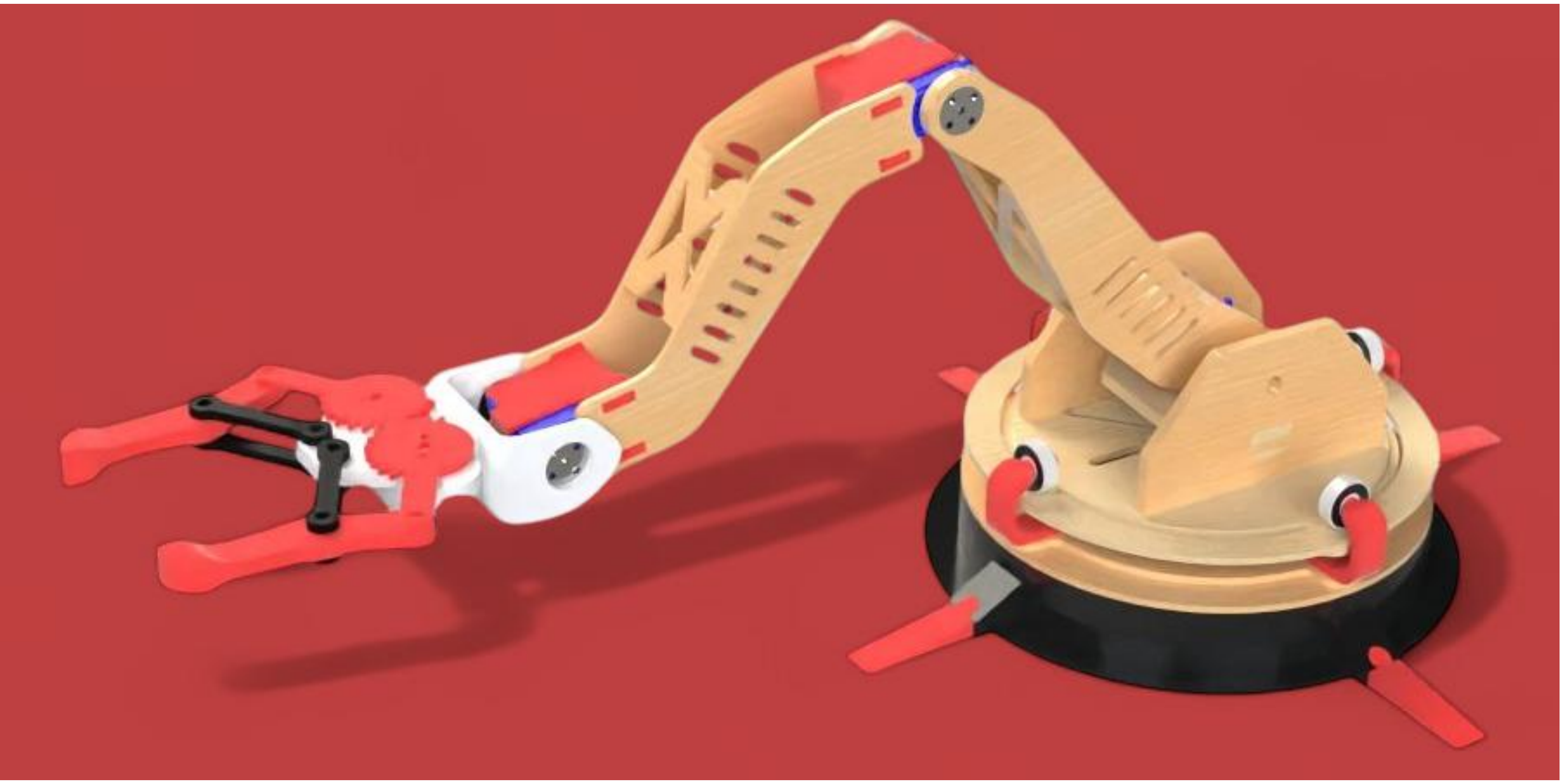
Robotics project

Invisible Arm (Gripper)



Team 6

Names and IDs

	Name	ID
1	Fares Eldamanhoury	**7936**
2	Raed Eldamanhoury	**7931**
3	Nehal Osama	**7933**
4	Afraim Gamil	**7901**
5	Fathy Fayez	**8531**
6	Omar Ayman	**8218**
7	Mostafa Ehab	**8212**
8	Adam Ezzat	**7904**
9	Ibrahim Mohamed	**7897**
10	Abdelghfour Alaa	**7426**
11	Mahmoud fathy	**7565**
12	Salem El-sayed	**7949**
13	Youssef Amr	**7945**

Table of Content

Introduction 3

Description 3

CAD Design 4

Servo Holder Assembly 4

FEA 5

Roboanalyzer 6

Matlab Calculations 8

Contours and Results 10

Wiring Diagram 11

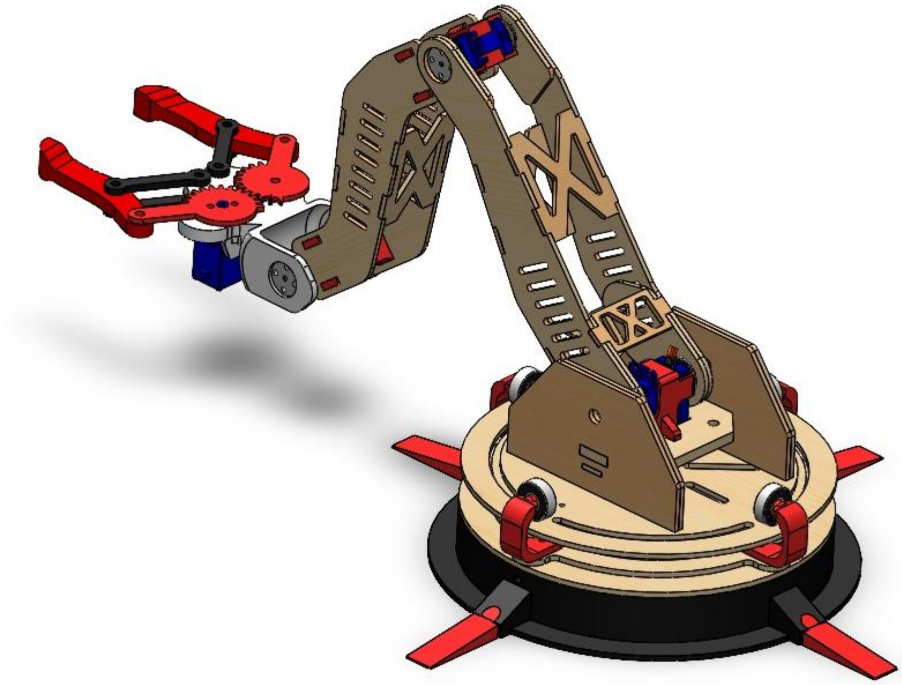
1. Introduction

Robotics has become an essential part of modern industry, automating tasks to improve efficiency, precision, and safety. One of the most widely used robotic components is the gripper arm, designed to mimic the human hand in grasping and manipulating objects. These arms are commonly used in manufacturing, packaging, and assembly lines to handle materials with accuracy and speed. Gripper arms come in various designs, such as mechanical, pneumatic, or vacuum-based, depending on the application. Their integration into production systems reduces labor costs and increases productivity. As technology advances, robotic grippers continue to evolve, offering smarter and more adaptive solutions.

