



A COMPARATIVE STUDY ON THE POSITION CONTROL METHOD OF DC SERVO MOTOR WITH POSITION FEEDBACK BY USING ARDUINO

Amirul Syafiq Sadun, Jamaludin Jalani and Jumadi Abdul Sukor

Faculty of Engineering Technology, Universiti Tun Hussein Onn Malaysia, Batu Pahat, Johor, Malaysia

E-Mail: amirul@uthm.edu.my

ABSTRACT

Unlike Direct Current (DC) motors, the DC servomotor angular position can be controlled at a specific angle by using a control signal. Arduino has recently been used in a wide range of control approach, especially involving closed-loop system with servomotor due to its flexibility and simplicity. In order to control the DC servo motor, there are three different methods that can be used by using Arduino namely Arduino Integrated Drive Electronics (IDE), Support target for Simulink (Support Package) and Arduino Input/Output (IO) Package. This paper presents a comparative study on the servo motor position control (with voltage feedback) by using the above methods to find out the best technique to control the complex servo motor system with closed-loop position feedback.

Keywords: arduino IDE, support target for simulink, arduino IO package, position feedback, simulink, and servo motor.

INTRODUCTION

A servo motor is defined as an actuator that has the capability to be controlled precisely in terms of rotary and angular position [1]. This type of motor is controlled with a specific angular rotation with the help of an additional servomechanism. In general, this type of motor consists a simple electrical motor, attached to a servomechanism that affects the performance of the motor [2]. This motor is commonly used in the field of automation and robotics, especially for the applications that require a precise position control of the mechanism. Typically, there are three types of servo motor which are positional rotation servo, continuous rotation servo, and linear servo motor. The positional rotation servo is the most common type of servo motor used where the servo output shaft can be controlled to rotate in the angular range of 0 to 180 degrees only. This type of servo is found in many electronic applications such as small scaled robot, radio-controlled cars, and many other applications [2]. On the other hand, the continuous rotation servo motor works almost similar to the positional rotation servo motor, however the angular range for the rotation is in the range of 0 to 360 degrees. Furthermore, this type of servo motor has a different approach in the control signal, where rather than setting the static position of the servo, the signal is interpreted as the direction and speed of rotation [3]. The range of command signal causes the servo to rotate in the direction of clockwise or counterclockwise at a varying speed based on the desired command signals. So far, this type of servo motor is commonly used in mobile robots or small-scaled robotic arm. For the Linear servo motor, its working principle is also similar to the positional rotation servo motor but instead of circular rotation, this type of servo movement is back and forth. Commonly, this type of servo is applied in the heavy duty systems such as actuators in a large model of the airplane system [4].

LITERATURE REVIEW

Overview of positional rotation servo motor

There are two types of positional rotation servo motor which are DC servo motor that is controlled by DC, and Alternating Current (AC) servo motor which is controlled by AC. The servo motor "internal" positional feedback consists of either encoder feedback or voltage feedback. The built-in encoder inside the servo motor provides a pulse signal in which it will be processed by the servo controller to get the position feedback [5]. In contrast, the voltage feedback consists of a built-in potentiometer which is coupled with the servo motor axis to get the relative angular position depending on the value of the potentiometer [6]. In particular, the DC servo motor is mostly utilized in the field of electronic, robotics and automation system, especially for special types of applications where the rotation of the motor only requires a certain angular movement such as for robot arm, robot gripper, and etc [7]. Unlike typical DC motor, the main advantage of DC servo motor is it can provide an angular precision where it will only rotate to the desired position depending on the signal given. A typical DC motor only involves an OFF and ON voltage control signal, while the control signal for servo motor is in the form of pulses that directly proportional to the angle of rotation [8]. Currently in the market, the DC servo motor comes in a variety of shapes and sizes [7]. It also has physical stops placed with the gear mechanism to prevent rotation beyond the limits as well as to protect the rotational sensor (potentiometer) [7]. In general, positional rotation servo motor system can be divided into two types which are closed-loop and open-loop.

