International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 36-42
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

RANDOM NUMBER GENERATION BY USING RESEEDING MIXING PSUEDO RANDOM NUMBER GENERATOR

C.M.VishnuRaj, Mr.S.Yuvaraj.
,Department of ECE, SRM University
Email: c.m.vishnuraj@gmail.com, yuvasivasanthi@gmail.com

ABSTRACT

My Paper Proposes that Pseudo random number generator (PRNG) has been widely used in Monte Carlo simulations, test pattern generation, cryptography, and telecommunication.[1][2] Linear PRNGs, such as linear feedback shift registers (LFSRs)[3][4], linear congruential generators (LCGs[5]), and multiple recursive generators (MRGs[6]), can produce long-period random number sequences. When implemented, linear PRNGs are efficient in throughput rate and hardware cost, but the output random numbers of such generators are predictable due to their linear structure. Some nonlinear PRNGs dealt with the predictability problem, but incurred higher hardware cost and more process time. Recently, nonlinear chaos-based PRNGs (CB-PRNGs)[7] with lower hardware cost were proposed. I present a new reseeding-mixing method to extend the system period length and to enhance the statistical properties of a chaos-based logistic map pseudo random number generator (PRNG)[3][4]. The reseeding method removes the short periods of the digitized logistic map and the mixing method extends the system period length to 2^253 by ―XORing with a DX generator.

Keywords: Chaotic map, mixing, period extension, pseudo random number generator (PRNG), reseeding,Linear Feedback shift Register (LFSR), Linear Congruential Generator (LCG), Carry-Look ahead Adder (CLA), Gate Equivalent (GE)

1. INTRODUCTION

Pseudo random number generator (PRNG) has been widely used in Monte Carlo simulations, test pattern generation, cryptography, and telecommunication systems. A good PRNG should have characteristics of: long-period random number sequence,a fit in statisticalproperties, ahighthroughput-rate andanunpredictability.

Linear PRNGs, such as linear feedback shift registers (LFSRs) , linear congruential generators (LCGs) , and multiple re- cursive generators (MRGs) can produce long-period random number sequences. When implemented, linear PRNGs are efficient in throughput rate and hardware cost, but the output random numbers of suchgeneratorsarepredictableduetotheirlinearstructure.Somenon- linear PRNGs in dealt with the predictability problem, but incurredhigherhardwarecostandmoreprocesstime.Recently,nonlinear chaos-basedPRNGs(CB-PRNGs) with lowerhardwarecostwereproposed.However,thereexistshortperiodsinsimpleCB-PRNG due to quantization error. The throughput performance of these CB-PRNGsisusuallylowduetothefactthattheycanonlyproduceone random bit in an iteration, and there is no assurance for the output sequences of these CB-PRNGs to have satisfactory statistical properties. In this brief, we propose a reseeding-mixing PRNG (RM-PRNG) that consists of a CB-PRNG and a long-period MRG. The reseeding methodre move sthe disadvantagesofshortperiodsinCB-PRNGwhile the mixing of the CB-PRNG with an MRG pushes the overall system period length to a value based on simple theoretical calculation. High throughput rate (6.4 Gb/s) is achieved by outputting multiple bits per iteration and the hardware efficiency is validated by using a TSMC 0.18µm CMOS process. Furthermore, good statistical qualities of the random numbers produced by our RM-PRNG are confirmed by SP800-22 tests.

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 36-42
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

2. EXISTING SYSTEM

I proposes a hardware oriented 80-bit-key binary additive stream cipher. The key stream generator consists of ten nonlinear feedback shift registers whose output sequences are combined by a Boolean function of algebraic degree four. The design size of the key stream generator is about 2200 GE. In 130 nm CMOS-technology, a throughput of more than 1 Gbps can be achieved. The length of the initial value used for resynchronization can be any multiple of eight between zero and eighty. The maximum amount of key stream that can be used between two resynchronization steps is 268 bits. A parallel implementation of the stream cipher produces one byte of key stream per clock cycle.

