# Speed Estimation Algorithm with Specified Bandwidth for Incremental Position Encoder

Alecksey Anuchin*, Anton Dianov†, Dmitry Shpak*, Valentina Astakhova* and Ksenia Fedorova*

* Moscow Power Engineering Institute, Moscow, Russian Federation, e-mail: *anuchin.alecksey@gmail.com*

† Samsung Electronics, Digital Appliances division, Suwon, South Korea, e-mail: *anton.dianov@samsung.com*

*Abstract*—**This paper proposes speed estimation method for the incremental position encoders. This method uses variable angular path to keep the bandwidth and accuracy in the specified range regardless of the current speed. This paper also describes existing speed estimation methods, their strong and weak points. Moreover, non-idealities of the incremental encoders are discussed and compensation algorithm is proposed. Proposed speed estimation algorithm uses period-based calculation method with the variable angular path. The experimental results obtained in the flux-vector control system of a Permanent Magnet Synchronous Motor (PMSM) prove feasibility and show perfect performance of the algorithm even in the low speed range at startup.**

***Keywords-angular velocity; incremental encoder; speed estimation; quadrature encoder pulse module; motion control.***

## I. INTRODUCTION

Incremental position encoders are widely used in the modern electric drives as a position and speed sensors. Modern microcontrollers typically contain special peripheral devices named Quadrature Encoder Pulse (QEP) module, which convert the encoder output sequences “A” and “B” into direction and clock. These signals are used to increment or decrement a counter, which is used to obtain the digital code of the rotor angular position.

The speed estimation can be done in two different ways [1-5, 9]: by evaluation of the position derivative with respect to time, or by using observers like phase-locked loop (PLL) or Kalman based filters [5-7]. The second approach is not suitable for the high dynamic drives due to filtering time delays, but it can be used with the derivative calculation methods to improve the accuracy by making prediction of speed estimation between the instants of time when the derivative is calculated.

There are also two methods for derivative calculation described in many papers. They are period- and frequency-based methods.

The frequency-based method counts number of pulses received from the encoder during fixed period of time. The bandwidth of the method is defined by this fixed period; and its accuracy depends on the motor speed. The higher motor speed, the bigger number of pulses is detected during estimation time and the smaller speed estimation error is. So this method is not suitable for low speeds and gives relatively good accuracy only in high speed region.

The period-based method measures the time of a period of “A” and/or “B” pulses from the position encoder. The period of time is measured via CPU timer of the microcontroller, therefore the bigger is the speed, the smaller is the period of time, the smaller is the CPU timer counter increment, and the poorer calculation accuracy is. This method gives better accuracy on low speeds, but if the speed is too small, the bandwidth is reduced significantly, because the duration of the A signal period can be unpredictable long.

In many papers authors combines these two methods [3, 5, 10]. The main goal of their research is to make smooth switching between two different speed estimation systems, when one method becomes inefficient while accuracy of another method increases.

In this paper the period-based method is taken as the basic method, and it is significantly improved. The estimation is done not for a single cycle of “A” or/and “B” sequence, but for a number of signal cycles. If the drive accelerates and the time between two timestamps reaches the predefined minimum value, where accuracy significantly decreases, the number of counting cycles for the position counter doubles. The period between timestamps and the accuracy increases two times, though the bandwidth is decreased. If the drive slows down, the period may reach its maximum value, where the number of counting cycles for the position counter is decreases two times. The similar method was considered in [8], but most of the modern microcontrollers do not allow to change setting of the position signal frequency divider in real-time.

This algorithm modifies the angle increment two times every time to provide the fixed range bandwidth of the speed estimator. It changes the speed estimation range dynamically according to the drive speed. Thus, there is no need to implement the frequency-based method at all, because the switching of the speed range provides better and predictable accuracy regardless of the speed. The operation in the low speed region is also considered.

