Estimation and Control of DC Motor Speed Using an Encoder with a Model and Kalman Filter-based Method

Hung D. Nguyen*

**Australian Maritime College/University of Tasmania, Launceston, Tas 7250 Australia (Tel: 03-6324-9350; e-mail: nguyenhd@amc.edu.au)*

Abstract: This article presents a method to estimate and control the speed of a dc motor using an encoder with a mathematical model and Kalman filter. An encoder attached to the shaft of a dc motor provides an angular displacement signal in form of pulse counts. For control of the dc motor speed the speed measurement should be accurate and precise. The article is about issues of the motor speed estimation from the encoder pulse counts due to the resetting of pulse counts in the encoder and solutions to overcome these issues. The proposed method is able to solve the issues.

Tóm tắt: Bài báo này trình bày phương pháp ước lượng và điều khiển tốc độ động cơ một chiều dùng encoder bằng mô hình toán và bộ lọc Kalman. Encoder kết nối với trục của một động cơ cung cấp tín hiệu đo góc dưới dạng số đếm xung. Để điều khiển tốc độ động cơ thì số đo tốc độ cần phải chuẩn xác và chính xác. Bài bào này viết về các vấn đề ước lượng vận tốc động cơ từ số đếm xung do sự đặt lại số đếm xung và các giải pháp khắc phục các vấn đề này. Phương pháp đề xuất có thể khắc phục được các vấn đề ước lượng vận tốc.

Keywords: encoder, dc motor speed control, speed estimation, model-based method and Kalman filter.

1. INTRODUCTION

One of the common methods to measure velocity of a dc motor is to use a digital encoder attached to its shaft. A digital encoder provides angular displacement in the form of pulse counts. The pulse counts contain information of angular displacement of the rotor shaft. If the pulse counts per time instant are obtained, then the motor velocity can be estimated.

The main purpose of this article is to:

- state issues of speed estimation from an encoder's pulse counts;
- provide solutions to overcome the issues;
- outline the design of a speed controller; and
- develop computer simulation programs for control engineering education.

The control programs are made with LabVIEW of National Instruments and can be used for control engineering education purposes. The multifunction data acquisition kit and control programs are used to demonstrate a brush dc motor, digital encoder, data acquisition card, and LabVIEW control program based on the PID control law.

The article is organized as follows: Section 1 Introduction is about motivation, scope and objectives of the work, Section 2 Brief description of experiment equipment, Section 3 Estimation of dc motor speed and issues, Section 4 Solutions to overcome issues, Section 5 Experiments and development of simulation and real-world control programs for control engineering education and training purposes, and Section 6 highlights some conclusions and recommendation for future work.

2. BRIEF DESCRIOPTION OF EQUIPMENT

2.1 Inexpensive Multifunctional Kit HDL 9001

The experimental equipment used for this work is a "Multifunction kit HDL 9001" that was developed by a group of young engineers and scientists at www.hocdelam.org. The kit consists of an DAQ (I/O) interface card with an AVR microcontroller (ATMEGA8), dc motor, 400 ppr encoder, dc power supply and motor drive as shown in Fig. 1. The dc motor is controlled by a PWM signal through an LM18200-based H-bridge motor drive and the speed is measured by a 400-ppr encoder.

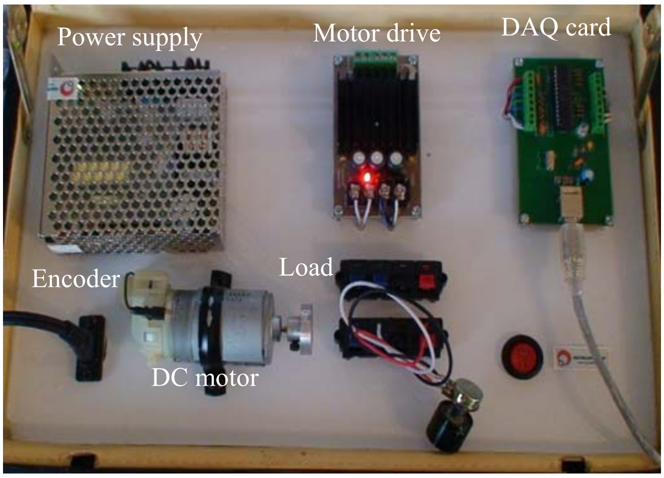

Fig. 1. Picture of the multifunctional kit HDL-9001

2.2 Control and Simulation Tools – National Instruments LabVIEW

The multifunctional kit HDL 9001 is interfaced with LabVIEW via device driver using a USB-serial adapter. The functional block diagram of the experimental equipment is shown in Fig. 2.

