Logic Designs based on Pseudo-Random Binary Codes for the Natural Code Converter

Goran Miljkovic[1], Dragan Denic[2], Milan Simic[3] and Aleksandar Jocic[4]

[1]Faculty of Electronic Engineering University of Niš
Aleksandra Medvedeva 14
18000 Niš, Serbia
goran.miljkovic@elfak.ni.ac.rs

[2]Faculty of Electronic Engineering at University of Niš
Aleksandra Medvedeva 14
18000 Niš, Serbia
dragan.denic@elfak.ni.ac.rs

[3]Faculty of Electronic Engineering at University of Niš
Aleksandra Medvedeva 14, 18000 Niš, Serbia
milan.simic@elfak.ni.ac.rs

[4]Faculty of Electronic Engineering at University of Niš
Aleksandra Medvedeva 14, 18000 Niš, Serbia
aleksandar.jocic@elfak.ni.ac.rs

ABSTRACT: *The systems and their components have the natural code convertors that use the pseudo-random binary code. In simplicity and hardware implementation, the natural code convertors have gains. We have used the Galois generator to apply the serial natural code convertor to reduce conversion time. The ideal way is to use a logic design that is important for proper functioning of Galois based serial code converter. Now we have given details of the functionality of serial code converter through simulation examples. Moreover, we have given detailed timing analysis of Fibonacci and Galois based serial pseudorandom/ natural code converter.*

Keywords: Pseudorandom Binary Sequence (PRBS), Pseudorandom/Natural Code Converter, Fibonacci and Galois Architecture for Generation of PRBS Code

Received: 29 May 2021, Revised 25 July 2021, Accepted 4 August 2021

DOI: 10.6025/jdp/2021/11/4/113-118

1. Introduction

The absolute pseudorandom position encoders are well-known digital transducers which could be used for absolute position measurement in industry, robotics, electrical power engineering, elevators, telescopes/antennas, printers, etc. Absolute posi-

tion determination in these encoders is based on property of n-bit pseudorandom binary sequence (PRBS) that sliding window of length n, which passes along a sequence, will extract unique code word in every moment [1, 2]. Also, the code words are now longitudinally lined, and two successive code words are overlapping and differ only in one bit. The PRBS signal is exploited in other fields, cryptography, telecommunications, testing of VLSI circuits [3], testing of gas sensors [4], measurement of the frequency response, etc. The absolute pseudorandom position encoder consists from following functional parts: code reading system [2, 5], code scanning [2, 6, 7], error detection [8] and pseudorandom/natural code conversion [2, 9, 10].

The pseudorandom/natural code conversion method is compromise between two opposite requests, minimal conversion time and low complexity and price of used hardware. The different pseudorandom/natural code conversion methods can be sorted to three distinct groups: parallel [7], serial [2, 9] and serial-parallel [7]. The parallel pseudorandom/natural code conversion is the fastest, but requests large ROM memory for storing of code conversion table especially in the case of high-resolution encoder. The serial pseudorandom/natural code conversion is simplest, less hardware expensive, but slower method. The compromise solution is the serial-parallel code converter, which combines previous two methods.

In the paper are detailed described improved solution of serial pseudorandom/natural code converter based on Galois generator of PRBS which can be used in absolute pseudorandom position encoder or in other fields where PRBS is applied. The presented serial pseudorandom/natural code converter is then implemented in software NI Multisim, integrated capture and simulation design environment. These simulations are used for functional and timing analysis of serial pseudorandom/natural code converters.

