Home

MEASUREMENT TECHNIQUES OF LFSR SEQUENCES

A. Ahmad, Sameer Al-Busaidi, Ahmed Al-Naamany and Mufeed Juma Al-Mushrafi

Information Engineering Department
Department of College of Engineering, Sultan Qaboos University
Fax. 00968-513454 / 00968-513416

E-mails: afaq@squ.edu.om / albusaid@squ.edu.om/ naamany@squ.edu.om/ mufeed@omantel.net.om

Abstract - Our study in this paper is focused mainly on the importance of LFSR in various applications and to present the current research status in this direction. How to measure the parametric issues related to the applications of pseudo random binary sequences would be the main theme of this work.

I. Introduction

The ease of implementation and simple operations of Linear Feedback Shift Registers (LFSRs) have found them to fit into a wide range of multiple uses in digital systems design (e.g. see Table 1; below:).

TABLE 1
Systems using LFSRs

Systems / Applications	Manufacture model	LFSR size
Cyclic Redundancy Check (CRC)	CRC-12	12
	CRC-16	16
	CCITT	16*
	AUTODIN-II	32
Radio Amateurs (Spread-spectrum)	SS-7	7
	SS-13	13
	SS-19	19
Cellular Telephone (European)	A5 - I	19
	A5 - II	22
	A5 - III	23
	A5 - IV	17
ATM Networks	CRC-32	32
GPS Satellite	GPSS - I	10
	GPSS - II	10*

* (Modified structure)

The systems which uses the applications of the pseudo random binary sequences generated by LFSRs include:

a. Data Encryption / Decryption
b. Digital Signal Processing
c. Wireless Communications / ATM Networks
d. Built-in Self-Test (BIST)
e. Data Integrity / error control and coding
f. Data Compression
g. Pseudo random Number Generation (PN)
h. Direct Sequence Spread Spectrum
i. Scrambler / De-scrambler
j. Optimized Counters
k. Cell Phone Technology
l. GPS Satellite Systems
m. Nonlinear Biological Systems
n. Computer and Video games, and etc.

Furthermore, many fields of research (e.g. physics and even finance) are increasingly relying on large computer simulations to study phenomenon that cannot be observed directly for obvious reasons. In these circumstances, the use of the sequences generated by LFSRs is an alternate validation methodology to avoid an unforeseen subtle [1], [2].

As, it is evident from the above-mentioned fact that the study of art and theory of generating and applying of sequences produced by LFSRs are becoming an essential for engineers and scientists in the present scenario of integrated information technology.

Although, the LFSR has wide range of applications due to its attributes of simple implementation as well as operation; But many issues restricts the designers and users to build and use the systems to meet their own requirements based on the desired application environment conditions. Considering an n bit LFSR, which can produce the pseudo random binary sequences of lengths ranging from n to 2^n-1. The range of pseudo random binary sequence's growth depends on the factors of characteristic polynomials and as well as on polynomial seeds. To search an appropriate set of polynomial seed which along with a suitable set of characteristic polynomial from the sets of ranges 2^n-1 and 2^{n--1} respectively becomes an uphill complex task. Associated with these, the problem of the selection of pseudo random binary sequence whose properties are demanded in the justification of the design of the system. Thus, despite their good pseudo noise (PN) like statistics, Linear Feedback Shift Registers cannot ensure of generating such sequences, which can ensure satisfactory performance in terms of complexity unless prohibitively high lengths of LFSRs are employed. But the high length of LFSRs has to face a severe challenge of application time along with the considerable amount of extra hardware. So to overcome the drawbacks and to maintain the complexity performance of the system, a nonlinear structure of LFSR could be imposed [1] – [4].

Therefore, LFSR is popular in systems applications but the associated issues are complex Hence, testing of the sequences is rather approaching to become an updated task of system engineers and scientists rather than acquiring the knowledge of simply using such sequences.

