e-ISSN: 3110-4894 (online-media)
Volume 2 No 2 May 2026 | Page: 39-50
DOI: https://doi.org/10.64803/joeer.v2i2.28

Journal of Electrical Engineering Research

Journal Homepage: https://ejurnal.raskhamedia.or.id/index.php/joeer

Applying Arduino Libraries on Matlab to Design Servo Motor Controller in Sewing Machine

Trung Ngo Kien[1*]

[1]Faculty of Mechanical and Electrical Engineering, Hanoi University of Industry and Trade, Hanoi, Viet Nam
Author Email: [1*]trungnk@hict.edu.vn

ARTICLE INFO

Article history:

Received April 30, 2026
Revised May 10, 2026
Accepted May 10, 2026
Publish May 31, 2026

ABSTRACT

Currently, servo motors are extensively employed in manufacturing equipment, including the garment industry, owing to their high precision and ease of speed variation. Control problems in electronic sewing machines are primarily characterized by motor speed regulation to generate needle motion. The operating principle of the servo motor involves transmitting motion to the main shaft via a belt drive. Consequently, motor speed control can be formulated as a problem of simulating the main shaft motion, which directly affects the machine's actuators, such as forward/reverse stitching, needle stopping, and precise positioning. This paper presents the Arduino library within the MATLAB environment and its application in the design of a controller for a servo motor.

Keywords:

DC Servo, controller, sewing machine, arduino, matlab

Corresponding Author:
Trung Ngo Kien,
Faculty of Mechanical and Electrical Engineering
Hanoi University of Industry and Trade, Hanoi, Viet Nam
Email: trungnk@hict.edu.vn

1. INTRODUCTION

In recent years, the field of mechatronics has experienced rapid growth to meet the increasing demand for electromechanical engineering applications. Microcontrollers and programmable logic controllers (PLCs) have been widely adopted in industrial production systems, including the textile and garment industry, to enhance automation and control performance. The study of microcontroller-based control systems in textile manufacturing, along with the modeling and simulation of servo motor operations, provides an effective approach for developing control strategies in electronic sewing machines. In such systems, motion control is achieved through a servo motor with encoder-based feedback. The position signal from the encoder is continuously compared with a reference input, and the controller generates appropriate control signals to minimize the tracking error and ensure accurate position control. This paper presents the Arduino library integrated within the MATLAB environment and its application in the design and implementation of a servo motor controller for a single-needle sewing machine..

DOI: https://doi.org/10.64803/joeerv.2i2.28

2. ARDUINO AND MATLAB LIBRARY

2.1 Overview

The Arduino board is a microcontroller-based platform designed for programming and interfacing with hardware devices such as sensors, motors, and other peripherals. Arduino has been widely adopted as the processing unit in various applications, ranging from simple to complex systems, due to its advantages, including ease of use, a simple programming language, low cost, and its open-source nature in both hardware and software. Arduino has evolved through a collaborative development model that allows users worldwide to build, enhance, and contribute to the platform. Microcontrollers on Arduino boards can be programmed using the C++ language, with code compiled via the Arduino IDE (Integrated Development Environment) and associated compilers into binary machine code. With Arduino, control algorithms can be easily tested and validated in real-world applications, particularly through its integration with specialized software such as MATLAB, LabVIEW, and others.

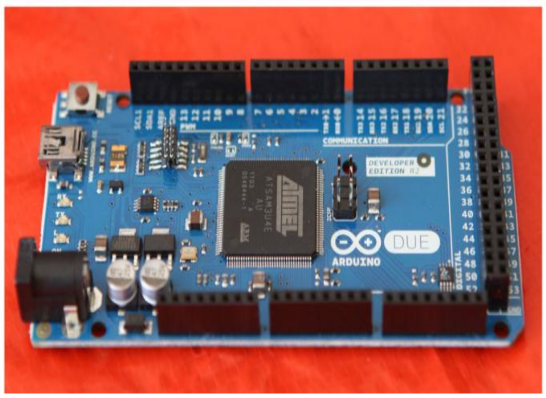

Figure 1. Arduino Board

The Arduino IDE (Integrated Development Environment) is the official open-source software used to facilitate code writing and compilation for Arduino boards. The main program, commonly referred to as a sketch, is developed within the IDE and compiled into a HEX file, which is then uploaded to the microcontroller on the board. The IDE environment primarily consists of two main components: a code editor and a compiler. The former is used for writing the required code, while the latter is responsible for compiling and uploading the code to the Arduino board. This environment supports both C and C++ programming languages.

Figure 2. Arduino IDE startup interface

MATLAB, short for Matrix Laboratory, is a software platform developed by MathWorks for programming, numerical computation, and high-level data visualization. Currently, MATLAB provides thousands of built-in commands and utility functions. In addition to its core functions, MATLAB offers specialized commands and functions through various toolboxes, which extend its capabilities to address domain-specific problems. These toolboxes are highly useful and widely utilized in applications such as digital signal processing, image processing, power electronics, fuzzy logic, and others.

