Generation of Nonlinear Feedback Shift Registers with special-purpose hardware

Tomasz Rachwalik, Janusz Szmidt,
Robert Wicik, and Janusz Zabłocki

Military Communication Institute
ul. Warszawska 22A, 05-130 Zegrze, Poland

Abstract. The nonlinear feedback shift registers (NLFSR) are used to construct pseudorandom generators for stream ciphers. Their theory is not so complete as that of the linear feedback shift registers (LFSR). In general, it is not known how to construct NLFSRs with maximum period. The direct method is to search for such registers with suitable properties. We used the implementation of NLFSRs in Field Programmable Gate Arrays (FPGA) to perform a corresponding search. We also investigated local statistical properties of the binary sequences ganerated by NLFSRs of order 25 and 27.
Key words: Nonlinear feedback shift registers. Maximum period. Linear complexity. Hardware implementation. Randomness properties.

1 Introduction

Feedback shift registers (FSR) sequences have been widely used in many areas of communication theory, as key stream generators in stream ciphers cryptosystems, peudorandom number generators in many cryptographic primitive algorithms, and testing vectors in hardware design. Golomb's book [5] is a pioneering one that discusses this type of sequences. A modern treatment of the subject is contained in Golomb and Gong [6].

The theory of linear feedback shift registers (LFSR) is understood quite well. In particular, it is known how to construct the LFSRs with maximum period; they correspond to primitive minimal polynomials over the binary field $\mathbb{F}_2$. The primitive LFSRs have a drawback as their linear complexity is equal to their order. In recent years, nonlinear feedback shift registers (NLFSR) have received much attention in designing numerous cryptographic algorithms such as stream ciphers and lightweight block ciphers to provide security in communication systems. In most cases, NLFSRs have much bigger linear complexity than LFSRs of the same order. However, not much is known about cyclic structures of NLFSRs; most of the known results are collected in Golomb's fundamental book [5].

We used the implementation of NLFSRs in Field Programmable Gate Arrays (FPGA) to perform a search of NLFSRs of the order up to $n = 27$, the maximum period equal to $2^n - 1$ and a possibly simple algebraic structure of the feedback function. We also investigated local statistical properties of the binary sequences generated by NLFSRs of order 25 and 27. We hope to continue this research further.

2 Feedback Shift Registers

In this section, we give definitions and basic facts about feedback shift registers (FSR). We use $\mathbb{F}_2$ to denote the binary finite field. $\mathbb{F}_2[x]$ denotes the ring of polynomials in the indeterminate x and with coefficients from $\mathbb{F}_2$. Let $\mathbb{F}_2^n$ be the n-dimensional vector space over $\mathbb{F}_2$ consisting of the n-tuples of elements of $\mathbb{F}_2$. Any function from $\mathbb{F}_2^n$ to $\mathbb{F}_2$ is referred to as a *Boolean function* on n variables. A sequence of elements $\mathbf{s} = (s_0, s_1, \ldots)$ of $\mathbb{F}_2$ is called a *binary sequence*. A sequence $\mathbf{s} = (s_i)_{i=0}^{\infty}$ is called *periodic* if there is a positive integer p such that $s_{i+p} = s_i$ for all $i \geq 0$. The least positive integer with this property is called a *period*.

A binary n-stage feedback shift register is a mapping $\mathfrak{F}$ from $\mathbb{F}_2^n$ into $\mathbb{F}_2^n$ of the form

$$\mathfrak{F} : (x_0, x_1, \ldots, x_{n-1}) \longmapsto (x_1, x_2, \ldots, x_{n-1}, f(x_0, x_1, \ldots, x_{n-1}),$$

where f is a Boolean function on n-variables which is called the *feedback function.* The shift register is called a *linear feedback shift register* (LFSR) if $\mathfrak{F}$ is a linear transformation from the vector space $\mathbb{F}_2^n$

into itself. Otherwise, the shift register is called a *nonlinear feedback shift register* (NLFSR). The shift register is called *nonsingular* if the mapping $\mathfrak{F}$ is a bijection. Further, we will consider only nonsingular and mostly nonlinear feedback shift registers. It can be proved (see e.g. [5]) that the feedback function of a nonsingular feedback shift register has the form

$$f(x_0, x_1, \ldots, x_{n-1}) = x_0 + F(x_1, \ldots, x_{n-1}), \tag{1}$$

where F is a Boolean function on $n-1$ variables.