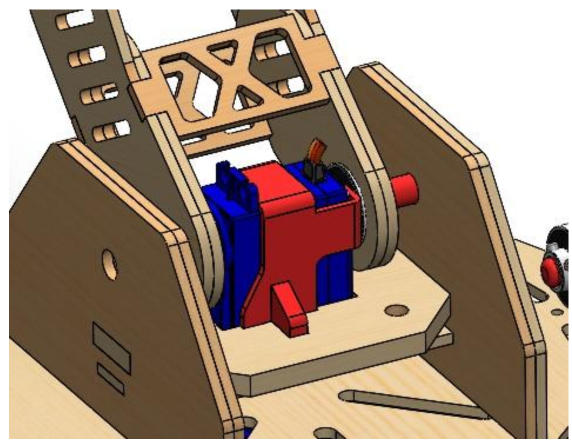
2. Description

Gripper arm designed for industrial applications. Our design includes a multi-jointed arm structure with rotational joints, powered by servo motors, allowing for a wide range of motion. The base is circular, it offers 360-degree rotation for enhanced flexibility. The gripper at the end features a gear-driven mechanism, enabling it to open and close precisely to grasp various objects. This type of robotic arm is commonly used for pick-and-place tasks, material handling, or light assembly operations. The structure is modular and fabricated from wood, laser-cut, and 3-D printing making it ideal for prototyping and educational use.

3. CAD Design

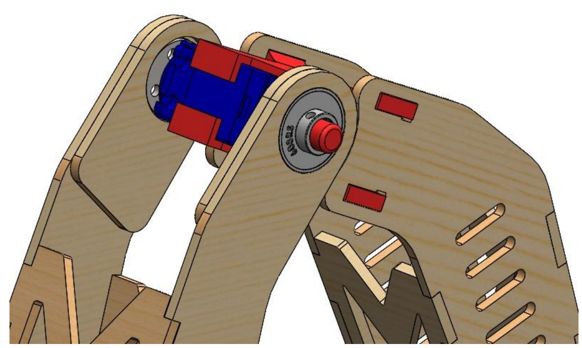


3.1. Servo Holder Assembly



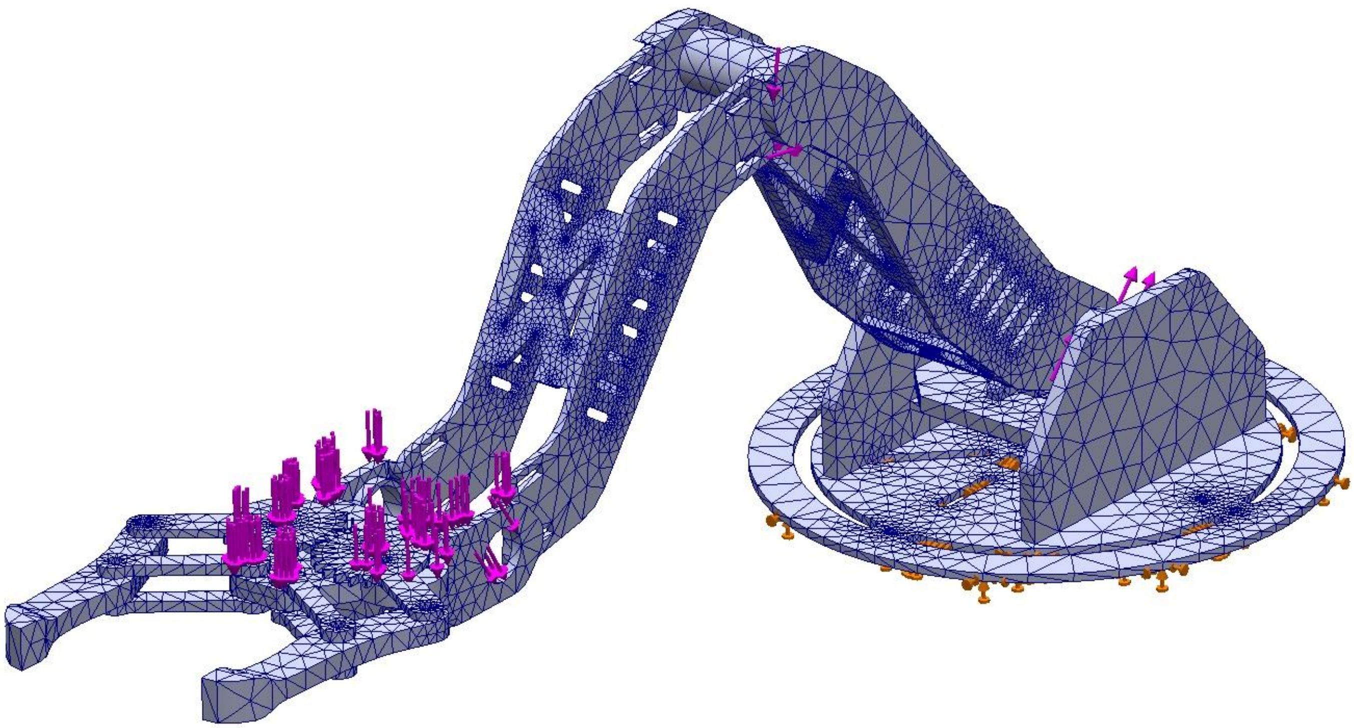
Regarding the servo mounts, we used custom 3D-printed parts designed in CAD software to precisely fit the servo shape while maintaining high strength. These mounts act as the main structural support between the linkages, ensuring the servo is firmly fixed and able to handle the applied loads.

In addition, Bearings were incorporated to allow smooth rotational motion between the linkages. It was accounted for this in the design of the servo's triple-holder, which connects the two linkages. The bearing was integrated into this holder to enable free movement between the linkages without friction to reduce the mechanical load on the servo.

