New Address Shift Linear Feedback Shift Register Generator

Kholood J. Moulood
Department of Mathematical, Tikrit University, College of Education for Women, Salahdin.
E-mail: khmsc2006@yahoo.com

Abstract

In this paper we introduced a design of new pseudo random generator, which generate binary sequences that would be used as an encryption key in Stream Cipher Cryptosystem (SCC). The proposed generator consists of number of Linear Feedback Shift Registers (LFSR's), which are considered the basic unit of SCC and some nonlinear functions. The proposed cryptosystem called Address Shift LFSR (ASLFSR) cryptosystem. Lastly, ASLFSR generator subjects to set of Basic Efficient Criteria (BEC) to examine its output to prove its efficiency.
[DOI: 10.22401/JNUS.20.1.20]

Keywords: Stream Cipher Cryptosystem, Linear Feedback Shift Register, Address Shift LFSR, Basic Efficient Criteria.

1. Introduction

Cryptography is the study of information hiding and verification. It includes the protocols, algorithms and strategies to securely and consistently prevent or delay unauthorized access to sensitive information and enable verifiability of every component in a communication. Cryptography is derived from the Greek words: kryptós, "hidden", and gráphein, "to write" - or "hidden writing". People who study and develop cryptography are called cryptographers [12]. Cryptography is an interdisciplinary subject, drawing from several fields. Before the time of computers, it was closely related to linguistics. Nowadays the emphasis has shifted, and cryptography makes extensive use of technical areas of mathematics, especially those areas collectively known as discrete mathematics. This includes topics from number theory, information theory, computational complexity, statistics and combinatorics. It is also a branch of engineering, but an unusual one as it must deal with active, intelligent and malevolent opposition [4]. When information is transformed from a useful form of understanding to an opaque form of understanding, this is called **encryption**. When the information is reverted back into a useful form, it is called **decryption**. Intended recipients or authorized use of the information is determined by whether the user has a certain piece of secret knowledge. Only users with the secret knowledge can transform the opaque information back into its useful form. The secret knowledge is commonly called the **key**, though the secret knowledge may include the entire process or algorithm that is used in the **encryption/decryption**. The information in its useful form is called **plaintext** (or cleartext); in its encrypted form it is called **ciphertext**. The algorithm used for encryption and decryption is called a cipher [11].

2. Literatures Survey

The Geffe generator is defined by three maximum-length LFSRs whose lengths r_1, r_2, r_3 are pair wise relatively prime, with nonlinear combining function:

$$F(x_1,x_2,x_3)=x_1*x_2\oplus(1\oplus x_2)*x_3=x_1*x_2\oplus x_2*x_3\oplus x_3$$

The keystream generated has period $(2^{r_1}-1)(2^{r_2}-1)(2^{r_3}-1)$ and linear complexity $LC=r_1r_2+r_2r_3+r_3$. The Geffe generator is cryptographically weak because information about the states of LFSR1 and LFSR3 leaks into the output sequence. Despite having high period and moderately high linear complexity, the Geffe generator succumbs to correlation attacks [5].

Jennings Generator scheme uses a multiplexer to combine two LFSR's [7]. The multiplexer, controlled by LFSR-1, selects 1 bit of LFSR-2 for each output bit. There is also a function that maps the output of LFSR-2 to the input of the multiplexer. The key is the initial state of the two LFSR's and the mapping function.

Multispeed Inner-Product Generator, by Massey and Rueppel [9], uses two LFSR's clocked at two different speeds. LFSR-2 is

clocked d times as fast as LFSR-1. The individual bits of the two LFSR's are ANDed together and then XORed with each other to produce the final output bit of the generator.

Ali F. H. [2] introduces the mathematical process to generate a sequence from two generators of The Multiplicative Cyclic Group (MCG). The two generators with some initial variables (keys) make a unit called MCG unit. A number of MCG units are combined with each other by a combining logical function to get MCG system.

