J. Basic. Appl. Sci. Res., 3(9)7-19, 2013
© 2013, TextRoad Publication

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

An Implementable Speed Measurement Method in Order to Eliminate Quantization Error while Using Rotary Incremental Encoders

Mahdi Maaref, Alireza Rezazadeh, Hassan Shokouhandeh

School of Electrical and Computer Engineering, Shahid Beheshti University, Iran

ABSTRACT

In this paper, we propose an implementable method to calculate motor speed base on incremental encoders. This method solves two major problems of calculating speed by the aid of quadrature encoder signals; first the discrepancies of speed pattern especially in high and low speeds and second the quantization error which occurs by differentiation of discrete quantity (i.e. motor position). In order to get a method which, has an ability of calculating the speed in wide range, it is inevitable to use different algorithms for different ranges of speed. But the important issue is to have a soft switching between these algorithms which is the scope of this approach.

Our proposed method achieves high accuracy from low resolution encoders which is more efficient. The simulation and experimental results validate the precision of proposed method especially, in the moment of changing measurement algorithm.

KEYWORDS—rotary incremental encoder, quadrature pulses, encoder resolution, sampling time.

I. INTRODUCTION

Motor Speed control is one of the most attractive issues in mechanical systems and by developing new automatic drivers for motors, it gained significant attention worldwide. In recent years a huge amount of papers published discussing different methods of measuring, estimating, modeling and controlling the motor speed [1-3]. It is undeniable that in every control loop the feedback measurement is needed. In rotational systems, some sensors are utilized in order to measure feedback speed or position as well as absolute and incremental encoders [2-5]. The incremental encoders measure the position with a precise accuracy and repeatability and based on their structures of pulse generation, benefit from high performance and flexibility [6]. Furthermore, their less price and divers range of pulses (from 2500 pulses in custom servo ac drivers to more than 130000000 pulses in high performance direct drive motors) make them useful in wide range of applications. But some drawbacks such as limited controller bandwidth in slow speeds and quantization error (which leads to some fluctuations superimposed on measured speed) can generate significant distortion and performance degradation in the system.

To solve these problems, many literatures have been developed manipulated methods [7-10] but most of them suffer from huge amount of calculations and it is hard to implement them in many dedicated processors such as FPGAs and small Microchips.

The other solution is to use encoders with higher resolution and prevent involving with complicated calculations.

Since such encoders are more expensive, lots of literatures [4-5] focused on developing efficient methods using low level calculations. In this case, there are 2 custom approaches that measure speed based on incremental encoders:

1- Counting pulses during constant time
2- Measuring time between two successive pulses

These two methods have been presented [5], known as frequency/period measurement methods respectively. Also a relation to determine the quantization error has been developed and finally a novel mixed mode method is presented.

In [6] and [7], an incremental encoder is simulated using Matlab/Simulinks and the basic equations of pulses A and B based on angle of the shaft are established.

A novel technique aimed at increasing encoder resolution is presented in [8]. The proposed logical circuit consists of D-type flip-flops and can be implemented in FPGA. Furthermore speed is calculated using data fusion functions.

In [9], an interesting method has been presented which calculated speed using fast Fourier transformation. This method deals with high amount of calculations and it is hard to be implemented in simple processors. But the simulations showed that this method has a very precise accuracy in calculating the speed.

In order to manipulate stored encoder data, a method is presented in [10] which uses the time stamping. The proposed method extended the observation interval of stored encoder events using a skip operation. Although the simulation results show an improvement in velocity estimation, this method has complicated and time consuming calculations and it is hard to get proper controller bandwidth. It is obvious that there are different methods in order to measure the speed using

[*]**Corresponding Author:** Mahdi Maaref, School of Electrical and Computer Engineering, Shahid Beheshti University, Iran. m_maaref@sbu.ac.ir,

incremental encoders. However, in this survey, the main idea is to find the source of quantization error and manipulate an algorithm to eliminate it. This will lead to establish a method which satisfies the required accuracy and controller bandwidth while using fewer calculations.

Since proposed algorithm has limits on measuring low speeds, an alternative algorithm is presented in order to measure the speed in lower region. Thus, the marginal speed between two proposed algorithms calculated and also an integrated method presented which switches between them. Mathematically, we prove that proposed integrated method has ability to measure the speed continuously even at marginal speed and the simulation and experimental results validate this fact.

This paper is organized as follows: Basic concepts of an optic incremental encoder are presented in section II. In section III a technique presented and employed aimed at increasing the encoder resolution. In sections IV and V, the proposed methods are established for speed measuring in high and low speeds respectively. In VI, the integrated model is completed. So, the numerical example is resolved again using proposed method to compare the strength of algorithms. In section VII simulation results are depicted and the proposed method is validated. Also in section VIII an experimental results are explained. At the end, in section IX we conclude the advantages of proposed method.

II. Basic concepts of incremental encoders

An incremental encoder is one of the most widely used of position sensors due to its low cost and ability to produce precise signals that can interpret to movement information such as position and velocity [6-8]. There are different kinds of incremental encoders like magnetic or optic ones. Apart from their structure, the output signals of incremental encoders are the same. Basically an incremental encoder is a transducer between rotary motion and two related pulses called “A Pulse” and “B Pulse”. In a case of optic one, it consists of a rotary disc and two light emitting sensors to produce these pulses. As shown in Fig.1, rotary disk has certain gaps, so when it rotates, the light beams are reflected sequentially and by transforming them into electrical pulses, the output signal of encoder is produced. Also it can be seen that light emitting sensors are sedentary and independent from encoder’s shaft.