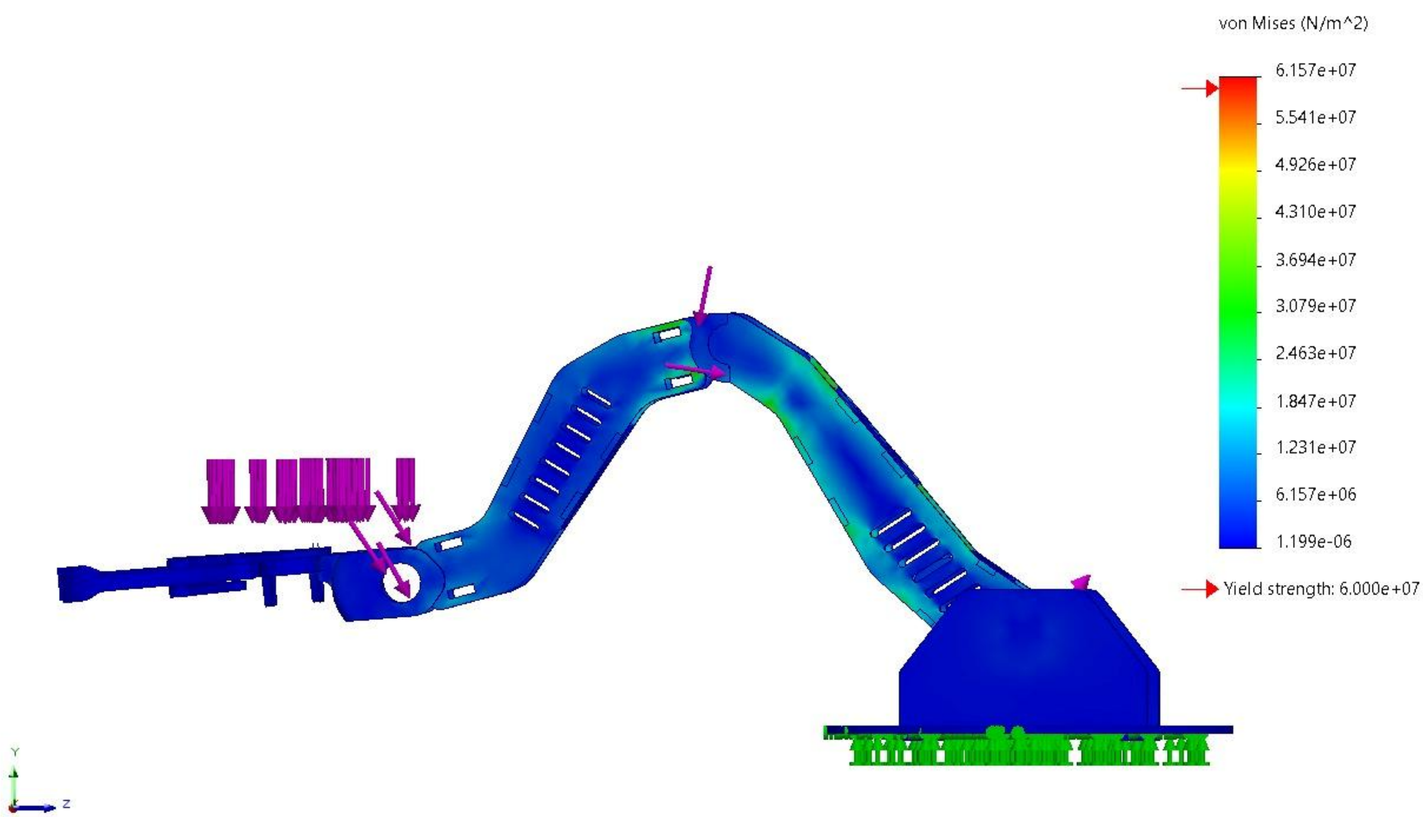


3.2. FEA

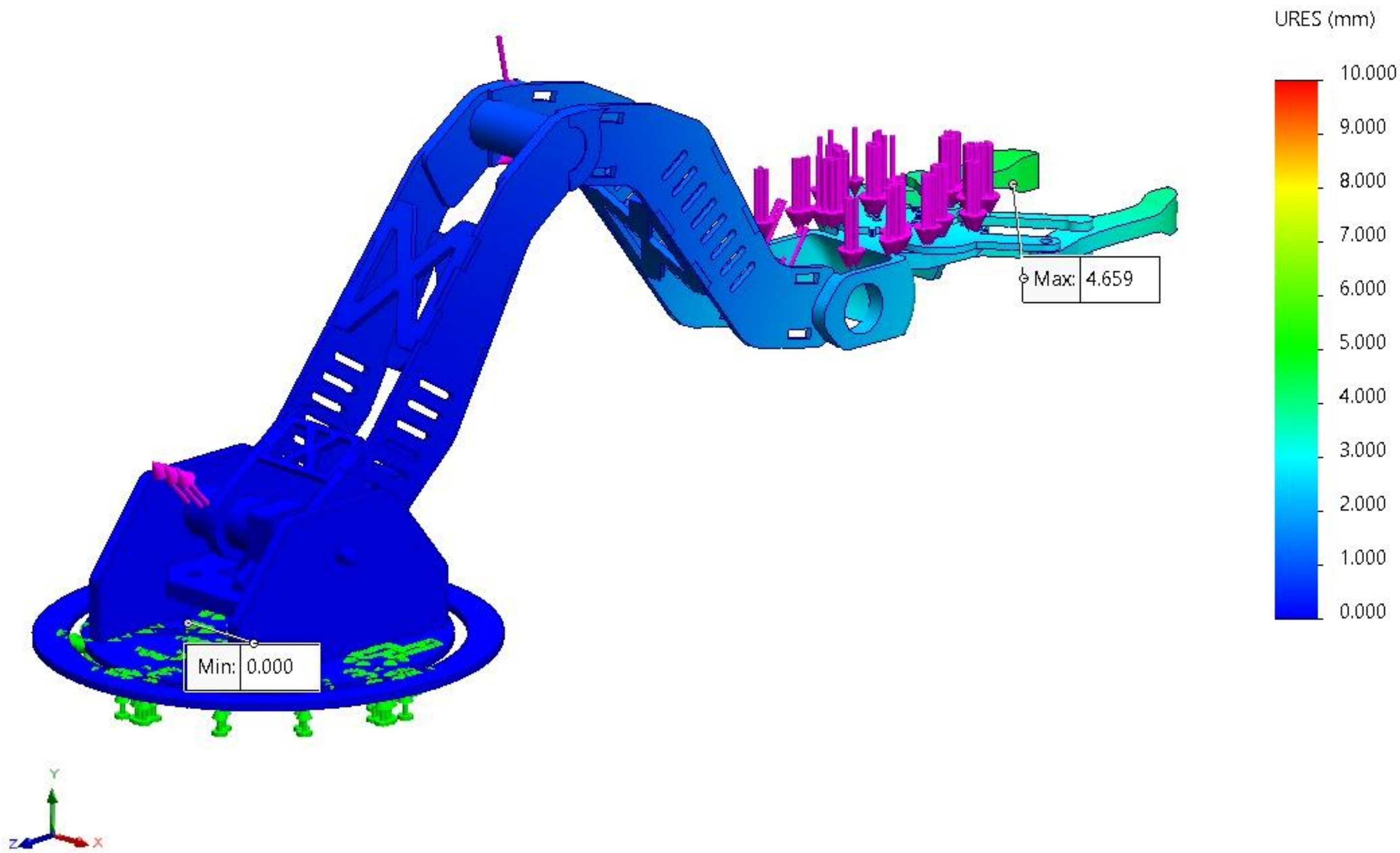
3.2.1. Force Distribution and Meshing



3.2.2. Stress Analysis



3.2.3. Displacement Analysis



4. Roboanalyzer

In this section, we used RoboAnalyzer, a 3D model-based simulation tool, to visualize and verify the forward and inverse kinematics of our robotic arm.

