Journal of Robotics and Control (JRC)
Volume 4, Issue 6, 2023
ISSN: 2715-5072, DOI: 10.18196/jrc.v4i6.20087 

# Handling Four DOF Robot to Move Objects Based on Color and Weight using Fuzzy Logic Control

Emmanuel Agung Nugroho [1], Joga Dharma Setiawan [2], M. Munadi [3]
[1] Mechatronics Department, Indorama Engineering Polytechnic, Purwakarta, Indonesia
[1, 2, 3] Mechanical Engineering Department, Diponegoro University, Semarang, Indonesia
Email: [1] emmanuelagung78@gmail.com, [2] joga.setiawan@gmail.com, [3] munadi@ft.undip.ac.id

***Abstract*—Manipulators are increasingly used in industry to improve efficiency in jobs that require precision, long duration, and repetitive work. This research was conducted on a laboratory scale to control manipulators on a pick-and-place system in the product storage and packing area. The object of this research is a four-degree-of-freedom (4-DOF) manipulator controlled using a fuzzy logic system. The hardware used is a conveyor machine to model the product delivery process, Dobot Magician as a 4-DOF manipulator, HX711 load cell serves as a weight sensor, TCS-3200 serves as a color sensor, and Arduino Mega 2560 as a controller. The software used is Dobot Studio as the main program to control the movement of the robot and Matlab to develop the Fuzzy Logic Control (FLC) function, which is embedded in the Arduino. Fuzzy logic control processes weight variables and color variables read by sensors as information data to control the movement of the manipulator. The results showed that the manipulator was able to pick up and place objects according to the path-planning coordinates. The testing data states that the precision and accuracy of the average coordinates of product pick and place against the path planning has an error deviation of 1.8%.**

***Keywords—Manipulator 4 Degree of Freedom; Fuzzy Logic Control; Path Planning; Arduino Mega 2560; Loadcell HX711; Color Sensor TCS3200.***

## I. INTRODUCTION

In recent decades, there has been a significant surge in the use of robots for industrial automation. This trend is a response to the challenges faced by industries to improve their production efficiency, accuracy, speed and capacity. The integration of robotics technology into industrial processes has enabled industries to address these issues effectively, leading to increased productivity and better results [1]–[4]. In compliance with the demand for smart and efficient industrial machines, researchers are conducting extensive studies in the area of robotics. The goal is to augment the intelligence of robots and enhance their operational capacities [5]–[8]. This research was conducted on a laboratory scale using the Dobot Magician manipulator as a model for the application of industrial robots in pick and place applications. This research aims to prove the application of fuzzy logic control on manipulators for pick and place functions with input parameters influenced by weight variables and color variables [9]–[12].

## II. RELATED WORK

To realize this research, a study of related matters is carried out, including the specifications of the Dobot Magician as the main model in this study, the kinematics of the Dobot Magician, the Denafit Hartenberg method to express the relationship between Dobot Magician joints and the Fuzzy logic control system.

### A. Dobot Magician

The Dobot Magician is a robotic arm that has four degrees of freedom (DOF). It is made up of a base, a shoulder, an elbow, and a wrist. The base, shoulder, and elbow are powered by a stepper motor, while the wrist is powered by a servo motor. The Dobot Magician's end Effector is compatible with a variety of attachments, such as suction cups, grippers, laser printers, 3D printing hot-ends, and pen holders for drawing graphs [12]–[15]. Table I displays the impressive specifications of the Dobot Magician.

TABLE I. SPECIFICATIONS OF THE DOBOT MAGICIAN [13]

| Parameter | Specifications |
|---|---|
| Number of Axes | 4 (Base, Shoulder, Elbow Wrist) |
| Payload | 500 gram |
| Max. Reach | 320 mm |
| Position Repeatability | 0.2 mm |
| Communication | USB/Wifi/Bluetooth |
| Power supply | 100V – 240 V; 50/60 Hz |
| Power In | 12V / 7A DC |
| Power Consumtion | 60 watt max |
| Working Temperature | -10ºC – 60º C |

The Dobot Magician has a structure diagram that consists of four joints, known as $J_1$, $J_2$, $J_3$, and $J_4$. The initial balance point is at $J_2$, with coordinates of (0, 0, 0). The movement position is defined as (x, y, z). The Dobot Magician comes equipped with a power supply and active indicator lights. The manipulator's forearm length is 135 mm from the base, rear arm length is 147 mm, front arm length is 160 mm, and end effector length is 72 mm. Fig. 1 displays the mechanical structure diagram of Dobot Magician.

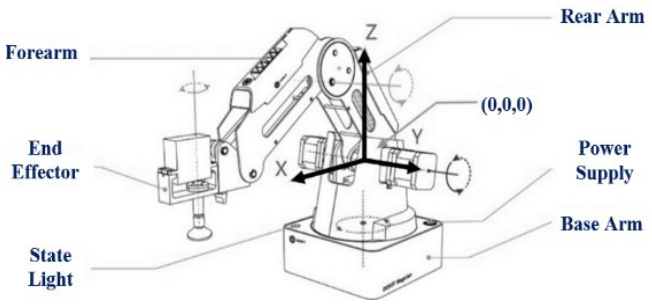


Fig. 1. Mechanical structure of Dobot Magician [2]

Fig. 1 shows that the Dobot Magician operates on a mechanical structure that enables it to move rotationally. The rotation of the drive motors at $J_1$, $J_2$, $J_3$, and $J_4$ determines all the robot movements. Additionally, Fig. 2 displays the dimensions and working area of the Dobot Magician, providing a comprehensive understanding of its capabilities.

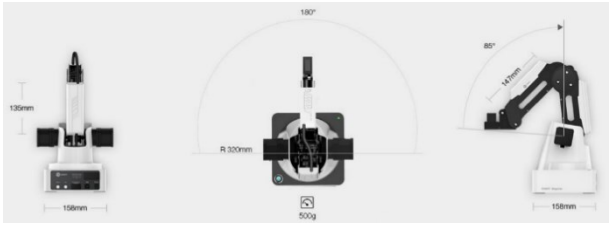

Fig. 2. Dimensions and working area of Dobot Magician [2]

In Fig. 2, the robot's working area range is described to have a radius of 320 mm. The rear arm is capable of movement up to $180^0$, while the base has a rotational range of ± $135^0$. The shoulder joint has a movement radius of up to $85^0$, and the front arm has a movement radius of -$10^0$ to $95^0$. Table II shows the joint axes and movements of each joint.

TABLE II. WORKING AREA AND WORKING SPEED OF DOBOT

| Axis | Range | Maximum speed |
|---|---|---|
| Joint 1 (Base) | -$135^0$ to $135^0$ | $320^0$/s |
| Joint 2 (Rear arm) | $0^0$ to $85^0$ | $320^0$/s |
| Joint 3 (Fore arm) | -$10^0$ to $95^0$ | $320^0$/s |
| Joint 4 (Rotation servo) | $90^0$ to -$90^0$ | $480^0$/s |

### B. Robotic Kinematics

In the working mechanism of a manipulator, two types of analyses are commonly used: kinematics analysis and dynamics analysis [16]–[19]. Kinematics analysis is defined as the study of robot movement in terms of speed, position, and acceleration while ignoring the forces that affect the robot [20], [21]. The Robot kinematics system consists of two types of movement analysis: forward kinematics and inverse kinematics [22]–[27]. Forward kinematics uses the angle of movement as a reference to obtain the position coordinates of the robot's end effector. Forward kinematic roles in determining the position of the end of the effector, planning the trajectory, and controlling joint movement [28]–[30]. Inverse kinematics uses the position coordinate as a reference to obtain the angular value of the robot joint movement [31]–[33]. Inverse kinematics has the role of controlling the position of the end of the effector, joint movement planning, and trajectory control [34].

The robot kinematics system requires a control system as a driving force consisting of input or reference, controller system, and robot mechanical system. The input reference is in the form of position, velocity, or acceleration and is expressed in a vector coordinate of position (P) and orientation (x, y, z). The output is an angle $\theta$ ($\theta_1$, $\theta_2$, $\theta_3$… $\theta_n$) where n is the number of joints on the robot. A control diagram of the robot kinematics system is shown in Fig. 3.