2. Improved Serial Pseudorandom/Natural Code Converter

Duration of pseudorandom/natural code conversion process, the conversion time, becomes critical at high resolution absolute pseudorandom position encoders. The serial pseudorandom/ natural code converters are based on the fact that is possible to find the actual value of the position p by counting of the steps that shift register with reverse feedback needs until it reaches the initial state by successive shifting from the read pseudorandom n-bit code word. One solution is the serial pseudorandom/ natural code converter based on Fibonacci generator [9]. Improved solution is based on faster Galois generator of PRBS which can be used in serial code converter instead of Fibonacci generator [9]. This generator also uses a shift register, the content of which is modified at every step by a binary-weighted value of the output stage, using XOR gates. The improved solution of serial pseudorandom/natural code converter based on Galois generator can be seen in Fig. 1. A 31-bits long PRBS generated by a 5-bit shift register with the feedback set for direct generation law [5,4,3,1] is used for the encoding of the code track. It consists of the following bits: 11100001101010010001011111101100. The read pseudorandom code bits with one code reading head x5 are loaded into five-cell bidirectional code assembly register. The actual content of the code assembly register corresponding to the current position is loaded into a shift register of Galois generator.

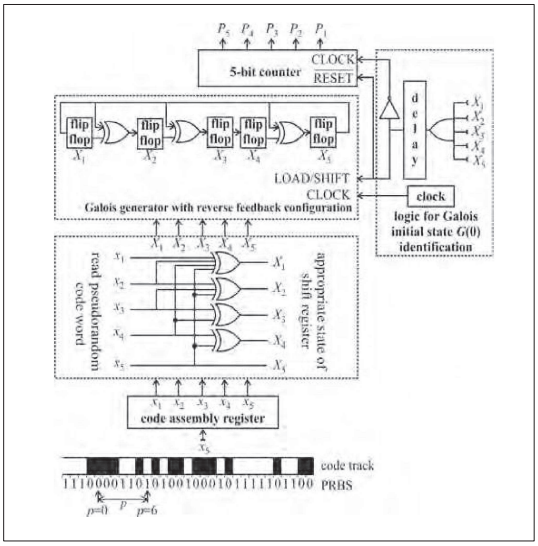

Figure 1. Serial pseudorandom/natural code converter based on Galois generator

This solution is faster because it has reduced number of logic gates (XOR gates) in the feedback loop. The position of XOR gates between flip flops in the Galois shift register is determined by the used feedback set [5, 4, 2, 1] for reverse generation law. Adequate feedback sets for different resolutions of PRBS sequence can be found in the literature [10]. Now, one XOR gate exists in the feedback configuration compared to three serials connected XOR gates in the case of Fibonacci generator. But, the additional logic that read code word converts to the appropriate content of the shift register is needed for proper functioning of this converter. This logic only participates on the beginning of each code conversion cycle and does not further participate in the code conversion process. The 5-bit counter and additional logic for Galois initial state $G(0)$ identification are also needed in this serial pseudorandom/natural code converter.

Logic for initial adjustment of read pseudorandom code word is needed because the read n-bit pseudorandom code word in real time is not identical to the n-bit current content of the shift register, which was the case at Fibonacci generator. Procedure of this adjustment logic design is detailed explained in the reference [9] for $n = 5$ bit resolution and chosen feedback set. Following this procedure for logic design in the case of resolution $n = 5$ and feedback set for Galois generator [5, 4, 2, 1], the following set of equations is obtained:

$$X_5 = x_5 \quad (1)$$
$$X_4 = x_3 \oplus x_4 \quad (2)$$
$$X_3 = x_3 \oplus x_4 \quad (3)$$
$$X_2 = x_2 \oplus x_3 \oplus x_5 \quad (4)$$
$$X_1 = x_1 \oplus x_2 \oplus x_4 \oplus x_5 \quad (5)$$

