Improving and Testing the Pseudorandom Position Encoders with LabVIEW

Ivana S. Stojković[1], Goran S. Miljković[2] and Dragan B. Denić[3]

Abstract – **This paper discusses the pseudorandom position encoders and its basic integral parts, but special attention is given to the new approach in their development process. It is proposed a new and more flexible method for improving and testing pseudorandom encoders, which is based on virtual instrumentation. In particular, problems of generating pseudorandom sequence and also pseudorandom to natural code conversion are discussed, realized and tested using software package LabVIEW.**

Keywords - **Position measurement, Pseudorandom code, Position encoders, Pseudorandom binary sequence, Code conversion, Virtual instrument, LabVIEW.**

I. Introduction

Nowadays, the computers are very powerful and important tool in scientific research, especially in the cases where complete realization of the particular problem can be very expensive and complex. With the increasing performance of computer, the virtual instrument technology has greatly advanced over the years. The used virtual instruments (VI) in this paper are designed using the graphical programming language LabVIEW. Because a virtual instrument is based on a generic computer, its capability is constrained by the hardware limitations of the host computer in terms of CPU speed, bus system structure, and I/O interface configuration. A virtual instrument, in principle, is a computer-based, software driven instrument for test, measurement, signal processing and process control purposes.

The virtual instruments are very flexible compared to traditional hardware-based instruments, which provides the fact that they are based on software, personal computers and modular hardware. Their functions are chosen by end users and can be extended or modified according to users desires, in contrast with traditional instruments where the functions are vendor defined by hardware structure of the instrument and can not be changed. Virtual instrumentation concept is widely applied and its area of using is in constant increasing.

Digital measurement transducers, named as encoders, serve as devices for detecting angular and linear positions of machines and other moving objects. Position encoders convert position information into a binary code. The pseudorandom position encoders are the latest development trend in position measurement methods applied at industrial movable systems.

[1] Ivana S. Stojković is with the Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia and Montenegro, E-mail: ivana.stojkovic@elfak.ni.ac.rs

[2] Goran S. Miljković is with the Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia and Montenegro, E -mail: goran.miljkovic@elfak.ni.ac.rs

[3] Dragan B. Denić is with the Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia and Montenegro, E-mail: dragan.denic@elfak.ni.ac.rs

During development of position encoders, such as pseudorandom encoder, the goal is to avoid expensive realizations of experimental systems. For example, the realization of the real code track with different reading methods [6], which can be applied, would be very impractical and expensive.

Modeling and simulation of encoder integral parts is performed on different ways in the past from using discrete components [1], microprocessors [7], and field programmable gate arrays (FPGA) [5] to using a virtual instrumentation concept, which is considered in this paper. Advantages and disadvantages of using virtual instrumentation during development and testing of pseudorandom encoders are pointed out through this paper. Some realized encoder parts with LabVIEW are given and detailed explained.

II. A Basic Principle of Pseudorandom Encoding for Position Measurement

A pseudorandom binary sequence of length 2^n-1 is considered, that is:

$$\{S(p)/\,p = 0,1,\ldots\ldots 2^n - 2\} \qquad (1)$$

Since it is written with one bit per sector, a code track is divided in 2^n sectors. Term $S(p)$ represents content of the n^{th} element of the shift register after p shifts to the left. The absolute position identification is based on the "window property" of pseudorandom binary sequence. According to this, any n-bit code word {S(p+n-k) / k = n,...,1} obtained by a window of width n {x(k) / k = n,...,1} scanning the PRBS, is unique and may fully identify window's absolute position p relative to the beginning of sequence, [1], [3] and [4], as is shown in the Fig. 1.

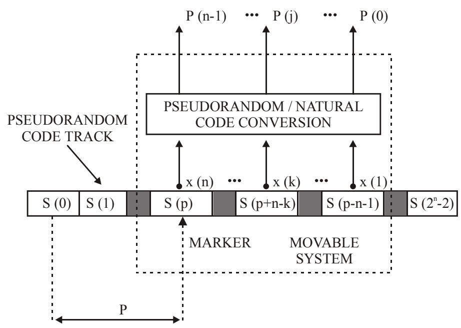

Fig. 1. Absolute position p is entirely identified by the pseudorandom code word $\{x(k)/k=n,\ldots,1\}$

Considering that pseudorandom sequences are periodical, the well-known property of pseudorandom sequences is used [5] in the sense that two n-tuples for two consecutive positions contain an identical array of $(n-1)$ bits. Also, it is of no importance that any n-bit word is adopted as the initial one.