### C. Denavit Hartenberg Method

The Denavit Hartenberg (DH) method is one of the most commonly used methods in determining robot kinematics parameters. DH parameters are used to describe the relationship between each joint on the manipulator. Without a description of the relationship between joints, robot kinematics cannot be implemented. In a four DOF manipulator with revolute joints $\theta_1$, $\theta_2$, $\theta_3$, $\theta_4$, and $\theta_5$ each connected by a link, it gives a kinematics picture that the manipulator has four links (link 1, link 2, link 3, and link 4), with the end-effector as an object that can be lifted and placed. [36]–[42]. The DH method is expressed in four parameters: twisting angle $\theta_i$, length of link $L_i$, link offset $d_i$, and the joint angle [43]–[46]. To describe the relationship between joints and links in the 4 DOF manipultor, it can be explained as follows:

1. First Joint ($\theta_1$):
   - $\theta_1$ is the rotation angle along axis $Z_0$ (first joint rotation).
   - $d_1$ is the distance along axis $X_0$ to the first joint (link length 1).
   - $L_1$ is the length of link 1 connecting the first joint to the second joint.
   - $\alpha_1$ is the angle between axis $Z_0$ and axis $Z_1$.
2. Second Joint ($\theta_2$):
   - $\theta_2$ is the rotation angle along axis $Z_1$.
   - $d_2$ is the distance along axis $Z_1$ to the third joint.
   - $L_2$ is the length of link 2 connecting the second joint to the third joint.
   - $\alpha_2$ is the angle between axis $Z_1$ and axis $Z_2$.
3. Third Joint ($\theta_3$):
   - $\theta_3$ is the rotation angle along axis $Z_2$ (third joint rotation).
   - $d_3$ is the distance along axis $Z_2$ to the fourth joint (length of link 3).
   - $L_3$ is the length of link 3 connecting the third joint to the fourth joint.
   - $\alpha_3$ is the angle between axis $Z_2$ and axis $Z_3$.
4. Fourth Joint ($\theta_4$):
   - $\theta_4$ is the rotation angle along axis $Z_3$.
   - $d_4$ is the distance along axis $Z_3$ to the end-effector.
   - $L_4$ is the length of link 4 connecting the fourth joint to the end-effector.
   - $\alpha_4$ is the angle between axis $Z_3$ and axis $Z_4$ (end-effector).

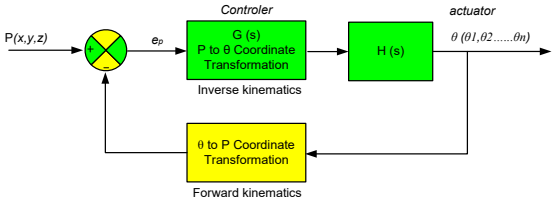


Fig. 3. Control diagram of kinematics robot system [35]

According to this parameter explanation, we can formulate a modification of the DH Dobot Magician parameter to obtain a homogeneous transformation matrix that represents the position and orientation of the goal frame with respect to the Dobot Magician reference frame [47]–[49]. The setting of DH parameters allows the manipulator to pick and place objects with precision in pick and place applications frame. The DH parameter modification on Dobot Magician as intended is shown in Table III [50]–[52].

TABLE III. MODIFIED DH PARAMETERS OF DOBOT MAGICIAN [9]

| Frame i | $L_i$ | $\alpha_i$ | $d_i$ | $\theta_i$ |
|---|---|---|---|---|
| 1 | 0 | 90° | $L_1$ | $\theta_1$ |
| 2 | $L_2$ | 0° | 0 | $\theta_2$ |
| 3 | $L_3$ | 0° | 0 | $\theta_3$ |
| 4 | $L_4$ | 0° | 0 | $\theta_4$ |
| 5 | 0 | 90 | $-L_5$ | $\theta_5 = 0°$ |

*Frame 5 is an optional end off effector attached to wrist frame 4.*

### D. Fuzzy Logic Control

Fuzzy logic control is used to map an input variable into a firm value at the output using a rule-based decision-making process that classifies the degrees of membership of fuzzy sets and fuzzy rules [53]–[56]. The computational process of fuzzy logic can be divided into four main stages, namely:

1. Fuzzification, a process in fuzzy logic systems in which numerical inputs or data that are firm are converted into linguistic variables or fuzzy variables [57], [58].
2. Rule Base contains a set of fuzzy rules that serve to determine decisions, consisting of IF-THEN statements in fuzzy logic format. The statements in these fuzzy rules use connectors such as AND, OR, or NOT, and the degree of truth is calculated for the predicate set [59], [60].
3. A fuzzy inference engine is one of the core components in a fuzzy logic system that is responsible for making decisions based on the rules defined in the fuzzy rule base [60], [61].
4. Defuzziffication, the final stage is defuzzification, which involves returning the fuzzy calculation result (fuzzy set) into a variable that corresponds to its range in the real world. The defuzzifier also uses membership function to map fuzzy set values into real variables. This stage provides explicit information from the fuzzification process to be executed by the program [62].

The use of fuzzy logic in 4 DOF manipulator control research has many advantages, namely:

1. Able to convert input uncertainty into definite logic that allows the manipulator to respond appropriately.
2. Fuzzy logic is able to accommodate many input variables into a certain value making it easier for the manipulator to execute the program.
3. Fuzzy logic output is easily implemented on embedded microcontrollers including the Arduino Mega 2560 [63]–[66].

## III. METHODS AND MATERIALS

### A. Implementation of the Denavit Hartenberg Method on Dobot Magician

The Denavit-Hartenberg (DH) method is an appropriate way to model manipulator link and joint parameters that can be used in kinematics and control calculations. Based on the homogeneous Denavit Hartenberg equations, it is possible to determine the kinematics of the manipulator, control the position and orientation of the end of effector, plan the path for the pick and place task on the manipulator, and interpolate the motion between coordinate points in the pick and place task. Through the Dobot Magician parameters as shown in Table I, Table II, and Table III, the Denavit Hartenberg homogeneous matrix is obtained in the equation (1) to (5).

$$A_1^0 = \begin{bmatrix} C\theta_1 & -S\theta_1 & 0 & L_1C\theta_1 \\ S\theta_1 & C\theta_1 & 0 & L_1S\theta_1 \\ 0 & -1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$A_2^1 = \begin{bmatrix} C\theta_2 & -S\theta_2 & 0 & L_2C\theta_2 \\ S\theta_2 & C\theta_2 & 0 & L_2S\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$A_3^2 = \begin{bmatrix} C\theta_3 & S\theta_3 & 0 & L_3C\theta_3 \\ S\theta_3 & -C\theta_3 & 0 & L_3S\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$A_4^3 = \begin{bmatrix} C\theta_4 & -S\theta_4 & 0 & L_4C\theta_4 \\ S\theta_4 & C\theta_4 & 0 & L_4S\theta_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$A_5^4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -L_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

From equations (1) to (5), we get a homogeneous matrix that connects frame 1 to frame 5 as (6).

$$A_5^0 = A_1^0A_2^1A_3^2A_4^3A_5^4 \quad (6)$$

### B. Forward and inverse kinematics

- *Forward Kinematics*

For a four Degree of Freedom (4-DOF) manipulator, forward kinematics plays a role in determining the position and orientation of the end of effector based on the joint rotation angles of the four manipulator joints. Forward kinematics requires closed-loop vector analysis to obtain position equations. The number of equations for a manipulator is determined by the joint count and arm length [67]–[69]. Based on the homogeneous matrix equation of Dobot Magician shown in equation (6), the forward kinematic equation is obtained as (7).

$$\begin{bmatrix} Px \\ Py \\ Pz \end{bmatrix} = \begin{bmatrix} C\theta_1\,(L_3C(\theta_2+\theta_3) + L_2C\theta_2) \\ S\theta_1\,(L_3C(\theta_2+\theta_3) + L_2C\theta_2) \\ L_1 + L_3S(\theta_2+\theta_3) + L_2S\theta_2 \end{bmatrix} \quad (7)$$

- *Inverse Kinematics*

Inverse kinematics is used to set the joint movement angle ($\theta$) of Dobot Magician. Inverse kinematics on a manipulator is always more complex than forward kinematics because the equations solved are non-linear and produce finite solutions. Inverse kinematics is process to calculate the joint angles required for the manipulator to achieve a certain position and orientation at the end-effector. Based on forward kinematics equation (7), the inverse kinematic parameter approach to obtain $\theta_1$, $\theta_2$, $\theta_3$, and $\theta_4$ is follows.

