ISSN: 2395-1680 (ONLINE)
DOI: 10.21917/ijme.2016.0025

FPGA IMPLEMENTATION OF A PSEUDO NOISE SEQUENCE USING CHAOTIC TENT MAP FOR SATELLITE COMMUNICATION

K.L. Sudha[1], A. Rajagopal[2] and Dundi Ajay[3]
Department of Electronics and Communication Engineering, Dayananda Sagar College of Engineering, India
E-mail: [1]klsudha1@rediffmail.com, [2]gopiluck@gmail.com, [3]dundi.ajay@gmail.com

Abstract

Pseudo Noise (PN) sequences are an integral part of satellite navigation system. In the past few years' modernization of these systems had led to the addition of new frequency signals or bands such as the L1c and L5 signals thereby necessitating new PN sequences. Chaos based binary PN sequences have been used mainly for cryptographic applications and have not been experimented for other applications. In this paper one such chaos based function namely the Tent map is studied. The Tent map is implemented using shift registers and the output is further modified to obtain optimum correlation values. The correlation properties are studied and compared against Gold sequences. Finally a Verilog hardware description language code is written for implementation on a Field Programmable Gate Array (FPGA) evaluation kit.

Keywords:

Pseudo Noise Sequence; Tent Map; Shift Registers; Correlation; FPGA

1. INTRODUCTION

Pseudo Noise (PN) sequences are defined as a binary coded sequence of 1's and 0's having properties which make them appear noise like but, can be reproduced exactly at some other time and location. The study of PN sequences has spanned close to 60 years and in all these years various sequences have been developed and their properties for various applications such as spread spectrum, cryptography, radar, etc. have been studied. Satellite navigation systems such as Global Positioning Systems (GPS) have used PN sequences based on linear feedback shift register such as maximal length sequences and Gold sequences. The data to be transmitted are modulated by these shift register sequences and transmitted on the L1 (1575.42MHz) and the L2 (1227.60MHz) channel. The length of the PN sequences for the L1 channel is 1023 bits. Gold sequences was the preferred choice as PN sequences for the L1 channel considering the ease of generation using shift registers (length of shift registers = 10, thus 210-1 = 1023), adequate number of sequences of length 1023 bits and also the sequences had good correlation properties meaning less interference among user's data. Modernization of GPS systems has led to the addition of new channels such as L1c (1575MHz) and L5 (1575.42MHz) which require PN sequences to be of length 10230 bits. Gold sequences exist only for preferred pair of primitive polynomials and their length is in terms of powers of 2, hence these sequences are not ideal candidates for L1c or the L5 channel. This has led to the search of new sequences and one of them being chaos based sequences.

Chaos is an everywhere phenomena which broadly arises from non-linear systems. A chaotic system is defined as a system whose output are neither periodic nor converge to a fixed point, they are sensitive to initial conditions of the system and are very hard to predict by an eavesdropper. Many methods to generate PN sequences from chaotic maps have been proposed in literature. The application of chaotic sequences for spread spectrum was first explained in [1] and their suggestion to overcome the periodicity of chaotic sequences in digital systems is very convincing. The authors in [2] have proposed to use a logistic map sequence for direct sequence spread spectrum and are of the opinion that logistic map does not offer much improvement over maximal length sequence when used in a multi-user environment, but when security and implementation are concerned logistic map sequences offer a promising picture. Bernoulli and Tent map implementation using extended-Non Linear Shift Registers (e-NFSR) is proposed in [3]. Taking cognizance of the work done in [3], authors in [4] have approximated a Tent map sequence using primitive polynomials and a tent-like matrix and demonstrated that for long lengths the approximated tent map sequence does have binary valued auto-correlation. The authors in [5] have generated spreading sequences based on tent map of length 10230 bits and have proposed the use of this sequence for the Galileo satellite system. An FPGA implementation of a PN sequence using tent map and logistic map is shown in [6]. The authors have used a digitization block and rounded off the real number sequence to convert to a binary PN sequence.

In this paper the chaos based PN sequences as described in [5] is implemented by using extended Linear Feedback Shift Registers (e-LFSR) on an FPGA and the auto and cross correlation properties are compared against Gold sequences of similar length i.e. 10230 bits. The rest of the paper is organized as follows; chaotic tent map is explained in section 2, the shift register structure of the map along with the their properties are presented in section 3, FPGA implementation results are shown in section 4 and finally section 5 concludes the paper.