DOI: https://doi.org/10.64803/joeerv.2i2.28

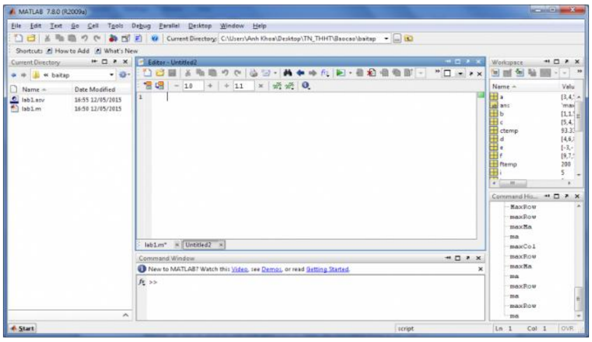

Figure 3. MATLAB interface

Simulink is a toolbox within MATLAB used for modeling, simulation, and analysis of dynamic systems. It provides a comprehensive library of functional blocks, including input/output blocks, signal sources, and linear and nonlinear components. Therefore, it enables users to simulate, analyze, and modify system models at any stage of development

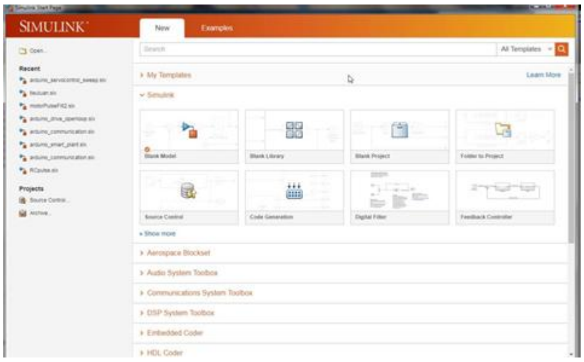

Figure 4. Simulink interface

2.2 Installation of Arduino Library in MATLAB/Simulink

Step 1: Launch MATLAB.

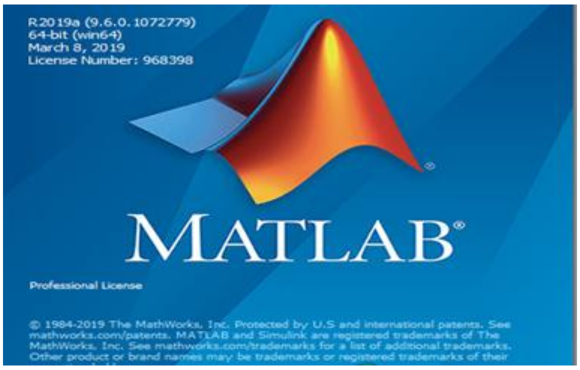

Figure 5. MATLAB startup interface

Step 2: Click the Add-Ons tab in the Home section.

Figure 6. Select the Add-Ons option

Step 3: **Select** Get Hardware Support Packages**.**

Figure 7. Looking For Hardware Packages

Step 4: Open the support package installer and select installation from the Internet.

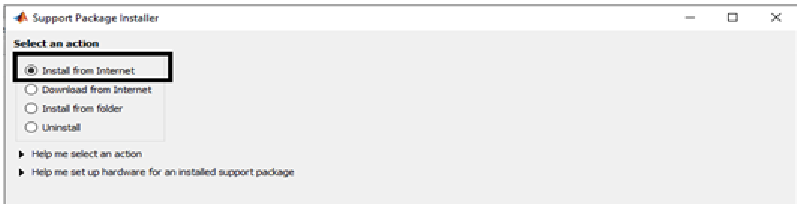

Figure 8. Installing the support package

Step 5: A new window appears displaying all available MATLAB support packages. Locate the Arduino package in the list and click on it to proceed with the installation. A login window will then prompt for MathWorks account credentials. If an account is not available, please create one and continue. Note that both support packages, Simulink and MATLAB, should be installed.

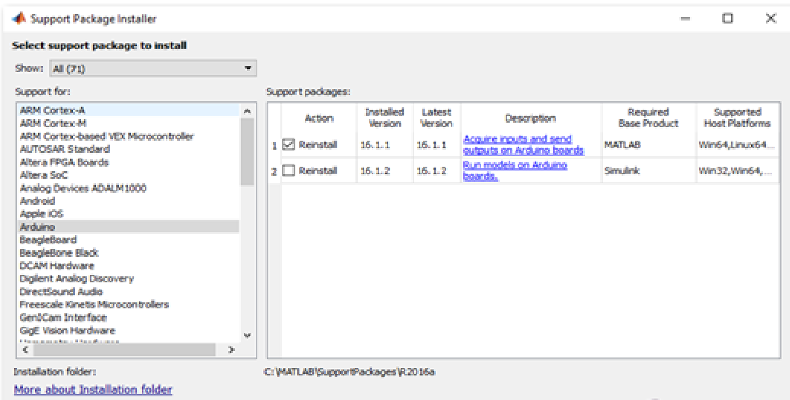

Figure 9. Arduino support package

Verify the installed hardware support packages: After completing the installation, it is necessary to check whether the packages are available in MATLAB by entering the following command in the Command Window:

a=arduino()