*1. Approach $\theta_1$*

$P_x = x - L_4$; $P_y = y$ and $P_z = z + L_5$

$$\frac{S\theta_1}{C\theta_1} = \frac{P_y}{P_x}, \tan\theta_1 = \frac{P_y}{P_x} \quad (8)$$

$$\theta_1 = \tan^{-1}\left[\frac{P_y}{P_x}\right] = \tan^{-1}(P_x, P_y) \quad (9)$$

2. *Approach* $\theta_3$

To make it easier to find the cosine equation, it is expressed as (10) to (13).

$$a = \frac{P_x}{C\theta_1} = L_3C(\theta_2 + \theta_3) + L_2C\theta_2 \quad (10)$$

$$b = P_z - L_1 = L_3S(\theta_2 + \theta_3) + L_2S\theta_2 \quad (11)$$

So, we get:

$$a^2 + b^2 = {L_2}^2 + {L_3}^2 + 2L_2L_3C\theta_3 \quad (12)$$

then:

$$C\theta_3 = \frac{a^2 + b^2 - {L_2}^2 + {L_3}^2}{2L_2L_3} \quad (13)$$

By assuming the object is still within the working range of the Dobot then:

$S\theta_3 = \pm\sqrt{1 - Cos^2\theta_3}$

then:

$$\theta_3 = tan^{-1}\left(\frac{Sin\theta_3}{Cos\theta_3}\right) \quad (14)$$

3. *Approach* $\theta_2$

At the same time $\theta_2$ is obtained with the following equation (15) and (16).

$$a = (L_3C\theta_3 + L_2)C\theta_2 - (L_3S\theta_3)S\theta_2 \quad (15)$$

$$b = (L_3S\theta_3 + L_2)S\theta_2 - (L_3S\theta_3)C\theta_2 \quad (16)$$

Equations (17) and (18) can be written in another form:

$a = m\,C\theta_2 - n\,S\theta_2$

$b = m\,S\theta_2 - n\,C\theta_2$

When

$m = L_3C\theta_3 + L_2$ and $n = L_3S\theta_3$,

If $p = \pm\sqrt{m^2 + n^2}$

we get

$$\varphi = tan^{-1}\left(\frac{n}{m}\right) \quad (17)$$

if: $m = p\,Cos\varphi$ and $n = p\,Sin\,\varphi$, then:

$$a = p\,C\varphi C\theta_2 - p\,S\varphi S\theta_2 \quad (18)$$

$$b = p\,C\varphi S\theta_2 - p\,S\varphi C\theta_2 \quad (19)$$

From equations 18 and 19 we obtain

$a = p\,C(\varphi + \theta_2)$

$b = p\,S(\varphi + \theta_2)$

then:

$$\varphi + \theta_2 = tan^{-1}\left(\frac{b}{a}\right) \quad (20)$$

Thus obtaining:

$$\theta_2 = tan^{-1}\left(\frac{b}{a}\right) - tan^{-1}\left(\frac{n}{m}\right) \quad (21)$$

*C. Manipulator Working System Diagram*

This research combines two control systems, namely the Dobot Studio to control the movement of the manipulator, and the Arduino Mega 2560 which serves as a fuzzy program controller [74]–[76]. The combination of dual control is possible on the Dobot Magician manipulator because the device is equipped with a communication interface that facilitates the Dobot can be controlled with additional external controls [69], [77]–[79]. At the output of the system is Dobot Magician, a four DOF manipulator driven by stepper motors at the joint and servo motors at the End of the Effector. In this research, the end of the effector used is a suction cup model with a pneumatic vacuum working principle. The suction cup was chosen because it is most suitable for the surface contours of the object to be moved. The schematic diagram of the manipulator working system for picking up and placing objects based on weight and color is shown in Fig. 4.

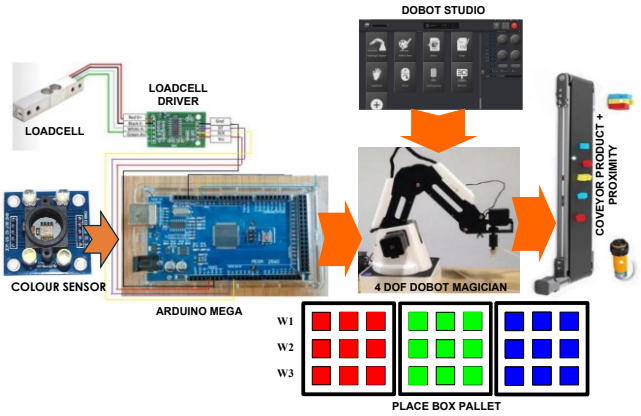


Fig. 4. Block diagram of the control system of the object-moving manipulator

Based on Fig. 4, it can be explained that controlling the movement of the Dobot as an object mover based on the weight and color of the object requires an integrated control system consisting of input, control process, and output. In the input section consists of 3 sensors, namely:

1. HX711 loadcell: is a weight sensor that has a weight capacity of 1.5 kg. In this study, the weight of random objects is 50 g - 750 g [43], [70].
2. Color sensor TCS-3200: capable of detecting 64 color gradations derived from the four main colors of red, clear, green, and blue. For this study, the workpiece is modeled in three dominant colors: red, green, and blue [71], [72].
3. Proximity sensors employ the principle of electromagnetic radiation to detect workpieces without direct contact [73]. In this research, Proximity sensor to read products that come on the conveyor to the Dobot work area.

The control process is integrating Dobot Studio and Arduino Mega 2560 to produce a working mechanism algorithm for weight and color-based pick and place manipulators can be explained in Fig. 5.

Fig. 5 describes the Manipulator's flowchart algorithm as product picking and placement, starting from the conveyor working to placing objects according to the weight and color categories of the object. The working mechanism of the weight- and color-based object-moving manipulator can be explained as:

1. The workpiece represents the production object, which will be randomly placed on the conveyor and delivered until it stops at the pickup point by the proximity sensor.
2. From the initial position the manipulator moves towards the pickup point to pickup the workpiece.
3. The workpiece picked up by the EoE of Dobot Magician will be placed on the color sensor to detect the object's color and then moved to the weight sensor area to weigh the object's weight.
4. EoF will move the object back to the place point based on the color and weight data. Place point is an area to store workpieces based on color and weight criteria.
5. There are 3 groups of pallet baskets based on color, each color group is capable of storing 9 products. Row 1 (W1) stores very light and light products, row 2 (W2) stores medium products and row 3 (W3) stores heavy and very heavy products.

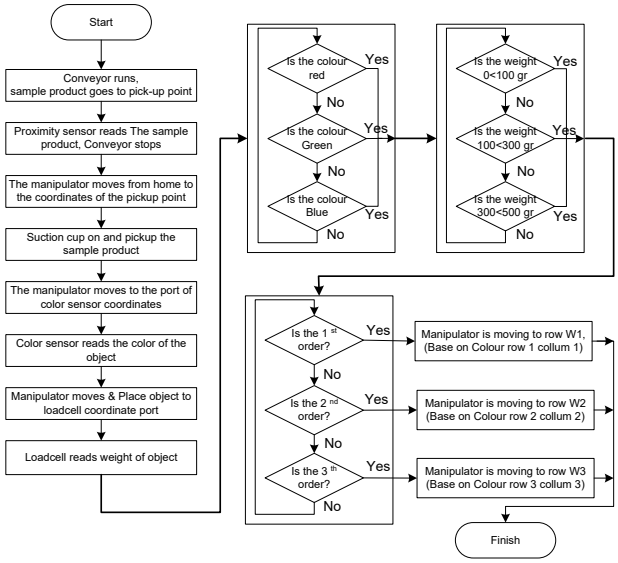


Fig. 5. Manipulator handling system Program Flowchart

### D. *Fuzzy Logic Controller Manipulator*

Fuzzy logic control is implemented with the Mamdani method, this method was chosen to facilitate setting membership values with the consideration that the value of each variable used is not linear. Input variables and output variables in fuzzy systems are needed to emphasize decisions between several uncertainty options of values and conditions. In this system, there are 2 input variables, WEIGHT and COLOR, and 1 output variable PLACE. Below is explanation of the input and output variables in this fuzzy logic control system:

1. Variable WEIGHT: There are 6 weight input variables, each variable has a type and value parameter as follows:
   a. VERYLITE : Trimf type, Params [0 100 200]
   b. LITE : Trimf type, Params [100 200 300]
   c. MEDIUMLITE : Trimf type, Params [200 300 400]
   d. MEDIUMHEAVY : Trimf type, Params [300 400 500]
   e. HEAVY : Trimf type, Params [400 500 600]
   f. VERYHEAVY : Trimf type, Params [500 600 700]
2. Variable COLOR: There are 3 color variables (red, yellow, and blue), each color has a value range of 512 data. Type and value parameters of color variables are as follows:
   a. RED : Trimf type, Params [0 256 512]
   b. GREEN : Trimf type, Params [256 512 768]
   c. BLUE : Trimf type, Params [512 768 1024]
3. Variable PLACE: there are 9 membership functions and 27 membership values representing 27 coordinates of product placement:
   a. RED color Place
      a) R3.0 : Trimf type, Param [-2 1 4]
      b) R3.1 : Trimf type, Param [1 4 6]
      c) R3.2 : Trimf type, Param [6 9 12]
   b. GREEN color Place
      a) G2.0 : Trimf type, Param [11 14 17]
      b) G2.1 : Trimf type, Param [12 15 18]
      c) G2.2 : Trimf type, Param [16 19 22]
   c. BLUE color Place
      a) B1.0 : Trimf type, Param [19 22 25]
      b) B1.1 : Trimf type, Param [22 25 28]
      c) B1.2 : Trimf type, Param [25 28 30]

