Implementation of SIMON 32/64 and 64/128 block cipher and Study of Cross-Correlation between Input and Output Sets and Linear Span.

Ahmad Sghaier Omar

ECE 628 Computer Network Security Project Report

Department of Electrical and Computer Engineering, University of Waterloo,

Email: a2sghaie@uwaterloo.ca

Abstract: *In this project, a lightweight cryptography block cipher SIMON, which is designed towards hardware efficient implementations, is realized in C/C++ language (SIMON 32/64 & 64/128) and an analysis was performed to check on its feature of cross correlation between input/output sets and among output sets, based on a message block generated by an 8 and 9 degree LFSR for 255 and 511 m-sequences. Computations of lower bound and upper bound in both cases have shown degree of correlation with maximum value for the upper bound set around 70 in SIMON 32/64 and upper bound set at 111 in SIMON 64/128 . The cipher ,based on the linear span computation using Berlekamp-Massy algorithm, was declared not immune to linear span attacks as the computation has shown a linear span for certain components to be less than N/2, with a profile of probability of 1/3 in 1 million iterations.*

1- Introduction

Lightweight cryptography has emerged as a domain that focuses on the design of cryptography algorithms that can satisfy the constraints in environments imposed by limited computing capability in devices such as smart sensors, RFID tags, wearable technology and many other devices falling into the category of sensor networks and Internet of Things devices. Those devices with small footprint, limited computing capability, low-power and being attached closer to humans has put more challenges on ensuring being secure and immune to attacks. Based on these two opposing situations of limited capability and requirement of efficient and secure cryptography, a number of lightweight cryptography algorithms were proposed. Among those, in June 2013 [1], the U.S. National Security Agency (NSA) has announced the specifications of two families of algorithms, SIMON and SPECK, that are ,as intended by the NSA, respectively, optimized lightweight cryptographic algorithms for hardware and software implementations given the silicon area occupied or code size required for certain performance metric. In this report, the implementation and analysis will focus on SIMON algorithm and will be further narrowed on two variants which are SIMON 32/64 and SIMON 64/128.

2- SIMON Structure Explained

The SIMON algorithm, as stated above, is an NSA approved algorithm with 10 variants known in the form of SIMON *2n/tn*, where *n* is the word size and has lengths of (16, 24, 32,48 or 64 bits), while *2n* is the block size and *tn* is the key size. In this project, we are concerned about the variants SIMON32/64 and SIMON 64/128, such that *n*=16 and 32, block size is 32 and 64 bits and the key is 64 and 128 bits, respectively [1].

The algorithm in all its variants is classified as a low cost, small foot-print Feistel structure-based block cipher, where in each round this structure is composed of one nonlinear operation and number of left shift and bitwise addition operations, as depicted in Figure.1, [1].

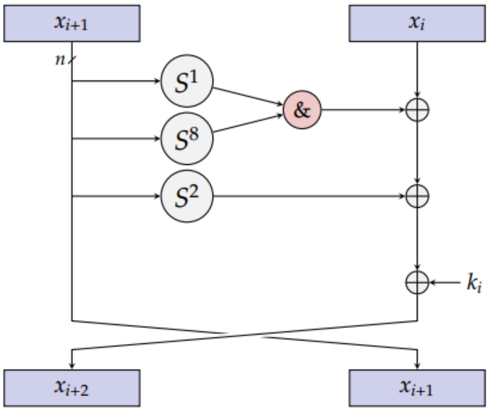

Figure 1 – SIMON one round Feistel Structure

Each block message of length *2n* bits is divided into two *n* bit words treated as left word (x_{i+1}) and right word (x_i), in each round the left word (x_{i+1}) is swapped into the right word of next round and the left word of the next round (x_{i+2}) is obtained by executing the shifting, AND and XOR operations of (x_{i+1}) with (x_i) and the round key (k_i).

The round keys (k_i's) are generated using key schedule function based on a master key and with three different functions structures that is built from two, three or four words, each of length *n*. In this project, as we are using the SIMON 32/64 and SIMON 64/128 where the key schedule function is built from four words of 16 or 32 bits as shown in figure 2, [1].

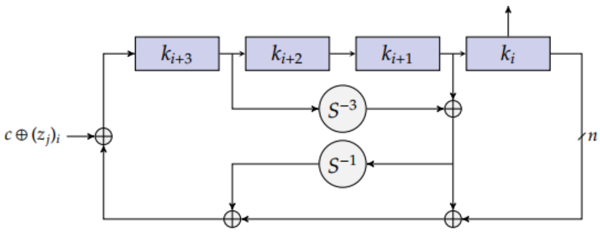

Figure 2 – Key Scheduling Function

3- Objectives of this project

Since SIMON was introduced lately, less than two years ago, so definitely many characteristics of this block cipher are not yet analyzed, even though some studies related to linear and differential cryptanalysis were performed to test further the proposed algorithm, [2], [3], [4] and [5]. In this project, the aim is to implement the SIMON32/64 and SIMON 64/128 algorithm variants in C/C++ language and study the cross correlation between the input and output of the algorithm as well as the cross correlation between sets of the output. Furthermore, the linear span of the output sets will be analyzed to check the cipher block immunity against linear span attacks.

The project suggests a configuration of the block cipher to be used with an LFSR of degree set to generate an m-sequence of period 255 or 511, this configuration will mimic the use of the block cipher as a key derivation function where the LFSR acts as PRSG and with the use of master key and the block cipher the output stream can be used as session keys or used also to build a stream cipher.

4- Project setup

Henceforth, the setup will be described based on SIMON 32/64 and later the setup for the SIMON 64/128 will highlight only the changes in the main setup. In the case of SIMON 32/64, the key is set as K = (0x80000000000000000000), while the message block is generated by an LFSR of degree 8 with a primitive polynomial $x^8 + x^7 + x^2 + x + 1$, and an initial state $(a_0, a_1, \ldots, a_7) = (1,0, \ldots ,0)$ to produce an m-sequence $a = (a_0, a_1, \ldots, a_{254})$. The message block of 32-bit each is derived from slicing the 255 bits m-sequence into 32-bits messages by using a sliding window with a step of 1-bit, to ensure building an array of 255*32 where none of these messages are repeated, where each message block (y_i) is given by:

$$y_i = (a_i, a_{i+1}, \ldots, a_{i+31}), i = 0,1, \ldots ,254$$

The output of the 32-bit key stream vector for the fed message blocks is given by:

$$s_i = Enc_{Simon,k}(y_i), i = 0,1, \ldots ,254.$$

$$Y = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_{254} \end{pmatrix} \overset{Enc}{\Longrightarrow} S = \begin{pmatrix} s_0 \\ s_1 \\ \vdots \\ s_{254} \end{pmatrix}$$

Based on this configuration the analysis will focus on studying the cross correlation between the 32 columns of the key stream matrix S with those in the message block matrix Y, where each is a sequence of 255 bits, and each column in Y is an m-sequence that is generated by the LFSR or a shifted version. Also, the cross correlation between the 32 columns of S will also be studied to investigate the upper and lower bound of each pair cross correlation.
In case of SIMON 64/128, a similar setup was used, except for the modifications needed to accommodate the different word and key lengths. The key is chosen as per the standard document test vector. The Y matrix was generated in the same approach with the use of an LFSR defined by the primitive polynomial $x^9 + x^4 + 1$, and an initial state $(a_0, a_1, \ldots, a_8) = (1,0,\ldots,0)$ to produce an m-sequence $a = (a_0, a_1, \ldots, a_{511})$. The message block is formed through extracting 64 bits words from the 511 m-sequence to build an array of 64*511, where each y_i is given by: $y_i = (a_i, a_{i+1}, \ldots, a_{i+63}), i = 0,1,\ldots,511$
The analysis was performed first on a single iteration using the given key. However, to draw a general conclusion both the cross correlation and linear span computations were performed over 10*4 and 10*6 different keys and the frequency of occurrence for each component was recorded.

5- Description of Implementation

The algorithm in this project was implemented in C/C++ language according to the standard document [1]. Here also, the description of the implementation will focus on the SIMON 32/64 given that the only difference will be in the parameters listed in previous section.
The implementation was structured to create the message block through a function that implements the 8 degree LFSR and extracts the 32 bit messages in an array of 255 elements. Similarly, the function key scheduling function named KeyLUT was used to construct a look-up table of 32 elements each of width 16 bits through the NLFSR structure. These two functions are called at the beginning of the program to build the two arrays, and Figure 3 describes the implementation approach.
The main encryption function was built as per the algorithm specifications and was tested along with the key scheduling function using the test vectors supplied in the specification document [1], which are given here.

SIMON 32/64	***Key:*** *1918 1110 0908 0100,*
	Plaintext: *6565 6877,* ***Ciphertext:*** *c69b e9bb*
SIMON 64/128	***Key:*** *1b1a1918 13121110 0b0a0908 03020100*
	Plaintext: *656b696c 20646e75* ***Ciphertext:*** *44c8fc20 b9dfa07a*

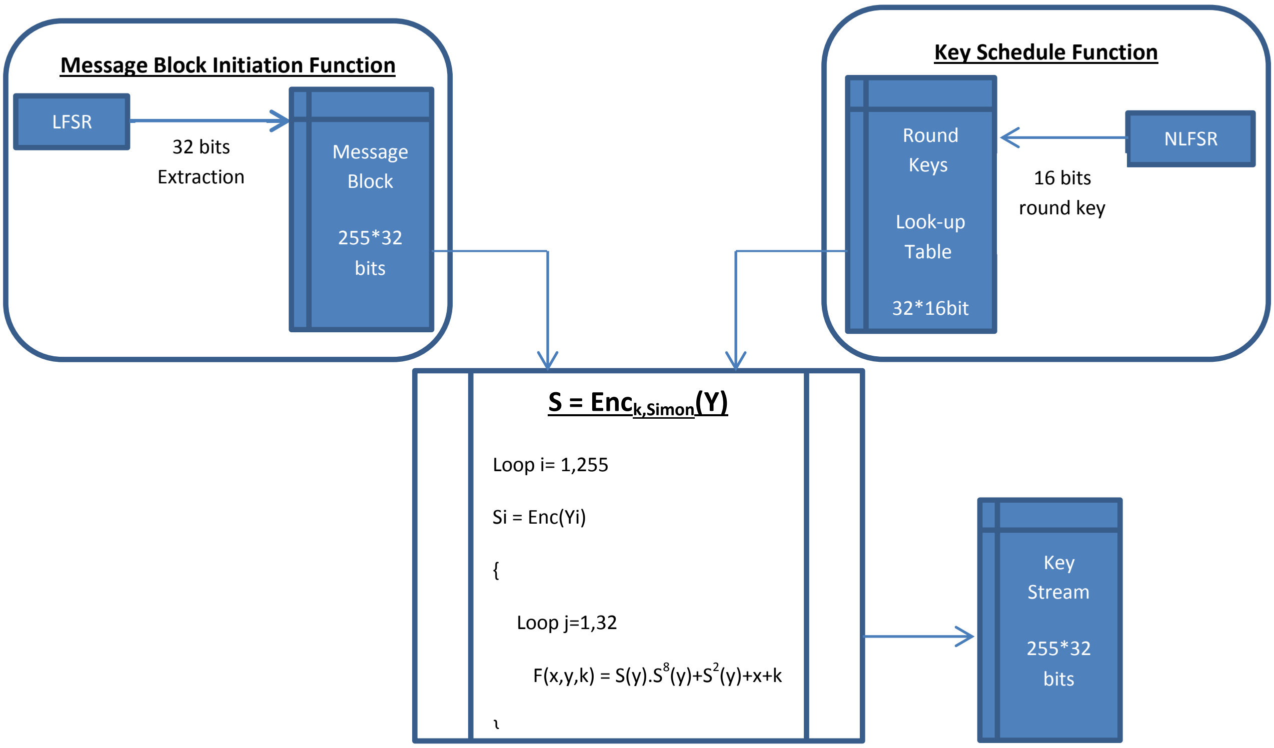

Figure 3 – Description of the Implementation Approach

6- Cross Correlation Computations Results and Analysis

In this section we discuss the results of the implementation and the focus will be on the analysis of the computation of cross-correlation between the input and output sets, and among the output sets then deriving the conclusion based on this.

