

REAL TIME SPEED CONTROL OF ELECTRIC MOTOR VIA MATLAB

Osman ULKIR¹

¹ *Department of Electronics and Automation, Mus Alparslan University, Mus, 49100, Turkey*

Abstract

In this study, it is aimed to keep the electric motor speed at the desired reference value by applying constant and variable reference signals to the system input. Speed control of a permanent magnet brushed Direct Current (DC) motor was performed in real time using the PID control method. Speed control of a dc motor mechanism was performed by accessing the embedded system via Matlab & Simulink software. Data processing cards are normally required for real-time control with Matlab. Since these cards are for commercial purposes and their prices are high, an electronic card based on open code system was used in this project. The robustness of the system was tested by observing the response of the system to disturbing effects and different reference inputs using magnetic braking technique in experimental studies.

Keywords: Electric motor, speed control, PID, MATLAB.

INTRODUCTION

Direct current motors are preferred in many industrial applications due to the important advantages provided by their operating characteristics. Direct current motors are used in applications where adjustable speed and precise position control are required in fast transportation in the industry, electric trains, electric vehicles, and paper industry [1]. There are a variety of DC motors in a wide variety of sizes and prices. Prices are very cheap compared to Alternating Current (AC) motors. In addition, DC motors are compatible with very simple and stable control methods. Another advantage is its high efficiency and high starting torque for sudden load increases [2].

Many years ago, most of the Servo motors used for position control were powered by Alternating Current (AC). Due to the difficult and nonlinear characteristics of the control of AC motors, DC motors are preferred in many applications. On the other hand, brush and commutator in the DC motor make maintenance difficult and increase the cost [3]. Brushless DC motors have emerged as a result of studies carried out due to such problems. Today, brushless DC motors have begun to replace conventional DC motors.

Nowadays, DC motor speed and position control is performed and DC motor control methods also vary [4]. A few of these are traditional P, PI, PID, fuzzy logic based, nonlinear, adaptive variable structure, model reference adaptive control, artificial neural networks, feed forward computed moment control methods [5-6].

PID control method is a method used frequently in industrial applications due to its simple structure [7]. However, the need for a mathematical model despite its simple structure and its inability to give successful results in nonlinear systems can be shown as the disadvantages of this control method [8].

In this study, the speed control of the DC motor was performed with the Arduino Mega 2560 development board using the Matlab & Simulink package program. In the study, the PID control algorithm has been developed to ensure the best response of the system. The effect of the designed controller on the system behavior is observed by applying different coefficient ratios and the most suitable PID parameters are determined. Two different reference input signals were used in the system. First, constant reference value is used with Matlab & Simulink block. The speed control of the system is provided by applying variable reference signals with the

potentiometer at the other reference mark. In our real-time Simulink model, tachometer is used as feedback. The required PWM was formed after the appropriate PID controller was applied by comparing the speed information obtained from the tachometer with the reference value. Thanks to the PWM applied to motor drivers, the motor has reached the desired speed value.

In this study, the speed control of the DC motor was performed with the Arduino Mega 2560 development board using the Matlab & Simulink package program. In the study, the PID control algorithm has been developed to ensure the best response of the system. The effect of the designed controller on the system behavior is observed by applying different coefficient ratios and the most suitable PID parameters are determined. Two different reference input signals were used in the system. First, constant reference value is used with Matlab & Simulink block. The speed control of the system is provided by applying variable reference signals with the potentiometer at the other reference mark. In our real-time Simulink model, tachometer is used as feedback. The required PWM was formed after the appropriate PID controller was applied by comparing the speed information obtained from the tachometer with the reference value. Thanks to the PWM applied to motor drivers, the motor has reached the desired speed value.

ARDUINO DEVELOPMENT CARD

Arduino is an open source development board. Due to its easy programming and being cheap, Arduino appeals to large user masses every day.

In the Arduino development boards produced by Atmel company, they are programmed with their own software Wiring program, basically C based. The fact that the Arduino development board can easily communicate with frequently used software such as MATLAB and LABVIEW for control purposes enables these boards to be widely used today.

Arduino Mega 2560 can be powered by USB and an external adapter or battery. The power source is selected automatically. The card can be powered from an external source

between 6-20V. If an external power source above 12V is used, the voltage regulator may overheat and damage the board. Therefore, it is recommended to use 7 to 12 Volts [9].

It is the Arduino type that uses the Arduino Mega ATmega 2560 processor that we will use in the hardware section. It has 54 digital input / output pins, 14 of which can be used as PWM outputs. It has 16 analog input pins. It has 16 MHz crystal oscillator, USB connection, ICSP header and reset button. By connecting a computer with USB, it can be programmed from the computer by serial communication and data communication can be provided. Figure 1 shows the pins and inputs / outputs of Arduino. Arduino can work in all operating systems.