The main goal of this paper is to construct a new stream cipher system generator, generates good statistical properties digital sequences could be used in cryptography. The proposed generator considered as a stream cipher system which is depends on linear feedback Shift Registers (LFSR's). The LFSR unit considered a basic unit which the stream cipher systems depend on.

The four basic efficiency criteria, periodicity, linear complexity, randomness and correlation immunity are applied to measure the efficiency of the pseudo random sequences which are generated from the proposed generator.

3. Stream cipher

Symmetric key encryption schemes are divided in two main classes: **block ciphers** and **stream ciphers**. A stream cipher encrypts each bit/word independently. This is useful in areas where the buffering of data is not possible or where the encryption should be done as soon as an element arrives. This is also useful when the bandwidth is limited and when short messages are transmitted because there is no need of padding [6].

There are three main classes of stream ciphers: the one-time pad, the synchronous, and the self-synchronizing stream cipher. The one-time pad has a key of the same size as the plaintext and allows it to be used only once. The other two classes use one key for multiple encryptions. If the encryption only depends on the key, each encryption produces the same key stream which is then combined with the plaintext. Such a behavior facilitates attacks on the stream cipher. Thus, most of the stream cipher schemes use an additional public Initialization Vector (IV) which changes for each encryption. The initial state and the key stream then depend on the key and the IV [10].

Cryptography comes with a wide range of techniques in order to provide solutions for different security requirements – and these techniques require random sequences for many different purposes. One of the most important roles randomness plays in cryptography is represented by cryptographic keys which determine the transformation of the plaintext into cipher text and vice versa [1]. Considering that both the encryption and the decryption algorithms are publicly known together with all the cipher texts transmitted between the sender and receiver, the security of the whole cryptosystem is dependent on how the key information is managed: generated, agreed on, applied, stored and destroyed. The knowledge of the key entails the access to the secret message, thus the choice of the key space and the key derivation method is critical [10].

Most practical stream-cipher designs center around **Linear Feedback Shift Registers**. A linear feedback shift register (LFSR) is a shift register whose input bit is a linear function of its previous state [6].

The only linear function of single bits is **XOR**, thus it is a shift register whose input bit is driven by the Exclusive-OR (XOR) of some bits of the overall shift register value. The initial value of the LFSR is called the **seed**, and because the operation of the register is deterministic, the stream of values produced by the register is completely determined by its current (or previous) state. Likewise, because the register has a finite number of possible states, it must eventually enter a repeating cycle. However, an LFSR with a well-chosen feedback function can produce a sequence of bits which appears random and which has a very long cycle [9].

4. Basic Efficiency Criteria (BEC) for SCC

The basic criteria of key generator efficiency can be defined as the ability of key generator and its sequence to withstand the mathematical analysis which the cryptanalyst can be applied on them. The criteria of key generator efficiency depend on some/all elements of basic units of key generator, In the following sections, we will introduce these basic criteria.

4.1 Periodicity Criteria

The sequence $S=s_0,s_1,s_2.$ is said to be **n-periodic** if $s_i=s_{i+n}$ for all $i\geq 0$. The sequence s is **periodic** if it is n-periodic for some positive integer n. The period of a periodic sequence S is the smallest positive integer n for which S is n-periodic. If S is a periodic sequence of period n then the cycle of S is the subsequence S^n [8].

Let P(S) represent the period of the sequence S, let $P(S_i)$ be the period of the sequence S_i, $1\leq i\leq k$, then

$$P(S) = \text{lcm}(P(S_1),P(S_2),\ldots,P(S_k)) \quad \ldots(3.1)$$

4.2 Linear Complexity Criteria

The **linear complexity** of a finite binary sequence S^n, denoted $LC(S^n)$, is the length of the shortest LFSR that generates a sequence having S^n as its first n terms [12].

We used The **Berlekamp-Massey algorithm** to compute the linear complexity.