Firstly, as per the above mentioned setup, the detailed content of message block array Y and the output key stream vectors S for the two variants are listed in appendix A.2.

SIMON 32/64

- ***Input-Output sets cross-correlation***

For this part of cross-correlation analysis, the two matrices Y and S where serialized to generate a 32 columns per each matrix, the analysis then performed by applying the cross correlation equation between each column in S and the m-sequence of the 8 degree LFSR since it represents the first column in Y matrix since the rest of columns in Y are shifted versions of column1.

$$C_\tau(y_a, s_b) = \sum_{i=0}^{254} (-1)^{y_i + s_{i+\tau}} \quad ; \tau = 0,1, \dots, 254$$

The analysis lists the upper and lower bound for the cross-correlation of each vector in S with the m-sequence vector and highlights the range of values for the upper bound. The detailed values are found in Appendix A.1, while herein figure 4 depicts those values.

The values show that the maximum value of cross-correlation is at 69 while the lower bound is set at 1, and the range of the upper bound is between 41 and 69. According to the relationship of cross correlation and sequence length ($Cross\ correlation \leq c\sqrt{N}$), it is found that c is in the range of 2.5675 and 4.32. This indicates that the SIMON block cipher has a significant degree of correlation between input and output sets.

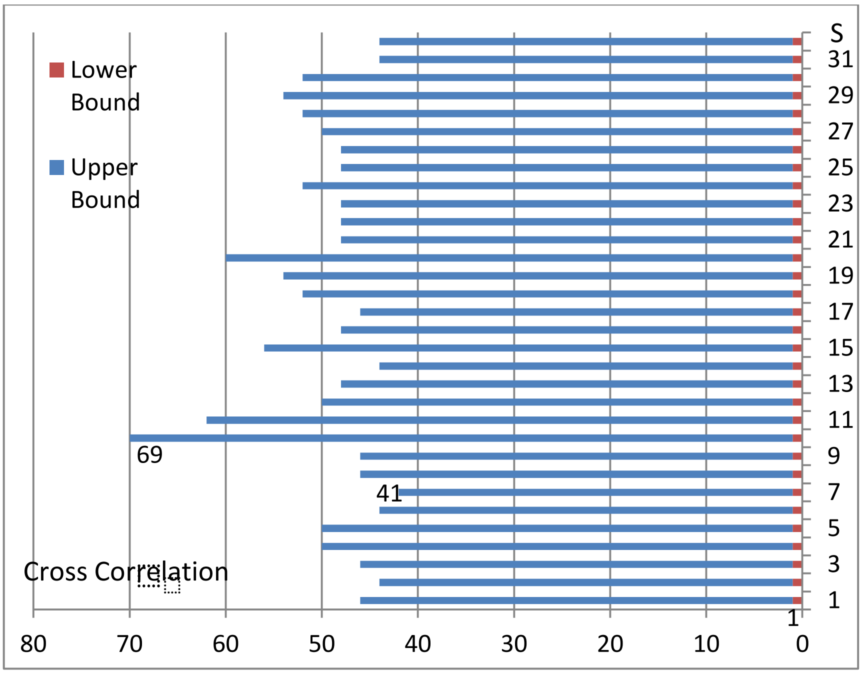

Figure 4 – Cross-Correlation Upper bound and lower bound values between key stream matrix columns and message block matrix columns for SIMON 32/64

- ***Output sets cross-correlation***

In addition to calculating the cross-correlation between the m-sequence input vector and the columns of key stream matrix, also in this subsection we comment on the calculations performed for the cross-correlation among the columns of the key stream matrix. This calculations were performed by using the above mentioned cross correlation function and by iterating through the S matrix columns as in the pseudo code below

Loop I = 0,30

Loop J=I+1,31

Loop Tao = 0,254

$C = cross\text{-}correlation(S_i, S_{j+tao})$

This will ensure calculating the cross-correlation among all vectors without the need for repeating the calculations for the cases when the indexes are exchanged (e.g. $C(S_1,S_2)$ and $C(S_2,S_1)$).

The results of these calculations are represented by Figure 5, which plots them as a two-dimensional heat map. According to the figure below and the detailed results, it is found that the lower bound is set at 1 and the upper bound between all columns pairs is 67, ranging between 35 and 67, which also shows a significant degree of correlated output sets.

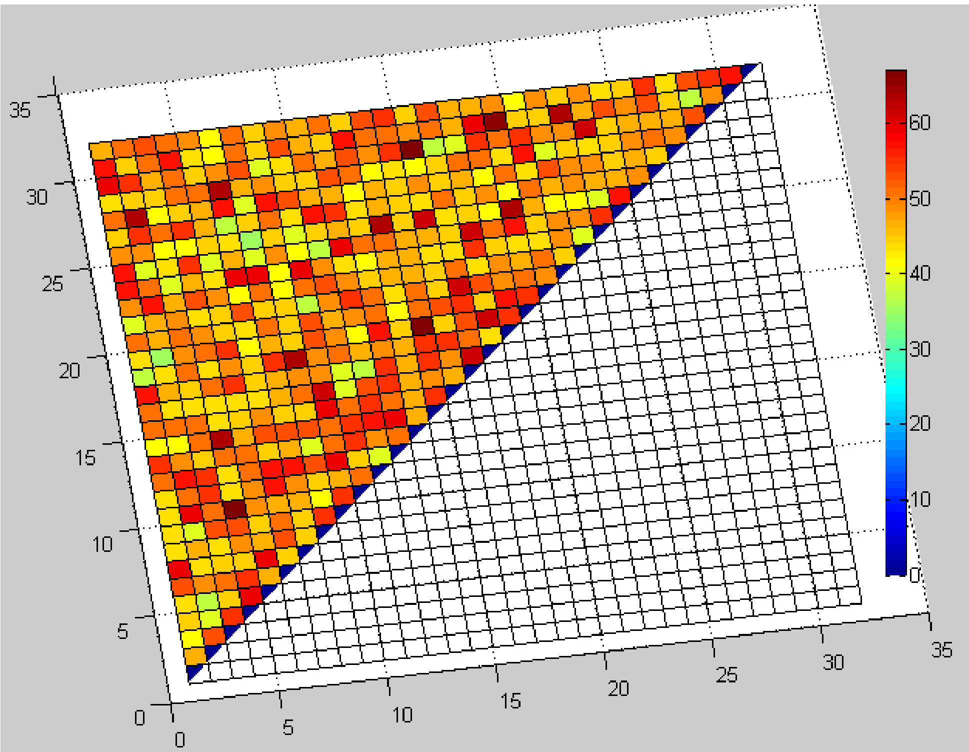

Figure 5 – Cross-Correlation Upper bound values among key stream matrix columns for SIMON 32/64.

A run of 10*4 different keys was performed and the maximum value of the upper bound was recorded and its frequency of occurrence. The distribution showed concentration around the values of 37 to 69 with the lowest value was recorded at 35 occurring 72 times and highest value was 89 occurred only once in 10,000 iterations. The figure below plots the frequency of occurrence of each cross correlation value. In the case of SIMON 64/128 similar behavior was captured over a run of 10*4 keys with highest value at 131 and lowest at 53 with approximately the same shape of frequency distribution, the figure for the case of SIMON 64/128 can be found in appendix A.4.

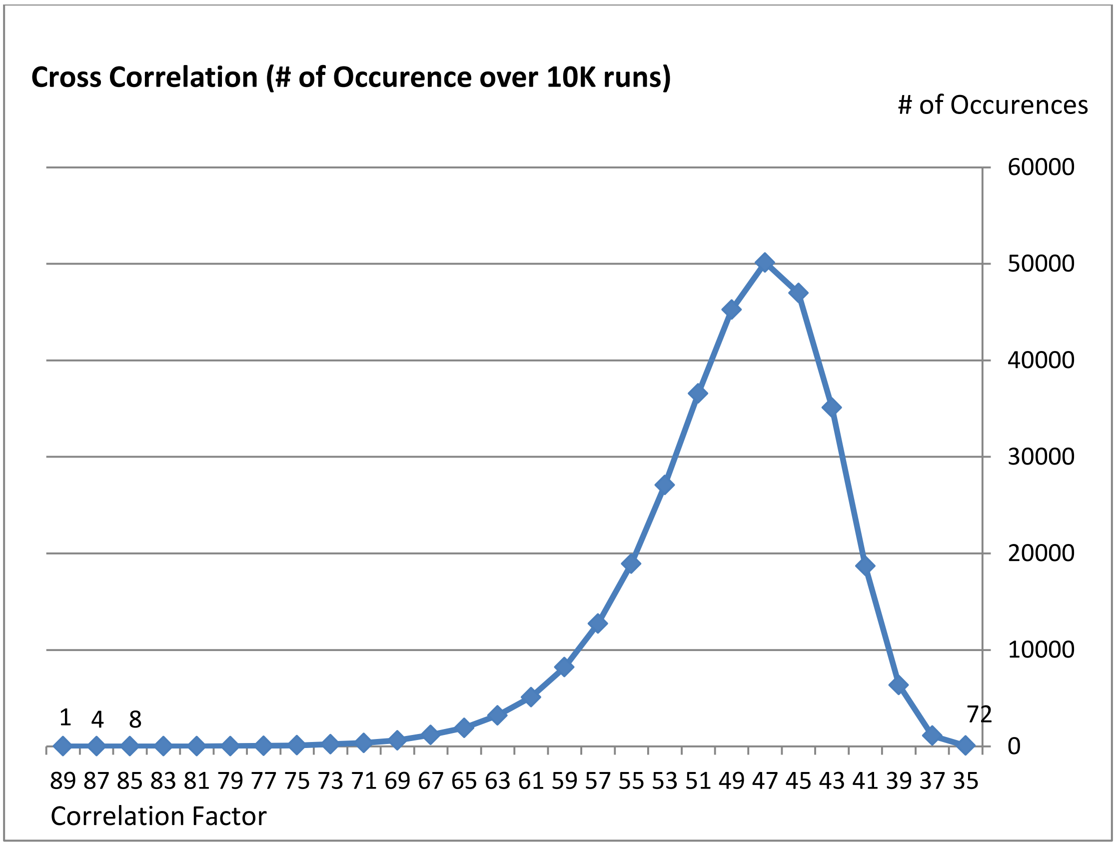

Figure 6 – Cross-Correlation Upper bound values Frequency Distribution over 10K runs.