2. TENT-CHAOTIC MAP

For a signal to be classified as chaotic, it must have the following properties [7]:

- The signal must be deterministic so that it can be reproduced exactly at any given time without any variation.
- The signal must be sensitive to initial conditions so that any minute change in the values should produce totally different signal.
- The signal must be noise like random in nature so that it is mistaken for noise.

"Sensitive dependence on initial conditions" requires that trajectories originating from very nearly identical initial conditions will diverge exponentially quickly. This phenomenon is common to chaos theory. Just a small change in the initial conditions can drastically change the long-term behavior of a

system. In this work chaotic systems are represented as dynamic discrete systems meaning that for each point x_i we have $x_{i+1} = F(x_i)$. F is the mapping from $R \rightarrow R$, where R is the set of real numbers.

Chaotic signals are not binary valued signals hence in order to use them in spread spectrum application they have to be converted into binary signals. There are various methods to convert them to binary signals and one such method is the threshold method where the threshold is the mean of the chaotic values generated by the map. That is, let 'w' be a real valued chaotic sequence. The binary chaotic sequence is now defined as,

$$B_w(i) = \begin{cases} 0; & \text{if } w(i) < \text{threshold} \\ 1; & \text{if } w(i) \geq \text{threshold} \end{cases}; \quad i \text{ is the index} \tag{1}$$

The Tent chaotic map is given as follows:

$$x_{n+1} = \begin{cases} ax_n; & 0 \leq x_n < 0.5 \\ a(1-x_n); & 0.5 \leq x_n \leq 1 \end{cases} \tag{2}$$

where, x_0 is the initial state, $x \in [0, 1]$ and 'a' $\in [1, 2]$ are the control parameters. The parameter 'a' is also known as the bifurcation parameter where bifurcation is the abrupt change in the qualitative behavior of a system. The bifurcation plot is used for calculating the initial condition of the function as proper selection of it leads to a random chaotic sequence. The bifurcation plot for the above tent map is as shown in Fig.1.

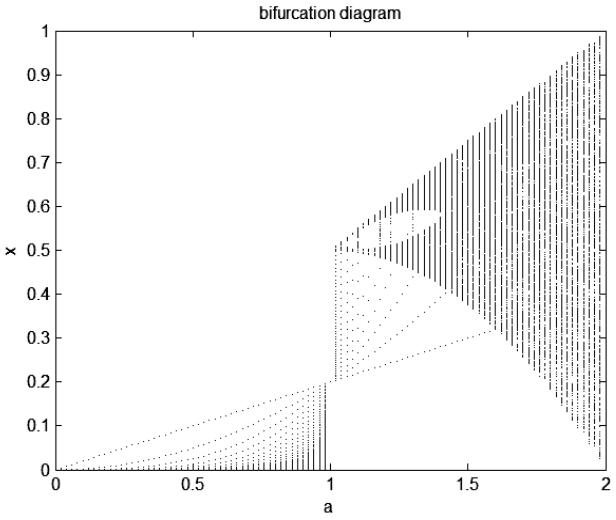

Fig.1. Bifurcation plot of tent map

From the above plot it can be seen that setting the bifurcation parameter 'a' closer to 2 leads to random sequences. To understand the sensitivity of the tent map to initial conditions, Fig.2 shows two binary sequences each of length 100 bits labelled as PN sequence-1 with 'a' = 1.995 and x_0 = 0.100 and PN sequence-2 with 'a' = 1.995 and x_0 = 0.101. The noise like waveform is the sequence of real numbers obtained from Eq.(2) and using Eq.(1) the real numbers are converted to 1's and 0's.

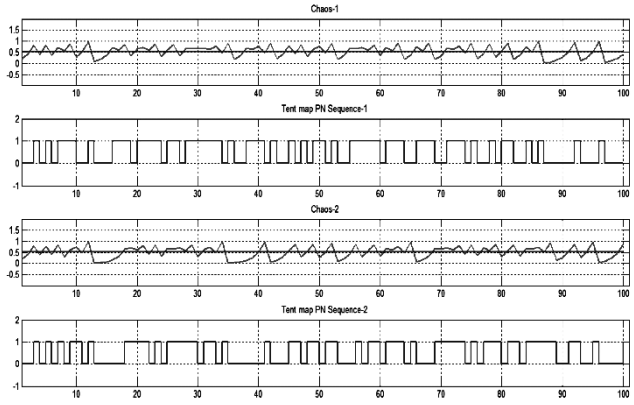

Fig.2. Chaotic sequences for different initial conditions

One of the disadvantages of chaotic maps is the sensitivity of the map to initial conditions since a small change in initial condition will lead to a completely different PN sequence. Also different hardware PN sequence generators at the transmitter and the receiver will lead to different PN sequences even for the same initial conditions. Hence to overcome the above difficulties the tent function in Eq.(2) can be approximated by using shift registers.