Open-loop and closed-loop dc servo motor

An open-loop DC servo motor consists a system that has no feedback reference which can be referred to during the motor operation [9]. Therefore, the system has no mechanism to determine errors or to verify whether the system is performing as expected. In the case of open-loop



DC servo motor, there is no feedback reference that comes out from the servo motor which makes the servo system open-loop when connected to an external controller. However, this type of servo motor is not entirely open-loop. It has an “internal feedback” mechanism that gives the position feedback to the servo controller inside the motor case. The position feedback comes from the potentiometer that is connected to the servo gear mechanism. The potentiometer reading is then converted to a voltage which the values are proportional to the position of the servo shaft. However, the drawback of the “internal feedback” features is that it only provides position feedback for the servo controller which is located inside the motor [10]. The block diagram of an open-loop DC servo motor with “Internal” voltage feedback is illustrated as in Figure-1 workpiece [11]:

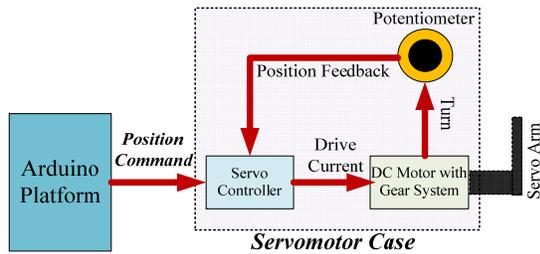


Figure-1. The block diagram of an open-loop DC servo motor with “internal” voltage feedback.

In contrast, a closed-loop DC servo motor consist a system that operates while having a feedback mechanism for a system to verify the desired and actual condition of the output. In the case of DC servo motor, having a closed-loop feedback is a major advantage as the main controller can identify the current position of the motor as well as it can track any errors. Currently in the market, certain models of DC servo motor come with the closed-loop feedback in the form of voltage (potentiometer) or pulses (encoder). As a matter of fact, these feedbacks work similarly to the “internal” feedback where it is proportional to the angular position of the motor shaft. This paper focuses on the use of DC servo motor with voltage position feedback as shown in Figure-2.

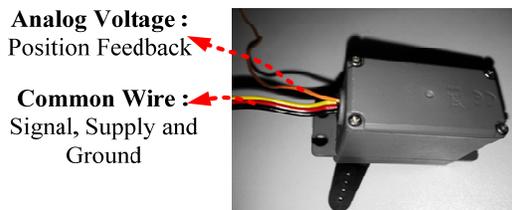


Figure-2. The block diagram of DC servo motor with closed-loop voltage feedback.

Specifically for closed-loop voltage feedback type, the values that come from the potentiometer are located inside the servo motor case in which the voltage

range is most likely to be the same as the “internal” voltage feedback [11]. Figure-3 shows the block diagram of a DC servo motor with closed-loop voltage feedback:

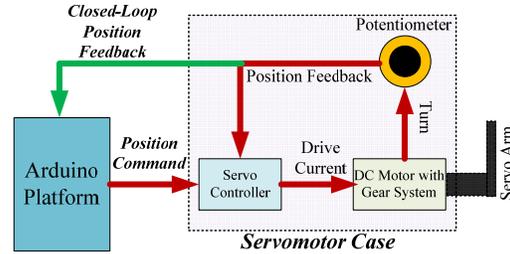


Figure-3. DC servo motor with closed-loop voltage feedback.

Closed-loop DC servo motor with position feedback

The servo motors are controlled by a range of pulses namely Pulse Width Modulation (PWM). The PWM signal determines the shaft position while the pulse duration that was sent will turn the rotor to the desired position either in clockwise or counter clockwise direction. Theoretically, the servo controller expects to receive a set of pulses in every 20 milliseconds (ms) where the motor angular rotations are determined by the pulse width. The relations between pulse width and angular rotation of the servo motor are shown in Table-1 [7]:

Table-1. Process parameters and their levels.

Pulse Width (PW)	Angular Rotations (AR)
$1ms \leq PW \leq 1.5ms$	$0^\circ \leq AR \leq 90^\circ$
$1.5ms \leq PW \leq 2ms$	$90^\circ \leq AR \leq 180^\circ$

In the case of closed-loop DC servo motor, the voltage drop across the potentiometer; which is located inside the servo motor, acts as a closed-loop feedback mechanism. During the servo motor operation, the potentiometer resistance value will increase or decrease, depending on the rotation of the DC motor. By taking advantage of the properties, the resultant voltage range can be divided by the range of angular rotation to get the step voltage for each angular position. In order to identify the relation between voltage and angular value, a measurement was carried out by using Arduino which acts as “DAQ” to receive the voltage feedback from the servo motor. This method will be discussed further in the next topic. Figure-4 shows the measured values of voltage (V) against Angle (Degree):

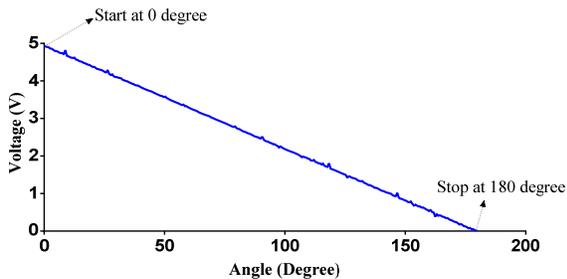


Figure-4. Measured values of voltage (V) against angle (degree).