The motor speed estimation and control programs were made with LabVIEW software. Fig. 3 shows a snapshot of the LabVIEW speed estimation and control program and user interface (Front Panel).

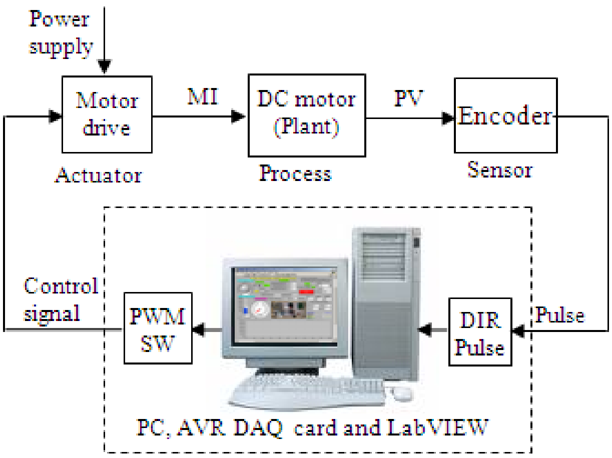

Fig. 2. Schematic diagram of the motor control system

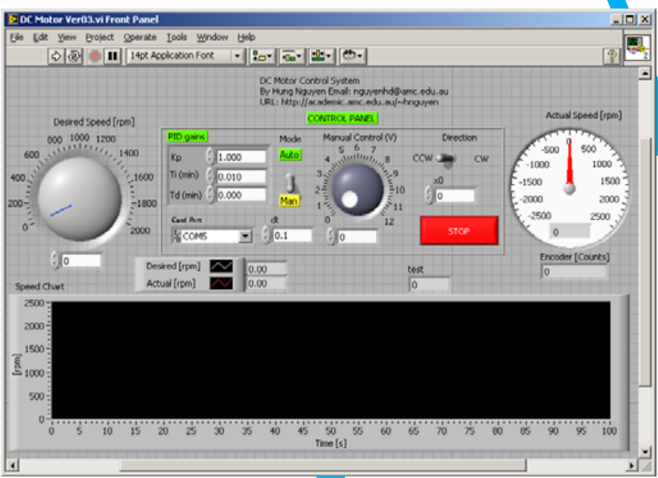

Fig. 3. Snapshot of the dc motor control system

3. ISSUES OF SPEED ESTIMATION

3.1 Speed Estimation Algorithm

The speed, ω_m [rpm], of the motor is estimated by (National Instruments, 2011a, b)

$$\omega_m = \frac{x_1 - x_0}{ppr \times \Delta t} \times \frac{60}{1} \text{ [rpm]} \tag{1}$$

where

x_0: old count (stored in memory by a shift register)
x_1: current count (loaded from the encoder)
ppr: pulse per revolution (= 400 for the encoder used in experiments)
$\Delta t = t_2 - t_1$: counting interval

The speed estimation algorithm is summarized by the flowchart as shown in Fig. 4.

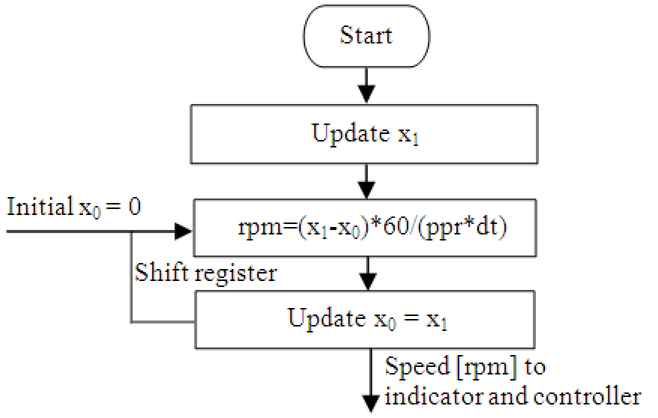

Fig. 4. Flowchart of the speed estimation algorithm

The speed estimation algorithm is executed with Formula Node in the LabVIEW program as shown in Fig. 5. In order to obtain the number of pulse counts per time interval a Shift Register is used. The time interval is taken by using the function "Wait until Next ms Multiple as shown in Fig. 6.

Fig. 5. Speed estimation algorithm using Formula Node

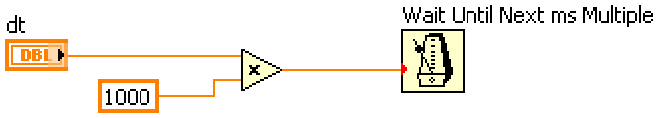

Fig. 6. Time interval by using a Wait Until Next ms Multiple

3.2 Issues of Speed Estimation

When the above-mentioned speed estimation algorithm is executed, three issues occurred to the estimated speed values.

