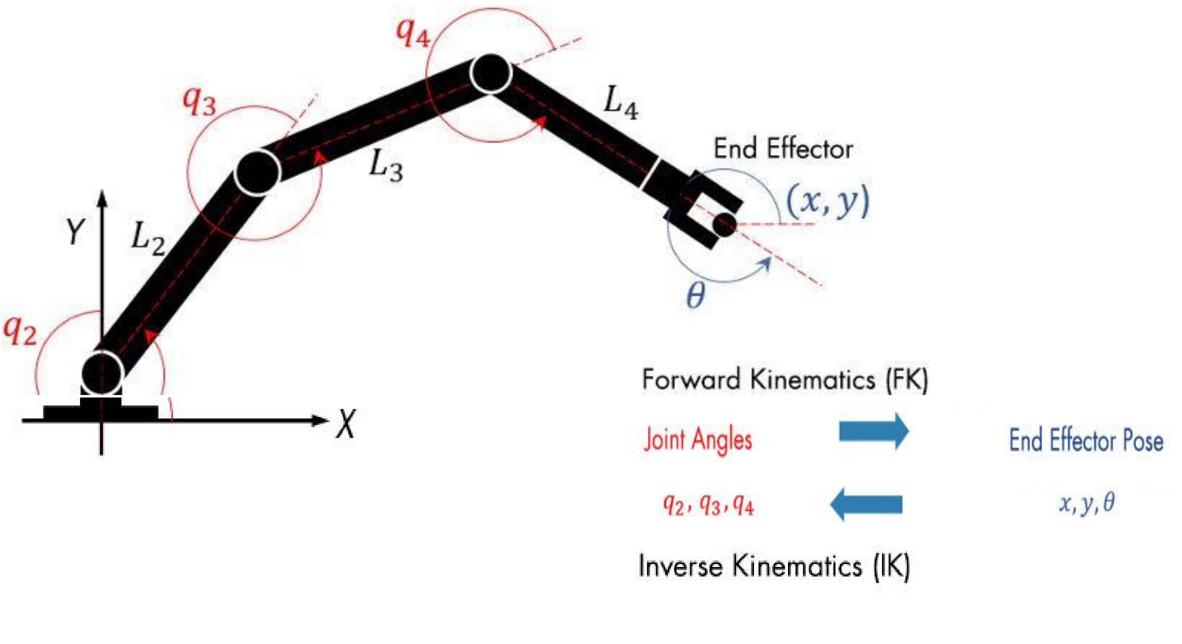
Project 1: Determination of the Position of an End effector for a 3R manipulator using Robo Analyzer.

Abstract:

An **industrial robot** is comprised of a **robot manipulator**, power supply, and controllers. Robotic manipulators can be divided into two sections, each with a different function: **Robot Arm** and Body. Robot manipulators are created from a sequence of **link** and **joint** combinations. The links are the rigid members connecting the joints or axes. There are 2 types of joints: **Linear joint** and **Rotary joint**. The rotary joint consists of the **Revolute joint (R)** and a **Twisting joint (T)**. Each manipulator has its **DOF** (Degree of freedom), the number of degrees of freedom is equal to the **total number of independent movements** that an object can perform. In this project, we are using a **3R Manipulator** whose **DOF** is **3** and consists of **3 Revolute joints**. The Kinematic model of robots is to describe the non-linear relationship between the position and orientation of the end-effector and the displacement of each joint, which is an important content of robot calibration. We use the link and joint parameters to carry out the kinematic analysis (Kinematics) of manipulators. **Length of link** (**a**), **Angle of twist of link** (**α**), **Offset of link** (**d**), **Joint Angle** (**θ**). We obtain the **position and orientation** information of the end-effector of a manipulator of known geometry for a given set of joint angles in **forward kinematics**. On the other hand, in **inverse kinematics**, we **determine the joint angles** of a manipulator of known geometry for a given position and the orientation information of its end-effector. The kinematic equations between the working space and the joint space are deduced by the homogeneous transformation principle. In the present project, kinematic analysis is done using **ROBOANALYZER** software to find out the position of the tool or end effector for different and joint variables.



