TAMPERE UNIVERSITY OF TECHNOLOGY

Robot Modelling, Control and Programming

Assignment 01

Group member:

Farid Khosravi 267964

Mehdi Mahmoodpour 267958

Palash Halder 267962

<u>ASE – 9406Autumn 2016</u>

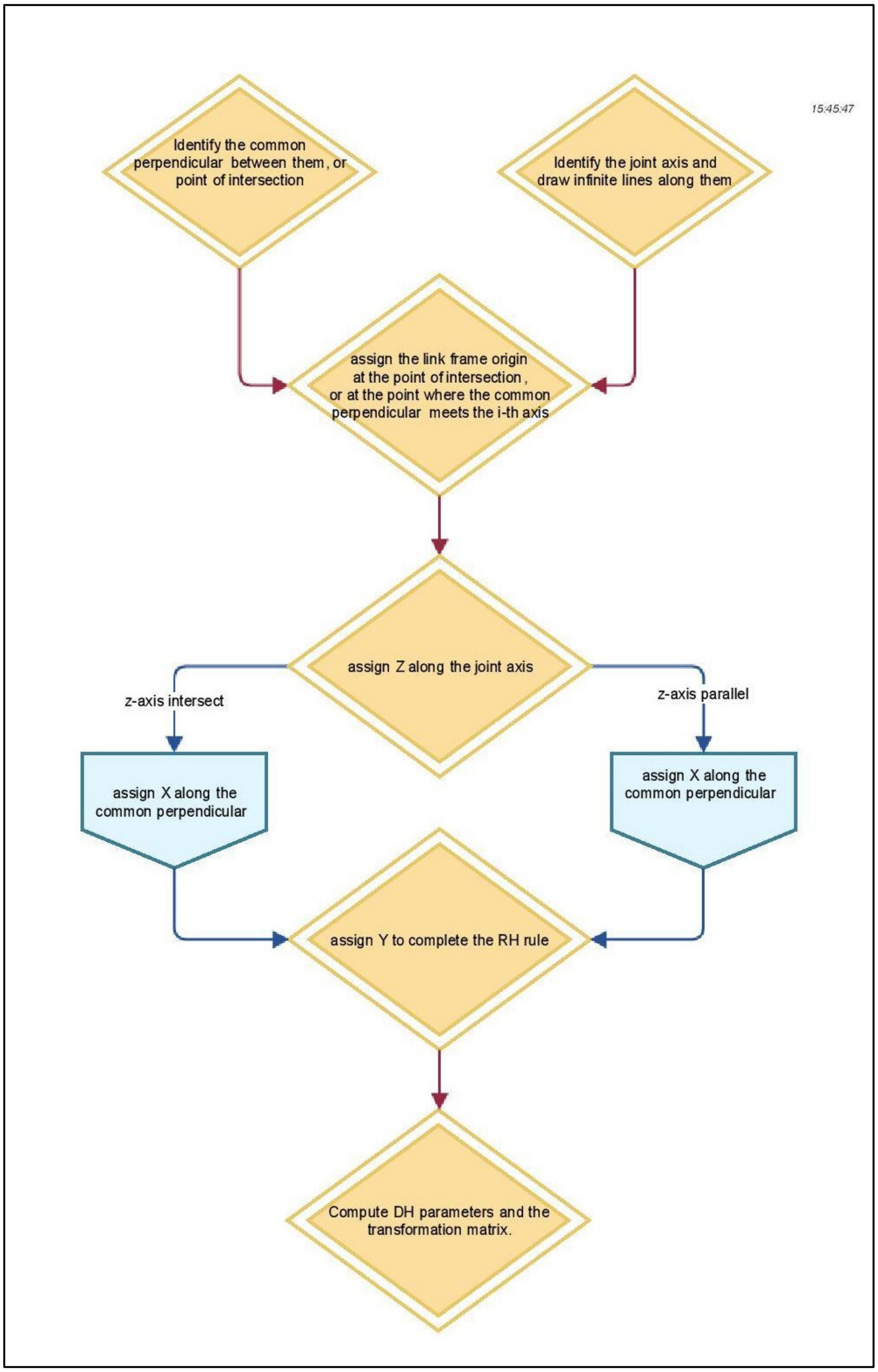
Contents

Objective: Design and analyze an automatic manipulator (Puma)

Standard DH Parameters:

The system is Puma and consists 6 joints and 6 links:

- The procedure to assign frames of links is shown in below flow chart:



Choices made for assigning frames:

- In this assignment, we chose the base frame coincident with the frame number 1. This allowed easier computation of DH parameters.
- We chose link frames so that to have maximum distance between joints to calculate DH parameters much easier.
- Below diagram depicts the link frames on the robot that have been selected for this assignment.