$$\{S(n-k)/k=n,......,1\} \qquad (2)$$

Further, it can be considered that the pseudorandom binary sequence is generated using a shift register with a feedback, according to the following algorithm [5],

$$X(0)=X(n)\oplus c(n-1)X(n-1)\oplus...........\oplus c(1)X(1),$$
$$X(i)=X(i-1),\ \text{for}\ i=n,........1,$$

where feedback coefficients $c(n\text{-}1)$ are foreclose defined, as shown in Table I. After pseudorandom code reading is done, it is necessary to convert it into the natural code. In the case of high-resolution encoders, the method of parallel conversion using memory elements is not acceptable. Since the pseudorandom code is so specific it is possible to apply serial code conversion method [1]. The translation of the scanned pseudorandom *n*-tuple into a more natural representation is done sequentially, [2]. Conversion of these *n* bits into the natural code, following the serial code conversion method, is performed based on the property of the PRBS generating algorithm that it is reversible. The code conversion algorithm is shown in Fig. 2. The content of the register for code word generating, for the given position *p*, will be

$$\{S(p+n-k)/k=n,......,1\}, \qquad (3)$$

for each $p=0,1,..,2^n-2$.

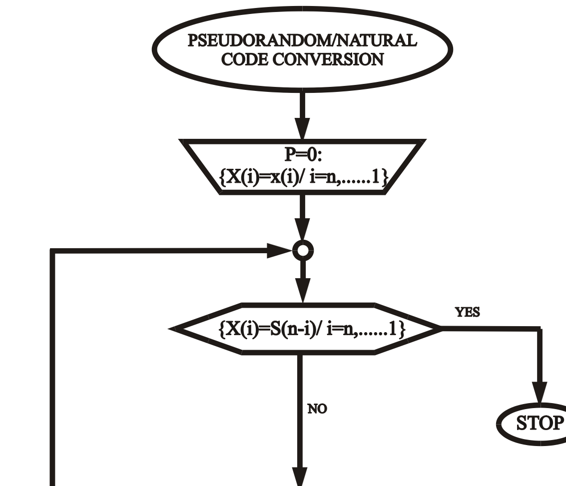

Fig. 2. The code conversion algorithm

It is based on the idea that it is possible to find the actual value of the position *p*, simply by counting the steps that the shift register with the inverse feedback needs until it reaches the initial state by successively shifting from the read pseudorandom *n*-bit word (2). Therefore, the code conversion algorithm (Fig. 2) starts by setting *n*-bit "X" variable at the current value of the *n*-tuple that is provided by the reading heads. After that, "X" is modified recurrently to the "reverse PRBS" generating low specified in Table I.

TABLE I

Shift register length n	Feedback in case of direct PRBS $X(0)=X(n)\oplus c(n-1)X(n-1)\oplus...\otimes c(1)X(1)$
3	$X(0)=X(3)\oplus X(1);$
4	$X(0)=X(4)\oplus X(1);$
5	$X(0)=X(5)\oplus X(2);$
6	$X(0)=X(6)\oplus X(1);$
7	$X(0)=X(7)\oplus X(3);$
8	$X(0)=X(8)\oplus X(4)\oplus X(3)\otimes X(2);$
9	$X(0)=X(9)\oplus X(6);$
10	$X(0)=X(10)\oplus X(3);$
11	$X(0)=X(11)\oplus X(2);$
12	$X(0)=X(12)\oplus X(6)\oplus X(4)\otimes X(1)$
13	$X(0)=X(13)\oplus X(10)\oplus X(6)\otimes X(4);$
14	$X(0)=X(14)\oplus X(13)\oplus X(8)\otimes X(4);$
Shift register length n	Feedback in case of reverse PRBS $X(n+1)=X(1)\oplus b(2)X(2)\oplus...\otimes b(n)X(n)$
3	$X(4)=X(1)\oplus X(2);$
4	$X(5)=X(1)\oplus X(2);$
5	$X(6)=X(1)\oplus X(3);$
6	$X(7)=X(1)\oplus X(2);$
7	$X(8)=X(1)\oplus X(4);$
8	$X(9)=X(1)\oplus X(3)\oplus X(4)\otimes X(5);$
9	$X(10)=X(1)\oplus X(3);$
10	$X(11)=X(1)\oplus X(4);$
11	$X(12)=X(1)\oplus X(3);$
12	$X(13)=X(1)\oplus X(2)\oplus X(5)\otimes X(7);$
13	$X(14)=X(1)\oplus X(5)\oplus X(7)\otimes X(11);$
14	$X(15)=X(1)\oplus X(5)\oplus X(9)\otimes X(14);$

The algorithm will cycle in this way until the "X" reaches the predefined initial state (2). When this state is finally reached, algorithm stops cycling, and the current value of the index "*p*" represents the wanted *n*-bit natural code corresponding to the initially read pseudorandom *n*-tuple, [1].

