The Indonesian Journal of Computer Science
www.ijcs.net
Volume 13, Issue 5, October 2024
https://doi.org/10.33022/ijcs.v13i5.4373

Design and Development of a 5-DOF SCARA Robot Arm for Robotics Education in a STEM Laboratory

Rokhmat Febrianto[1], Djukarna[2], I Wayan Adi Saputra[3], Agung Alfiansyah[4], Syafrudi[5]
rokhmat.febrianto@prasetiyamulya.ac.id[1], djukarna@unpar.ac.id[2], wayanadisaputra26@gmail.com[3], agung.alfiansyah@prasetiyamulya.ac.id[4], syafrudi@prasetiyamulya.ac.id[5]
[1,3,4,5]Department of Computer Systems Engineering, Universitas Prasetiya Mulya, Indonesia
[2]Department of Physics Education, Universitas Katolik Parahyangan, Indonesia

Article Information

Received : 19 Aug 2024
Revised : 18 Sep 2024
Accepted : 3 Oct 2024

Keywords

SCARA, Arm Robot, Microcontroller, Kinematics, STEM.

Abstract

The development of a 5-degree-of-freedom (DOF) SCARA robot arm was successfully achieved for educational use within the CSL Laboratory at the School of Applied STEM, Universitas Prasetiya Mulya. The design utilizes cost-effective, locally sourced materials and an open-source control system based on Processing Java and Arduino C. These features make the SCARA robot arm an accessible tool for students to learn robotics, particularly in the areas of kinematics, control, and programming. Extensive testing of the robot's inverse kinematics algorithm showed promising results, with average error rates of 1.20% for the Inner Arm, 4.21% for the Outer Arm, and 3.39% for the Z-axis. These low error rates highlight the robot's precision in movement. This research not only met its objective of creating an accessible platform for teaching robotics but also demonstrated potential for future development in robotics education and industrial applications.

A. Introduction

Robotic technology has been widely applied in various sectors, including industry, medicine, education, and others. This technological trend introduces new challenges in engineering education, particularly in the field of robotics. In the educational process, technologies derived from industry must be naturally integrated so that students can directly experience the benefits and limitations of these technologies during their learning [1]. This approach aligns with STEM (Science, Technology, Engineering, and Mathematics) education, which has recently seen a significant increase in interest and is considered highly effective [2].

The emphasis on STEM education is driven by the urgent need to create a workforce with advanced technical skills and the ability to address complex real-world problems. STEM education provides a comprehensive learning experience by integrating academic concepts with practical applications, allowing students to connect their studies with the challenges faced by society today. This interdisciplinary approach is crucial in shaping a STEM-literate population capable of innovation and adaptation in a rapidly evolving world [3]. In line with STEM education, modern robotics learning demands an interdisciplinary approach [4] and requires supportive laboratories and teaching tools.

In robotics education, traditional methods often emphasize classroom-based teaching, where students participate in hands-on group activities and experiments. Nevertheless, independent learning strategies, such as trial and error, present certain risks, particularly in terms of safety. Conversely, project-based learning (PBL) has become increasingly popular, especially within engineering disciplines [5]. Furthermore, the use of case studies has been shown to improve students' critical thinking, problem-solving abilities, and higher-order cognitive skills, in addition to enhancing their conceptual understanding and willingness to engage with the material [6], [7]. Microcontrollers, in particular, have been recognized as valuable tools for PBL, serving a wide range of functions in both everyday life and automated industrial processes [8]. Recently, affordable, open-source microcontrollers like Arduino have gained prominence, proving to be highly beneficial for lab settings [9] and prototyping work [10]. Not only are they cost-effective, but they are also accessible to individuals with basic knowledge of programming and electronics.

Robots can play a significant role in achieving educational goals [11] and are highly applicable in STEM education [12]. Specifically, in relation to industry, the SCARA (Selective Compliance Assembly Robot Arm) type is a particular kind of robot arm designed with unique features to optimize performance in tasks that demand high precision and speed. The SCARA robot arm was first developed in the 1970s and has since been widely adopted across various industries [13]. Its distinctive characteristic is its design, which allows for rapid and accurate movement within a flat (2D) plane, albeit with limited degrees of freedom. Typically, a SCARA robot arm features three or four axes that operate simultaneously to move the end-effector. The freedom of movement in these axes enables the SCARA robot arm to perform both horizontal and vertical motions with high precision, along with rotation around a specific axis [14]. The primary advantage of using SCARA robot arms lies in their effectiveness in pick-and-place tasks, sorting, and assembly operations [15]. In the manufacturing sector, SCARA robot arms are frequently utilized to pick up components or objects from one location and place them with great accuracy in a designated spot, thereby making assembly or processing more efficient and precise, and boosting productivity. Additionally, SCARA robot arms are known for their speed and reliability. Their rigid and compact design allows for swift and precise movements, making them ideal for operations with short work cycles. Moreover, the robust and durable structure of SCARA robot arms ensures they can withstand harsh industrial environments [16]. However, the availability of SCARA robot modules in STEM labs is still limited, making it challenging for students to gain a comprehensive understanding of this advanced 5-DOF SCARA robot in real-world applications.

Therefore, this paper will discuss the design and development of a 5-DOF SCARA robot used in a STEM laboratory for educational purposes. In the context of robotics education, the application of SCARA robot arms is crucial in providing students with practical experience in industrial robotics development and programming. Students can learn about the kinematics, control, and programming of SCARA robot arms, which are highly sought-after skills in today's industry. Overall, SCARA robot arms are highly effective solutions for robotics tasks that require precision and high speed. Whether in manufacturing or in the context of robotics education, SCARA robot arms deliver reliable performance and enhance operational efficiency.

B. Related Works

The SCARA robot arm is a specialized type of robotic arm designed to optimize performance in tasks that demand high precision and speed. First developed in the 1970s [17], the SCARA robot arm has since become widely used across various industries. Its primary feature is a design that allows for fast and accurate movements within a two-dimensional plane (2D) while operating with limited degrees of freedom . Typically, a SCARA robot arm consists of three or four axes that work simultaneously to maneuver the end-effector. These axes enable the SCARA robot arm to perform horizontal and vertical movements with high precision, as well as rotation around a specific axis [18].

The key advantages of using a SCARA robot arm lie in its effectiveness for applications such as pick-and-place tasks, sorting, and assembly operations. In the manufacturing industry, SCARA robot arms are frequently employed to pick up components or objects from one location and place them with high accuracy in a designated spot. This capability enhances the efficiency and precision of assembly or processing tasks, ultimately boosting productivity [19].

SCARA robot arms are also known for their speed and reliability. Their rigid and compact design enables fast, precise movements, making them ideal for operations with short cycle times. Additionally, SCARA robot arms are built to be strong and durable, allowing them to withstand the demanding conditions of industrial environments [20].

In the field of robotics education, SCARA robot arms play an essential role in providing students with practical experience in industrial robotics development and programming. Through working with SCARA robots, students can learn important skills in kinematics, control, and programming skills that are highly valued in the industry today. Overall, SCARA robot arms are a highly effective solution for robotic tasks that demand precision and speed. Whether in manufacturing or educational contexts, SCARA robot arms offer reliable performance and contribute to increased operational efficiency.

C. Proposed Method

The design of the 5-DOF SCARA robot arm can be divided into two main components: the mechanical construction of the arm and the controller system. The mechanical construction focuses on the physical structure and movement capabilities, ensuring that the arm can perform precise tasks with stability and accuracy. On the other hand, the controller system is responsible for managing the movements of the robot arm, translating programmed commands into real-time actions. To achieve accurate motion, the SCARA robot requires precise kinematic calculations, which are essential for determining the correct angles and positions of the joints to ensure that the robot arm moves according to the intended program. These calculations are critical for enabling the robot to perform complex tasks with precision, making it a versatile tool in both educational and industrial settings.