Based on Figure-4, it can be seen that when the voltage increases, the angular position decreases. Therefore, this type of feedback will definitely improve the functioning of servo motor where it is possible to monitor the desired angular position and the actual angular position simultaneously.

CONTROLLING THE DC SERVO MOTOR BY USING ARDUINO PLATFORM

There are three known methods of DC servo motor position control which are by using:

- Arduino IDE
- Support target for Simulink (Support Package)
- Arduino IO package

Method A is the most common method that can be used to control the servo motor where the users use the Arduino IDE software to do the coding. Next, after the program verification, the “.hex” file is uploaded to the board [12]. Conversely for Method B, different approach is used where instead of using Arduino IDE software, the programming is performed by using MATLAB Simulink. In addition, this method requires an installation of the support package in order to get the library features for MATLAB Simulink [9]. The Simulink blocks or algorithm will then be uploaded to the board via serial transfer. The third method which is Method C works almost similar to Method B, however the algorithm blocks will run on the computer instead of being uploaded to the board. Therefore, the Arduino acts as a “DAQ” or “IO Card” to become a medium for input or output of the system. Similarly, this method requires an installation for the Arduino IO package and the targeted Arduino must be uploaded with the given IDE file. All three methods have the same basic structure which includes programming, verification and communication [9].

The programming implementation for Method A and Method B are different, (i.e. Arduino IDE vs MATLAB, Simulink). However, both methods require a similar process where the verified program codes or blocks need to be uploaded on the Arduino board. For both methods, the Arduino board will immediately start to operate once the uploading process completed. Here, it is very important to note that the parameter of the program cannot be modified since the program has already been uploaded or deployed on the board. In addition, it is

possible to monitor the raw data of input and output for Method A and Method B via the “serial monitor”. However for the purpose of this study, the “serial monitor” method is not suitable for closed-loop servo motor system which requires a real time and consistence data acquisition of the feedback. Hence for Method A and Method B, another targeted Arduino hardware is used as data acquisition hardware “DAQ” to monitor the analog feedback Voltage. On the other hand, Method C allows us to change and monitor the parameters while the program is running. This is possible due to the developed program or algorithm only runs on MATLAB Simulink. This does not only provide a faster control method, but also flexibility during the designing process. In addition, the communication between Simulink and Arduino board only occurs for the analog/digital input and output of the system. Figure-5 summarizes the flow of each of the method:

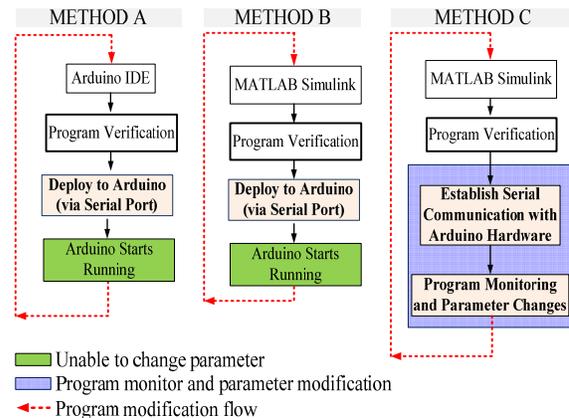


Figure-5. The flow summary of each of the method.

Figure-5 also shows that Method C is potentially the best method in controlling servo motor by using Arduino platform. It is vital to reemphasize that Method C offers flexibility and simplicity for the designing process. In addition, the features make the Arduino works almost similar to “DAQ” or “IO Card” where the data can be acquired in real time while controlling the input and output [9].

SOFTWARE AND HARDWARE SETUP

For the purpose of this study, the hardware was setup to directly control the position while tracking the actual position of the servo motor. The hardware setup consists of Arduino Mega as the main controller and the power source of the servo motor. In the case of Method A and Method B, another Arduino board (UNO) was introduced to act as the data acquisition hardware (DAQ) for the purpose of monitoring the analog voltage feedback from the servo motor. The analog feedback data were then read in real time by using Arduino IO package. In like manner, the actual angular position of the servo motor can be observed. Unlike those two methods, the hardware setup for Method C only requires an Arduino Mega to



control and acquire feedback data from the servo motor, where it do not require any additional Arduino board. The

summary of the hardware setup is shown in Figure-6.