## II. PRINCIPLES OF INCREMENTAL ENCODER OPERATION AND SPEED ESTIMATION

### A. *Operation of the Incremental Encoder*

Incremental encoder produces a sequence of two signals with phase shift dependent on the direction of the rotation. In the simplest case this signals can be produced by a system of

The research was performed with the support of the Russian Science Foundation grant (Project № 16-19-10618).

light emitting diode, a metal or glass disc with the slits, and two photodiodes. The light passes through the slits onto photodiodes, which are phase shifted at 90° corresponding to a single slit. The signals from these two photodiodes – “A” and “B” – carry information about current rotor angular position. If position changes the signals produce a number of pulses proportional to a position deviation. The main parameter of any incremental encoder is its resolution in pulses per revolution — $K$. The direction of the shaft angle change can be obtained from the phase shift of these two signals.

Some encoders have index channel “Z”, which provides one pulse per revolution for homing and verification of “A” and/or “B” channels. But in this particular case this channel is not used because it is necessary only for positioning.

These two signals come to the inputs of the microcontroller, which are connected with the QEP peripheral device internally. QEP module of the microcontroller is used to evaluate position by means of its hardware. It also provides all necessary data to make speed estimation by means of the software.

The signals “A” and “B” passes to the QEP decoder logic, which analyzes the upcoming edges and state of the signals and produces “QDIR” (direction) and “QCLK” (clock) signals for a position counter. Position counter is the *QPOSCNT* register in the peripheral memory window, which stores the current rotor position as shown in the Fig. 1. Its value can be limited according to the number of pulses of the position encoder. The position counter counts every edge of the signals “A” and “B”, therefore it increments by:

$$N = 4K \tag{1}$$

per one revolution. As it has been mentioned, there are two possible ways of speed estimation – period and frequency-based method, which are described below.

### B. Frequency-based Method

The frequency-based method uses the constant period of time to count pulses from the sensor. The speed can be estimated by the following equation:

$$\omega = \frac{2\pi}{T} \cdot \frac{n}{N}, \tag{2}$$

where $T$ is the duration of the measurement, $N$ is the number of clock pulses per revolution, $n$ is the number of counted pulses from the position encoder during the specified duration.

This method has two main disadvantages. The higher the bandwidth of the speed estimation, the smaller $T$; the smaller number of counted pulses, the bigger error of the speed estimation is. The lower the motor speed, the smaller the number of counted pulses, the bigger the error of the speed estimation is. The maximum absolute error of the speed estimation is:

$$\Delta\omega = \frac{2\pi}{T} \cdot \frac{1}{N}. \tag{3}$$

For example, for the encoder with $N = 40000$ and the bandwidth of 4 kHz for a speed loop, the error is:

$$\Delta\omega = \frac{2\pi}{1/4000} \cdot \frac{1}{40000} = 0.628\,\frac{\text{rad}}{\text{s}}, \tag{4}$$

or

$$\Delta n = 6 \text{ rpm}. \tag{5}$$

An example of the situation with the low accuracy for this method is shown in the Fig. 2. The speed is constant, and $T$ is also constant. During this period of time number of countered pulses can be 6 or 7, thus the accuracies in these cases differ approximately at 17%.

### C. Period-based Method

The period-based method measures period of the time, which is taken for the specified angle modification (increment or decrement). If the angle increment is equal to a single period of the “A” or “B” signal, then the speed can be evaluated by the following equation:

$$\omega = \frac{2\pi}{\Delta T} \cdot \frac{4}{N}, \tag{6}$$

where $\Delta T$ is the time difference for the specified angle increment. In its turn, with the growth of the speed the time difference decreases as it is shown in the right part of Fig. 3. Since the time difference is measured with CPU cycles, the resolution for 150 MHz CPU is 6.67 ns. For a spindle drive at 12 000 rpm the encoder output frequency reaches 2 MHz and the relative error is:

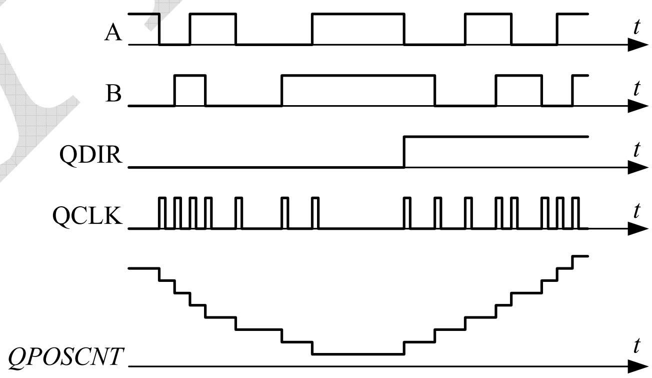


Figure 1. Signals and their processing by the QEP peripheral module.