SIMON 64/128

In the case of SIMON 64/128, the same analysis approach was applied on the first 32 columns of the Y and S matrices. It was found that for input-output cross correlation the maximum value of upper bound is 93 with a range between 61 and 93, while the lower bound is set at 1. Furthermore, the cross correlation between output sets was upper bounded by a maximum value of 111 and ranging between 53 and 111, while also it is lower bounded by a value equals 1.

The two figures below, Figure 6 and 7 depict the results for the SIMON 64/128. Here also, it is clear that cross correlation values are indicating correlation between input and output sets and among output sets, and it is noticed also that the cross correlation value did not double as per the block size but it increased by not more than 1.65 times than the values obtained in the case of SIMON 32/64.

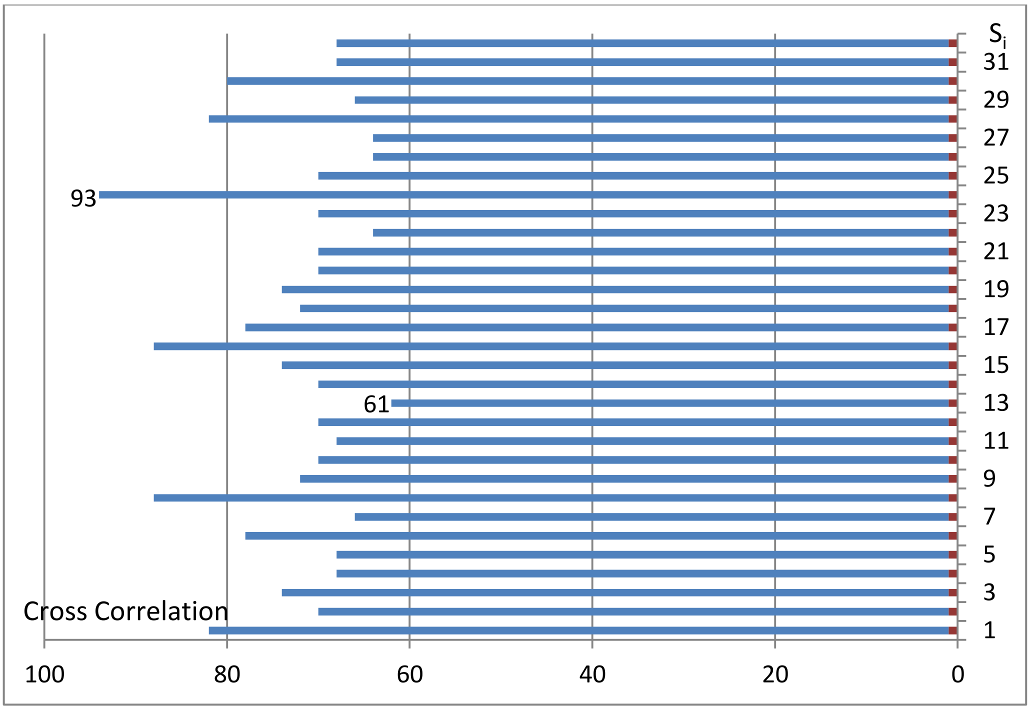

Figure 4 – Cross-Correlation Upper bound and lower bound values between key stream matrix columns and message block matrix columns for SIMON 32/64

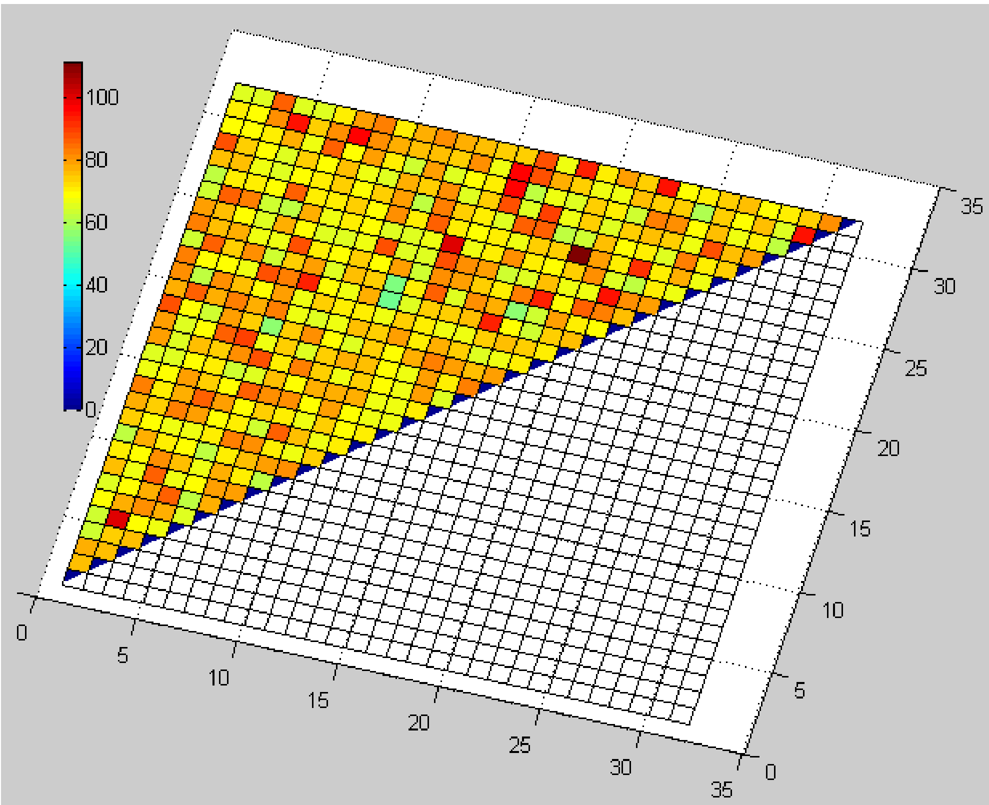

Figure 7 – Cross-Correlation Upper bounds among key stream columns for SIMON 64/128.

7- Linear Span Results and Analysis

The stream of output columns for the case of SIMON32/64 were further analyzed by computing the linear span of each column, being a sequence of 255 bits, using the Berlekamp-Massey algorithm [6]. The analysis first looked at few keys and listed the different values obtained for the linear span. For the SIMON 32/64, the results for certain columns has shown a linear span values of 124, 125, 126 and 127 which means that those output sets of columns have a linear span LS=n, where 2n<N.

This indicates that by observing 2n consecutive bits the LFSR that can generate the whole sequence can be derived using the Berlekamp-Massy algorithm and this results that the cipher is not considered immune against linear span attacks. The results of computations using the Berlekamp-Massy algorithm over four keys are shown in the figure below.

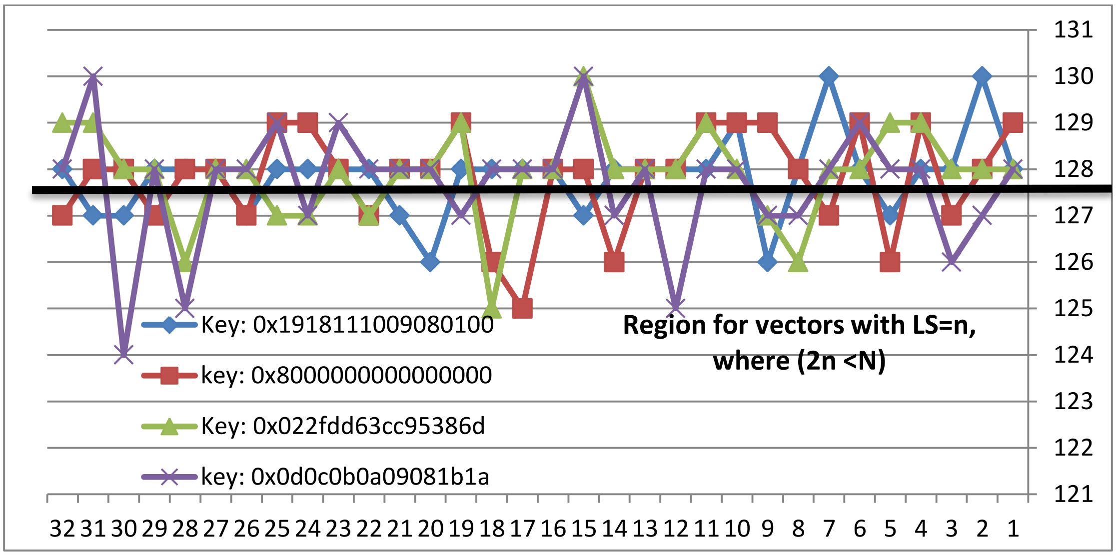

Figure 8 – Linear Span Computations over 4 different keys.

To further conclude the results, a run using 10*6 different keys was performed to list the frequency of linear span occurrence per each column. The linear span computations confirm the linear span attack likelihood on SIMON32/64, where it presents values of linear span as low as 113. In addition a run for a subset of 10*4 keys and using double the sequence length (N=510) affirmed those results, where the maximum linear span value did not excced 255. Similarly, the case of SIMON 64/128 as illustrated in appendix A.4 showed similar results with linear span values as low as 247.

Moreover, the probability distribution of the frequency of occurrence of the linear span values is shown hereafter for the case of SIMON32/64. The graph indicates with probability of 1/3 to have a linear span less than N/2. These results could be attributed to the simple design and in specific the simple non-linear function (AND operation) which is a quadratic function. However, further analysis is required to identify and relate that to the Fiestel structure used.

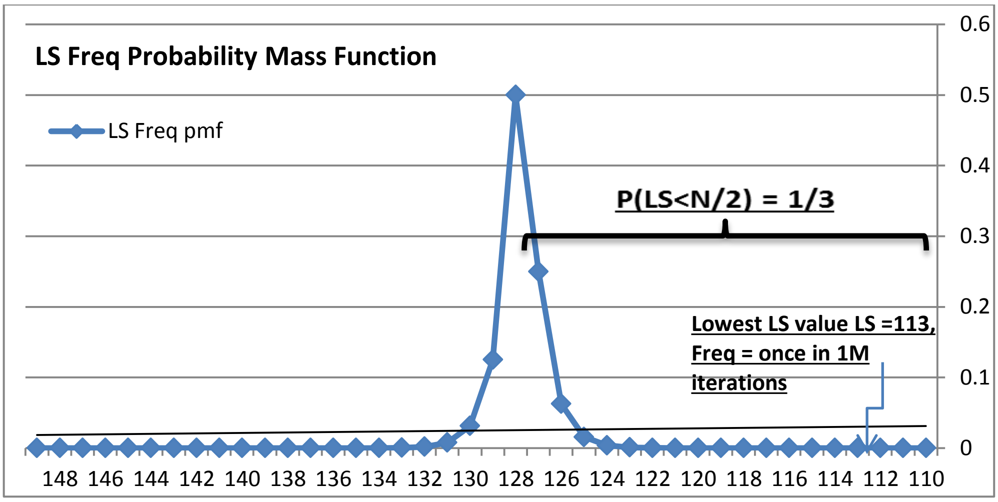

Figure 9 – Linear Span Probability Mass Function derived from 1M iterations.

