LFSR Base Random Number Generator Design in 50nm Technology

Prof. Jaikaran Singh, Prof. Mukesh Tiwari, Shradha Sharma

Abstract – The random numbers are unpredictable numbers. These numbers are used in error checking and testing hardware for digital communication. It also uses to protect the medical, financial and personal data of entities connected to these networks. In this paper a four and eight bit linear feedback shift register is design using 50nm technology for random number generator. This generator outputs a predefined sequence of binary 1 and binary 0 with the same probability. A sequence of consecutive \( n^2(2n -1) \) bits comprise one data pattern, and this pattern will repeat itself over time.

Keywords – LFSR, PRBS, Maximum Length, Tapping.

I. INTRODUCTION

The pseudorandom number generator i.e PRBS generates the random sequence of binary numbers. Here the random word represents the value of an element of the sequence is not dependent of the values of any of the other elements. The pseudo word represents deterministic behaviour and after N elements it starts to repeat itself. A linear feedback shift register with its output tapping is feedback to input through xor gate is use to generates random number sequence.

A binary sequence (BS) is a sequence of N bits, \( a_j \) for \( j = 0, 1, ..., N - 1 \), i.e. m ones and Nm zeros. A binary sequence is pseudo-random (PRBS) if its autocorrelation function,

\[
C(v) = \sum_{a_j = a}^N (a_j a_j + v)
\]

has only two values:

\[
C(v) = m \text{ if } v = 0 \pmod{N}
\]

\[
C(v) = mc \text{ if } v \neq 0 \pmod{N}
\]

where

\[
c = (m - 1)(N - 1)
\]

is called the duty cycle of the PRBS.

The design of PRBS generator is based on the linear feedback shift register (LFSR), which consists of ‘n’ master slave flip-flops. The PRBS generator produces a predefined sequence of 1’s and 0’s, with 1 and 0 occurring with the same probability.

II. RELATED WORK (LITERATURE SURVEY)

In our base paper publish by Fabio Pareschi, Gianluca Setti, and Riccardo Rovatti in IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: REGULAR PAPERS, VOL. 57, NO. 12, DECEMBER 2010. They work on general features of an RNG and highlight the drawbacks of the classical solutions, showing how they can be overcome by the chaos-based approach. They also review pipeline ADCs, in particular the 11/2 bit per stage architecture which appears to be the best suited for the reuse of building blocks in chaos-based true random generators. They present the design and the validation by means of suitably improved randomness tests of two different implementations of high-performance true-random number generators which use a discrete-time chaotic circuit as their entropy source. The proposed system has been developed from a standard pipeline Analog-to-Digital converter (ADC) design, modified to operate as a set of piecewise-linear chaotic maps. The evolution of each map is observed and quantized to obtain a random bit stream. With this approach it is possible to obtain, on current CMOS technology, a data rate in the order of tens of megabit per second. Furthermore, author can also prove that the design is tamper resistant in the sense that a power analysis cannot leak information regarding the generated bits. This makes the proposed circuit perfectly suitable for embedding in cryptographic systems like smart cards, even more so if one consider that it could be easily obtained by reconfiguring an existing pipeline ADC.

The paper publish by Walter Aloiisi and Rosario Mita on title "Gated-Clock Design of Linear-Feedback Shift Registers" in IEEE TRANSACTIONS ON Circuits and Systems—II: Express Briefs, Vol. 55, No. 6, June 2008 did the work on LFSR present a method to reduce the power consumption of the popular linear feedback shift register. The proposed scheme is based on the gated clock design approach and it can offer a significant power reduction, depending on technological characteristics of the employed gates. Moreover, the analytical condition that must be satisfied to achieve a power reduction of the gated-clock circuit has been found. Theoretical analysis was validated through many transistor-level SPECTRE simulations in CADENCE environment by using the 0.35- m digital standard cells technology supplied by AMS. Simulation results have shown a power reduction of about 10% with a mean error of about 3% with respect to theoretical derivations. Theoretical analysis was validated through many transistor-level SPECTRE simulations by using the 0.35- m standard cells supplied by AMS. The efficiency of the proposed gated-clock LFSR design depends on the technological parameters of the adopted gates. Moreover, the concrete advantage of the novel design should be evaluated also considering the increased silicon area that is required to implement the gating circuit.

The paper published by David B. Thomas, and Wayne Luk on title "The LUT-SR Family of Uniform Random Number Generators for FPGA Architectures" at IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 21, No. 4, April 2013 did their work on Field-programmable gate array (FPGA) optimized random.
number generators (RNGs) are more resource-efficient than software-optimized RNGs because they can take advantage of bitwise operations and FPGA-specific features. However, it is difficult to concisely describe FPGA-optimized RNGs, so they are not commonly used in real-world designs. This paper describes a type of FPGA RNG called a LUT-SR RNG, which takes advantage of bitwise XOR operations and the ability to turn lookup tables (LUTs) into shift registers of varying lengths. This provides a good resource-quality balance compared to previous FPGA-optimized generators, between the previous high-resource high-period LUT-FIFO RNGs and low-resource low-quality LUTOPT RNGs, with quality comparable to the best software generators.