Based on the type data and parameter values of 2 input variables and 1 output variable, the fuzzy membership function can be calculated with the equation (22) and (23) [80], [81].

$$\mu[x] = \begin{cases} 0 \\ \dfrac{(x-a)}{(b-a)} \\ 1 \end{cases} ; \begin{matrix} x \le a \\ a \le x \le b \\ x \ge b \end{matrix} \quad (22)$$

$$\mu[x] = \begin{cases} \dfrac{(b-x)}{(b-a)} \\ 0 \end{cases} ; \begin{matrix} a \le x \le b \\ x \ge b \end{matrix} \quad (23)$$

Membership functions are very important to provide certainty to the value of each fuzzification process so as to produce definite values in Fuzzy control applications. Based on equations (22) and (23), the membership function for each input variable and output variable in this system can be described. The weight membership function is shown in Fig. 6.

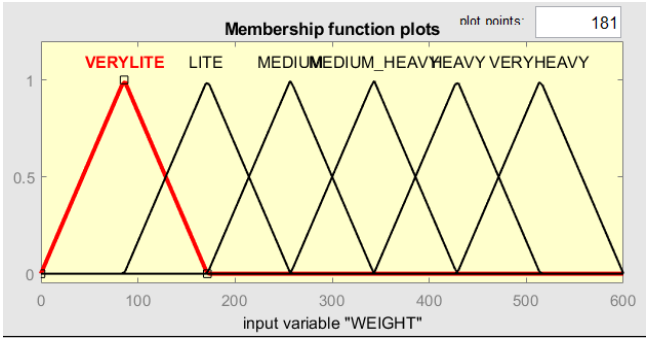


Fig. 6. Weight input variable

Fig. 6 is the result of the membership function for the weight variable as input which consists of 6 membership functions. The weight membership value is 0 - 700 grams and

uses a triangular membership function type. This weight membership function value adjusts the ability to lift weights on Dobot Magician which has a limit of 500 grams.

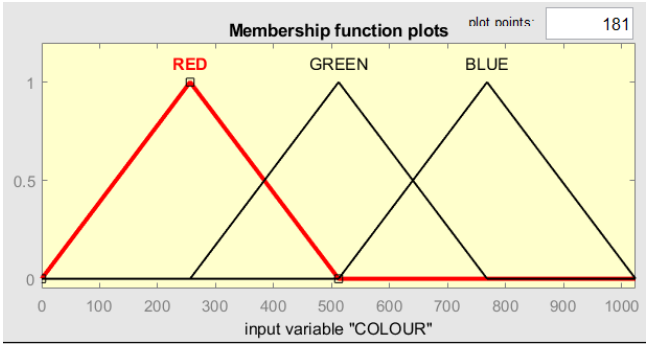


Fig. 7. Color input variable

Fig. 7 shows the color membership function with a membership value range of 0 - 1024 in RGB units. Each color has 512 membership values in the RGB system and uses a triangular membership function type.

In the output section, the membership function displays 3 place variables representing 3 color unit groups and 3 weight unit groups. This program is adapted to real conditions where there are 3 pallets for storing objects based on color and weight categories. The fuzzy graph of the output variables is shown in Fig. 8.

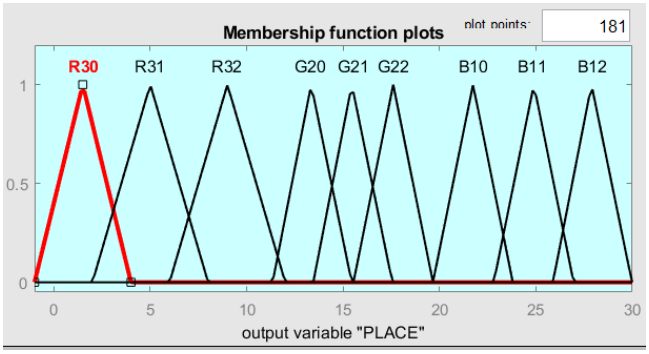


Fig. 8. Output variable “PLACE” on fuzzy system

By using 2 input variables and an output variable, an Aggregation graph can be created, which is the process of combining input variables and output variables through a fuzzy rule base. Fig. 9 shows the Aggregation graph in the pick and place process with the 4 DOF manipulator in this study.

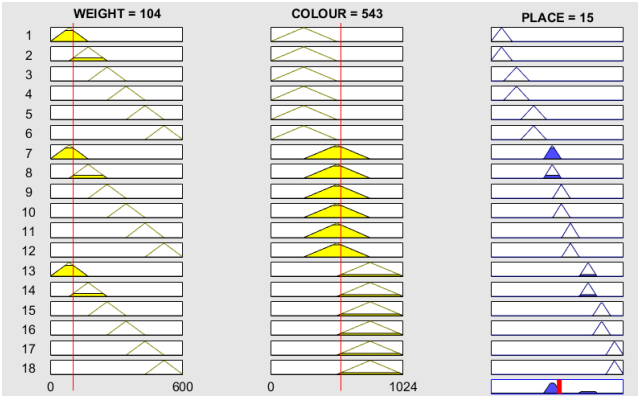


Fig. 9. Pick and Place fuzzifikasi Rules

Fig. 9 is the rule viewer obtained from the Aggregation process using 18 rules to represent 6 WEIGHT membership functions, 3 COLOR membership functions, and 9 PLACE membership functions. The rule editor uses AND connection so the rule base implementation is as follows:

*IF WEIGHT is VERYLITE and COLOR is RED Then OUTPUT PLACE is R30*
*IF WEIGHT is LITE and COLOR is RED Then OUTPUT PLACE is R30*
*Ect.*

The 18 rules representing the values of 2 input variables and an output variable are organized in Table IV.

TABLE IV. INPUTS AND 1 OUTPUT FUZZIFICATION RULES

| NO | INPUT | | OUTPUT |
|---|---|---|---|
| | WEIGHT | COLOR | PLACE |
| 1 | VERYLITE | RED | R30 |
| 2 | LITE | RED | R30 |
| 3 | MEDIUM | RED | R31 |
| 4 | MEDIUM HEAVY | RED | R31 |
| 5 | HEAVY | RED | R32 |
| 6 | VERYHEAVY | RED | R32 |
| 7 | VERYLITE | GREEN | G20 |
| 8 | LITE | GREEN | G20 |
| 9 | MEDIUM | GREEN | G21 |
| 10 | MEDIUM HEAVY | GREEN | G21 |
| 11 | HEAVY | GREEN | G22 |
| 12 | VERYHEAVY | GREEN | G22 |
| 13 | VERYLITE | BLUE | B10 |
| 14 | LITE | BLUE | B10 |
| 15 | MEDIUM | BLUE | B11 |
| 16 | MEDIUM HEAVY | BLUE | B11 |
| 17 | HEAVY | BLUE | B12 |
| 18 | VERYHEAVY | BLUE | B12 |

Table IV, describes two inputs and an output in the mamdani fuzzy rule system. With this rule pattern, the movement of the robot arm will be precision according to the color and weight of the object.

## IV. RESULTS AND DISCUSSION

### A. Pick and Place Coordinate Board

Based on the working mechanism of this robot, there are several coordinates that determine the Dobot Magician trajectory, namely:

1. Coordinate 1: on the conveyor where the proximity sensor is placed.
2. Coordinate 2: at the position of the color sensor
3. Coordinate 3: at the position of the weight sensor area
4. Coordinate 4: At the position of the product storage pallet, where there are 27 storage coordinates.

Dobot Magician is trained to move at each of these coordinates through the teaching and playback mechanism. that is, how to train Dobot Magician by directing the end of effector at each coordinate it will pass through and store the coordinates in the memory of the Dobot Studio Program. At each destination position the coordinates are stored as shown in Fig. 10.

Fig. 10 is one of the Dobot Studio monitor displays that stores each position coordinate that will be traveled by the robot. The complete coordinates of the pick and place position of the object are stated in Table V.