The first issue is that the value of estimated speed suddenly changes to an extremely large value of opposite sign (called "a jumping value") when the encoder count is reset to zero. The 400-ppr encoder has a maximum number of counts at 2^{16}–1 (65535), when $x_0 = 2^{16}$–1 or –(2^{16}–1) the number of counts per time instant x_1–x_0 has an extremely large value. The jumping value of estimated speed causes the control system to be impossible to control the motor speed. The second issue is the jumping value with the same sign when the LabVIEW program starts. This also causes the control algorithm impossible to control. Two issues are shown in Fig. 7a and 7b.

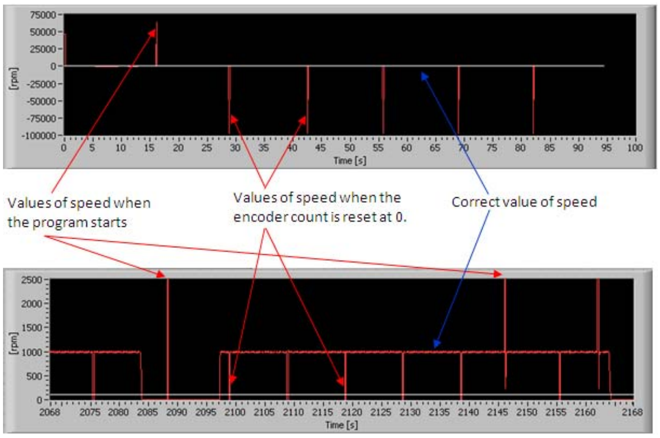

Fig. 7a Estimation of speed using the encoder pulse counts and "jumps" of the speed value when the LabVIEW program starts and the encoder count is reset at zero

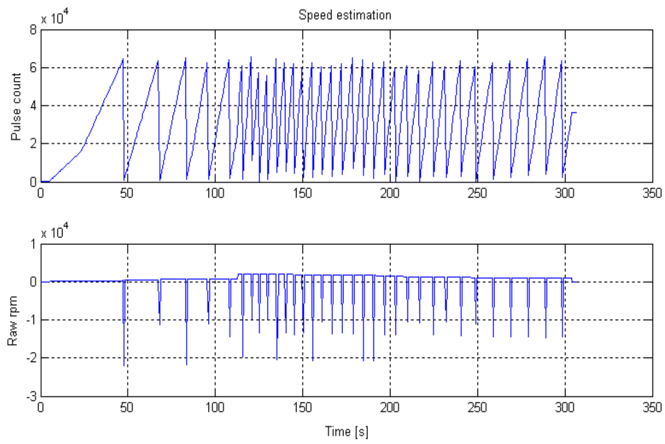

Fig. 7b Pulse count (top) and estimated speed (bottom)

The third issue is the variation of estimated speed value when the motor runs at high speeds. The estimated speed looks like a noisy signal. By the experiments it is seen that the variation of the estimated speed depends on time interval (dt) and speed of the motor as shown in Fig. 8.

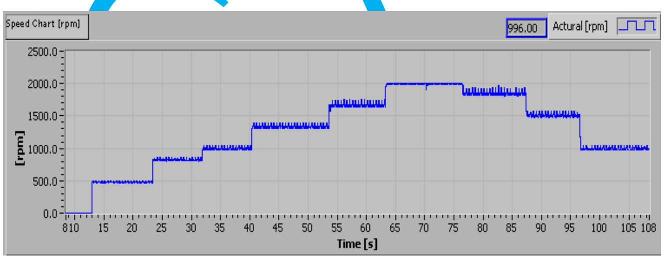

Fig. 8 Variation of estimated speed

In order to control the dc motor speed using the estimated speed it is necessary to design filter algorithms or a solution to remove jumping and varying values.

4. THE SOLUTIONS

4.1 Model-based Speed Estimation Method

In order to eliminate jumping values of estimated speed using the above algorithm, it is necessary to detect the jumping value and assign the jumping value to an approximate value, the first issue and the second issue may be resolved. The model-based speed estimation method uses a mathematical model of speed to generate approximate values of speed at the same input voltage (armature voltage). Thus an armature voltage dependent speed model is used. A mathematical model of speed is added in the program. It provides approximation of the motor speed. The estimated speed is compared with the maximum speed in order to detect the jumping values. Once the jumping value is detected it is assigned to the approximate value generated by the model as shown in Fig. 9.