1. Design of a 5-DOF SCARA Robot Arm

The mechanical construction of the robot arm is divided into five sections, corresponding to the number of degrees of freedom of the arm. These sections can be seen in the following Figure 1.

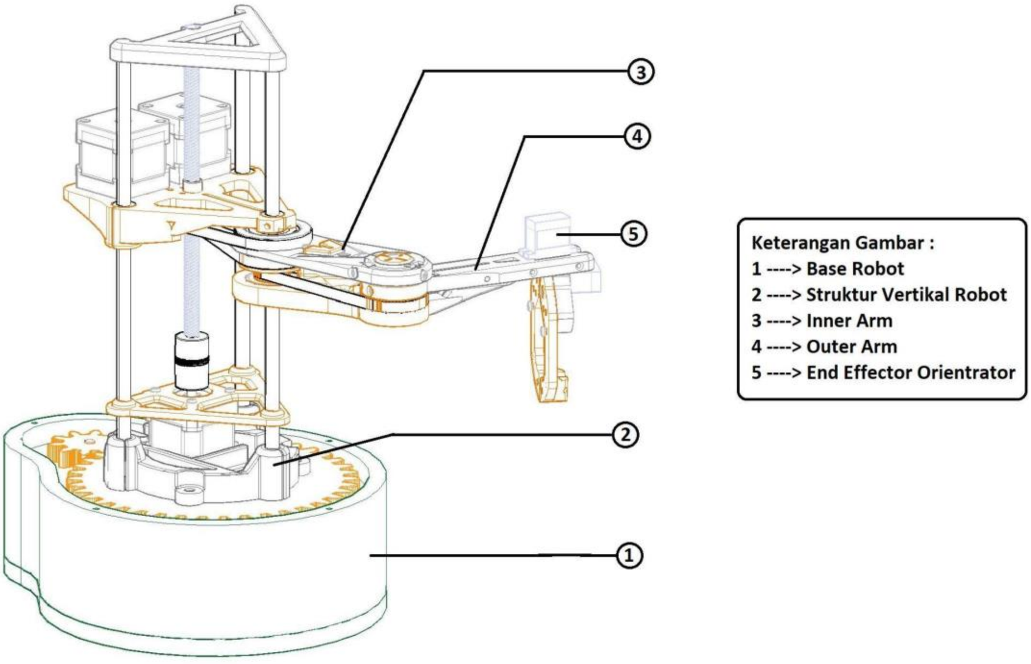

Figure1. Robot Arm Construction Parts.

In general, the mechanical construction of the robot arm is made from PLA (Polylactic Acid) plastic and corrosion-resistant stainless steel type 304. The production of the robot arm is carried out using 3D printing and simple machining, allowing the entire manufacturing process to be completed in-house at the CSL laboratory at Universitas Prasetiya Mulya.

2. Design of Robot Control System

The primary function of the robot control system is to manage the robot's movements, whether manually or automatically. This control system is implemented using a computer or microcontroller. The core of this research is the development of a cost-effective and flexible control system that can be easily programmed by students.

The block diagram of the robot control system using a PC is shown in Figure 2 below. The central control unit of this system is located on the computer (PC). The computer receives inputs, performs calculations, and then sends the results to the driver to move the stepper and servo motors accordingly. A microcontroller is required to connect the computer with the driver. For the 5-DOF SCARA robot, an Atmega 2560-Arduino microcontroller is used. A CNC shield v3 and A4988 stepper driver are utilized to drive the stepper motors, while the servo motors can be controlled directly using PWM through the microcontroller. Limit switches are installed on the robot arm to provide input to the microcontroller and computer when the arm reaches its movement limits.

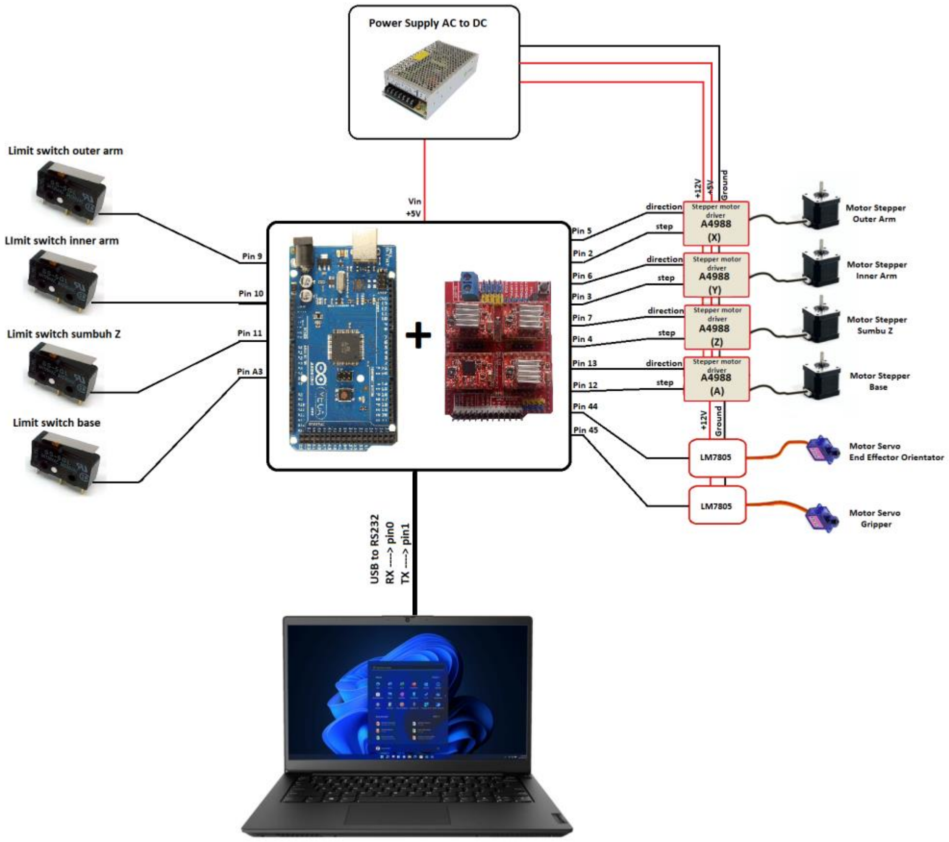

Figure 2. Block Diagram Of 5 DOF SCARA Robot Control System Using Computer.

The main control center of the robot is located in the computer (PC), while the Arduino Mega2560 microcontroller functions as the interface between the robot and the PC. The robot controller program on the computer is created using the Java-Processing language, while the controller (interface) program on the Arduino Mega2560 is developed using the Arduino IDE.

Communication between the computer and the robot is done via a serial-USB communication system. The computer sends commands in the form of serial string data, which is then interpreted by the microcontroller into joint commands, direction, and the magnitude of the movement angles. The communication sequence between the computer and the robot can be illustrated by the block diagram below.

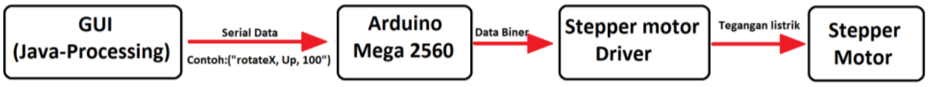

Figure 3. command flow in the SCARA robot controller.