Our study in this paper is focused mainly on the importance of LFSR in various applications and to present the current research status in the direction of measurement of the statistical properties of pseudo random binary sequences

II. Pseudorandom Binary Sequences [4]

A Linear Feedback Shift Register (LFSR) is a mechanism for generating a sequence of pseudo random binary bits. The register (see Figure 1) consists of a series of cells that are set by an initialization vector. A clock regulates the behavior of the register and at each clocking instant, the contents of the cells of the register are shifted right by one position, and the XOR (⊕) of a subset of the cell contents is placed in the leftmost cell. One bit of output is usually derived during this update procedure. LFSRs are fast and easy to implement in both hardware and software. With a judicious choice of *feedback taps [c_1, c_2,..., c_n]*, the sequences that are generated can have a good and different statistical appearance. Table 2 elaborates the functioning of LFSR with an example (Example 1) of considering a 4-bit LFSR.

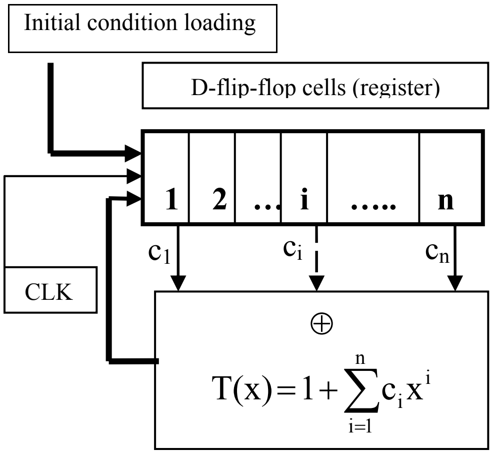
Initial condition loading
D-flip-flop cells (register)
1
2
...
i
.....
n
CLK
c_1
c_i
c_n
⊕
$T(x)=1+\sum_{i=1}^{n} c_i x^i$

Fig. 1. An n-bit LFSR's structure

Example 1: Let n = 4, $c_1 = c_2 = c_4 = 1$, $c_3 = 0$ with initial state (0,1,1,0) then we have the output sequence from the output point of the 4th D-flip-flop as demonstrated in Table 2.

TABLE 2
Functioning of a 4-bit LFSR's structure

Time (clock)	**LFSR States**	**Output**
0	0,1,1,0	x
1	1,0,1,1	0
2	0,1,0.1	1
3	0,0,1,0	1
4	0,0,0,1	0
5	1,0,0,0	1
6	1,1,0,0	0
7	0,1,1,0	0

Since we have reached the initial state again, this LFSR produces a sequence, s(x) = [.., y_7, y_6, .., y_2, y_1] = (..0010110..) with period 7. There are 2^n possible states, but the all zero state cannot be achieved unless it is started with it, and to start with, all zero state will switch to again in the same state. So therefore, there are 2^n - 1 possible states, so this is the maximum possible period. And, a sequence produced by an n length LFSR that has period 2^n-1 is called a *PN-sequence* (or a pseudo-noise sequence or m-sequence). LFSR's can be characterized by the *characteristic polynomial*. The characteristic polynomial of an LFSR can be given

Top

as function of the feedback taps in the form of the polynomial as shown in Eq. 1, below.

$$T(x) = 1 + \sum_{i=1}^{n} c_i x^i \qquad (1)$$

Where $c_n = 1$ by the definition of LFSR.

III. Some Facts On LFSR'S Theory [4]

1. Every polynomial T (x) with coefficients in GF (2) divides $x^m + 1$ for some m. The smallest m for which this is true is called the *period* of T (x).

2. An *irreducible* (can not be factored) polynomial of degree n has a period, which divides $2^n - 1$.

3. An irreducible polynomial of degree n whose period is $2^n - 1$ is called a *primitive polynomial*.

Theorem 1: A LFSR produces a PN-sequence if and only if its characteristic polynomial is a primitive polynomial.