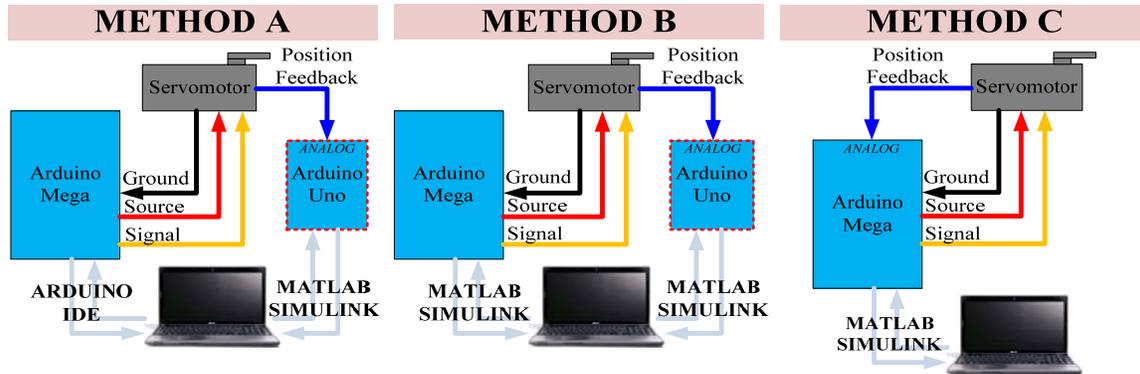


Figure-6. The summary of the hardware setup.

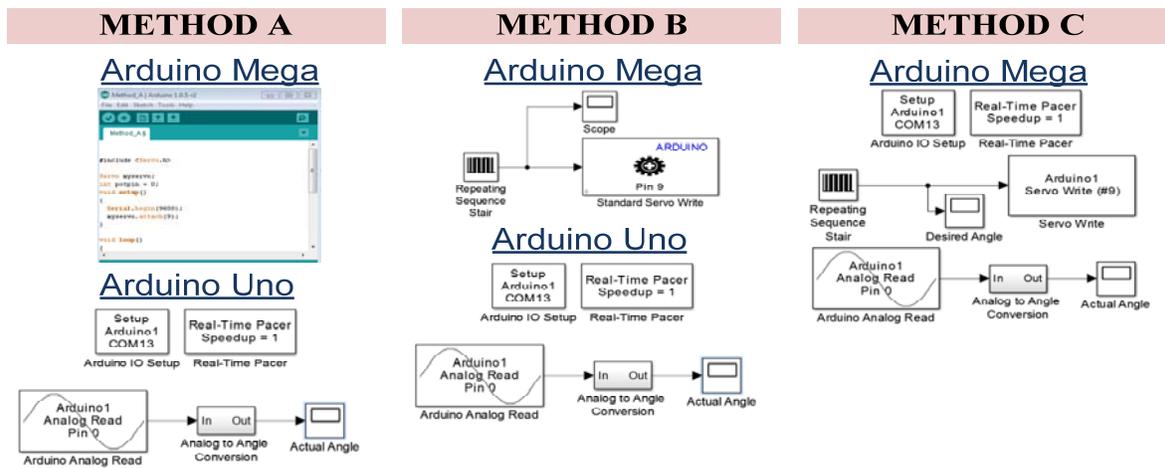


Figure-7. The instructions and Simulink blocks for each method.

In this study, it can be assumed that the desired angular position of all the three methods came from the same source. Therefore, a similar program has been coded for each method where the desired angular position of the servo motor alternating from 0 degree to 180 degree within 15 seconds. A simple instruction was prepared so that the data for the desired and the actual angular position of the motor can be observed. The instructions and Simulink blocks for each method are summarized as in Figure-7.

RESULTS AND ANALYSIS

In order to see the efficacy of each method, the performance of the servo motor response was observed. A desired angle was set (straight blue line) and the actual angle was monitored (intermittent red line).The results of all the three methods are shown in Figure-8.

Based on the results, it can be seen that all the three methods have almost an identical performance. The average times taken for the servo motor to reach the

desired angle are closely similar, either in the clockwise or counter clockwise direction (i.e. from 0 s to 15 s). Hence, it is noteworthy to emphasize that Method C provides an alternative technique to control the servo motor. Its flexibility and simplicity allow a greater choice for the development of servo motor control design. In addition, the technique is faster due to the fact that online parameter change is permitted.

DISCUSSION AND CONCLUSION

In summary, the use of positional rotation servo with position feedback is definitely important in any closed-loop electronic system. The advantages are not only that the angular position can be controlled precisely, but the feedback of actual position can also be obtained during the system operation. The voltage feedback which comes from the built-in potentiometer is accurate enough to represent the actual angular position of the motor. Moreover, introducing the position feedback features with the presented three methods will improve the control



method of servo motor from open-loop to closed-loop system. The three methods used which are Arduino IDE, Support target for Simulink (Support Package) and Arduino IO package were compared in terms of the suitability and performance for future application in a more complex closed-loop system. Having said that,

choosing the most suitable method is important for a more complex closed-loop servo motor system in order to achieve an efficient and reliable result.

