KULIYYAH OF ENGINEERING (MECHATRONICS)

MCTE 4352

ROBOTIC

ROBOTIC PROJECT

SECTION 1

LECTURER:

DR TANVEER SALAH

MOHAMAD DANIAL HUSAINI BIN NOORDIN	1711867

SCARA ROBOT DIMENSION

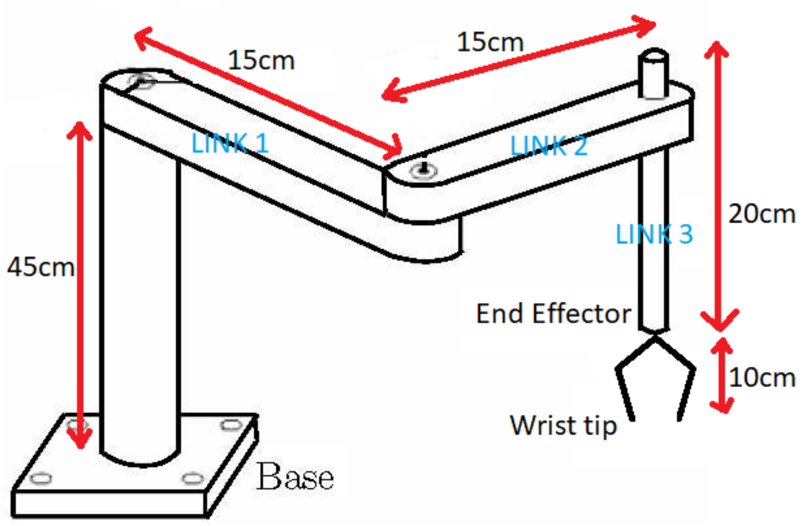

Figure 1: Dimension of SCARA Robot

FINDING DH PARAMETERS

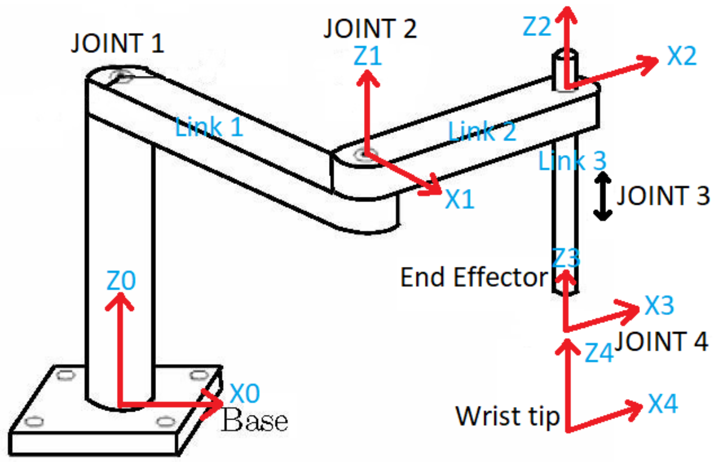

Figure 2: Assumption of x and z directions

Link	a_i	α_i	d_i	θ_i
0-1	a1	0	d1	Θ1*
1-2	a2	0	0	Θ2*
2-3	0	0	-d3*	0
3-4	0	0	0	Θ4*

Table 1: DH table

$$A_i = R_{z,\theta_i}\mathrm{Trans}_{z,d_i}\mathrm{Trans}_{x,a_i}R_{x,\alpha_i} \tag{3.10}$$

$$= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & 0 \\ s_{\theta_i} & c_{\theta_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

After we solve the calculation of the above matrix, we can find the final transformation matrix as below.

$$\begin{bmatrix} \cos(\theta_1+\theta_2+\theta_4) & -\sin(\theta_1+\theta_2+\theta_4) & 0 & a_2\cos(\theta_1+\theta_2)+a_1\cos(\theta_1) \\ \sin(\theta_1+\theta_2+\theta_4) & \cos(\theta_1+\theta_2+\theta_4) & 0 & a_2\sin(\theta_1+\theta_2)+a_1\sin(\theta_1) \\ 0 & 0 & 1 & d_1-d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

From the final transformation matrix, we can simplify this matrix in order to find the formula of x, y, z, θ_1 and θ_2.

