New Type of Linear and Angular Displacement Transducer Based on Pseudorandom Encoding

Dragan B. Denić[1], Ivana S. Ranđelović[2], Jelena R. Đorđević[3] and Goran S. Miljković[4]

Abstract - **Pseudorandom encoder as a new type of absolute encoder with one code track coded by applying pseudorandom binary sequences, is considered in this paper. A method for parallel reading of pseudorandom code using photodetector array is proposed. Simultaneously a problem of zero position adjustment at encoder installation is considered and a concrete solution in accordance with requests of high technologies encoders is proposed.**

Keywords - **Position measurement, Pseudorandom encoder, Pseudorandom binary sequence, Zero position adjustment.**

I. Introduction

Digital linear and angular displacement sensors, well - known as a encoders, are electromechanical components, whose basic function is providing measurement information about sensor head position in relation to measuring scale. One good alternative to the classical absolute encoder is absolute encoder which uses a technique of longitudinal coding. By using of one code track an absolute position is coded by using of pseudorandom binary sequence so that each groupe of *n* successive bits represents the unique code word. Two successive code words overlap and differ in only one bit [2, 3]. Instead of *n* bits per quantization step now we have only one bit and in that way it is opened a possibility of achieving greater accuracy of position measurement besides eliminating problem of great number of code tracks.

There are some special applications where it is impossible to demand such initial movement of the movable system (MS), such as systems for level measurement using digital position converters. It can achieve by applying detectors with integrated photodetector array [1], or using CCD cameras [4]. Since there is a great need for such a solution, an algorithm for reducing its performance time, using linear integrated photodetector array for parallel pseudorandom code reading, will be considered in this paper. Such way of coding opens a possibility of serial pseudorandom code reading by using only one detector [10, 11]. For new position defining it is enough to read only one next bit on code track. Such encoders exist on market under term the virtual encoder [12].

Their basic disadvantage is in fact that they request initial movement at system starting in order to collect first n bits from code track which define the first position.

II. Parallel code reading of pseudorandom code and its usage in position encoders

One of the important features of pseudorandom binary sequence (PRBS) is based on the “window property” which is represented as $\{S(p)/p=0, 1, ..., 2^n-2\}$. Window of width n gives n-bit code word $\{S(p+n-k)/k=n,...,1\}$ which is unique and may fully identify window’s absolute position p relative to the beginning of sequence, Fig. 1. Code words are now arraying linearly or longitudinally (but not transversely as in the case of classic coding), and they overlap. The first $(n-1)$ bits of such code word are identical to the last $(n-1)$ ones of the previous code word.

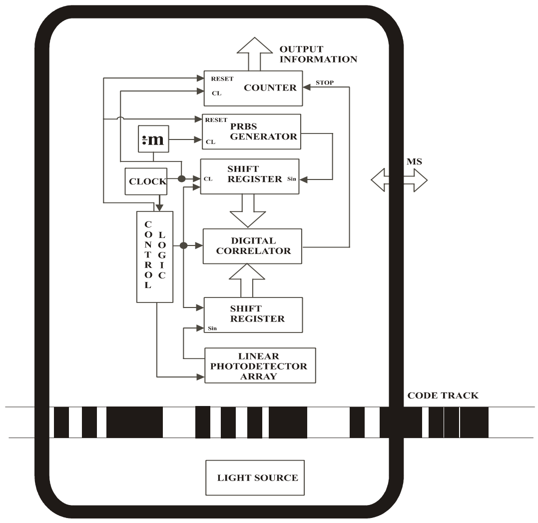

Fig. 1. High-resolution pseudorandom encoder with applied method of parallel code reading

For realizing of true pseudorandom absolute encoder, it is needed to apply any kind of parallel code reading method and in order to avoid a need for initial movement. A usage of *n* particular detectors is unacceptable to encoders with high measurement resolution. It is problem to dispose those *n* detectors on such little space [12]. One of the possibility is applying of integrated photodetector arrays for pseudorandom code reading. Integrated photodetector arrays are available on the market with different intervals between photodetectors. Those intervals are 13μm, 10μm, 7μm and smaller. It should use a large number of photodetectors in order to increase absolute position measurement precision. On Fig. 1 is shown one solution which represents the basic principles of parallel