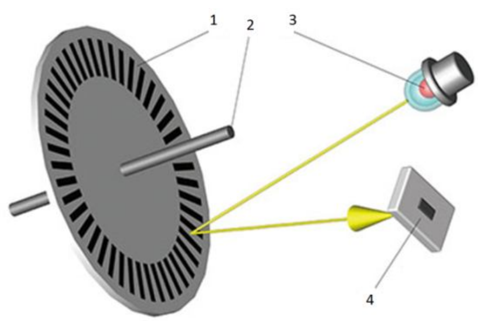

Fig. 1: Construction of an optic incremental encoder [11]

So the regulated position of sensors A and B can lead to two output wave forms which are 90 degree out of phase. This helps us to detect the direction of movement due to lead or lack of phases. In this Figure only one of the light sensors is traced and numbered parts are as:

1- Main disc and gaps
2- Motor shaft
3- Light emitting element
4- Light receiving element

An example pattern of pulses A and B in incremental speed situation is shown in Fig. 2. By comparison of ω_1 and $\omega_2$in this Fig., it can be seen that by increasing the rotating speed, frequency of pulses increase.

Furthermore, Number of gaps on the rotary disc (N_E) is another variable that affects frequency of pulses A and B. In other words N_E illustrates the number of produced pulses due to 360 degree rotation. Thus, if the encoder produces N_0 pulses in time T_S, the angle and speed of rotation are calculated as follows:

$$\theta = \frac{N_0 .360}{N_E} \, [\deg ree] \qquad (1)$$

$$\omega = \frac{N_0.360}{N_E.T_S} [\deg ree.s^{-1}] \qquad (2)$$

Consequently, if the encoder rotates at speed ω [degree.s^{-1}], the frequency (f_P) of pulses and time between two adjacent ones (T_P) are calculated as:

$$f_p = \frac{N_0}{T_S} = \frac{\omega.N_E}{360} \ [s^{-1}] \ , \ T_p = \frac{360}{\omega.N_E} \ [s] \qquad (3)$$

Equation (2) establishes the easiest methods calculate the rotating speed which is known as Frequency Measurement in some literatures [4]. This method calculates the speed by assuming the constant value for sampling time (T_S) and

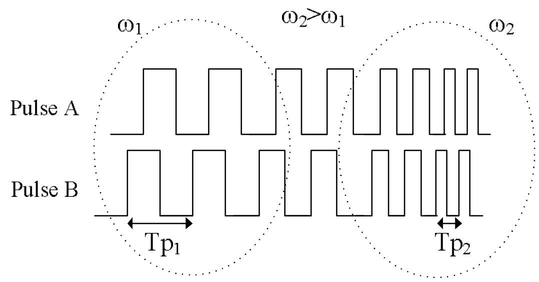

Fig. 2: Diagram for output pulses of encoder for an incremental speed pattern

counts the observed pulses (N_0) during that time interval. Then speed can be calculated using equation (2). But, it is obvious that this method suffers from quantization error, because N_E and T_S are supposed constant and the only variable is N_0 which accepts discrete values. So, although speed changes continuously, this method measures it discretely. Thus, in order to decrease quantization error, we have to increase N_E, because speed has inverse relationship with N_E and increasing N_E, decreases the impact of step changes of N_0. Calculating the minimum required N_E will help us to choose the most efficient encoder in order to satisfy design's features. Decreasing the value of N_E, decreases the encoder's cost, but there are two major constraints: first, accuracy of speed measurement and second, data sampling time.

By assuming the maximum acceptable distortion is equal to Δω% of rotating speed, and the sampling time is equal to T_s, then the minimum required number of N_E is calculated as follows:

$$\frac{\Delta\omega}{100}.\omega = \frac{1\times 360}{N_E.T_S} \Rightarrow N_{E\min} = \frac{36000}{T_S.\omega_{\min}.\Delta\omega} \qquad (4)$$

In equation (4), ω_{min} determines the minimum rotating speed. For a numerical example, assume that the minimum acceptable distortion is 0.1% of rotating speed and ω_{min} is equal to 0.01 d.s^{-1} and T_s is 10ms. So, equation (4) calculates the minimum N_E as follows:

$$N_{E\min} = \frac{36000}{10\times 10^{-3}\times 0.01\times 0.1} = 360000000 \qquad (5)$$

As it shown in equation (5), quantization error forces very high value of N_E in order to achieve desired accuracy and sampling time. It is obvious that preparation of encoder with 360000000 pulses per round is very hard and costly. To solve this problem, this approach proposed a method to eliminate the quantization error in order to reach high accuracy while using low resolution encoders.