Existing System Algorithm

Specification of the Stream Cipher The basic idea behind the proposed stream cipher is to adopt the well proven design principle of the KSG in, but with the linear feedback shift registers replaced by suitable nonlinear feedback shift registers. The nonlinear counterparts of binary LFSRs with primitive characteristic polynomials are binary N-stage NLFSRs which will produce sequences of least period 2N - 1 for any nonzero initial state. Linear Complexity and Period of Key stream any two nonzero output sequences of a given primitive NLFSR are shifted versions of each other. The sequences therefore have the same linear complexity, which we call the linear complexity of the shift register.

3. EXISTING SYSTEM DRAWBACKS

- Hardware cost.
- Lower Throughput Rate
- unpredictable period length
- It's not sure that the random numbers produced by these mixed PRNGs will have acceptable statistical properties.

Proposed System

Pseudo random number generator (PRNG) has been widely used in Monte Carlo simulations, test pattern generation, cryptography, and telecommunication systems A good PRNG should have characteristics of:

1) Long-period random number sequence
2) Fit in statistical properties
3) A high throughput rate
4) Unpredictability

Linear PRNGs, such as linear feedback shift registers (LFSRs), linear congruential generators (LCGs), and multiple recursive generators (MRGs), can produce long-period random number sequences. When implemented, linear PRNGs are efficient in throughput rate and hardware cost, but the output random numbers of such generators are predictable due to their linear structure. Some nonlinear PRNGs dealt with the predictability problem, but incurred higher hardware cost and more process time. Recently, nonlinear chaos-based PRNGs (CB-PRNGs) with lower hardware cost were proposed. We present a new reseeding-mixing method to extend the system period length and to enhance the statistical properties of a chaos-based logistic map pseudo random number generator (PRNG). The reseeding method removes the short periods of the digitized logistic map and the mixing method extends the system period length to 2^253 by ―XORing ‖ with a DX generator.

4. PROPOSED SYSTEM EXPLANATION

Fig. 1 shows the schematic diagram of the RM-PRNG, which is composed of three modules: Nonlinear Module, Reseeding Module, and Vector Mixing Module.

Nonlinear module:

We use the LGM as the next-state construction function in the Nonlinear Module so that

Xt+1=F(Xt)= ɤ Xt(1-Xt) t≥0 →(1)

With ɤ =4 and X0 (0,1) as an initial seed. Choosing a value 4 for ɤ not only makes the LGM chaotic but also simplifies the implementation of (1) to merely left-shifting the product of Xt and (1-Xt) by 2 b[10]. However, the state size decreases from 32 to 31 b, because the dynamics Xt and (1- Xt) in (1) are the same. This is equivalent to a degradation of resolution by 1 b. In addition, fixed points (at Xt and 0.75) as well as short periods exist when the LGM is digitized.

Reseeding module:

The removal of the fixed points by the reseeding mechanism is obvious. When the fixed point condition is detected or the reseeding period is reached, the value Zt+1 loaded to the state register will be perturbed away from Xt+1 in the RCU by the fixed pattern according to the formula

$$Zt+1[j] = \{ Zt+1\, j\,.\ 1 \le j \le 32 - L;$$

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 36-42
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

$$Ri, 33 - L \leq j \leq 32, i = j + L - 32 \rightarrow (2)$$

Where subscripts i,j are the bit-index, L is integer, and R. In order to minimize the degradation of the statistical properties of chaos dynamics, the magnitude of the perturbation of the fixed pattern should be small compared with Xt. Here, we set L=5 so that the maximum relative perturbation is only
(25 - 1) /232and the degradation can be ignored. [5]

Vector mixing module:

An efficient MRG, called the DX generator, serves as the ALG in Vector Mixing Module. Specifically, we choose the DX generator[3][4] with the following recurrence equation:

Yt+1=Yt+BDX .Y → (3)

Using an efficient search algorithm, we find that the particular choice of BDX =228+28and M=231-1 gives the maximum period of the DX generator. The LSBs of Yt+1 and that of Xt+1 are mixed in the Output Construction unit [8][9]using a XOR operation to obtain the least significant bits of the output according to the equation

OUTt+1[1:31] ≤Yt+1[1:31] → (4)

Then, the most significant bit (MSB) of Xt+1 is attached to OUTt+1 [1:31] to form the full 32-b output vector OUTt+1.

