Virtual instrumentation used for displacement and angular speed measurements

N. PATRASCOIU A. POANTA A. TOMUS and B. SOCHIRCA

***Abstract*—**This paper presents a virtual instrument built in LabVIEW used like a software solution to implement a displacement and angular speed measurement for a mobile that is moving in a circular direction. The same virtual instrument can be used also for measuring displacement and speed on the linear direction, by converting the linear movement into a circular movement or by using a linear incremental encoder. For this, first is determined the function through which is possible to detect the direction of movement and then the algorithm through which are makes the measurements and these are implemented through the graphical programming language used in LabVIEW..

***Keywords*—**algorithm, quadrature encoder, measurement, virtual instrument, data acquisition board.

I. INTRODUCTION

MOVEMENT is defined as a physical quantity of a mechanical change through is possible to provide information about position of a material point or mobile against a reference system. Quantities derived from this, which may be considered, are: position, distance or proximity.

The measure of these quantities is represented by displacement against a reference. In many applications the displacement is considered as a vector so it is necessary to calculate both size and direction for this quantity. Usual procedure for calculating the size of the displacement is to use an incremental sensor that generates a pulse train so by counting of these pulses is generates a number that represent the size of the displacement.

Considering the physical relationship between displacement and velocity, by determining the size of displacement using an incremental sensor, are obtained methods for determining the size of velocity. Most used such methods are based on measurement of elapsed time between successive pulses or counting of pulses number during the prescribed time. As with the displacement case, are frequently situations where the velocity should be considered a vector so it besides determining its value, is necessary to determine direction of movement [1].

Same incremental sensor can be used to detect the direction of movement if it provides two trains of pulses shifted by one quarter of the period, in which case it is named quadrature encoder. A quadrature encoder can have up to three channels – Ch A, Ch B, and Ch Z [2].

There are data acquisition boards (DAQ) that accept at their counters signals provided by encoder quadrature signals such as the NI 622x, NI 625x, and NI 628x (M series devices) but there are also data acquisition boards that not accept such signals like the NI PCI-6024 (E series devices). In this case it is necessary to achieve a logical system to detection the direction of motion and also to increment or decrement a counting value depending of the direction of movement. Such a system can be achieved through hardware structure but also can be done by a software solution using digital inputs and counters on the data acquisition board.

II. CONNECTING QUADRATURE ENCODERS TO DATA ACQUISITION BOARD

For the data acquisition boards that accept at their counters signals provided by encoder quadrature the values of the counters increases or decreases depending on the relative time of occurrence of the two trains of pulses. So, when channel A (Ch A) leads channel B (Ch B) in a quadrature cycle, the counter increments and when channel B leads channel A in the same quadrature cycle, the counter decrements [3].

Fig.1 shows a quadrature cycle and the resulting increments and decrements for the counter value.

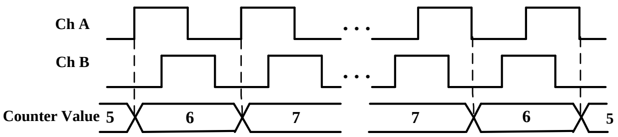

Fig.1. The input signals for data acquisition board

For the data acquisition boards which have no specific inputs for quadrature signals or reconfigurable digital inputs like the NI PCI-6024 (E series devices) it is necessary to achieve a logical system to detection the direction of motion and also to increment or decrement a counting value depending of the direction of movement. Such a system can be achieved through hardware structure but also can be done by a software solution using digital inputs and counters on the data acquisition board.

If the hardware solution will be chosen, may be used specialized circuits for signal conditioning. For example, the LS7084 quadrature clock converter from LSI Computer Systems, Inc. converts the A and B signals from an encoder into a clock and up/down signal that it can connect directly to the data acquisition board [4].

Fig. 2. illustrates the connections of the encoder to the data acquisition board using LS7084 quadrature clock converter were DIO represents the digital inputs connections of the data acquisition board used for increment or decrement of the counter depending on the logic level of the signal UP/DOWN.

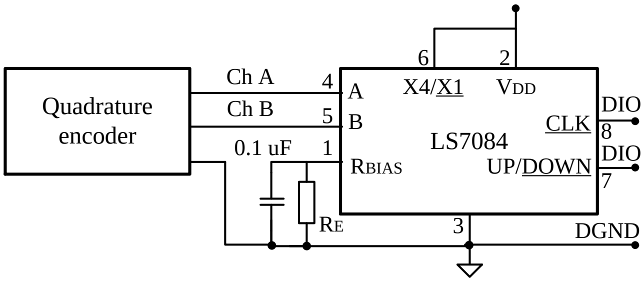

Fig. 2. Encoder connection to DAQ using LS7084 clock converter circuit

If the software solution will be chosen is first necessary to determine a logical function through which is possible to control the direction of counting according to the direction of movement.

III. Problem Formulation

As was mentioned above for make measurements for either linear or angular displacement is first necessary to determine its direction. After that, it can control the counting direction for the counter through which is determined the value of displacement by counting the train of pulses.

To measure angular displacement and angular speed we use PCI-6024E, a data acquisition board from National Instruments that has 8 digital I/O (DIO0 … DIO7) lines (TTL/CMOS) and two 24-bit counter/timers without having the dedicated inputs to connect a quadrature encoder [5]. Control the operation of the data acquisition boards is achieved through a program written in LabVIEW graphical programming language called virtual instrument.