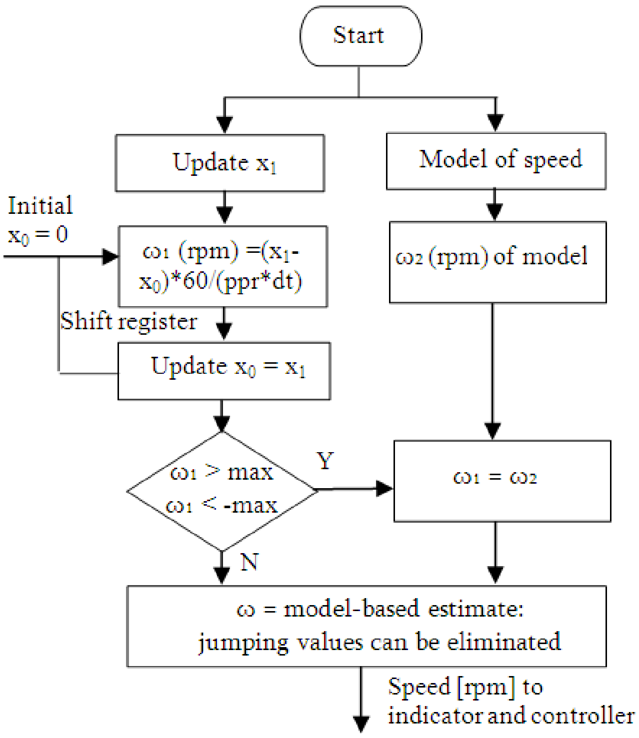

Fig. 9 Flowchart of the model-based speed estimation algorithm

The motor speed estimated by the proposed model-based estimation algorithm is shown in Fig. 10.

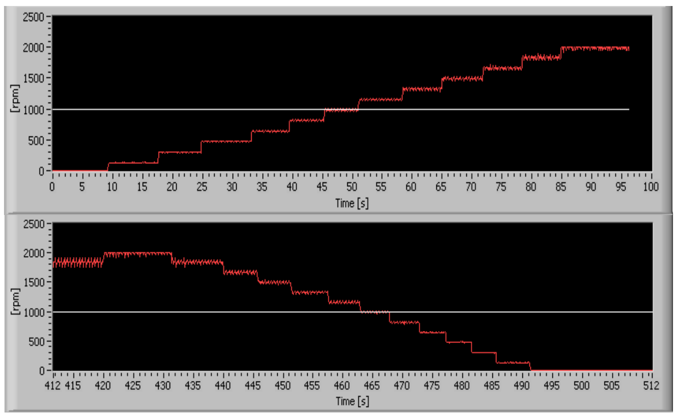

Fig. 10 Jumping values of speed are eliminated by a model-based speed estimation algorithm

Fig. 10 shows that the jumping values of speed due to the resetting of pulse counts are eliminated by the model based speed estimation algorithm.

4.2 Kalman Filter

In order to obtain the estimated speed measurement without variation a Kalman filter (Kalman, 1960, Welch and Bishop, 2004) is designed in the speed estimation algorithm as shown in Fig. 11.