Based on the previous logic equations (1) to (5) the following matrix can be written:

feedback set [5, 4, 2, 1]

	x_5	x_4	x_3	x_2	x_1
X_5	**1**	**0**	**0**	**0**	**0**
X_4	**1**	**1**	**0**	**0**	**0**
X_3	**0**	**1**	**1**	**0**	**0**
X_2	**1**	**0**	**1**	**1**	**0**
X_1	**1**	**1**	**0**	**1**	**1**

where columns are in order $\{x_5, x_4, x_3, x_2, x_1\}$, and rows in order $\{X_5, X_4, X_3, X_2, X_1\}$. It can be seen that for coefficients [5, 4, 2, 1] in rows, the adequate diagonals are filled with "1" beginning from defined coefficients. It is highlighted diagonal filled with "1" for coefficient 5 in the feedback set. So, for any feedback set of coefficients this matrix can be easily written, and then from there adequate equations for adjustment logic can be easy obtained. This procedure is checked for different resolutions from 3 to 16 bit, and various possible feedback sets [10]. Based on previous considerations it is confirmed that exist the clear rule for matrix filling for any chosen resolution and the feedback set. The numbers in feedback set defines adequate starting position in rows, X_1, ..., X_n, from which starts with filling of diagonals with "1". Filled matrix directly gives set of logic equations which is needed for design of adequate logic for initial adjustment of read pseudorandom code word. This logic directly converts current content of Fibonacci shift register to adequate content of Galois shift register in adequate PRBS generator.

3. Simulations and Analysis of Serial Converters

For purpose of detailed functional and timing analysis of presented solution of serial pseudorandom/natural code converter, simulation of hardware realization is performed in software environment NI Multisim 11.0, Figure 2. In Multisim software can be fund several useful tools and parts for digital circuits. They range from digital gates (both families, TTL and CMOS are supported), indicators (such as LED's, LCD displays, Logic Analyzers and Word Generators), wires, data buses and power supplies. The most widely used subfamily for digital circuit simulation is the 74LSXX series where the digits after the LS indicate the gate's function. This subfamily is used for simulation of presented serial pseudorandom/natural code converters, and

applied logic gates in the simulations with their propagation delays are listed in Table 1.

The conversion process is executed for concrete example of read pseudorandom code word and for defined initial code word. 7-bit binary counter is used for counting of needed steps to shift register for finishing of one code conversion cycle. The read 5-bit code word $x = x_1, x_2, x_3, x_4, x_5$ is loaded in the 5 D flip flops on the beginning of the code conversion process. It is used example where read pseudorandom code word is $x = \{0, 1, 1, 0, 1\}$, and initial code word is $X(0) = \{1, 1, 1, 0, 0\}$. It is needed six clock periods to the shift register when its state becomes equal to the initial state, which is output of the binary counter in the case of Fibonacci generator. In the case of Galois based code converter, Figure 2, the read pseudorandom code word $x = \{0, 1, 1, 0, 1\}$ is firstly converted to appropriate state of the Galois shift register $X_{appr.state} = \{0, 1, 1, 1, 1\}$ based on equations (1)-(5), and then is also needed six clock periods to the shift register when its state becomes equal to Galois initial state $G(0) = \{0, 0, 1, 0, 0\}$. Galois initial state is obtained from initial state $X(0)=\{1, 1, 1, 0, 0\}$ using equations (1)-(5). It is proved proper functioning of proposed hardware realization for different examples of read and initial pseudorandom code words. States of the shift register in Fibonacci and Galois generator in each clock period during shifting from read pseudorandom code word to the initial code word of presented example are listed in Table 2. In the simulations of serial converter, Figure 2, is also used AND gate with 4 inputs together with one AND gate for initial state identification.

Logic gates delays	**Used logic gates from 74LSXX series**	**Propagation**
AND	74LS08D	15 ns
OR	74LS32D	22 ns
INV	74LS04D	15 ns
XOR	74LS86D	30 ns
D flip flop	74LS74D	$t_s = 20$ ns $t_h = 5$ ns $t_{CLK\text{-}Q} = 25$ ns
AND with 4 inputs	74LS21D	15 ns

Table 1. Logic gates and adequate propagations delays

To find the maximal working frequency of the whole digital circuit, which is the most significant difference between two then is also needed six presented solutions, it is needed to analyze propagation delays of each path. Combinational propagation delays are additive. t is possible to determine the propagation delay of a larger combinational circuit by adding the propagation delays of the circuit components along the longest path. Frequency must be determined by locating the longest path among all the flip-flop paths in the circuit.