This data acquisition board uses the National Instruments DAQ-STC system timing controller for time-related functions, fig.3. The DAQ-STC consists of three timing groups that control analog input, analog output, and general-purpose counter/timer functions. These groups include a total of seven 24-bit and three 16-bit counters and a maximum timing resolution of 50 ns. The DAQ-STC makes possible such applications as buffered pulse generation, equivalent time sampling, and seamless changing of the sampling rate [5].

To achieve the determination of displacement direction is needed in these conditions to be used two digital inputs to connect to the signals A and B carried over from the quadrature encoder. Displacement value is obtained by counting the pulses A or B, and its direction is determined by counting purposes, otherwise said, by increment or decrement the counter value. The angular speed value is obtained by counting pulses during the prescribed time [7], [8].

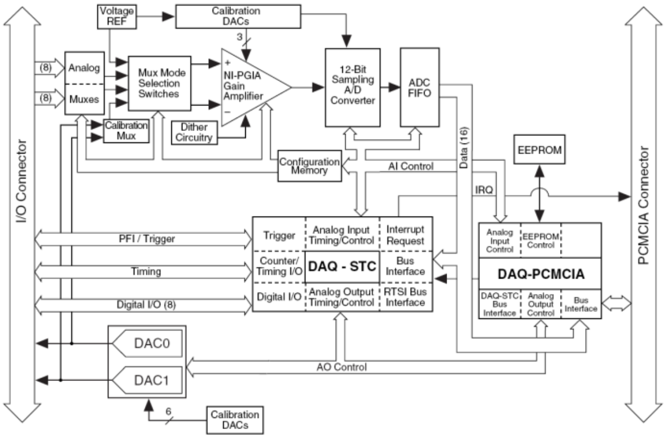

Fig. 3. DAQ PCI-6024E Block Diagram

To realize the virtual instrument for displacement and angular speed measurement based on two trains of pulses shifted by one quarter of period is necessary to synthesize a control command for establish counting direction and an algorithm to control the counters and to extract the content of the counters in the prescribed time.

IV. Problem Solution

The software solution of the problem consists in realizing the virtual instrument through which it can identify the meaning of motion and it can measure the angular speed and displacement values.

A. *Identification of movement direction*

For synthesizing the command control signal called *Counting Selection* is considered a chart signals (Fig.4.) that identify all the possibilities of combining the two pulse trains according to the direction of rotation [9].

Following the diagram signals depicted in Fig. 4, 8 distinct states denoted by S_i ($i = 0 ... 7$) can be identified, corresponding to 8 possible combinations of logic levels for both Ch A and Ch B signals and output signal *Counting*

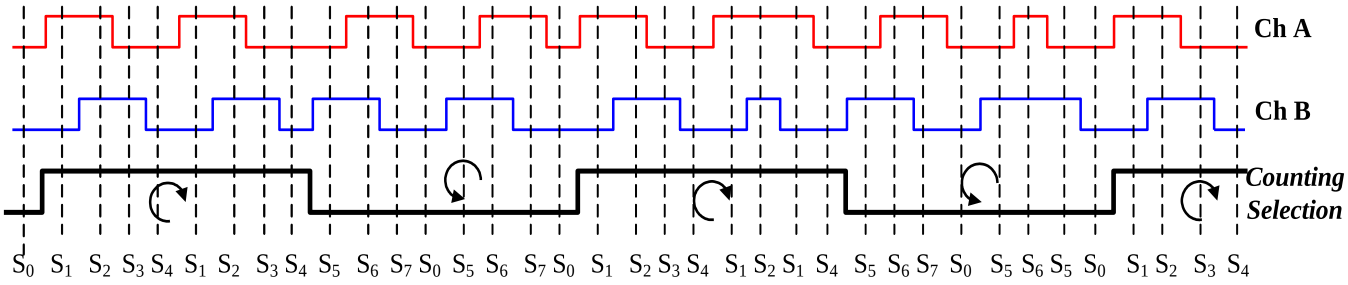

Fig. 4. Signals chart that identify the direction of displacement

Selection. Based on this chart states is constructed the states transition graph shown in Fig. 5.

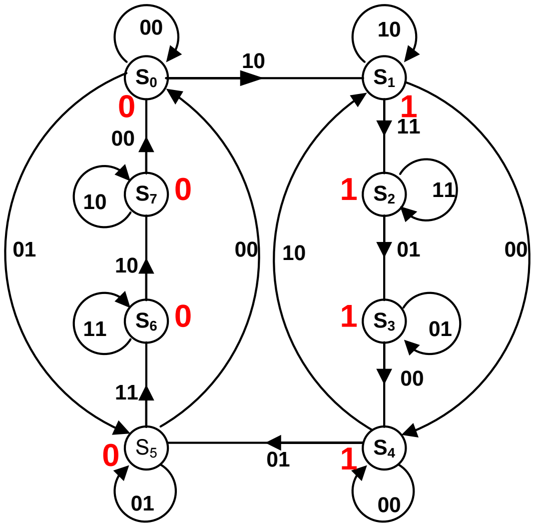

Fig. 5. Transition graph of states

It consists of nodes represented by the 8 states previously identified and arcs represented by binary combinations of both Ch A and Ch B signals through which made the transition between states. Each node is characterized by the logical level of the signal *Counting Selection* and has an arc with logical combination of both Ch A and Ch B signals for which do not change the status for state of respectively node.