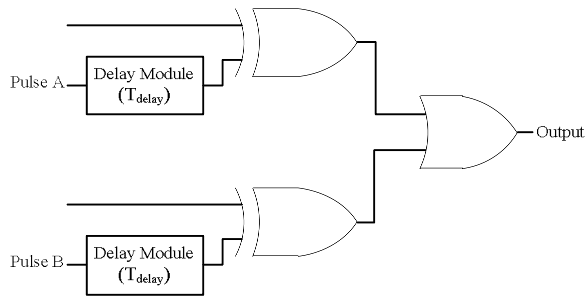

Fig. 3: Logical circuit in order to increase encoder resolution

III. Resolution Increasing

Before we go to the measuring method, we present a technique to increase encoder resolution. The main idea is to use both edges of pulses A and B. So it can be supposed that the number of encoder pulses per round is $4 \times N_E$ instead of N_E. Fig. 3 shows a proposed logical circuit that multiply N_E by 4. Increasing the encoder resolution up to $4 \times N_E$ and making pulse width independent from rotating speed are two major benefits of this circuit. As it is depicted in Fig. 4, output pulse width corresponds to delay time, neither than rotating speed and it has same value of T_{delay} in two different speeds ω_1 and ω_2. So, based on equations (1-3) and explained technique, the rotating speed, pulse frequency and time between two adjacent pulses can be rewritten as follows:

$$\omega = \frac{N_0.90}{N_E.T_S} \; [\deg ree.s^{-1}] \qquad (6)$$

$$f_p = \frac{N_0}{T_S} = \frac{\omega.N_E}{90} \; [s^{-1}] \; , \; T_p = \frac{90}{\omega.N_E} \; [s] \qquad (7)$$

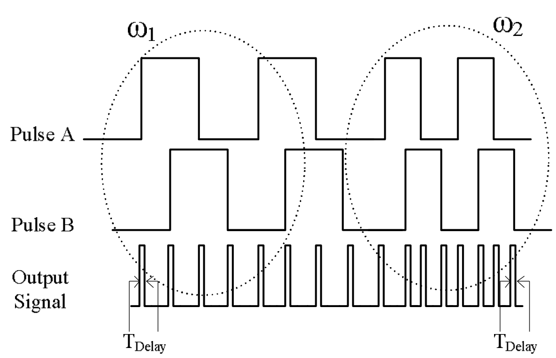

Fig. 4: The output of proposed logical circuit. The widths of output pulses are equal in different speeds

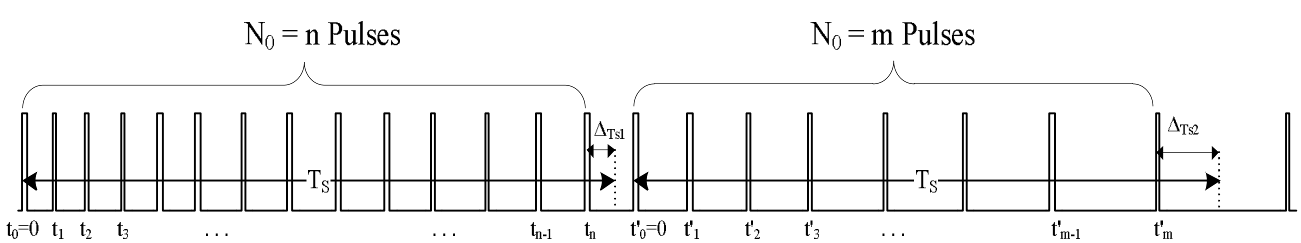

Fig. 5: Diagram of two periods of speed measurement using proposed method

Note, that the maximum time delay must be less than minimum time between two adjacent pulses. So:

$$T_{DelayMax} < T_{PMin} \Rightarrow T_{DelayMax} < \frac{90}{N_E \,.\omega_{\max}} \qquad (8)$$

IV. PROPOSED METHOD

As mentioned before, using Frequency Measurement caused quantization error. This error occurs due to counting pulses during constant sampling time. In the other word, as it can be seen in equation (6), the left hand side has continuous variable (ω), but the right hand side has too constant variables (N_E, T_S) and one discrete variable (N_0). If we can change one of the constants into continuous variable, both sides have continuous values and quantization error is eliminated. Thus, this approach focused on changing T_S into continuous variable instead of constant one. Fig. 5 shows a proposed algorithm flowchart in order to regulate T_S based on time of pulses observation. These pulses are produced by logical circuit presented in section III and all of them have the same width equal to T_{delay}. If T_{delay} has small values, the difference between times of rising and falling edges of pulse (posedge and negedge respectively) is negligible. But in order to reach to higher accuracy in calculations, the time of posedges of pulses are sampled. The algorithm can be summarized in 4 steps:

1- Start data sampling at the posedge of encoder pulse. (t= 0)
2- Count the number of pulses and save encoder posedge times until $t < T_S$
3- Calculate the speed as a division of counted number of pulses and last pulse posedge time.
4- Wait until next posedge of encoder and go to step one.

Fig. 6 shows two period of speed measurement using proposed method. In this figure we supposed that speed decreases continuously. As it can be seen from Fig. 6, t = 0 is fixed into the first pulse posedge. As time passes, the pulses are counted (N_0) and also times of pulse posedges are saved (t_0, t_1,…,t_n). After passing T_S seconds, algorithm calculates the speed as follows:

$$\omega = \frac{N_0.90}{N_E.t_n}\,[\deg ree.s^{-1}] \qquad (9)$$

where, t_n is time of last pulse posedge during sampling time T_S. As it shown in Fig. 6, Δ_{Ts} is defined as a time interval between last posedge and the end of sampling time (T_S). So, speed can be calculated as:

$$\omega = \frac{N_0.90}{N_E.(T_S - \Delta_{TS})}\,[\deg ree.s^{-1}] \qquad (10)$$

In this equation, both hand sides contain continuous variables (ω and Δ_{Ts}). So, it is possible to measure the speed continuously which, leads to elimination of quantization error.