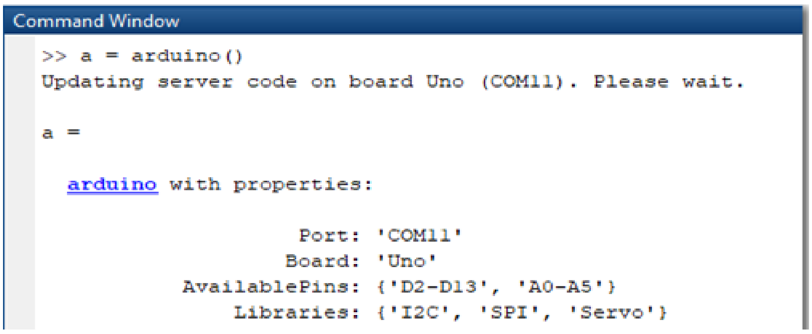

Figure 10. Verification of installation

DOI: https://doi.org/10.64803/joeerv.2i2.28

If more than one microcontroller is connected to the PC, it is necessary to verify the port number. Navigate to the Control Panel, then to **Devices and Printers**, and check the assigned COM port number.

Figure 11. COM port number

When the Arduino is connected to COM11, the port number must be updated in the code before compilation.

a = arduino(com11, uno)

```
>> a = arduino('com11', 'uno')

a =

  arduino with properties:

                    Port: 'COM11'
                   Board: 'Uno'
```