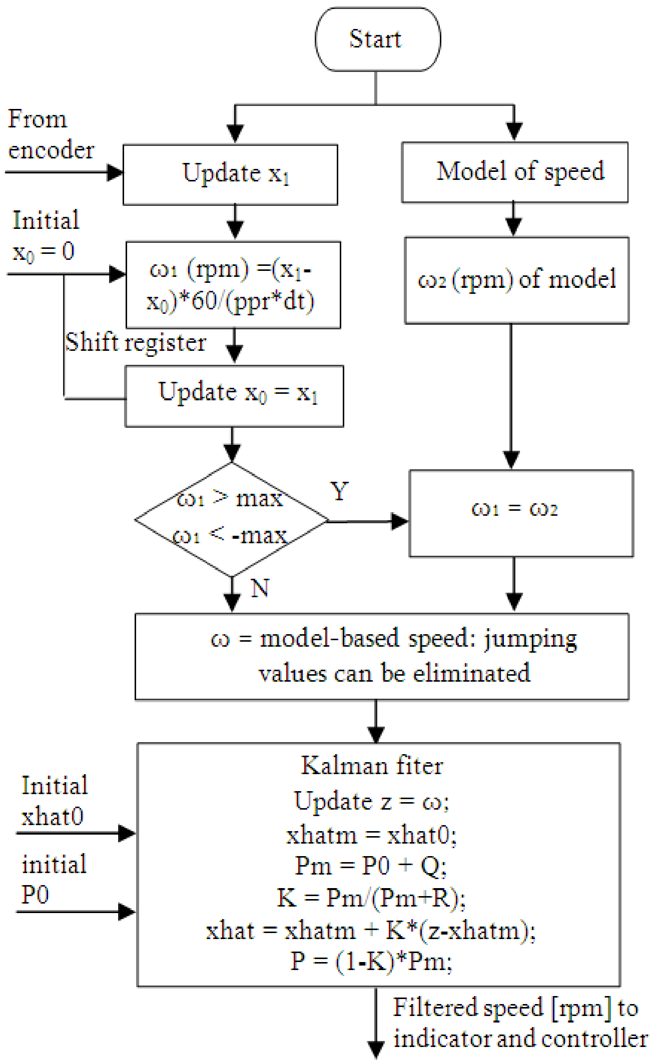

Fig. 11 Flowchart of the model- and Kalman filter- based speed estimation algorithm

5. DEVELOPMENT OF CONTROL PROGRAMS

5.1 Mathematical Model and Simulation

In order to comprehend the control algorithm a mathematical model for the motor used in this article has been developed. The motor is a permanent magnetic dc motor that is similar to an armature-controlled dc motor. An equivalent circuit diagram for the motor is shown in Fig. 12.

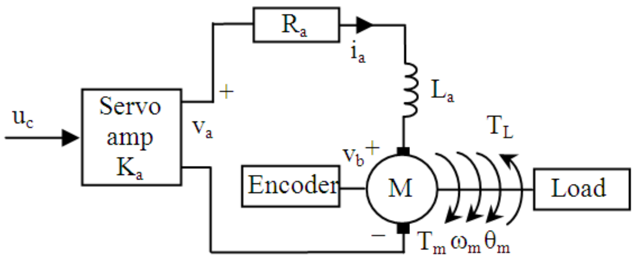

Fig. 12 Equivalent circuit of a PM dc motor

By referring to Fig. 14 the following equations are derived based on Kirchhoff's laws and Newton's second law.

$$v_a = K_a u_c \tag{2}$$

$$L_a \frac{di_a}{dt} + R_a i_a = v_a - v_b \tag{3}$$

$$v_b = K_b \omega_b \tag{4}$$

$$T_m = K_m i_a \tag{5}$$

$$J\frac{d\omega_m}{dt} + B\omega_m = T_m - T_L \tag{6}$$

$$\frac{d\theta_m}{dt} = \omega_m \tag{7}$$

Equations (2) through (7) lead to the block diagram as shown in Fig. 13. This block diagram is used for making simulation programs with LabVIEW.

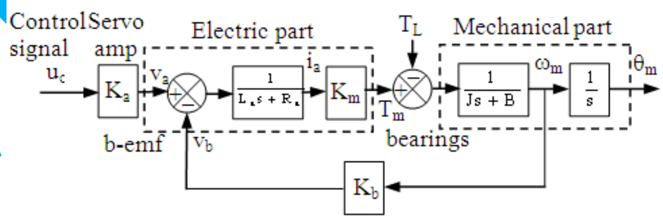

Fig. 13 Block diagram of the open-loop motor system

For convenient programming Equations (3) through (7) can be written in the state space form below.

$$\begin{bmatrix} \dot{i}_a \\ \dot{\omega}_m \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} -R_a/L_a & -K_b/L_a & 0 \\ K_m/J & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} i_a \\ \omega_m \\ \theta_m \end{bmatrix} + \begin{bmatrix} 1/L_a \\ 0 \\ 0 \end{bmatrix} v_a + \begin{bmatrix} 0 \\ -1/J \\ \dot{\theta}_m \end{bmatrix} T_L \tag{8}$$

$$\mathbf{Y} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i_a \\ \omega_m \\ \theta_m \end{bmatrix} \tag{9}$$

A computer simulation program for the open-loop dc motor system can be made with LabVIEW and LabVIEW Control Design and Simulation Module. If the Control Design and Simulation Module is not available a numerical integration method can be used instead. Fig. 14 shows a simulation program.