$$x = a_2\cos(\theta_1+\theta_2) + a_1\cos(\theta_1) \text{.............} (1)$$

$$y = a_2\sin(\theta_1+\theta_2) + a_2\cos(\theta_1) \text{..............} (2)$$

$$z = d_1 - d_3 \text{...} (3)$$

$(1)^2+(2)^2$

$$x^2 + y^2 = {a_1}^2 + {a_2}^2 + 2a_1a_2\cos(\theta_2)$$

$$hence, \theta_2 = \pm\cos^{-1}\frac{x^2 + y^2 - {a_1}^2 - {a_2}^2}{2a_1a_2}$$

$$Next,$$

$$\frac{x}{\cos(\theta_1)} = \frac{a_2\cos(\theta_1+\theta_2) + a_1\cos(\theta_1)}{\cos(\theta_1)}$$

$$\frac{x}{\cos(\theta_1)} = a_1 + a_2\cos(\theta_2) - a_2\sin(\theta_2)\tan(\theta_1)$$

$$\frac{y}{\cos(\theta_1)} = a_1\tan(\theta_1) + a_2\cos(\theta_2)\tan(\theta_1) + a_2\sin(\theta_2)$$

$$\frac{x}{y} = \frac{a_1+a_2\cos(\theta_2)-a_2\sin(\theta_2)\tan(\theta_1)}{a_1\tan(\theta_1)+a_2\cos(\theta_2)\tan(\theta_1)+a_2\sin(\theta_2)}$$

$$\tan(\theta_1) = \frac{a_1y + a_2y\cos(\theta_2) - a_2x\sin(\theta_2)}{a_1x + a_2x\cos(\theta_2) + a_2y\sin(\theta_2)}$$

After we have found all the formula, we can do the forward kinematics and inverse kinematics techniques in order to find the value of x, y, z, θ_1 and θ_2. We can simply find the value of x, y, z, θ_1 and θ_2 in excel by inserting all the formula that we got.

x	y	z	θ1	θ2
30	0	0	0	0
21.2132	21.2132	0	0.785398163	0
-21.2132	21.2132	20	2.35619449	0
-21.2132	-21.2132	20	3.926990817	0
21.2132	-21.2132	0	5.497787144	0
30	-7.4E-15	0	6.283185307	0

Table 2: Forward Kinematics (table from excel)

x	y	z	θ1	θ2
30	0	0	0	0
21.2132	21.2132	0	44.96739114	0.001138
-21.2132	21.2132	20	-45.03260886	0.001138
-21.2132	-21.2132	20	44.96739114	0.001138
21.2132	-21.2132	20	-45.03260886	0.001138
30	0	0	0	0

Table 3: Inverse Kinematics (table from excel)

From the tables above, we can see the close loop path of the SCARA Robot. The value of θ in Table 2 is written in radians (0.785398163 radians = 45 degrees, 2.35619449 radians = 135 degrees, 3.926990817 radians = 225 degrees, 5.497787144 radians = 315 degrees, and 6.283185307 radians = 360 degrees) while the value of θ in Table 3 is written in degree. If we compare all the values in

Table 2 and Table 3, we can see that all the value is same. However, we might see that the value of θ_1 is different from both tables. Actually, all the values of θ_1 in Table 2 and Table 3 is same. The value of θ_1 in Table 3 is written in reference angle.

MATLAB CODING

```
%% Assign d,a,alpha and theta
%Mohamad Danial Husaini bin Noordin (1711867)
% a1=15, a2=15, d3=20
L1 = Link('d',0,'a',15,'alpha',0);
L2 = Link('d',0,'a',15,'alpha',0);
L3 = Link([0,20,0,0,1],'standard');
L3.qlim = [0 20]; %to limit the travel distance of link 3
L4 = Link('d',0,'a',0,'alpha',0);
Rob = SerialLink ([L1 L2 L3 L4],'name','SCARAROBOT');
Rob.plot([0 0 0 0],'workspace',[-60 60 -60 60 -60 60]);
Rob.teach

%% take matrix value from excel file 1711867.xlsx
inputdata = readmatrix("1711867.xlsx", "sheet", "fkine_scara", "Range",
"A2:E7");

%% transformation
q0 = [0 0 0 0];%initial position x=30 ,y=0, z=0
T1 = transl(inputdata(1,1),inputdata(1,2),inputdata(1,3));
T2 = transl(inputdata(2,1),inputdata(2,2),inputdata(2,3));
T3 = transl(inputdata(3,1),inputdata(3,2),inputdata(3,3));
T4 = transl(inputdata(4,1),inputdata(4,2),inputdata(4,3));
T5 = transl(inputdata(5,1),inputdata(5,2),inputdata(5,3));
T6 = transl(inputdata(6,1),inputdata(6,2),inputdata(6,3));

%% inverse kinematics
q1 = Rob.ikine(T1,[0 0 0 0],[1,1,1,0,0,0]);
q2 = Rob.ikine(T2,[0 0 0 0],[1,1,1,0,0,0]);
q3 = Rob.ikine(T3,[0 0 0 0],[1,1,1,0,0,0]);
q4 = Rob.ikine(T4,[0 0 0 0],[1,1,1,0,0,0]);
q5 = Rob.ikine(T5,[0 0 0 0],[1,1,1,0,0,0]);
q6 = Rob.ikine(T6,[0 0 0 0],[1,1,1,0,0,0]);

%% forward kinematics
q1_T1 = Rob.fkine(q1);
q2_T2 = Rob.fkine(q2);
q3_T3 = Rob.fkine(q3);
q4_T4 = Rob.fkine(q4);
q5_T5 = Rob.fkine(q5);
q6_T6 = Rob.fkine(q6);

%% simulation of the robot
t = [0: .5:2]';

% Path 1 = q0-->q1
path1 = jtraj(q0,q1,t);
Rob.plot(path1,'workspace',[-60 60 -60 60 -60 60]);
pause(0.02);

% Path 2 = q1-->q2
path2 = jtraj(q1,q2,t);
```

```
Rob.plot(path2,'workspace',[-60 60 -60 60 -60 60]);
pause(0.02);

% Path 1 = q2-->q3
path3 = jtraj(q2,q3,t);
Rob.plot(path3,'workspace',[-60 60 -60 60 -60 60]);
pause(0.02);

% Path 1 = q3-->q4
path4 = jtraj(q3,q4,t);
Rob.plot(path4,'workspace',[-60 60 -60 60 -60 60]);
pause(0.02);

% Path 1 = q4-->q5
path5 = jtraj(q4,q5,t);
Rob.plot(path5,'workspace',[-60 60 -60 60 -60 60]);
pause(0.02);

% Path 1 = q5-->q6
path6 = jtraj(q5,q6,t);
Rob.plot(path6,'workspace',[-60 60 -60 60 -60 60]);
pause(0.02);

% Path 1 = q6-->q7
path7 = jtraj(q6,q1,t);
Rob.plot(path7,'workspace',[-60 60 -60 60 -60 60]);
pause(0.02);

% Path 1 = q1-->q0
%we can see that the path will go back to q0. So, it is a close loop path.
path8 = jtraj(q1,q0,t);
Rob.plot(path8,'workspace',[-60 60 -60 60 -60 60]);
pause(0.02);
```