Figure 12. COM port configuration in the code

2.3 Functional Blocks

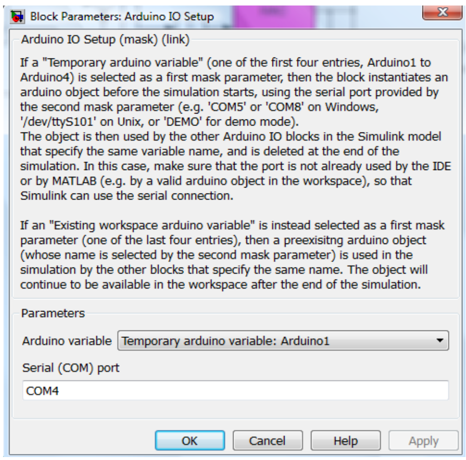

Figure 13. Arduino IO setup

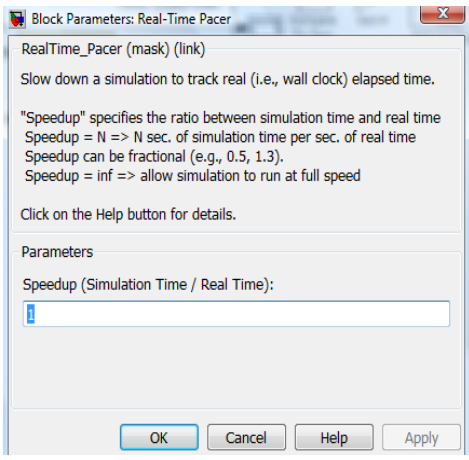

Figure 14. Real-Time Pacer

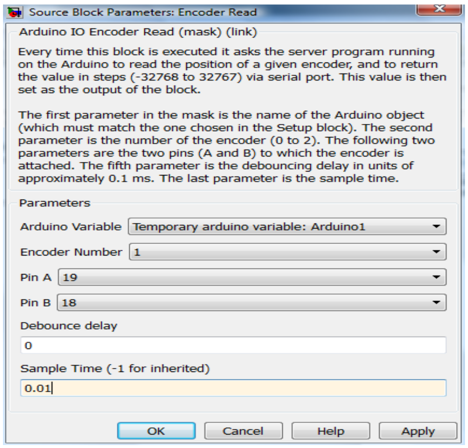

Figure 15. Encoder read

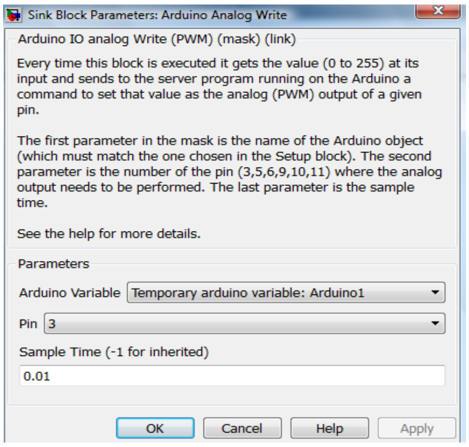

Figure 16. Arduino analog write

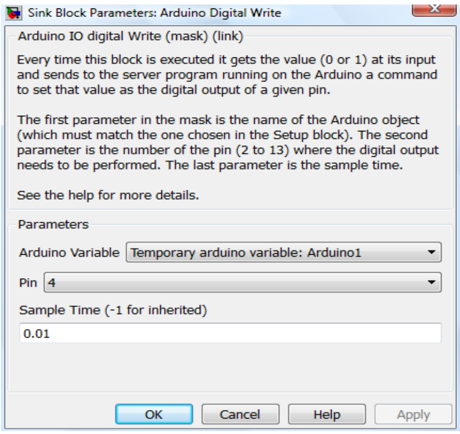

Figure 17. Arduino digital write

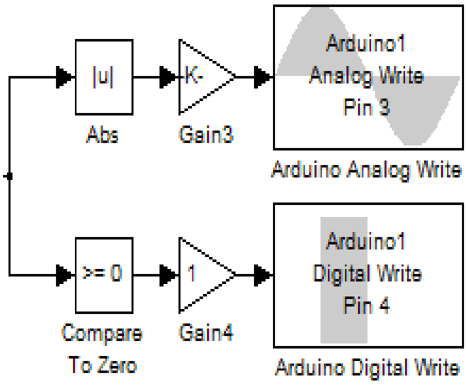

Figure 18. Reversing control

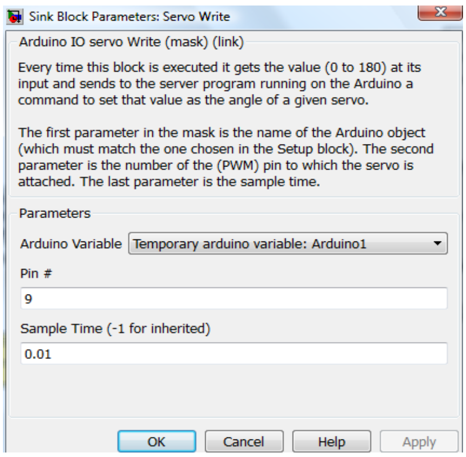

Figure 19. Servo write

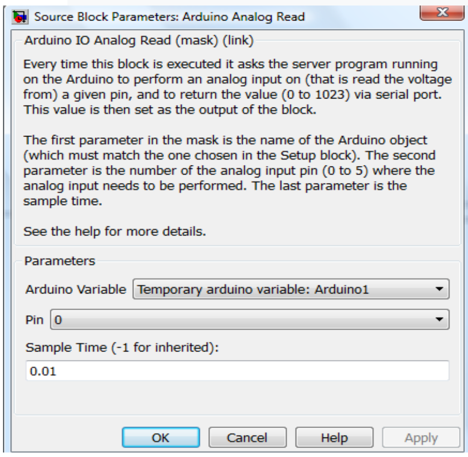

Figure 20. Arduino analog read

3. RESULT AND DISCUSSION

3.1 Control Problem in a Single-Needle Sewing Machine

a. **Precise Stop Position Control**: Electronic sewing machines utilize a speed sensor to determine the needle position, combined with an electromagnetic system to drive the main shaft and bring the needle bar to the desired stopping position. In the experimental setup, the signal from the encoder is compared with the reference input and fed into a PID controller, which computes the required stopping position and generates control signals to adjust the motor shaft to the desired position.

b. **Speed control**: A potentiometer is used to set a specific rotational speed. A potentiometer is also used to adjust the rotational speed (i.e., fast or slow needle motion). The foot pedal (control signal via a potentiometer) is used to regulate the speed. Accordingly, the controller output governs the motor operation in two states: stop and run with gradually increasing speed. The controller adjusts the motor speed through an analog signal from the potentiometer, similar to how pressing the pedal increases speed and releasing it decreases speed. Block diagram of the foot pedal control system shown in Figure 21:

Figure 21. Block diagram of the foot pedal control system

c. **Forward and Reverse rotation control: t**he motor rotates forward, then reverses for a specified number of revolutions (which can be adjusted), and subsequently continues rotating forward. When a stop command is issued, the motor performs a forward rotation followed by a reverse rotation for a specified number of revolutions (adjustable), and then resumes forward rotation until reaching the predefined number of revolutions before coming to a complete stop. The controller is programmed to regulate the motor speed flexibly. Upon receiving stop or reverse commands, the motor must stop and change direction precisely at the required position.

Bold line: forward stitching direction

Thin line: reverse stitching direction

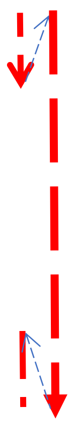

Figure 22. Illustration of needle motion direction**.**

3.2 Control structure design

- The control objective in this system is to achieve precise position control through a servo motor. In the presence of any deviation, the position signal from the encoder is fed back and compared with the reference input of the system. Based on this comparison, the controller generates appropriate control signals to drive the motor such that the output closely tracks the reference signal. The control structure of the system is illustrated in Figure 23.