III. METHODOLOGY

The 4 bit and 8 bit linear feedback shift register is design using delay flip flop connected with a synchronize clock pulse. This flip flop is design with 50nm technology. The channel length of NMOS and PMOS transistor is 0.05um and channel width is 0.1um. The delay flip-flop is design using transmission gate logic. With the Microwind layout simulator:
1. Increase number of bits in Random Number generator by generating the sequence using a Linear feedback shift register.
2. Create maximum shift register length.
3. Long-period generators to be implemented using only a small amount of logic.

IV. IMPLEMENTATION

The delay flip flop is design using master slave delay latch arrangement. The four and eight number of delay flip-flop is series connected to form shift register with the same synchronize clock pulse. LFSR is an n-bit shift register which pseudo-randomly scrolls between 2n-1 values, but does it very quickly because there is minimal combinatorial logic involved. Once it reaches its final state, it will traverse the sequence exactly as before.

In nonlinear shift register the content of one flip flop is shifted towards the succeeding flip flop and binary '0' is entered on first flip flop. For example if a shift register contains the bit pattern 1101, a shift (to the right in this case) would result in the contents being 0110; another shift yields 0011. After two more shifts, things tend to get boring since the shift register will never contain anything other than zeroes. In nonlinear shift register the content of one flip flop is shifted towards the succeeding flip flop and binary '0' is entered on first flip flop. For example if a shift register contains the bit pattern 1101, a shift (to the right in this case) would result in the contents being 0110; another shift yields 0011. After two more shifts, things tend to get boring since the shift register will never contain anything other than zeroes. This shift register between parallel and serial data with delay of serial bit stream. This can be done by serial in and serial out data, serial in and parallelout data, parallel in and serial out data and parallel in and parallel out data. This is shown in fig 1.

V. FEEDBACK ACTION

In linear feedback shift register all flip flop are connected like nonlinear shift register and the result is fed back into the register's input bit. The output of flip flop is connected towards the input through xor logic gate, and the result of this feedback is inserted into the shift register sequence during the shift, filling the position that is emptied as a result of the shift.

Due to this feedback the register sequence loops through repetitive sequences of pseudo-random pattern. The selection of flip flop output taps determines length of the sequence before the sequence repeats. The content of LFSR and its flip flop outputs connected to XOR logic together describe the state of the LFSR. The content of LFSR will never be all zeros. In this case the LFSR will never change state.

VI. MAXIMUM LENGTH SEQUENCE

The maximum length sequence of LFSR means content of shift register sequences through all possible values except all of the bits being 0. The length of the sequence before repetition occurs depends upon two factors, the feedback taps and the initial state. An LFSR of n number of flip flops is capable of producing every possible state during the period N=2n-1 shifts, but will do so only if proper output of flip flop is feedback through XOR logic gate. For example, consider an eight stage LFSR which consist of a sequence of every possible combination of ones and zeros after 255 shifts. Such a sequence is called a maximal length sequence. There is no direct way to decide if a tap sequence is maximal length. But, there are some ways to show if one is not maximum length sequence: 1) For an even number of taps. 2) The tap values in sequence are all relatively prime e.g 12, 9, 6, 3.

Four Tap Sequence;
A tap sequence of [4, 1] describes the primitive polynomial
\[1 + X + X^4\]
Fig 2. CMOS layout design of 4 bit LFSR with tapping of $1+s^1+s^4$

Fig 3 shows the CMOS layout design of 4 bit LFSR with tapping of $1+s^1+s^4$. In this layout four D flip-flops are cascaded in serial in serial out configuration. The feedback tapping is taken from the first and fourth flip flops output and connected to the input terminal of first D flip flop input through XOR logic gate. Initially the first flip flop is set at logic level one and second, third and fourth flip flop is reset to logic level zero. After this initialization the output of each flip flop will shift to the next flip flop at each clock edge trigger. To set the flip flops, a set signal is pass through PMOS transistor.

VI. Conclusion

In an LFSR, the bits contained in selected positions in the shift register are combined in some sort of function and the result is fed back into the register's input bit. By definition, the selected bit values are collected before the register is clocked and the result of the feedback function is inserted into the shift register during the shift, filling the position that is emptied as a result of the shift. Feedback around an LFSR's shift register comes from a selection of points (taps) in the register chain and constitutes XORing these taps to provide tap(s) back into the register. Register bits that do not need an input tap, operate as a standard shift register. It is this feedback that causes the register to loop through repetitive sequences of pseudo-random value. The choice of taps determines how many values there are in a given sequence before the sequence repeats. The implemented LFSR uses a one-to-many structure, rather than a many-to-one structure, since this structure always has the shortest clock-to-clock delay path.

REFERENCES


AUTHOR’S PROFILE

Jaikaran Singh
Associate Professor, Department of ECE SSSIST, SEHORE.
Email: jksingh81@gmail.com

Mukesh Tiwari
Assistant Professor, Department of ECE SSSIST, SEHORE.
Email: mukesh.tiwari79@gmail.com

Shradha Sharma
PG scholar Dept. of ECE SSSIST, Sehore. My birth place is SHAHDOL (M.P) and date is 15/11/1987. I have my Bachelors’ of Engineering from Globus Engineering college Bhopal (M.P) in 2010 and now pursuing my M.Tech from SSSIST, SEHORE (M.P) from RGPV university.
Email: shradha.ece15@gmail.com