes in Computer Science*, pages 2–21. Springer, 1990.
2. Alex Biryukov, Christophe De Cannière, An Braeken, and Bart Preneel. A toolbox for cryptanalysis: Linear and affine equivalence algorithms. In Eli Biham, editor, *EUROCRYPT*, volume 2656 of *Lecture Notes in Computer Science*, pages 33–50. Springer, 2003.
3. Marcus Brinkman and Gregor Leander. On the classification of apn functions up to dimension five. *International Workshop on Coding and Cryptography*, 2007.

4. Claude Carlet, Pascale Charpin, and Victor Zinoviev. Codes, bent functions and permutations suitable for des-like cryptosystems. *Des. Codes Cryptography*, 15(2):125–156, 1998.
5. Nicolas Courtois and Josef Pieprzyk. Cryptanalysis of block ciphers with overdefined systems of equations. In Yuliang Zheng, editor, *ASIACRYPT*, volume 2501 of *Lecture Notes in Computer Science*, pages 267–287. Springer, 2002.
6. Nicolas Courtois and Josef Pieprzyk. Cryptanalysis of block ciphers with overdefined systems of equations. Cryptology ePrint Archive, Report 2002/044, 2002. http://eprint.iacr.org/.
7. L. Knudsen. private communication.
8. Gilles Lachaud and Jacques Wolfmann. The weights of the orthogonals of the extended quadratic binary goppa codes. *IEEE Transactions on Information Theory*, 36(3):686–, 1990.
9. C. S. Lorens. Invertible boolean functions. *IEEE Trans. Electronic Computers*, 13(5):529–541, 1964.
10. Mitsuru Matsui. Linear cryptoanalysis method for des cipher. In *EUROCRYPT*, pages 386–397, 1993.
11. Kaisa Nyberg. Perfect nonlinear s-boxes. In *EUROCRYPT*, pages 378–386, 1991.
12. Kaisa Nyberg. Differentially uniform mappings for cryptography. In *EURO-CRYPT*, pages 55–64, 1993.
13. O. S. Rothaus. On "bent" functions. *J. Comb. Theory, Ser. A*, 20(3):300–305, 1976.

A List of Representatives

G_0	$0, 1, 2, 13, 4, 7, 15, 6, 8, 11, 12, 9, 3, 14, 10, 5$
G_1	$0, 1, 2, 13, 4, 7, 15, 6, 8, 11, 14, 3, 5, 9, 10, 12$
G_2	$0, 1, 2, 13, 4, 7, 15, 6, 8, 11, 14, 3, 10, 12, 5, 9$
G_3	$0, 1, 2, 13, 4, 7, 15, 6, 8, 12, 5, 3, 10, 14, 11, 9$
G_4	$0, 1, 2, 13, 4, 7, 15, 6, 8, 12, 9, 11, 10, 14, 5, 3$
G_5	$0, 1, 2, 13, 4, 7, 15, 6, 8, 12, 11, 9, 10, 14, 3, 5$
G_6	$0, 1, 2, 13, 4, 7, 15, 6, 8, 12, 11, 9, 10, 14, 5, 3$
G_7	$0, 1, 2, 13, 4, 7, 15, 6, 8, 12, 14, 11, 10, 9, 3, 5$
G_8	$0, 1, 2, 13, 4, 7, 15, 6, 8, 14, 9, 5, 10, 11, 3, 12$
G_9	$0, 1, 2, 13, 4, 7, 15, 6, 8, 14, 11, 3, 5, 9, 10, 12$
G_{10}	$0, 1, 2, 13, 4, 7, 15, 6, 8, 14, 11, 5, 10, 9, 3, 12$
G_{11}	$0, 1, 2, 13, 4, 7, 15, 6, 8, 14, 11, 10, 5, 9, 12, 3$
G_{12}	$0, 1, 2, 13, 4, 7, 15, 6, 8, 14, 11, 10, 9, 3, 12, 5$
G_{13}	$0, 1, 2, 13, 4, 7, 15, 6, 8, 14, 12, 9, 5, 11, 10, 3$
G_{14}	$0, 1, 2, 13, 4, 7, 15, 6, 8, 14, 12, 11, 3, 9, 5, 10$
G_{15}	$0, 1, 2, 13, 4, 7, 15, 6, 8, 14, 12, 11, 9, 3, 10, 5$

Table 6. Representatives for all 16 classes of optimal 4 bit S-boxes (G_3 is equivalent to the invers mapping.)