The conversion process is executed for concrete example of read pseudorandom code word and for defined initial code word. 7-bit binary counter is used for counting of needed steps to shift register for finishing of one code conversion cycle. The read 5-bit code word $x = x_1x_2x_3x_4x_5$ is loaded in the 5 D flip flops on the beginning of the code conversion process. It is used example where read pseudorandom code word is x={0,1,1,0,1}, and initial code word is $X(0) = \{1, 1, 1, 0, 0\}$. It is needed six clock periods to the shift register when its state becomes equal to the initial state, which is output of the binary counter in the case of Fibonacci generator. In the case of Galois based code converter, Figure 2, the read pseudorandom code word $x = \{0, 1, 1, 0, 1\}$ is firstly converted to appropriate state of the Galois shift register $X_{appr.state} = \{0, 1, 1, 1, 1\}$ based on equations (1)-(5), and then is also needed six clock periods to the shift register when its state becomes equal to Galois initial state $G(0) = \{0, 0, 1, 0, 0\}$. Galois initial state is obtained from initial state $X(0) = \{1, 1, 1, 0, 0\}$ using equations (1)-(5). It is proved proper functioning of proposed hardware realization for different examples of read and initial pseudorandom code words. States of the shift register in Fibonacci

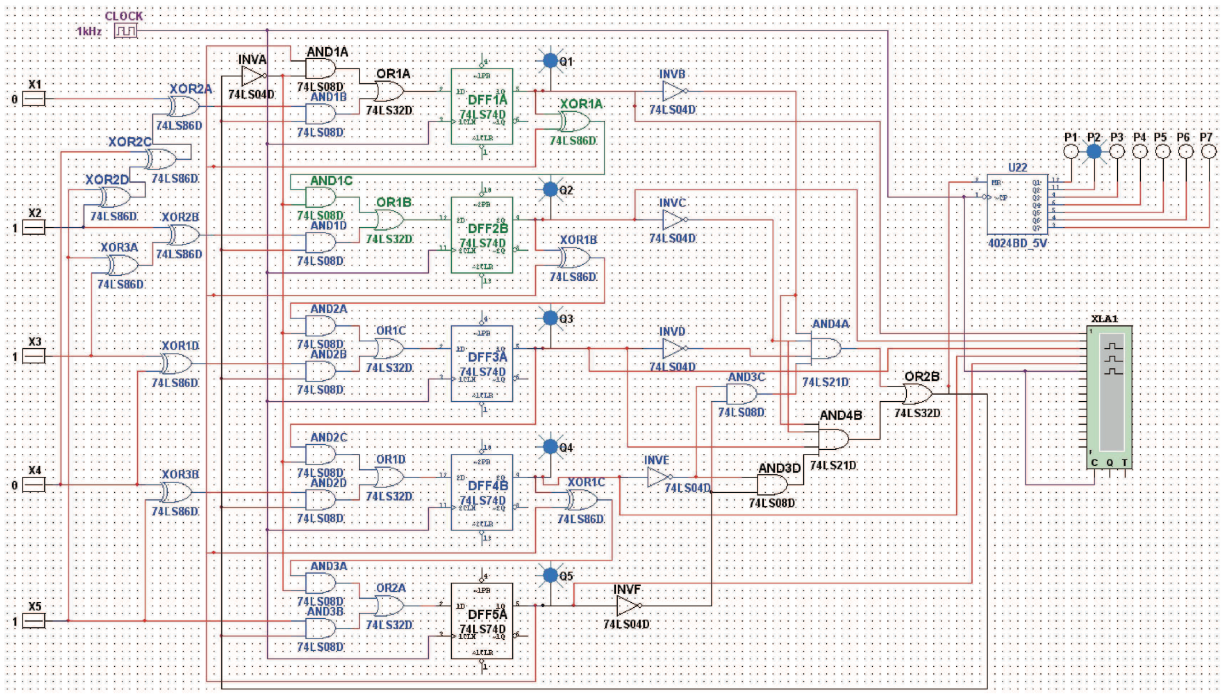

Figure 2. Simulation of serial pseudorandom/natural code converter based on Galois generator in Multisim