B. *Identification the logical function*

Based on the transition graph is built the primitive matrix that contain on the columns the correlation between combination of the input signals Ch A and Ch B and at least one stable state and on the rows contains all possible transitions from one internal stable state. This is accompanied by full matrix of the output that contains the values of output variable during both states and transitions. Number of columns is $M = 2^m$ where m is the number of input variables and in this case $m = 2$ then $M = 4$ columns. Columns are cyclic coded so that from one column to another does not change more than one input variable and therefore is used Gray code.

Primitive matrix of states is presented in Table 1.

Table 1.

States				
AB	00	01	11	10
S_0	$\mathbf{S_0}$	S_5	-	S_1
S_1	S_4	-	S_2	$\mathbf{S_1}$
S_2	-	S_3	$\mathbf{S_2}$	-
S_3	S_4	$\mathbf{S_3}$	-	-
S_4	$\mathbf{S_4}$	S_5	-	S_1
S_5	S_0	$\mathbf{S_5}$	S_6	-
S_6	-	S_5	$\mathbf{S_6}$	S_7
S_7	S_0	-	-	$\mathbf{S_7}$

Counting Selection
0
1
1
1
1
0
0
0

To identify a minimal configuration of the sequentially system is built a reduced matrix of states. Technique used to reduce the number of states from primitive matrix, is based only on the equivalence from the theory of sequential automatic and reduction of state is through merger or annexation in compliance with specific rules [3]. Applying these rules it can obtain the reduced matrix of states and the corresponding output shown in Table 2.

Table 2.

States				
AB	00	01	11	10
S_0	$\mathbf{S_0}$	S_5	-	S_1
S_{123}	S_4	$\mathbf{S_3}$	$\mathbf{S_2}$	$\mathbf{S_1}$
S_4	$\mathbf{S_4}$	S_5	-	S_1
S_{567}	S_0	$\mathbf{S_5}$	$\mathbf{S_6}$	$\mathbf{S_7}$

Counting Selection
0
1
1
0

To obtain the excitation functions of the sequentially system is required to encode the reduced matrix states. It notes the existence of 4 reduced states so that would be necessary to encrypt their by two state variables i.e. x_1 and x_2. To take account of hazard that occurs due to simultaneous change of more than one input variable during transition between two states is used the Gray code [10]. Matrix that represents the encoding of reduced states is shown in Table 3.

Table 3.

States				
AB x_1x_2	00	01	11	10
00	$\mathbf{S_0}$	S_5	-	S_1
01	S_4	$\mathbf{S_3}$	$\mathbf{S_2}$	$\mathbf{S_1}$
11	$\mathbf{S_4}$	S_5	-	S_1
10	S_0	$\mathbf{S_5}$	$\mathbf{S_6}$	$\mathbf{S_7}$

Counting Selection
0
1
1
0

To construct the excitation functions, represented by logic functions for states x_1 and x_2 is necessary to build matrices of transition for reduced states and their number must be equal to the number of state variables. These matrices are shown in Table 4. respectively Table 5. in which notation **x** means states impossible during operation.

Table 4.

State x_1				
AB x_1x_2	00	01	11	10
00	0	1	**x**	0
01	1	0	0	0
11	1	1	**x**	0
10	0	1	1	1

Counting Selection
0
1
1
0

Table 5.

	State x_2				**Counting Selection**
AB x_1x_2	00	01	11	10	
00	0	0	**x**	1	0
01	1	1	1	1	1
11	1	0	**x**	1	1
10	0	0	0	0	0

Applying the method of synthesis of logical functions based on Karnaugh diagrams it can identify the logical functions for the excitation variables [11] (states) x_1, x_2 respectively for function that represent the output signal *Counting Selection* as follows:

$$\begin{aligned} x_1 &= B \cdot x_1 + \overline{A} \cdot \overline{B} \cdot x_2 + B \cdot \overline{x_1} \cdot \overline{x_2} + A_1 \cdot x_1 \cdot \overline{x_2} \\ x_2 &= \overline{x_1} \cdot x_2 + A_1 \cdot \overline{x_1} + \overline{B} \cdot x_2 \\ Counting\ Selection &= x_2 \end{aligned} \tag{1}$$

From equations (1) it can be seen that the output signal is identical to the state x_2, which simplifies implementation with logic gates for the scheme that generate the control signal for counting direction.

Based on logical functions (1) can create a logical diagram of the system through which make selection for direction of counting, shown in Fig.6.

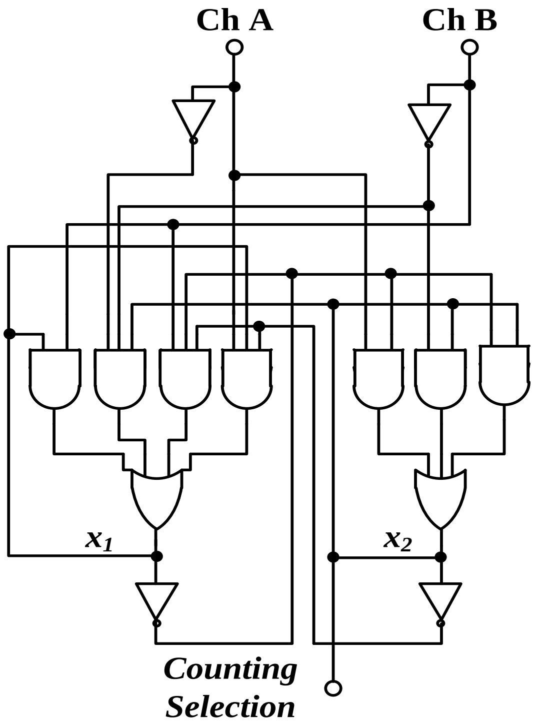

Fig.6. Logical diagram of the *Counting Selection* function

Checking the correctness of system operation was achieved in the first phase, by simulation and for this was used Multisim® program from National Instruments [12]. Based on simulation results presented in Fig.7 it can observe that selection signal Counting Selection changes its logical levels in according to the direction of rotation given by the sequence of pulse trains Ch A, respectively Ch B.