Consider a binary sequence $\mathbf{s} = (s_i)_{i=0}^{\infty}$ whose first n terms $s_0, s_1, \ldots, s_{n-1}$ are given and whose remaining terms are uniquely determined by the recurrence relation

$$s_{i+n} = f(s_i, s_{i+1}, \ldots, s_{i+n-1}) \quad for\ all\ \ i \geq 0. \tag{2}$$

We call $\mathbf{s}$ an output sequence of the feedback shift register given by (1). The binary n-tuple $(s_0, s_1, \ldots, s_{n-1})$ is called the *initial state vector* of the sequence $\mathbf{s}$ or the *initial state* of the feedback shift register. The recurrence relation (2) can be implemented in hardware as a special electronic switching circuit consisting of n memory cells which is controlled by an external clock to generate the sequence $\mathbf{s}$ (see Figure 1).

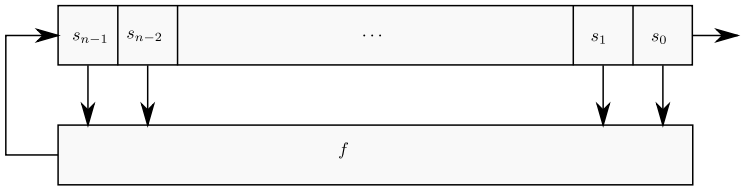

Fig. 1. A block diagram of a Feedback Shift Register

The period of an output sequence of a binary n-stage nonsingular FSR is at most 2^n. There are some sequences with maximum period.

Definition 1. The de Bruijn sequence of order n $(a_0, \ldots, a_{2^n-1})$ of elements from the binary field $\mathbb{F}_2$ is a sequence of period 2^n in which all different n-tuples appear exactly once.

It was proved by Flye Sainte-Marie [3] in 1894 and independently by de Bruijn [1] in 1946 that the number of cyclically equivalent sequences satisfying the Definition 1 is equal to

$$B_n = 2^{2^{n-1}-n}. \tag{3}$$

Definition 2. The modified de Bruijn sequence of order n $(a_0, \ldots, a_{2^n-2})$ is a sequence of period 2^n-1 obtained from the de Bruijn sequence of order n by removing one zero from the tuple of n consecutive zeros.

In 1990 Mayhew and Golomb [10] investigated sequences satisfying the Definition 2 and their linear complexity. These sequences were called by Gammel *et al.* [4] the *primitive sequences*. In the case of linear feedback shift registers these sequences are generated by primitive polynomials and their theory is understood quite well [8]. The primitive sequences are very important in cryptographic applications since:

1. They exist. There are B_n primitive sequences altogether (the linear and nonlinear ones). The number of primitive LFSRs is equal to
$$\frac{\varphi(2^n-1)}{n},$$
where φ denotes the Euler phi function, hence there are much more NLFSRs than LFSRs.
2. The primitive sequences have good statistical properties. They satisfy Golomb's main postulates. The linear complexity of a NLFSR (the order of a LFSR generating the same sequence) is much bigger than 2^{n-1} and many of them have the most possible linear complexity equal to 2^n-2. Let us recall that the linear complexity of a primitive LFSR of order n is just equal to n.
3. There are primitive NLFSRs for which the Algebraic Normal Form of the Boolean function F in formula (1) is quite simple; it has low algebraic degree and a possibly small number of terms. Since there are $2^{2^{n-1}}$ different Boolean functions on $n-1$ variables, hence the probability that a randomly chosen function of the form (1) is a primitive NLFSR is equal to
$$\frac{2^{2^{n-1}-n}}{2^{2^{n-1}}} = \frac{1}{2^n}$$
and as n grows it becomes smaller.

The task is to find primitive NLFSRs with a possibly simple algebraic form and this is much more difficult. A method how to construct such primitive NLFSRs is not known and we have to search for them. Gammel *et al.* [4] found simple primitive NLFSRs up to the order 33 and they used them in the design of the stream cipher Achterbahn, but neither the method of searching nor the average time needed to find such good NLFSRs have been revealed.

It is also an open problem to prove lower bounds of the linear complexity of NLFSRs. Mayhew and Golomb [10] investigated all modified de Bruijn sequences of order 5 and 6; there are $2^{11} = 2048$ and 2^{26} of them, respectively. It appears that there is a very small number of such sequences with low linear complexity. In the case of $n = 5$, there are no NLFRSs with linear complexity equal to 10 and there are only 10 sequences with linear complexity equal to 15. One can form a conjecture that for the order n of NLFSR being a prime number the lower bound of the corresponding linear complexity is equal to $3n$. It is implied by a more general conjecture formed in Kyureghyan's paper [7] and the results of [10]. The upper bound of the linear complexity of NLFSRs is $2^n - 2$ and this bound is tight. We calculated the linear complexity of the NLFSRs no $1 \div 4$ given in section 5 and it is equal to $2^{25} - 2$ for all of them. There is a recent interest in searching for and constructing primitive NLFSRs suitable for cryptographic applications, see [2], [9], [13].