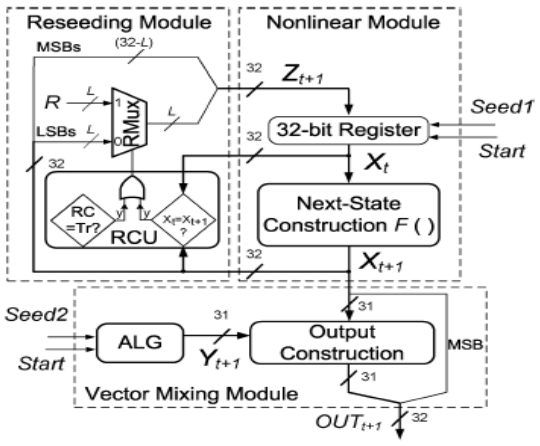

Figure. 1 Structure of the proposed RM-PRNG

5. PROPOSED SYSTEM TECHNIQUE

Pseudo Random Number Generator

RM-PRNG schematic diagram is shown in Fig. 1, which is composed of three modules: Nonlinear Module, Reseeding Module, and Vector Mixing Module. In a 32-b implementation, the Nonlinear Module has a controlled 32-b state register and a Next-State construction circuitry[8]. The controlled register stores the state value Xt which can be set to Seed1 by the Start command. The Next-State construction circuitry produces the next state value Xt+1 according to the recursive formula Xt+1=F(Xt)[9]. For each generated state value, the reseeding control unit (RCU) in the Reseeding Module compares the values of Xt+1 and Xt for checking the fixed point condition (Xt+1=Xt), and increases the reseeding counter (RC) at the same time. The RC will be reset and the reseeding operation will be activated when either the fixed point condition is detected or the RC reaches the reseeding period Tr. When reseeding is activated, the state register will be loaded through the reseeding multiplexer (RMux) with a value. Otherwise, the value of Xt+1 [11]is directly loaded into the state register. The output of the proposed RM-PRNG is obtained by mixing Xt+1 with the output Yt+1 from an auxiliary linear generator (ALG) in the Vector Mixing Module according to the rule.

Fig. 1 Structure of the proposed RM-PRNG

Proposed system advantages:

- High Throughput Rate
- Less hardware cost

Digitization

Digitization is a process in which the map G: y→ y is replaced with the map F:x → x. Digitization is not a unique process. However, in many cases one can identify "a natural way" in doing this[6][9]. Thus, for example, if $\beta = \{c_0, \ldots, c_{2m} - 1\}$ is a finite partition of the phase space y, then $x = \{0, \ldots, 2^m - 1\}$and F is the restriction of G on x (assuming that such restriction exists).In our work, we make digitization of the logistic map as follows:

First method: Firstly, the chaotic sequence $\{X_k\}$is generated through Equations (1), which has to be amplified by a scaling factor (10^4) and round to integer-sequence according to Equations

$$Z_k = \text{round}((x_k * 10^4) \bmod 256) \rightarrow (5)$$

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 36-42
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

This transformation implies that, when the randomly generated chaotic sequence (input values) is uniformly distributed, the output of the digitization process is also uniformly distributed Random numbers have been used extensively in many simulation applications like Monte Carlo Integration or computer modeling. But recently security applications have increased the need for strong (secure) random number generation like automatic password generation, encryption algorithms, on-line gambling etc. Thus random number generation has become a challenging and an interesting task. Most classical random number generators, generate sequences that are either linear or predictable hence not suitable for cryptographic and security applications. Others generate sequences that even though they are secure they are not cryptographically strong and above all are slow in execution. Also recent advances in random number generation like the construction of Multiple Recursive Generator (MRG) with large orders, Fast Multiple Recursive.