The characteristic polynomial of Example 1 is: $T(x) = x^4 + x^2 + x + 1 = (x + 1)(x^3 + x^2 + 1)$ and so is reducible and therefore, not primitive. Whereas, $T(x) = x^4 + x + 1$ or $T(x) = x^4 + x^3 + 1$ are irreducible polynomials and capable of producing sequences of period 15 and so, is a primitive polynomial. It is also to be noted here that all irreducible polynomials cannot be guaranteed as primitive polynomial.

PN-sequences satisfy all of Golomb's conditions for pseudo-randomness. Turning now to the suitability of LFSR circuits for cryptographic environments.

C1: One can obtain sufficiently large periods by taking n large enough. In fact, n = 166 will give a period of $2^{166} - 1 > 10^{50}$.

C2: Being simple Boolean circuits, LFSR's are extremely easy to implement and are very fast.

C3: Given 2n consecutive plaintext bits, $s_k, s_{k+1}, ..., s_{k+2n-1}$ we can write down a system of n equations in the n unkowns $c_0, ..., c_{n-1}$ which is non-degenerate and so has a unique solution. This gives the characteristic polynomial and so the LFSR to the cryptanalyst [5].

Thus, LFSRs should not be used in cryptographic work. However, despite this, LFSR's are still the most commonly used technique in cryptography and will continue to be as building blocks in more secure systems [6]. But it is for sure, that the sequences generated by single LFSRs are not secure [7], [8]. Hence, a powerful mathematical framework has been developed over the years, which allows for their straightforward analysis of binary sequences. Therefore, a cryptographer likes to convince himself that the sequences, which are produced, are random enough to be secure. Analyzing the sequences does this. These analyses are done for the sequence to make sure that those sequences satisfy randomness criteria. The idea of randomness reflects the impossibility of predicting the next digit of the sequence from all the previous ones. Any sequence can be random, if it satisfies the three Golomb postulates. These three randomness postulates are used to measure the randomness of a periodic binary sequence of period p. These postulates are about the properties of the number of 0s and 1s, distribution of run lengths of groups of 1s and 0s and, the auto-correlation function of the binary sequence. However, the statistical tests can be performed to provide a quantitative measure of randomness. They measure the relative frequencies of certain patterns of zeros and ones in a sequence. It is impossible to give mathematical proof that a generator is indeed a random bit generator, but the statistical tests can detect certain kinds of weakness the generator may have. Therefore, the main aim of the statistical tests is to investigate the randomness of large binary strings which is the key stream produced by random and/or pseudo random generator. There are numerous statistical tests, which can be applied to a sequence [9], [10]. Some of these statistical tests are: Chi- Square test (X^2 - Test), Frequency test, Serial test, Run test and, Poker test

IV. Randomness Measurements – Statistical Approach [3], [4], [9], [10]

Let test is applied to a sequence of positive integers $[Sn] = [S_0, S_1, S_2 \ldots]$. This sequence purports to be independently and uniformly distributed between 0 and f-1, where f is the degree of freedom and it is different from test to test depends on the number of categories. The value f should be large enough so that the test is meaningful, but not so large that the test becomes impractically difficult to be carried out. A basic test used in a number of statistical tests is the chi-square Test (X^2 - test). It analyses a distribution of n events and the probabilities associated with each event in the form of distribution. The X^2 – test statistics can be written as:

$$v = \sum_{m=1}^{k} \frac{\left(N_m - nP_m\right)^2}{nP_m} \qquad (2)$$

where
Nm = the number of occurrences of observation m;
Pm = the theoretical probability of occurring m;
n = total number of observations, and,
k = number of categories.
If the entries are as: v < 1% or v > 99%, we reject the sequence as being not sufficiently random. If v lies between 1% and 5% entries or between 95% and 99% entries, the number are suspect. Otherwise, it is acceptable.