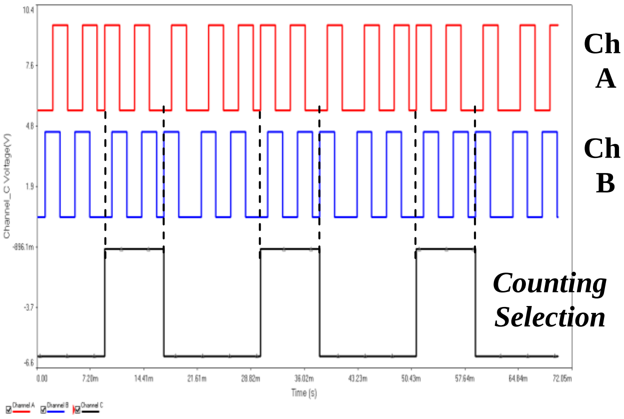

Fig.7. Simulation results for the *Counting Selection* function

C. *Counting Selection function implemented in LabVIEW*

A program developed in LabVIEW is called a virtual instrument (VI) and it has two components the block diagram that represent program itself and the front panel that is user interface. Through such a virtual instrument can be controlled the operation of the data acquisition board PCI-6024 whose digital inputs DIO0 and DIO1 are used for acquisition of Ch A and Ch B signals from incremental sensor.

To achieve virtual instrument is used DAQ Assistant function that creates, edits, and runs tasks using NI-DAQmx that is data acquisition driver [11]. Read through this function

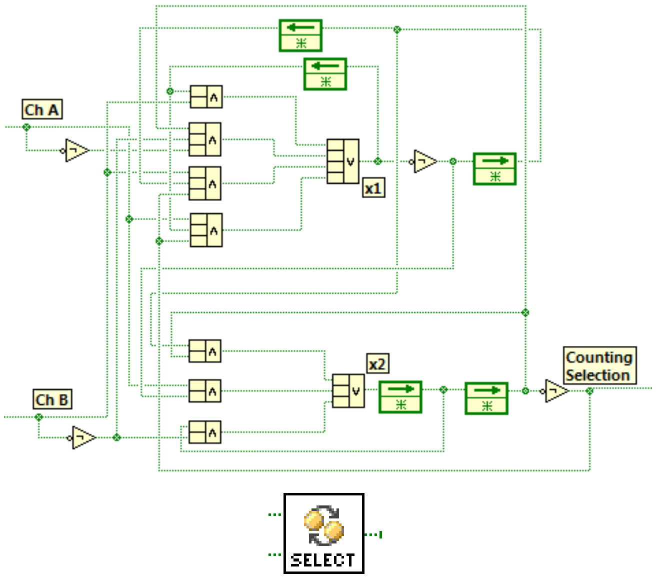

Fig.8. Software implementation for the *Counting Selection* function

is an array with eight boolean components corresponding to the eight digital inputs of the data acquisition board and through Index Array function are selected components with index 0 and 1 that correspond to digital inputs DIO0 respectively DIO1. Thus the two components will be the inputs Ch A and Ch B of the system developed for determining the direction of displacement, system that generates the output signal Counting Selection.

In Fig.8 is shown software implementation for the *Counting Selection* function based on the logical diagram presented in Fig.6. Are used Compound Arithmetic/Logic functions through which can select basic arithmetic or logic operations with two or more variables and Feedback Node which stores data from the virtual instrument execution to be used to the next execution of the virtual instrument.

The software implementation for the *Counting Selection* function it's the SELECT subprogram (subVI) represented by icon (Fig.8.) into the measurement main program.

V. ANGULAR DISPLACEMENT AND SPEED MEASUREMENT

Determination of displacement is achieved by counting the increments that corresponds to the slots of incremental encoder. They are expressed in degrees and the value of an increment for angular displacement corresponds to the relation between the angle at the center of the circle and the number of slots.

Angular speed measurement is based on counting of pulses during the prescribed time. The basic measuring process of pulse during a prescribed time method is shown in Fig.9.

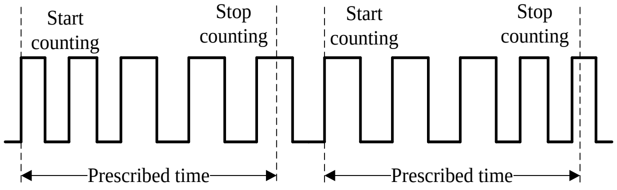

Fig.9. Pulse counting during prescribed time.

The duration of a measurement cycle is fixed and set a priori. The speed pulse counter and the timer are both started at a rising edge of the speed pulse. The pulse counter is stopped when the timer runs to the end of the prescribed time. The angular speed is then derived from the content of the pulse counter and the prescribed time.

This method can result in a loss of up to one speed pulse. As the duration of speed pulse increases with decreasing speed, this method has poor measurement accuracy at low speed.