[1] Dragan B. Denić is with the Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia and Montenegro, e-mail: ddenic@elfak.ni.ac.yu

[2] Ivana S. Ranđelović is with the Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia and Montenegro, E-mail: rivana@elfak.ni.ac.yu

[3] Jelena R. Đorđević is with the Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedeva 14, 18000 Niš, Serbia and Montenegro, E-mail: jelenadj@elfak.ni.ac.yu

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

reading solutions until now as based on using of linear photodetector array so as those represented in references [1,5].

Reading of the code track is with a light source at one side and an integrated circuit which consists linear photodetector array, at the other side. A state read after *k* tact pulses puts into the shift register. Let us read (*n+c*) bits and let nominal value of photodetectors number per one code bit be *m* and *n* is a number of bits needed for detecting the absolute position. Usually *c* is one additional bit and it increases a system redundancy, which provides detection of code reading errors. A condition $k \geq (n+c)\ m$ is always accomplished. Read code word is in the following form {00000111...1100 00111...1100....}. A transition is detected at the border of two elements. In the ideal case, a number of consecutive "1" or "0" per one bit equals *m*. It may occur a deviation due to not ideal drift of code elements on code track. Conversion into natural code is done after the reading of total output code. A generator of used PRBS starts from the code word that corresponds to initial "0" position on the code track. Generator core is a shift register with appropriate feedback. With each tact that conduces generating of next PRBS byte, *m* tacts from the shift register are performed. A code identical to the one that would be read in the case of continuous MS movement from the position "0", is obtained that way. In an ideal case, after a certain number of register shifts, a code word identical to the read one would be obtained. That correspondence could be simply detected by digital comparator circuit. A number of steps counted by a counter until the moment of correspondence represents output position information in the natural code. Unfortunately, as said before, an error would often occur in practice, and it is enough that one bit is read from (*m+1*) detectors and the PRBS generator will not generate such code word, thus, a correspondence will not be detected. This is why a digital correlator is used, although it is a much complex solution, but it solves the problem, Fig. 1. Accuracy of the detected code is increased introducing a greater value for *c*. Accuracy of correlator output does not depend much on the accuracy of defining boundary locations of code elements. Let us observe the following example. Let the distance between code element locations be 0.5mm and let the 11-bit PRBS (2047 quantization steps) be applied. In this case, a measurement range of 1m is realized. Now, let the distance between code elements be 10μm, which means that for the range of 1m, 100000 positions are required. This could be achieved using a 17-bit code. At least (17+1)10μm=180μm of code track should be read by photodetector array. An optical system for enlargement of the figure can be applied here. If the figure is enlarged 10 times, a photodetector circuit of at least 1800μm is required. If distance of 10μm between detectors is satisfied with an array of 256 photodetectors, total space resolution is now 1μm. In the worst case, the shift register connected to the PRBS generator, should make 10x100000 shifts until the moment it detects the correspondence. It is altogether 0,1s at the tact of 10MHz, i.e. 100ms. That makes the boundary work frequency of the system 10Hz, which is inadmissibly small even for general purpose encoders. In the reference [5] it is pointed to a possibility that a classic pseudorandom/natural code conversion is done using a simple digital comparator. Additional fine position could be defined based on the detected, defined transitions. Further in the paper, this idea will be elaborated and with some additional modifications, we will attempt to realize a solution where it is not necessary to define rough position for each new code reading.

III. NEW ALGORITHM WITH REDUCED PROCESSING TIME

The same function can be done using microprocessors and the appropriate software. A realization using one modern programmable logical circuit is possible. An example of a possible algorithm realization of a new solution proposed here is shown in Fig. 2. The basic idea is to exclude the digital correlator because it leads to a more complex system. Our goal is to achieve shorter time for PRBS reconstruction read from the track, and then use the well-known algorithm of pseudorandom/natural code conversion. The other idea does not consider constant determination of the rough position, but only for certain values of the fine ones. The fine position is defined by a number of rightmost photodetectors. A momentary position of the movable system is directly obtained without using any arithmetical operations, with the fine position resolution defined by the distance of two consecutive photodetectors in the photodetector array circuit.