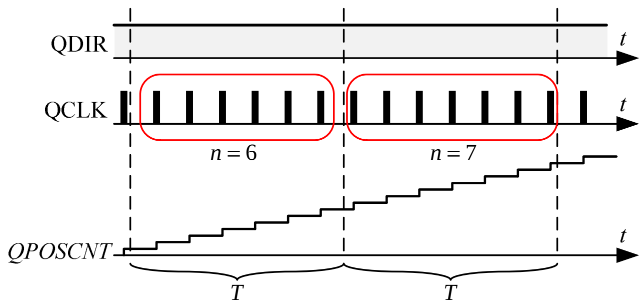


Figure 2. Frequency-based method diagram.

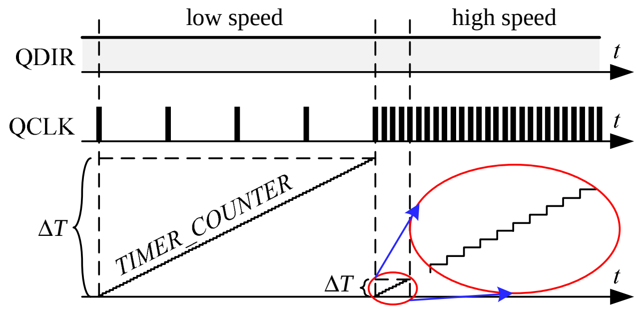


Figure 3. Period-based method diagram.

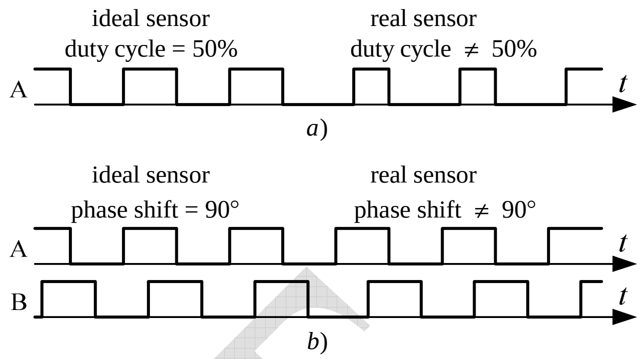


Figure 4. Incremental encoder non-idealities.

$$\delta\omega = \frac{f_{sensor}}{f_{CPU}} = \frac{2\,000\,000}{150\,000\,000} = 1.3\%. \tag{7}$$

These examples are given for motor drive operating in the wide speed range, and this problem may be not so important for the middle speed range drives, however the purpose of this paper is developing of the universal solution.

## III. Variable Angular Path Method

### A. *Incremental Encoder Non-idealities*

There are two main encoder non-idealities that should be taken into account, especially for the period-based method of the speed measurement. The first problem happens, when signal line comparator inside the encoder has incorrect reference value and duty cycle of the signal “A” or/and “B” is not equal to 50%, as it is shown in Fig. 4a. The second problem arises when the phase shift between “A” and “B” signal is not equal to 90° as shown in the Fig. 4b. This error in some encoders reaches 15° and more and can deviates over the revolution. Therefore, in this case the measurable angle path should be multiple of four and its minimum value must be four, otherwise speed estimation error can be unacceptable high.

### B. *Combined Method and its Accuracy*

The accuracy of the period-based method depends on the angular path for the single speed estimation, the encoder resolution and the CPU frequency. All these parameters have an influence on the bandwidth of the speed estimator. The higher the encoder resolution, the higher the bandwidth is. The higher the CPU frequency, the higher the accuracy of the time measurement is. The higher the bandwidth, the higher the minimum measured speed could be.