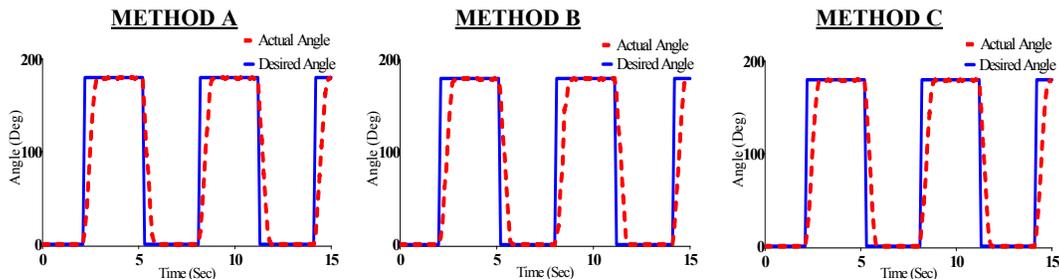


Figure-8. The results of the desired angle against the actual angle.

In conclusion, it is found that Method C is the most suitable method for the application mentioned above. Although the hardware performance of all the three methods are almost equal, Method C was found to be the easiest, simplest and fastest method of integration compared to the other Method A and B. The major advantage of the method is the ability for the user to monitor and change the program parameter in real time while all of the algorithm (control approach) runs on the computer, not on the Arduino.

ACKNOWLEDGEMENTS

This topic is a part of the project which is funded by the Fundamental Research Grant Scheme (FRGS), Vot 1480. The authors also wish to thank the Faculty of Engineering Technology, Universiti Tun Hussein Onn Malaysia for providing a platform to carry out the research activities.

REFERENCES

- [1] N. Pinckney. 2006. Pulse-width modulation for microcontroller servo control. *IEEE Potentials*, vol. 25, no. 1, pp. 27–29.
- [2] R. Ross. 2014. Investigation into soft-start techniques for driving servos. *Mechatronics*, vol. 24, no. 2, pp. 79–86.
- [3] O. Sudjanll. 2013. Trajectory Control of Analog Servo Motor with Limited State Information using Estimated Discrete Time Model. pp. 25–27.
- [4] Z. X. Z. Ximei and G. Q. G. Qingding. 2006. Robust Control for Dual Linear Motors Servo System. 2006 CES/IEEE 5th Int. Power Electron. Motion Control Conf., vol. 3, pp. 6–8.
- [5] H. Yu. 2011. A high performance microchip control system design for DC motor. 2011 Int. Conf. Electron. Commun. Control. ICECC 2011 - Proc., pp. 1210–1213.
- [6] A. Bejo, W. Pora, and H. Kunieda. 2009. Development of a 6-axis robotic arm controller implemented on a low-cost microcontroller. 2009 6th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol. ECTI-CON 2009, vol. 1, no. 1, pp. 328–331.
- [7] H. Eglowstein. 2012. Introduction to Servo Motors. http://www.sciencebuddies.org/science-fair-projects/project_ideas/Robotics_ServoMotors.shtml.
- [8] N. Khongkoom, a. Kanchanathep, S. Nopnakeepong, S. Tanuthong, S. Tunyasirirut, and R. Kagawa. 2000. Control of the position DC servo motor by fuzzy logic. 2000 TENCON Proceedings. *Intell. Syst. Technol. New Millenn. (Cat. No.00CH37119)*, vol. 3, pp. 354–357.
- [9] M. A. Akbar. 2014. Simulation of Fuzzy Logic Control for DC Servo Motor using Arduino based on Matlab / Simulink. 2014 Int. Conf. *Intell. Auton. Agents, Networks Syst.*, pp. 42–46.
- [10] M. Aburaia, E. Markl, and K. Stuja. 2015. New Concept for Design and Control of 4 Axis Robot Using the Additive Manufacturing Technology. *Procedia Eng.*, vol. 100, pp. 1364–1369.
- [11] B. Earl. 2013. About Servos and Feedback. <https://learn.adafruit.com/analog-feedback-servos/about-servos-and-feedback>.
- [12] A. M. Al-Busaidi. 2012. Development of an educational environment for online control of a biped robot using MATLAB and Arduino. *Mechatronics*, 2012 9th Fr. 7th Eur. Congr. Res. Educ. Mechatronics (REM), 2012 13th Int'l Work., pp. 337 – 344.