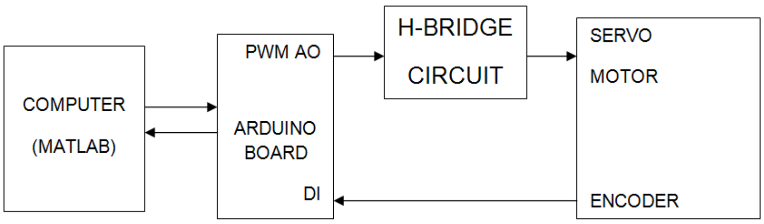

Figure 23. Control structure of the system.

DOI: https://doi.org/10.64803/joeerv.2i2.28

- The controller employs a PID control algorithm to regulate the motor speed according to the reference value (Figure 24), corresponding to the desired needle motion.

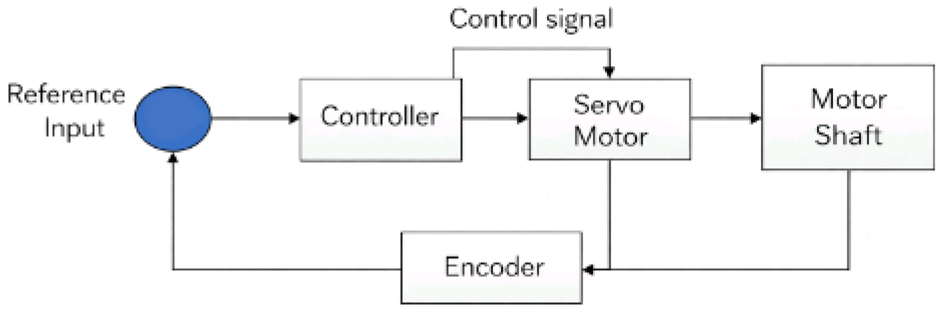

Figure 24. Motor speed control structure.

- The control of motor rotation direction (Figure 25) describes the forward and reverse motion of the needle. It is defined that if signal A leads signal B, the rotation is in the clockwise (CW) direction. In this case, when the encoder shaft rotates clockwise, signal A leads signal B by a phase difference of 90 degrees. Conversely, if the encoder shaft rotates in the counterclockwise (CCW) direction, signal B leads signal A. At this point, signal B has a phase lead of 90 degrees relative to signal A.

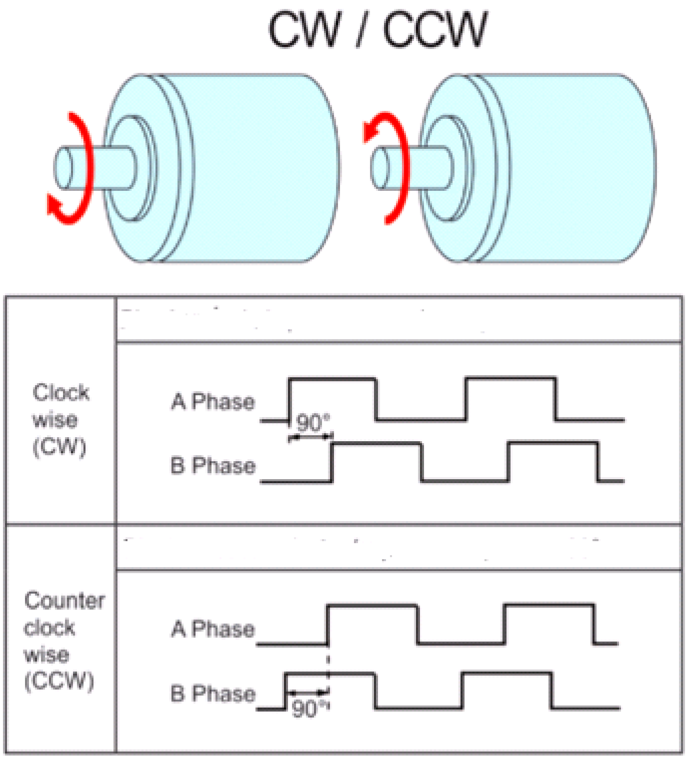

Figure 25. Forward and reverse control of the motor.

The PID controller is designed in the MATLAB/Simulink environment and is implemented in connection with the system via an Arduino board, as illustrated in Figure 26.