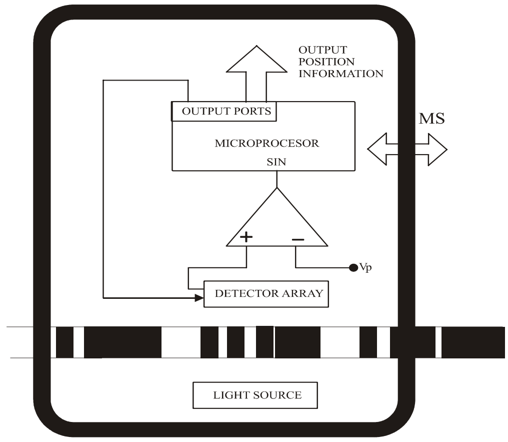

Fig. 2. An example of an encoder realization using a microprocessor

As it is shown in Fig. 2., a microprocessor defines the operation of photodetector array circuit using control signals. A reading of the recorded PRBS sequence is done in a way that series of voltage levels, which indicate a light intensity on the corresponding elements of the photodetector array, is lead to the input of a comparator. On that way, with a defined voltage level V_p, this series of voltage levels is converted to a series of logical "0" and "1" at the comparator output, that is then lead to a serial microprocessor port. Afterwards, the microprocessor performs the functions according to the software whose algorithm is shown in Fig. 3.

Bits obtained on this way are memorized and then analyzed. Memorized binary code word is in the form of {...000111...1111000...}, considering that *m* consecutive detectors are used for reading of one bit. However, as it is previously mentioned, it is possible that some deviations occur - {...0001011...1101000...}. Detector ordinals are then determined. Since nominal period (a number of detectors per one bit) is known, a difference of two consecutive detector ordinals is approximately *dm, d={1,2,..., n}*. All deviations greater then this one can be updated, but the periods between the transitions will be different afterwards.

In any case, a calculation of period mean value is proposed here. In the ideal case, obtained mean period (meaning the integer part of the mean value) will be equal to the nominal period. In reality, there will be deviations, and if they are greater than the proposed value *k*, the user should be informed. The system can track down those deviations and perform diagnostics in the part of additional functions. Based on the estimated mean period, certain corrections are eventually performed and the detector ordinals are determined. Fine position is now simply evaluated as a number of detectors that are outside of the last determined transition. The MS movement direction is determined based of the previous value of the fine position. When the boundary between bits is known, it is easy to perform the reconstruction of the read part of PRBS and then the pseudorandom/natural code conversion. The value of the rough position is that way determined. Since the rough position has the same values for a great number of simultaneous code readings, there is no need for permanent performing of this, according to the necessary time, the most important part of the algorithm. Initial idea about the moment of rough position determination, which is completely logical, imposes as the natural first one, and considers rough position determination whenever the fine position equals 0. However, detail elaboration of this idea indicates various additional problems. Firstly, during the MS movement to the left, the next fine position value is *m-1*, and it is necessary to decrease the rough position value by one. This correction of position value by one is a well known problem which occurs in the case of all parallel code reading methods, and it is solved by implementing correction logic as shown in [8]. Additional problem is that the correction is not to be done always, but only in specific moments. All this could be solved, but the complication of the algorithm would be significant, and it would even be necessary to record the preceding fine position value. Secondly, it is possible, for some reason, to skip value *fine position=0*, which way the rough position determination would also be skipped. Naturally, detection of this kind of situation would further complicate the proposed algorithm. The idea which avoids mentioned problems considers the rough position determination whenever the leap of fine position occurs. During the MS movement to the right, the fine position leaps from value "*fine position = m-1*" to "*fine position = 0*". When the leap is detected, the fine position will equal 0, and the rough position will be determined then, which is identical to the above-described approach. However, during the MS movement to the left, the fine position leaps from value "*fine position = 0*" to "*fine position = m-1*". The rough position is now determined for the value "*fine position = m-1*" and will be decreased by one relative to the case when it is determined for the value "*fine position = 0*". This means that the correction logic is no longer needed, which way the system is simplified and its reliability increased. If the leap "*fine position = 0*" or "*fine position = m-1*" occurs, this will not affect system performance, because the rough position determination is no longer linked to a specific value of the fine position. At the end, this kind of system is immune to errors that cause fine position leap. In these cases, after such error occurs, the system automatically determines the rough position, which is necessary to be done after occurrence of every error.