A. Frequency Test

It is used to determine whether a sequence is biased or not. The first requirement that the sequence must meet the criteria that its numbers are uniformly distributed between 0 and f -1. For each integer r, 0 ≤ r < f, count the number of times that Sj = r, for 0 ≤ j ≤ n, and then apply the X^2 - test using the number of categories k = f and probability $p_m = 1/f$ for each category. In this test, the number n should be taken at least 5f. This test is also called equi-distribution test.

Example 2: Let a sequence is:
010011111010000011100010010011011010110111 1011000110100101110. Applying Equation (2) we get v = 0.267. From X^2 -table, the theoretical value = 3.84. This clearly indicates that this value of v obtained for the sequence satisfies the frequency tests and, thus, passes the randomness test using the technique of frequency test.

B. *Serial Test*

This test is used to make sure that the sequence in pairs of successive numbers is uniformly distributed in an independent manner or not. To carry out the serial test, we count the number of times that pairs $(N_{2j}, N_{2j+1}) = (q, r)$ counted for 0 ≤ j < n, these counts are to be made for each pair of integers (q, r) with 0 ≤ q, r < f and the X^2-test is applied to these $k = f^2$ categories with probability $1/ f^2$ in each category. As with frequency test, f may be any convenient number, but it will be smaller than the values suggested above since a valid X^2-test should have n large compared to k ($n = 5f^2$ at least). Also, for this test, the number of sequence that is taken should be 2n in order to make test for n observation.

Example 3: Considering the same sequence of Example 2 and, applying Equation (2) and using of X^2 – table, the sequence can be declared as either pass or fail of serial test of randomness measurements.

C. Run Test

A sequence is tested for run up and run down. This means we examine the length of monotone sub-sequences of the original sequence, i.e., segments, which are increasing or decreasing. As an example of the definition of run, consider the sequence of ten numbers "1275849763". Now put vertical lines at the left and right most positions and between Nj and Nj+1 whenever Nj > Nj+1. From these values the following values are obtained

|127 |58| 49| 7| 6| 3|

This displays the run ups. As noted from above there is a run of length 3, followed by two runs of length 2, and three runs of length 1. In case of binary sequence the run test counts the number of runs of ones (blocks) and runs of zeros (gaps) for each possible run length. The Run Test checks the distribution of run lengths with hypothesis distribution for which the expected number blocks of length B (k) is

$$E(B(K)) = \frac{n-k+3}{2^{k+2}} \qquad (3)$$

where
B (k): number of blocks of length k,
G (k): number of gaps of length k,
E (k): expected number of blocks/gaps of length k.

$$v = \sum_{k=1}^{m} \frac{\left(B(k) - E(B(k))\right)^2}{E(B(k))} + \sum_{k=1}^{m} \frac{\left(G(k) - E(G(k))\right)^2}{E(G(k))} \qquad (4)$$

The maximum value of k, equals to m, for the condition E (k) > 5. Then the degree of freedom will be 2m - 2. The failure of this test indicates that there is a bad distribution of run lengths.

Example 4: The same sequence of Example 2 is considered again and, while using Eqs. (3) and (4) we get v = 1.26. Here, m = 2, so that degree of freedom = 2m - 2 = 2 and from X^2 - able the value is 5.99 which is greater than 1.26, indicates that this sequence satisfy the run test.

D. Poker Test

The aim of the poker test is to show that there is an equal number of each of the 2m possible hand

Top

patterns. This test partitions the stream into **F** hands. The size m of those hands can be any length. Any fractional hands remaining in the total data are ignored. For a stream of size n, the total number of hands, **F**, is the integral part of n/m.

A prevalence or lack of any hand pattern would give great deal of information. It should be noted that the actual hand size to test might depend on the nature of the plaintext. For example, if the plaintext were eight bit ASCII characters it would be recommended to examine hand size, m, of length eight bits [54].

Example 5:

The sequence of Example 2 can be divided into 7 hands each one contain 8 bits and the remaining from the total data are ignored. Based upon the judgment of occurrences of all of a kind, one of a kind, two of a kind, 3 of a kind and finally four of a kind and use of Eq. (2) one can compute the value of v. In this case it comes out to be 3.252. From X^2 – table, the theoretical value of v = 9.49, so that even poker test is also satisfied for this sequence.