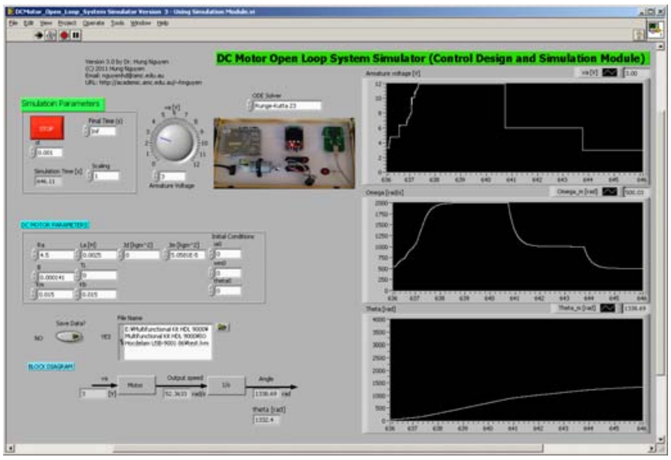

Fig. 14 Front Panel of LabVIEW simulation program

5.2 Simulator and Real Motor Control Program for Educational Purpose

A PC, microcontroller or microprocessor can be used to control a dc motor with an arrangement shown in Fig. 15. Control of a dc motor by a PC requires a data acquisition card.

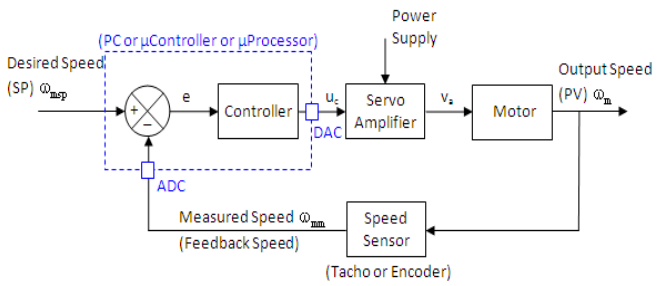

Fig. 15 Means to control a dc motor

With LabVIEW a program including a simulator using a mathematical model and real motor control system can be combined as shown in Fig. 16. The switch between the simulator and real motor control system is done by using a Case Structure in the LabVIEW program as shown in Fig. 17 and 18.

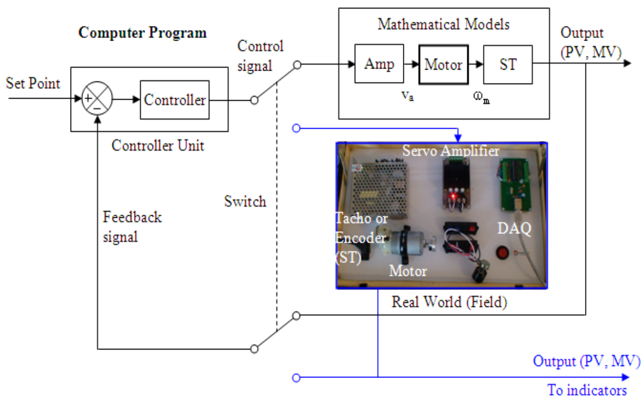

Fig. 16 Conceptual diagram of a combined simulator and real-motor controller program