and Galois generator in each clock period during shifting from read pseudorandom code word to the initial code word of presented example are listed in Table 2. In the simulations of serial converter, Figure 2, is also used AND gate with 4 inputs together with one AND gate for initial state identification.

Output of the binary counter	**Fibonacci generator shift register content: {X1X2X3X4X5}**	**Galois generator shift register content: {X1X2X3X4X5}**
Read pseudorandom code word, P=0	01101	01111
P=1	00110	11010
P=2	00011	01101
P=3	00001	11011
P=4	10000	10000
P=5	110000	1000
Initial code word, P = 6	11100	00100

Table 2. Shift register content in Fibonacci and Galois generator

To find the maximal working frequency of the whole digital circuit, which is the most significant difference between two presented solutions, it is needed to analyze propagation delays of each path. Combinational propagation delays are additive. It is possible to determine the propagation delay of a larger combinational circuit by adding the propagation delays of the circuit components along the longest path. Frequency must be determined by locating the longest path among all the flip-flop paths in the circuit.

Propagation delays along longest paths in both simulations are (Figure 2.):

- for Fibonacci based converter (from the output of the fifth flip flop DFF5A, through three XOR gates in the feedback loop, then AND1A, OR1A, to the input of the first flip flop DFF1A):

$T_1 = t_{CLK\text{-}Q}(5) + 3\,t_{pd}(\text{XOR}) + t_{pd}(\text{AND}) + t_{pd}(\text{OR}) + t_s(1) = 25\text{ ns} + 3\text{x}30\text{ns} + 15\text{ ns} + 22\text{ ns} + 20\text{ ns} = 172\text{ ns},$

- for Fibonacci based converter (from the output of the fifth DFF5A, through logic for initial state identification INVF, AND3D, AND4B, OR2B, INVA, then AND1A, OR1A to the input of the first flip flop DFF1A):

$T_2 = t_{CLK\text{-}Q}(5) + t_{pd}(\text{INV}) + 2\,t_{pd}(\text{AND}) + t_{pd}(\text{OR}) + t_{pd}(\text{INV}) + t_{pd}(\text{AND}) + t_{pd}(\text{OR}) + t_s(1) = 25\text{ns} + 15\text{ns} + 2\text{x}15\text{ns} + 22\text{ ns} + 15\text{ ns} + 15\text{ ns} + 22\text{ ns} + 20\text{ ns} = 164\text{ ns}.$

So, maximal working frequency of Fibonacci based converter would be,

$$f_{max\text{-}fibonacci} = 1/T_1 = 5.81\text{MHz}. \quad (6)$$

- for Galois based converter (from the output of the first flip flop DFF1A, through one XOR gates in the feedback loop XOR1A to the input of the second flip flop DFF2B):

$T_3 = t_{CLK\text{-}Q}(5) + t_{pd}(\text{XOR}) + t_{pd}(\text{AND}) + t_{pd}(\text{OR}) + t_s(1) = 25\text{ ns} + 30\text{ns} + 15\text{ ns} + 22\text{ ns} + 20\text{ ns} = 112\text{ ns}$

- T_2 is the same as in the case of the Fibonacci based converter, so maximal working frequency of Galois based converter would be,

$$f_{max\text{-}galois} = 1/T_2 = 6.09\text{MHz} \quad (7)$$

From previous timing analysis, it can be concluded that Galois generator is faster than Fibonacci. The pseudorandom/natural code converter based on Galois generator has additional propagation delay due to the logic for initial conversion of read pseudorandom code word which is equal to propagation delay of three serial connected XOR gates (XOR2A, XOR2C, and XOR2D). Also, the contamination delays along each path are greater than or equal to the destination flip flop hold time, so the circuit will operate as designed.