The algorithm used to measurement the angular displacement and speed is shown in Fig.10 [13].

The data acquisition board PCI-6024E used to build the virtual instrument has two counters *ctr0* and *ctr1*.

For the two counters *ctr1* and *ctr0* counting values may be increasing (Count Up) when their value increases with each pulse applied to the entry CtriSource (i = 0 or 1) in domain [0 ... 2^{24} = 16777216] or may be decreasing (Count Down) when its value decreases with each pulse applied to the entry CtriSource (i = 0 or 1) in domain [2^{24} = 16777216 ... 0].

The two counters are used as follows: counter *ctr0* to count downward, counter *ctr1* to count upward.

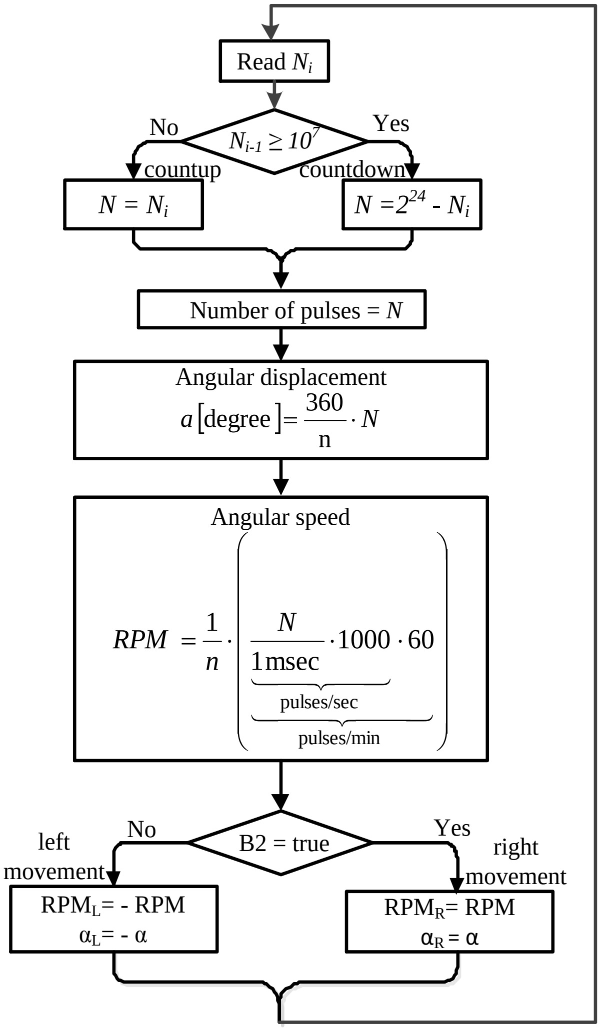

Fig.10. Angular displacement and speed measurement algorithm

To determine the correct values for both displacement and speed is required as counting upwards or downwards, depending on the direction of motion, to be made permanent from the value determined in previous measurements. To do that is considered a constant, 10^7, and in relation to its value, is identified the counting sense, ascending or descending

So if it detects, from the previous measurement, a number of pulses N_{i-1} higher than 10^7 is considered selected counter *ctr1* and counting is carried downwards ever since the maximum counter value (2^{24}) which requires that the value of the number of pulses and hence calculation of displacement and angular speed values are obtains by difference between constant 2^{24} and value of *N*.

If it detects, from the previous measurement, a number of pulses N_{i-1} lower than 10^7 is considered selected counter *ctr0* and counting is carried upwards ever since the minimum counter value (0) and calculation of displacement and angular speed values are obtains by value of *N*.

The number of pulses *N* result for current measurement is stored into the variable CTR, that is:

$$CTR = N = \begin{cases} N_i & if \;\; N_{i-1} > 10^7 \\ 2^{24} - N_i & if \;\; N_{i-1} \geq 10^7 \end{cases} \qquad (2)$$

so that this number will be available for the next measurement.

Value of angular speed expressed in revolutions per minute (RPM) is computed with (3) where *n* represents the number of pulses corresponding to one complete revolution and *N* represents the counting pulses:

$$RPM = \frac{1}{n} \cdot \left(\underbrace{\underbrace{\frac{N}{1\ \text{msec}}}_{pulses/\sec} \cdot 1000 \cdot 60}_{pulses/\min} \right) \qquad (3)$$

The implementation by virtual instrumentation for the relation (3), that is a subprogram (subVI) called RPM in the main program is shown in Fig.11.

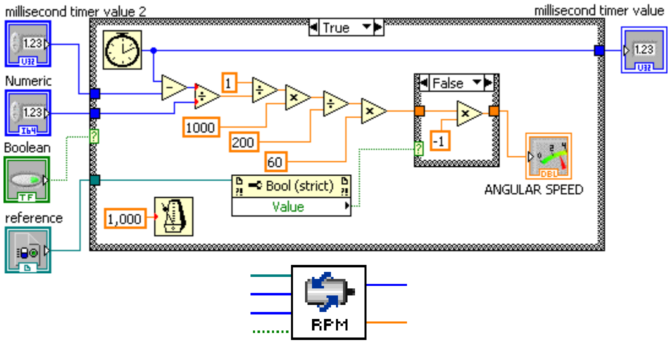

Fig.11. The virtual implementation of the angular speed measurement

Getting prescribed time necessary for calculating speed is achieved by using function Tick Count (ms) that returns the timer value, in milliseconds; between passages consecutive in the While loop.