G_0	$x^{14} + x^{13} + g^7x^{12} + g^5x^{11} + g^5x^{10}$ $+g^5x^9 + x^8 + g^6x^6 + g^{13}x^5 + g^8x^4 + g^{13}x^3 + g^{13}x^2$
G_1	$x^{14} + g^5x^{13} + x^{12} + gx^{11} + g^3x^{10}$ $+g^4x^9 + g^{14}x^8 + g^4x^7 + g^{12}x^6 + x^5 + g^{13}x^4 + x^3 + g^{13}x^2 + gx$
G_2	$x^{14} + g^6x^{13} + g^{13}x^{12} + g^{10}x^{10}$ $+g^6x^9 + gx^8 + g^2x^7 + g^{13}x^6 + g^{11}x^5 + g^2x^4 + g^{13}x^3 + g^2x^2 + g^6x$
G_3	$x^{14} + g^{11}x^{13} + gx^{12} + g^3x^{11}$ $+g^5x^9 + g^7x^8 + g^8x^7 + g^4x^6 + g^{11}x^5 + g^2x^4 + g^4x^3 + g^{11}x^2$
G_4	$x^{14} + g^{11}x^{13} + g^7x^{12} + gx^{11}$ $+g^8x^{10} + g^{13}x^9 + g^{11}x^8 + g^2x^6 + gx^5 + g^2x^4 + g^7x^3 + gx^2 + g^8x$
G_5	$x^{14} + g^{13}x^{13} + g^9x^{12} + g^6x^{11} + g^{10}x^{10}$ $+g^7x^9 + g^{10}x^8 + g^7x^7 + g^8x^6 + g^{12}x^5 + g^{12}x^4 + x^3 + g^{11}x^2 + g^5x$
G_6	$x^{14} + g^4x^{13} + g^3x^{12} + g^2x^{11} + x^{10}$ $+g^{11}x^9 + g^2x^8 + gx^7 + g^2x^6 + g^9x^5 + g^4x^4 + g^9x^3 + g^{12}x^2 + g^{11}x$
G_7	$x^{14} + gx^{13} + g^9x^{12} + gx^{11} + g^7x^{10}$ $+g^6x^7 + g^{10}x^6 + gx^5 + g^8x^4 + g^2x^3 + g^6x^2 + g^9x$
G_8	$x^{14} + gx^{13} + x^{12} + g^{10}x^9 + g^{14}x^8$ $+g^{12}x^7 + g^9x^5 + g^8x^4 + g^{13}x^3 + g^{11}x^2 + g^6x$
G_9	$x^{13} + g^7x^{12} + g^5x^{11} + gx^{10}$ $+g^{11}x^9 + g^{11}x^8 + g^3x^7 + g^4x^6 + g^5x^5 + gx^4 + g^7x^3 + x^2 + g^6x$
G_{10}	$x^{13} + g^{13}x^{12} + g^7x^{11} + g^7x^{10}$ $+g^{14}x^9 + g^{10}x^7 + gx^6 + g^5x^5 + g^7x^4 + g^{12}x^3 + g^6x$
G_{11}	$x^{14} + gx^{13} + x^{12} + g^7x^{11} + g^{13}x^{10}$ $+gx^9 + g^{11}x^8 + g^{14}x^7 + g^3x^6 + g^6x^5 + gx^4 + g^{14}x^3 + g^{14}x^2 + g^9x$
G_{12}	$x^{14} + g^{10}x^{13} + gx^{12} + g^4x^{11} + g^{14}x^{10}$ $+g^4x^9 + g^5x^8 + g^2x^7 + g^9x^6 + g^4x^5 + g^8x^4 + g^{14}x^3 + g^5x^2 + x$
G_{13}	$x^{14} + g^{12}x^{13} + g^8x^{12} + g^8x^{11} + g^{14}x^{10}$ $+gx^9 + g^8x^8 + g^{14}x^7 + g^6x^6 + x^5 + g^{14}x^4 + g^{12}x^3 + gx^2 + g^{14}x$
G_{14}	$x^{14} + g^8x^{13} + g^{10}x^{12} + gx^{11} + gx^{10}$ $+g^9x^9 + x^7 + g^{10}x^6 + g^7x^5 + g^4x^4 + g^2x^3 + g^{12}x^2 + g^{14}x$
G_{15}	$x^{14} + g^6x^{13} + g^{13}x^{12} + g^5x^{10} + x^9$ $+x^8 + x^7 + g^2x^6 + g^{11}x^5 + g^{10}x^4 + g^4x^3 + gx^2 + g^3x$