We modeled a 3-degree-of-freedom robotic manipulator consisting of three revolute joints (R-R-R), each represented as a link in RoboAnalyzer. The following links were defined:

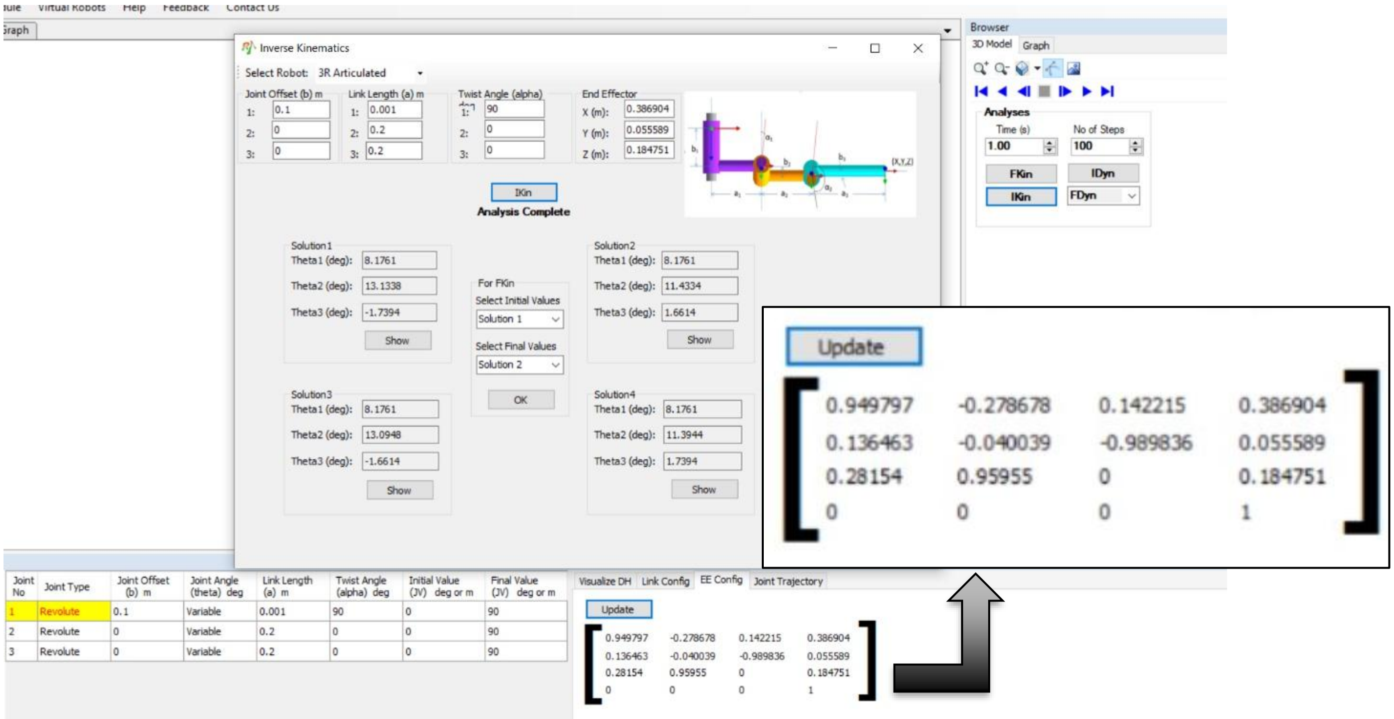
- Link 1**:** Base to first joint.
- Link 2**:** First joint to second joint.
- Link 3**:** Second joint to end effector

These values were based on the same dimensions used in our CAD and MATLAB calculations.

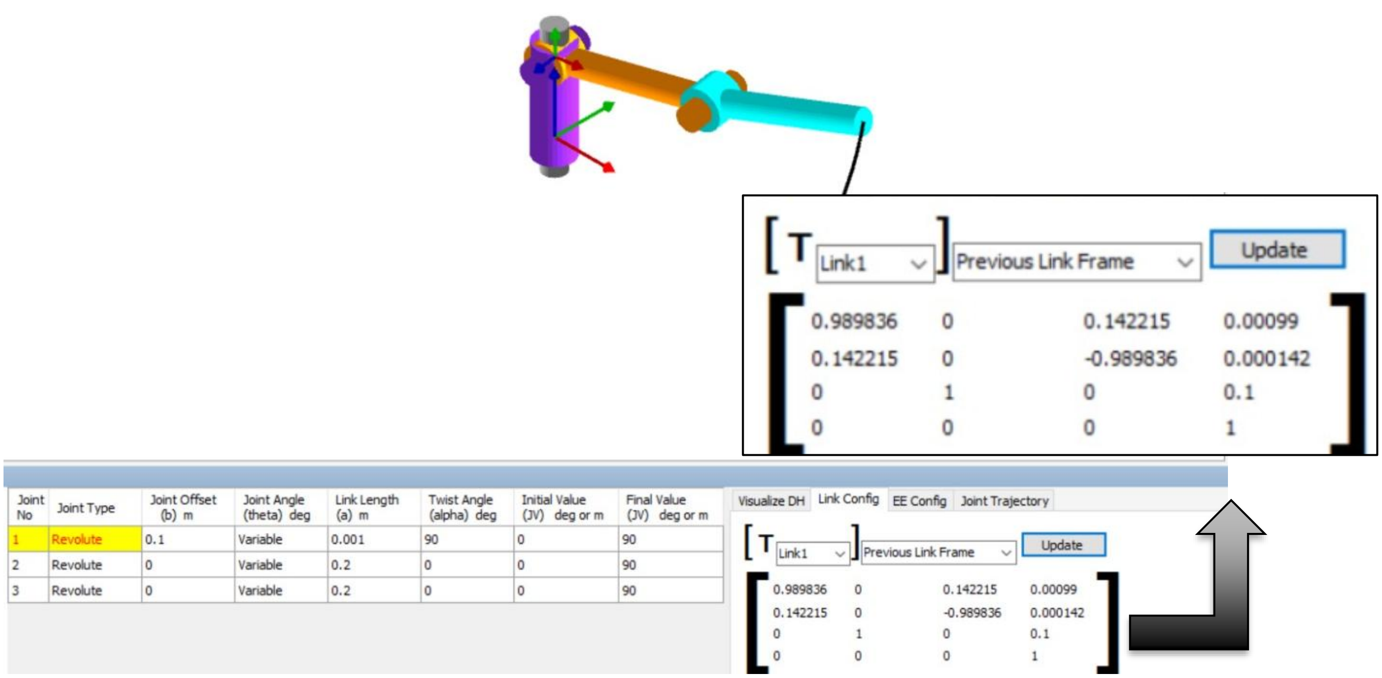
We performed simulations to observe:

- Forward Kinematics: By inputting joint angles (θ_1, θ_2), we confirmed the resulting end-effector position matches the theoretical calculations from MATLAB.
- Inverse Kinematics (via Trial): By varying end-effector positions, we observed the corresponding joint configurations, verifying the arm's range and limitations.
- Workspace Visualization: RoboAnalyzer helped us visualize the full reach of the manipulator and ensured that our design satisfies the task requirements.