Value of angular displacement α, expressed in degrees, is computed with relation (4) that has the virtual implementation shown in Fig.12

$$\alpha[\text{degree}] = \frac{360}{n} \cdot N \qquad (4)$$

Motion in clockwise (right direction) is considered that be displayed with positive sign and motion in counterclockwise (left direction) is considered that be displayed with negative.

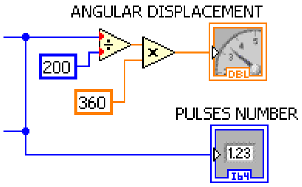

Fig.12. The virtual implementation of the angular displacement measurement

VI. MEASUREMENT SYSTEM IMPLEMENTATION WITH VIRTUAL INSTRUMENT

The main program algorithm is shown in Fig.13 and this includes SubVI's SELECT and RPM and has a like basic structure an While Loop that ensure the continuous running of the program until the user stop it through the STOP button [13].

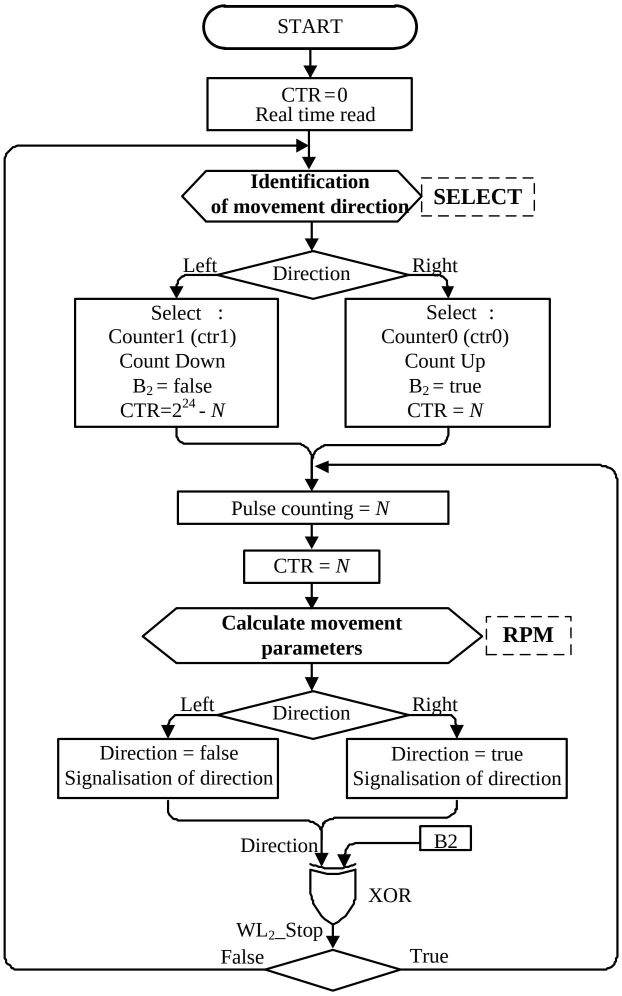

Fig.13. Main program algorithm

Actually two loops are used, a main While Loop and a secondary While Loop.

The basic While Loop begins with the identification of the movement direction; continue with settings for counter selection, counting direction, variable CTR value and boolean variable B2 value and after that call the secondary While Loop.

The secondary While Loop allows the displacement and velocity measurements in a situation that does not change the direction of movement by successive readings of the incremental transducer. Control of this loop is made by testing the boolean variable WL2_Stop:

$$WL_2\_Stop = Direction \oplus B2 \qquad (5)$$

where *Direction* is a boolean variable with the same values like *Counting Direction* and values of boolean variable B2 are: "true" for Count Up and "false" for Count Down, set on main While loop.

Thus, if *WL2_Stop* is "true" means that is identified the same direction of movement as in the basic While Loop and continues to measure the displacement and velocity, and if *WL2_Stop* is "false" means that is has changed the direction of movement and returns to the basic While Loop.

Based on this algorithm is built the virtual instrument whose block diagram is shown in Fig.14.

applying it to the selection of a terminal structure of the Case structure. Through this structure is also selected one of the counters *ctr1* or *ctr0* so that the counting upwards is performed by counter *ctr0* and counting downwards is performed by counter *ctr1*. For these counters counting values may be increasing (Count Up) when their value increases with each pulse applied to the entry CtriSource (i = 0 or 1) or may be decreasing (Count Down) when its value decreases with each pulse applied to the same entry CtriSource.