The Berlekamp-Massey algorithm is an efficient algorithm for determining the linear complexity of a finite binary sequence S^n of length n. The algorithm takes n. The Berlekamp-Massey algorithm is an efficient algorithm for determining the linear complexity of a finite binary sequence S^n of length n. The algorithm takes n iterations, with the N^{th} iteration computing the linear complexity of the subsequence S^N consisting of the first N terms of S^n.

Berlekamp-Massey algorithm

INPUT: a binary sequence $S^n = s_0, s_1, s_2,\ldots, s_{n-1}$ of length n.

OUTPUT: the linear complexity $L(S^n)$ of S^n, $0 \leq L(S^n) \leq n$.

PROCESS:

1.Initialization. $C(D)\leftarrow 1$, $r\leftarrow 0$, $m\leftarrow -1$, $B(D)\leftarrow 1$, $N\leftarrow 0$.

2. While ($N < n$) do the following:

2.1 Compute the next discrepancy d.

$$d\leftarrow(s_N+\sum_{i=1}^{L} c_i s_{N-i})\bmod 2.$$

2.2 If $d = 1$ then do the following:

$T(D)\leftarrow C(D)$, $C(D)\leftarrow C(D) + B(D).D^{N-m}$.

If $r\leq N/2$ then $r\leftarrow N + 1 - r$, $m\leftarrow N$, $B(D)\leftarrow T(D)$.

2.3 $N\leftarrow N + 1$.

3. Return(r).igr5".

4.3 Correlation Immunity Criteria

Let $X_1,X_2,\ldots,X_n$ be independent binary variables, each taking on the values 0 or 1 with probability 1/2. A Boolean function $f(x_1,x_2,\ldots, x_n)$ is m^{th}-order correlation immune if for each subset of m random variables $X_{i_1}, X_{i_2},\ldots,X_{i_m}$ with $1\leq i_1<i_2<\ldots<i_m\leq n$, the random variable $Z = f(X_1,X_2,\ldots,X_n)$ is statistically independent of the random vector ($X_{i_1}, X_{i_2},\ldots,X_{i_m}$); equivalently, $I(Z; X_{i_1}, X_{i_2},\ldots,X_{i_m})=0$ [7].

The function $f(x_1,x_2,\ldots,x_n)= x_1\oplus x_2\oplus\ldots\oplus x_n$ is $(n-1)^{th}$ order correlation immune.

4.4 Randomness Criteria

The sequence of crypto keys that are generated using the cryptosystem in this paper must achieve a good statistical random properties and pass the random standard tests which are Frequency test, Serial test, Poker test, Runs test and Autocorrelation test [11][13].

i.Frequency Test

The purpose of this test is to determine whether the number of 0's and 1's in (the output sequence) s are approximately the same, as would be expected for a random sequence. Let n_0, n_1 denote the number of 0's and 1's in s, respectively. The statistic used is:

$$X_1 = \frac{(n_0-n_1)^2}{n} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

ii.Serial Test

The purpose of this test is to determine whether the number of occurrences of 00, 01, 10, and 11 as subsequences of (the output sequence) s are approximately the same, as would be expected for a random sequence. Let n_0, n_1 denote the number of 0's and 1's in s, respectively, and let n_{00}, n_{01}, n_{10}, n_{11} denote the number of occurrences of 00, 01, 10, 11 in s, respectively. Note that $n_{00} + n_{01} + n_{10} + n_{11} = (n - 1)$ since the subsequences are allowed to overlap. The static used is

$$X_2 = \frac{4}{n-1}(n_{00}^2 + n_{01}^2 + n_{10}^2 + n_{11}^2) - \frac{2}{n}(n_0^2 + n_1^2) + 1 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

iii.Poker Test

Let m be a positive integer such that $\frac{n}{m} \geq 5$. and let $k = \frac{n}{m}$ Divide the sequence s into k non-overlapping parts each of length m, and let n_i be the number of occurrences of the i^{th} type of sequence of length m, $1 \leq i \leq 2^m$. The poker test determines whether the sequences of length m each appear approximately the same number of times in s, as would be expected for a random sequence. The statistic used is

$$X_3 = \frac{2^m}{k}\left(\sum_{i=1}^{2^m} n_i^2\right) - k \quad \text{..... (3)}$$

iv. Runs Test

The purpose of the runs test is to determine whether the number of runs (of either zeros or ones) of various lengths in the sequence s is as expected for a random sequence. The expected number of gaps (or blocks) of length i in a random sequence of length n is

