Title – Modeling and Analysis of a 5 DOF Robotic Arm Manipulator

Objective

- To use technical skills gained from the Embedded Systems course in implementing a 5-DoF Robot Arm.

Introduction

This project develops a kinematic model for a 5-DOF (Degrees of Freedom) industrial robot that controls its hand position. In Robotics terms, we call the hand position as end effector position.

An easier method to control the end effector position is by changing the servo position using `servo.write()` function. This can essentially move the end-effector to any random position.

The goal of this project is to control the end effector coordinate position in a precise manner. By coordinate position, we mean the xyz points that it can reach given its configurations (link lengths, etc.).

We achieve this goal by developing a mathematical model of the whole robot. This is done by assigning individual xyz coordinate frames where each servo motor is located, and then analytically estimating where the end effector coordinate frame might be on the reference frame. The reference frame is where the first joint is located **in case of our project.** This whole process is called **Forward Kinematics**.

To make the end effector reach a specific xyz coordinate position in the robot's workspace we perform **Inverse Kinematics,** where we use the equations of Forward Kinematics to come up with the required joint angles for that specific xyz coordinate place. Workspace is the set of all positions that the robot's end effector is able to reach.

This project was a bright opportunity for us to achieve what we had learned about Robot Mechanics and implementing them using the techniques we learned in Embedded Systems course.

Components

The project is based on the following components.

1. ESP32
2. Robot Frame
3. Servo Motors

The ESP32 microcontroller was used for better processing efficiency and memory and basic AD002 servo motors were used. So we were able to achieve the tasks using basic <Servo.h> library, and advanced servo motor control techniques were not required for this project.

Structure and Mathematical Modeling

We used the frame of Adeept 5 DoF Robot Arm on which all the servo motors are properly attached.

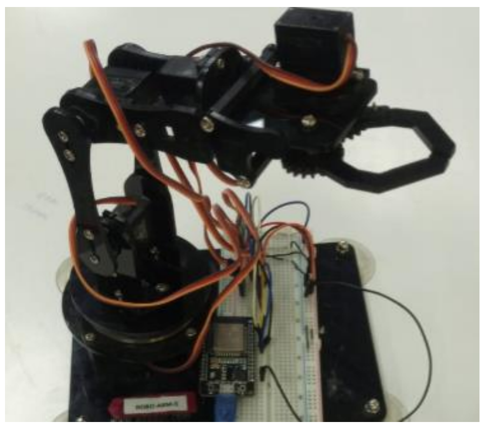

Figure 1

The following diagram shows how we abstract and model the robot structure that helps us in calculations. For ease, only the first 3 Revolute Joints (or servo motors) are modeled using Inverse Kinematics and the remaining 2 Revolute Joints are taken as Wrist and Grip. So the overall structure is indeed 5 DoF.

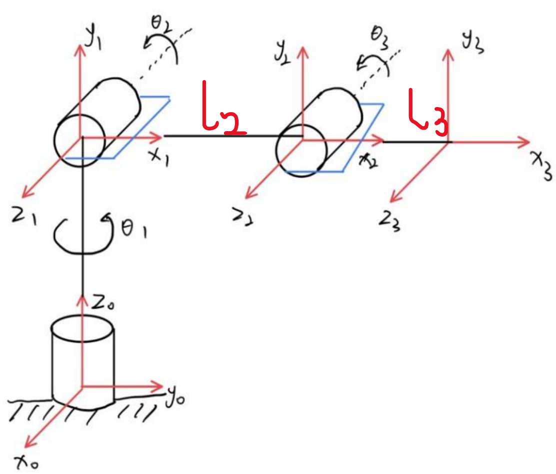

Figure 2

After obtaining this structure, we fill in the following table.

{i}	α_{i-1}	a_{i-1}	d_i	θ_i
1.	0	0	l_1	θ_1
2.	90°	0	0	θ_2
3.	0	l_2	0	θ_3
4.	0	l_3	0	0

This is called the DH table, and {i} corresponds to the frame number; the frame number is also a kind of joint number. Using forward kinematics techniques, we reach the following equation that relates the servo motor position and end effector position.

```
T04 =

[cos(q2 + q3)*cos(q1), -sin(q2 + q3)*cos(q1),  sin(q1), cos(q1)*(l3*cos(q2 + q3) + l2*cos(q2))]
[cos(q2 + q3)*sin(q1), -sin(q2 + q3)*sin(q1), -cos(q1), sin(q1)*(l3*cos(q2 + q3) + l2*cos(q2))]
[        sin(q2 + q3),          cos(q2 + q3),        0,      l1 + l3*sin(q2 + q3) + l2*sin(q2)]
[                   0,                     0,        0,                                      1]
```

Figure 3

From this matrix equation in Figure 3 we get to know that the xyz position has the following equations,

$$x = l_2 c_1 c_2 + l_3 c_1 c_{23}$$

$$y = l_2 s_1 c_2 + l_3 s_1 c_{23}$$

$$z = l_2 s_2 + l_1 s_{23}$$

Where ‘c’ and ‘s’ correspond to cosine and sine angles.