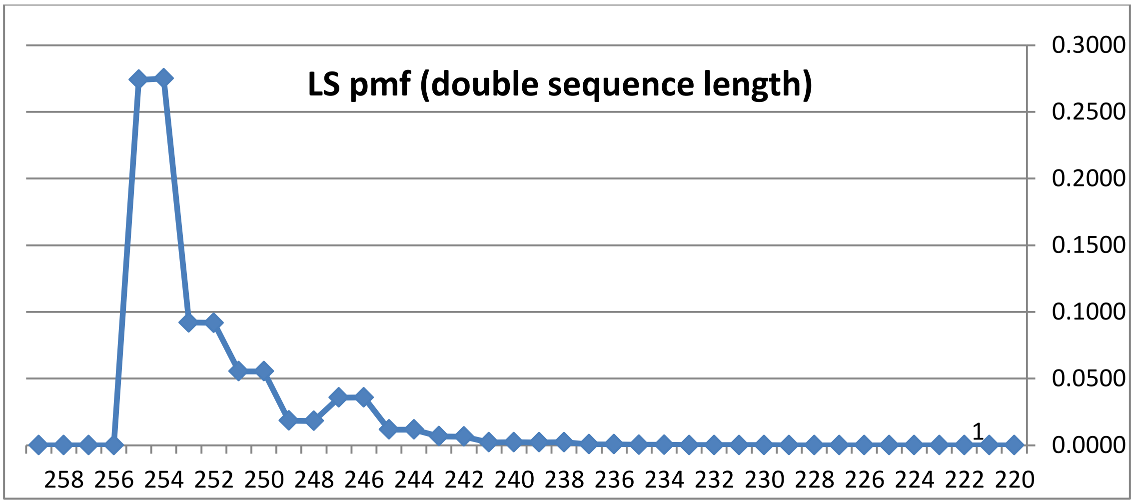

Figure 10 – Linear Span PMF (10K iterations using double length sequence - N=510).

8- Analysis on the Evaluation of Lightweight Crypto Systems

In this section, answers to the three questions of evaluating lightweight crypto systems are provided.

Firstly, adversary model that should be considered in the reality of lightweight cryptography derives mainly from the applications that are related to small devices with limited capabilities. These adversary models could be: impersonation attack, cloning attack, replay attack, eavesdropping attack, data correlation and association attacks, physical attacks, denial-of-service attack, tracking and location-related attack. Furthermore, the adversary model can be classified as being either passive such as in eavesdropping or active as in replay attacks or DoS attacks.

Secondly, regarding the argument that the amount of data encrypted by a single lightweight device, e.g., Simon32/64, during its functional lifetime will be tiny, and data to which an adversary has access will likely remain small might not be enough to assume a relaxed security environment. Actually having a tiny amount of information exchanged which is in most cases repetitive and very well known in its format and given the simplicity of deign lightweight crypto such as SIMON, this makes the adversary capability to perform cryptanalysis or side channel attacks easier with minimum data sets and computing power.

Thirdly, the point that there is a price to be paid (with every encryption) for making lightweight cryptography theoretical resistance on all known cryptanalytic attacks can be related to different techniques that can be applied but the price for these techniques will impose prices either on financial cost, time, complexity or equipment design and its constrained environment definition. For example, a price would come from device tampering to resist physical attacks which will result increase of cost and design. Also, introducing extra clocks or some background noise to defend against side channel attacks will result complexity of design and could also affect performance.

9- Conclusion

In this project, a lightweight cryptography algorithm was implemented in C/ C++ language, which is a new algorithm specified by the NSA in 2013 named SIMON. The algorithm as promised by the standard document is a lightweight algorithm that was implemented in less than 100 lines of code for the core functions.

The setup proposed in this project to study the cross correlation characteristics revealed that the algorithm has shown a significant degree of correlation between input and output sets as well as among output sets.

The results show that the upper bound for input-output cross correlation (in SIMON 32/64 and 64/128) was upper bounded by a maximum values at 69 and 93, respectively. While the cross correlation among output sets is at maximum upper bounded by 67 in the case of SIMON 32/64 and it is at 111 in SIMON 64/128. Iterations of 10K different keys confirmed those findings, and also showed the frequency of occurrence of those correlation factors.

Furthermore, the linear span computations on the output sequence showed susceptibility of the proposed block cipher for linear span attack, where a linear span of values less than half the sequence length were obtained in both cases with probability of 1/3.

The primary conclusion is that the studied cipher showed that its behavior does not provide the required randomness features which are essential for block ciphers, while future work is needed to design practical attacks based on those findings and study the other cipher SPECK using the same approach to compare with its associate SIMON.

References

1. *Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., and Wingers, L. The Simon and Speck Families of Lightweight Block Ciphers. Cryptology e-Print Archive, Report 2013/404 (2013). http://eprint.iacr.org/.*
2. *Abdelraheem, M. , Alizadeh, J. , Alkhzaimi, H. , Reza Aref, M. , Bagheri, N. , Gauravaram, P. , and Lauridsen,M. Improved Linear Cryptanalysis of Reduced-round SIMON. Cryptology e-Print Archive, Report 2014/681 (2014). http://eprint.iacr.org/.*
3. *Alkhzaimi, H., Lauridsen, M. Cryptanalysis of the SIMON Family of Block Ciphers. Cryptology e-Print Archive, Report 2013/543 (2013). http://eprint.iacr.org/.*
4. *Alizadeh, J. , Alkhzaimi, H., Reza Aref, M. , Bagheri, N. , Gauravaram, P. , Kumar, A. , Lauridsen, M. , and Sanadhya, S. Cryptanalysis of SIMON Variants with Connections. Cryptology e-Print Archive, Report 2013/663 (2013). http://eprint.iacr.org/.*
5. *Abed, F., List, E., Lucks, S., Wenzel, J. Differential Cryptanalysis of Reduced-Round Simon. Cryptology e-Print Archive, Report 2013/526 (2013). http://eprint.iacr.org/.*
6. *Chen, L., and Guang, G. Communication System Security, Chapman & Hall/CRC, (2012).*

Appendix A.1 - Cross Correlation Values (Between Input and Output streams)

SIMON 32/64

0 {max: 45, min: 1}	1 {max: 43, min: 1}	2 {max: 45, min: 1}	3 {max: 49, min: 1}
4 {max: 49, min: 1}	5 {max: 43, min: 1}	**6 {max: 41, min: 1}**	7 {max: 45, min: 1}
8 {max: 45, min: 1}	**9 {max: 69, min: 1}**	10 {max: 61, min: 1}	11 {max: 49, min: 1}
12 {max: 47, min: 1}	13 {max: 43, min: 1}	14 {max: 55, min: 1}	15 {max: 47, min: 1}
16 {max: 45, min: 1}	17 {max: 51, min: 1}	18 {max: 53, min: 1}	19 {max: 59, min: 1}
20 {max: 47, min: 1}	21 {max: 47, min: 1}	22 {max: 47, min: 1}	23 {max: 51, min: 1}
24 {max: 47, min: 1}	25 {max: 47, min: 1}	26 {max: 49, min: 1}	27 {max: 51, min: 1}
28 {max: 53, min: 1}	29 {max: 51, min: 1}	30 {max: 43, min: 1}	31 {max: 43, min: 1}

SIMON 64/128

0 {max: 81, min: 1}	1 {max: 69, min: 1}	2 {max: 73, min: 1}	3 {max: 67, min: 1}
4 {max: 67, min: 1}	5 {max: 77, min: 1}	6 {max: 65, min: 1}	7 {max: 87, min: 1}
8 {max: 71, min: 1}	9 {max: 69, min: 1}	10 {max: 67, min: 1}	11 {max: 69, min: 1}
12 {max: 61, min: 1}	13 {max: 69, min: 1}	14 {max: 73, min: 1}	15 {max: 87, min: 1}
16 {max: 77, min: 1}	17 {max: 71, min: 1}	18 {max: 73, min: 1}	19 {max: 69, min: 1}
20 {max: 69, min: 1}	21 {max: 63, min: 1}	22 {max: 69, min: 1}	**23 {max: 93, min: 1}**
24 {max: 69, min: 1}	25 {max: 63, min: 1}	26 {max: 63, min: 1}	27 {max: 81, min: 1}
28 {max: 65, min: 1}	29 {max: 79, min: 1}	30 {max: 67, min: 1}	31 {max: 67, min: 1}

Appendix A.2 - Results of Y matrix and S Matrix

SIMON 32/64 (KEY = 0x8000000000000000)

