# A Shifted Box with Variable Stage Feed Back Function

**Prof. DR. Imad Hussein Al-Husaini -** Iraqi commission for Computer and Informatics

**Dr. Jane Jaleel Stephan -** Iraqi commission for Computer and Informatics

**Ahmed Hussein Ali -** Baghdad College for Economic Sciences

**Abstract.**

The objective of this research is to make the **S**hifted **B**ox (***SB***) which can be defined as two-dimensional matrix of (N*M), unlimited and random content by making the values of rows and columns, changing at a time, before shifting using along **L**inear **F**eedback **S**hift **R**egister (**LFSR)** with **I**nitial **F**ixed **R**andom **T**able **(IFRT)** to increased the complexity.

## 1. Introduction

The goal of cryptography is to insure confidentiality and authenticity of information. This is performed by a public ciphering function that involves a secret key in a certain mode of operation. The union of the ciphering function and the mode of operation constitute the cryptographic system. The three actors of a cryptographic system are the sender, the receiver and the adversary. In a symmetric ciphering system, the plain text is encrypted into a cryptogram using a secret key shared by the sender and the receiver. The goal of the adversary is to get information about the plain text. This can be done by recovering the secret key, but there may exist other ways, in particular to retrieve partial information [1].

A cryptographic system is called unconditionally secure if the adversary has no better strategy than choosing the plain text at random. In the practical world, the security level is assessed taking into account the protection which is supposed to be insured, the data and computation power at the disposal of the adversary. Knowledge of only the cryptogram is not sufficient to decrypt. In the so called chosen plain text attack, the adversary is supposed to know the cryptograms that correspond to messages of his choice. A cryptographic system is called semantically secure if the adversary cannot distinguish the cryptogram from a pure random sequence with reasonable amount of time and computation power. Finally, it is admitted by the cryptographic community that a ciphering function is secure if the adversary has no better strategy than trying all the possible secret keys in a chosen plain text attack [2].

## 2. Stream Cipher Structure

A stream cipher is a symmetric key cipher where plain text bits are combined with a pseudorandom cipher bit stream (key stream), typically by an exclusive-or (XOR) operation as show in figure 1. In a stream cipher the plain text digits are encrypted one at a time, and the transformation of successive digits varies during the encryption. An alternative name is a state cipher, as the encryption of each digit is dependent on the current state. In practice, the digits are typically single bits or bytes.

Stream ciphers represent a different approach to symmetric encryption from block ciphers. Block ciphers operate on large blocks of digits with a fixed, unvarying transformation. This distinction is not always clear-cut: in some modes of operation, a block cipher primitive is used in such a way that it acts effectively as a stream cipher [3].

Stream ciphers typically are executed at a higher speed than block ciphers and have lower hardware complexity.