DOI: https://doi.org/10.64803/joeerv.2i2.28

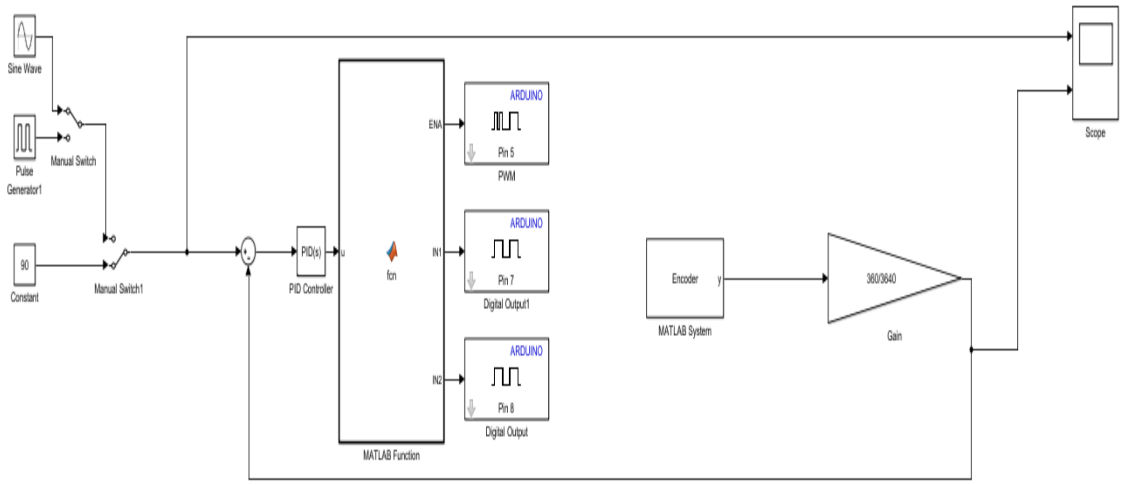

Figure 26. Servo motor control system using Arduino.

3.3 Experimental setup

- The JGB37-545 DC servo motor features a metal construction, providing high durability and stability. It is widely used in applications such as robotic systems, vehicles, and boats. The motor is equipped with a gearbox offering multiple gear ratios, allowing flexible selection between torque and speed (higher torque corresponds to lower speed, and vice versa). The motor is manufactured using high-quality materials, including pure copper windings, 407-grade steel laminations, and high-strength magnetic components, resulting in superior performance and durability compared to low-cost alternatives available on the market (which typically use aluminum windings and weaker magnets). The JGB37-545 DC servo motor is also integrated with a dual-channel AB encoder, enabling precise measurement and control of position and rotational direction in high-accuracy applications such as PID control and autonomous robotics systems.

Figure 27. DC Servo JGB37-545

JOEER, Volume 2 No 2 May 2026 **e-ISSN: 3110-4894 (online-media)**
DOI: https://doi.org/10.64803/joeerv.2i2.28

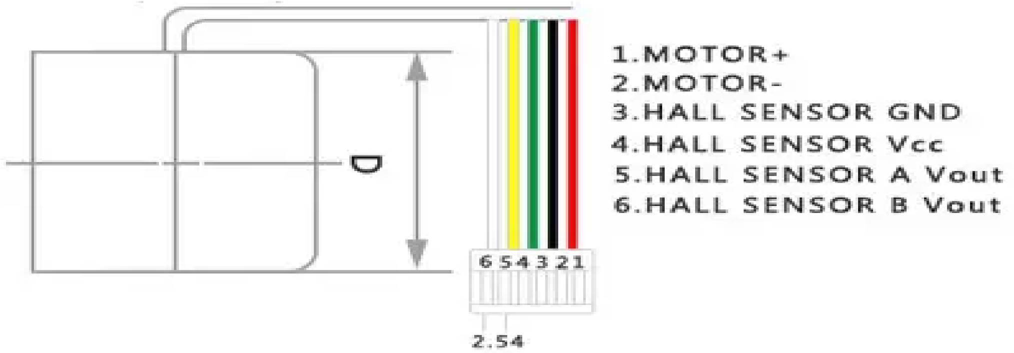

Figure 28. Encoder wiring diagram of the JGB37-545 DC servo motor.