IDX	Yi	Si	IDX	Yi	Si	IDX	Yi	Si
0	f5a53f01	aed087a9	**53**	e2e99549	7d895fae	**106**	721eef41	5306735d
1	fad29f80	8a992db3	**54**	7174caa4	3ef88bd6	**107**	b90f77a0	eaf352ed
2	7d694fc0	9ff21273	**55**	b8ba6552	391b61a3	**108**	5c87bbd0	f858781e
3	3eb4a7e0	27d0551d	**56**	dc5d32a9	f51b0d26	**109**	ae43dde8	172879f2
4	1f5a53f0	9c9d0cc7	**57**	6e2e9954	2707cf65	**110**	d721eef4	b6e4fc67
5	8fad29f8	c0192016	**58**	37174caa	03e7e586	**111**	6b90f77a	db31f94e
6	47d694fc	2b2dcb31	**59**	9b8ba655	c2c28e6c	**112**	35c87bbd	828aa077
7	a3eb4a7e	e302d4ad	**60**	cdc5d32a	52baec5b	**113**	1ae43dde	c2e1d738
8	d1f5a53f	9b9fbed9	**61**	e6e2e995	388573b7	**114**	8d721eef	511b8214
9	68fad29f	1729592f	**62**	f37174ca	2a41a28b	**115**	c6b90f77	9291b116
10	347d694f	1f0173af	**63**	f9b8ba65	002cdce6	**116**	e35c87bb	926a5731
11	9a3eb4a7	851ab670	**64**	7cdc5d32	1c4ea5dc	**117**	f1ae43dd	e8173715
12	4d1f5a53	3f6efd95	**65**	be6e2e99	830b84af	**118**	78d721ee	f7ee3ee1
13	268fad29	6ed8efeb	**66**	df37174c	1dc7f4d8	**119**	3c6b90f7	643c0da9
14	1347d694	fdb8282a	**67**	6f9b8ba6	7e0a1248	**120**	9e35c87b	a97a9624
15	09a3eb4a	2d9a8728	**68**	b7cdc5d3	3f33d0f9	**121**	cf1ae43d	b45f5ab5
16	84d1f5a5	97c06452	**69**	5be6e2e9	99eb5dfa	**122**	678d721e	f426db92
17	4268fad2	4a29bf1c	**70**	2df37174	4134fa28	**123**	b3c6b90f	a0935ca1
18	a1347d69	80cc7675	**71**	16f9b8ba	0205741f	**124**	d9e35c87	69b7bfce
19	509a3eb4	72fb898b	**72**	0b7cdc5d	5c232f58	**125**	6cf1ae43	2eaeab2f
20	284d1f5a	4af5319b	**73**	05be6e2e	37bbeb73	**126**	b678d721	e30702fc
21	14268fad	8c3c5b5f	**74**	02df3717	214fdf2b	**127**	db3c6b90	fc2a2563
22	8a1347d6	afd7c49e	**75**	816f9b8b	8015d70d	**128**	ed9e35c8	bbdf5b0e
23	4509a3eb	c09a293e	**76**	40b7cdc5	7349681d	**129**	f6cf1ae4	c53cd99f
24	2284d1f5	7594b578	**77**	205be6e2	36528065	**130**	fb678d72	d442d2af
25	914268fa	fac74c93	**78**	102df371	634719f7	**131**	fdb3c6b9	7bf9f268
26	48a1347d	876674b6	**79**	0816f9b8	bd0e2ba1	**132**	fed9e35c	8b69ffec
27	24509a3e	1d057a88	**80**	040b7cdc	3fa67f9b	**133**	ff6cf1ae	c62c812b
28	92284d1f	bf1878e7	**81**	8205be6e	4817c080	**134**	7fb678d7	47de32d8
29	4914268f	1bff8cab	**82**	4102df37	d91543d4	**135**	bfdb3c6b	f9d4d5b0
30	a48a1347	b3c561f9	**83**	a0816f9b	51800a19	**136**	5fed9e35	e98bf682
31	524509a3	2171f9f9	**84**	d040b7cd	602c98b0	**137**	aff6cf1a	98c68ea8
32	a92284d1	7459af64	**85**	e8205be6	4432158c	**138**	57fb678d	d9a13ff5
33	54914268	da03f7d7	**86**	f4102df3	1d55be61	**139**	abfdb3c6	0440b2bd
34	aa48a134	ebaaac55	**87**	7a0816f9	837fbc5b	**140**	d5fed9e3	bb7c4037
35	5524509a	86744536	**88**	bd040b7c	1cbeac6e	**141**	eaff6cf1	5a168dd1
36	2a92284d	25445403	**89**	de8205be	9c9a827b	**142**	757fb678	7.60E+12
37	95491426	d9acf44c	**90**	ef4102df	4d77e810	**143**	3abfdb3c	399e8cfe
38	caa48a13	1526c106	**91**	77a0816f	4f603dda	**144**	9d5fed9e	6c496c1b
39	65524509	6045f736	**92**	bbd040b7	0da2c0c3	**145**	ceaff6cf	6e1c3027
40	32a92284	4259f44e	**93**	dde8205b	da3045f2	**146**	e757fb67	2651ac5a
41	99549142	971b92a2	**94**	eef4102d	9645033b	**147**	73abfdb3	1893a981
42	4caa48a1	933a261b	**95**	f77a0816	af07af0f	**148**	39d5fed9	94422f07
43	a6552450	9f501128	**96**	7bbd040b	5df9220d	**149**	9ceaff6c	8ca905c2
44	d32a9228	4058520c	**97**	3dde8205	e698c251	**150**	4e757fb6	42271ef0
45	e9954914	6c2f0ae1	**98**	1eef4102	9801bc1c	**151**	273abfdb	ee2683ea
46	74caa48a	75cd6d38	**99**	0f77a081	a88514f2	**152**	939d5fed	b971bc04
47	ba655245	3e2db655	**100**	87bbd040	69d342ee	**153**	c9ceaff6	51234c26
48	5d32a922	88cc1ef0	**101**	43dde820	64ae8051	**154**	64e757fb	1cc1d4a2
49	2e995491	c0eae7ad	**102**	21eef410	88d9ed9b	**155**	b273abfd	03a8d0fd
50	174caa48	e2aa8e56	**103**	90f77a08	7071dffc	**156**	5939d5fe	b5c3bc18
51	8ba65524	d774c45d	**104**	c87bbd04	5de267cb	**157**	ac9ceaff	bd7121cf
52	c5d32a92	ace12b31	**105**	e43dde82	f7c150bf	**158**	564e757f	a0e00f62

IDX	Yi	Si	IDX	Yi	Si	IDX	Yi	Si
159	2b273abf	208f8a9a	211	2f2c4663	20df033a			
160	15939d5f	0d125913	212	97962331	b7cd5990			
161	0ac9ceaf	25cacdbf	213	4bcb1198	7605eeb2			
162	8564e757	859faa09	214	25e588cc	8499ab39			
163	c2b273ab	91ecd8c3	215	12f2c466	4257e75b			
164	615939d5	30ba6d6e	216	89796233	584d47d4			
165	b0ac9cea	56d8882b	217	44bcb119	5f1437ed			
166	d8564e75	a47d4c73	218	a25e588c	6b392ec5			
167	ec2b273a	f79ff84f	219	512f2c46	39843aff			
168	7615939d	3b62ffd1	220	a8979623	959c9f69			
169	3b0ac9ce	0477af55	221	d44bcb11	6c2cecdf			
170	1d8564e7	0dcf4695	222	6a25e588	01feff62			
171	0ec2b273	e2fb85ff	223	b512f2c4	0e61fbda			
172	7615939	ec73795c	224	da897962	4f75a7f6			
173	83b0ac9c	9a450bec	225	6d44bcb1	0fa0ab6f			
174	c1d8564e	9279c92f	226	36a25e58	a1a0d647			
175	60ec2b27	958ecda2	227	1b512f2c	591b95bc			
176	30761593	3b5c5094	228	0da89796	88eeba1b			
177	183b0ac9	e702843f	229	06d44bcb	87e251fb			
178	0c1d8564	4f5b1c55	230	036a25e5	98228b06			
179	860ec2b2	ba7a3f0c	231	01b512f2	50e833b1			
180	c3076159	66a6cfd8	232	80da8979	a99ddf8d			
181	e183b0ac	7dbf47af	233	c06d44bc	c3a5c592			
182	70c1d856	3fde7ec4	234	e036a25e	e6f2e6f4			
183	3860ec2b	9758297f	235	f01b512f	f9bb2ea7			
184	1c307615	6b1e0ff5	236	f80da897	a931950e			
185	8e183b0a	958b6a8b	237	fc06d44b	b46c9ed4			
186	c70c1d85	241ba6ca	238	7e036a25	4d5e3fa3			
187	63860ec2	56697c3b	239	3f01b512	62bfe427			
188	31c30761	5bf8a7af	240	9f80da89	deb313f9			
189	98e183b0	e24bc96c	241	4fc06d44	ba4a4d0d			
190	cc70c1d8	e82d3ede	242	a7e036a2	f6e21cda			
191	663860ec	84c4fb15	243	53f01b51	1ca0a004			
192	331c3076	9658f6ba	244	29f80da8	f6dc8448			
193	198e183b	cf92d369	245	94fc06d4	fb7de31f			
194	8cc70c1d	9da586d2	246	4a7e036a	c662f104			
195	4663860e	f97b2d72	247	a53f01b5	3baaa255			
196	2331c307	99ea0814	248	d29f80da	37b362f8			
197	1.20E+186	b95c5144	249	694fc06d	90138379			
198	88cc70c1	1acf0a26	250	b4a7e036	b68e5145			
199	c4663860	ab721c80	251	5a53f01b	8d0dc747			
200	62331c30	84faaac2	252	ad29f80d	5a86903c			
201	b1198e18	8ba21706	253	d694fc06	5ef62654			
202	588cc70c	3a50b8f9	254	eb4a7e03	2826cbe8			
203	2c466386	a9e95799						
204	962331c3	6580dd59						
205	cb1198e1	b183090c						
206	e588cc70	b577cd9d						
207	f2c46638	38698bff						
208	7962331c	6ff33f50						
209	bcb1198e	2487cdf7						
210	5e588cc7	e0c93364						

SIMON 64/128 (KEY = 0x1b1a1918131211100b0a090803020100)