For presented example of 5-bit pseudorandom binary sequence, 31 unique pseudorandom code words define absolute position. So, for the worst case, if the code converter needs 30 clock periods to convert the read pseudorandom code word in relation to the initial pseudorandom code word which present zero position, maximal duration of one conversion cycle for Fibonacci and Galois generator would be,

$$T_{conv\text{-}fibonacci} = 30\text{x}1/f_{max\text{-}fibonacci} = 5.16\mu\text{s} \quad (8)$$

$$T_{conv\text{-}galois} = 3\text{x}\,t_{pd}(\text{XOR}) + 30\text{x}1/f_{max\text{-}galois} = 5.01\mu\text{s} \quad (9)$$

So, the serial pseudorandom/natural code converter based on Galois generator is relatively faster than Fibonacci based for,

$$\delta_{\%} = \frac{T_{conv-\,fibonacci} - T_{conv-galois}}{T_{conv-\,fibonacci}}100\% = \frac{5.16 - 5.01}{5.16}100\% = 2.9\% \quad (10)$$

Duration of code conversion time depends on used resolution of PRBS and used feedback set (for same resolution different feedback sets exist, with different number of coefficients).

4. Conclusion

Improved, Galois based implementation of serial pseudorandom /natural code converters is presented and analyzed. It is proved reducing of conversion time at Galois based serial converter, which is its main advantage, by using detailed timing analysis. It is examined full functionality of both, Fibonacci and Galois based serial converters through simulation in NI Multisim software. The solution for easier designing of necessary logic for initial adjustment of read pseudorandom code word at Galois based converter is also given.

Acknowledgement

Research activities presented in this paper, are supported by funds of the Ministry of Science and Technological Development of Republic of Serbia, having the reference project number TR32045.

References

[1] MacWilliams, F. J., Slone, N. J. A. (1976). Pseudo-random sequences and arrays, *Proceeding of IEEE*, 64 (12) 1715-1728.

[2] Petriu, E. M., Basran, J. S. (1989). On the position measurement of automated guided vehicles using pseudorandom encoding, in IEEE Trans. IM, 38 (3) 799-803.

[3] Rau, J. C., Wu, P. H., Ho, Y. F. (2008). A novel reseeding mechanism for improving pseudo-random testing of VLSI circuits, in Tamkang *Journal of Science and Engineering,* 11 (2) 175-184.

[4] Amrani, M.E.H., Dowdeswell, R. M., Payne, P.A., Persaud, K. C. (1998). Pseudo-random binary sequence interrogation techniques for gas sensors, in Sensor. Actuat. B-Chem. 47, 118-124.

[5] Arsic, M., Denic, D. (1993). New pseudorandom code reading method applied to position encoders, *in Electron. Lett*. 29, 893-894.

[6] Denic, D., Miljkovic, G. (2009). Code reading synchronization method for pseudorandom position encoders, in Sensor. Actuat. *A Phys*. 150, 188-191.

[7] Petriu, E. M., Basran, J. S., Groen, F. C. A. (1993). Automated guided vehicle position recovery, *In: IEEE Trans. Instrum. Meas*. 39, 254-258.

[8] Denic, D., Arsic, M. (1993). Checking of pseudorandom code reading correctness, *in Electron. Lett*. 29, 1843-1844.

[9] Denic, D., Stojkovic, I. (2010). Pseudorandom/natural code converter with parallel feedback logic configuration, In: *Electron. Lett*. 46, 921-922.

[10] New Wave Instruments, Linear feedback shift registers: Implementation, M-sequence properties, feedback tables. (2004). http://www.newwaveinstruments.com/resources/articles/m_sequence_linear_feedback_shift_register_lfsr.htm