Generator (FMRG) and DX (system of multiple recursive generators proposed by Deng and Xu generators does not generate a strong random number sequences. Though MRGs have extremely long period of length with good empirical performance, its recurrence equation can be solved given a small set of its generated sequence, this implies that MRGs and FMRGs are not strong cryptographic generators[8]. We propose an algorithm that will transform linear sequences generated by both classical LCG, MRGs, FMRGs and DX generators and make them cryptographically strong generators by hiding the entire sequence generated by the generators, thus it will be difficult for cryptanalyst to predict or infer the generator sequence if even the partial sequence or the parameters or knowledge of the algorithm used in the transformation of the generators are known[2][3].

Experimental results and tests have shown that classical generators like LCGs that generate pseudorandom linear sequences are not suitable for cryptographic purposes, even though it is simple, efficient and easy to generate. Other classical generators like BBS, RSA, and BM[4][5]etc that are thought to be secure are equally not good enough for cryptographic purposes as they are slow in generating the next random bit sequence. Also the recent advances in random number generation (MRGs and FMRGs) are fast and efficient in generating linear sequences with long periods and good empirical performance, but still they are not cryptographically strong as the linear system can be predicated using a system of unique k equations. Our proposed algorithm produces a strong pseudorandom sequence that is suitable for cryptographic purposes and difficult to predict/infer by transforming the linear sequences and breaking its linear structure. The transformation hides the linear bits of the generated linear sequence preventing the attacker from accessing the generated output sequence, even with the knowledge of the partial sequence, parameters of the generators and the algorithm used in transforming the generator sequence. Thus knowing the parameters and partial sequence of the generators does not pose any threat any longer as the prediction of the generator sequence will no longer be an easy one.

6. CHAOS THEORY and CHAOTIC SYSTEM

- Nonlinear transformation
- Infinite number of states
- Infinite number of iterations

1.Initial condition
2.Final state
3.Initial conditions and parameters

Pseudo-Chaotic and Cryptographic Systems

This CHAPTER studies on cryptographic systems based on finite-state approximations of chaos (i.e. pseudo-chaos). Two approaches are considered to store the system state on a computer: (i) the floating-point format of real numbers and (ii) 'plain' binary strings or m-dimensional cubes.

7. LOGISTIC MAP

A similar transformation has become of the most famous chaotic maps. In 1976, Mitchell Feigenbaum studied the complex behavior of the so-called logistic map

$$x_{n+1} = 4rx_n(1 - 2x_n), \qquad \rightarrow(6)$$

where x $x\varepsilon(0,1)$ and $r\varepsilon(0,1)$For any long sequence of N numbers generated from the seed $_{x0}$ we can calculate the Lyapunov exponent given by

$$\lambda(x_0) = \frac{1}{N}\sum_{n=1}^{N} \log|r(1 - 2x_n)| \quad \rightarrow(7)$$

For example, the numerical estimation for r = 0.9 and N = 4000 is $\lambda(0.5) \approx 0.7095$.

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 36-42

available at : www.ijcns.com
ISSN: 0976-268X

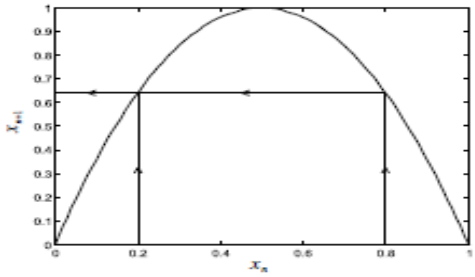

Figure 2 - The logistic map for r = 0.99.

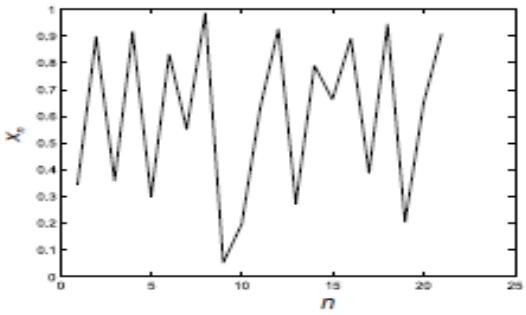

Figure 3- A chaotic sequence generated with the logistic map for x0 = 0.34 and r = 0.99.