IDX	Yi	Si	IDX	Yi	Si
0	9D97B0D5390C4201	389C408B9886E191	54	93F6926FCB50A276	8312DCC613ECAD24
1	4ECBD86A9C862100	83B539220153D4AE	55	C9FB4937E5A8513B	5141AC3411987BA5
2	2765EC354E431080	659AC6BE09824748	56	64FDA49BF2D4289D	A26D0D4B15CBCC6E
3	13B2F61AA7218840	E59F748C3D1E25E7	57	B27ED24DF96A144E	E763CE0F236392ED
4	89D97B0D5390C420	3CD5C2937324868A	58	593F6926FCB50A27	4A9E79BD9E582262
5	44ECBD86A9C86210	319C95CCA479AC1	59	AC9FB4937E5A8513	03AFCFECADFF8EB7
6	A2765EC354E43108	2F9FA35B97C9FC0	60	564FDA49BF2D4289	4B6A2A5ABAC0FFAC
7	513B2F61AA721884	644A06EDAAE0E30	61	2B27ED24DF96A144	F856E677DD8AED2F
8	289D97B0D5390C42	A4872D29812FE4AD	62	9593F6926FCB50A2	CF26FF30F3EF9168
9	144ECBD86A9C8621	FA11D21C66F42E6D	63	CAC9FB4937E5A851	4C39F0126CE3E4A6
10	0A2765EC354E4310	7295633C0CC1F5E5	64	6564FDA49BF2D428	D81E4D01843425B9
11	8513B2F61AA72188	F966B570E414FF2B	65	32B27ED24DF96A14	5896212FF7C72BA2
12	4289D97B0D5390C4	30CD701FC81D63E	66	99593F6926FCB50A	88D25A923F726FD2
13	A144ECBD86A9C862	185D072416686FC6	67	CCAC9FB4937E5A85	9713344CB0DDF24F
14	50A2765EC354E431	6DDE6FE348EAECD	68	66564FDA49BF2D42	9C2ABFB2E7F98A7B
15	A8513B2F61AA7218	026E88801619E226	69	332B27ED24DF96A1	D2FA32F9FE8FD37E
16	D4289D97B0D5390C	E4B4AE72BEFDBBD	70	199593F6926FCB50	D4A68E1B1D805059
17	6A144ECBD86A9C86	35BEE43D5B1CB9E1	71	0CCAC9FB4937E5A8	3FCBA5E1A1C8C99C
18	B50A2765EC354E43	4F9119757787F208	72	066564FDA49BF2D4	71CCFF41E8287B91
19	5A8513B2F61AA721	961C56C4698D6083	73	0332B27ED24DF96A	6A23CDD8A30FA34F
20	2D4289D97B0D5390	D947ECA3BB620F4	74	0199593F6926FCB5	EB58FCABE8A75292
21	96A144ECBD86A9C8	A4CD6A6A5EE09B75	75	80CCAC9FB4937E5A	D456CDE60E5DA303
22	CB50A2765EC354E4	E1224626CF82E262	76	C066564FDA49BF2D	BA7B317F952765F6
23	E5A8513B2F61AA72	6B120E5CF396D87C	77	60332B27ED24DF96	ED2CF1F8BD5AD553
24	F2D4289D97B0D539	004CBF8EDA35E811	78	30199593F6926FCB	6CF1CE57460E4CEF
25	F96A144ECBD86A9C	5226B70B90C2F9FC	79	180CCAC9FB4937E5	DB878EBE47026DA3
26	FCB50A2765EC354E	9450F2B81953ECDC	80	8C066564FDA49BF2	98FB38E94EC48D34
27	7E5A8513B2F61AA7	742B75D3CAAF9028	81	C60332B27ED24DF9	534D43A01D91CC5F
28	BF2D4289D97B0D53	E9468E7D1C0DA7B6	82	630199593F6926FC	2DCABFBB3DFD18B1
29	DF96A144ECBD86A9	EB8F69C6C0DAD125	83	3180CCAC9FB4937E	1CC1D80A00827F3D
30	6FCB50A2765EC354	0EAF784332661FDE	84	98C066564FDA49BF	236B2DECA6D105E8
31	37E5A8513B2F61AA	DC520903F808115D	85	4C60332B27ED24DF	5A6F6A3996D5C89C
32	9BF2D4289D97B0D5	ECAFB0C3BB6A8500	86	A630199593F6926F	BE447EC77233D184
33	4DF96A144ECBD86A	A46123A535318742	87	53180CCAC9FB4937	C002795B4B9DCD75
34	26FCB50A2765EC35	87D9534BE0FA97CB	88	298C066564FDA49B	D197A9365263AD5D
35	937E5A8513B2F61A	AA0BA6BF24368348	89	14C60332B27ED24D	B4D2CE8CD72F97F0
36	49BF2D4289D97B0D	86D34465954FD305	90	8A630199593F6926	8DDC5B9D77745833
37	24DF96A144ECBD86	94118C86A36D2563	91	C53180CCAC9FB493	716C7451D07C03A1
38	926FCB50A2765EC3	2ACB9514D255A1B6	92	6298C066564FDA49	D92274832B6881A8
39	4937E5A8513B2F61	22949C6DFC860D9D	93	B14C60332B27ED24	2F9DC327A02C8097
40	A49BF2D4289D97B0	B8B784E277583EBF	94	58A630199593F692	B6254FD76D321E99
41	D24DF96A144ECBD8	8F1932E277433FFE	95	2C53180CCAC9FB49	83DA62C359DA0058
42	6926FCB50A2765EC	248465829EC04BC3	96	96298C066564FDA4	2BE7347D54A9E80B
43	B4937E5A8513B2F6	DE34E3300213440D	97	4B14C60332B27ED2	A3EE02586A919B73
44	DA49BF2D4289D97B	72E12E60002EEFD5	98	A58A630199593F69	C2D74AE3AC5BED17
45	ED24DF96A144ECBD	D27AD29B54AB5C97	99	D2C53180CCAC9FB4	2C4191BA3885DABF
46	F6926FCB50A2765E	6DA4E0089C79F9EA	100	E96298C066564FDA	16CBCDB20D0D0B91
47	FB4937E5A8513B2F	59110C59A02222B8	101	F4B14C60332B27ED	E1AA5F40B133D0BC
48	FDA49BF2D4289D97	0D0AD817AE17696B	102	FA58A630199593F6	638EC8AB4C214D3C
49	7ED24DF96A144ECB	AC0B5B6286700323	103	FD2C53180CCAC9FB	18B1DF26A53AA4E4
50	3F6926FCB50A2765	DC18101491245F29	104	FE96298C066564FD	BA2AEF298D38D180
51	9FB4937E5A8513B2	387FFFA62450EFA8	105	7F4B14C60332B27E	53A3B0B2430DA6BE
52	4FDA49BF2D4289D9	23297FE8AF446504	106	BFA58A630199593F	CD574A6974B7019D
53	27ED24DF96A144EC	6ED25B26D999B746	107	5FD2C53180CCAC9F	68B2CCB0E54AC490

IDX	Yi	Si	IDX	Yi	Si
108	2FE96298C066564F	A8280AD90D2E7181	165	DAC177C79A6B8D17	6A33CF32D1C63BBD
109	17F4B14C60332B27	BAF7EE3A8827BD76	166	6D60BBE3CD35C68B	61170E040AD89361
110	8BFA58A630199593	0FC3EDE2C02DA037	167	B6B05DF1E69AE345	4AF921E2D721B16B
111	45FD2C53180CCAC9	491C719EFAA1F443	168	DB582EF8F34D71A2	F89FE95DCB052AFC
112	A2FE96298C066564	4AC99D5A19042119	169	EDAC177C79A6B8D1	CC9EA9F2C4330DD1
113	D17F4B14C60332B2	BF4FBABC8548B9F0	170	76D60BBE3CD35C68	A9D6D5E9E13CF811
114	68BFA58A63019959	EA6694CF73F8856C	171	BB6B05DF1E69AE34	9DBCE72B69FD84E6
115	345FD2C53180CCAC	9CB8134695A74344	172	DDB582EF8F34D71A	EE15A0361E2248F6
116	1A2FE96298C06656	C040E70839F43006	173	6EDAC177C79A6B8D	534DE9E6C902400C
117	8D17F4B14C60332B	9F6EA277C3D76D67	174	376D60BBE3CD35C6	AA4BB10FF9FF5C10
118	C68BFA58A6301995	81123AA6AF406234	175	1BB6B05DF1E69AE3	E8CEF33BB7712A7C
119	E345FD2C53180CCA	3C690650CA9887BE	176	0DDB582EF8F34D71	68A2C2A26D3CA6BC
120	71A2FE96298C0665	6315DEFD44A0284A	177	06EDAC177C79A6B8	72B5F77FDF00DB17
121	B8D17F4B14C60332	73B3B60D9D138971	178	8376D60BBE3CD35C	DDD161EB0DB2B865
122	5C68BFA58A630199	FCACFAD07DDAB574	179	41BB6B05DF1E69AE	B1C51A123A7CCCB7
123	AE345FD2C53180CC	291196182886D0DE	180	A0DDB582EF8F34D7	E16B738A3650E8E8
124	D71A2FE96298C066	63F2990B575EE3E7	181	D06EDAC177C79A6B	86EDB365A4D67A54
125	6B8D17F4B14C6033	D72EE8BD7C9CE14C	182	68376D60BBE3CD35	7737692EF95ED809
126	35C68BFA58A63019	9C2BB1B429AB2E96	183	B41BB6B05DF1E69A	CC056A56FBFE4473
127	9AE345FD2C53180C	93E2CD1EC05C7432	184	5A0DDB582EF8F34D	8BA86DFB0052699C
128	4D71A2FE96298C06	563CD62EA1CF3454	185	AD06EDAC177C79A6	121511C468EF5269
129	A6B8D17F4B14C603	EB9104AA38FC8B65	186	D68376D60BBE3CD3	378C846FDE3B0F0B
130	D35C68BFA58A6301	BEC7704631705038	187	EB41BB6B05DF1E69	736AA3F76FB4E4F3
131	69AE345FD2C53180	43416C6AF5A4E81B	188	F5A0DDB582EF8F34	3526A80721C6CB19
132	34D71A2FE96298C0	D3F66D626B6291EB	189	FAD06EDAC177C79A	69D53B2B61404BB6
133	9A6B8D17F4B14C60	30FEDC61413B15E5	190	7D68376D60BBE3CD	F1951371CA434F49
134	CD35C68BFA58A630	99BD7841295C3D54	191	BEB41BB6B05DF1E6	AD11B3BE2C12B1E2
135	E69AE345FD2C5318	C2AD3262B04A908D	192	5F5A0DDB582EF8F3	E1D0C0ECB7080573
136	F34D71A2FE96298C	04F54BE664C3B5A9	193	AFAD06EDAC177C79	AF1364E021E80A31
137	79A6B8D17F4B14C6	CCB00866CDE93D0B	194	57D68376D60BBE3C	435117C42D15980B
138	3CD35C68BFA58A63	2205358E41E93312	195	ABEB41BB6B05DF1E	4D04185BF70E35F9
139	1E69AE345FD2C531	500F7F5703872DFC	196	55F5A0DDB582EF8F	E1E3B799DDE1A3DB
140	8F34D71A2FE96298	25DAC8DC815B6C5B	197	AAFAD06EDAC177C7	8E30A329D9BCB487
141	C79A6B8D17F4B14C	E440258580A9B39C	198	557D68376D60BBE3	82E1BE6063C7AA69
142	E3CD35C68BFA58A6	82006D381535671A	199	2ABEB41BB6B05DF1	F033039DD5D6649A
143	F1E69AE345FD2C53	0230A7BAD98953AD	200	155F5A0DDB582EF8	0B4BF55BB808A368
144	F8F34D71A2FE9629	9A34ACED8CEE98B7	201	0AAFAD06EDAC177C	FE23149F28E2EF2C
145	7C79A6B8D17F4B14	43E65444E28D7BC1	202	0557D68376D60BBE	283DB670B7DFAF2B
146	BE3CD35C68BFA58A	7C3B162582187ADD	203	02ABEB41BB6B05DF	CF9C89F680A99D62
147	DF1E69AE345FD2C5	B49FB4D82CCCD061	204	8155F5A0DDB582EF	E1B488F6D0269E7C
148	EF8F34D71A2FE962	42D1054F2D85AA94	205	40AAFAD06EDAC177	9C5BB288B626DE4D
149	77C79A6B8D17F4B1	DA4D67ADAADD078E	206	A0557D68376D60BB	9A4C5AF54BAA3B0E
150	BBE3CD35C68BFA58	AB800AB239F4905E	207	502ABEB41BB6B05D	F2ADA59BCFFB9AAB
151	5DF1E69AE345FD2C	2D41DC2CAC99EC8E	208	28155F5A0DDB582E	6E0B132626ACEF11
152	2EF8F34D71A2FE96	6AB0F4ABB76EBE64	209	940AAFAD06EDAC17	1CE26DE15831D630
153	177C79A6B8D17F4B	BA6771D64704C491	210	4A0557D68376D60B	DF4C483BA35FAD3D
154	0BBE3CD35C68BFA5	386B8A7A41194937	211	A502ABEB41BB6B05	71E85CAE44BF3482
155	05DF1E69AE345FD2	16A4B1F10932A8A4	212	528155F5A0DDB582	EBCCAE1494706629
156	82EF8F34D71A2FE9	B5464ADFFD535E43	213	A940AAFAD06EDAC1	666E1666A82515BC
157	C177C79A6B8D17F4	60CA144EE6353890	214	D4A0557D68376D60	B0ED90EA159D5BDF
158	60BBE3CD35C68BFA	FAB076B245C9A692	215	EA502ABEB41BB6B0	540498B374287B73
159	B05DF1E69AE345FD	FA42C96FED74618B	216	F528155F5A0DDB58	24A0E8ED1D922E0A
160	582EF8F34D71A2FE	F1403D4F5BEDCFBF	217	7A940AAFAD06EDAC	07442F68124FACB8
161	AC177C79A6B8D17F	D0048DF3E1042834	218	3D4A0557D68376D6	E394BD2139A9AAD8
162	D60BBE3CD35C68BF	368F58885B220711	219	9EA502ABEB41BB6B	F339E1F4E65C2D05
163	6B05DF1E69AE345F	8D34874F3691DE76	220	4F528155F5A0DDB5	271870FF4232E570
164	B582EF8F34D71A2F	DD8736FE9489ADE6	221	A7A940AAFAD06EDA	77C7B16BF4EF3F53