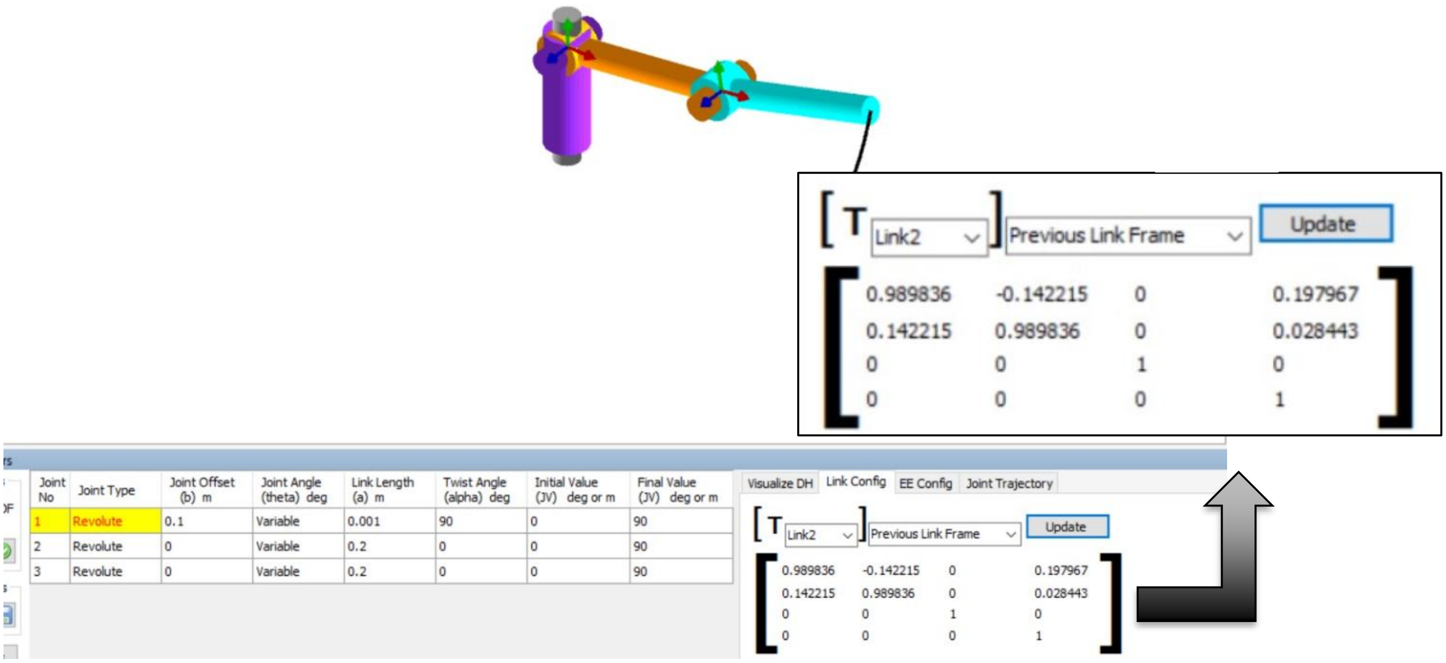
These simulations were crucial for validating our mathematical model before implementing the physical prototype.


Inverse Kinematics
Select Robot: 3R Articulated
Joint Offset (b) m
Link Length (a) m
Twist Angle (alpha)
End Effector
X (m): 0.386904
Y (m): 0.055589
Z (m): 0.184751
IKin
Analysis Complete
Solution1 Theta1 (deg): 8.1761 Theta2 (deg): 13.1338 Theta3 (deg): -1.7394
Solution2 Theta1 (deg): 8.1761 Theta2 (deg): 11.4334 Theta3 (deg): 1.6614
Solution3 Theta1 (deg): 8.1761 Theta2 (deg): 13.0948 Theta3 (deg): -1.6614
Solution4 Theta1 (deg): 8.1761 Theta2 (deg): 11.3944 Theta3 (deg): 1.7394
For FKin
Select Initial Values: Solution 1
Select Final Values: Solution 2
Update
0.949797 -0.278678 0.142215 0.386904
0.136463 -0.040039 -0.989836 0.055589
0.28154 0.95955 0 0.184751
0 0 0 1
Joint No | Joint Type | Joint Offset (b) m | Joint Angle (theta) deg | Link Length (a) m | Twist Angle (alpha) deg | Initial Value (JV) deg or m | Final Value (JV) deg or m
1 | Revolute | 0.1 | Variable | 0.001 | 90 | 0 | 90
2 | Revolute | 0 | Variable | 0.2 | 0 | 0 | 90
3 | Revolute | 0 | Variable | 0.2 | 0 | 0 | 90
Visualize DH
Link Config
EE Config
Joint Trajectory

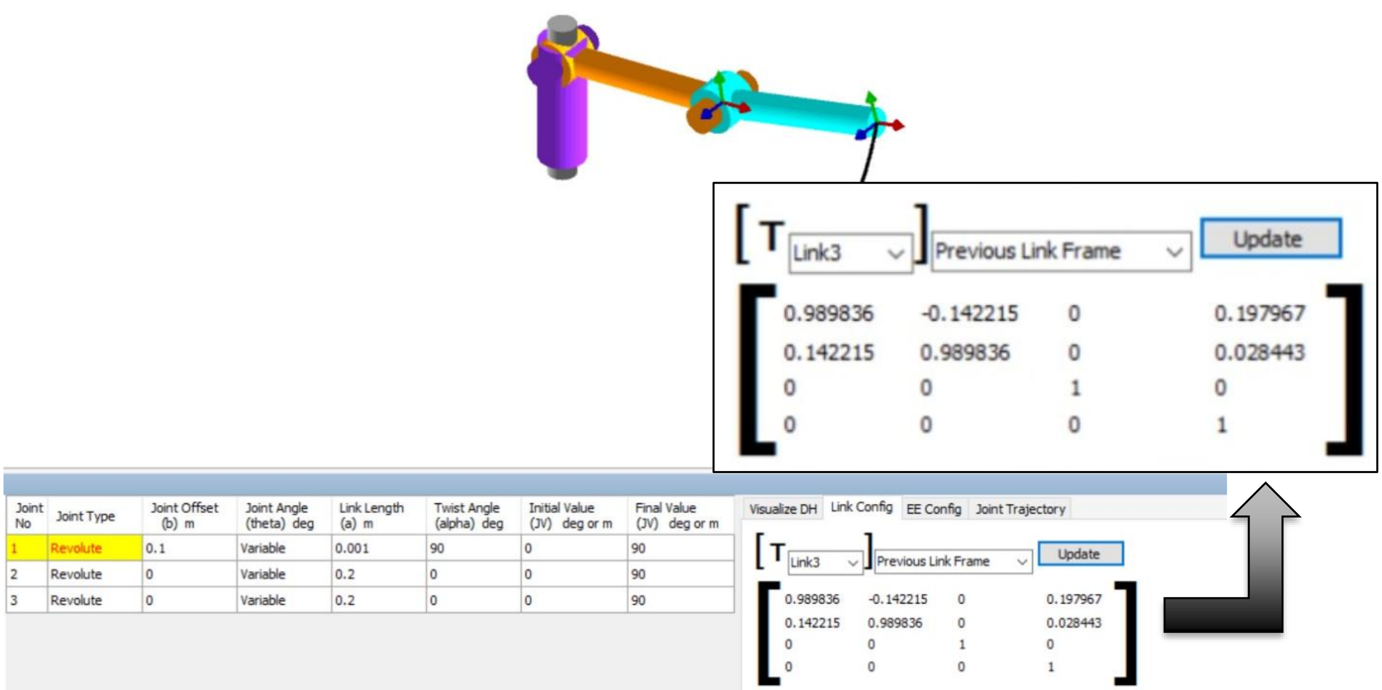
- **Link 1**


T Link1 Previous Link Frame
Update
0.989836 0 0.142215 0.00099
0.142215 0 -0.989836 0.000142
0 1 0 0.1
0 0 0 1
Joint No | Joint Type | Joint Offset (b) m | Joint Angle (theta) deg | Link Length (a) m | Twist Angle (alpha) deg | Initial Value (JV) deg or m | Final Value (JV) deg or m
1 | Revolute | 0.1 | Variable | 0.001 | 90 | 0 | 90
2 | Revolute | 0 | Variable | 0.2 | 0 | 0 | 90
3 | Revolute | 0 | Variable | 0.2 | 0 | 0 | 90
Visualize DH
Link Config
EE Config
Joint Trajectory