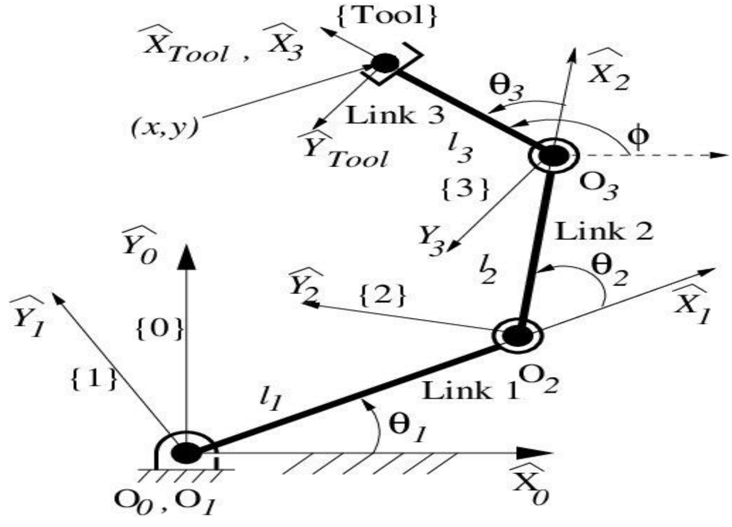
Introduction to 3R PLANAR MANIPULATOR:

The mathematical modeling of spatial linkages is quite involved. It is useful to start with planar robots because the **kinematics of planar mechanisms** is generally much **simpler to analyze**. Also, planar examples illustrate the basic problems encountered in robot design, analysis, and control without having to get too deeply involved in mathematics.

We will start with the example of the planar manipulator with **three revolute joints**. The manipulator is called a planar **3R manipulator**.

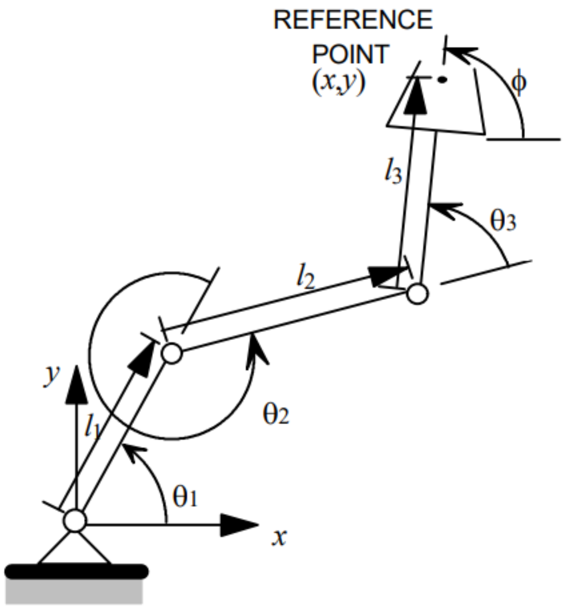
While there may not be any three degrees of freedom (DOF) in industrial robots with this geometry, the planar 3R geometry can be found in many robot manipulators. For example, the **shoulder swivel**, **elbow extension**, and **pitch of the Cincinnati Milacron T3 robot** can be described as a planar 3R chain.

Similarly, in a four DOF SCARA manipulator, if we ignore the prismatic joint for lowering or raising the gripper, the other three joints form a planar 3R chain. Thus, it is instructive to study the planar 3R manipulator as an example.



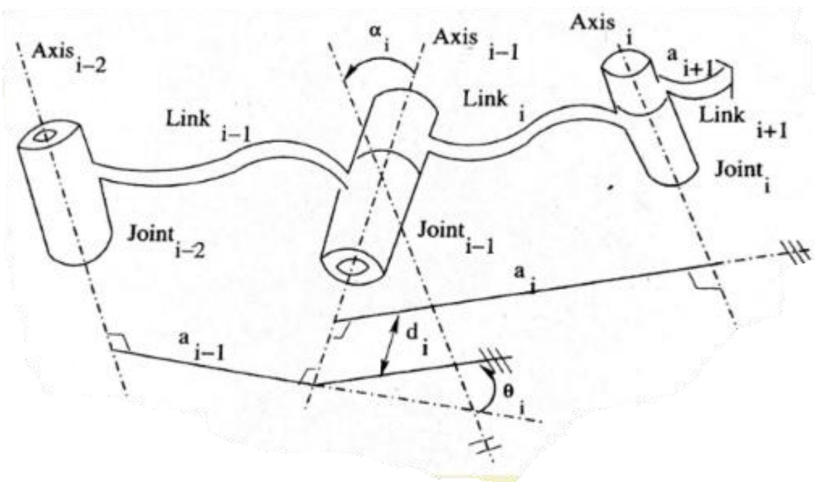
Robot Kinematics:

To specify the geometry of the planar 3R robot, we require **three parameters, l1, l2, and l3**. These are the **three link lengths**. In the figure, the three **joint angles** are labeled **θ1, θ2, and θ3**. These are obviously **variable**.