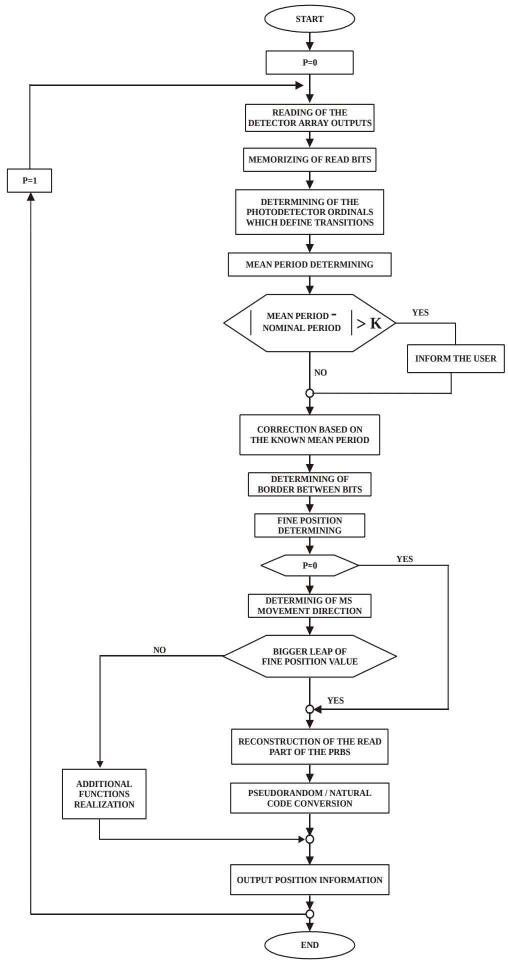

Fig. 3. Algorithm of the proposed solution

IV. DETERMINING OF MOMENTARY POSITION WITH DIRECT ADJUSTED ZERO POSITION

It comes to the unmatchings in the starting positions when installs an absolute encoder on movable system shaft. Therefore it is needed to adjust that zero position of position encoder corresponds to zero position of movable system, respectively directly accomplishing of zero position adjusting.

One of the leader factory in the world for production of new encoder generation, Stegmann, presented on the market a great choice of the incremental and absolute encoders with a different mechanical interface, resolution and with the new electronic features which are in the scope of the regular standards [9]. Today, it requests from encoders a flexibility, a reliability, a simple installation, and based on that it is created a new technology for the encoder production which enables directly adjusting of zero pozition. On Fig. 4. it is shown an example of the absolute encoder with such possibility.

If at installation of encoder CA6S on shaft of the movable system exists an umatching in the starting positions, directly adjusting of the starting position is done by pressing a taster. According to the function method of classical encoders, this solution is most probably based on storing of the correction parameters which almost figuring within function algorithm of such encoder.

Fig. 4. Stegmann absolute encoder, CA6S

Most probably and most natural is that momentary position of encoder for zero position of the movable system exactly represents that memory correction value. Anyway, this approach requests an additional operation in each subsequent process of the position measurement, and that operation is a correction of each read new position. That points out to the fact that with such approach time of position measurement increases. That disadvantage is certainly less important in relation to the big quality which is obtained by possibility of direct adjusting of zero position. A necessity of permanent memory existing doesn't represent the additional system complexing with respect that modern encoder should represent or in future will represent the intelligent sensor. In that case standard dictates that it comprises a memory element with permanent record in which are occured a factory number, a detecting and deciding to whom defined area quantity belongs, a processing of measured data, upkeep and identification of address, acceptance of command/data and generating data/command, autocalibration, zero and offset adjusting, and autotesting, besides a measurement of the physical quantity. Furthermore, from the modern encoders are requested or will be certainly requested an independence in deciding at control, local processing and statistic besides direct installation on the spot. Therefore, the existing of permanent memory element is or will be something necessarily and dictated with standard.