- **Link 2**


T Link2 Previous Link Frame Update
0.989836 -0.142215 0 0.197967
0.142215 0.989836 0 0.028443
0 0 1 0
0 0 0 1
Joint No | Joint Type | Joint Offset (b) m | Joint Angle (theta) deg | Link Length (a) m | Twist Angle (alpha) deg | Initial Value (JV) deg or m | Final Value (JV) deg or m
1 Revolute 0.1 Variable 0.001 90 0 90
2 Revolute 0 Variable 0.2 0 0 90
3 Revolute 0 Variable 0.2 0 0 90
Visualize DH
Link Config
EE Config
Joint Trajectory

- **Link 3**

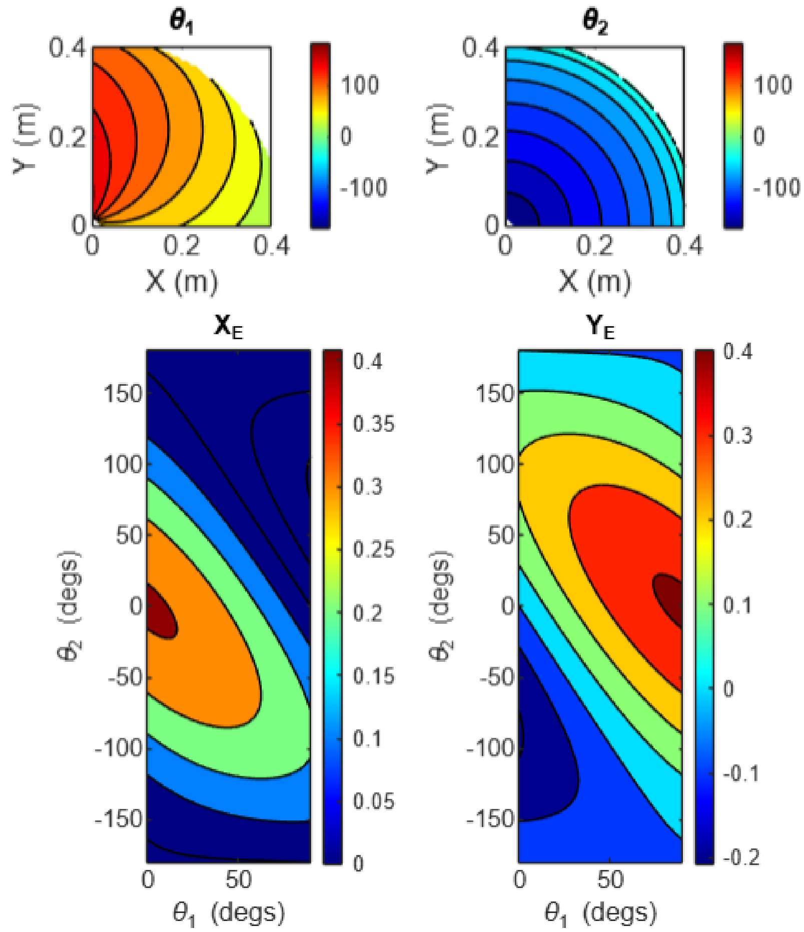

T Link3 Previous Link Frame Update
0.989836 -0.142215 0 0.197967
0.142215 0.989836 0 0.028443
0 0 1 0
0 0 0 1
Joint No | Joint Type | Joint Offset (b) m | Joint Angle (theta) deg | Link Length (a) m | Twist Angle (alpha) deg | Initial Value (JV) deg or m | Final Value (JV) deg or m
1 Revolute 0.1 Variable 0.001 90 0 90
2 Revolute 0 Variable 0.2 0 0 90
3 Revolute 0 Variable 0.2 0 0 90
Visualize DH
Link Config
EE Config
Joint Trajectory