Fig. 7 shows N_0 and Δ_{Ts} while ω decreases. It can be seen that when rotating speed decreases, Δ_{Ts} increases until Δ_{Ts} reaches T_P. Then, Δ_{Ts} turn to zero and N_0changes to $N_0 - 1$ and again by decreasing ω, Δ_{Ts} increases from zero to new T_P. The idea of adjusting T_S based on time of pulses is used in some other approaches [5]. But the strength of proposed method is in the ability of measuring the speed continuously while N_0 changes discretely. We mathematically prove it and also simulation results validate this fact that in proposed method, speed is measured continuously even when N_0 switches to N_0-1 which is neglected in previous works. Fig. 8 shows the moment before and after switching N_0 to N_0-1.At speed ω_1, Δ_{Ts} isequal to zero. It means that the time window T_S spreads from one posedge to another posedge. So, with very small decline in speed $(\omega_1 - \omega_2 \approx 0)$ last posedge moved out from time window and N_0 changes toN_0-1. Mathematically we have to prove that ω_1 and ω_2 (measured speed before and after switching N_0 to N_0-1) are equal in order to show that our method measures the speed continuously even when N_0 changes. As explained before, at the moment of switching, Δ_{Ts} switches from zero to T_P (Fig. 8) . If we assume $\omega_1 = \omega_2$ then:

$$\omega_1 = \omega_2 \Rightarrow \frac{N_0.90}{N_E.(T_S - 0)} = \frac{(N_0 - 1).90}{N_E.(T_S - T_P)}$$

Based on equation (7)

$$\frac{N_0.90}{N_E.T_S} = \frac{(N_0 - 1).90}{N_E.(T_S - \frac{90}{\omega.N_E})} \Rightarrow \frac{N_0}{T_S} = \frac{N_0 - 1}{T_S - \frac{90}{\omega.N_E}}$$

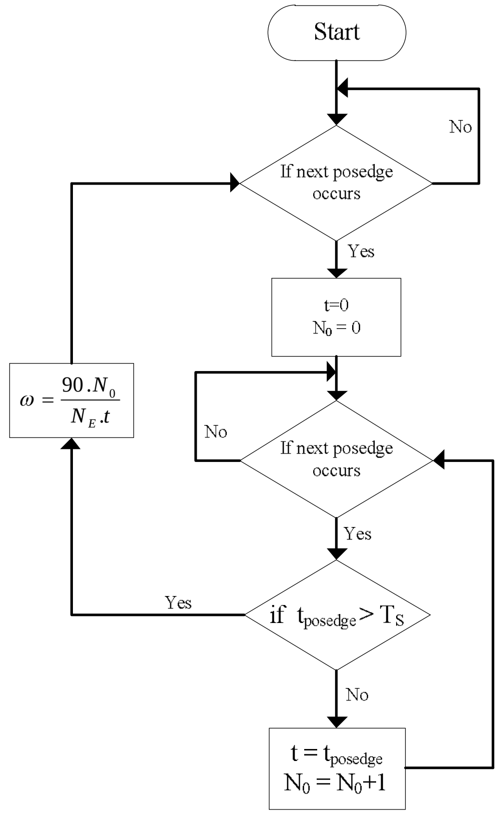

Fig. 6: Implementable flowchart for proposed algorithm

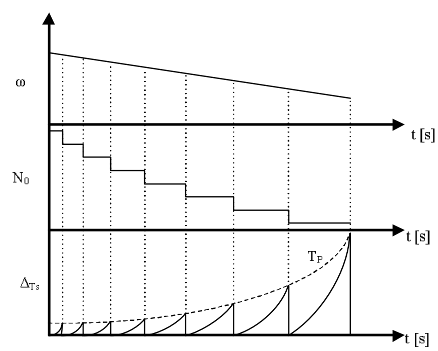

Fig. 7: Behavior of ΔTs and N0 while speed decreasing

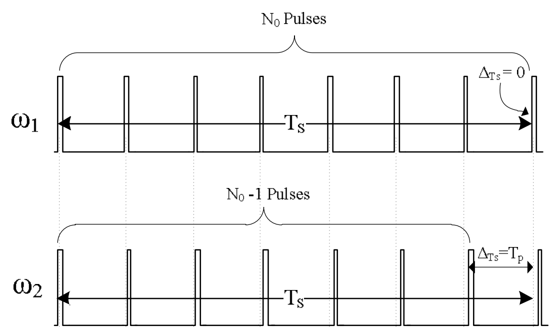

Fig. 8: Speeds before and after switching N0 to N0-1. Mathematically ω1 = ω2

Based on equation(6)

$$\frac{N_0}{T_S} = \frac{N_0 - 1}{T_S - \frac{90}{\frac{N_0.90}{N_E.T_S}.N_E}} \Rightarrow \frac{N_0}{1} = \frac{N_0 - 1}{1 - \frac{1}{\frac{N_0.1}{1}}}.$$