$e_i = (n\text{-}i+3)/2^{i+2}$

Let k be equal to the largest integer i for which $e_i \geq 5$ Let B_i, G_i be the number of blocks and gaps, respectively, of length i in s for each i $1 \leq i \leq k$. The statistic used is

$$X_4 = \sum_{i=1}^{k} \frac{(B_i - e_i)^2}{e_i} + \sum_{i=1}^{k} \frac{(G_i - e_i)^2}{e_i}" \quad \text{..... (4)}$$

V. Autocorrelation Test

The purpose of this test is to check for correlations between the sequence s and (non-cyclic) shifted versions of it. Let d be a fixed integer, $1 \leq d \leq \lfloor n/2 \rfloor$. The number of bits in s not equal to their d-shifts is $A(d) = \sum_{i=0}^{n-d-1} s_i \oplus s_i + d$ where $\oplus$ denotes the XOR operator.

The statistic used is

$$X_5 = 2\left(A(d) - \frac{n-d}{2}\right)/\sqrt{n-d} \quad \text{..... (5)}$$

5. Design of Address Shift LFSR (ASLFSR) Cryptosystem

5.1 The Generator Components

The generator consists of the following contents:

- **Bank System** which consist of sixteen shift registers of variable length.
- **Address Linear Feedback Shift Registers Unit** (ALFSRU) which consists of four shift registers.
- **Shifting Address Unit (**SAU).
- **Fixed Memory of size 256 byte** (FM256).
- **Balance Unit** (BU) consists of two feedback register**.**

5.2 Key Management

The key consists of:

- **Basic Key (BK)**: From text key file, daily exchanged.
- **Message Key (MK)**: Randomly generated, each letter exchange.

5.3 The Generator Initialization

- **BK** mixed with **MK** to get (4) bits randomly as address to choose four Shift Registers from Bank System.
- For the mixing of BK and MK the **ALFSRU** will filled.
- The **ALFSRU** moved to fill **FM265** with observance of non repetition byte generation.
- **SR2, SR4** moved to fill **SR5, SR6** in **BU** continuously.

5.4 The Generator Movement

- Generate four bits randomly to use them as address to choose which Shift registers would fill (SR1, SR2, SR3, SR4) i.e (0101) means that fifth, sixth, seventh, and eighth shift registers would chosen to fill SR1, SR2, SR3,SR4 continuously.
- **ALFSRU** moved to get an address consists of 4 binary digits addresses and 4 position to get the Shifting Byte (SB).
- **SB** shifted by **S** value obtained from the below relationS= (KB2) mod 8, KB2=0 at the beginning
- The Shifting Byte **(SB**) used to get address from **FM256** to get **KB1 KB1 = FM256(SB)**
- **BU** moved twice times to get **KB2**
- The final Key (**KB)**= KB1 $\bigoplus$KB2

Fig.(1): illustrate the proposed cryptosystem.

6. Experiment and Results

In this paper the program is designed using visual basic programming language and the following results are obtained".

6.1 Periodicity

For the various lengths the periodicity is: 5.8401×10^{42} ".

6.2 Linear Complexity

Table (1), shows the linear complexity test using Berlekamp_Massey Algorithm".

Table (1)
Linear complexity test.

	Key length		
Test	160000	200000	2400000
Linear complexity	613	4128	50315

6.3 Correlation immunity

Table (2), shows the Correlation Immunity test.

Table (2)
The Correlation Immunity test.