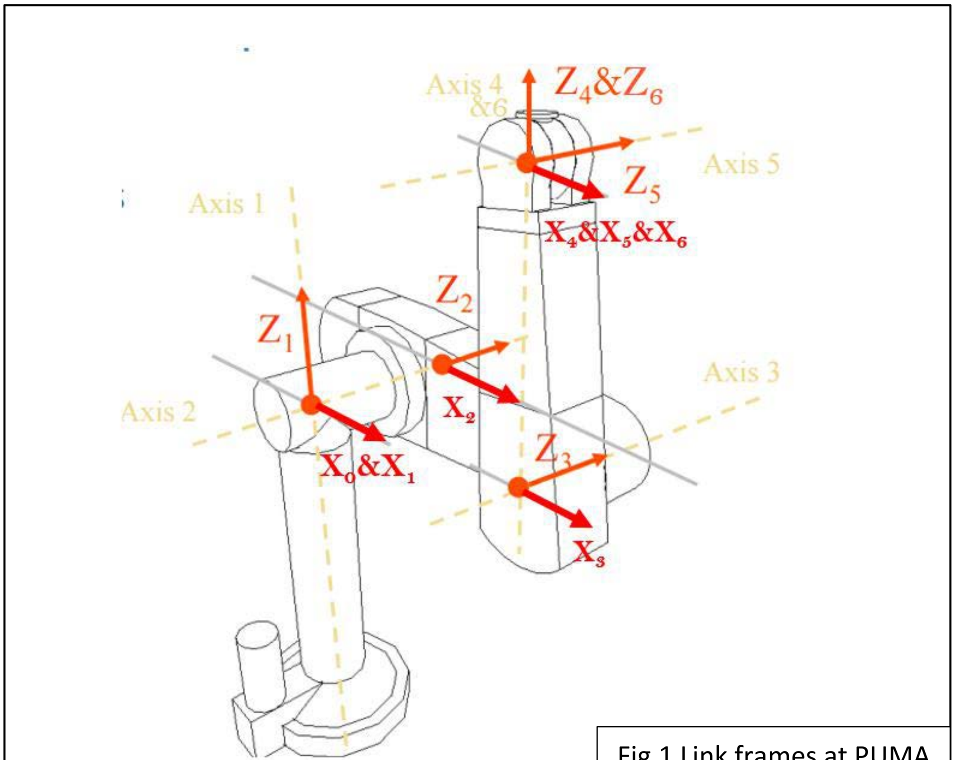


Fig.1 Link frames at PUMA

Options:

To assign frames we had other alternatives for X and Z axes. For example for Joint number 1 it was possible to select X_1 axis in opposite direction and as a result other X axes should be selected in opposite direction. Also to assign common perpendicular between Z_2 & Z_3 There are infinite choices to be selected. Here we chose the link frames at the end point of each link along the axis of rotation. This choice allowed to consider the maximum distance between the joints.

Modified DH parameter & Transformation Matrix:

- Following matrix shows the DH parameters that have been defined for puma robot.

```
+---+-----------+-----------+-----------+-----------+-----------+
| j |     theta |         d |         a |     alpha |    offset |
+---+-----------+-----------+-----------+-----------+-----------+
|  1|         q1|          0|          0|          0|          0|
|  2|         q2|       0.44|          0|     -1.571|          0|
|  3|         q3|      -0.12|       1.05|          0|          0|
|  4|         q4|       1.23|          0|      1.571|          0|
|  5|         q5|          0|          0|     -1.571|          0|
|  6|         q6|          0|          0|      1.571|          0|
+---+-----------+-----------+-----------+-----------+-----------+
```

