```
clc;
clear all;
close all;
```

Defining DH Parameters

```
syms l1 l2 lb theta1 theta2 d3 ao
l1_dh = [theta1 lb+2*ao l1-2*ao 0]
```

l1_dh =

$$\begin{bmatrix} \theta_1 & \mathrm{lb}+2\,\mathrm{ao} & l_1-2\,\mathrm{ao} & 0 \end{bmatrix}$$

```
l2_dh = [theta2 ao l2-ao 0]
```

l2_dh =

$$\begin{bmatrix} \theta_2 & \mathrm{ao} & l_2-\mathrm{ao} & 0 \end{bmatrix}$$

```
l3_dh = [0 -d3+ao ao 0]
```

l3_dh =

$$\begin{bmatrix} 0 & -d_3+\mathrm{ao} & \mathrm{ao} & 0 \end{bmatrix}$$

Making Symbolic A Matrix

```
syms theta d a alpha
A = [cos(theta) -sin(theta)*cos(alpha) sin(theta)*sin(alpha) a*cos(theta);
    sin(theta) cos(theta)*cos(alpha) -cos(theta)*sin(alpha) a*sin(theta);
    0 sin(alpha) cos(alpha) d;
    0 0 0 1]
```

A =

$$\begin{bmatrix} \cos(\theta) & -\cos(\alpha)\sin(\theta) & \sin(\alpha)\sin(\theta) & a\cos(\theta) \\ \sin(\theta) & \cos(\alpha)\cos(\theta) & -\sin(\alpha)\cos(\theta) & a\sin(\theta) \\ 0 & \sin(\alpha) & \cos(\alpha) & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

```
A1 = simplify(subs(A,[theta,d,a,alpha],l1_dh));
A2 = simplify(subs(A,[theta,d,a,alpha],l2_dh));
A3 = simplify(subs(A,[theta,d,a,alpha],l3_dh));
T = A1*A2*A3
```

T =

$$\begin{bmatrix} \cos(\theta_1)\cos(\theta_2)-\sin(\theta_1)\sin(\theta_2) & -\cos(\theta_1)\sin(\theta_2)-\cos(\theta_2)\sin(\theta_1) & 0 & \mathrm{ao}\ (\cos(\theta_1)\cos(\theta_2)-\sin(\theta_1)\sin(\theta \\ \cos(\theta_1)\sin(\theta_2)+\cos(\theta_2)\sin(\theta_1) & \cos(\theta_1)\cos(\theta_2)-\sin(\theta_1)\sin(\theta_2) & 0 & \mathrm{ao}\ (\cos(\theta_1)\sin(\theta_2)+\cos(\theta_2)\sin(\theta \\ 0 & 0 & 1 & \\ 0 & 0 & 0 & \end{bmatrix}$$

Modelling the Robot from DH Parameters

```
L(1) =  Link(double(subs(l1_dh,[l1,l2,lb,theta1,theta2,d3,ao],[1,1,1,0,0,1.2,0.05])),'s
L(1).qlim = pi/180 * [-90 90];
```

```
L(2) =  Link(double(subs(l2_dh,[l1,l2,lb,theta1,theta2,d3,ao],[1,1,1,0,0,1.2,0.05])),'s
L(1).qlim = pi/180 * [-90 90];
L(3) =  Link([double(subs(l3_dh,[l1,l2,lb,theta1,theta2,d3,ao],[1,1,1,0,0,1.2,0.05])),1
```

```
L =
Revolute(std):  theta=q1   d=1.1          a=0.9         alpha=0          offset=0
Revolute(std):  theta=q2   d=0.05         a=0.95        alpha=0          offset=0
Prismatic(std): theta=0              d=q3 a=0.05        alpha=0          offset=0
```

```
L(3).qlim = [0 2];
scara_robot = SerialLink(L);
scara_robot.name = 'SCARA Robot';
scara_robot.plot([0 0 0],'workspace',[-2 2 -2 2 0 2])
scara_robot.teach
```