We solve these transcendental equations and derive equations for the thetas.

$$\theta_1 = atan2(y, x)$$

$$r = \sqrt{x^2 + y^2}$$

$$\theta_2 = atan2\left(\frac{z}{r}\right) - atan2\left(\frac{l_3 s_3}{l_2 + l_3 c_3}\right)$$

$$\theta_3 = atan2(s_3, c_3)$$

Methadology

After doing all this, we proceed towards coding.

The code is focused more on pick and place operations using the coordinate positions (xyz) of the object to be picked and the placement destination (xyz).

The robot first moves to the XYZ position of the object using inverse kinematics. Then it releases the gripper to release the object at the destination XYZ position, using inverse kinematics.

Instead of immediately reaching the robot to the desired xyz coordinates, interpolation techniques have been incorporated to make the robot's movement smooth and stable-looking.

So the overall working is,

1. The robot approaches the object position and grabs the object using the wrist and gripper
2. Robot moves to the top position
3. Robot releases the object at the destination position using the wrist and gripper.

Code Flow Diagram

Figure 4 briefly demonstrates how the code performs inverse kinematics for our robot.

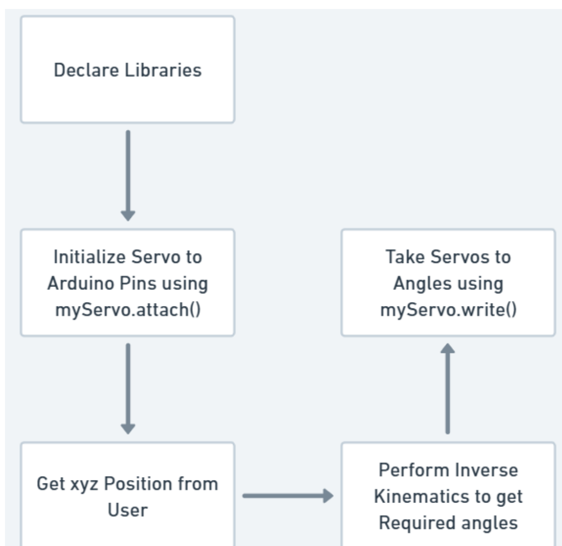

Figure 4

Figure 5 demonstrates the working of the project.

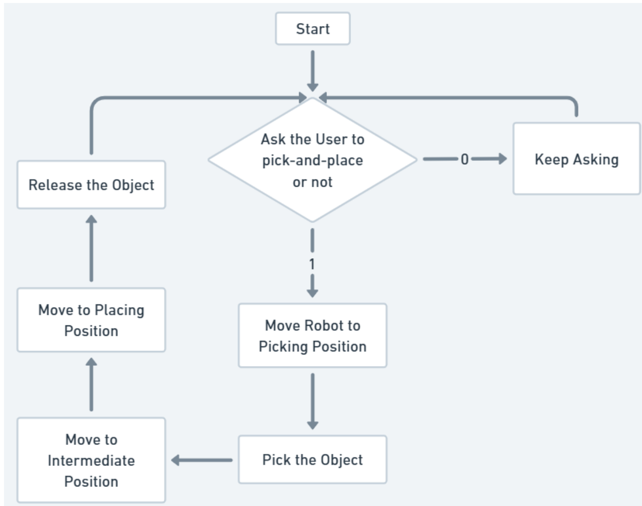

Figure 5

MATLAB Simulation

MATLAB simulations are performed to verify whether our equations work or not. We mainly focused on the pick-and-place operation, a task that has applications. Figure 6 represents the object picking position, and Figure 8 represents the object placing position. Figure 7 is the intermediate position. We can see the robot is performing tasks as it is intended to.