Fig. 1. Arduino Mega development card

PID CONTROL

PID is a widely used control method today. It is widely applied in many of the applications in the industry [10]. PID control is one of the most used applications in motor control due to its ease of use, simplicity and effective solution. The general block diagram of this control method is given in Figure 2.

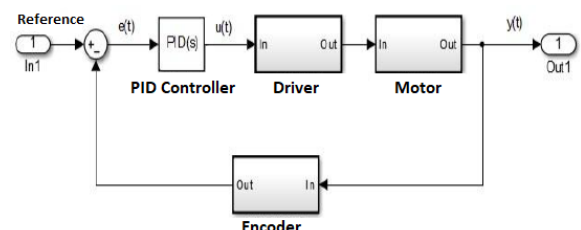


Fig. 2. PID control block diagram

As seen in Figure 2, the difference between the reference value determined and the current position value of the system received through the encoder is sent to the controller as error data. The controller that processes the error data generates an output, which in turn enables

PWM generation. Thanks to PWM applied to the motor with suitable drivers, the motor is moved in a logical way.

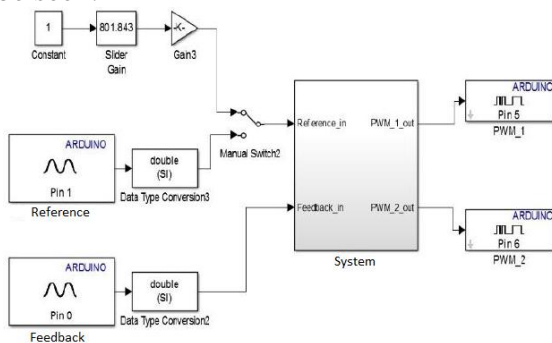
Each of the proportional, integral and derivative gains that make up the PID control affect the operation of the system in various ways. The proportional controller (K_p) is effective in reducing the rise time but never eliminates the steady state error. The integral controller (K_i) is effective in eliminating the steady state error but may make the temporary response worse. The derivative controller (K_d) is effective in increasing system stability, reducing overshoot, and improving transient response. PID controller output:

$$u(t) = K_p e(t) + K_i \int e(t) + K_d \frac{de(t)}{dt} \quad (1)$$

$$e(t) = r(t) - y(t) \quad (2)$$

Here $u(t)$ is the control signal, $e(t)$ is the error signal, and $y(t)$ is the output signal. In PID control, the K_p , K_d and K_i coefficients must be determined appropriately in order for the system output to follow the desired reference value. In this method, the $u(t)$ controller output is obtained by multiplying the error signal $e(t)$ with the proportional term K_p , the integral term of the error signal K_i and the derivative of the error signal with the K_d derivative term.

The reference speed information received from the potentiometer is transferred to the Matlab & Simulink environment via the Arduino board and the required PID control is applied, allowing the system output to reach the desired reference value. Figure 3 shows our model developed in Simulink environment. In our model, our reference sign received from Arduino and the application of the PWM signal required for speed control can be seen.



3. Simulink model of system

Fig.

In this study, by trial and error method, PID parameters K_p , K_i , K_d were determined as 9, 0.8 and 0.01, respectively. It has been observed that with these values, the system output performs better in reaching the desired reference value.

EXPERIMENTAL SETUP AND RESULTS

Real-time speed control of the dc motor will be done with the experimental setup designed to be used in the laboratories of universities. According to the speed information to be entered by the user, the speed control required for the speed of the dc motor to reach the desired value was carried out with PID control. The reference input signal to the system is realized by two methods. The first is a fixed reference mark with Simulink blocks, and the other is a variable reference signal using a potentiometer.

Magnetic braking method was used to see the effect of the system against disruptive effects and its effect on the system response was observed. PID coefficients were experimentally selected as $K_p = 9$, $K_i = 0.8$, $K_d = 0.01$; system response is provided to be the best.

System responses are obtained for different reference input values below. The results obtained from the system with experimental setup are given in this section. The movement of the system response was observed by applying different reference entries. First of all, as the input value, the motor was asked to rotate at 1300 rpm, then the speed of the engine was requested to increase to 2300 rpm. These input signals have been applied to the system in real time using a potentiometer. The response of the system to these input signals is given in Figure 4.

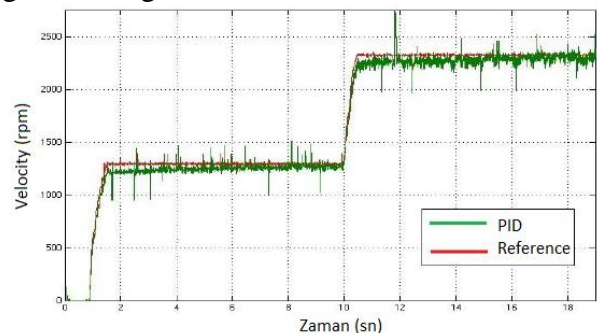


Fig. 4. Variable reference signal and system response

In Figure 5, a fixed reference signal is applied to the system over Matlab. As the input value, the motor was asked to rotate at 2000 rpm. The response of the system to this input signal is given in Figure 5.