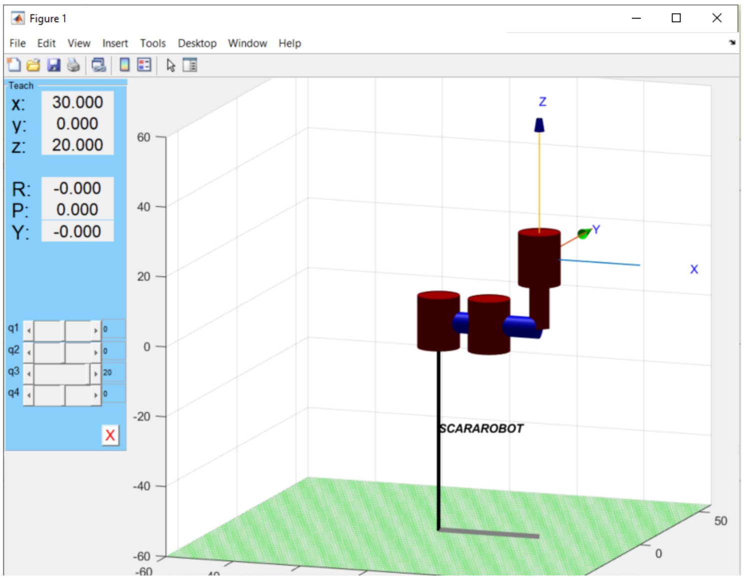

Figure 3: Simulation of SCARA Robot using MATLAB Robotics Toolbox

DISCUSSION

The MATLAB coding is divided into three parts. The first part is for the assignment of the length of all links. The second part is for determining the close loop path of the SCARA robot. The third part is for the simulation of the SCARA robot.

In the first part, the link function will take values of theta, d, a and alpha from DH table as shown in Table 1. I assign a1=15cm, a2=15cm and d3=20cm. Then, I use qlim function for link 3 in order to limit the travel distance of link 3 from 0cm to 20cm. After that, I use SerialLink function to combine all the links that I have assign and give name to the SCARA robot. Plot function is used to plot the graph of the SCARA robot. Next, I use readmatrix function to make the MATLAB take the data from the excel for creating the path of the robot.

In the second part, I use translation, inverse kinematics and forward kinematics function. Translation function will receive the data from inputdata variable. Inverse kinematics function uses numerical solution method. Hence, I have to give the initial guess of the solution in an array form. The array in inverse kinematics function is used to initialize the angle and the mask array. The forward kinematics function is used to check whether I can get the same coordinate points that I used for the IK solution for the RTB toolbox.

In the third part, I show all the paths of my robot. I have 8 paths. The paths will start at x=30cm, y=0cm and z=0cm and it will also end in the same position. So, it is a close loop path since it ends at the same position as it started.