Communication from the robot to the computer is more straightforward. The robot is equipped with limit switches to "inform" the controller when the robot arm reaches the end of its movement range. The limit switch sends a voltage signal to the controller (Arduino Mega2560), which then determines the next step or stops the arm's movement. The block diagram below shows the signal flow from the robot's limit switch to the Arduino Mega2560 controller.

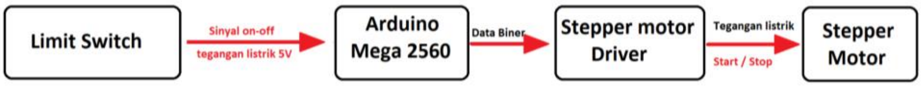

Figure 4. signal flow from the limit switch to the SCARA robot controller.

The control system of this SCARA robot is currently an open-loop control system, which can be further developed into a closed-loop control system by implementing feedback mechanisms. This SCARA robot is equipped with a Graphical User Interface (GUI) that makes it easier for users to control the robot.

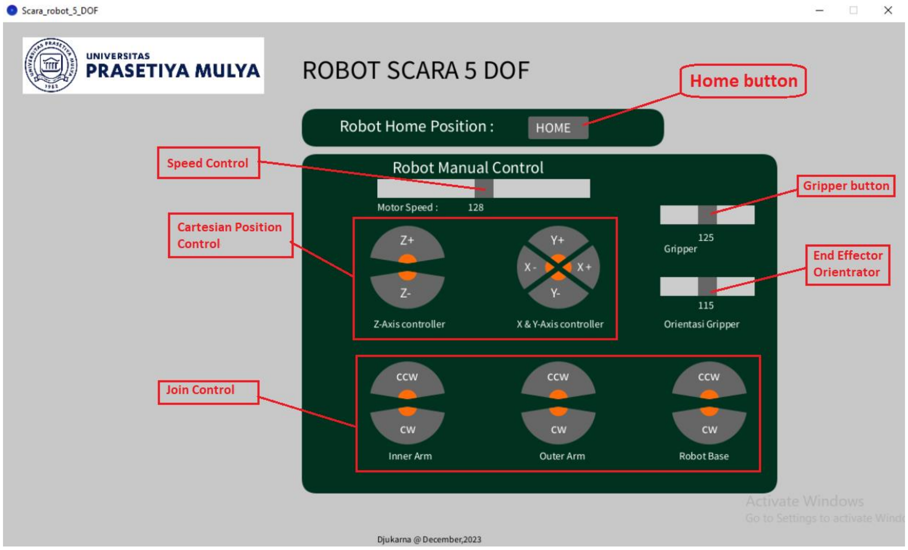

Figure 5. Graphic User Interface Robot SCARA 5 DOF.

3. Denavit-Hartenberg (DH) Parameters for a 5-DOF SCARA Robot

A robot arm is composed of a combination of links and joints. This leads to a complex and intricate construction of the transformation matrix for the robot arm. The more links and joints the robot arm has (which can also be referred to as degrees of freedom (DOF)), the more complex the transformation matrix required. One method that can be used to simplify the construction of this transformation matrix is the Denavit-Hartenberg (DH) method.

The Denavit-Hartenberg (DH) method is a notation technique and convention used to model the kinematic relationships between segments in the kinematic chain of a robot. This method is named after two engineers, Jacques Denavit and Richard Hartenberg, who introduced it in 1955. The DH method is highly useful in the analysis of kinematics and motion planning of robots.

In general, the DH method describes each segment or link on the robot using four geometric parameters, providing a complete description of the relative orientation and position between segments. These four parameters are: link length (a), rotation about the Z-axis (α), parallel axis distance (d), and rotation about the X-axis (θ).

The application of the DH method involves placing a local coordinate system on each joint or segment of the robot and then defining the homogeneous transformation (transformation matrix) between the coordinates sequentially. By using DH parameters, the transformation matrix can be determined to describe the relative displacement and orientation between segments.

The DH method offers the advantage of enabling a simple and systematic kinematic representation, facilitating easy mathematical analysis. Using DH notation allows for quick and efficient kinematic analysis of robots, which is crucial in designing motion controllers and trajectory planning for robots.

D. Result and Discussion

The SCARA robot arm is composed of four segments, each of which can be defined using the Denavit-Hartenberg method. The following outlines the DH parameters for each segment of the SCARA robot arm.

The Degrees of Freedom and the Local Coordinate System for each segment of the 5-DOF SCARA Robot Arm are illustrated in Figure 6. The Denavit-Hartenberg parameters corresponding to these segments are summarized in Table 1 below. This table provides a comprehensive overview of the key parameters used to model the kinematic relationships between the links and joints of the robot, which is essential for accurate control and motion planning.

Table 1. D-H Parameters for the 5-DOF SCARA Robot

DOF-n	Rotation Z (α_n)	Distance Z (α_n)	Distance X (d_n)	Rotation X (θ_n)
0	θ_0	0	l_0	0
1	0	Z	0	0
2	θ_1	0	l_1	0
3	θ_2	0	l_2	0
4	θ_3	0	0	0

(Note: DOF is counted from 0, so for 5 DOF, it starts from DOF 0 to DOF 4.)

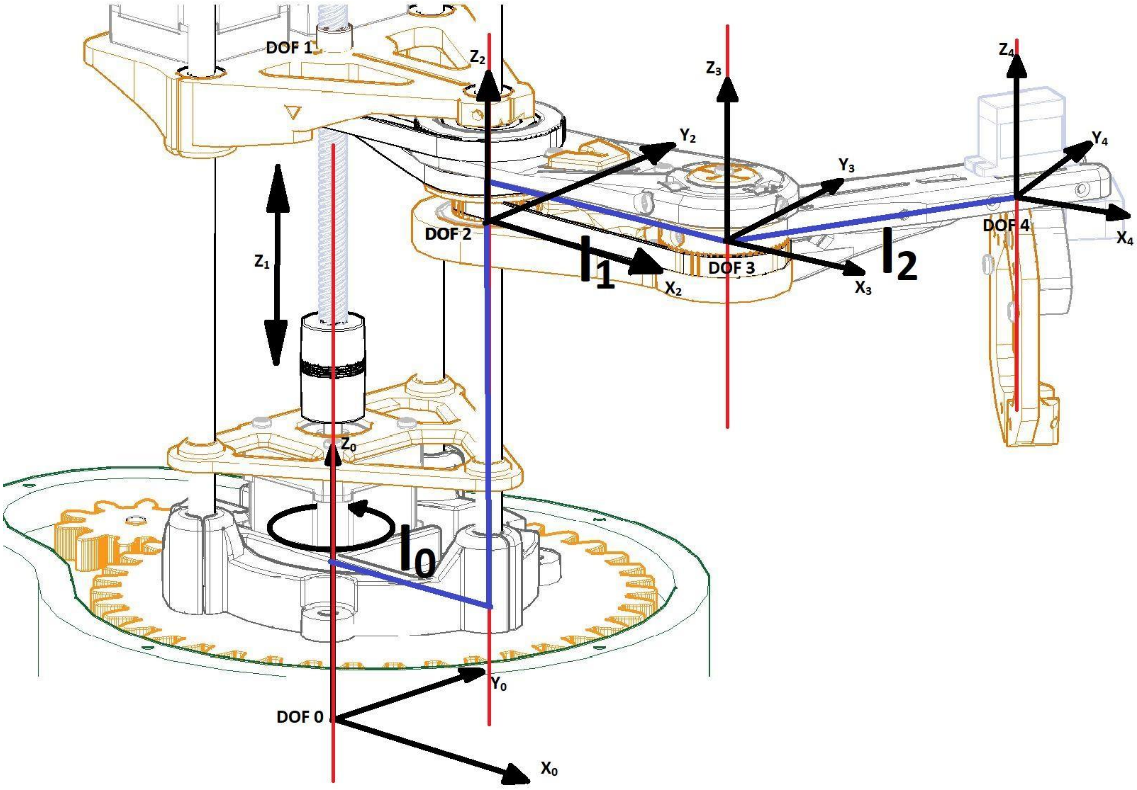

Figure 6. Degrees of Freedom and Local Coordinate System for Each Link and Joint on the 5-DOF SCARA Robot Arm.