The accuracy of the time measurement for the specified bandwidth is:

$$\delta t = \frac{f_{bw}}{f_{CPU}}, \tag{8}$$

where $f_{bw}$ is the bandwidth. For the various combinations of these parameters the accuracy is given in Table I.

TABLE I. Accuracy vs Bandwidth and CPU Frequency

| $f_{bw}$, Hz | Accuracy, % | | |
|---|---|---|---|
| | ***TMS320LF240xA 40 MHz*** | ***TMS320F280xx Piccolo 60 MHz*** | ***TMS320F2833x Delfino 150 MHz*** |
| 1000 | 0.0025 | 0.001667 | 0.000667 |
| 2000 | 0.005 | 0.003333 | 0.001333 |
| 4000 | 0.01 | 0.006667 | 0.002667 |
| 8000 | 0.02 | 0.013333 | 0.005333 |

The minimal measured speed for the specified bandwidth is defined by the following equation:

$$\omega_{min} = \frac{2\pi}{N} 4 f_{bw}, \tag{9}$$

where 4 is the minimal angular path in counting pulses. For the various encoder resolutions (pulses per revolution – ppr) and the specified bandwidth the minimal speed is given in Table II. It can be seen that the resolution of the encoder should be selected according to the desired bandwidth of the speed estimator and the minimum speed of the drive. Actually, it is still possible to measure speeds smaller than given in Table II, but the bandwidth of the speed estimation will be decreased in that case. Therefore, the feedback for the speed loop will be delayed, which may result in the instability of the speed control loop.

TABLE II. Minimal Estimated Speed vs Bandwidth and Encoder Resolution

| $f_{bw}$, Hz | $\omega_{min}$, rad/s | | | | | | |
|---|---|---|---|---|---|---|---|
| | ***1000 ppr*** | ***3000 ppr*** | ***10000 ppr*** | ***30000 ppr*** | ***100000 ppr*** | ***300000 ppr*** | ***1000000 ppr*** |
| 1000 | 6.28 | 2.09 | 0.628 | 0.209 | 0.0628 | 0.0209 | 0.0063 |
| 2000 | 12.56 | 4.18 | 1.256 | 0.418 | 0.1256 | 0.0418 | 0.0125 |
| 4000 | 25.13 | 8.37 | 2.513 | 0.837 | 0.2513 | 0.0837 | 0.0251 |
| 8000 | 50.26 | 16.75 | 5.026 | 1.675 | 0.5026 | 0.1675 | 0.0503 |

On other hand, the increase of the encoder resolution results in growth of the encoder signals frequency. Microcontroller input signal frequency is typically limited with one quarter of the CPU frequency, otherwise peripheral device is not able to detect input pulses correctly. Another problem of the high frequencies is that the accuracy of the time period measurement is decreased. The higher the input frequency, the smaller time period for the speed estimation is. For instance, if the sensor signals frequency reaches 2 MHz, accuracy of the time measurement and, respectively, speed estimation is as shown in (7).

To reduce the speed estimation error at high speed many authors suggest to switch to the frequency-based method and to count the number of encoder pulses during specified period of time. This switching between two methods should be done during operation at high speed. Figure 5 shows the dependence between accuracies of the methods and the motor speed for the desired bandwidth of 8000 Hz, an encoder with 10,000 pulses per revolution and a controller with 60 MHz CPU frequency. The optimal switching point is crossing of the error curves, which approximately corresponds to 215 rad/s. The accuracy in this point is the same for both methods and equals to 0.58%.

If desired accuracy is higher, the angular path of 8 or more counting clock pulses should be used. But in this case the bandwidth of the speed estimation will be reduced. The critical error can be evaluated by the following equation [10]:

$$\delta\omega = \sqrt{\frac{f_{bw}}{4 f_{CPU}}}, \tag{10}$$

and it depends on CPU frequency and desired bandwidth. Therefore, if the bandwidth is high and the CPU frequency is small it may not be possible to keep the sufficient accuracy for the whole speed range. Thus, the different approach should be used.