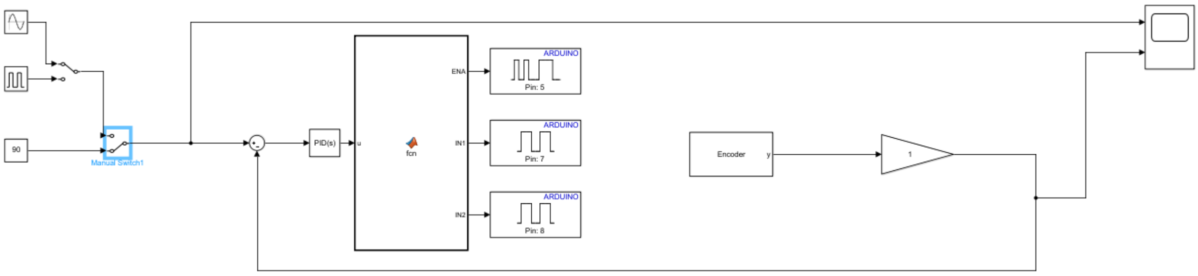

Figure 29. Input signal with a 90° angle

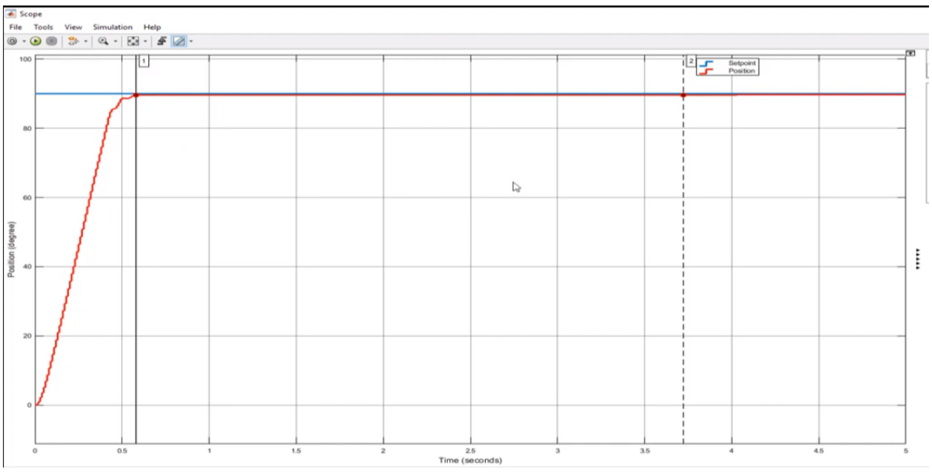

Figure 30. Experimental results with a preset input angle of 90°.

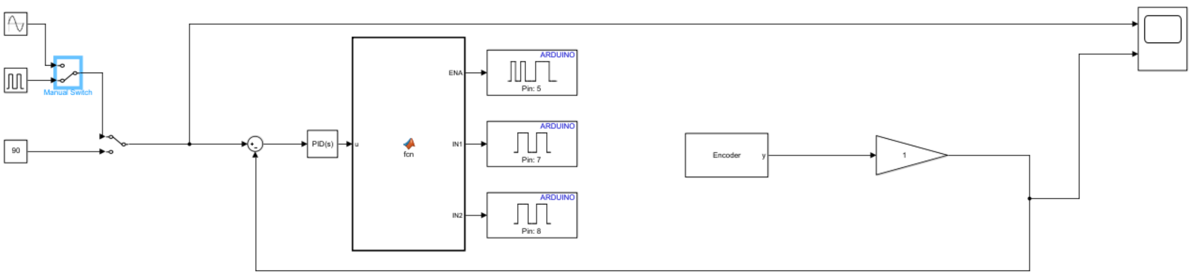

Figure 31. Input signal in the form of a square wave

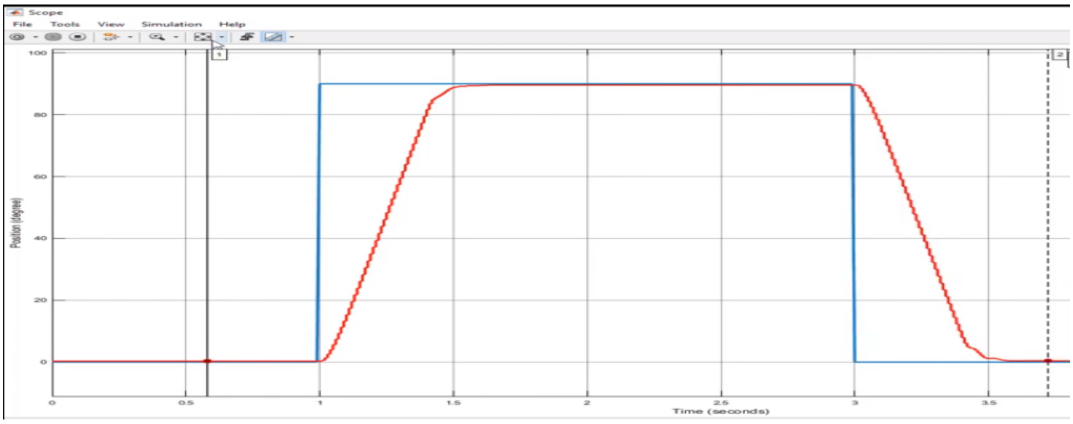

Figure 32. Experimental results with a square wave input.

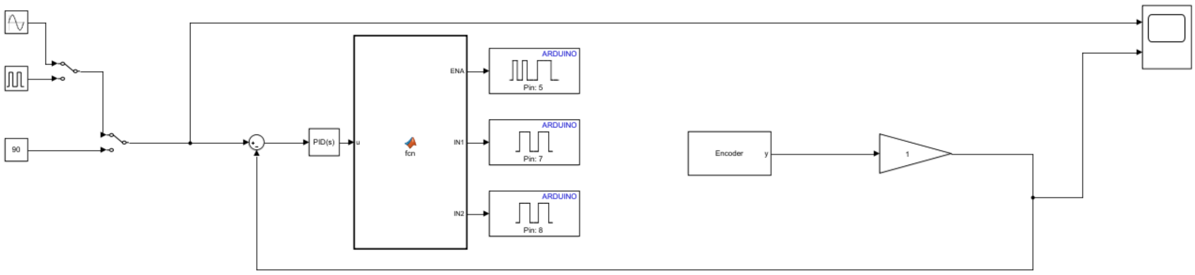

Figure 33. Input signal in the form of a sine wave.