The joint variables and link lengths for a 3R planar manipulator

The precise definitions for the link lengths and joint angles are as follows.



Length of link i (ai): It is the mutual perpendicular distance between Axisi-1 and Axisi

Angle of twist of link i (αi): It is defined as the angle between Axisi-1 and Axisi

Offset of link i (di): It is the distance measured from a point where ai-1 intersects the Axisi-1 to the point where ai intersects the Axisi-1 measured along the said axis

Joint Angle (θi): It is defined as the angle between the extension of ai-1 and ai measured about the Axis Notes: i-1

Another set of variables that is useful to define is the set of coordinates for the end effector. These coordinates define the position and orientation of the end effector. With a convenient choice of a reference point on the end effector, we can describe the position of the end effector using the coordinates of the reference point (x, y) and the orientation using the angle f. The three end effector coordinates (x, y, f) completely specify the position and orientation of the end effector.

Homogenous Transformation Matrix

To keep representation matrices square, if we represent both orientation and position in the same matrix, we will add the scale factors to the matrix to make it a 4 X 4 matrix. If we represent the orientation alone, we may either drop the scale factors and use 3x 3 matrices or add a 4th column with zeros for a position to keep the matrix square. Matrices of this form are called homogeneous matrices, and we write them as follows:

F is a 4x4 matrix that can describe a translation, rotation, or both in one matrix

$$F = \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Could be rotation around the z-axis, x-axis, y-axis, or a combination of the three.

$$R_z = \begin{bmatrix} cos(\theta) & -sin(\theta) & 0 \\ sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\theta) & -sin(\theta) \\ 0 & sin(\theta) & cos(\theta) \end{bmatrix} \qquad R_y = \begin{bmatrix} cos(\theta) & 0 & -sin(\theta) \\ 0 & 1 & 0 \\ sin(\theta) & 0 & cos(\theta) \end{bmatrix}$$

$$\mathrm{Trans}_{z_{n-1}}(d_n) = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_n \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

$$\mathrm{Rot}_{z_{n-1}}(\theta_n) = \left[\begin{array}{ccc|c} \cos\theta_n & -\sin\theta_n & 0 & 0 \\ \sin\theta_n & \cos\theta_n & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

$$\mathrm{Trans}_{x_n}(r_n) = \left[\begin{array}{ccc|c} 1 & 0 & 0 & r_n \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

$$\mathrm{Rot}_{x_n}(\alpha_n) = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_n & -\sin\alpha_n & 0 \\ 0 & \sin\alpha_n & \cos\alpha_n & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