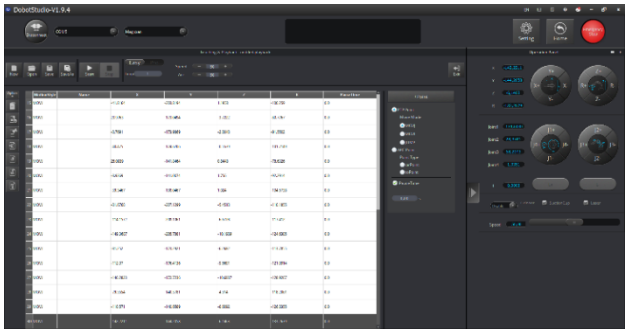

Fig. 10. Coordinate determination with teaching and playback method

TABLE V. PICK AND PLACE COORDINATE POSITION OF DOBOT MAGICIAN WORKING

| No | Name | X | Y | Z |
|---|---|---|---|---|
| 1 | Pick | 215.3 | 11.3 | -6.0 |
| 2 | Place 1 (colour check) | 299.1 | -81.8 | -9.0 |
| 3 | Place 2 (weight check) | 224.6 | -176.5 | 21.0 |
| 4 | Red 30.0 | 134.2 | -211.4 | -14.5 |
| 5 | Red 30.1 | 102.2 | -210.5 | -14.2 |
| 6 | Red 30.2 | 67.9 | -210.2 | -14.2 |
| 7 | Red 31.0 | 134.0 | -175.5 | -9.6 |
| 8 | Red 31.1 | 99.8 | -176.6 | -8.5 |
| 9 | Red 31.2 | 66.7 | -175.7 | -5.7 |
| 10 | Red 32.0 | 132.2 | -144.5 | -1.0 |
| 11 | Red 32.1 | 100.5 | -142.4 | -3.5 |
| 12 | Red 32.2 | 66.8 | -142.1 | -3.3 |
| 13 | Green 20.0 | 23.3 | -206.7 | -1.7 |
| 14 | Green 20.1 | -10.2 | -206.8 | 0.5 |
| 15 | Green 20.2 | -41.8 | -203.8 | 1.1 |
| 16 | Green 21.0 | 23.9 | -173.0 | -3.7 |
| 17 | Green 21.1 | -8.7 | -173.9 | -2.8 |
| 18 | Green 21.2 | -40.4 | -174.6 | -0.5 |
| 19 | Green 22.0 | 25.0 | -141.8 | 0.8 |
| 20 | Green 22.1 | -8.9 | -141.0 | 1.7 |
| 21 | Green 22.2 | -39.3 | -138.0 | 1.0 |
| 22 | Blue 10.0 | -81.6 | -207.1 | -5.4 |
| 23 | Blue 10.1 | -114.1 | -208.1 | -6.6 |
| 24 | Blue 10.2 | -149.3 | -205.7 | -10 |
| 25 | Blue 11.0 | -81.2 | -173.2 | -6.2 |
| 26 | Blue 11.1 | -112.0 | -176.4 | -9.5 |
| 27 | Blue 11.2 | -146.2 | -172.7 | -10.6 |
| 28 | Blue 12.0 | -79.5 | -140.3 | -4.5 |
| 29 | Blue 12.1 | -110.5 | -140.6 | -8.6 |
| 30 | Blue 12.2 | -142.2 | -144.2 | -6.1 |

Table V shows the X, Y and Z position coordinates targeted by the manipulator's work activities. Determination of these position coordinates is very important to limit the movement of the manipulator to positions that have been planned, such a model is also commonly called the path planning method. Determination of coordinates is done by the teaching and Playback method, which determines the position coordinates through movements guided by humans. Mechanically, each desired coordinate destination becomes the robot's stopping points in the key and is stored in the studio memory and from all these positions the manipulator is given work instructions so that the robot will pass through each position inputted from the program. Fig. 11 shows the teaching and playback mechanism on Dobot Magician.

Fig. 11 is the stage of determining the coordinate position of the Dobot Magician end of the effector. Each coordinate data stored in Dobot Studio will be called using the Arduino program, based on the fuzzy program instructions stored in Arduino.

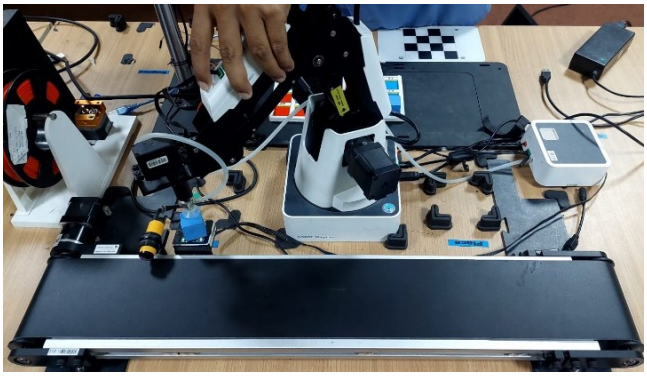

Fig. 11. Teaching and playback Dobot Magician

### B. *Fuzzification Testing for Color and Weight*

The fuzzification process produces data in a numerical format that represents the degree of membership of color and weight. Color and weight data is obtained through the Arduino output based on color sensor and loadcell readings. Arduino output is input data to the Dobot Magician control interface. With this mechanism, the movement trajectory of the Dobot Magician will be influenced by the defuzzification data generated from Arduino. The results of weight and color testing on the defuzzification process generated by Arduino are shown in Table VI.

TABLE VI. DEFUZZIFICATION TESTING

| NO | WEIGHT | COLOR | PLACE | POSISI |
|---|---|---|---|---|
| 1 | 49.1 | 146 | 1.5 | RED 30.0 |
| 2 | 74.5 | 264 | 2.07 | RED 30.1 |
| 3 | 165 | 289 | 3.66 | RED 30.2 |
| 4 | 220 | 289 | 5.44 | RED 31.1 |
| 5 | 395 | 146 | 7.26 | RED 32.0 |
| 6 | 96.4 | 475 | 10.4 | GREEN 20.0 |
| 7 | 231 | 475 | 11 | GREEN 20.1 |
| 8 | 369 | 493 | 14.2 | GREEN 21.1 |
| 9 | 416 | 506 | 16.3 | GREEN 22.0 |
| 10 | 515 | 506 | 18.1 | GREEN 22.2 |
| 11 | 104 | 698 | 18.9 | BLUE 10.0 |
| 12 | 129 | 760 | 21.1 | BLUE 10.2 |
| 13 | 231 | 760 | 23.2 | BLUE 11.1 |
| 14 | 362 | 760 | 25.1 | BLUE 12.0 |
| 15 | 409 | 884 | 26.9 | BLUE 12.2 |

Table VI describes the defuzzification results processed inside the Arduino. For example, the table explains that when the WEIGHT value is 49.1 and the COLOR value is 146, the PLACE value is 1.5, while the value of 1.5 at the PLACE position is at the RED 30.0 coordinate.

### C. *Manipulator Movement Analysis*

The movement of the Manipulator is mainly controlled through Dobot Studio. In this research, Dobot Studio processes the data generated by Arduino through the communication interface as information to program Dobot to perform movement activities according to the position of its final destination, PLACE. In this case, programming with the teaching and playback method was chosen because of its ease of executing the program. In practice, the robot arm has executed the program instructions with very good accuracy. This can be proven by comparing the difference between the coordinates in the program and the final destination coordinates. This comparison is calculated using Matlab

based on the forward kinematic approach based on equation (7). The accuracy test data of the robot arm movement is shown in Table VII.

With 12 samples of robot arm movements, the accuracy error of product placement on each pallet is 0.537 mm. This is still within the permissible placement position limits, meaning that if there is a position shift of 0.537 mm, the workpiece is still in the specified storage box. From each coordinate position shift to the path planning calculation in the table above, it can be calculated that the average deviation of product sample placement errors is 1.8%. Product placement accuracy is very important to avoid product accumulation in the same position or products placed outside the proper storage area.

TABLE VII. TESTING THE MOVEMENT ACCURACY OF THE ROBOT ARM

| NO | PROGRAM COORDINATES | CALCULATION COORDINATES | DIFFERENCE (mm) |
|---|---|---|---|
| 1 | X = 47.552 | X = 47.866 | 0.314 |
| 2 | Y = -193.864 | Y = -194.811 | 0.946 |
| 3 | Z = 31.935 | Z = 30.022 | 1.913 |
| 4 | X =47.552 | X = 47.552 | 0 |
| 5 | Y =-193.864 | Y = -193.864 | 0 |
| 6 | Z = 3.935 | Z = 4.387 | 0.452 |
| 7 | X = 200.053 | X = 199.552 | 0.501 |
| 8 | Y = -207.012 | Y = -207.86 | 0.848 |
| 9 | Z = 36.178 | Z = 36.935 | 0.757 |
| 10 | X = 200.053 | X = 200.053 | 0 |
| 11 | Y = -207.012 | Y =-207.012 | 0 |
| 12 | Z =56.145 | Z = 55.445 | 0.714 |
| The average difference (mm) | | | 0.537 |
| Average Deviation of Error (%) | | | 1.8 |

### *D. Manipulator Movement Trajectory*

Based on the kinematics equation of the Dobot Magician and the length parameter of each Dobot rigid body, the trajectory graph of the Dobot Magician end of the effector can be visualized using Matlab programming. This trajectory graph provides a visual representation of the Dobot Magician's crossing. This analysis helps to identify the route of movement to avoid the risk of collision with other objects around the manipulator area. The movement of the Dobot Magician manipulator in a sample coordinate X = 47.552, Y = -193.864 and Z = 31.935 is expressed in the Matlab graph in Fig. 12.