1. Forward Kinematics

In forward kinematics, the changes in joint angles are given to the robot, which then moves to a specific coordinate point (x, y, z) based on the transformation matrix provided. Mathematically, it can be written as:

$$(x, y, z) = f(\theta_n, l_n) \tag{1}$$

Based on the Denavit-Hartenberg parameters in Table 1, the primary determinants of forward kinematic motion are θ_0, θ_1, and θ_2, while the angle θ_4 only determines the orientation of the end effector, which can be ignored for simplicity. Translational movement occurs only at DOF 1 along the Z-axis. The free-body diagram for the 5-DOF SCARA robot can be seen in the following Figure 7.

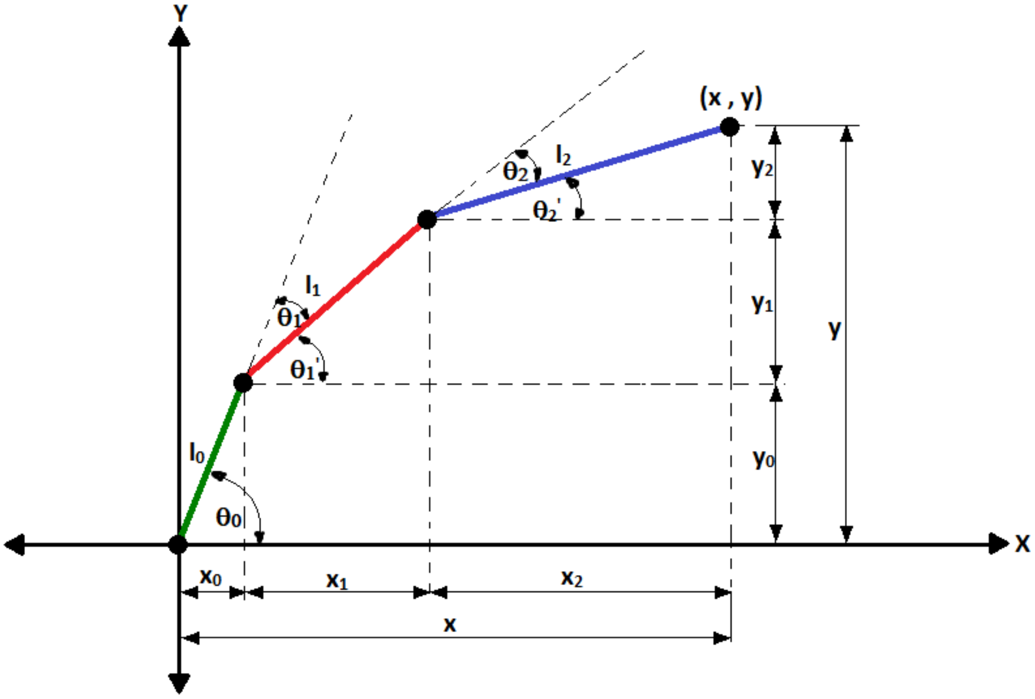

Figure 7. Free-body diagram for a robot with 3 links.

From Figure 7, obtained:

$$x = x_0 + x_1 + x_2 \tag{2}$$
$$y = y_0 + y_1 + y_2 \tag{3}$$

This can be expanded as:

$$x = l_0.cos(\theta_0) + l_1.cos(\theta_1^{'}) + l_2.cos(\theta_2^{'}) \tag{4}$$
$$y = l_0.sin(\theta_0) + l_1.sin(\theta_1^{'}) + l_2.sin(\theta_2^{'}) \tag{5}$$

The angles used for the robot arm are not $\theta_1^{'}$ and $\theta_2^{'}$, but rather θ_1 and θ_2, and the relationship between these angles is:

$$\theta_1^{'} = \theta_0 - \theta_1 \;\; and \;\; \theta_2^{'} = \theta_0 - \theta_1 - \theta_2 \tag{6}$$

Thus, equations 4 and 5 can be rewritten as:

$$x = l_0.cos(\theta_0) + l_1.cos(\theta_0 - \theta_1) + l_2.cos(\theta_0 - \theta_1 - \theta_2) \tag{7}$$
$$y = l_0.sin(\theta_0) + l_1.sin(\theta_0 - \theta_1) + l_2.sin(\theta_0 - \theta_1 - \theta_2) \tag{8}$$

In a SCARA robot, movement along the Z-axis occurs through translation. Therefore, the Z-axis movement can be added to complete equations 7 and 8:

$$z = n\,.\,z_0 \tag{9}$$

The forward kinematic transformation matrix for a 3-link SCARA robot, as shown in matrix below, can be arranged as follows:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos(\theta_0) & \cos(\theta_0 - \theta_1) & \cos(\theta_0 - \theta_1 - \theta_2) & 0 \\ \sin(\theta_0) & \sin(\theta_0 - \theta_1) & \sin(\theta_0 - \theta_1 - \theta_2) & 0 \\ 0 & 0 & 0 & n \end{bmatrix} \bullet \begin{bmatrix} l_0 \\ l_1 \\ l_3 \\ z_o \end{bmatrix} \qquad (10)$$

This is the forward kinematic transformation matrix for a 3-link SCARA robot, where l_0, l_1, and l_2 are the lengths of the links connecting the robot's joints. These link lengths are constant. The coordinates (x, y, z) represent the end-effector's position in space. By inputting the angle values for each robot arm joint, the robot can move to the specified (x, y, z) position.

2. Inverse Kinematics

In inverse kinematics, the target coordinates (x, y, z) are known. To move the robot arm to the (x, y, z) point, the joint angles $(\theta_0, \theta_1, and\ \theta_2)$ must be calculated based on the (x, y, z) coordinates. Mathematically, it can be written as:

$$(\theta_n, Z) = f(x, y, z) \qquad (11)$$

Using basic trigonometric equations, the three joint angles can be calculated. However, for robot arms with multiple segments and joints, calculations using only trigonometric methods can yield multiple solutions. Two approaches to inverse kinematics can be used.

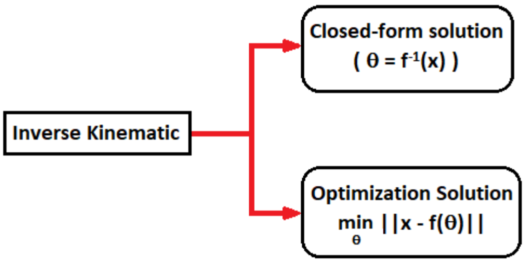

Figure 8. Two inverse kinematic solutions

The closed-form method is reliable for systems with no more than two segments. If there are more than two segments, the Optimization Solution method is a more appropriate choice. Another option is to impose boundary conditions on specific angles to reduce the number of segments in the calculations, though this reduces the robot's motion flexibility.

For a SCARA robot with 3 segments (Base, Inner Arm, and Outer Arm), this process involves calculating the angles for each joint to position the robot's end effector at the target coordinates. The free-body diagram for a 3-segment SCARA robot arm, shown in the following Figure 9, visually represents the relationship between the segments and the forces acting on them. This diagram is a critical step in deriving the mathematical equations that will allow us to calculate the necessary joint angles ($\theta_0, \theta_1, and\ \theta_2$) based on the robot's desired position in the workspace.