In this paper the basic idea is to develop and improve only some parts of pseudorandom position encoder using modern technologies. Therefore the pseudorandom sequence generating and code conversion pseudorandom / natural are tested by software package LabVIEW [8].

III. A DEVELOPMENT PROCESS AND TESTING THE PSEUDORANDOM POSITION ENCODER BASED ON PRINCIPLE OF VIRTUAL INSTRUMENT

Technological improvements are producing different solutions of pseudorandom encoders in order to achieve that position sensors become easier to connect to the industrial application. Specialist applications inspire advanced developments in existing technologies, while novel techniques extend the capabilities and applicability of position encoders.

Because pseudorandom encoder is one complex device, during its development often is much easier to test solution through analyzing of its integral parts. As can be seen from Fig.1 some of important parts of pseudorandom encoders are code track, reading heads, pseudorandom to natural code conversion, electronic block for signals processing and determining of position information. Different modifications of pseudorandom encoder solutions, which perform during development process, must be experimentally tested and verified before their practical realization. In the past, various experimental systems are used during development process of position encoders. Some of them are included automated path-guided vehicles [3], microprocessors [4], and computers [6]. Here is proposed method of improving and testing pseudorandom encoders based on virtual instrumentation. Text based programming language used in microprocessors is replaced with intuitive, easier graphical programming language. LabVIEW promotes the concept of hierarchical and modular programming. First, an application is divided into a series of simple subtasks. Next, the VI is built to accomplish each subtask and then those VIs are combined on a top-level block diagram to complete the larger task. Modular programming is a plus because you can execute each subVI by itself, which facilitates debugging. The goal is to point out advantages and limitations of this approach.

The important part of pseurodanom encoder is code track or encoder disk and corresponding reading heads, which gives signals according to moving of shaft. Multiple realization of modified code tracks or reading systems during development process would be very expensive. So, signals on reading heads outputs are usually simulated. Some different methods of pseudorandom sequence generation are applied in [5]. For example, the approach that uses discrete components has the advantages of being simple and inexpensive, but is not easy for changes. The advantage of using a microcontroller is that it is easy to make changes to the program. FPGA-based implementation is flexible, very fast, but the equipment that uses is rather expensive, and VHDL programming is not something everyone is familiar with. In this paper is presented solution for pseudorandom sequence generation, which will help for simulation of signals on reading heads outputs, for different moving routes of shaft. This solution is realized in software package LabVIEW, and provides generation of pseudorandom sequences of different length, which is determined by the parameter on the front panel. The block diagram of the realized pseudorandom sequence generator is shown in Fig. 3, where are used data from Table I.

Second important part of pseudorandom encoder is block for pseudorandom/natural code conversion. The most important features of this block are the time of code conversion and reliability. One type of code conversion block is realized as virtual instrument, and is shown in Fig. 4. To perform code conversion this block must have information about length of pseudorandom sequence, and time of code conversion depends from this quantity. For example, for easier understanding, if n is 4 and the initial state is binary word presented as 1011, output information is position p = 5, which is representation of code conversion.

Further, the electronic block of pseudorandom position encoder perform processing of acquired signals and determines the position information.

Using of proposed approach gives the obvious advantages in relation to previous methods, such as easier learning of graphical programming language LabVIEW, using of computer resources, easier modification of solution. LabVIEW also gives possibility of complex problem dividing to simplier tasks, and the various tools for results presentation.