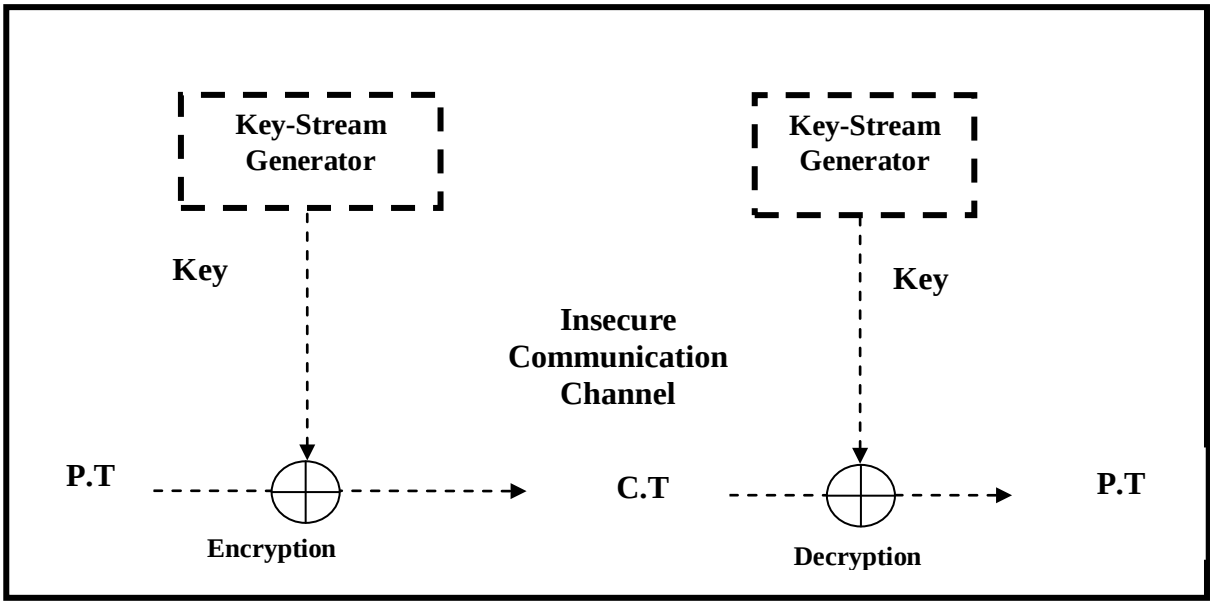



**Figure 1 Stream Cipher Structure**

There is a vast body of theoretical knowledge on stream ciphers, and various design principles for stream ciphers have been proposed and extensively analyzed. However, there are relatively few fully-specified stream cipher algorithms in the open literature. This unfortunate state of affairs can partially be explained by the fact that most stream ciphers used in practice tend to be proprietary and confidential.

In contrast, stream ciphers process plaintext in blocks as small as a single bit, and the encryption function may vary as plaintext is processed; thus stream ciphers are said to have memory. They are sometimes called state ciphers since encryption depends on not only the key and plaintext, but also on the current state [4].

### 3. Variable Position Generator

In this paper we will review a proposed non-*linear* shifted box with variable stage feed back function to determine a position of the second stage for the feedback function , figure 2 view the flowchart of the propose paper.

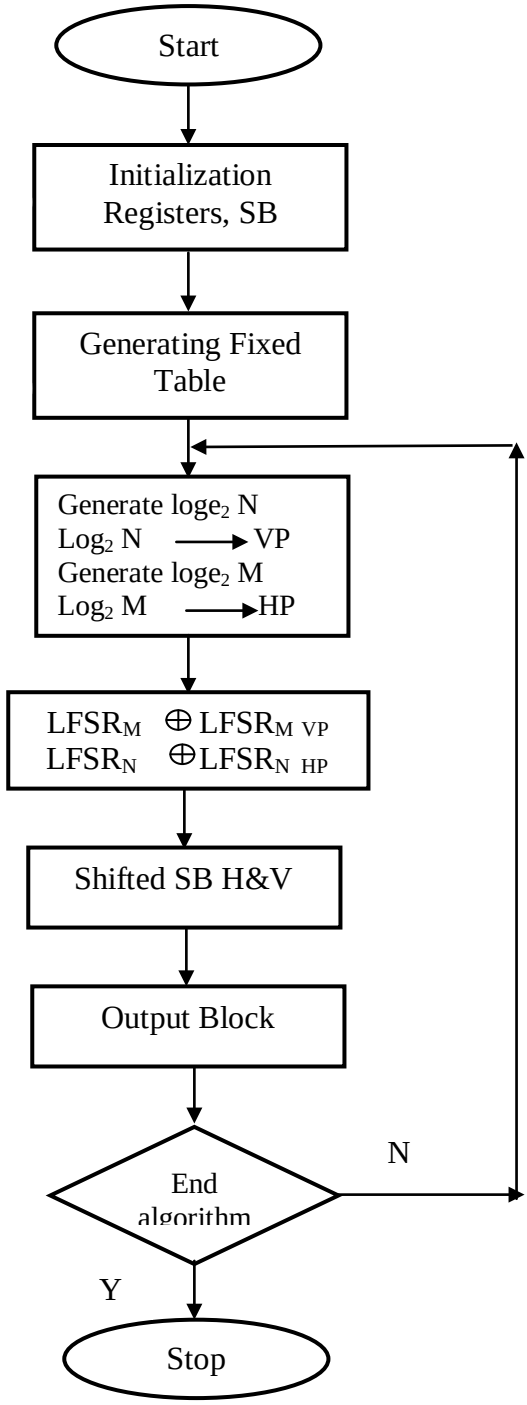



**Figure 2** shifted box with variable stage feed back function flowchart

We use an LFSR as a generator (see figure 3) to generate a group of bits For example we select an LFSR of length 42, the tapping stages can be defined as 42, 7, 4, and 3 which produce a maximum period [5].