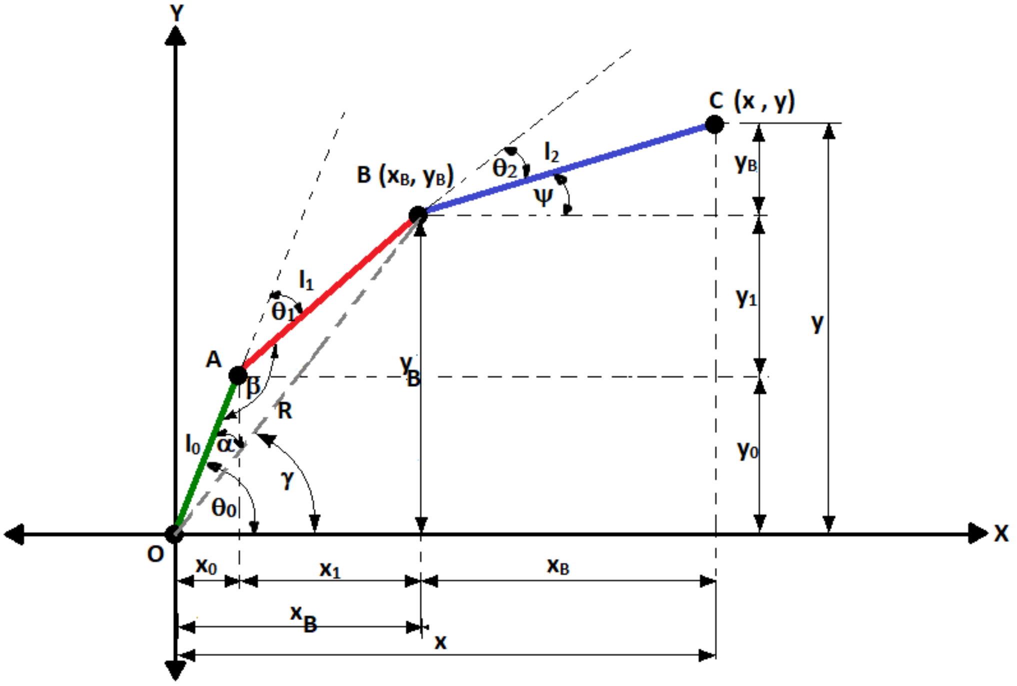

Figure 9. Free-body diagram for a 3-segment SCARA robot

The forward kinematic transformation matrix for Figure 9 was presented in Equations 7 and 8. Consequently, the inverse kinematic equations can be derived and written as follows:

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \end{bmatrix} = f \begin{bmatrix} x \\ y \end{bmatrix} \qquad (12)$$

For a 3-segment SCARA robot, there are 3 control angles ($\theta_0, \theta_1, and\ \theta_2$), but only 2 data points (x and y) are known. Thus, solving the inverse kinematic equations cannot yield a unique solution, resulting in multiple possible solutions. To simplify the inverse kinematic equations for a 3-segment robot, the third segment (link) can be reduced by finding the coordinate point B (x_B, y_B), and then continuing with the inverse kinematics for the remaining 2 segments ($l_0\ and\ l_1$).

The coordinate point B (xB, yB) can be calculated as:

$$x_B = x' - l_2.cos(\psi) \qquad (13)$$

$$y_B = y' - l_2.sin(\psi) \qquad (14)$$

Using trigonometric calculations, two solutions are obtained as follows:

First Solution:

$$\theta_0 = \boldsymbol{cos^{-1}}\left(\frac{\boldsymbol{l_0^2 + x_B^2 + y_B^2 - l_1^2}}{\boldsymbol{2.l_0.(\sqrt{x_B^2 + y_B^2})}}\right) + \boldsymbol{tan^{-1}}\left(\frac{\boldsymbol{y_B}}{\boldsymbol{x_B}}\right) \quad (15)$$

And

$$\theta_1 = \pi - cos^{-1}\left(\frac{l_0^2 + l_1^2 - \boldsymbol{x_B^2} - \boldsymbol{y_B^2}}{2.l_0.l_1}\right) \quad (16)$$

Second Solution:

$$\theta_0 = \boldsymbol{cos^{-1}}\left(\frac{\boldsymbol{l_0^2 + x_B^2 + y_B^2 - l_1^2}}{\boldsymbol{2.l_0.(\sqrt{x_B^2 + y_B^2})}}\right) + \boldsymbol{tan^{-1}}\left(\frac{\boldsymbol{y_B^2}}{\boldsymbol{x_B^2}}\right) \quad (17)$$

And

$$\theta_1 = cos^{-1}\left(\frac{\boldsymbol{x_B^2} + \boldsymbol{y_B^2} - l_0^2 - l_1^2}{2.l_0.l_1}\right) \quad (18)$$

The first and second inverse kinematic solutions for a 3-segment SCARA robot yield two equations with three unknown variables. The best final solution can be obtained using a numerical method approach.

3. SCARA Robot Test Results

In this section, the test results for the SCARA robot arm are presented, focusing on the evaluation of its arm movements and the validation of the inverse kinematics algorithm. The primary goal of these tests was to ensure accurate performance across the 3 degrees of freedom that control the arm's movements. Each segment of the arm was individually tested to verify whether the movements matched the given commands. These tests are crucial to confirm that the mechanical design and control system work effectively, ensuring precise execution of the specified tasks.

The inverse kinematics algorithm, essential for calculating the joint angles needed to position the robot's end effector at a specific target, was also tested. The algorithm was validated by moving the arm to a series of predefined target points and comparing the calculated angles with the actual positions achieved. The results demonstrate the effectiveness of the inverse kinematics solution in guiding the robot's arm to its intended positions, ensuring reliable and precise movement for complex tasks.

Arm Movements

At this stage, the testing focused on evaluating the movement of the two robotic arms, specifically the Inner Arm and Outer Arm, along with assessing the accuracy of the Z-Axis. Following the calibration process, several key details were identified. For the Inner Arm, every 100 steps resulted in a movement of 50 degrees, which corresponds to a scalability factor of 2 steps per degree. The Outer Arm, on the other hand, moved 31 degrees for every 100 steps, yielding a scalability factor of 3.2 steps per degree. Lastly, the Z-Axis showed that every 200 steps led to a displacement of 8 cm, translating to a scalability factor of 25 steps per cm. These results provide a clear understanding of the relationship between step inputs and movements across different components of the robot.