V. Realization of Measurement Tools

We have simulated the all types of the test tools. Due to scarcity of the space below is provided the simulated model for the Serial test in Figures 2. The models are simulated using *MATLAB SIMULIK.*

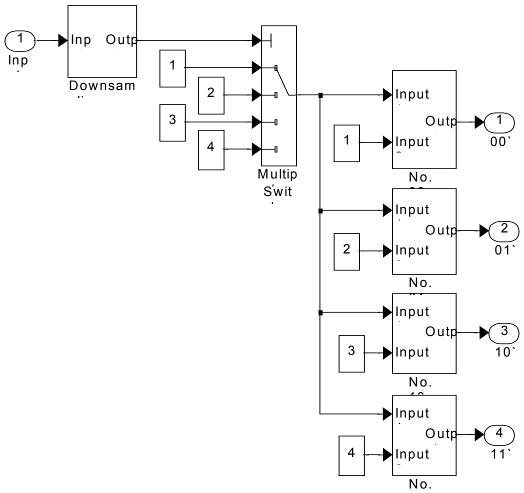

Figure 2: Simulation model for Serial test

Using the down sampling by 2 (see Figure 2) we divide the sequence in all possible combinations of two. The blocks No. 00, No. 01, No. 10, and No. 11 check how many times 00, 01, 10, and 11 combinations respectively have occurred in the sequence.

VI. Conclusion

Through this communication we mainly tried to inculcate the culture of LFSR in the age where the security has become a vital issue. How to measure the parametric issues related to the applications of pseudo random binary sequences in a simplified manner is demonstrated through the examples. Also, described in this paper is the realization of simulation models for these parametric tests.

Acknowledgements

The acknowledgements are due to authorities of Sultan Qaboos University (Sultanate of Oman) for providing research support grant (SQU - DOPSAR/ IG/ ENG/ EING/ 02/ 03).

References

[1]. Chi-Chun Lo, and Yu-Jen Chen, ‘Secure Communication Mechanisms For GSM Networks’, IEEE Transaction On Consumer Electronics, vol.45 no. 4, pp.1074 – 1080, 1999.

[2]. Rueppel, R. A., ‘Analysis and Design of Stream Ciphers’, New York, NY: Springer, 1986

[3]. Alaa Eldin Abdel Rhman Ibrahim Omar, ‘Evaluation And Cryptanalysis Of Cryptographic System’, Ph.D. Thesis, Ain Shams University, 1996

[4]. Solomon W. Golomb, ‘Shift Register Sequences, Aegean Park Press, Revised Edition 1982.

[5]. Massey, J. L., Shift register synthesis and BCH Decoding’, IEEE Transactions On Information Technology, vol. IT-15, Jan 1969

[6]. J. D_j. Golic, ‘Cryptanalysis of Three Mutually Clock-Controlled Stop/Go Shift Registers’, IEEE Transactions on Information Technology, Vol. 46, No. 3, pp.1081-1090, May 2000.

[7]. Chi-Kwong Chan, Cheng, L.M., ‘The CHNN Nonlinear Combination Generator’, IEEE International Conference On Electronics, Circuits and Systems, vol.2, pp.257-260, 1998

[8]. Rueppel, R.A., ‘Good Stream Ciphers Are Hard To Design’, International Carnahan Conference On Security Technology, (1989 proceedings), pp.163 – 174, 1989

[9]. Kasselman, P.R., ‘A Statistical Test For Stream Ciphers Based On The Maximum Order Complexity’, South African Symposium On Communication and Signal Processing, pp.213 – 218, 1998.

[10]. Vattulainen, I., Ala-Nissila, T., and Kankaala, K., Physical tests for Random Numbers in Simulations, Physical Review Letters, Vol. 73, Number 19, 7 November 1994, pp. 2513-2516.