As it was mentioned, the change of the specified angular position path for the period-based method affects on the bandwidth but also this shifts the error vs speed curve down as shown in Fig. 6. Each curve with a longer path is more precise but the bandwidth reduces.

The maximum bandwidth is can be obtained for a curve for 4 position clock increment. It provides the maximum bandwidth on slow speeds and when the speed reaches $\omega_{min}$ from Table II speed estimator also reaches the specified speed evaluation frequency. With continue of the speed growth the rotor passes the angular path more and more quickly and the frequency increases. At the speed two times higher than the minimum one ($2\omega_{min}$) the bandwidth will be also two times higher than it is specified, though there is no need to keep it on this level. Thus, it is possible to change the angular path to make it two times higher than the initial one. The change of the angular path should be done every time the speed doubles. So, the bandwidth will remains unchanged regardless of the speed deviation and the accuracy will stay unchanged too.

Figure 7 shows the dependence of the time difference and the speed on various angular paths. The angular path at zero speed and low speed region is equal to 4 counting pulses. It corresponds to the speed range, which is equal to zero. With the growth of the speed the measuring time reaches the minimum value and the speed range increments. If the speed decreases, then the speed range decrements as the time difference reaches its maximum value.

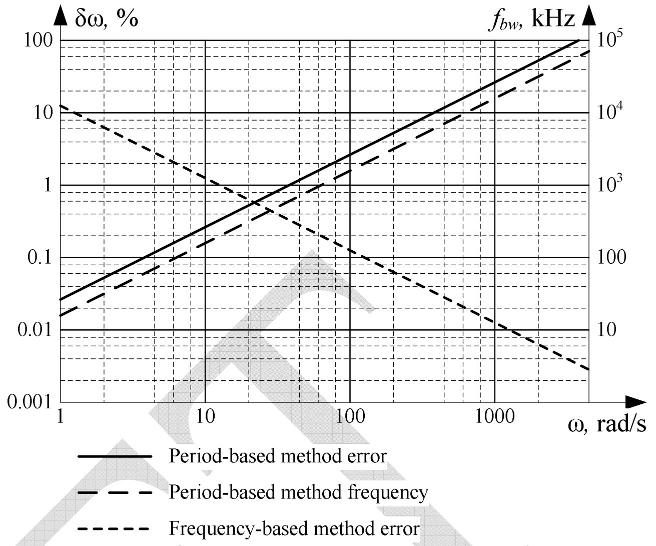


Figure 5. Accuracy and bandwidth for speed estimation methods vs motor speed: Green line – error for the frequency-based method; blue line – error for the period-based method; brown line – speed estimation frequency or the bandwidth frequency $f_{bw}$.

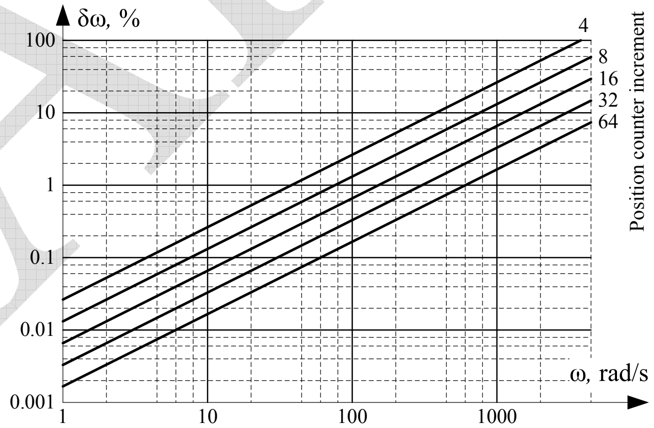


Figure 6. Accuracy of the period-based method vs speed for different angular position paths.