3. SHIFT REGISTER STRUCTURE OF TENT MAP

PN sequences such as maximal length sequences or M-sequences are generated by means of Linear Feedback Shift Register (LFSR).

A primitive polynomial $P(x) = x^m + p_{m-1}x^{m-1} + p_{m-2}x^{m-2} + \ldots\ldots + p_1x + 1$ of degree 'm' is required to construct M-sequences of length $N = 2^m$-1. The LFSR structure of M-sequences can also be represented by means of a matrix equation as [4]:

$$y'_{n+1} = By'_n \tag{3}$$

where, $y'_n = (y_{n,0}, y_{n,1}, \ldots . y_{n,m-1})^T$ is the state of the shift register at time 'n' and '**B**' is a matrix defined such that,

$$\mathbf{B} = \begin{bmatrix} p_{m-1} & p_{m-2} & \cdots & p_1 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}$$

For an arbitrary 'm' the matrix **B** which describes a tent map can be written as follows:

$$\mathbf{B}_{\text{tent map}} = \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,m-1} & b_{1,m} \\ 1 & 0 & \cdots & 0 & 1 \\ 0 & 1 & \cdots & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 1 \end{bmatrix}$$

where,

$b_{1,1} = p_{m-1} + 1$

$b_{1,2} = p_{m-2} + p_{m-1}$

$b_{1,3} = p_{m-3} + p_{m-2}$

.

.

$b_{1,m} = p_1$

As an example consider a primitive polynomial $p(x) = x^4 + x + 1$, then the matrix **B** for tent map will be:

$$\mathbf{B}_{\text{tent map}} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

Hence the shift register structure for the above matrix is as shown in Fig.3.

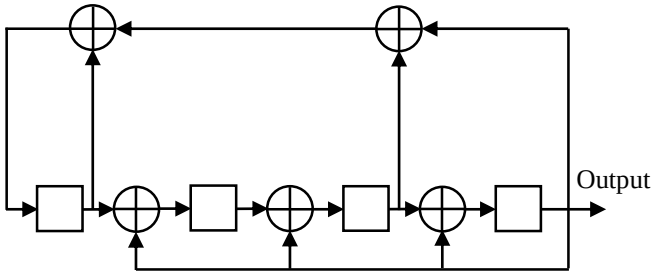

Fig.3. Shift register structure of tent map sequence of length 15 bits

The initial state of the shift register should not be all zero state and the output sequence is periodic with $N = 2^4-1 = 15$ bits. If the states of the register are converted to decimal representation and a graph of next state against the present state is plotted, it results in the shape of a tent. Hence the name tent map. The plot is as shown in Fig.4.

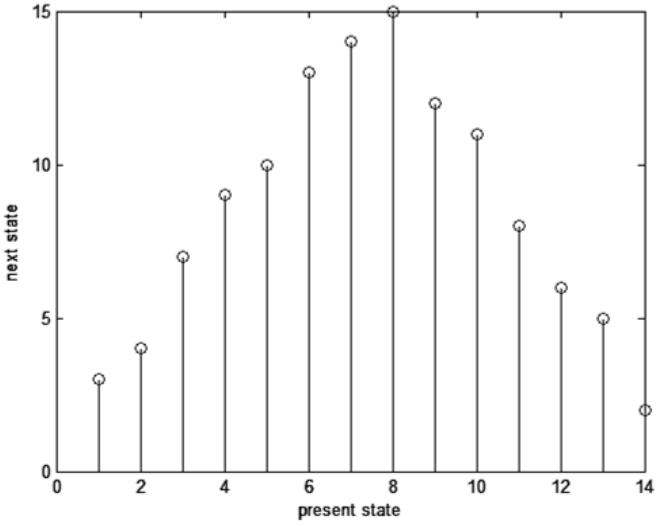

Fig.4. Return map of Tent chaos map

Procedure to generate the modified sequence:

- To generate the tent map sequence of length 10230 bits, two primitive polynomials $p_1(x) = x^{14} + x^{10} + x^6 + x + 1$ and $p_2(x) = x^{14} + x^8 + x^6 + x + 1$ are chosen. The choice of the polynomial is random.
- The chaos sequence obtained from $p_1(x)$ and $p_2(x)$ which are represented as $C1$ and $C2$ respectively are truncated to 10230 bits. The sequences are generated by shift registers as explained previously. The cross-correlation [8] between $C1$ and $C2$ is calculated. The cross-correlation should exhibit one predominant peak for a delay of $L = 10230/2=5115$.
- Corresponding to this delay, the first 'L' bits of $C2$ are inverted and the remaining bits are retained as they are. This new sequence is now represented as $C3$.