Table 7. Representatives for all 16 classes of optimal 4 bit S-boxes as Polynomials. g denotes a primitive element in F_{16}^*

R_0	0, 3, 5, 6, 7, 10, 11, 12, 13, 4, 14, 9, 8, 1, 2, 15
R_1	0, 3, 5, 8, 6, 9, 10, 7, 11, 12, 14, 2, 1, 15, 13, 4
R_2	0, 3, 5, 8, 6, 9, 11, 2, 13, 4, 14, 1, 10, 15, 7, 12
R_3	0, 3, 5, 8, 6, 10, 15, 4, 14, 13, 9, 2, 1, 7, 12, 11
R_4	0, 3, 5, 8, 6, 12, 11, 7, 9, 14, 10, 13, 15, 2, 1, 4
R_5	0, 3, 5, 8, 6, 12, 11, 7, 10, 4, 9, 14, 15, 1, 2, 13
R_6	0, 3, 5, 8, 6, 12, 11, 7, 10, 13, 9, 14, 15, 1, 2, 4
R_7	0, 3, 5, 8, 6, 12, 11, 7, 13, 10, 14, 4, 1, 15, 2, 9
R_8	0, 3, 5, 8, 6, 12, 15, 1, 10, 4, 9, 14, 13, 11, 2, 7
R_9	0, 3, 5, 8, 6, 12, 15, 2, 14, 9, 11, 7, 13, 10, 4, 1
R_{10}	0, 3, 5, 8, 6, 13, 15, 1, 9, 12, 2, 11, 10, 7, 4, 14
R_{11}	0, 3, 5, 8, 6, 13, 15, 2, 7, 4, 14, 11, 10, 1, 9, 12
R_{12}	0, 3, 5, 8, 6, 13, 15, 2, 12, 9, 10, 4, 11, 14, 1, 7
R_{13}	0, 3, 5, 8, 6, 15, 10, 1, 7, 9, 14, 4, 11, 12, 13, 2
R_{14}	0, 3, 5, 8, 7, 4, 9, 14, 15, 6, 2, 11, 10, 13, 12, 1
R_{15}	0, 3, 5, 8, 7, 9, 11, 14, 10, 13, 15, 4, 12, 2, 6, 1
R_{16}	0, 3, 5, 8, 9, 12, 14, 7, 10, 13, 15, 4, 6, 11, 1, 2
R_{17}	0, 3, 5, 8, 10, 13, 9, 4, 15, 6, 2, 1, 12, 11, 7, 14
R_{18}	0, 3, 5, 8, 11, 12, 6, 15, 14, 9, 2, 7, 4, 10, 13, 1
R_{19}	0, 3, 5, 10, 7, 12, 11, 6, 13, 4, 2, 9, 14, 1, 8, 15

Table 8. Representatives for all 20 Classes of Serpent-type S-boxes

$S-box$	G_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}	G_{13}	G_{14}	G_{15}
$W(0)$	108	108	108	90	90	90	90	90	108	96	96	90	90	90	96	96
$W(4)$	60	60	60	76	76	76	76	80	60	68	68	76	72	80	72	72
$W(-4)$	36	36	36	44	44	44	44	40	36	44	44	44	48	40	40	40
$W(8)$	27	27	27	22	22	22	22	20	27	25	25	22	24	20	23	23
$W(-8)$	9	9	9	8	8	8	8	10	9	7	7	8	6	10	9	9

Table 9. Walsh spectra for all 16 Classes of 4 bit S-boxes ($W(a)$ is the number of Walsh coefficients of value a)

$S-box$	R_0	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}	R_{11}	R_{12}	R_{13}	R_{14}	R_{15}	R_{16}	R_{17}	R_{18}	R_{19}
$W(0)$	108	108	108	96	96	96	108	108	96	108	108	108	108	96	108	96	108	108	108	108
$W(4)$	60	60	60	72	68	68	60	60	72	60	60	60	60	68	60	72	60	60	60	60
$W(-4)$	36	36	36	40	44	44	36	36	40	36	36	36	36	44	36	40	36	36	36	36
$W(8)$	27	27	27	23	25	25	27	27	23	27	27	27	27	25	27	23	27	27	27	27
$W(-8)$	9	9	9	9	7	7	9	9	9	9	9	9	9	7	9	9	9	9	9	9

Table 10. Walsh spectra for all 20 Classes of Serpent-type S-boxes ($W(a)$ is the number of Walsh coefficients of value a)