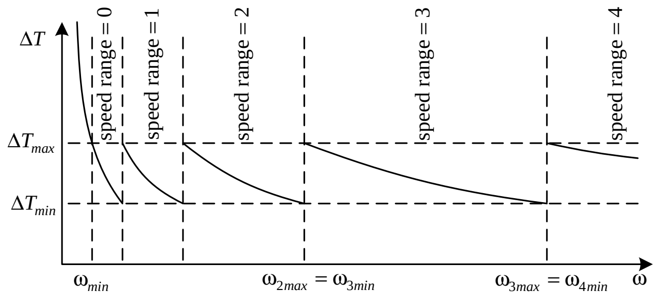


Figure 7. Time difference vs speed for various speed ranges.

The maximum time difference value can be evaluated by the following equation:

$$\Delta T_{max} = \frac{1}{f_{bw}}. \qquad (11)$$

The angle path depends on the current speed range and can be expressed as:

$$\Delta\theta = \frac{2\pi}{N} 4 \cdot 2^r, \qquad (12)$$

where $r$ is the speed range. Hence, the speed estimation can be done by:

$$\omega = \frac{\Delta\theta}{\Delta T} = \frac{8\pi}{N}\frac{1}{\Delta T} 2^r. \qquad (13)$$

Equation (13) is divided into 3 parts. First part $\frac{8\pi}{N}$ is the encoder constant and depends on its resolution. Second part $\frac{1}{\Delta T}$ is the inverse value of the time difference. The formats of these two values should be chosen to obtain the right answer without overflow for any possible change of the time difference. The last part $2^r$ is the arithmetic left-shift of the result according to the current speed range value.

## IV. Experimental Results

The proposed method has been implemented in various electric drives [11, 12]. This paper shows results obtained in the Texas Instruments DRV8302-HC-C2-KIT, which contains 3-phase BLDC & PMSM motor kit with DRV8302 and Piccolo MCU. Microcontroller operates at 60 MHz. A permanent magnet synchronous motor is connected to the inverter and coupled with the position encoder HEDM-5500#B06 with 1000 pulses per revolution as shown in Fig. 8.

The main parameters of the speed estimator are shown in Table III.

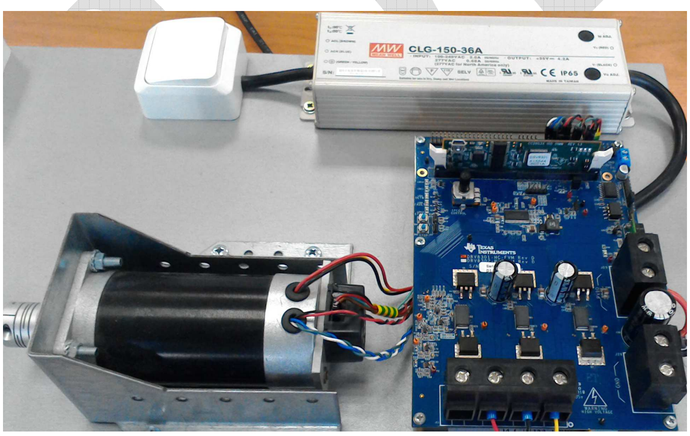


Figure 8. Motor kit with the PMSM and 1000 ppr incremental encoder.

TABLE III. Speed Estimator Parameters

| Parameter | Value | Units |
|---|---|---|
| $f_{CPU}$ | 60 | MHz |
| $f_{bw}$ | 2 | kHz |
| $\Delta T_{max}$ | 30 000 | ticks of CPU timer |
| $\Delta T_{min}$ | 15 000 | ticks of CPU timer |
| $K$ | 1000 | ppr |
| $\omega_{min}$ | 12.6 | rad/s |

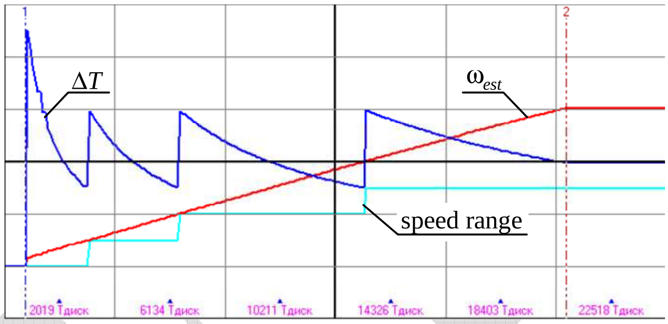


Figure 9. Transient of the speed estimator variables during motor startup ($\omega_{est}$ estimated speed — 50 rad/s per division, $\Delta T$ time difference — 10 000 CPU cycles per division, $r$ speed range — 2 per division, time — 42 ms per division).