$^{n-1}T_n = \mathrm{Screw}_Z \ X \ \mathrm{Screw}_X$

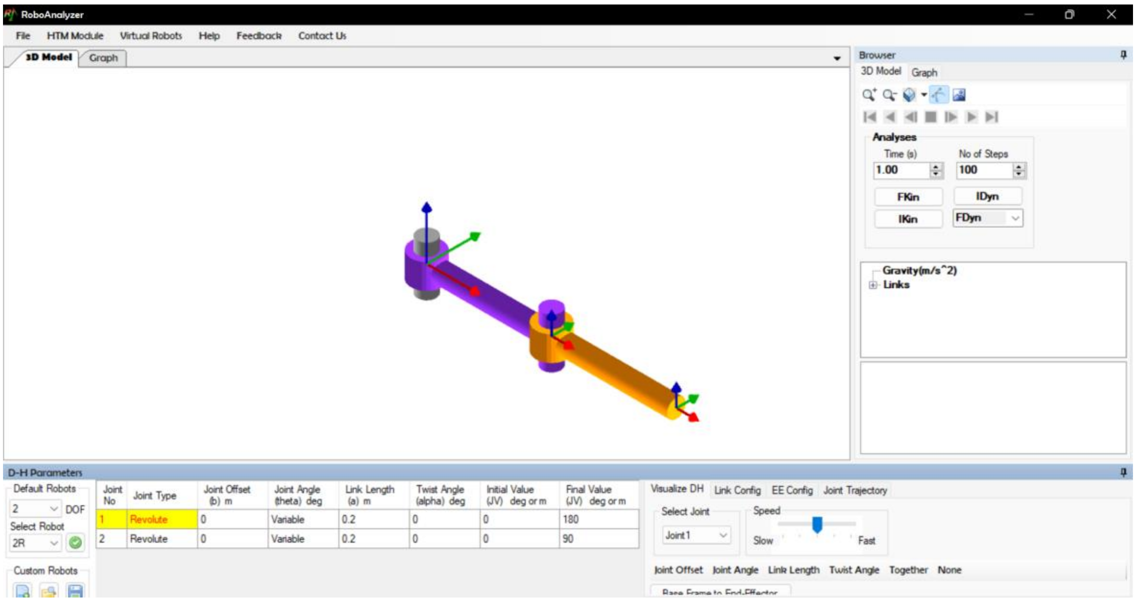
$$^{n-1}T_n = \left[\begin{array}{ccc|c} \cos\theta_n & -\sin\theta_n\cos\alpha_n & \sin\theta_n\sin\alpha_n & r_n\cos\theta_n \\ \sin\theta_n & \cos\theta_n\cos\alpha_n & -\cos\theta_n\sin\alpha_n & r_n\sin\theta_n \\ 0 & \sin\alpha_n & \cos\alpha_n & d_n \\ \hline 0 & 0 & 0 & 1 \end{array}\right] = \left[\begin{array}{ccc|c} & R & & T \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

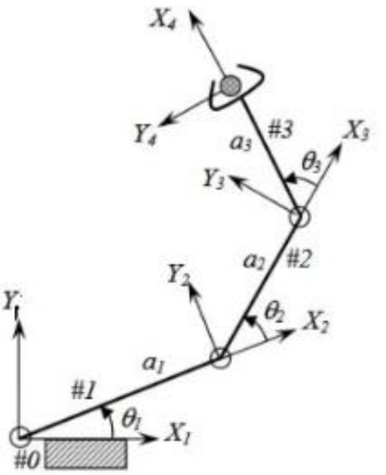
where *R* is the 3×3 submatrix describing rotation and *T* is the 3×1 submatrix describing translation.

Robo Analyzer:

RoboAnalyzer® is a 3D model-based software that can be used to teach and learn Robotics concepts.
It is an evolving product developed in the Mechatronics Lab, Department of Mechanical Engineering at IIT Delhi, New Delhi, India. Its development started under the guidance of Prof. S.K.Saha in order to support the learning/teaching of the topics covered in his book "Introduction to Robotics" published by Tata McGraw Hill, New Delhi.

Interface of Robo Analyzer Software





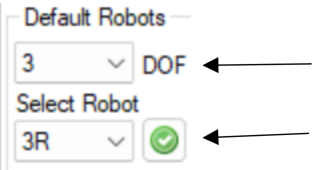
Joint Parameters

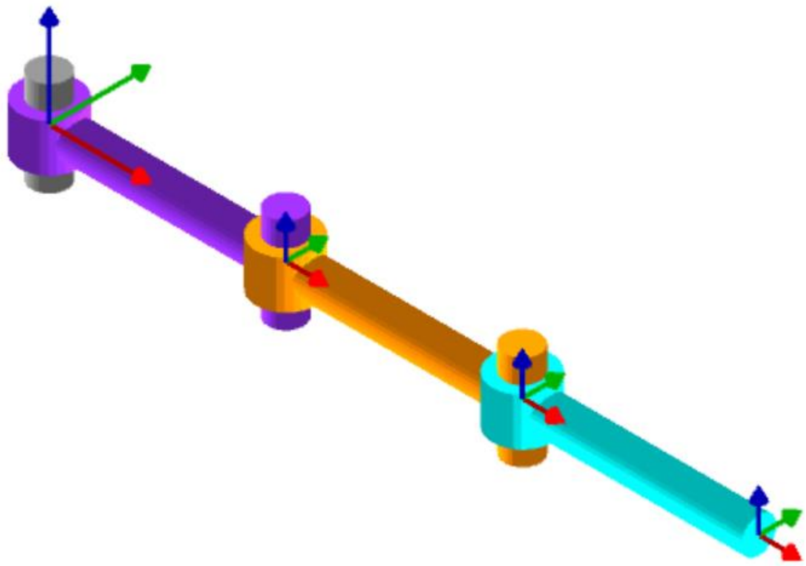
Joint(i)	Joint Type	Joint offset(b) meters	Joint angle(θ) degrees	Link length(a) meters	Link twist(α) degrees
1	Revolute	0	0 to 90	0.2	0
2	Revolute	0	0 to 90	0.2	0
3	Revolute	0	0 to 90	0.2	0

KINEMATIC ANALYSIS OF 3R ROBOT using robo analyzer

3D Model of 3R robot

First, we must import the 3R robot model into the scene so, we must change the Default option of Robot on the left side bottom-most bar. We must change the **DOF** of the robot to **3** and then we must select the 3R robot then tap on the green tick mark to import our model.