3 The FPGA implementation

We implemented an algorithm for searching nonlinear feedback shift registers of order n having maximum period $2^n - 1$ using hardware devices from our previous projects. They were equipped with Altera EP3C80 Field Programmable Field Arrays. We used Altera Quartus II v.9.0. design software to simulate and compile the current project.

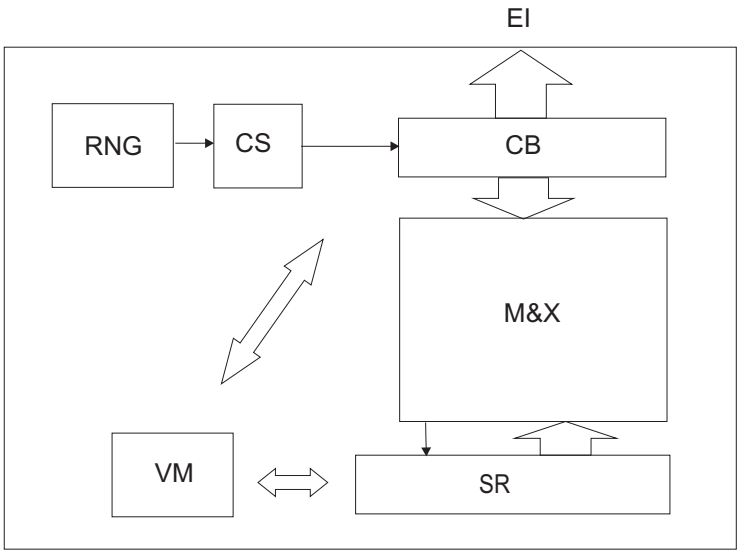

Fig. 2. A single module of the searching machine

The random NLFSR searching module (RNSM) consists of a random number generator (RNG), a coefficients selector (CS), a coefficients buffer (CB), multiplexers and XOR block (M&X), a shift register (SR), and a verification machine (VM). Random numbers are taken from the RNG. Coefficients are downloaded byte by byte into the CS, where their values and repetitions are controlled. Then the bytes go to the CB, whose task is to store combinations of coefficients during the test. The multiplexers define the feedback function of NLFSR according to the data buffered in the CB. Their outputs are connected to the XOR gate. Next, the output of the XOR function feeds the SR. The SR is set with a seed value at the beginning of a searching process by the VM and it starts to shift. After the first repetition of the seed the test is finished. A positive result is sent to the Ethernet Interface (EI), which is the same for all implemented modules. A negative result starts a new process of random generation and testing.

The attempts to find NLFSR were made by drawing 32 taps. Four of them feed a four-input AND gate. There are also two three-input AND gates and four two-input AND gates. We also implemented a version with 40 taps but there are not any results up to now.

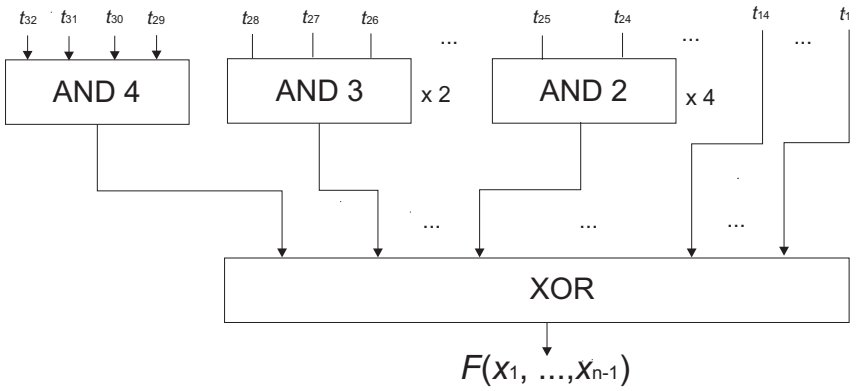

Fig. 3. The structure used to generate NLFSR

A single RNSM provides a superior search compared to the application of the same functionality embedded in a fairly efficient PC. For example, to obtain NLFSRs of order 15 with period $2^{15}-1$, we had to wait on average 3 seconds using the RNSM, whereas working on a PC it took 5 minutes. During our search for NLFSRs of order 25 nad 27 with maximum periods 128 RNSMs were implemented in four physical devices. The 32 modules implemented in a single device worked and stored results independently. The four devices were connected to a hub and a personal computer (PC) with the Wireshark sniffer. The FPGA was clocked with 65.536 MHz, although the maximum possible clocking is 128 MHz. The average time to find one NLFSR of order 25 was 4 hours and the average time to find one NLFSR of order 27 was 21 hours, respectively.