In order to direct adjusting of zero position, in paper is proposed solution that with pressing the taster, read code word is addopted for starting code word. It stores as such in flash memory of encoder itself as starting code word and in relation to it code conversion is perfomed, with respect to that can take anyone code word in relation to who, code conversion will perform. With direct adjusting of zero position by users with pressing the taster and by addopting of read code word for starting code word, doesn't request a mechanical movement or encoder rotation if exists an unmatching in starting positions during encoder installation on shaft of the movable system. This has a great importance in environments where encoder is used and where is very hard performing of encoder mechanical movement, with respect that a usage of encoder is wide. Therewith, the additional operation is eliminated, respectively a correction of each read value, because if on this way read code word wouldn't addopt for started, it would represent a correction factor which in each subsequent process of position measurement would add or substract from current read value. Addopting of read code word for starting is especially important and for time saving of position measurement, with respect that needed time for position measurement performing is one of the basic problems at pseudorandom encoders. By using of this solution it doesn't come to the encoder complexing besides it significally gets on time saving of position measurement. After storing of code word in flash memory, that memorizes as starting code word, it performs a code conversion pseudorandom/natural, according to the algorithm which is shown on Fig. 3.

V. CONCLUSION

There are usages where an initial movement of movable system after power is turned on, is unacceptable or even impossible. Then it is neccessary to perform parallel code reading and for case of high measurement resolutions the linear detector arrays or CCD sensors are used. In paper is suggested the solution which significally decreases a basic problem, and that is significant time needed for performing of position measurement. Simultaneously proposed solution provides in the simple way direct adjusting of zero position. Read code word for that arbitrary position is memorised as starting code word and in relation to it code conversion is performed, according to algorithm which is shown in the paper. On this way, additional operation is eliminated.

REFERENCES

[1] J.T.M Stevenson and J.R. Jordan, "Absolute position measurement using optical detection of code patterns", *J. Phys E. Sci. Instrum. 21*, 1140-1145, 1988.

[2] Jones B.E. and K. Zia, "Digital Displacement transducer Using Pseudorandom Binary Sequences and a Microprocessor", *IMEKO/IRAC, Symposium Proceedings*, pp. 368-379, London, 1980.

[3] Arsić, M., Denić, D.: "Position measurements based upon the application of pseudorandom encoding", *Facta Universitatis, Ser. Electronics and Energetics*, no. 6, pp. 13-23, 1993.

[4] Khalfallah, H., Petriu, E.M., Groen, F.C.A.: "Visual position recovery for an automated guided vehicle", *IEEE Trans. Instrum. and Meas.*, vol. 41, no. 6, pp. 906-910, 1992.

[5] Johnston J. S. "Position measuring apparatus", UK Patent Application no GB 2126 444A.

[6] Arsić, M., Denić, D.,: "Code convertor pseudorandom/natural applied at position encoders", *ETRAN, Ser. Elektronika*, pp. 164-167, 1995.

[7] Denić, D., Pešić, M., Arsić, M.: "New algorithm of code conversion pseudorandom/natural", Proceedings *ETAN*, II sveska, pp. 39-44, Belgrade, 1993.

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

[9] http: // www.stegmann.com

[10] Arsić, M., Denić, D.,: "New pseudorandom code reading method applied to position encoders", *Electronic letters* no. 10, vol. 29, pp. 893-894, 1993.

[11] Denić, D., Ranđelović I., M., Arsić, M.: "Absolute position measurement using the method of pseudorandom code parallel reading", *ICEST 2003 Conference Proceedings*, pp. 394-397, Sofia, Bulgaria, 2003.

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