- The transformation matrixes for puma robot with considered DH parameters are shown in the following.

```
T01 =

[ cos(theta1), -sin(theta1), 0, 0]
[ sin(theta1),  cos(theta1), 0, 0]
[           0,            0, 1, 0]
[           0,            0, 0, 1]


T12 =

[ cos(theta2), -sin(theta2),  0,   0]
[           0,            0, -1, -d2]
[ sin(theta2),  cos(theta2),  0,   0]
[           0,            0,  0,   1]


T23 =

[ cos(theta3), -sin(theta3), 0,  a2]
[ sin(theta3),  cos(theta3), 0,   0]
[           0,            0, 1, -d3]
[           0,            0, 0,   1]


T34 =

[ cos(theta4), -sin(theta4),  0,   0]
[           0,            0, -1, -d4]
[ sin(theta4),  cos(theta4),  0,   0]
[           0,            0,  0,   1]


T45 =

[  cos(theta5), -sin(theta5), 0, 0]
[            0,            0, 1, 0]
[ -sin(theta5), -cos(theta5), 0, 0]
[            0,            0, 0, 1]


T56 =

[ cos(theta6), -sin(theta6),  0, 0]
[           0,            0, -1, 0]
[ sin(theta6),  cos(theta6),  0, 0]
[           0,            0,  0, 1]
```

- The transformation matrix from joint 0 to joint 6 can be derived by multiplying matrixes together :

`T06 = T01*T12*T23*T34*T45*T56`

To check the solution, the element r33 of transformation matrix as a sample is compared with given matrix in inverse kinematic lecture and as it can be seen it is correct.

$$ {}^{0}_{6}T = {}^{0}_{1}T\,{}^{1}_{6}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ r_{33} = s_{23} c_4 s_5 - c_{23} c_5 $$

```
ans =

sin(theta2 + theta3)*cos(theta4)*sin(theta5) - cos(theta2 +
theta3)*cos(theta5)
```

- The homogenous transformation matrix for the robot when all the joint angles are set to zero is given below.

```
ans =

    1.0000         0         0    1.0800
         0    1.0000         0    0.3200
         0         0    1.0000    1.3650
         0         0         0    1.0000
```

As it can be seen the position of X, Y and Z are: 1.08, 0.32 and 1.365 respectively.

Robotics Toolbox:

- To model Puma robot by "Robotic Toolbox" six links of robot should be defined by `Link` command.

```
            %             theta    d         a     alpha

L(1) = Link([  0       0        0        0       0], 'modified');
L(2) = Link([  0       0.44     0      -pi/2     0], 'modified');
L(3) = Link([  0      -0.12    1.05      0       0], 'modified');
L(4) = Link([  0       1.23     0       pi/2     0], 'modified');
L(5) = Link([  0       0        0      -pi/2     0], 'modified');
L(6) = Link([  0       0        0       pi/2     0], 'modified');
```

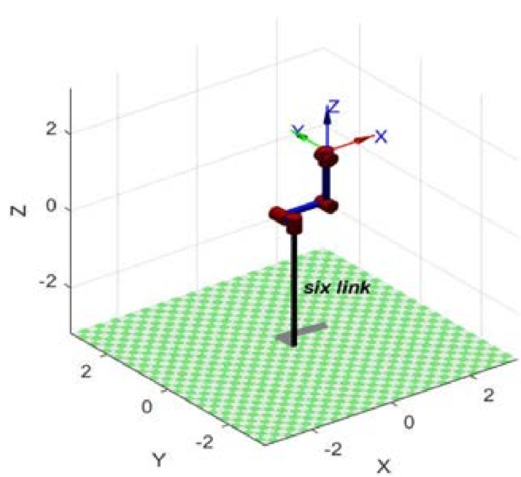
In the next step the robot links of robot should be attached to each other by

```
six_link = SerialLink(L,'name','six link')
```