IDX	Yi	Si	IDX	Yi	Si
222	D3D4A0557D68376D	775B2CA46D0D1768	**279**	CC24575E4B9C0EE9	D8A04F040D696013
223	E9EA502ABEB41BB6	1A4DCACBF7A484A4	**280**	E6122BAF25CE0774	7E96BFBA8496562D
224	74F528155F5A0DDB	F9B32033DBC6E2D3	**281**	730915D792E703BA	298BFAF05150461E
225	BA7A940AAFAD06ED	6A3D736249E0AF9D	**282**	39848AEBC97381DD	3092A122A623AB76
226	DD3D4A0557D68376	30127904F375C91F	**283**	1CC24575E4B9C0EE	F43F058869127867
227	EE9EA502ABEB41BB	EC681949C8F07F6E	**284**	0E6122BAF25CE077	111B1EE2C95B2F52
228	774F528155F5A0DD	8B5AE5C3966012FD	**285**	8730915D792E703B	490E08893633DCAA
229	3BA7A940AAFAD06E	B9C334FA58B743B9	**286**	439848AEBC97381D	66D559D99FF0FD65
230	1DD3D4A0557D6837	1E9AFEA09F29D67F	**287**	A1CC24575E4B9C0E	827A5665111507C8
231	0EE9EA502ABEB41B	41BFD5F3B03CF11C	**288**	D0E6122BAF25CE07	EABA7088F3696CB2
232	0774F528155F5A0D	6A987AE72C5D69C5	**289**	E8730915D792E703	CE6AC5C3F7097F12
233	03BA7A940AAFAD06	A1E57D08E23A32D4	**290**	F439848AEBC97381	9B4BC274F85E62DB
234	81DD3D4A0557D683	48F222BFCF94F0D4	**291**	7A1CC24575E4B9C0	4B710B92B7CC87C3
235	C0EE9EA502ABEB41	64568E8358F5E4D9	**292**	BD0E6122BAF25CE0	0228D63C98A181AB
236	E0774F528155F5A0	8928C3D51AA25B51	**293**	DE8730915D792E70	85526AC0FE51BB79
237	703BA7A940AAFAD0	74414DDFE7E02A5F	**294**	6F439848AEBC9738	AAF060967678D804
238	381DD3D4A0557D68	F33C920DEE5F9039	**295**	B7A1CC24575E4B9C	4EC6FB81281AE192
239	9C0EE9EA502ABEB4	1B982C38DAA12008	**296**	DBD0E6122BAF25CE	6F06F3E08C6461A7
240	CE0774F528155F5A	81D51A86CDDB0288	**297**	6DE8730915D792E7	2CBCB783A03056BA
241	E703BA7A940AAFAD	D92E2DF6EBAF05DB	**298**	36F439848AEBC973	D606309C27138BE2
242	7381DD3D4A0557D6	6F4379C3BB7398A1	**299**	9B7A1CC24575E4B9	55A81A7E22B070FE
243	B9C0EE9EA502ABEB	01F9D1B4C301F1A7	**300**	CDBD0E6122BAF25C	E79C8B447C580777
244	5CE0774F528155F5	7DDABA1D0AB183DF	**301**	66DE8730915D792E	86784C57CF056BCA
245	2E703BA7A940AAFA	4337EB0FB2A2DFF3	**302**	B36F439848AEBC97	6DD168F5F33E3FA5
246	97381DD3D4A0557D	36AEC0EC4B3AED9A	**303**	59B7A1CC24575E4B	7BEE1C6FC44CEFC5
247	4B9C0EE9EA502ABE	1A89A7B15D744AB5	**304**	2CDBD0E6122BAF25	42C41485739299CB
248	25CE0774F528155F	E755628C77BD33FE	**305**	166DE8730915D792	8007772D6D59C921
249	92E703BA7A940AAF	874FD3C52C8AD8EA	**306**	0B36F439848AEBC9	8886487A92DA7DBF
250	C97381DD3D4A0557	ED8D457CEAA088F8	**307**	859B7A1CC24575E4	14B5C6AB76BE32C4
251	E4B9C0EE9EA502AB	3C1953EDC8051867	**308**	C2CDBD0E6122BAF2	B0D286ECF0BDBE03
252	F25CE0774F528155	431A9BC66B3CFB04	**309**	E166DE8730915D79	471A3D7F5C7FEE46
253	792E703BA7A940AA	138D9B61D5B6B1BA	**310**	70B36F439848AEBC	FFAF6000D38F0D49
254	BC97381DD3D4A055	C87A601C630F873F	**311**	B859B7A1CC24575E	381103BF205EA791
255	5E4B9C0EE9EA502A	C9A9C736138DA8FF	**312**	DC2CDBD0E6122BAF	9743B94982BFACD3
256	AF25CE0774F52815	36507987564FAA6B	**313**	EE166DE8730915D7	029D812AD2E31381
257	D792E703BA7A940A	74C2C7358F717F12	**314**	F70B36F439848AEB	E661631CC8AD9760
258	EBC97381DD3D4A05	F264A4EF6E52B45D	**315**	7B859B7A1CC24575	B816FC762CAC3DD1
259	75E4B9C0EE9EA502	71B4B053DD732F26	**316**	3DC2CDBD0E6122BA	4EBBA51F37C858B3
260	BAF25CE0774F5281	1AC50F8690DC2B8F	**317**	1EE166DE8730915D	03084F565311BEB9
261	5D792E703BA7A940	0F8AE9D922E7E68F	**318**	0F70B36F439848AE	E2C8E1287DC35047
262	AEBC97381DD3D4A0	6675B3DF7A7A4146	**319**	87B859B7A1CC2457	530072F65A54151D
263	575E4B9C0EE9EA50	7868F78DEFD4AA1F	**320**	C3DC2CDBD0E6122B	36811CB39469F8F7
264	2BAF25CE0774F528	EEEC0599D150EBF8	**321**	E1EE166DE8730915	7CB6DE5133FC5607
265	15D792E703BA7A94	3E7C938758FE3E09	**322**	F0F70B36F439848A	C77FA77DE867D957
266	8AEBC97381DD3D4A	8E7C0C461653797A	**323**	F87B859B7A1CC245	FD636166FD8573A6
267	4575E4B9C0EE9EA5	DF8D791AFA2A496A	**324**	FC3DC2CDBD0E6122	A5D57D3BB3FD8823
268	22BAF25CE0774F52	EBEA31AA27EDB851	**325**	FE1EE166DE873091	9A5F23044BDE587A
269	915D792E703BA7A9	55C4090681478AA3	**326**	FF0F70B36F439848	9D5EF4E3DDE4B719
270	48AEBC97381DD3D4	409F3435E613760E	**327**	FF87B859B7A1CC24	F05F3D987134CE20
271	24575E4B9C0EE9EA	32DF5BE59C01114D	**328**	7FC3DC2CDBD0E612	786DBC586D7B7055
272	122BAF25CE0774F5	94C2BA694A2D2B2D	**329**	3FE1EE166DE87309	8562853E59F4630D
273	0915D792E703BA7A	C1C851D3BF4A7DCC	**330**	1FF0F70B36F43984	54645B0AC1EDAF24
274	848AEBC97381DD3D	C02D90ABACC049EF	**331**	0FF87B859B7A1CC2	576F363E38C97833
275	C24575E4B9C0EE9E	316CF0214C671866	**332**	07FC3DC2CDBD0E61	D1D69B5FA1EBF3CE
276	6122BAF25CE0774F	CD3C6EA472279319	**333**	83FE1EE166DE8730	9E003F100DB0C1BD
277	30915D792E703BA7	0CE7447A3B6618A0	**334**	C1FF0F70B36F4398	AF20B610ED6F0348
278	9848AEBC97381DD3	5FB1895F5E29D445	**335**	E0FF87B859B7A1CC	63043789D0CCD77D