4 Randomness properties

The purpose of this section is to check experimentally the randomness properties of subsequences of the sequences generated by NLFSRs of section 5. The modified deBruijn sequences of order n have period 2^n-1 and all different n-tuples appear only once, except the allzero tuple. The whole sequence generated by NLFSR should have good statistical properties; since there is a nonlinear feedback, we also decided to check the statistical properties locally, for subsequences generated by NLFSR starting from randomly chosen initial state vectors. Let $s = s_0, s_1, \ldots, s_{m-1}$ be a binary sequence of length m. We test the randomness using seven basic statistical tests from [11], [14]. These are:

1. **Frequency test** – the purpose of this test is to determine whether the number of 0's and the number of 1's in the investigated sequence s are approximate the same, as it would be expected for a random sequence.
2. **Serial test** - the purpose of this test is to determine whether the number of occurrences of 00, 01, 10, 11 as subsequences of s are approximate the same, where the subsequences are allowed to overlap.
3. **Two bit test** - it verifies whether the number of occurences of subsequences 00, 01, 10, 11 are approximate the same, where the subsequences are not overlapping.
4. **8-bit poker test** - it verifies whether bytes of each possible value appear approximate the same number of times.
5. **16-bit poker test** - it verifies whether 16-bit words of each possible value appear approximate the same number of times.
6. **Runs test** - the purpose of this test is to determine whether the number of runs of either zeros or ones of various lengths (here from 1 to 22 bits) in the sequence s are as expected for a random sequence.
7. **Autocorrelation test** - the purpose of this test is to check for correlations between the sequence s and shifted versions of it (here by 1,2, ... , up to 8 bits).

The tests $1 \div 6$ use as a reference distribution the chi-square distribution with suitable number of degree of freedom and the seventh test uses the standard normal distribution. The observed frequencies of events are compared with their expected frequencies. We do not use hypothesis testing in a classical manner, where the hypothesis H_0 is verified using the calculated statistics. All events are possible, so we split the calculated statistics into 8 classes from A to H according to the range of significance level. The class A identifies a group of the best statistics and the class H identifies the worst case in terms of randomness, but all cases are possible with suitable probabilities as it is shown in Table 1.

Table 1. Percentages of appearances of classes

Classes	A+B+C	A	B	C	D	E	F	G	H
%	95	80	10	5	2.5	1.5	0.5	0.4	0.1

We tested subsequences produced by NLFSRs of section 5 starting from randomly selected initial states. First, we generated the full period sequences and then each sequence was divided into subsequences of 2^{20} bits each:

- $4 \cdot 2^5$ binary subsequences for NLFSRs of order 25 (no $1 \div 4$)
- $3 \cdot 2^7$ binary subsequences for NLFSRs of order 27 (no $5 \div 7$)

The obtained results of experiments are given in Table 2. It shows that the percentages of appearances of classes of statistics for 1 Mbit subsequences are similar to the expected appearances of classes for random sequences. These results indicate that the examined NLFSRs have good statistical properties.

Table 2. Percentages of appearances of classes of subsequences

NLFSR	A+B+C	A	B	C	D	E	F	G	H
1	94.64	81.70	8.04	4.91	1.79	2.23	0.89	0.45	0.00
2	94.20	81.25	8.04	4.91	3.13	1.34	0.45	0.89	0.00
3	95.98	86.61	5.80	3.57	1.34	2.68	0.00	0.00	0.00
4	96.43	82.14	8.93	3.36	2.23	1.34	0.00	0.00	0.00
5	94.64	78.79	10.16	5.69	2.68	1.23	1.00	0.33	0.11
6	95.98	80.25	11.94	3.79	2.23	0.67	0.78	0.33	0.00
7	95.20	82.59	8.71	3.91	3.01	1.00	0.22	0.22	0.33