$$N_0 = \frac{N_0 - 1}{1 - \frac{1}{N_0}}. \Rightarrow N_0 = N_0 \qquad (11)$$

Equation (11) validates the hypothesis of $\omega_1 = \omega_2$. So, the proposed method doesn't suffer from quantization error which occurs at the moment of switching encoder pulses from N_0 to N_0-1. But there is an important issue corresponding to measuring Δ_{Ts}. Mostly in programmable devices, time intervals are measured with high frequency clocks. Fig. 9 shows measuring Δ_{Ts} using high frequency clock f_{hf}. So, in implemented design, equation (10) is changed to:

$$\omega = \frac{N_0.90}{N_E.(T_S - \frac{n}{f_{hf}})} [\deg ree.s^{-1}] \qquad (12)$$

where, f_{hf} is frequency of used clock and n is number of counted clocks among Δ_{Ts}. Thus, it seems that there is another quantization error due to counting "n" in equation (12). Certainly it is impossible to eliminate it because of time measurement limitations, but it is very simple to calculate the required boundary of f_{hf} in order to achieve tolerable distortion in speed measurement. From equation (12) Δω can be calculated as:

$$\Delta\omega = \frac{N_0.90}{N_E}.(\frac{1}{T_S} - \frac{1}{T_S - \frac{1}{f_{hf}}}) [\deg ree.s^{-1}] \qquad (13)$$

which yields:

$$\Delta\omega = \frac{N_0.90}{N_E.T_S}.(\frac{1}{T_S.f_{hf} - 1}) \Rightarrow$$

$$\frac{\Delta\omega}{\omega} = (\frac{1}{T_S.f_{hf} - 1}) \qquad (14)$$

As a numerical example, common values for f_{hf} and T_s are 50 MHz and 1ms respectively. So from equation (14) the distortion in speed due to high frequency quantization error can be calculated as $\frac{\omega}{2 \times 10^5}$ which is obviously negligible.

V. Period Measurement

The proposed method has a down limitation on speed. Because when ω decreases, T_p increases while T_S is roughly constant. So in low speeds, there are not enough pulses in time window to be counted and our method isn't useful. The critical speed defines as a minimum speed which can be calculated. In this speed T_p and T_S are equal. It means that there are exactly two pulses at the beginning and end of time window (Fig. 10). The value of critical speed can be calculated simply as follows:

If $\omega = \omega_{critical}$ then $T_P = T_S$

So, based on (7)

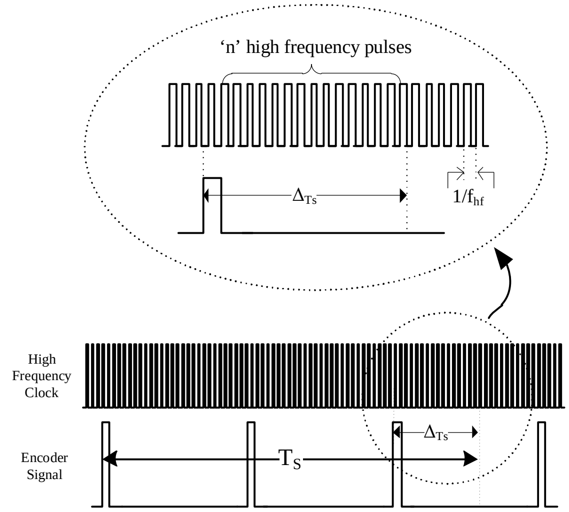

Fig. 9: Measuring the Δ_{Ts} using high frequency clock f_{hf}

$$T_S = \frac{90}{\omega . N_E} \Rightarrow \omega_{critical} = \frac{90}{T_S . N_E} \qquad (15)$$

Equation (15) establishes a tradeoff design between encoder pulse number per round (N_E) and band width (T_S).In other words, it shows that if desired application requires fast measurement of speed among the low speeds, N_E must get higher values according to equation (15).In the other hand, if design constraints, forced us to use low resolution encoder (low N_E); equation (15) calculates the limit on T_S value in order to calculate the low speeds precisely. As it shown in Fig. 10, if speed decreases to lower values than critical speed ($T_P > T_S$), it is impossible to measure it during time windows with length of T_S. Alternatively, there is two major ways to solve measuring problem in lower speeds:

1- Increasing N_E in order to set $\omega_{critical}$ equal or less than minimum required rotating speed.
2- Using period measurement method in speeds lower than critical speed.(increasing T_S dynamically)

Increasing N_E is possible by several ways as well as adding mechanical gearbox to the system or using high resolution encoder via direct drive systems. Using mechanical gearbox multiplies the N_E by gearbox factor and increases N_E. But gearbox has some drawbacks. For example add backlashes and reduce positioning accuracy. However, increasing N_E is costly and in some cases impossible to find encoder which satisfies the constraint on equation (15). So the period measurement is a good alternative option to manipulate measurements in low speeds. The main idea is to increase T_S from one pulse posedge to its adjacent pulse posedge. Here again high frequency clock is used to measure T_S. An Implementable flowchart for the period measurement is depicted in Fig.11.
In this method T_S and ω calculated as follows:

$$T_S = \frac{n}{f_{hf}} \qquad (16)$$

$$\omega = \frac{f_{hf} . 90}{N_E . n} \; [\deg ree . s^{-1}] \qquad (17)$$

where, "n" is the number of high frequency clocks counted between two adjacent pulses' posedges. Based on equation (16), in the period measurement, T_S is not constant and has inverse relationship with rotating speed. So, maximum value of T_S occurs in minimum rotating speed which, is calculated as:

$$T_{S\max} = \frac{90}{N_E . \omega_{\min}} \; [s] \qquad (18)$$

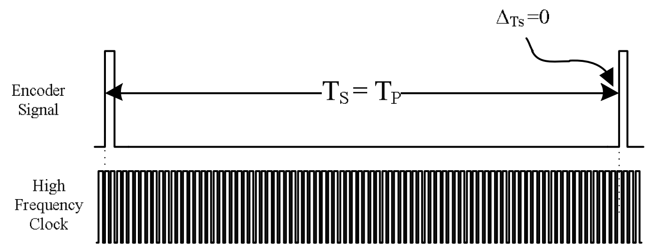

Fig. 10: Diagram of output pulses at the critical speed

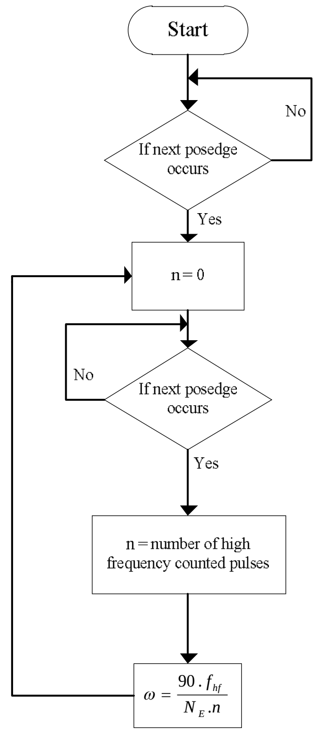

Fig. 11: An implementable flowchart for period measurement

VI. Integrated method

In sections IV and V, two different methods were presented in order to measure the rotating speed. As discussed before pulse counting was more suitable for high speeds and measuring time distance between two adjacent pulses was useful in low speeds. Also the marginal speed between these two methods were calculated and defined as a critical speed. However, this survey focused on developing previous works in order to eliminate quantization error. So, to devise an integrated method which has ability of measuring speed in wide range, it is enough to merge these two methods together. In other words, we have to change the measurement method in each step, only if the comparison between the measured speed and the critical speed shows its necessity. The idea of merging different methods together is not a new one. In recent years, several papers [4-6] published regarding to develop integrated speed measurement. But the most important issue is to have a soft switching between different methods which is mostly neglected in previous works. The phrase of "soft switching" means that the measured speed must change continuously during switches from one method to another one. In mathematical words, soft switching happens if and only if:

$$\omega_{\max-method1} = \omega_{\min-method2} \tag{19}$$

An integrated method without soft switching must be compensated with estimating or filtering algorithms undeniably and this raises the complexity of system. In the following we prove that our proposed method has a soft switching. At the critical speed, from equation (10), (17) and (19) we have to prove that:

$$\frac{N_0.90}{N_E.(T_S - \Delta_{TS})} = \frac{f_{hf}.90}{N_E.n} \qquad (20)$$

In equation (20), left and right hand sides are referred to speeds which measured by proposed algorithm and period measurement method respectively. At the critical speed, we have to show that these two sides are equal in order to show our integrated method has a soft switching. As it shown in Fig. 10, in critical speed Δ_{Ts}= 0 and N_0 = 1. So, (20) is simplified to:

$$\frac{1}{T_S} = \frac{f_{hf}}{n} \qquad (21)$$

Equation (21) is truly based on equation (16). So our first assumption has been proved which claims to establish an integrated method with soft switching. It means that when rotating speed fluctuates around critical speed, and measurement method is changed from one method to another one, the measured speed doesn't suffer from method changing and it is calculated continuously. The presented method is a flexible method and establishes a design tradeoff based on four main elements: T_S, T_{Smax}, ω_{min} and $\omega_{critical}$. Fig. 12 shows the characteristic curve of proposed method. Equations (18) and (15) are used to trace this curve. Also, the minimum required value of N_E for desired design can be determined using them. Now, the first numerical example discusses again and at this time it is solved using proposed method. In exampled system T_S = 10ms and ω_{min}= 0.01 d/s. If we suppose that it is acceptable to raise T_S to 50ms in speeds lower than 0.1 d/s, we can write:

$\omega_{critical}$ = 0.1 [d.s^{-1}]
T_{Smax} = 50 ms

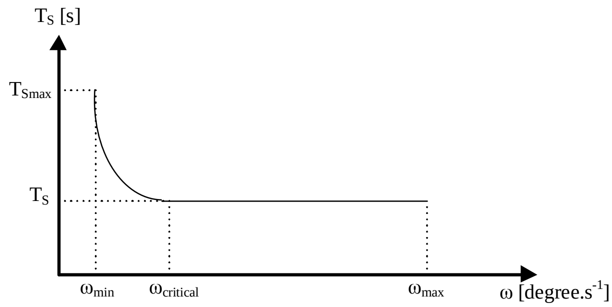

Fig.12: Characteristic curve for proposed method

So from equation (18) we have:

$$50 \times 10^{-3} = \frac{90}{0.01 \times N_E} \Rightarrow N_E = 180000$$

And from equation (15):

$$0.1 = \frac{90}{10 \times 10^{-3} \times N_E} \Rightarrow N_E = 90000$$

The minimum required value for N_E which satisfies all constraints of design is 180000. Again using (15) we have:

$$\omega_{critical} = \frac{90}{10 \times 10^{-3} \times 180000} = 0.05[\deg ree.s^{-1}]$$

So the characteristic curve of design will be as Fig. 13. It is obvious that the proposed method reduces the minimum required value of N_E from 360000000 to 180000. (0.05% improvement). This is the most advantage of this method which eliminates the quantization error and makes it possible to reach to high accuracy and proper controller bandwidth while using low resolution encoders.