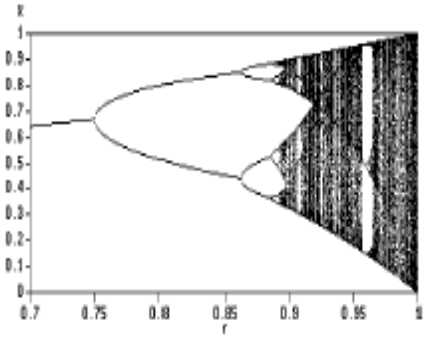

Figure 4 - Bifurcation of logistic map.

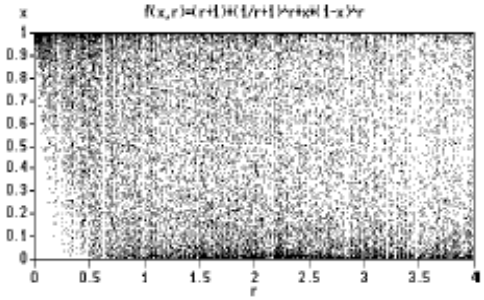

Figure 5 - Attractor points corresponding to different values of the parameter r in the Matthews map

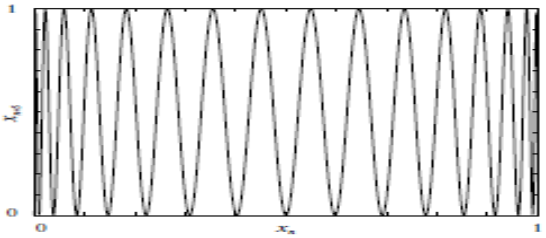

Figure 6 - The analytical solution at n = 5 of the logistic map for r = 1

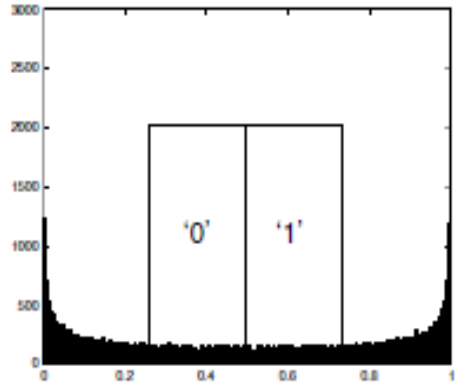

Figure7 - The Probability Density Function of a state sequence produced by the Logistic system with an incomplete partition

With certain values of the parameter r, the generator delivers a sequence, which appears pseudo-random .The Freigenbaum shows the values of $_{xn}$ on the attractor for each value of the parameter r. As r increases, the number of points in the attractor increases from 1 to 2, 4, 8 and infinity. In this area (r ! 1) it was considered difficult to estimate the final state of the system (without performing n iterations) given initial conditions $_{x0}$, or vice-versa - to recover $_{x0}$ (which can be a key or a plaintext) from $_{xn}$. This complexity was regarded as a fundamental advantage in using continuous chaos for cryptography.

For the boundary value of the control parameter r = 1 the analytical solution is:

$$x_n = sin^2\,(2^n \arcsin\sqrt{x_0}\,) \qquad \rightarrow(8)$$

Hence, the state $_{xn}$ can be computed directly from $_{x0}$ (without performing n iterations. Figure 4.11 gives the solution for $_{x5}$ = f5 ($_{x0}$). The number of spikes increases with n, illustrating the sensitivity to the initial conditions. the logistic map to generate a sequence of floating point numbers, which is then converted into a

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 36-42
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

binary sequence. The binary sequence is XOR-ed with the plain-text, as in the one-time pad cipher The parameter r together with the initial condition x_0 form a secret key.