5. MATLAB Calculations

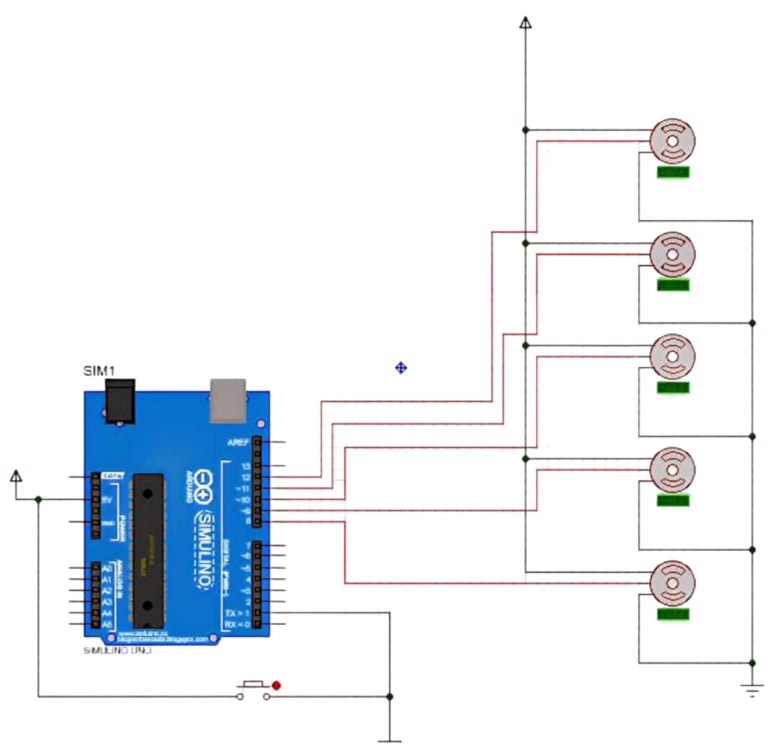
```
syms L_1  L_2 theta_1 theta_2 XE YE
L1 = 0.22;
L2 = 0.208;
XE_RHS = L_1*cos(theta_1) + L_2*cos(theta_1+theta_2)
YE_RHS = L_1*sin(theta_1) + L_2*sin(theta_1+theta_2)
XE_MLF = matlabFunction(XE_RHS,'Vars',[L_1 L_2 theta_1 theta_2]);
YE_MLF = matlabFunction(YE_RHS,'Vars',[L_1 L_2 theta_1 theta_2]);
t1_degs_row = linspace(0,90,100);
t2_degs_row = linspace(-180,180,100);
[tt1_degs,tt2_degs] = meshgrid(t1_degs_row,t2_degs_row);
tt1_rads = deg2rad(tt1_degs);
tt2_rads = deg2rad(tt2_degs);
X_mat = XE_MLF(L1,L2,tt1_rads,tt2_rads);
Y_mat = YE_MLF(L1,L2,tt1_rads,tt2_rads);
plot_XY_given_theta_2dof(tt1_degs,tt2_degs,X_mat,Y_mat,(L1+L2))
E_EQ = XE == XE_RHS;
YE_EQ = YE == YE_RHS;
S = solve([E_EQ YE_EQ], [theta_1 theta_2])
simplify(S.theta_1)
simplify(S.theta_2)
TH1_MLF{1} = matlabFunction(S.theta_1(1),'Vars',[L_1 L_2 XE YE]);
TH1_MLF{2} = matlabFunction(S.theta_1(2),'Vars',[L_1 L_2 XE YE]);
TH2_MLF{1} = matlabFunction(S.theta_2(1),'Vars',[L_1 L_2 XE YE]);
TH2_MLF{2} = matlabFunction(S.theta_2(2),'Vars',[L_1 L_2 XE YE]);
[xmat,ymat] = meshgrid(0:0.01:0.4,0:0.01:0.4);
tmp_th1_mat = TH1_MLF{1}(L1,L2,xmat,ymat);
tmp_th2_mat = TH2_MLF{1}(L1,L2,xmat,ymat);
tmp_th1_mat = rad2deg(tmp_th1_mat);
tmp_th2_mat = rad2deg(tmp_th2_mat);
th1_mat = NaN(size(tmp_th1_mat));
th2_mat = NaN(size(tmp_th2_mat));


tf_mat = imag(tmp_th1_mat) == 0;
th1_mat(tf_mat) = real(tmp_th1_mat(tf_mat));

tf_mat = imag(tmp_th2_mat) == 0;
th2_mat(tf_mat) = real(tmp_th2_mat(tf_mat));
plot_theta_given_XY_2dof(xmat,ymat,th1_mat,th2_mat)
the_J = jacobian([XE_RHS YE_RHS],[theta_1 theta_2])
function plot_theta_given_XY_2dof(X_mat,Y_mat,theta_1_mat_degs,...
                                  theta_2_mat_degs)

xlab_str = 'X (m)';
ylab_str = 'Y (m)';
```

```
figure;
hax(1) = subplot(1,2,1);
   contourf(X_mat, Y_mat, theta_1_mat_degs);
      clim(hax(1), [-180 180]);
      colormap(gca,'jet'); colorbar
      xlabel(xlab_str, 'Interpreter', 'tex');
      ylabel(ylab_str, 'Interpreter', 'tex');
      title(hax(1), '\theta_1', 'Interpreter', 'tex')
      axis('equal')
hax(2) = subplot(1,2,2);
   contourf(X_mat, Y_mat, theta_2_mat_degs);
      clim(hax(2), [-180 180]);
      colormap(gca,'jet'); colorbar
      xlabel(xlab_str, 'Interpreter', 'tex');
      ylabel(ylab_str, 'Interpreter', 'tex');
      title(hax(2), '\theta_2', 'Interpreter', 'tex')
      axis('equal')

end


function
plot_XY_given_theta_2dof(theta_1_mat_degs,theta_2_mat_degs,...

                                  X_mat,Y_mat,a_cmax)

xlab_str = '\theta_1 (degs)';
ylab_str = '\theta_2 (degs)';

figure;
hax(1) = subplot(1,2,1);
   contourf(theta_1_mat_degs, theta_2_mat_degs, X_mat);
      clim(hax(1), [0 a_cmax]);
      colormap(gca,'jet'); colorbar
      xlabel(xlab_str, 'Interpreter', 'tex');
      ylabel(ylab_str, 'Interpreter', 'tex');
      title(hax(1), 'X_E', 'Interpreter', 'tex')
hax(2) = subplot(1,2,2);
   contourf(theta_1_mat_degs, theta_2_mat_degs, Y_mat);
      clim(hax(1), [0 a_cmax]);
      colormap(gca,'jet'); colorbar
      xlabel(xlab_str, 'Interpreter', 'tex');
      ylabel(ylab_str, 'Interpreter', 'tex');
      title(hax(2), 'Y_E', 'Interpreter', 'tex')

end
```