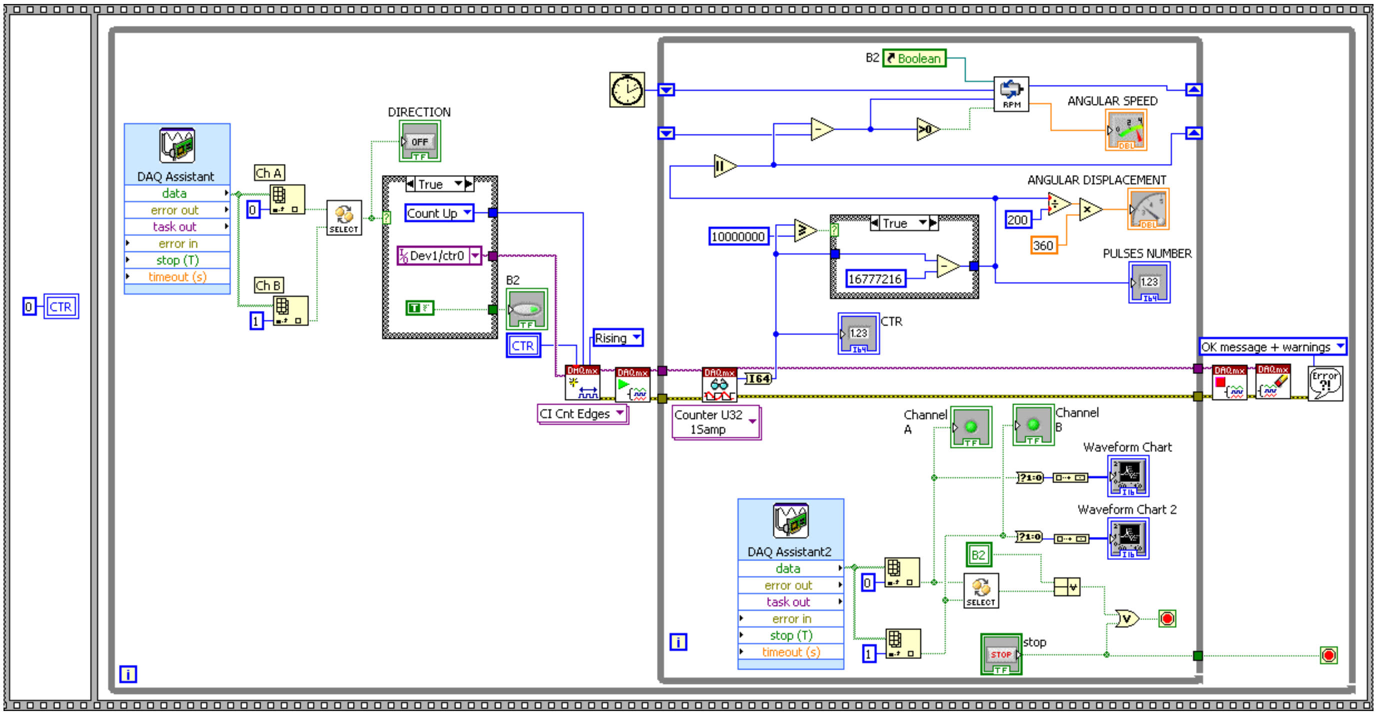

Fig.14. Diagram bloc of the virtual instrument

Acquisition is executed in two sequences and the program begins with reset of the local variable CTR and timing setting that will be used for defining the graphical representation of X-signal for the Channel A (Ch A signal) respectively Channel B (Ch B signal) graphical indicators.

The input signals Ch A and Ch B are taken from the incremental sensor through line 0 (DIO0) and line 1 (DIO0) of the digital port (Digital I/O) from the data acquisition board PCI – 6024E using DAQ Assistant function.

This function creates, edits, and runs tasks using NI-DAQmx that is data acquisition driver [14]. Reading through this function is an array with eight boolean components corresponding to the eight digital inputs of the data acquisition board and through Index Array function are selected components with index 0 and 1 that correspond to digital inputs DIO0 respectively DIO1. Thus the two components will be the inputs Ch A and Ch B of the system developed for determining the direction of displacement, system that generates the output signal Counting Selection.

Once direction is selected, this it will be displayed on the front panel and the selection signal is also used for selecting the direction of counting (Count Up or Count Down) by

The number *N*, which counters values are incremented or decremented is returned by the function DAQmx Read (Counter U32 1CH 1Samp) [14]

Motion in clockwise is considered that be displayed with positive sign and motion in counterclockwise is considered that be displayed with negative. This convention require continuous tracking of the value of the two counters and this is achieved by using local variable CTR whose value is loaded into each of the two counters selected according counting sense through the input parameter initial count of the DAQmx Create Virtual Channel function.

Displaying number of pulses and calculating the displacement and angular speed, given the agreement between the direction of motion and sign of these dimensions, is achieved through a Case structure.

Selection of the two cases is done through the comparison between the value *N* that represent the output of the function DAQmx Read and constant 10^7 (considered to be cover for measurements made under the following conditions: measurement time for one direction of displacement about 33 minutes, n = 200 slots and maximum speed 1500 rev/min.)

Views of the front panel that is user interface, corresponding to the two directions of movement corresponding to two states of operation for the virtual instrument and an overview of the stand used for the virtual instrument testing are presented in Fig.15.