5 Examples of NLFSRs

The NLFSRs of order 25:

$$1: x_0 + x_8 + x_9 + x_{10} + x_{11} + x_{19} + x_{20} + x_{21} + x_{23} + x_6x_{21} + x_{10}x_{14} + x_{12}x_{20} + x_{19}x_{20} + x_4x_{18}x_{21} + x_{11}x_{18}x_{22} + x_1x_5x_7x_{23}$$

$$2: x_0 + x_6 + x_7 + x_8 + x_{11} + x_{14} + x_{15} + x_{18} + x_{19} + x_5x_{10} + x_7x_{21} + x_{11}x_{16} + x_{12}x_{17} + x_1x_{10}x_{18} + x_{15}x_{17}x_{22} + x_8x_{10}x_{15}x_{18}$$

$$3: x_0 + x_6 + x_{12} + x_{13} + x_{16} + x_{20} + x_{21} + x_{22} + x_3x_{18} + x_{13}x_{19} + x_{13}x_{20} + x_5x_{12}x_{20} + x_8x_{18}x_{22} + x_{12}x_{15}x_{21}$$

$$4: x_0 + x_6 + x_{11} + x_{14} + x_{16} + x_{17} + x_{18} + x_{19} + x_{23} + x_4x_{19} + x_4x_{21} + x_5x_{22} + x_9x_{19} + x_1x_{17}x_{23} + x_5x_7x_{18} + x_5x_{12}x_{19}$$

The NLFSRs of order 27:

$$5: x_0 + x_4 + x_8 + x_9 + x_{11} + x_{12} + x_{15} + x_{16} + x_{23} + x_{12}x_{22} + x_{13}x_{23} + x_{13}x_{25} + x_{22}x_{23} + x_7x_8x_{24} + x_{12}x_{14}x_{26} + x_6x_{11}x_{19}x_{22}$$

$$6: x_0 + x_1 + x_8 + x_{10} + x_{11} + x_{12} + x_{17} + x_{19} + x_{21} + x_{22} + x_{23} + x_6x_{25} + x_9x_{15} + x_{18}x_{23} + x_{23}x_{26} + x_2x_{20}x_{21} + x_{13}x_{21}x_{23} + x_5x_{18}x_{19}x_{23}$$

$$7: x_0 + x_1 + x_2 + x_5 + x_{10} + x_{13} + x_{15} + x_{24} + x_{26} + x_7x_{22} + x_{11}x_{18} + x_{13}x_{19} + x_{16}x_{25} + x_{24}x_{25} + x_{15}x_{25}x_{26} + x_8x_{10}x_{25}x_{26}$$

References

1. N. G. deBruijn. *A combinatorial problem.* Indag. Math., 8(1946), pp. 461-467.
2. E. Dubrova. *A list of maximum period NLFSRs.* Cryptology ePrint Archive, 2012/166. www.iacr.org
3. C. Flye Sainte-Marie. *Solution to question nr. 48.* L'Intermédiaire des Mathématiciens 1(1894). pp. 107-110.
4. B. M. Gammel, R. Goetffert, O. Kniffler. *Achterbahn 128/80.* The eSTREAM project, www.ecrypt.eu.org/stream/, www.matpack.de/achterbahn
5. S. W. Golomb. *Shift Register Sequences.* San Francisco, Holden-Day, 1967, revised edition, Laguna Hills, CA, Aegean Park Press, 1982.
6. S. W. Golomb, G. Gong. *Signal Design for Good Correlation. For Wireless Communication, Cryptography, and Radar.* Cambridge University Press, 2005.
7. G. M. Kyureghyan. *Minimal polynomials of the modified de Bruijn sequences.* Discrete Applied Math., 156(2008), pp. 1549-1553.
8. R. Lidl, H. Niederreiter. *Introduction to Finite Fields and their Applications (Revisited Edition).* Cambridge University Press, Cambridge, 1994.
9. K. Mandal, G. Gong. *Probabilistic generation of good span n sequences from nonlinear feedback shift registers.* University of Waterloo, preprint, 2012.
10. G. L. Mayhew, S. W. Golomb. *Linear spans of modified de Bruijn sequences.* IEEE Trans. Inform. Theory, 36(5)(1990), pp. 1166-1167.
11. A. J. Menezes, P. C. van Oorschot, S. A. Vanstone. *Handbook of applied cryptography.* CRC Press, 1997.
12. J. Szmidt. *On Kyureghyan's Conjecture.* In preparation.
13. M. S. Turan. *On the nonlinearity properties of maximum-length NFSR feedbacks.* Cryptology ePrint Archive, 2012/112. www.iacr.org
14. R. Wicik, M. Borowski. *Randomness testing of some random and pseudorandom sequences.* Military Communication Conference, Prague, 2008.