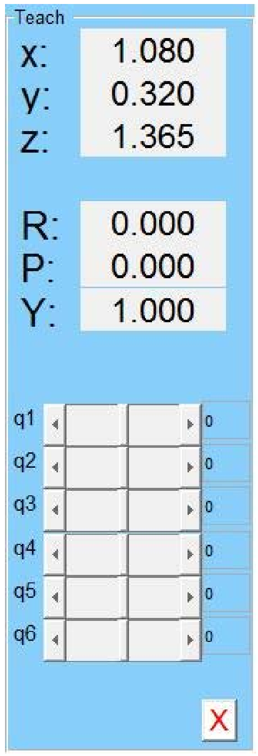
Then robot has been plotted by

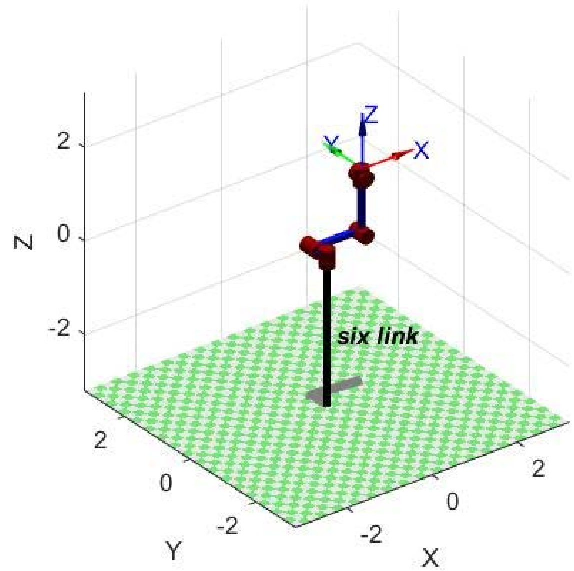
```
six_link.plot([0 0 0 0 0 0])
```

and the plot is shown below.



There is a possibility with `teach()` command to drive the robot physically and graphically by changing the angle of joints manually.

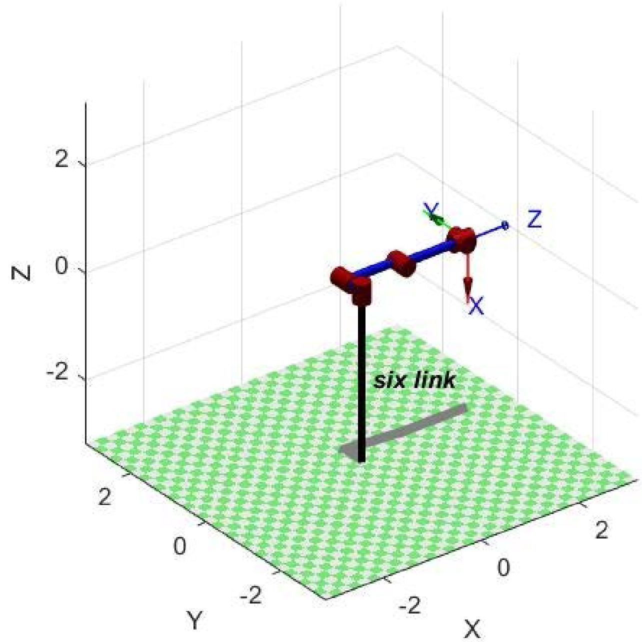




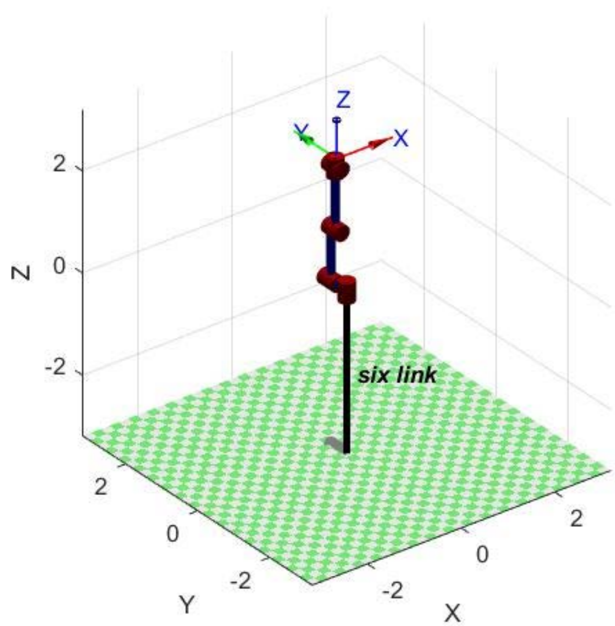
Forward Kinematic:

- To add tool to the robot, the command of `six_link.tool=transl([0.03,0,0.135])` was employed in assignment. The center of the gripping jaws is 135mm from the end of the robot and it is along the Z axis of the joint 6. Moreover, it drops down 30mm along to X axis.