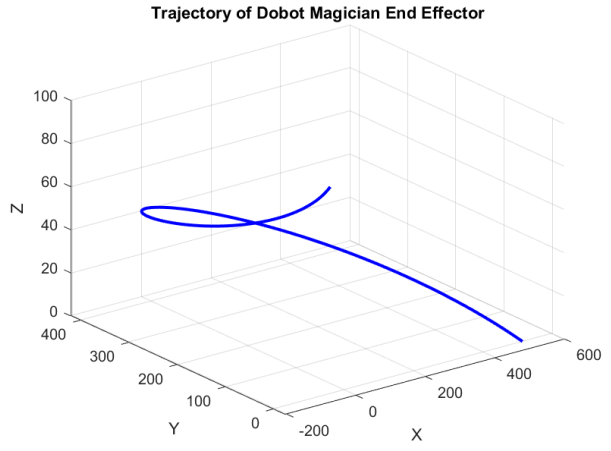


Fig. 12. Dobot Magician end of effector Trsjectory chart

In Fig. 12, x, y, and z represent the position of Dobot Magician's end of effector in 3D space when getting a certain coordinate input.

## V. CONCLUSION

This research has proven that the performance of Dobot Magician can be improved with fuzzy logic control through the collaboration of the Dobot Studio program with Arduino Mega through the communication interface provided. Fuzzy Logic Control interference has improved the performance of Dobot Magician so that it is able to make smarter decisions in executing two input variables, namely the weight and color of objects. So far, Dobot Magician only works monotonously according to the sequential order of the program. Through the collaboration of fuzzy logic control, Dobot Magician can work based on different conditions according to the input parameters it reads. The collaboration of Dobot Studio control with a Fuzzy logic control system for pick and place applications based on object weight and object color variables is able to control the movement of Dobot Magician with very high positional accuracy. The test data states that the accuracy and precision of the average coordinates of the product pick and place against the path planning has an error deviation of 1.8%. The challenge that occurs in this research is the color reading of the product with the color sensor, which is caused by differences in the light intensity of the lamps used. but this can be solved by testing in a fixed room with a fixed light intensity. This research can be continued by developing a Vision system using a camera to replace the color sensor.

## GLOSSARY

| | |
|---|---|
| Path Planning | Plan for the robot to move through |
| Trajectory | The route followed by the movement of the robot |
| Kinematics | Motion of objects and systems of objects without considering the forces that cause motion |
| Fuzzification | The process of converting an input from a firm (crisp) to a fuzzy (linguistic variable) form. |
| Rule Base | Rules that express causal relationships in fuzzy systems |
| Defuzzification | The process of converting a fuzzy output into a firm (crisp) value |
| Aggregation | Process of combining some fuzzy sets into 1 set |

## ACKNOWLEDGMENTS

The researcher is a lecturer at the mechatronics engineering technology study program of Indorama Purwakarta Polytechnic, currently pursuing a doctoral program at the faculty of mechanical engineering, Diponegoro University Semarang. This research is one of the mandatory research that must be carried out by students to complete the doctoral program. This research is supported by funding from the Indorama Education Foundation and independent funds.

## REFERENCES

[1] N. Simaan, R. M. Yasin, and L. Wang, “Medical technologies and challenges of robot-assisted minimally invasive intervention and

diagnostics," *Annu. Rev. Control. Robot. Auton. Syst.*, vol. 1, pp. 465–490, 2018.

[2] E. Tosello, N. Castaman, and E. Menegatti, "Using robotics to train students for industry 4.0," *Ifac-Papersonline*, vol. 52, no. 9, pp. 177–183, 2019.

[3] X. Jin *et al.*, "Visual-based data exchange system for internal and external networks in physical isolation," *Cogn. Robot.*, vol. 1, pp. 134–144, 2021.

[4] M. R. Islam, M. A. Rahaman, M. A. U. Zaman, and M. Habibur, "Cartesian trajectory based control of dobot robot," in *Proc. Int. Conf. Ind. Eng. Operations Manage*, pp. 1507-1517, 2019.

[5] M. A. Abo-Sinna, Y. Y. Abo-Elnaga, and A. A. Mousa, "An interactive dynamic approach based on hybrid swarm optimization for solving multiobjective programming problem with fuzzy parameters," *Appl. Math. Model.*, vol. 38, no. 7–8, pp. 2000–2014, 2014, doi: 10.1016/J.Apm.2013.10.013.

[6] A.-V. Duka, "Neural network based inverse kinematics solution for trajectory tracking of a robotic arm," *Procedia Technol.*, vol. 12, pp. 20–27, 2014, doi: 10.1016/J.Protcy.2013.12.451.

[7] M. Alebooyeh and R. J. Urbanic, "Neural network model for identifying workspace, forward and inverse kinematics of the 7-dof yumi 14000 abb collaborative robot," *Ifac-Papersonline*, vol. 52, no. 10, pp. 176–181, 2019, doi: 10.1016/J.Ifacol.2019.10.019.

[8] M. Soori, B. Arezoo, and R. Dastres, "Artificial intelligence, machine learning and deep learning in advanced robotics, a review," *Cogn. Robot.*, vol. 3, pp. 54–70, 2023, doi: 10.1016/J.Cogr.2023.04.001.

[9] S. A. Kouritem, M. I. Abouheaf, N. Nahas, and M. Hassan, "A multi-objective optimization design of industrial robot arms," *Alexandria Engineering Journal*, vol. 61, no. 12, pp. 12847-12867, 2022.

[10] R. Holubek, M. Janícek, and G. O. Tirian, "Verification of the voice control modification of robot-DOBOT Magician depending to change voice frequency," *J. Phys. Conf. Ser.*, vol. 2212, no. 1, 2022, doi: 10.1088/1742-6596/2212/1/012016.

[11] S. Chakraborty and P. S. Aithal, "Open Loop Automated Baby Cradle Using Dobot Magician and C#," *Int. J. Appl. Eng. Manag. Lett.*, vol. 6, no. 1, pp. 344–349, 2022, doi: 10.47992/Ijaeml.2581.7000.0141.

[12] P. S. Tsai, T. F. Wu, J. Y. Chen, And F. H. Lee, "Drawing System With Dobot Magician Manipulator Based On Image Processing," *Machines*, vol. 9, no. 12, 2021, doi: 10.3390/Machines9120302.

[13] N. O. M. Chilo, L. F. C. Ccari, E. Supo, E. S. Espinoza, Y. S. Vidal, and L. Pari, "Optimal Signal Processing for Steady Control of a Robotic Arm Suppressing Hand Tremors for EOD Applications," *IEEE Access*, vol. 11, pp. 13163-13178, 2023.

[14] M. Slim, N. Rokbani, B. Neji, M. A. Terres, and T. Beyrouthy, "Inverse Kinematic Solver Based on Bat Algorithm for Robotic Arm Path Planning," *Robotics*, vol. 12, no. 2, p. 38, 2023.

[15] P. P. Reboucas Filho, S. P. P. da Silva, V. N. Praxedes, J. Hemanth, and V. H. C. de Albuquerque, "Control of singularity trajectory tracking for robotic manipulator by genetic algorithms," *Journal of computational science*, vol. 30, pp. 55-64, 2019.

[16] D. Osiński and D. Jasińska-Choromańska, "Kinematic structure of turning modules in orthotic robots," *Procedia Eng.*, vol. 177, pp. 450–454, 2017, doi: 10.1016/J.Proeng.2017.02.244.

[17] Z. Yang, X. Pan, Y. Wang, and W. Tang, "Kinematics and dynamics analysis of the main motion system of reciprocating machine tools," *Iop Conf. Ser. Mater. Sci. Eng.*, vol. 394, no. 3, 2018, doi: 10.1088/1757-899x/394/3/032067.