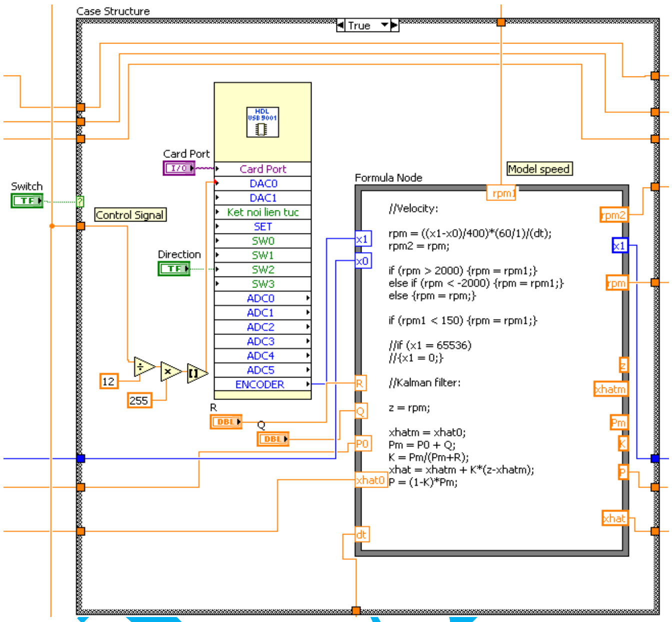

Fig. 17 Case Structure: True for real control system

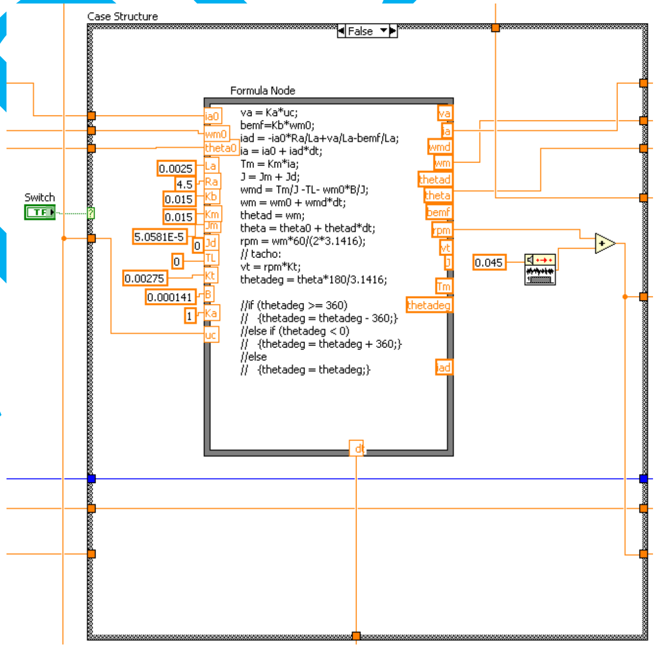

Fig. 18 Case Structure: False for the simulator

The controller algorithm with upper limit, lower limit and Auto/Man switch is shown in Fig. 19.

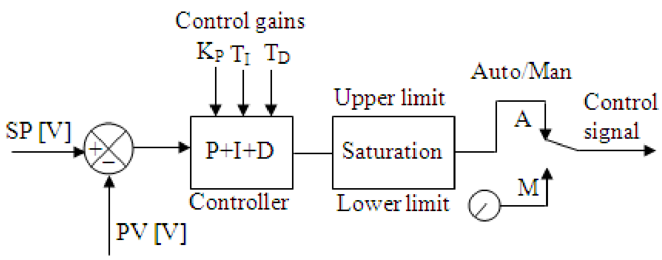

Fig. 19 Controller algorithm with upper limit and lower limit setting block and Auto/Man switch

For verification of the model- and Kalman filter-based speed estimation method a conventional PID control law is applied as follows

$$u_c(t) = K_P e(t) + \frac{K_P}{T_I}\int_0^t e(t)\,dt + K_P T_D \frac{de(t)}{dt} \qquad (10)$$

where K_P, T_I, and T_D are the proportional control gain, integral time (s) and derivative time (s), respectively. This PID control law can be programmed with the LabVIEW PID and Fuzzy Control Toolkit if it is available. If this toolkit is not available it is necessary to build one's own controller.

The combined simulator and real control program made with LabVIEW is shown in Fig. 20. The LabVIEW simulator and real control system program can be used to:

- run the simulator and real motor controller in Man and Auto modes;
- switch between the simulator and real motor controller. The simulator is run in classroom while the real motor controller is run in a lab where an external dc motor is available and connected to PC;
- visualize data (simulated results and actual results);
- simulate a motor speed sensor with random noise signal; and
- save data for analysis.

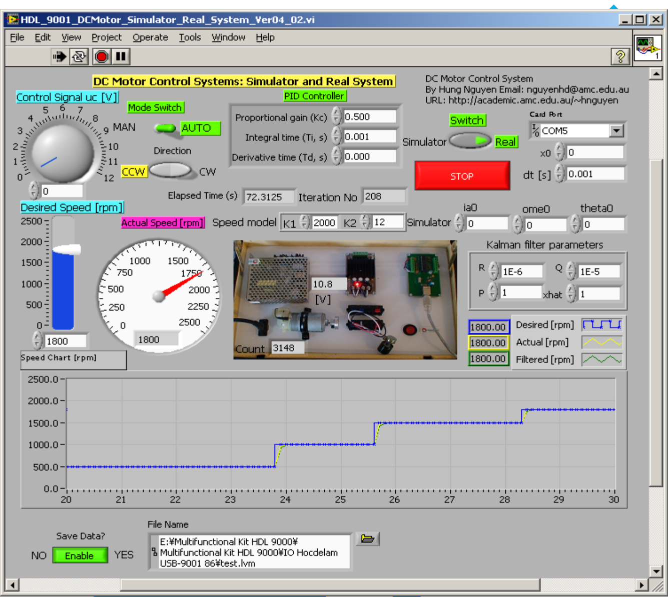

Fig. 20 Front Panel Window of LabVIEW program for the simulator and real motor control system

The system responses based on the model- and Kalman filter-based speed estimation method and the PI control are shown in Fig. 21 and Fig. 22.

Fig. 21 and Fig. 22 show that the process variable (speed) is following the reference signal very well and the speed control objective has been achieved.