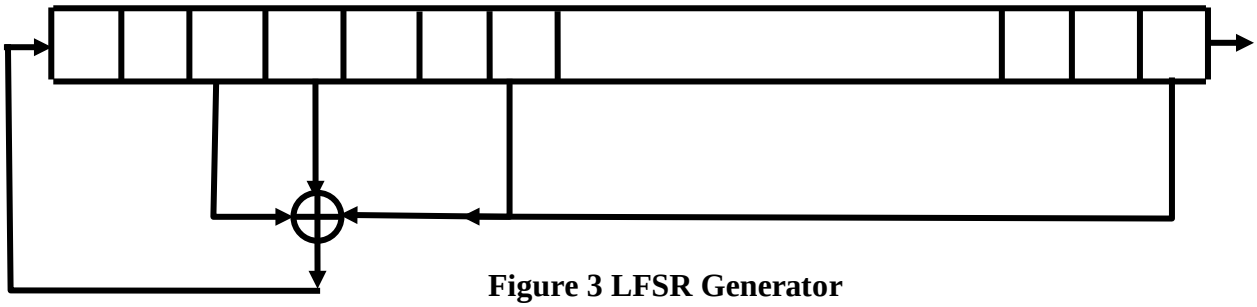


**Figure 3 LFSR Generator**

The initialization process is depending on the application which will produce a random sequence of length 42. To obtain the position of the second stage of the shifted box we must generate $\log_2 M$ for rows LFSR's and $\log_2 N$ for columns LFSR's. For our assumptions', we need to produce three bits to obtain the position of the second stage of vertical LFSR , and 4 bits to obtain the second stage of the horizontal LFSR.

### 3.1. Shifted Box shifting

**N**on-**L**inear **S**hifted **B**ox (**NLSB**) can be defined as two dimensional matrix of (N*M), where **N** is a number of rows and **M** is the number of columns, each row can be considered as a **L**inear **F**eedback **S**hift **R**egister (**LFSR**) and each column can be considered as an **LFSR** in the same time. All rows have the same predefined feedback function defined for a predetermined two tapping stages as show in figure 4. For example, assuming that we have a shifted box of 8x16 dimensions, in this case we need to generate a number from 0-7 for a vertical LFSR and from 0-15 for the horizontal LFSR, so we need three bits to represent the vertical stage and four bits for the horizontal stage.

| **L F S R -1** |
|---|
| **L F S R -2** |
| **L F S R -3** |
| **.** |
| **.** |
| **.** |
| **.** |
| **L F S R -M** |

**Figure 4. Non-Linear Shifted Box**

In the second hand, all columns have the same predefined feedback function defined for a predetermined two tapping stages as show in figure 5.

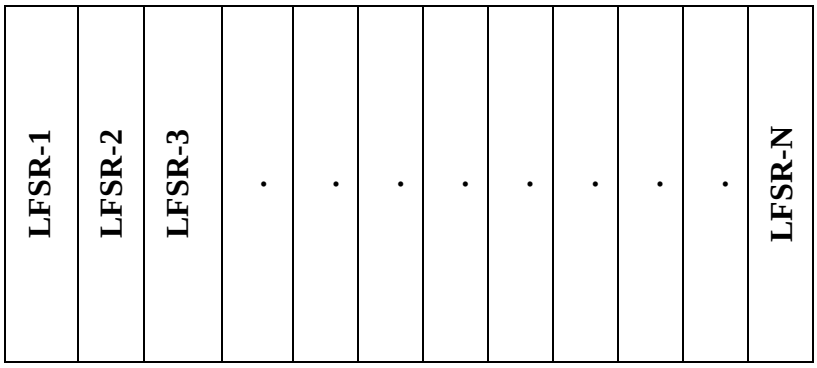



**Figure 5. Non-Linear Shifted Box**

The movement of these registers must be performed for the rows before columns or columns before rows and not for both at the same time. The initialization of the shifted box must be done by a predefined procedure with a sequence of bits of length N*M depending on the application that uses it.

The nonlinearity of the shifted box can be obtained from making the feedback function of these registers depend on a variable tapping stages.

One of the two tapping stages must be the last stage (stage number N for rows and stage number M for columns), so, the variability is by selecting the second stage for the tapping of the feedback function.

The operation of shifting for any row of (N) is the same of operation of all rows of the N in the box because its have the same tapping and the operation of shifting for any column of (M) is the same of operation of all columns of the M in the box because its have the same tapping.

In order to spread the effect of each single bit to others, the position of the bits must be changed for a long period and the permutation function have no calculated period, therefore, another position changing function needed to guaranteed that bits positions are changed periodically, this function besides the permutation function must produce a long period.