[18] C. A. My and V. M. Hoan, "Kinematic and dynamic analysis of a serial manipulator with local closed loop mechanisms," *Vietnam J. Mech.*, vol. 41, no. 2, pp. 141–155, 2019, doi: 10.15625/0866-7136/13073.

[19] J. Vavro, J. Vevro, P. Kováčiková, and R. Bezdedová, "Kinematic and dynamic analysis of planar mechanisms by means of the solid works software," *Procedia Eng.*, vol. 177, pp. 476–481, 2017, doi: 10.1016/J.Proeng.2017.02.248.

[20] T. O. Terefe, H. G. Lemu, and T. B. Tuli, "Kinematic modeling and analysis of a walking machine (robot) leg mechanism on a rough terrain," *Adv. Sci. Technol. Res. J.*, vol. 13, no. 3, p. 43–53, 2019, doi: 10.12913/22998624/109792.

[21] O. Hock and J. Šedo, "Forward and inverse kinematics using pseudoinverse and transposition method for robotic arm dobot," In *Kinematics*, Intech, 2017.

[22] S. Kucuk and Z. Bingul. *Robot kinematics: Forward and inverse kinematics* (pp. 117-148). London, UK: INTECH Open Access Publisher, 2006.

[23] N. Wagaa, H. Kallel, and N. Mellouli, "Analytical and deep learning approaches for solving the inverse kinematic problem of a high degrees of freedom robotic arm," *Engineering Applications of Artificial Intelligence*, vol. 123, p. 106301, 2023.

[24] S. Dikmenli, "Forward & inverse kinematics solution of 6-dof robots those have offset & spherical wrists," *Eurasian J. Sci. Eng. Technol.*, vol. 3, no. 1, pp. 14–28, 2022, doi: 10.55696/Ejset.1082648.

[25] D. R. Parhi, B. B. V. L. Deepak, D. Nayak, and A. Amrit, "Forward and inverse kinematic models for an articulated robotic manipulator," *International Journal of Artificial Intelligence and Computational Research*, vol. 4, no. 2, pp. 103-109, 2012.

[26] M. Almaged, "Forward and inverse kinematic analysis and validation of the abb irb 140 industrial robot," *Int. J. Electron. Mech. Mechatronics Eng.*, vol. 7, no. 2, pp. 1383–1401, 2017.

[27] M. Ben-Ari and F. Mondada. *Elements of robotics*. Springer Nature, 2017.

[28] J. Zhu and F. Tian, "Kinematics analysis and workspace calculation of a 3-dof manipulator," *Iop Conf. Ser. Earth Environ. Sci.*, vol. 170, no. 4, 2018, doi: 10.1088/1755-1315/170/4/042166.

[29] F. Gonçalves, T. Ribeiro, A. F. Ribeiro, G. Lopes, and P. Flores, "A recursive algorithm for the forward kinematic analysis of robotic systems using euler angles," *Robotics*, vol. 11, no. 1, pp. 1–20, 2022, doi: 10.3390/Robotics11010015.

[30] H. Z. Ting, M. Hairi, M. Zaman, M. Ibrahim, and A. Moubark, "Kinematic analysis for trajectory planning of open-source 4-DoF robot arm," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, pp. 769-777, 2021.

[31] M. Anschober, R. Edlinger, R. Froschauer, and A. Nüchter, "Inverse kinematics of an anthropomorphic 6r robot manipulator based on a simple geometric approach for embedded systems," *Robotics*, vol. 12, no. 4, 2023, doi: 10.3390/Robotics12040101.

[32] A. Fomin, A. Antonov, V. Glazunov, and Y. Rodionov, "Inverse and forward kinematic analysis of a 6-DOF parallel manipulator utilizing a circular guide," *Robotics*, vol. 10, no. 1, p. 31, 2021.

[33] B. Tam, T. A. O. Linh, T. Nguyen, T. Nguyen, H. Hasegawa, and D. Watanabe, "DE-based algorithm for solving the inverse kinematics on a robotic arm manipulators," *J. Phys. Conf. Ser.*, vol. 1922, no. 1, 2021, doi: 10.1088/1742-6596/1922/1/012008.

[34] S. Kucuk and Z. Bingul, "Inverse kinematics solutions for industrial robot manipulators with offset wrists," *Appl. Math. Model.*, vol. 38, no. 7–8, pp. 1983–1999, 2014, doi: 10.1016/J.Apm.2013.10.014.

[35] D. Rodríguez-Guerra, G. Sorrosal, I. Cabanes, and C. Calleja, "Human-Robot Interaction Review: Challenges and Solutions for Modern Industrial Environments," in *IEEE Access*, vol. 9, pp. 108557-108578, 2021, doi: 10.1109/ACCESS.2021.3099287..

[36] H. P. Nurba, D. Hadian, N. Lestari, K. A. Munastha, H. Mistialustina, and E. Rachmawati, "Performance Evaluation of 3 DOF Arm Robot With Forward Kinematics Denavit-Hartenberg Method For Coffee Maker Machine," *2022 16th International Conference on Telecommunication Systems, Services, and Applications (TSSA)*, pp. 1-6, 2022, doi: 10.1109/TSSA56819.2022.10063918..

[37] C. Faria, J. L. Vilaça, S. Monteiro, W. Erlhagen, and E. Bicho, "Automatic Denavit-Hartenberg Parameter Identification for Serial Manipulators," *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*, pp. 610-617, 2019, doi: 10.1109/IECON.2019.8927455.

[38] C. Corina, E. Pop, and M. Leba, "Modeling and Simulation for a 3 D Robot Controlled by a Pick and Place Application," in *WSEAS International Conference. Proceedings. Mathematics and Computers in Science and Engineering* (No. 10). World Scientific and Engineering Academy and Society, 2009.

[39] A. A. Hayat, R. G. Chittawadigi, A. D. Udai, and S. K. Saha, "Identification of denavit-hartenberg parameters of an industrial robot," *Acm Int. Conf. Proceeding Ser.*, pp. 4–9, 2013, doi: 10.1145/2506095.2506121.

[40] N. Correll, "Introduction To Autonomous Robots," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2016.

[41] J. A. Soares, "Urban sociology and research methods on neighborhoods

and health," *Handb. Urban Heal.*, pp. 361–378, 2006, doi: 10.1007/0-387-25822-1_18.

[42] L. Qingsheng and J. Andika, "Analysis of kinematic for legs of a hexapod using denavit-hartenberg convention," *Sinergi*, vol. 22, no. 2, p. 69, 2018, doi: 10.22441/Sinergi.2018.2.001.

[43] E. Akindele Ayoola, I. Awodeyi Afolabi, O. Matthews Victor, A. Alashiri Olaitan, K. Idowu Oriyomi, and J. Olaloye Folarin, "Development of an Electronic Weighing Indicator for Digital Measurement," *International Research Journal of Engineering and Technology*, vol. 5, no. 9, pp. 19-25, 2018.

[44] A. Renfrew, "Book review: introduction to robotics: mechanics and control," *Int. J. Electr. Eng. Educ.*, vol. 41, no. 4, pp. 388–388, 2004, doi: 10.7227/Ijeee.41.4.11.

[45] I. Agustian, N. Daratha, R. Faurina, A. Suandi, and S. Sulistyaningsih, "Robot Manipulator Control with Inverse Kinematics PD-Pseudoinverse Jacobian and Forward Kinematics Denavit Hartenberg," *J. Elektron. Dan Telekomun.*, vol. 21, no. 1, p. 8, 2021, doi: 10.14203/Jet.V21.8-18.

[46] D. K. Jain, S. Neelakandan, T. Veeramani, S. Bhatia, and F. H. Memon, "Design of fuzzy logic based energy management and traffic predictive model for cyber physical systems," *Computers and Electrical Engineering*, vol. 102, p. 108135, 2022.

[47] C. Mavroidis, E. Lee, and M. Alam, "A new polynomial solution to the geometric design problem of spatial RR robot manipulators using the denavit and Hartenberg parameters," *J. Mech. Des.*, vol. 123, no. 1, pp. 58-67, 2001.

[48] A. K. Kovalchuk and F. K. Akhmetova, "Denavit-Hartenberg Coordinate System for Robots with Tree-Like Kinematic Structure," vol. 5, no. 4, pp. 244–254, 2016.

[49] L. Radavelli, R. Simoni, E. De Pieri, and D. Martins, "A comparative study of the kinematics of robots manipulators by Denavit-Hartenberg and dual quaternion," *Mecánica Computacional*, vol. 31, no. 15, pp. 2833-2848, 2012.

[50] H. J. Chung *et al*. *A robust formulation for prediction of human running*. SAE Technical Paper, 2007.