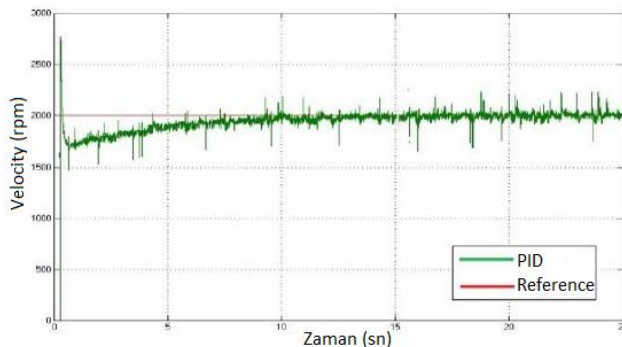


Fig. 5. Stable reference signal and system response

In Figure 6, magnetic braking effect on the system is observed. The motor was asked to rotate at 1000 rpm as a reference signal and magnetic braking was applied to the system for 15 seconds. With this magnetic braking effect applied, the motor slowed down and caught the reference signal again.

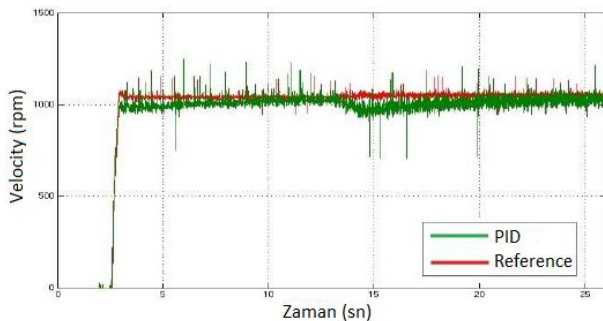


Fig. 6. System response with constant reference signal and magnetic braking

CONCLUSION

In this study, the speed control of a Direct Current (DC) motor is realized in real time using the PID control method. For real-time control of the model developed with the help of Matlab & Simulink in computer environment, an Arduino Mega 2560 development card was used to exchange data on the system with the computer. In the PID control method, better control is achieved by manipulating the coefficients. Due to the high cost of data processing cards, Arduino board was used. In addition, Arduino

board communicates with USB, providing fast data transfer and easy programming.

As is known, production technologies have now become based on automation systems. In this study, the PID control of the DC motor, which is one of the electric motors that forms the basis of all automation systems, was provided in real time.

REFERENCE

- [1] Chan, C.C., "Low Cost Electronic-Controlled Variable Speed Reluctance Motors", IEEE Transactions on Industrial Electronics, 95-100, 1987.
- [2] Siemens Training Education Program, STEP 2000 Series, "Basics of DC drives and related products".
- [3] J.B. Lee, J.H. Choi, J.K. Chung ve J.H. Lim, "Design and implementation of integrated drive circuit for a small BLDC motor," Sixth International Conference on Electrical Machines and Systems, Cilt: 2, s:491-494, 2003.
- [4] A.M. Sharaf ve C. Johnson, "A novel adaptive rule based speed controller for PMDC motor drives," IEEE International Symposium on Industrial Electronics, ISIE '95, s:272-278, 1995.
- [5] S.Saneifard, N.R. Prasad, H.A. Smolleck ve J.J. Wakileh, "Fuzzy-logic-based speed control of a shunt DC Motor," IEEE Transactions on Education, Cilt: 41, No: 2, s:159- 164, 1998.
- [6] N. Barakat ve R. Rajagopalan, "Speed control of a DC motor using a feed forward computer torque control scheme," IEEE International Symposium on Intelligent Control, s:432-437, 1996.
- [7] H.P.Huang. , "On line adaptive tuning for PID controllers" , IEEE Control Theory and Applications, vol. 149, pp. 60-67, January 2002.
- [8] C.C. Lee, "Fuzzy Logic in control systems: fuzzy logic controller: 2", IEEE Transactions on Systems, vol. 20, pp. 419- 435, April 1990.
- [9] Badamasi, Y. A. (2014, September). The working principle of an Arduino. In 2014 11th international conference on electronics, computer and computation (ICECCO) (pp. 1-4). IEEE.
- [10] Tang, K. S., Man, K. F., Chen, G., & Kwong, S. (2001). An optimal fuzzy PID controller. IEEE transactions on industrial electronics, 48(4), 757-765.