Test	Key length		
Correlation Immunity	160000	200000	240000
SR1	0.507	0.5048	0.49968
SR2	0.489	0.5016	0.49959
SR3	0.482	0.4969	0.49893
SR4	0.483	0.4972	0.4889

6.4 Randomness Tests

"The Statistical tests results obtained for different lengths of output sequences as explained below":

Tables (3)
Results of applying Frequency, Serial and Poker tests.

Tests	160000	200000	240000	Decision
Frequency	0.288	0.287	0.284	**pass**
Serial	2.416	2.403	2.392	**pass**
Poker	19.04	19.12	19.01	**pass**

Tables (4)
Results of applying autocorrelation test.

	Key length		
τ	**160000**	**200000**	**240000**
1	0.0091	0.3965	0.123
2	1.6033	0.0481	20.0384
3	0.0252	0.0003	1.2321
4	0.1005	0.1295	0.807
5	0.6280	2.8572	0.0282
6	0.5797	4.1645	0.1485
7	3.0465	0.449	3.3992
8	0.1450	0.176	0.2314
9	1.3816	0.221	0.0422
10	0	0.1294	0.2131

7. Conclusions

1. The proposed key generator has good statistical properties mentioned previously these properties give the generator the qualification to be used as encryption system practically.
2. Many tests has made been made to test the efficiency of the proposed system, like randomness, periodicity, complexity, …etc.
3. The high nonlinearity of the ASLFSRG is represented by the Address Shifting unit which gives high complexity.
4. In the construction of ASLFSRG, we take in consideration most of the cryptanalysis tools in order to establish strong key generator to withstand all the cryptanalyst abilities to break the cryptosystems.

The suggested system can be developed to encrypt not only text files, but we can encrypt image, audio, video or any other important files.

References

[1] Alan G., "Computer Security and Cryptography", A John Wiley & sons, Inc. publications, 2007.

[2] Ali F., "*Use the Multiplicative Cyclic Group to Generate Pseudo Random Digital Sequences*", Journal of Al-Rafidain University College for Sciences, Vol.20, pp.122-135, 2006.

[3] Brüer J., "*On Nonlinear Combination of Linear Shift Register Sequences*", Internal

Report, Cryptologia Magazine, Vol. XVII, No. 2, pp. 187-201, 1983.

[4] Christof P., "Applied Cryptography and Data Security", Ruhr-University Bochum/ Germany, 2005.

[5] Geffe, P., "*How to Protect Data with Ciphers that are Really Hard to Break*", Electronics pp. 99-101, Jan. 4, 1973.

[6] Ekdhal, P., "*On LFSR based Stream Ciphers Analysis and Design*", Ph.D. Thesis, November 21, 2003.

[7] Jennings, S., "*Autocorrelation Function of the Multiplexed Sequence*", IKE Proceedings, v. 131, n. 2, Apr 1984, pp.169-172.

[8] Kinga M., Alin S., "Generation and Testing of Random Numbers for Cryptographic Applications", Proceedings of the Romanian Academy, Series A, Vol 13, Number 4, pp. 368–377, 2012.

[9] Massey J., and Rueppel R., "*Linear Ciphers and Random Sequence Generators with Multiple Clocks*", Advances in Cryptology: Proceedings of EUROCRYPT 84, Springer-Verlag, pp. 74-87, 1985.

[10] Mattsson, J., "Stream Cipher Design", M.Sc. thesis, 2006.

[11] Menezes A., Vanstone S., "Handbook of Applied Cryptography", CRC Press, Inc., 1997.

[12] Schneier B., "Applied Cryptography, Second Edition", A John Wiley & sons, Inc. publications, ISBN: 0471128457, 1996.

[13] Wenbo Mao Hewlett-Packard Company, "Modern Cryptography: Theory and Practice", Prentice Hall PTR, ISBN: 0-13-066943-1, 2003.

[14] William Stallings, "Cryptography and Network Security Fifth edition", Pearson Education, Inc., 2011.