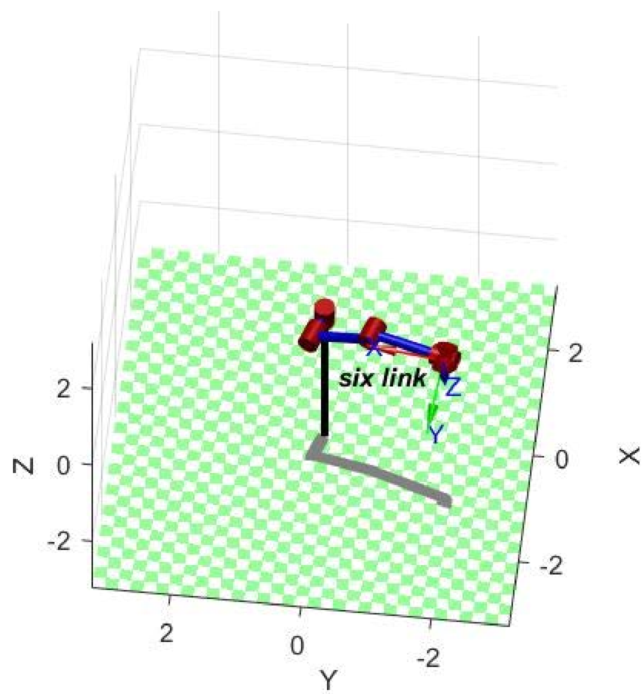
```
%Plot the robot in straight horizontal configuration
six_link.plot([0 0 pi/2 0 0 0])
six_link.fkine([0 0 pi/2 0 0 0])

ans =

    0.0000   -0.0000    1.0000    2.4150
    0.0000    1.0000    0.0000    0.3200
   -1.0000    0.0000    0.0000   -0.0300
         0         0         0    1.0000
```




```
%Plot the robot in straight vertical configuration
six_link.plot([0 -pi/2 pi/2 0 0 0])
six_link.fkine([0 -pi/2 pi/2 0 0 0])

ans =

    1.0000         0         0    0.0300
         0    1.0000         0    0.3200
         0         0    1.0000    2.4150
         0         0         0    1.0000
```



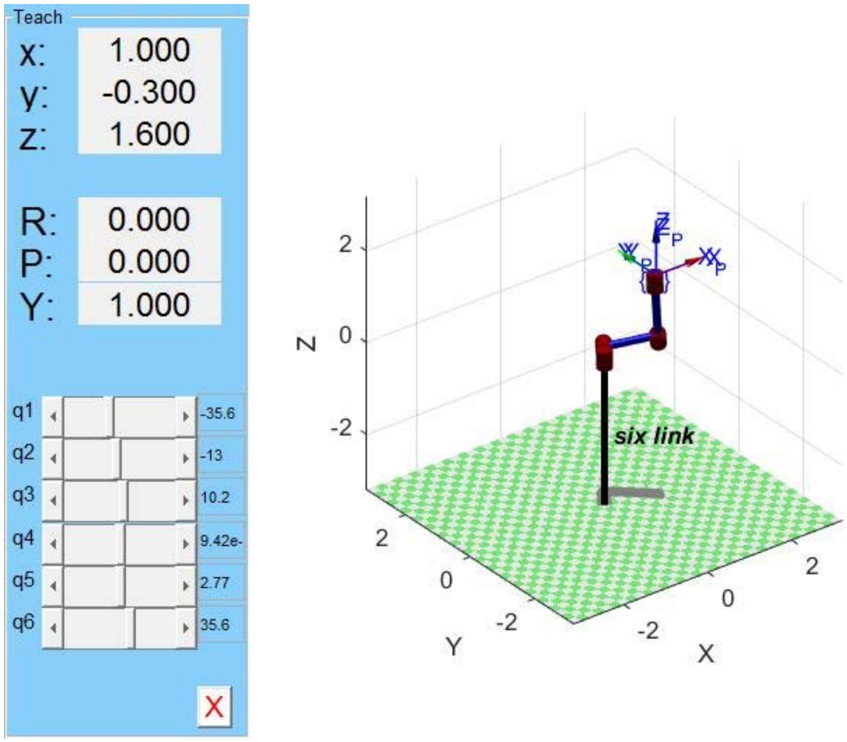
```
%Plot the robot in working position configuration
six_link.plot([-93 3.6 108 7.2 20 0])
six_link.fkine([-93 3.6 108 7.2 20 0])

ans =

   -0.0123   -0.5954   -0.8034   -1.1005
    0.9835    0.1379   -0.1173   -1.9409
    0.1806   -0.7915    0.5838    0.6392
         0         0         0    1.0000
```