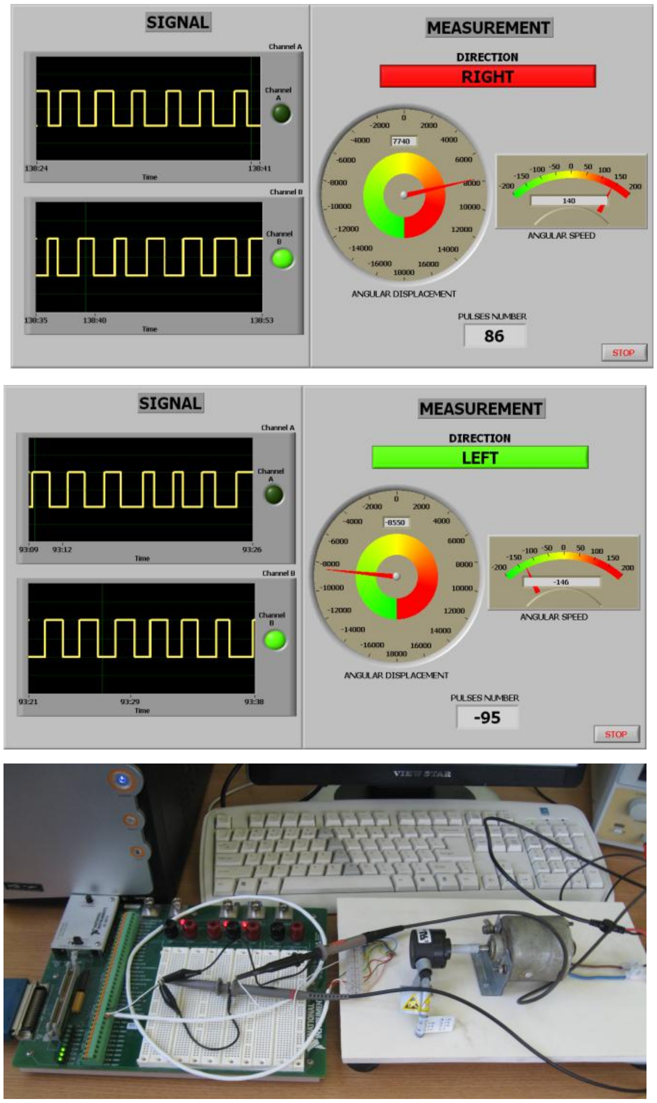

Fig.15. User interface of the virtual instrument and overview of the testing stand

VII. CONCLUSION

Using the virtual instrument in this form have a very high interest for data acquisition systems that no accept at their counters, signals provided by quadrature encoder and these data acquisition systems are used to measure displacements and/or angular velocities.

Function testing was done both for direction displayed and for measuring displacements or angular velocities. Tests were performed using quadrature encoders with 4, 200 (E6A2-CW5C) respectively 500 (HEDS – 5500) pulses/revolution for a wide range of speeds, connected to digital inputs of the PCI – 6024E data acquisition board, from which were used and the two counters ctr1 and ctr0

REFERENCES

[1] Webster.G.J. *The measurement, Instrumentation and Sensors Handbook*. CRC Press LLC, 1999, ch.6

[2] D.M. Purcaru, E. Niculescu, I. Purcaru, Measuring Method for Computerized Study of Rotary Incremental Encoders, *WSEAS Transactions on Electronics*, Vol. 3, 2006, pp. 349-355.

[3] D. M. Purcaru, E. Niculescu, I. Purcaru, "Experimental Measuring System with Rotary Incremental Encoder" in *WSEAS Transactions On Advances In Engineering Education*, Volume 4, September 2007, pp.180-186

[4] * * *, *LS7084 Datasheet* . LSI Computer Systems, Inc. Available: http://www.datasheetcatalog.org/datasheet/lsi/LS7083.pdf

[5] * * *. *6023E/6024E/6025E User Manual*. National Instruments Corporation. Austin, Texas, 2000.

[6] * * *, *DAQ M Series User Manual*. National Instruments Corporation. Austin, Texas, 2008.

[7] Yuhua L., Fengshou G. and others. The measurement of instantaneous angular speed. *Mechanical Systems and Signal Processing*, vol.19 (2005), pp.786–805

[8] H. K. Jafari, Design and Implementation of DC Motor Speed Controller Using Fuzzy-Adaptive Controllers. *WSEAS Transactions on Circuits and Systems*. Vol. 7, April 2008. pp. 203-212

[9] N. Patrascoiu, A. Poanta, M. Tomus, B. Sochirca, Detection of motion direction implemented by virtual instrumentation. *Recent Advanced in Automation & Information WSEAS Proceeding* (ICAI'10), (72-75)

[10] J. Carroll, D. Long, *Theory of Finite Automata*. Prentice Hall, New Jersey, 1989

[11] A. Valachi, M. Timis, M. Danubianu, Some Contributions to Synthesis and Implementation of Multifunctional Registers. *2nd WSEAS International Conference on Multivariate Analysis and its Application in Science and Engineering*. pp. 146-149

[12] * * *. *Multisim. User Manual*. National Instruments Corporation. Austin, Texas, 2008.

[13] N. Patrascoiu, A. Poanta, A. Tomus, B. Sochirca, A software solution for displacement and angular speed measurement through virtual instrumentation, *6th WSEAS International Conference on Applied Computing*, 2010, pp. 11-15.

[14] * * *, *NI-DAQ Function Reference Manual for PC Compatibles*, National Instruments, 1995, part number 320499C-0.

Nicolae Patrascoiu is born in Horezu, Gorj County, Romania on April 19, 1959, graduate high school "Electronica" in Bucharest in 1978 and University of Petrosani, Romania in 1985 winning the title engineer in electromechanical engineering, presented his doctoral thesis at the University of Petrosani in 1998, in Automation field, receiving his Ph.D.

His work experiences include: Researcher at Institute of Technological Engineering and Projection in Petrosani and since 1988, successively, Assistant, Lecturer and today Associate Professor at the University of Petrosani. He has published over 60 scientific papers and five books including: Sensors and Transducers (in Romanian) (Petrosani, Universitas, 2000), Data Acquisition and Processing. Virtual Instrumentation (in Romanian) (Bucharest, EDP, 2005). Instrumental electronics, data acquisition and virtual instrumentation are his research interest

Dr. Patrascoiu is member of Romanian Society of Control Engineering and Technical Informatics, secretary of the Petrosani branch