### 3.2. Initial Fixed Random Table (IFRT).

**IFRT** can define as two-dimensional matrix NxM that contain for example 8 rows x 16 columns and 128 cells felled with a whole number generated randomly that begin from 0 to 127 as show in figure 6. To represent any number in IFR, it mast have 7 bits that get it's from the output of the LFSR after converted its using ASCII code. IFRT is constant for each application.

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 69 | 82 | 105 | 127 | 76 | 2 | 56 | 40 | ***80*** | 98 | 56 | 22 | 55 | ***20*** | 29 | 30 |
| ***63*** | 3 | 96 | ***103*** | 128 | 50 | 106 | 76 | 4 | 71 | 6 | 45 | 70 | 27 | 81 | 10 |
| 104 | 124 | 5 | 73 | 107 | 105 | 72 | 49 | 114 | 57 | 117 | 90 | 25 | 8 | 116 | 120 |
| 75 | 16 | 74 | 7 | 77 | 86 | 104 | 108 | 58 | 72 | 0 | 23 | 121 | 1 | 59 | 44 |
| 42 | 125 | ***113*** | 94 | 9 | 62 | 18 | 48 | 91 | 109 | 21 | 89 | 68 | 45 | 93 | 24 |
| 83 | 95 | 51 | 126 | 78 | ***11*** | 65 | 87 | 14 | 19 | 28 | 46 | ***110*** | 99 | 115 | 119 |
| 64 | 97 | 84 | 102 | 54 | 114 | 13 | 92 | 17 | 61 | 47 | 75 | 101 | 118 | 68 | 26 |
| 41 | 52 | 85 | 53 | 79 | 66 | 112 | 15 | 111 | 67 | 123 | 88 | 60 | 112 | 100 | 43 |

**Figure 6 Initial Fixed Random Table (IFRT)**

### 3.3. Vertical Shiftin$_G$

The movement of the rows registers must be performed for a number of shifts determined by output of the LFSR, for example convert the first seven bits from the output of the LFSR (0111111) using ASCII code to a whole number (63) read the second 7 bits (1100111) which equal to 103 then the third 7 bits (1110001) which equal to 113 as show in figure 6.The three positions that corresponding these numbers in SB are (101) see figure 6, now convert 101 to a whole number to determine the row in SB that equal to 5 to be tapped in the feedback function as show in figure 7, so all content of row (**M**) number 8 and the content of row number 5 will be Xored correspondingly resulted a new row will be stored at row number 1 after shifting the rows down by 1, the last row will be ignored.

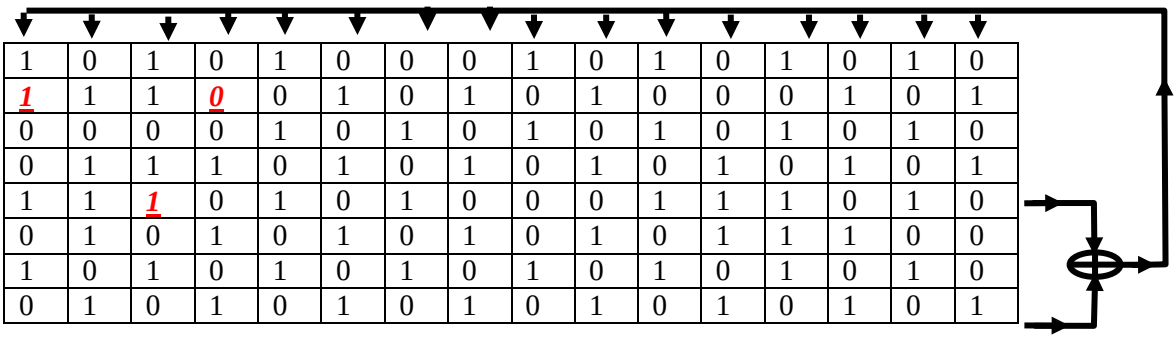
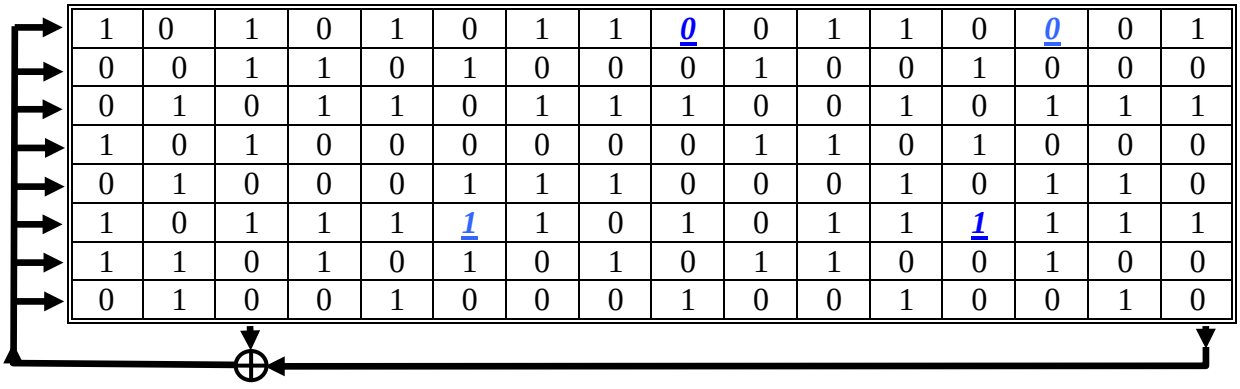