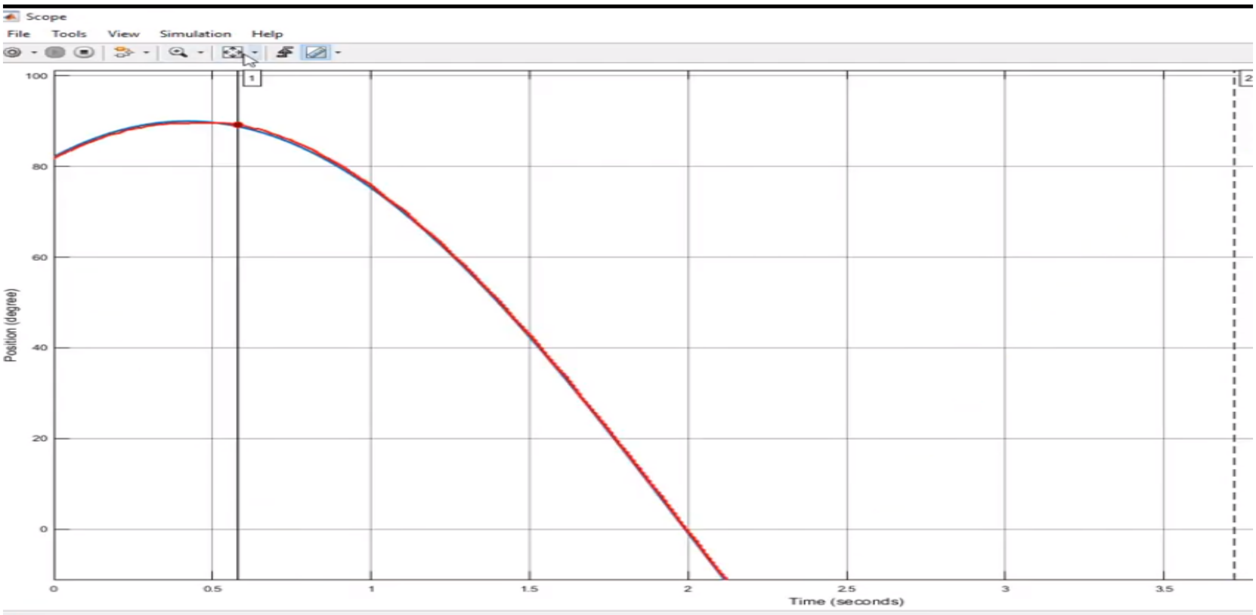

Figure 34. Experimental results with a sinusoidal input signal.

Remark: Motion control of the system is implemented via servo motor actuation. The position signal acquired from the encoder is fed back and compared with the system's reference value. Based on the resulting error, the controller generates an appropriate control signal to drive the motor such that the output tracks the reference input.

DOI: https://doi.org/10.64803/joeerv.2i2.28

4. CONCLUSIONS

Servo motors are widely applied in various industrial fields in general, and in the textile and garment industry in particular, such as in electronic sewing machines, bartacking machines, and buttonhole machines. This paper has presented the Arduino library on the MATLAB platform and its application in designing a servo motor controller for a single-needle sewing machine. The experimental results obtained from the implemented system satisfy the required performance criteria, demonstrating the feasibility of designing servo motor controllers for sewing machines as well as other industrial applications.

REFERENCES

[1]. Amol More et al., 2025, Modified Sewing Machine for Disabled People - An IOT and Arduino-Based Approach: Review, International Journal of Scientific Research in Engineering and Management, Vol. 9, Iss. 11.
[2]. Hung N. T. et al. , 2021, *Industrial Sewing Equipment*, Science and Technology Publishing House.
[3]. Hari K. K. et al., 2021, A systematic literature review on prototyping with Arduino: Applications, challenges, advantages, and limitations, Computer Science Review, Vol.40 - 100364
[4]. Quang N. P., 2006, Simulink MATLAB Textbook, Science and Technology Publishing House.
[5]. Truong D. N. et al. , 2018, Microcontrollers and Applications (Arduino User Guide), Youth Publishing House.
[6]. Vu Q. H. , 2011, Electric Motor Control Engineering, Education Publishing House.
[7]. Trento D. et al., 2020, Application of Arduino-Based Systems as Monitoring Tools in Indoor Comfort Studies, International Journal of Architectural Engineering Technology, 2020, Vol.7.
[8]. JUKI *Corporation, 2011,* DDL-8700-7 EM00e Manual.
[9]. Jinzhu Shen et al., 2025, Autonomous Sewing Technology and System: A New Strategy by Integrating Soft Fingers and Machine Vision Technology, Textiles 2025, Vol.5, Is. 45.