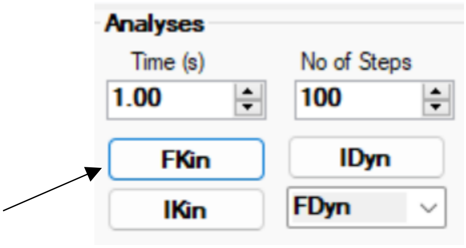


D-H Parameters

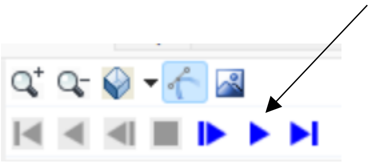
After importing the robot, we must change the joint parameters of our own values. Here I have selected the parameters as follows.

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	Variable	0.2	0	0	90
2	Revolute	0	Variable	0.2	0	0	90
3	Revolute	0	Variable	0.2	0	0	90

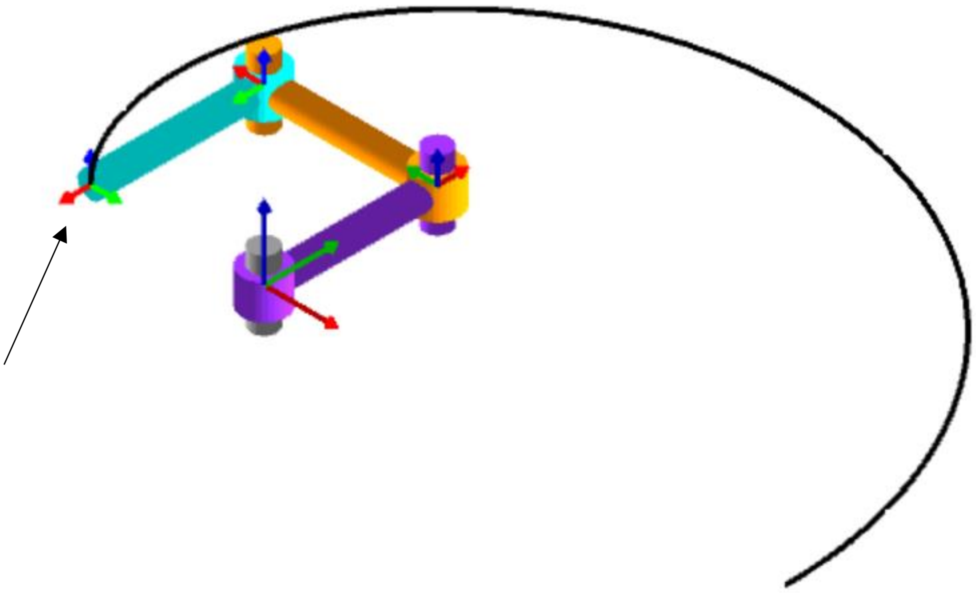
Analysis



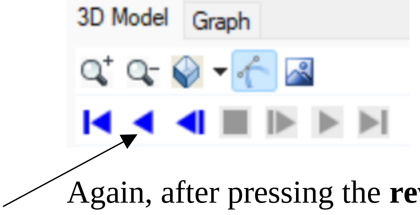
In the right-side bottom-most bar, we can see the analysis bar, we must tap on the **FKin (Forward Kinematics)** option to complete the analysis of Forward Kinematics.



After that, we must press the **play** button to analyze the **Forward Kinematics** of the robot and the robot will start moving and we can see in the below image that the end effector stops at the desired position of the given joint parameters data.