Inverse Kinematic:

- To solve inverse kinematic `six_link.ikunc` is used in this assignment.
- For the position A we have :

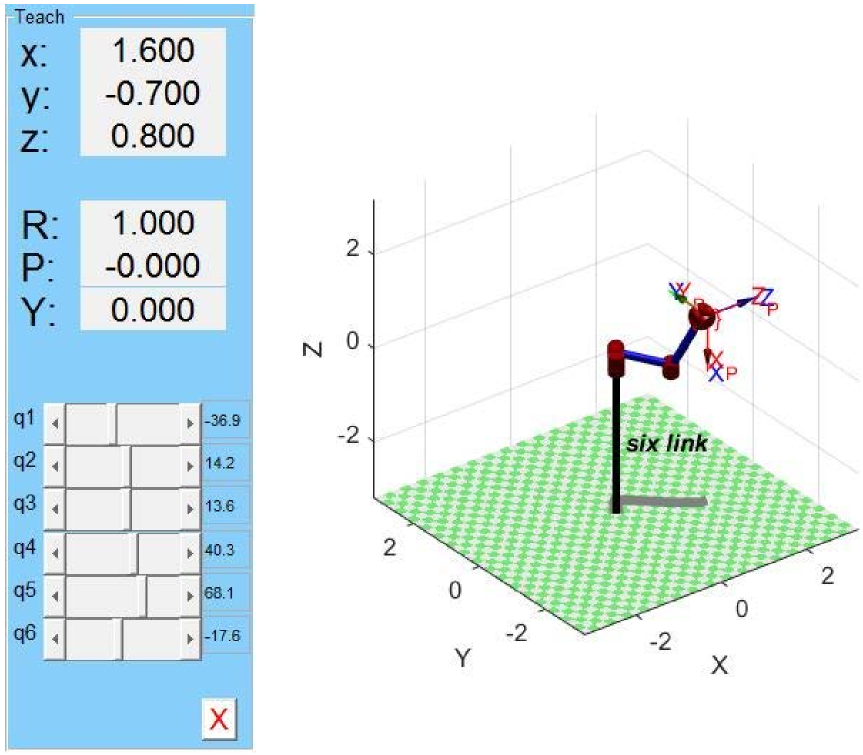
```
%directing the manipulator to the first point[-1,0.3,1.6]

p1=transl([1,-0.3,1.6])

Qp1=six_link.ikunc(p1,[0.01,0.01,0.01,0.01,0.01,0.01])

six_link.fkine(Qp1)

six_link.teach(Qp1)

hold on

trplot(p1,'frame','P','color','b')
```



- For position B it should be taken into consideration that the object has been rotated by 90 degrees about Y axis. So we have :

```
p2=transl([1.6,-0.7,0.8])*troty(pi/2)

Qp2=six_link.ikunc(p2,[0.1,0.1,0.1,0.1,0.1,0.1])

six_link.fkine(Qp2)

six_link.teach(Qp2)

hold on

trplot(p2,'frame','P','color','r')
```

and the plot is shown below:



To verify the correct angles of joints to reach the desired positions:

We can use either `six_link.fkine()`Function or `teach()` .

```
>> six_link.fkine(Qp1)

ans =

    1.0000    0.0000    0.0000    1.0000
   -0.0000    1.0000    0.0000   -0.3000
   -0.0000   -0.0000    1.0000    1.6000
         0         0         0    1.0000


>> six_link.fkine(Qp2)

ans =

    0.0000    0.0000    1.0000    1.6000
   -0.0000    1.0000   -0.0000   -0.7000
   -1.0000   -0.0000    0.0000    0.8000
         0         0         0    1.0000
```