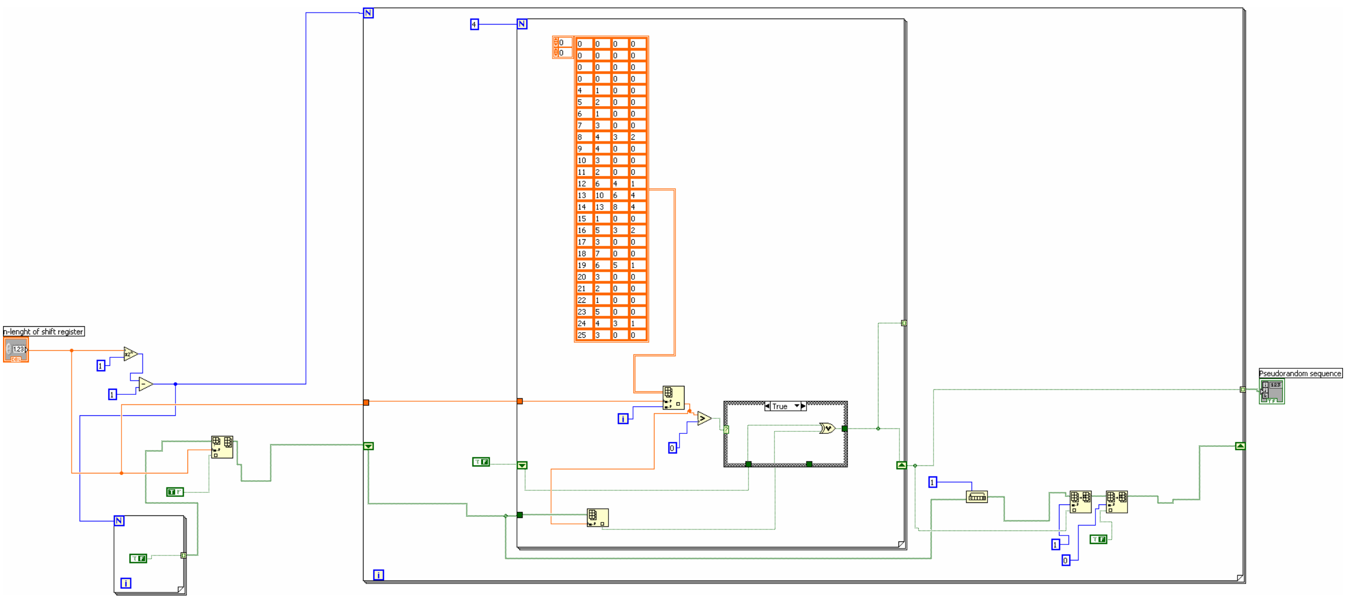

Fig. 3. Block diagram of generating pseudorandom seguence applied at pseudorandom position encoders

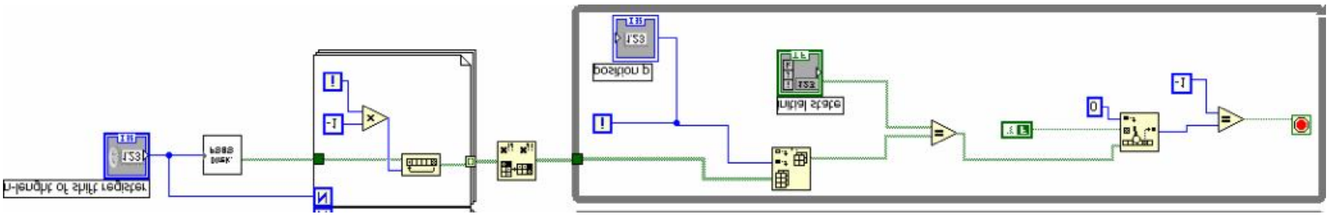

Fig. 4. Block diagram of code conversion pseudorandom / natural applied at pseudorandom position encoders

IV. Conclusion

The proposed solution for pseudorandom position encoders improving and testing has significant advantages in relation to previous methods in sense of reducing complexity during solution modification, cost and needed time. This solution maximally exploits the advantages of using computer and virtual instrumentation concept in developing of optimal pseudorandom encoder solution. The advantages of this approach are investigated and verified through realization of the two integral parts of pseudorandom encoder.

References

[1] E.M. Petriu, "Absolute-type position transducers using a pseudorandom encoding", IEEE Trans. Instrum. and Meas., vol. IM-36, no. 4, pp.950-955, 1987.

[2] E.M. Petriu, J.S. Basran, "On the position measurement of automated guided vehicles using pseudorandom encoding", IEEE Trans. Instrum. and Meas., vol. 38, no. 3, pp.799-803, 1989.

[3] E.M.Petriu, J.S. Basran, F.C.A. Groen, "Automated guided vehicle position recovery", IEEE Trans. Instrum. and Meas., vol. 39, no. 1, pp.254-258, 1990.

[4] D. Denić, I. Ranđelović, M. Rančić, "High-resolution pseudorandom encoder with parallel code reading", ISSN 1392-1215 Electronics and electrical engineering Nr 7(56), Kaunas, Litvania, pp.9-14, 2004.

[5] S. Engelberg, H. Benjamin, "Pseudorandom sequences and the measurement of the frequency response", IEEE Instrum. and Meas. Mag 8 (1), pp.54-59, 2005.

[6] D. Denić, I. Ranđelović, G. Miljković, "Recent trends of linear and angular pseudorandomencoder development", SPEEDAM, Conference Proceedings, pp.746–750, Taormina (Sicily), Italy, 2006.

[7] D.B. Denić, G.S. Miljković, "Code reading synchronization method for pseudorandom position encoders", Sensors and Actuators A 150, pp.188-191, 2009.

[8] LabVIEW, *User's Manual*, National Corporation, USA, 2004.