The conversion from floating point numbers to binary values is done by choosing two disjoint interval ranges representing 0 and 1. The ranges are selected in such a way, that the probabilities of occurrence of 0 and 1 are equal Note, the equiprobable mapping does not ensure the uniform distribution. Though the numbers of zeros and ones are equal, the order is not random. It has been pointed out by Wheeler [95] and Jackson [60] that computer implementations of chaotic systems yield surprisingly different behavior, i.e. it produces very short cycles and trivial patterns .Mathews [75] generalizes the logistic map with cryptographic constraints and develops a new map to generate a sequence of pseudo-random numbers

$$x_{n+1} = (r+1)\left(\frac{1}{r}+1\right)^{r} . x_n(1 - x\_n)^{r} , r \in (1,4) \rightarrow (9)$$

8. CONCLUSION

The proposed method is a hardware implementation of RM-PRNG to offer long periods and high throughput rate while adhering to established statistical standards for PRNGs. The reseeding mechanism solves the short-period problem originated from the digitization of the chaotic map, while mixing a CB-PRNG with a long-period DX generator extends the period length to the theoretically calculated value greater than 2^{253}. Replacing a hardware-demanding CB-PRNG with a hardware-efficient MRG, the hardware cost is reduced and the hardware efficiency achieves 0.538 Mb/s-gate. In addition, the high throughput rate [3][10](> 6.4 Gb/s) is attained because RM-PRNG can generate multiple random bits in an iteration. For randomness enhancement, the proposed reseeding-mixing method successfully improved the statistical properties of CB-PRNG and the random number sequences generated by the proposed RM-PRNG pass all the tests in NIST SP 800-22 test suite. With all these advantages, the proposed nonlinear RM-PRNG can a good candidate for potential applications in test pattern generation, telecommunication system and even cryptography if the security issue can be addressed properly.

FUTURE ENHANCEMENT:

By using Reseeding-Mixing method the proposed project will give higher throughput and lower hardware cost. And by using our PRNG, we can design the encryption decryption circuits.

REFERENCES

[1] J. E. Gentle, Random Number Generation and Monte Carlo Methods, 2nd ed. New York: Springer-Verlag, 2003.

[2] M. P. Kennedy, R. Rovatti, and G. Setti, Chaotic Electronics in Telecommunications. Boca Raton, FL: CRC, 2000.

[3] D. Knuth, The Art of Computer Programming, 2nd ed. Reading, MA: Addison-Wesley, 1981.

[4] A. Klapper and M. Goresky, "Feedback shift registers, 2-adic span, and combiners with memory," *J. Cryptology*, vol. 10, pp. 111–147, 1997.

[5] D. H. Lehmer, "Mathematical methods in large-scale computing units," in *Proc. 2nd Symp. Large Scale Digital Comput. Machinery, Cambridge*, MA, 1951, pp. 141–146, Harvard Univ. Press.

[6] P. C. Wu, "Multiplicative, congruential random-number generators with multiplier $\pm 2^{k1} \pm 2^{k2}$ and modulus ," *ACM Trans. Math. Software*, vol. 23, pp. 255–265, 1997.

[7] L. Y. Deng and H. Xu, "A system of high-dimensional, efficient, longcycle and portable uniform random number generators," ACM *Trans.ModelComput. Simul.*, vol. 13, no. 4, pp. 299–309, Oct. 1, 2003.

[8] L. Y. Deng, "Efficient and portable multiple recursive generators of large order," ACM Trans. Modeling Comput. Simul., vol. 15, no. 1, pp. 1–13, Jan. 2005.

[9] L. Blum, M. Blum, and M. Shub, "A simple unpredictable pseudo- random number generator," SIAM J. Comput., vol. 15, pp. 364–383, 1986.

[10] B. M. Gammel, R. Goettfert, and O. Kniffler, "An NLFSR-based stream cipher," in Proc. IEEE Int. Symp. Circuits Syst., 2006, pp. 2917–2920.

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 36-42
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

[11]D.Mukhopadhyay,D.R.Chowdhury,andC.Rebeiro,“ Theory of composing non-linear machines with predictable cyclic structures,”inProc. 8th Int. Conf. Cellular Autom. Res. Ind., 2008, pp. 210–219, Springer.

[12]D.Mukhopadhyay,“Groupproperties ofnon-linearcellularautomata,” J. Cellular Autom., vol. 5, no. 1, pp. 139–155, Oct. 2009.