[51] H. Afrisal, A. D. Setiyadi, M. A. Riyadi, R. Ismail, O. Toirov, and I. Setiawan, "Performance Analysis of 4-DOF RPRR Robot Manipulator Actuation Strategy for Pick and Place Application in Healthcare Environment," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 12, no. 6, pp. 2258-2265, 2022.

[52] C. Klug, D. Schmalstieg, T. Gloor, and C. Arth, "A Complete Workflow for Automatic Forward Kinematics Model Extraction of Robotic Total Stations using the Denavit-Hartenberg Convention," *J. Intell. Robot. Syst. Theory Appl.*, vol. 95, no. 2, pp. 311–329, 2019, doi: 10.1007/S10846-018-0931-4.

[53] E. A. Nugroho, N. R. Wibowo, R. Rizalludin, and M. Ruswanda, "Fuzzy system as four-joint robot movement control for moving goods based on time and object color," *Jurnal Ramatekno*, vol. 2, no. 2, pp. 7-15, 2022.

[54] S. N. Sivanandam, S. Sumathi, and S. N. Deepa. *Introduction to Fuzzy Logic using MATLAB*. Springer Science & Business Media, 2006.

[55] C. Y. Chen, M. G. Her, Y. C. Hung, and M. Karkoub, "Approximating a Robot Inverse Kinematics Solution using Fuzzy Logic Tuned by Genetic Algorithms," *Int. J. Adv. Manuf. Technol.*, vol. 20, no. 5, pp. 375–380, 2002, doi: 10.1007/S001700200166.

[56] M. Crenganiş, M. Tera, C.Biriş, and C. Gîrjob, "Dynamic analysis of a 7 DOF robot using fuzzy logic for inverse kinematics problem," *Procedia computer science*, vol. 162, pp. 298-306, 2019.

[57] A. K. Varshney and V. Torra, "Literature Review of the Recent Trends and Applications in Various Fuzzy Rule-Based Systems," *Int. J. Fuzzy Syst.*, vol. 25, no. 6, pp. 2163–2186, 2023, doi: 10.1007/S40815-023-01534-W.

[58] M. R. Mufid, N. R. Kusuma Saginta Putri, A. Fariza, and Mu'arifin, "Fuzzy Logic and Exponential Smoothing for Mapping Implementation of Dengue Haemorrhagic Fever in Surabaya," *2018 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*, pp. 372-377, 2018, doi: 10.1109/KCIC.2018.8628533..

[59] S. Maheswari, M. Shalini, and T. L. Yookesh, "Defuzzification Formula for Modelling and Scheduling a Furniture Fuzzy Project Network," *Int. J. Eng. Adv. Technol.*, vol. 9, no. 5, pp. 279–283, 2019, doi: 10.35940/Ijeat.A1048.1291s519.

[60] N. Sabounchi, K. Triantis, S. Sarangi, and S. Liu, "Fuzzy Modeling of Linguistic Variables in a System Dynamics Context," *Proc. 29th Int. Conf. Syst. Dyn. Soc.*, pp. 1–30, 2011.

[61] P. Hofmann, "Defuzzification strategies for fuzzy classifications of remote sensing data," *Remote Sensing*, vol. 8, no. 6, p. 467, 2016.

[62] K. S. Gilda and S. L. Satarkar, "Analytical overview of defuzzification methods," *International Journal of Advance Research, Ideas and Innovations in Technology*, vol. 6, no. 2, pp. 359-365, 2020.

[63] S. Razvarz and M. Tahmasbi, "Fuzzy Equations and Z-Numbers for Nonlinear Systems Control," *Procedia Comput. Sci.*, vol. 120, pp. 923–930, 2017, doi: 10.1016/J.Procs.2017.11.327.

[64] D. Behera and S. Chakraverty, "Solution to Fuzzy System of Linear Equations with Crisp Coefficients," *Fuzzy Inf. Eng.*, vol. 5, no. 2, pp. 205–219, 2013, doi: 10.1007/S12543-013-0138-0.

[65] V. B. Tarassov, "Development of Fuzzy Logics: from Universal Logic Tools to Natural Pragmatics and Non-Standard Scales," *Procedia Comput. Sci.*, vol. 120, no. 2017, pp. 908–915, 2017, doi: 10.1016/J.Procs.2017.11.325.

[66] B. Orazbayev, E. Ospanov, N. Kissikova, N. Mukataev, and K. Orazbayeva, "Decision-Making in the Fuzzy Environment on the Basis of Various Compromise Schemes," *Procedia Comput. Sci.*, vol. 120, pp. 945–952, 2017, doi: 10.1016/J.Procs.2017.11.330.

[67] J. Gallardo-Alvarado, M. A. Garcia-Murillo, L. A. Alcaraz-Caracheo, F. J. Torres, and X. Y. Sandoval-Castro, "Forward kinematics and singularity analyses of an uncoupled parallel manipulator by algebraic screw theory," *IEEE Access*, vol. 10, pp. 4513-4522, 2021.

[68] A. N. Sharkawy, "Forward and inverse kinematics solution of a robotic manipulator using a multilayer feedforward neural network," *Journal of Mechanical and Energy Engineering*, vol. 6, no. 2, 2022.

[69] H. Ren and P. Ben-Tzvi, "Learning inverse kinematics and dynamics of a robotic manipulator using generative adversarial networks," *Robotics and Autonomous Systems*, vol. 124, p. 103386, 2020.

[70] A. F. Hastawan *et al*., "Comparison of testing load cell sensor data sampling method based on the variation of time delay," in *IOP Conference Series: Earth and Environmental Science*, vol. 700, no. 1, p. 012018, 2021.

[71] M. S. Surbakti *et al.*, "Development of Arduino Uno-Based Tcs3200 Color Sensor and its Application on the Determination of Rhodamine B Level in Syrup," *Indones. J. Chem.*, vol. 22, no. 3, pp. 630–640, 2022, doi: 10.22146/Ijc.69214.

[72] A. Najmurrokhman, K. Kusnandar, F. Maulana, B. Wibowo, and E. Nurlina, "Design of a prototype of manipulator arm for implementing pick-and-place task in industrial robot system using TCS3200 color sensor and ATmega2560 microcontroller," in *Journal of Physics: Conference Series*, vol. 1375, no. 1, p. 012041, 2019.

[73] R. Das, "Automation of Tank Level using PLC and Establishment of HMI by Scada," *Iosr J. Electr. Electron. Eng.*, vol. 7, no. 2, pp. 61–67, 2013, doi: 10.9790/1676-0726167.

[74] B. M. Yilmaz, E. Tatlicioglu, A. Savran, and M. Alci, "Adaptive fuzzy logic with self-tuned membership functions based repetitive learning control of robotic manipulators," *Applied Soft Computing*, vol. 104, p. 107183, 2021.

[75] F. Wildani, R. Mardiati, E. Mulyana, A. E. Setiawan, R. R. Nurmalasari, and N. Sartika, "Fuzzy Logic Control for Semi-Autonomous Navigation Robot Using Integrated Remote Control," *2022 8th International Conference on Wireless and Telematics (ICWT)*, pp. 1-5, 2022, doi: 10.1109/ICWT55831.2022.9935458.

[76] Arduino. *Arduino Mega 2560 Datasheet*. Power, p. 3, 2015.

[77] M. Janíček, R. Ružarovský, K. Velíšek, and R. Holubek, "Analysis of Voice Control of a Collaborative Robot," *J. Phys. Conf. Ser.*, vol. 1781, no. 1, 2021, doi: 10.1088/1742-6596/1781/1/012025.

[78] M. Marsono, Y. Yoto, A. Suyetno, and R. Nurmalasari, "Design and programming of 5 axis manipulator robot with grblgru open source software on preparing vocational students' robotic skills," *Journal of Robotics and Control (JRC)*, vol. 2, no. 6, pp. 539-545, 2021.

[79] F. C. Can, "Arduino Based Planar Two DoF Robot Manipulator," *J. Mech. Eng. Autom.*, vol. 8, no. 3, 2018, doi: 10.17265/2159-5275/2018.03.002.

[80] L. Stěpničkova, M. Stěpnička, and D. Sikora, "Fuzzy Rule-Based

Ensemble with use of Linguistic Associations Mining for Time Series Prediction," *8th Conf. Eur. Soc. Fuzzy Log. Technol. Eusflat 2013 - Adv. Intell. Syst. Res.*, vol. 32, pp. 408–415, 2013, doi: 10.2991/Eusflat.2013.63.

[81] V. Afrian and S. Riyadi, "Comparison Of Different Rule Base Matrix In Fuzzy Logic Controller," *J. Phys. Conf. Ser.*, vol. 1444, no. 1, 2020, doi: 10.1088/1742-6596/1444/1/012018.