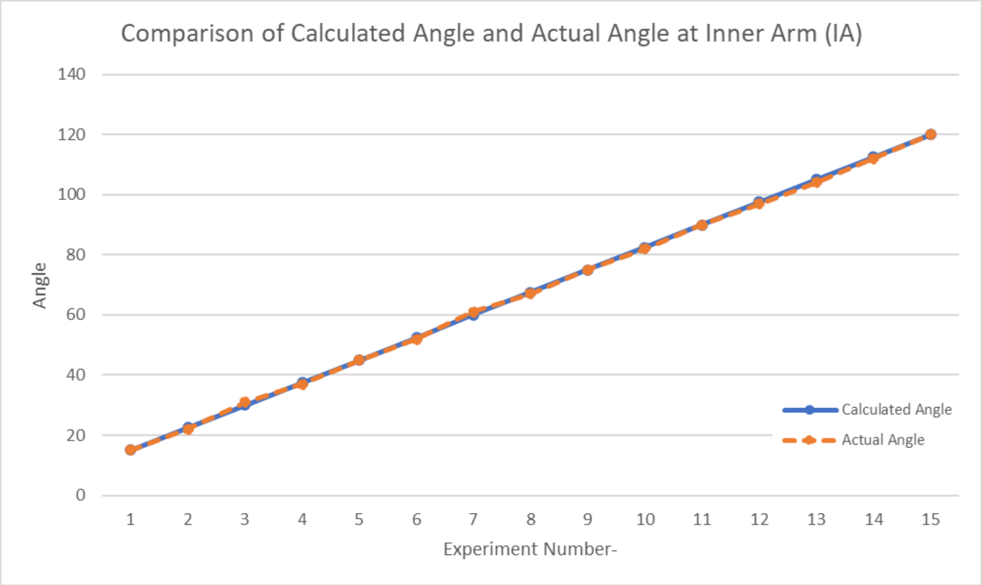

Figure 9. Comparison of Calculated Angle and Actual Angle at Inner Arm (IA)

The initial testing phase focused on evaluating the performance of the Inner Arm, with a total of fifteen trials conducted, each using incremental step increases of 15 to thoroughly assess the consistency and accuracy of the results. These trials were designed to rigorously verify the precision of the arm's movement in response to specific input commands. The data from this testing, which can be viewed in Figure 9 and Table 2, demonstrate that the Inner Arm operated with a high degree of accuracy. Across all trials, the error margin for each test remained below 5%, indicating a strong correlation between the input and the arm's actual movement. Furthermore, the overall average error across all tests was less than 1%, reflecting the reliability and precision of the arm's control system. For a more comprehensive breakdown of the test results, refer to Table 2, which details each trial's input, output, and corresponding error values.

Table 2. Comparison of Calculated Angle and Actual Angle at Inner Arm (IA)

Experiment-n	Step	Calculated Angle (°)	Actual Angle (°)	Error (%)
1	30	15	15	0,00
2	45	22,5	22	2,27

3	60	30	31	3,23
4	75	37,5	37	1,35
5	90	45	45	0,00
6	105	52,5	52	0,96
7	120	60	61	1,64
8	135	67,5	67	0,75
9	150	75	75	0,00
10	165	82,5	82	0,61
11	180	90	90	0,00
12	195	97,5	97	0,52
13	210	105	104	0,96
14	225	112,5	112	0,45
15	240	120	120	0,00
	The average error			**0,85**

The next phase of testing concentrated on evaluating the performance of the Outer Arm. Similar to the Inner Arm trials, fifteen tests were conducted with incremental step increases of 15 to ensure consistent and reliable results. These tests were aimed at validating the precision of the Outer Arm's movements in relation to the given input commands. The outcomes, presented in Figure 10 and Table 3, indicate that the Outer Arm performed with impressive accuracy. In each individual trial, the margin of error remained below 5%, showcasing a strong alignment between the input and the resulting arm movements. Additionally, the overall average error across all tests was calculated at just 0.97%, underscoring the effectiveness of the control system in maintaining high precision. For a detailed account of each trial, including input values, output results, and error percentages, refer to Table 3.

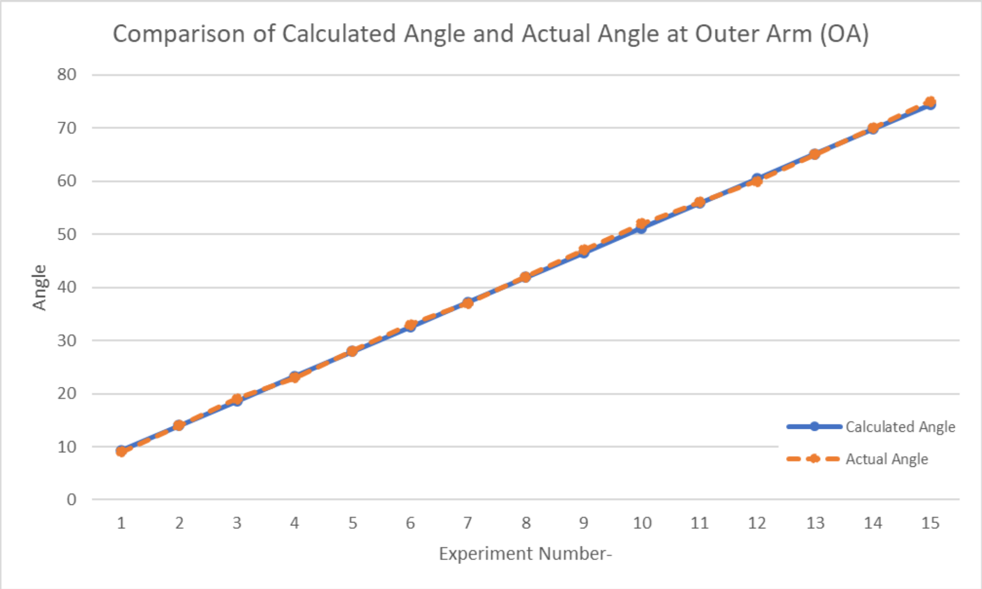

Figure 10. Comparison of Calculated Angle and Actual Angle at Outer Arm (OA)

Table 3. Comparison of Calculated Angle and Actual Angle at Outer Arm (OA)

Experiment-n	Step	Calculated Angle (°)	Actual Angle (°)	Error (%)
1	30	9,3	9	3,33
2	45	13,95	14	0,36
3	60	18,6	19	2,11
4	75	23,25	23	1,09
5	90	27,9	28	0,36
6	105	32,55	33	1,36
7	120	37,2	37	0,54
8	135	41,85	42	0,36
9	150	46,5	47	1,06
10	165	51,15	52	1,63
11	180	55,8	56	0,36
12	195	60,45	60	0,75
13	210	65,1	65	0,15
14	225	69,75	70	0,36
15	240	74,4	75	0,80
The average error				**0,97**

The final set of tests focused on the performance of the Z-Axis. Similar to the previous experiments, fifteen trials were conducted with incremental steps of 15 to evaluate the precision of the vertical movement. The objective of these tests was to ensure that the Z-Axis responded accurately to the input commands. As shown in Figure 11 and Table 4, the results revealed a solid level of precision, with each trial exhibiting an error margin of less than 5%. The overall average error across all the tests was recorded at 3.2%, indicating a slight but manageable deviation in the Z-Axis movement. For a comprehensive overview of the input values, resulting outputs, and corresponding error percentages, please refer to Table 4.