**Figure 7 Vertical shift steps**

### 3.4. Horizontal Shifting

The movement of the columns registers must be performed for a number of shifts determined by output of the LFSR, for example convert the first seven bits from the output of the LFSR (1010000) using ASCII code to a whole number (20) read the

second 7 bits (0001011) which equal to 11 then the third 7 bits (1100000) which equal to 96 and the fourth 3-16 bits (1101110) which equal to 110 see figure 8.The four positions that corresponding these numbers in SB are (0011) as show in figure 7, the number 0011 will determine the column number 3 to be tapped in the feedback function, so all content of column (**N**) number 16 and the content of column number 3 will be Xored correspondingly resulted a new column will be stored at column number 1 after shifting the columns to the right by 1, the last column will be ignored.



| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | ***0*** | 0 | 1 | 1 | 0 | ***0*** | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 | ***1*** | 1 | 0 | 1 | 0 | 1 | 1 | ***1*** | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |

**Figure 8 Horizontal shift steps**

## 4. Complexity Measur

Calculate the complexity of the algorithm need to specify the period of the key generator algorithm which will be equivalent to the complexity of the system [6]. The period of the **Shifted Box with Variable Stage Feed Back Function** algorithm can be computed by $\mathbf{2^{N*M}}$ (all possible length) $\mathbf{* 2^{\log_2 N} * 2^{\log_2 M}}$ (the Variable positions of the SB) $\mathbf{* 2^{42}}$ (the period of the LFSR) * 128! (All possible IPM) and the proposed algorithm will exceeds any input sequence in an applicable algorithm and might not be attacked in a brute-force attack. The algorithm cannot be attacked by known attacking method because of the high computational complexity of the permutation and matrix shifting.

## 5. Conclusions and Recommendations

### 5.1. Conclusions

The proposed algorithm has:-

1. A high complexity by implement the exponential function with fixed table, long periodicity, high nonlinearity.
2. The period of the algorithm can be computed by $\mathbf{2^{N*M}}$ (all possible length) $\mathbf{* 2^{\log_2 N} * 2^{\log_2 M}}$ (the Variable positions of the SB) $\mathbf{* 2^{42}}$ (the period of the LFSR) * 128! (all possible IPT).The periodicity of the algorithm obtained will exceed the length of any input media might be encrypted by any other algorithm.
3. Using LFSR with variable feed back stage will increase the nonlinearity of the algorithm.

### 5.2. Recommendations

The recommendations are that

1. Using long length of LFSR and a large fixed table that used to determine the position of the second stage will be increase the complicity.
2. Using **S**hifted Box (**SB**) in design stream cipher algorithm.

## 6. References

[1] Wikipedia the free encyclopedia: retrieved on **7/10/2008** from Http://en.wikipedia.org/wiki/Cryptoghrapy.

[2] Rolf Oppliger "*Contemporary Cryptography*" Artech House Boston London **2007.**

[3] Christof Paar and JPelzl "*Understanding Cryptography*", Springer, **2010**.

[4] Rainer A. Rueppel, "*Analysis and Design of Stream Ciphers*",Springer-Verlag Berlin, Heidelbergerg , **1986.**

[5] O. Goldreich, "*Foundations of Cryptography*", Department of Computer Science and Applied Mathematics, Weizmann Institute, **1995**.

[6] Beker and Piper, "*Cipher Systems*", Northwood Publication, U.K., **1982**.