6. CONCLUSIONS

In conclusions the article has discussed the issues of jumping values of motor speed in the estimation algorithm and proposed a solution to overcome the issues by applying a model- and Kalman filter-based estimation method. A simple PID control algorithm was used to verify the proposed estimation method.

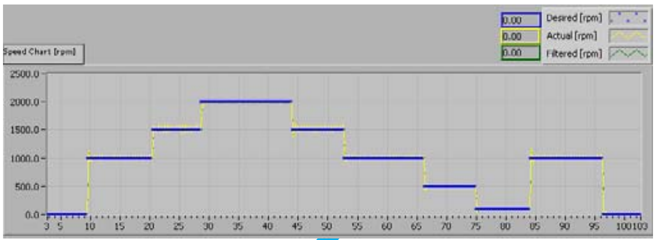

Fig. 21 Step responses

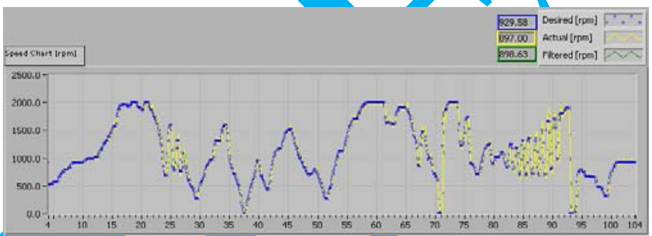

Fig. 22 Responses to any reference speed signal

The control of the motor speed was successful. The article has discussed the development of a computer simulator and real control system for purpose of education and training in automatic control engineering.

Some recommendations for further study of a dc motor speed control system and signal processing are as follows:

- A low-pass filter may be added in the motor speed estimation algorithm to remove unnecessary noise in the speed signal.
- A more complicated control algorithm should be developed to verify the speed estimation method.
- The proposed method can be applied in industrial control systems: control of motor in the cyclic and collective pitch propeller.

REFERENCES

Kalman, R.E. (1960). A New Approach to Linear Filtering and Prediction Problems. Research Institute for Advanced Study. Baltimore.

Merry, R.J.E., Van de Molengraft, M.J.G. and Steinbuch, M. (2009). Velocity and Acceleration Estimation for Optical Incremental Encoders. *The Journal of Mechatronics*. 10.1016/j.mechatronics.2009.06.010

National Instruments (2011a). Quadrature Encoder Velocity and Acceleration Estimation with CompactRIO and LabVIEW FPGA, accessed on 28/8/2011 at http://zone.ni.com/devzone/cda/tut/p/id/3921

National Instruments (2011b). Using Quadrature Encoders with E-Series DAQ Boards, accessed on 28/8/2011 at http://www.ee.nmt.edu/~wedeward/EE443L/FA99/an084.pdf

Nguyen, B.H., Ngo, H.B. and Ryu, J.H. (2009). Novel Robust Control Algorithm for DC Motors. *The 6th*

International Conference on Ubiquitous Robots and Ambient Intelligence (URAI2009).

Patrascouiu, N., Poanta, A., Tomus, A. and Sochirca, B. (2011). Virtual Instrumentation Used for Displacement and Angular Speed Measurements. *The International Journal of Circuits, Systems and Signal Processing*.

Welch, G. and Bishop, G. (2004). An Introduction to Kalman Filter. Department of Computer Science, University of North Carolina.

Zambada, J. (2010). Measuring Speed and Position with the QEI Module. Microchip Technology Inc.

BIOGRAPHY

Dr. Hung D. Nguyen is a lecturer in Marine Control Engineering at National Centre for Maritime Engineering and Hydrodynamics, Australian Maritime College, Australia. He obtained his BE degree in Nautical Science at Vietnam Maritime University in 1991, then he worked as a lecturer there until 1995. He completed the MSc in Marine Systems Engineering in 1998 at Tokyo University of Marine Science and Technology and then the PhD degree in Marine Control Engineering at the same university in 2001. During April 2001 to July 2002 he worked as a research and development engineer at Fieldtech Co. Ltd., a civil engineering related nuclear instrument manufacturing company, in Japan. He moved to the Australian Maritime College, Australia in August 2002. His research interests include guidance, navigation and control of marine vehicles, self-tuning and optimal control, recursive system identification, real-time control and hardware-in-the-loop simulation of marine vehicles and dynamics of marine vehicles.

Appendix A. NUMERICAL VALUES

Table 1 Numerical values of the dc motor parameters used in the simulator

R_a	4.5	B	0.000141	$i_a(0) = 0$
L_a	0.0025	T_L	0	$\omega_m(0)=0$
J	5.0581E-5	$K_m=K_b$	0.015	$\theta_m(0)=0$