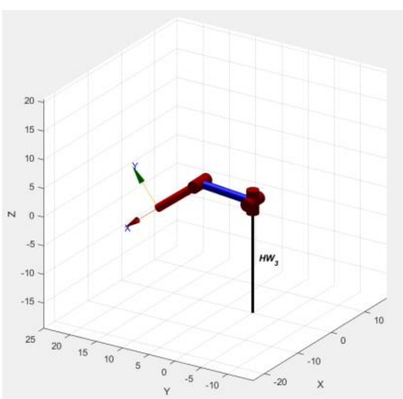

Figure 6

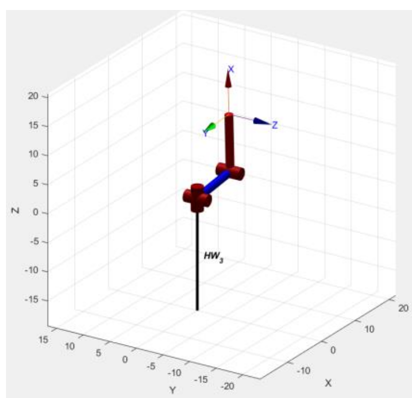

Figure 7

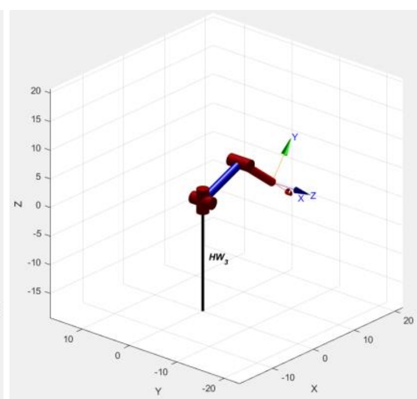

Figure 8

Practical Results

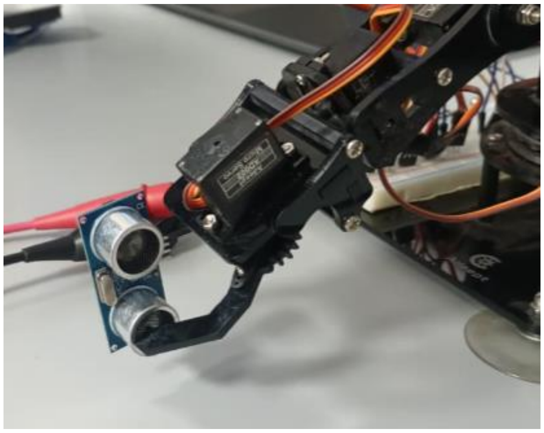

Figure 9

Figure 10

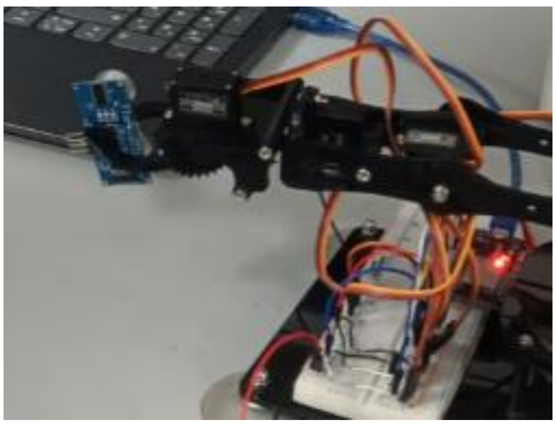

Figure 11

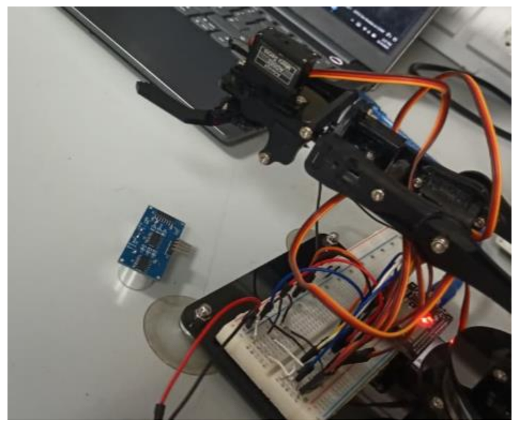

Figure 12

Conclusion

From this Lab Semester Project of Embedded Systems, we got the opportunity to implement both what we learned in the Introduction to Embedded Systems and in the Robot Mechanics course. The project introduced us to various challenges and improved our hands-on practice with practical equipment. It made us realize how theoretical and real-world implementations can vastly differ in achieving desired results. The project was accomplished, and the developed model and equations operate for the designed 5DoF robot.