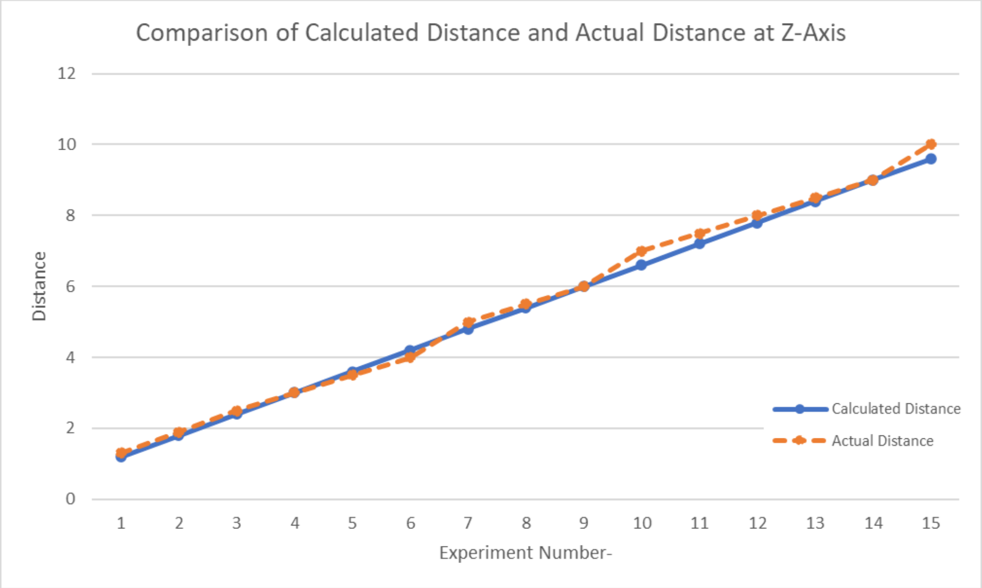

Figure 11. Comparison of Calculated Distance and Actual Distance at Z-Axis

Table 4. Comparison of Calculated Distance and Actual Distance at Z-Axis

Experiment-n	Step	Calculated Distance (cm)	Actual Distance (cm)	Error (%)
1	30	1,2	1,3	7,69
2	45	1,8	1,9	5,26
3	60	2,4	2,5	4,00
4	75	3	3	0,00
5	90	3,6	3,5	2,86
6	105	4,2	4	5,00
7	120	4,8	5	4,00
8	135	5,4	5,5	1,82
9	150	6	6	0,00
10	165	6,6	7	5,71
11	180	7,2	7,5	4,00
12	195	7,8	8	2,50
13	210	8,4	8,5	1,18
14	225	9	9	0,00
15	240	9,6	10	4,00
The average error				**3,20**

The experimental results for the Inner Arm, Outer Arm, and Z-Axis movements demonstrated distinct levels of precision across the three components of the SCARA robot. The Inner Arm, with an average error rate of less than 1%, exhibited the highest level of accuracy, confirming its highly reliable performance in translating step inputs into precise angular movements. Similarly, the Outer Arm performed exceptionally well, maintaining an average error rate of 0.97%, which is comparable to the Inner Arm, further validating the consistency of movement in the robot's primary arms.

In contrast, the Z-Axis displayed a slightly higher error rate, averaging at 3.2%. While this is still within an acceptable range for industrial and educational purposes, it reflects a marginally lower level of precision in vertical movements compared to the horizontal arm movements. This discrepancy may be due to differences in the mechanical structure or the scaling factor of the Z-Axis compared to the arms, which focus more on rotational motion.

Inverse Kinematics Algorithm

The next phase of testing focuses on the performance of the inverse kinematics algorithm, which is critical in ensuring accurate robotic movement. This phase follows a similar approach to the earlier movement tests, with a total of fifteen trials conducted. In each trial, the input coordinates (X and Y) are compared against the actual coordinates (X and Y) achieved by the robot after executing the algorithm. The purpose of these tests is to evaluate the accuracy of the robot's movements based on the provided input, with deviations between the input and output coordinates calculated as error percentages. By analyzing these errors across all trials, the average error rate is determined, offering a clear indication of the algorithm’s effectiveness in translating input coordinates into precise robotic actions.

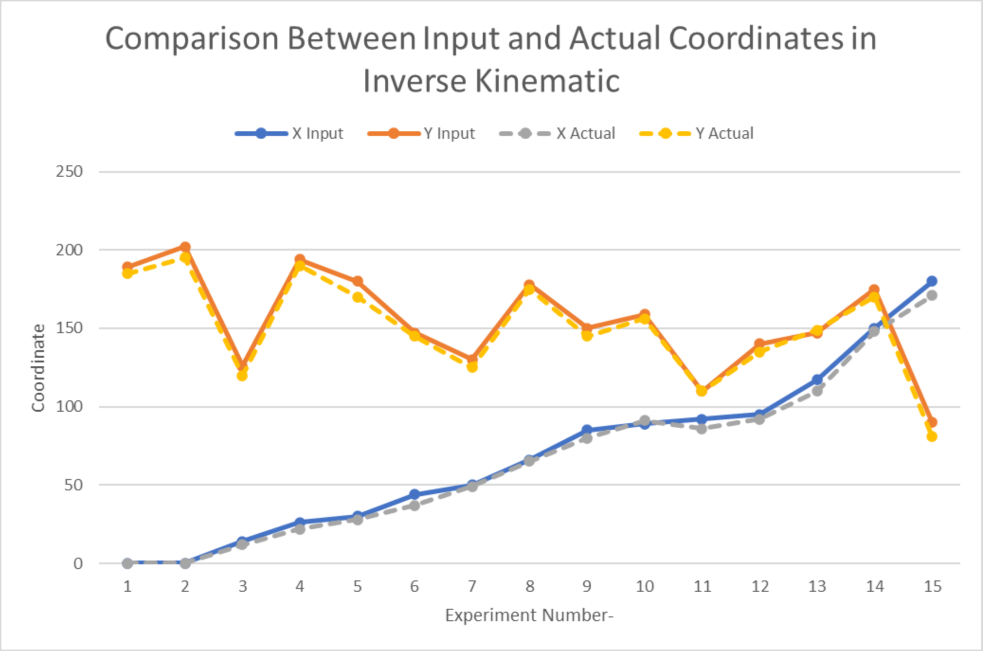

Figure 12. Comparison Between Input and Actual Coordinates in Inverse Kinematic

On average, the X-axis error was 6.41%, while the Y-axis had a lower average error of 3.35%. This indicates that the algorithm performed better in positioning the robot along the Y-axis, while adjustments to improve precision on the X-axis may be necessary. Overall, the testing highlights the strengths and limitations of the inverse kinematics algorithm, providing key insights for further optimization, with more detailed results available in Table 5 and Figure 12.

Table 5. Comparison Between Input and Actual Coordinates in Inverse Kinematic

Experiment-n	**Input Coordinates**		**Actual Coordinates**		**Error (%)**	
	X	**Y**	**X**	**Y**	**X**	**Y**
1	0	189	0	185	0,00	2,16
2	0	202	0	195	0,00	3,59
3	14	126	12	120	16,67	5,00
4	26	194	22	190	18,18	2,11
5	30	180	28	170	7,14	5,88
6	44	147	37	145	18,92	1,38
7	50	130	49	125	2,04	4,00
8	66	178	65	175	1,54	1,71
9	85	150	80	145	6,25	3,45
10	89	159	91	156	2,20	1,92
11	92	110	86	110	6,98	0,00
12	95	140	92	135	3,26	3,70
13	117	147	110	149	6,36	1,34
14	150	175	148	170	1,35	2,94
15	180	90	171	81	5,26	11,11
The average error					**6,41**	**3,35**

In more detail, the error observed in the X-axis across the fifteen trials suggests a notable variability in the system's precision. The error peaked at 18.92% in the 6th trial, where the input coordinate was *X*=44, but the actual result significantly deviated from the expected value. This sharp increase in error indicates potential challenges in the robot's ability to consistently align with the X-axis at certain positions, especially as the distance from the origin increases. In contrast, trials where the input coordinate was closer to zero, such as the 1st and 2nd experiments, yielded no significant error in the X-axis, highlighting that the system is more accurate in close-range movements. As the X input values increased beyond 50, the error generally grew, indicating a trend where larger distances introduce greater inaccuracies.

The Y-axis, while still showing some deviations, exhibited a smaller range of error compared to the X-axis. The largest error was recorded in the final trial (15th), where the input was *Y*=90 but the actual coordinate deviated by over 11%. Despite this outlier, the overall performance in positioning along the Y-axis was more stable, with most trials maintaining errors below 5%. Interestingly, the first few experiments (1st to 4th) showed a consistent error reduction in the Y-axis, which may indicate the system's tendency to better adjust as it moves vertically within a moderate range of values. The more stable performance in the Y-axis suggests that the inverse kinematics algorithm is more adept at handling movements along this axis, perhaps due to factors like system calibration or the nature of the robot's mechanics.