6. Contours and Results

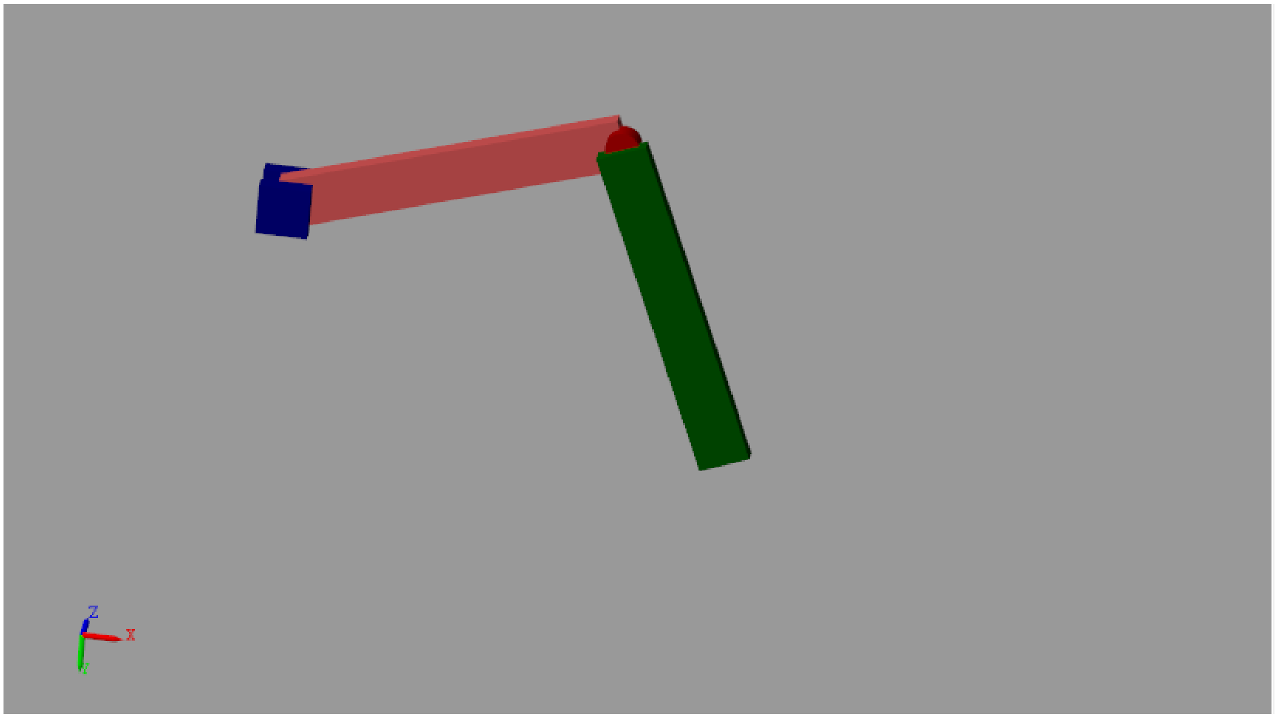


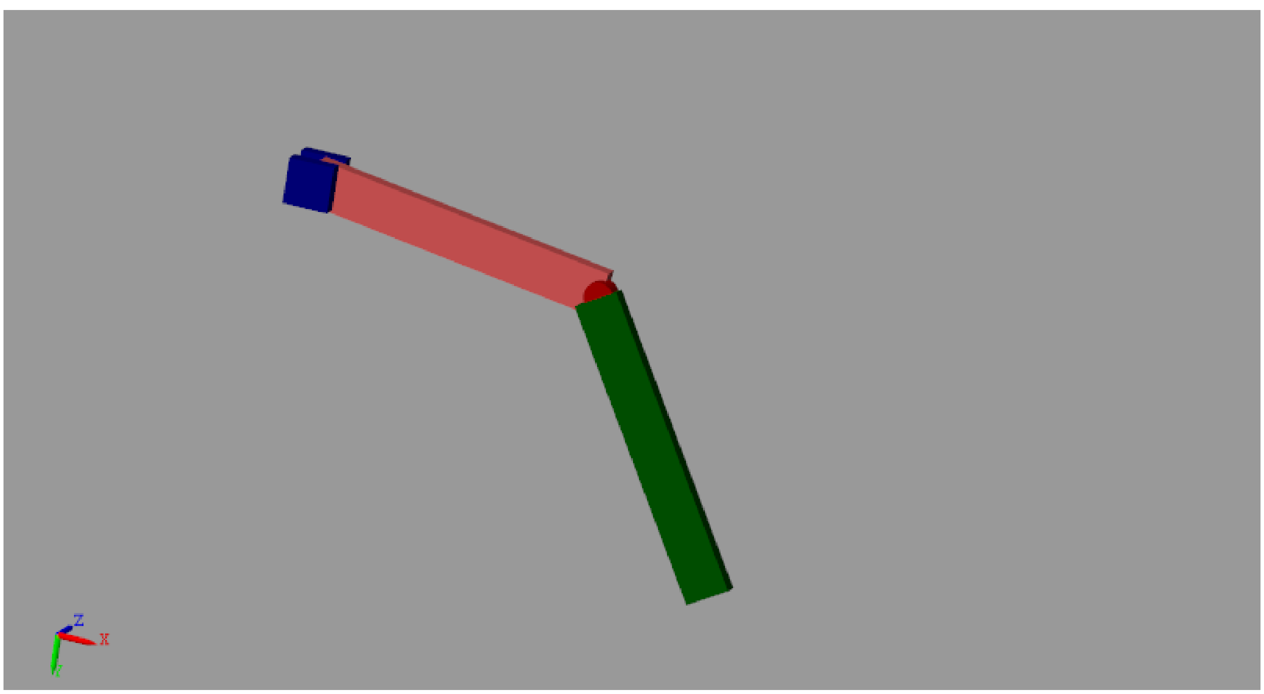
7. Wiring Diagram



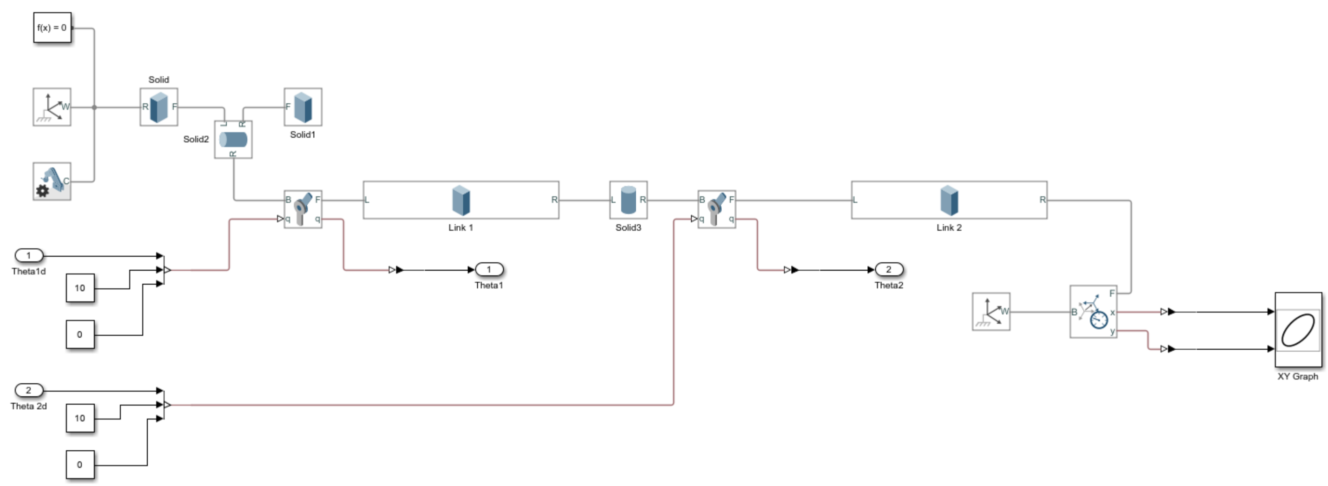
8. Simulink

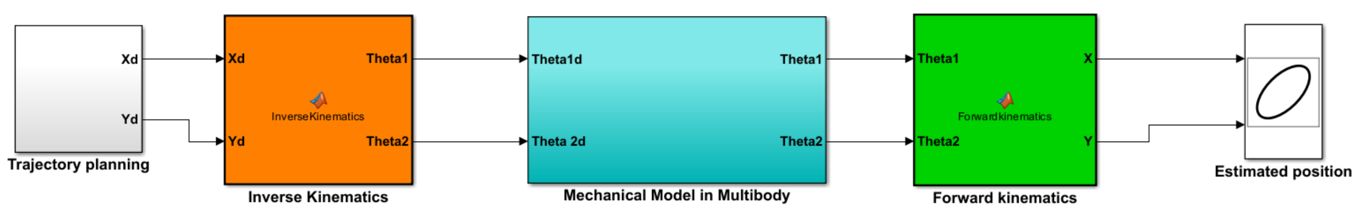
- Trajectory



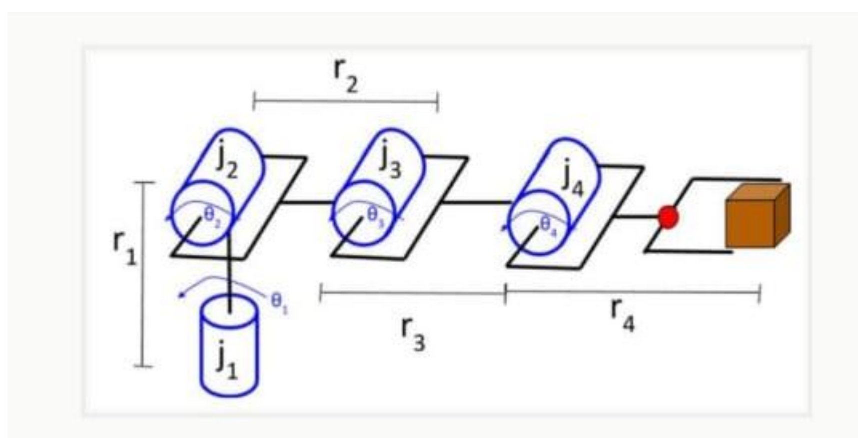


- Mechanical model in multibody





9. Torque calculations



L2=20 cm Mbox=0.3 Kg

L1=20.8 cm Mjoint=0.08 Kg

Mlink2=0.215 Kg

$$T2 = (L2 * Mbox) + (\frac{1}{2} * L2 * Mlink2)$$

$$T1 = ((L1 + L2) * Mbox) + \left(\left(L1 + \frac{L2}{2}\right) * Mlink2\right) + (L1 * Mjoint2) + \left(\frac{L1}{2} * Mlink2\right)$$

$$T1 = 22.1\ Kg/cm$$

$$T2 = 8.15\ Kg/cm$$