IDX	Yi	Si	IDX	Yi	Si
336	F07FC3DC2CDBD0E6	C6741A48F8857CBB	393	7CF16E52099D1F78	A109CF8BC1E2EDD1
337	783FE1EE166DE873	E21EB839DC2D827C	394	3E78B72904CE8FBC	69A1D5947D786C7F
338	BC1FF0F70B36F439	8C0A748325CEFEED	395	9F3C5B94826747DE	848FC9436B45ECDC
339	DE0FF87B859B7A1C	F7BD795F96A356B2	396	CF9E2DCA4133A3EF	EADF66D7550A89E5
340	EF07FC3DC2CDBD0E	A2BFD6402EB708D9	397	67CF16E52099D1F7	DB0C9FD132684BDB
341	F783FE1EE166DE87	2735A4BE0E6C2E3D	398	B3E78B72904CE8FB	3C4FD72CDE46A4F1
342	FBC1FF0F70B36F43	A48CC1C29C74FF1D	399	D9F3C5B94826747D	9C4136C457A91E44
343	7DE0FF87B859B7A1	165AEFA1D2240291	400	6CF9E2DCA4133A3E	285485069A599DEF
344	3EF07FC3DC2CDBD0	0637DC7386BDC409	401	367CF16E52099D1F	830C2A2FE43333B2
345	1F783FE1EE166DE8	10ACBA8716459537	402	1B3E78B72904CE8F	4C9D75160419B58D
346	8FBC1FF0F70B36F4	F108C5502A588274	403	8D9F3C5B94826747	EAB818E0D8329A6F
347	47DE0FF87B859B7A	A822BF6C6B9152C3	404	46CF9E2DCA4133A3	4B4C3E7503C9A1B9
348	A3EF07FC3DC2CDBD	595407E627B6CDB2	405	A367CF16E52099D1	909B65ED9B498507
349	D1F783FE1EE166DE	CC4DAB7AE2BF9649	406	51B3E78B72904CE8	7ADE4A8822CDA512
350	E8FBC1FF0F70B36F	F1D11105E36491B0	407	A8D9F3C5B9482674	5912FA785DE6FB19
351	747DE0FF87B859B7	B7758AB3ED029350	408	546CF9E2DCA4133A	9BF2D2D127FC7A9B
352	3A3EF07FC3DC2CDB	29FBD97225E7BFB1	409	2A367CF16E52099D	69B1CE6FF18E3F49
353	9D1F783FE1EE166D	5179786434BC2E52	410	951B3E78B72904CE	D4BDBB280BD556E3
354	CE8FBC1FF0F70B36	BBB2ADCB4A40A306	411	4A8D9F3C5B948267	E0EF8BA92FAAEBAE
355	6747DE0FF87B859B	C118CB3389D08447	412	2546CF9E2DCA4133	75182AE174CD8FEE
356	33A3EF07FC3DC2CD	771CB88F4E3B7623	413	12A367CF16E52099	A1290C277F9E5751
357	99D1F783FE1EE166	6EB1CF03FEFDB7B2	414	8951B3E78B72904C	CD8298B0EAFD09B1
358	4CE8FBC1FF0F70B3	9BF5B0ACBBE0EC1A	415	C4A8D9F3C5B94826	71C074DC28210041
359	26747DE0FF87B859	1BA4C19873276163	416	E2546CF9E2DCA413	A2D863DA7FB806D0
360	133A3EF07FC3DC2C	1DCC8AC4F39FAF41	417	712A367CF16E5209	79CD0C5F90D0A169
361	099D1F783FE1EE16	39DB8927535C41FA	418	38951B3E78B72904	25A4946F6C4E7D2B
362	04CE8FBC1FF0F70B	D87011747DF7FB42	419	1C4A8D9F3C5B9482	8ED2016051EB0CFD
363	826747DE0FF87B85	DCB4DE84C69594F1	420	8E2546CF9E2DCA41	E9D48E7A05571B40
364	4133A3EF07FC3DC2	99C545C88EF47685	421	C712A367CF16E520	395A8C05C75CD38D
365	2099D1F783FE1EE1	FAD32A4FB2F3FA9B	422	638951B3E78B7290	4B4411056B9FECF2
366	904CE8FBC1FF0F70	6AE85D38E1E50954	423	B1C4A8D9F3C5B948	DD27FD92D2CBE434
367	4826747DE0FF87B8	4F067D6FDB5B1C1E	424	D8E2546CF9E2DCA4	5F7EB736D6CAE1F3
368	A4133A3EF07FC3DC	A003D15D6D7ABD26	425	6C712A367CF16E52	2F09E0BE088B2413
369	52099D1F783FE1EE	EE78D2BB24474D99	426	B638951B3E78B729	401893101BFE3E4B
370	2904CE8FBC1FF0F7	28936114F0DFBA62	427	5B1C4A8D9F3C5B94	4D0AEE32BBC2E786
371	94826747DE0FF87B	9F9681F1FD6A0B34	428	AD8E2546CF9E2DCA	734E8F8A412BA851
372	CA4133A3EF07FC3D	ABF295684A4E58F0	429	56C712A367CF16E5	1911D4F21DAF0579
373	E52099D1F783FE1E	42F0B7D409477EAB	430	AB638951B3E78B72	3B1C3A5F644FE893
374	72904CE8FBC1FF0F	69D8877DA5A592EF	431	D5B1C4A8D9F3C5B9	21DDED94D0384190
375	B94826747DE0FF87	E7DAAED06C67689A	432	EAD8E2546CF9E2DC	06E6FCD25C3B1D7A
376	DCA4133A3EF07FC3	4955D08C6615F769	433	756C712A367CF16E	5D5040522A238852
377	6E52099D1F783FE1	06E2B1D074A1C41A	434	3AB638951B3E78B7	0E9DE483B4D6F704
378	B72904CE8FBC1FF0	806C15A6FAD53A38	435	1D5B1C4A8D9F3C5B	D592669CE0CE696F
379	5B94826747DE0FF8	FC8FEA02B29CF0F7	436	8EAD8E2546CF9E2D	D6545D61788AF665
380	2DCA4133A3EF07FC	B184B9330F882EF3	437	4756C712A367CF16	19ED4F6846242BC8
381	16E52099D1F783FE	BD15F7D7E272FE2C	438	23AB638951B3E78B	2D29DD3F34DAD458
382	8B72904CE8FBC1FF	55A7176D3BDA0B59	439	91D5B1C4A8D9F3C5	85261654938E820F
383	C5B94826747DE0FF	810D487D1BFC8B2C	440	C8EAD8E2546CF9E2	EB1BF42124B757F5
384	E2DCA4133A3EF07F	0686DECADC9460CD	441	64756C712A367CF1	5D669DA29C5F9BBD
385	F16E52099D1F783F	1019FEF9B1F024DA	442	323AB638951B3E78	49EBCAC048C9C682
386	78B72904CE8FBC1F	BFAA8FD42C2CF6FF	443	191D5B1C4A8D9F3C	D6652FE9FB11D31D
387	3C5B94826747DE0F	36C516366C187063	444	8C8EAD8E2546CF9E	FF1EB9266BD80154
388	9E2DCA4133A3EF07	047BE4BEA0E1DF99	445	464756C712A367CF	ADCDDD35730C2FA3
389	CF16E52099D1F783	39CE3A7F5690500C	446	2323AB638951B3E7	2ABAD2BC986E95A0
390	E78B72904CE8FBC1	4C77EB798E215ACA	447	1191D5B1C4A8D9F3	8EF98359B73E7889
391	F3C5B94826747DE0	8EE3E886511289C3	448	88C8EAD8E2546CF9	278B0C4E5A8BBEB7
392	F9E2DCA4133A3EF0	21DAC59D72484EA2	449	4464756C712A367C	BBCEB1F1C69577AC

IDX	Yi	Si	IDX	Yi	Si
450	22323AB638951B3E	618D2527A32345DC	**491**	0D5390C42011191D	BA1E81F8D05EE382
451	11191D5B1C4A8D9F	A4C59B1A4A09D921	**492**	86A9C86210088C8E	3B929AB864E203A5
452	088C8EAD8E2546CF	925AB5C02358A0F6	**493**	C354E43108044647	8BC5B29A53C632D2
453	04464756C712A367	6BF809C1B711390E	**494**	61AA721884022323	C24A9389D3B95E67
454	022323AB638951B3	725E055955ED2941	**495**	B0D5390C42011191	90ED5949E7E1A96E
455	011191D5B1C4A8D9	5B8AC4B9908DDA47	**496**	D86A9C86210088C8	7F7C860E5E92BD8F
456	0088C8EAD8E2546C	D5B77C771A4BCF75	**497**	EC354E4310804464	E6168162134B1948
457	804464756C712A36	6A39A4D0F42AE266	**498**	F61AA72188402232	B8ABA5344AE3FE81
458	4022323AB638951B	304FA547BF996525	**499**	7B0D5390C4201119	999B48295220BD2B
459	2011191D5B1C4A8D	A54FAB8637E3A1D0	**500**	BD86A9C86210088C	4E78E5FF4532E346
460	10088C8EAD8E2546	647BBF53DEAE5012	**501**	5EC354E431080446	64FB887329CB60A0
461	0804464756C712A3	6AD83BBC86CF348E	**502**	2F61AA7218840223	21E399F53F697EE8
462	84022323AB638951	B748279C022F8B5E	**503**	97B0D5390C420111	291A39BCD3F3F1A9
463	42011191D5B1C4A8	B58C8C6F89DC4FB3	**504**	CBD86A9C86210088	D6FA8C5105525BAE
464	210088C8EAD8E254	E17D63E28A3CC0B9	**505**	65EC354E43108044	9F340FC9DE50877A
465	10804464756C712A	C10D404BFEE92944	**506**	B2F61AA721884022	565847F04A335555
466	884022323AB63895	0D6FE7511C5971BF	**507**	D97B0D5390C42011	6D3E61E3797FE2D9
467	C42011191D5B1C4A	8D7C5D774D7328E4	**508**	ECBD86A9C8621008	2C19CF924184314A
468	6210088C8EAD8E25	D58DAF10EE827EB1	**509**	765EC354E4310804	817BEFE7A5A6A716
469	310804464756C712	E03242B83E2CDE57	**510**	3B2F61AA72188402	CC925EBEEB4E39CC
470	1884022323AB6389	EB74038EA1033261			
471	0C42011191D5B1C4	55CF23BA998BE5A0			
472	86210088C8EAD8E2	219DFBE447BFDB9B			
473	4310804464756C71	0D7A7021CB3142C4			
474	21884022323AB638	E6F62B591D6A303D			
475	90C42011191D5B1C	8573523FF26AE067			
476	C86210088C8EAD8E	F1EEC4D6047D18C8			
477	E4310804464756C7	4A681A9543151BFF			
478	721884022323AB63	41C93B16BA1BBC28			
479	390C42011191D5B1	60F506802A5B6AAA			
480	9C86210088C8EAD8	6A5C0C9C7FDCA2E2			
481	4E4310804464756C	DA78574487CA5118			
482	A721884022323AB6	5F1BAC9234589E24			
483	5390C42011191D5B	248B708A4352F25C			
484	A9C86210088C8EAD	3933AA202705408F			
485	54E4310804464756	624489740BCCFDD4			
486	AA721884022323AB	2CB721BA97C88B66			
487	D5390C42011191D5	715A0CEC3A15AFB2			
488	6A9C86210088C8EA	355D06ABD23BC8EC			
489	354E431080446475	B646CFA2CB1FE176			
490	1AA721884022323A	FBEA77DAB1E38B46			

Appendix A.3 – Linear Span Frequency Computations for SIMON 32/64

(over 1M runs for 32 vectors)

LS	Freq.	Average	Probability
110	0	0.00	0.00000
111	0	0.00	0.00000
112	0	0.00	0.00000
113	1	0.03	0.00000
114	0	0.00	0.00000
115	0	0.00	0.00000
116	1	0.03	0.00000
117	6	0.19	0.00000
118	33	1.03	0.00000
119	105	3.28	0.00000
120	491	15.34	0.00002
121	1983	61.97	0.00006
122	7824	244.50	0.00024
123	31290	977.81	0.00098
124	125378	3918.06	0.00392
125	500432	15638.50	0.01564
126	1999772	62492.88	0.06249
127	7998286	249946.44	0.24995
128	15998460	499951.88	0.49995
129	4001721	125053.78	0.12505
130	1001017	31281.78	0.03128
131	249855	7807.97	0.00781
132	62466	1952.06	0.00195
133	15592	487.25	0.00049
134	3975	124.22	0.00012
135	982	30.69	0.00003
136	251	7.84	0.00001
137	57	1.78	0.00000
138	17	0.53	0.00000
139	4	0.13	0.00000
140	1	0.03	0.00000
141	0	0.00	0.00000
142	0	0.00	0.00000
143	0	0.00	0.00000
144	0	0.00	0.00000
145	0	0.00	0.00000
146	0	0.00	0.00000
147	0	0.00	0.00000
148	0	0.00	0.00000
149	0	0.00	0.00000

Appendix A.4 – Correlation and Linear Span Frequency Computations

for SIMON 64/128