The comparison between the two axes reveals that the inverse kinematics algorithm struggles more with X-axis precision than with the Y-axis. The average error of 6.41% in the X-axis compared to 3.35% in the Y-axis further reinforces this conclusion. This disparity could be attributed to multiple factors, such as mechanical limitations, sensor inaccuracies, or the complexity of the kinematic equations governing movement along the X-axis. Given that precise positioning is crucial for robotic tasks, the higher error in the X-axis highlights a potential area for refinement. Future improvements might involve recalibrating the system or adjusting the algorithm to better handle the X-axis's specific requirements, especially at higher input values where the error trend appears more pronounced.

E. Conclusion

This research has made a substantial contribution to advancing robotics within the CSL Laboratory at the School of Applied STEM, Universitas Prasetiya Mulya. The primary objective of this study—to design and develop a self-manufactured SCARA robot arm with 5 degrees of freedom for educational purposes—was successfully achieved. This SCARA robot serves as a practical tool for enhancing the robotics education experience, providing students with valuable hands-on learning in areas such as kinematics, control systems, and programming. By integrating this robot into the laboratory environment, students gain critical experience with robotics technology that is highly applicable in modern industry, particularly in sectors requiring precision and high-speed operations.

The experimental results demonstrated the robot's capability to perform precise movements, with error rates consistently within acceptable ranges, particularly in the inverse kinematics tests where both X and Y coordinates were tested. This level of precision aligns well with the goal of providing an accurate and reliable platform for students to explore and experiment with industrial robotics concepts. Additionally, the use of locally sourced materials and an open-source control system (Processing Java and Arduino C) not only makes the system cost-effective but also ensures flexibility for further development. This flexibility is crucial for fostering innovation and experimentation, allowing students and researchers alike to continue pushing the boundaries of robotics technology.

Overall, the outcomes of this research go beyond the initial objectives. While it successfully developed a robust SCARA robot arm, it also opened up opportunities for future exploration, particularly in the realm of robotics education. The platform's flexibility and low-cost design could serve as a foundation for developing more affordable robotic systems, making advanced robotics more accessible in educational settings. Moreover, the successful demonstration of this system highlights the potential for broader applications, both in the classroom and in real-world industrial environments. Through this work, the foundation for future innovation in robotics education has been laid, contributing to both the field of robotics and the broader goal of equipping students with industry-relevant skills.

F. Acknowledgment

This research is supported and funded by the Internal Research Grant of STEM School, Universitas Prasetiya Mulya Number 0/4/11.02.1/1016/07/2023 Dated July 26, 2023 concerning "*Pengangkatan Tim Penelitian Penerima Hibah Internal Penelitian Periode 2 Tahun 2023 Sekolah STEM Terapan*".

G. References

[1] W. Grega, "Hardware-in-the-loop simulation and its application in control education," in *FIE'99 Frontiers in Education. 29th Annual Frontiers in Education Conference. Designing the Future of Science and Engineering Education. Conference Proceedings (IEEE Cat. No. 99CH37011*, IEEE, 1999, pp. 12B6-7.

[2] R. Pranata, "Stem Education in Science Learning: Systematic Literature Review," *Jurnal Penelitian Pendidikan IPA*, vol. 9, no. 8, pp. 424–431, 2023.

[3] R. W. Bybee, "What is STEM education?," 2010, *American Association for the Advancement of Science*.

[4] M. Shibata, K. Demura, S. Hirai, and A. Matsumoto, "Comparative study of robotics curricula," *IEEE Transactions on education*, vol. 64, no. 3, pp. 283–291, 2020.

[5] J. C. Perrenet, P. A. J. Bouhuijs, and J. G. M. M. Smits, "The suitability of problem-based learning for engineering education: theory and practice," *Teaching in higher education*, vol. 5, no. 3, pp. 345–358, 2000.

[6] A. Yadav, G. M. Shaver, and P. Meckl, "Lessons learned: Implementing the case teaching method in a mechanical engineering course," *Journal of Engineering Education*, vol. 99, no. 1, pp. 55–69, 2010.

[7] J. Zhu, R. Liu, Q. Liu, T. Zheng, and Z. Zhang, "Engineering students' epistemological thinking in the context of project-based learning," *Ieee transactions on education*, vol. 62, no. 3, pp. 188–198, 2019.

[8] V. Giurgiutiu, J. Lyons, D. Rocheleau, and W. Liu, “Mechatronics/microcontroller education for mechanical engineering students at the University of South Carolina,” *Mechatronics*, vol. 15, no. 9, pp. 1025–1036, 2005.

[9] A. D’Ausilio, “Arduino: A low-cost multipurpose lab equipment,” *Behav Res Methods*, vol. 44, pp. 305–313, 2012.

[10] Y. Ueyama, T. Kubo, and M. Shibata, “Robotic hip-disarticulation prosthesis: evaluation of prosthetic gaits in a non-amputee individual,” *Advanced Robotics*, vol. 34, no. 1, pp. 37–44, 2020.

[11] M. Garduño-Aparicio, J. Rodríguez-Reséndiz, G. Macias-Bobadilla, and S. Thenozhi, “A multidisciplinary industrial robot approach for teaching mechatronics-related courses,” *IEEE Transactions on Education*, vol. 61, no. 1, pp. 55–62, 2017.

[12] C. Kim, D. Kim, J. Yuan, R. B. Hill, P. Doshi, and C. N. Thai, “Robotics to promote elementary education pre-service teachers’ STEM engagement, learning, and teaching,” *Comput Educ*, vol. 91, pp. 14–31, 2015.

[13] A. Gasparetto and L. Scalera, “From the unimate to the delta robot: the early decades of industrial robotics,” in *Explorations in the History and Heritage of Machines and Mechanisms: Proceedings of the 2018 HMM IFToMM Symposium on History of Machines and Mechanisms*, Springer, 2019, pp. 284–295.

[14] S. H. Tay, W. H. Choong, and H. P. Yoong, “A Review of SCARA Robot Control System,” in *2022 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, IEEE, 2022, pp. 1–6.

[15] M. A. González-Palacios, M. A. Garcia-Murillo, and M. González-Dávila, “A novel tool to optimize the performance of SCARA robots used in pick and place operations,” *Journal of Mechanical Science and Technology*, vol. 35, pp. 4715–4726, 2021.

[16] M. Shariatee, A. Akbarzadeh, A. Mousavi, and S. Alimardani, “Design of an economical SCARA robot for industrial applications,” in *2014 Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*, 2014, pp. 534–539. doi: 10.1109/ICRoM.2014.6990957.

[17] W. S. Barbosa *et al.*, “Industry 4.0: examples of the use of the robotic arm for digital manufacturing processes,” *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 14, pp. 1569–1575, 2020.

[18] Y. He *et al.*, “Dynamic modeling, simulation, and experimental verification of a wafer handling SCARA robot with decoupling servo control,” *IEEE Access*, vol. 7, pp. 47143–47153, 2019.

[19] D. A. Jimenez-Nixon, M. C. Paredes-Sánchez, and A. M. Reyes-Duke, “Design, construction and control of a SCARA robot prototype with 5 DOF,” in *2022 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT)*, IEEE, 2022, pp. 1–6.

[20] S. Suri, A. Jain, N. Verma, and N. Prasertpoj, “SCARA industrial automation robot,” in *2018 international conference on power energy, environment and intelligent control (PEEIC)*, IEEE, 2018, pp. 173–177.