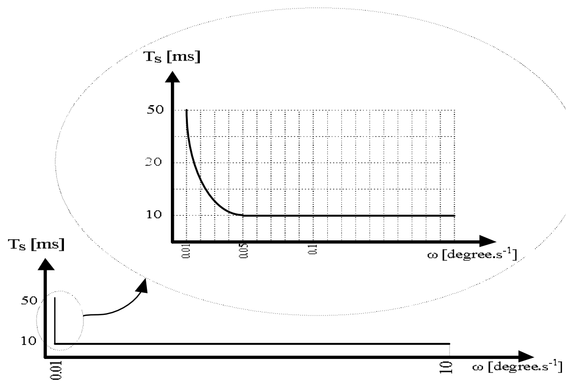

Fig. 13: Characteristic curve for an example design

VII. Simulations

In order to evaluate a comparison between the proposed method and previous works, a Matlab/Simulinx program is developed. In this program, an incremental encoder with 100000 pulses per round is simulated and T_S assumed to be 100 µs while speed changes in the ranges of 0.5 d.s^{-1} to 260 d.s^{-1}. In this simulation, proposed method is compared with two custom methods:

Method 1: pulse counting during constant T_S = 100 µs.

Method 2: measuring time between two successive pulses.

In all simulations high frequency clock assumed to be 50 MHz. Fig.14 and Fig.15 show the simulation results while encoder rotates in low and high speed margins respectively. In these simulations the Ramp Function uses as an input speed. As it can be seen method 1 is totally ineffectual in low speeds, but method 2 has good results in this region. The situation is different in high speeds which the results of method 2 are far from real speeds because in this region, the high frequency quantization error is non negligible using method 2. It is obvious that the proposed method has accurate results in both regions. Fig. 16 shows the transient response to the sinusoidal speed. This figure shows how method 1 suffers from quantization error which totally eliminated in proposed method. As this figure shows, the proposed method gives accurate results across the entire region for transient conditions.

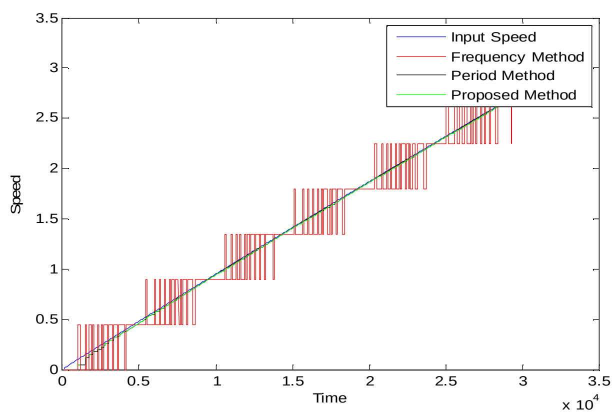

Fig. 14: Simulation results in low speed region

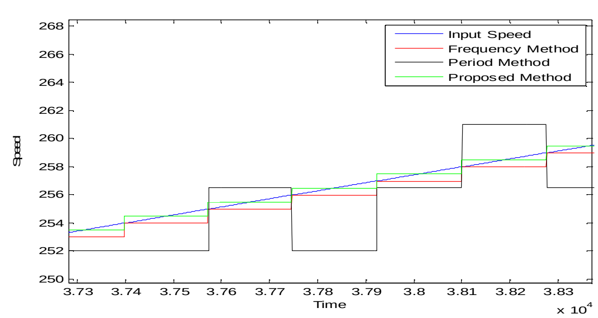

Fig. 15: Simulation results in high speed region

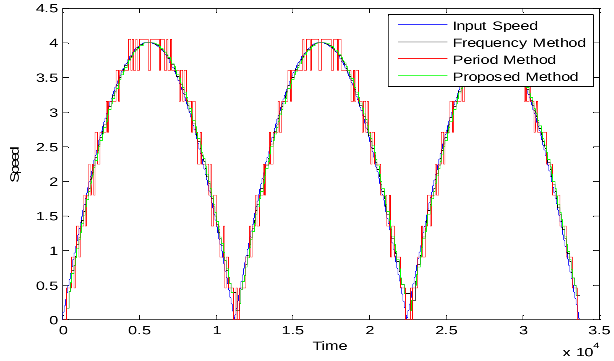

Fig. 16: The transient response to the sinusoidal speed

VIII. Experimental results

Fig.17 shows a NSK direct drive servo system with high resolution encoder (1200000 Pulses per round) and FPGA based controller which are assembled to test presented method. Thus, the proposed method has been implemented in a Xilinx Spartan 3E FPGA. Also, In order to test proposed algorithm, a low resolution output Encoder with 3600 pulses per round has been coupled to the rotor. The source code is written in Verilog language and a TCP/IP Ethernet protocol is used aimed at transferring real time results to PC. In this test, T_S and T_{delay} are assumed to be 10ms and 100ns respectively. Also the high frequency clock is 50MHz. In order to assess the precision of proposed method, the speed which has been measured with high resolution encoder is assumed to be a real speed and the proposed algorithm is implemented to measure the speed using low resolution encoder.