Once $C3$ is formed, new PN sequences D_k are formed from the equation:

$$D_k = F(C1) + T^k(C3) + F(C3) \text{ for } k = 1,2,\ldots \quad (4)$$

In Eq.(4), 'F' denotes flipping operator, and T^k denotes cyclically shifting the sequence either to the right or to the left by 'k' bits. The '+' operator is modulo-2 addition. Once these new sequences are formed the cross-correlation is calculated between D_k and $C1$ and $C3$. If the cross-correlation values are less than a predefined value then D_k is added to the set of $C1$ and $C3$ sequences else 'k' is incremented and the next sequence is computed. The predefined value here is set to the maximum cross-correlation value between $C1$ and $C3$.

The Fig.5 shows the comparison of auto correlation function (ACF) values between tent map sequence and Gold sequence of length 10230 bits. Gold sequences are formed by shift and an exclusive-or of m-sequences. The Fig.5 shows only three sequences of each type for representation purpose. The correlation values in dB scale are calculated as

$$\text{Correlation (dB)} = 10\log_{10}(\text{corr}^2/\text{length}^2) \quad (5)$$

In Eq.(5) 'corr' indicates the maximum value of correlation values in linear scale.

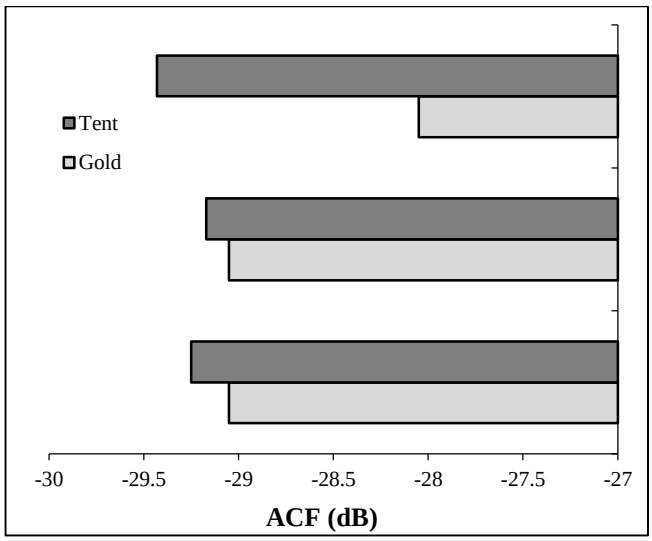

Fig.5. ACF comparison of Gold and tent sequences

It can be seen from the Fig.5 that the auto-correlation values of the Tent sequences are much better than Gold sequences. Similarly Fig.6 shows the cross-correlation function (CCF) comparison.

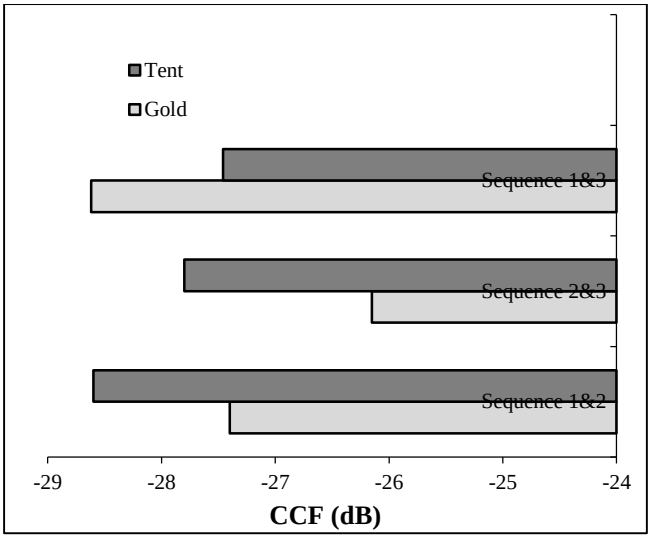

Fig.6. CCF comparison of Gold and tent sequences

It can be seen from Fig.6 that the auto and cross correlation values are much better for tent map sequences compared to Gold sequences. We were able to obtain adequate number of tent sequences whose correlation values were significantly better than that of Gold sequences. Since tent chaos sequences are also much harder to predict for an eavesdropper, these sequences can also be used for anti-jamming applications.