The motor starts with the rated torque in a close-loop flux-vector control system. Speed reference is set to 150 rad/s. The transients of the speed estimator variables are shown in the Fig. 9. At startup the estimated speed is zero. It becomes observable when the motor reaches 8 rad/s. It is smaller than the minimal speed from Table III, and the bandwidth for this speed is smaller than it is specified, therefore the $\Delta T$ value is also higher than $\Delta T_{max}$. Actually, zero speed limit can be set to any value. The only problem relates to smaller bandwidth, and necessity to check the overflow. For instance, if $\Delta T$ is represented with unsigned 16-bit number, then its value should never exceed 65535.

It can be seen that with the growth of the speed time difference changes inversely, and every time it reaches the value of 15 000 CPU cycles, the speed range increases and the time difference doubles. Therefore, the accuracy and the bandwidth of the speed estimator remain unchanged.

## V. Conclusions

The proposed variable angular path method is applicable for any incremental encoder and provides approximately constant accuracy according to the specified bandwidth in the whole speed range. This method was implemented for various microcontrollers from Texas Instruments.

The future investigations will be connected with the low speed operation, where this approach will be merged with PLL-based method.

## References

[1] A.F. Ilmiawan, et al., “An easy speed measurement for incremental rotary encoder using multi stage moving average method”, In Proc. of International Conference on Electrical Engineering and Computer Science (ICEECS), pp. 363–368, 2014.

[2] Shamim Choudhury. Motor Speed Measurement Consideration When Using TMS320C24x DSPs. Application report SPRA771. Texas Instruments, August 2001.

[3] David Alter. Using the Capture Units for Low Speed Velocity Estimation on a TMS320C240. Application brief: SPRA363. Texas Instruments, July 1997.

[4] F. Briz; J. A. Cancelas; A. Diez , “Speed measurement using rotary encoders for high performance AC drives”, In Proc. of Industrial Electronics, Control and Instrumentation Conference, vol.1, pp. 538–542, 1994.

[5] R. Petrella, et al., “Speed measurement algorithms for low-resolution incremental encoder equipped drives: a comparative analysis”, In Proc. of International Conference on Electrical Machines and Power Electronics (ACEMP), pp. 780–787, 2007.

[6] S. C. Venema, “A Kalman Filter Calibration Method For Analog Quadrature Position Encoders”, Master degree thesis, University of Washington, 1994, p. 99.

[7] G. Liu, “On velocity estimation using position measurements”, In Proc. of American Control Conference, vol.2, pp. 1115–1120, 2002.

[8] N. Johnson, et al., “Optimization of incremental optical encoder pulse processing”, In Proc. of International Multi-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), pp. 769–773, 2013.

[9] C. Alin Negrea, et al., “Error compensation methods in speed identification using incremental encoder”, In Proc. of International Conference and Exposition on Electrical and Power Engineering (EPE), pp. 441–445, 2012.

[10] C. Alin Negrea, et al., “An improved speed identification method using incremental encoder in electric drives”, In Proc. of IEEE International Conference on Automation Quality and Testing Robotics (AQTR), 2012, pp. 536–540, 2012.

[11] A. Anuchin, Y. Vagapov, “Configurable Control Systems of Power Converters for Instructional Laboratories”, 6th International conference on Internet Technologies & Application, Wrexham, Glyndwr University, 8-11 September 2015.

[12] A. Anuchin, C. Belloc, Y. Vagapov, “Development of Curriculum for a Postgraduate Course on Electric Drives and Motion Control”, 50th Universities Power Engineering Conference (UPEC 2015), Staffordshire University, 1-4 September 2015.