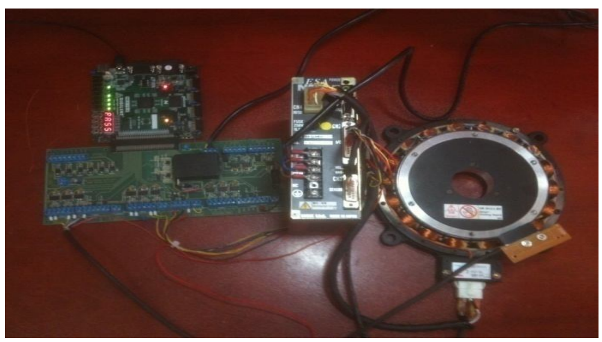

Fig. 17: Case study, NSK direct drive servo system and FPGA based controller

The steady and transient states results for different range of speeds are depicted in Figure 18-19. As it can be seen, proposed method benefits from stability and accuracy during different ranges of speeds. The experimental results validate that quantization error has been eliminated using proposed method.

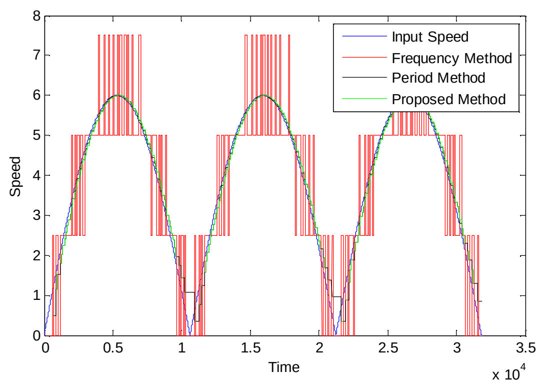

Fig. 18: Experimental results in transient states

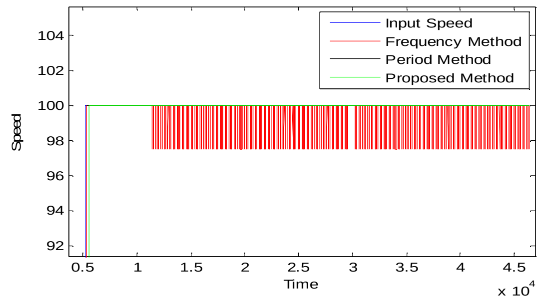

Fig. 19: Experimental results in steady states

IX. CONCLUSION

In this study, a novel implementable speed measurement method has been established based on quadrature signals of incremental encoder. First, the source of quantization error has been obtained and then an algorithm proposed in order to eliminate this error. Mathematically we proved that our method doesn't suffer from counting discrete pulses and has an ability to measure the speed continuously. The simulation and experimental results also validate this fact. In order to measure speed in low speed regions, an alternative algorithm has been presented and also an integrated method proposed which switches between two different algorithms in low and high speed regions. Finally, the results show that the proposed method has a precise measurement in both low and high speeds and reaches to more stable responses than other methods while dealing with low level calculations.

REFERENCES

[1] N. Johnson, *et al.*, "Optimization of incremental optical encoder pulse processing," in *Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 2013 International Multi-Conference on*, 2013, pp. 769-773.

[2] A. C. Negrea, *et al.*, "Error compensation methods in speed identification using incremental encoder," in *Electrical and Power Engineering (EPE), 2012 International Conference and Exposition on*, 2012, pp. 441-445.

[3]A. Abdagic, *et al.*, "Reduction of low-resolution incremental encoder usage effects on control performance by using a fuzzy controller," in *Information, Communication and Automation Technologies (ICAT), 2011 XXIII International Symposium on*, 2011, pp. 1-6.

[4] C. A. Negrea, *et al.*, "An improved speed identification method using incremental encoder in electric drives," in *Automation Quality and Testing Robotics (AQTR), 2012 IEEE International Conference on*, 2012, pp. 536-540.

[5]R. Petrella and M. Tursini, "An Embedded System for Position and Speed Measurement Adopting Incremental Encoders," *Industry Applications, IEEE Transactions on*, vol. 44, pp. 1436-1444, 2008.

[6] J. K. Mohan and N. Johnson, "Devising Simulink Optical Encoder Pulse Manipulation and its Evaluation," in *Intelligent Systems Design and Applications (ISDA), 2012 12th International Conference on*, 2012, pp. 640-644.

[7]R. C. Kavanagh, "Shaft encoder characterisation through analysis of the mean-squared errors in nonideal quantised systems," *Science, Measurement and Technology, IEE Proceedings -*, vol. 149, pp. 99-104, 2002.

[8]H. W. Chow and N. Cheung, "High speed processing of encoder information by using a dual-resolution approach," in *Industrial Electronics and Applications (ICIEA), 2010 the 5th IEEE Conference on*, 2010, pp. 644-649.

[9]H. ARABACI and O. B˙ILG˙IN, "A novel motor speed calculation method using square wave speed sensor signals via fast Fourier transform," *TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES*, vol. 20, 1.1.2012.

[10]R. J. E. Merry, *et al.*, "Velocity and acceleration estimation for optical incremental encoders," *Mechatronics*, vol. 20, pp. 20-26, 2010.

[11]*Technical Information:Rotary Encoder* Available: http://www.citizen-micro.com/tec/encoder.html