4. FPGA IMPLEMENTATION

The tent map sequence generator is implemented on Spartan-6 series FGPA xc6slx45. In this design we use Verilog Hardware Description Language (HDL) to implement the design on the FPGA. Since the design involved simple operations such as flipping and cyclically shifting the bits, the hardware resources was minimum as can be seen from Table.1. The value of the delay was fixed in the code to $L = 5115$.

Table.1. FPGA device utilization summary

Slice registers	67 out of 54576
Slice LUT's	79 out of 27288
Number of bonded IOB's	5 out of 218
Number of BUFG/BUFGMUX's	2 out of 16
Maximum frequency	279.252MHz
Target device	xc6slx45-2csg324

Part of the Register Transfer Logic (RTL) of the generator is shown in Fig.7. It can be concluded from Table.1 that the sequence generator is much simpler to generate than compared to the generator in [6]. The method to generate these sequences using shift registers is also much faster than the one proposed in [6].

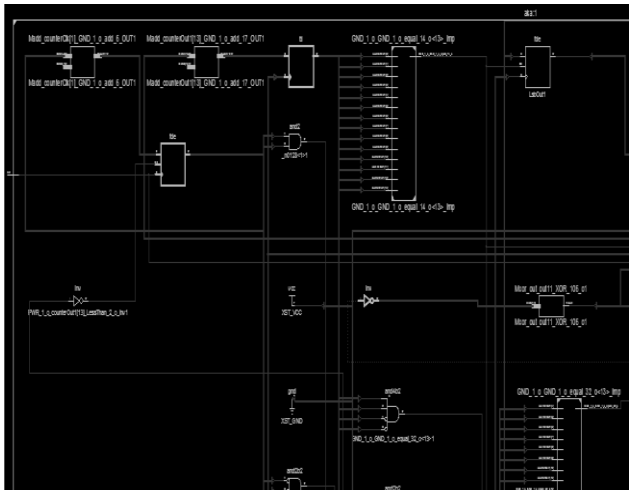

Fig.7. RTL schematic of the generator

5. CONCLUSION

In this paper, a chaotic binary sequence based on tent map is implemented on an FPGA. The tent map was chosen over other chaotic maps for its simplicity and robustness. The tent sequence obtained using shift registers was further modified using simple operations to obtain optimum correlation values which are essential for satellite navigation systems. The results indicated that in-spite of the truncation of the sequence the correlation properties observed were much better than compared to Gold sequences which have been the preferred choice of PN sequences over all these years. The shift register method of generation was simpler than the one proposed in [6] and it overcomes the drawback of these chaotic sequences when it came to synchronization at the receiver. It can therefore be concluded that these sequences are one of the possible candidates in future for satellite navigations systems.

ACKNOWLEDGMENTS

This work is supported by ISRO RESPOND programme [grant no. ISRO/RES/3/651/2013-14]. The authors would also like to thank the following people for their support during the work: Anjani Garg, Kislay and Mathew Junu Joy.

REFERENCES

[1] G. Heideri-Betani and C.D. McGillem, "A Chaotic Direct Sequence Spread Spectrum Communication System", *IEEE Transactions on Communications*, Vol. 42, No. 234, pp. 1524-1527, 1994.

[2] J. Tou, P. Yip and H. Leung, "Spread Spectrum Signals and The Chaotic Logistic Map", *Circuits Systems and Signal Processing*, Vol. 18, No. 1, pp. 59-73, 1999.

[3] Daisaburo Yoshioka, Akio Tsuneda and Takahiro lnoue, "On Transformation between Discretized Bernoulli and Tent Maps", *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. E88-A, No. 10, pp. 2678-2683, 2005.

[4] Alexander L. Baranovski, Frank Dachselt and Wolfgang Rave, "Nonlinear Dynamics of PN-Sequences",

Proceedings of 14th IST Mobile and Wireless Communications Summit, pp. 19-22, 2005.

[5] Mahmoud Hadef, Josh Reiss and Xiaodong Chen, "Chaotic Spreading Codes and their Generation", US 8085749, December 27, 2011.

[6] Himan Khanzadi, Mohammad Eshghi and Shahram Etemadi Borujeni, "Design and FPGA Implementation of a Pseudo Random Bit Generator Using Chaotic Maps", *IETE Journal of Research*, Vol. 59, No. 1, pp. 63-73, 2013.

[7] Peter Stavroulakis, "*Chaos Applications in Telecommunications*", Taylor & Francis Group, 2006.

[8] D.V. Sarwate and M.B. Pursley, "Crosscorrelation Properties of Pseudorandom and related Sequences", *Proceedings of the IEEE*, Vol. 68, No. 5, pp. 593-619, 1980.