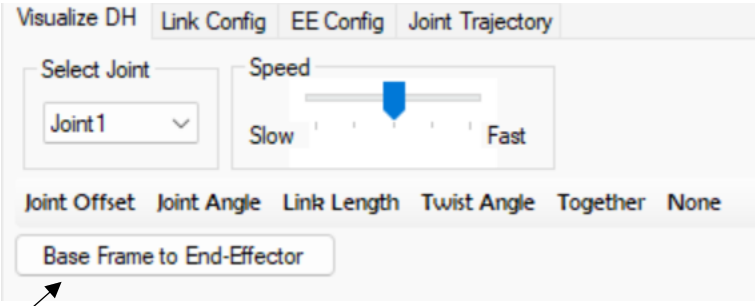


Isometric View of Robot (FKin)



Again, after pressing the **rewind** button, the robot comes to the **initial position**.

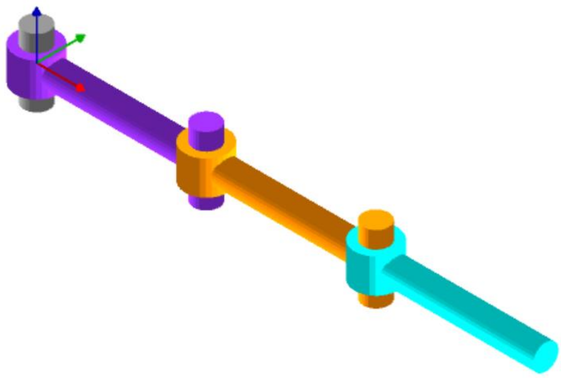
Frame Transformation



- **Joint 1**

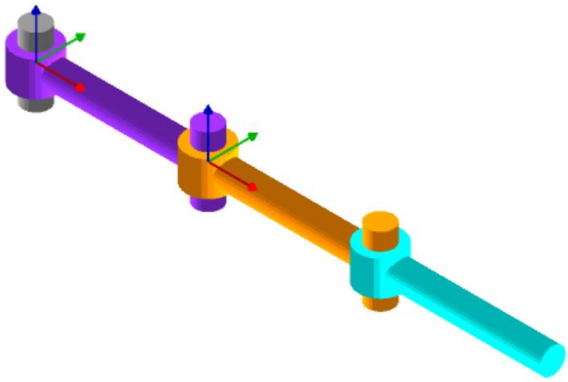
Joint offset = 0, so the frame of the axis doesn’t move in Z-direction

Joint angle = 0 to 90 (variable)



Link length = 0.2m so the axis translates 0.2m along X-direction

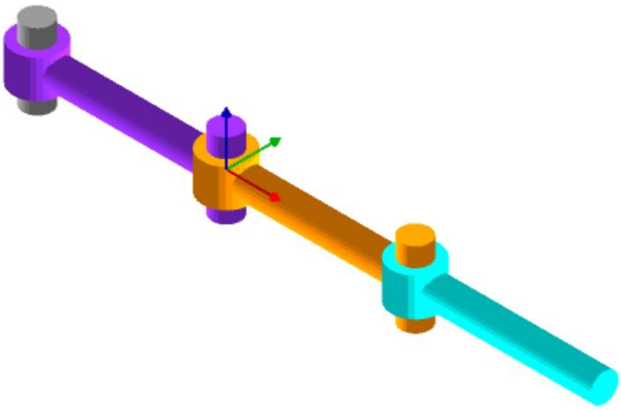
Link twist = 0, so the frame doesn’t make an angle between the (Z)Axisi-1 and (Z)Axisi



- **Joint 2**

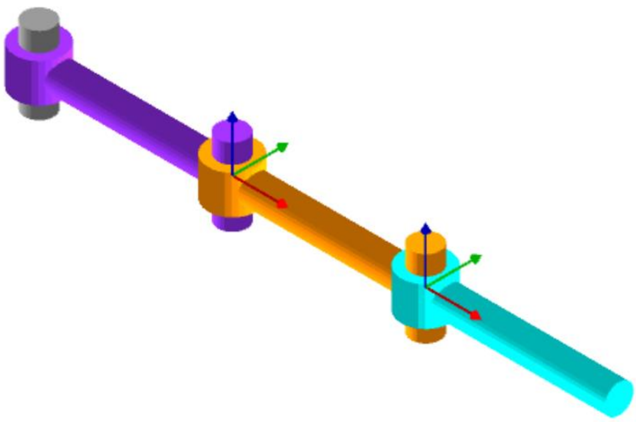
Joint offset = 0, so the frame of the axis doesn't move in Z-direction

Joint angle = 0 to 90 (variable)



Link length = 0.2m so the axis translates 0.2m along X-direction

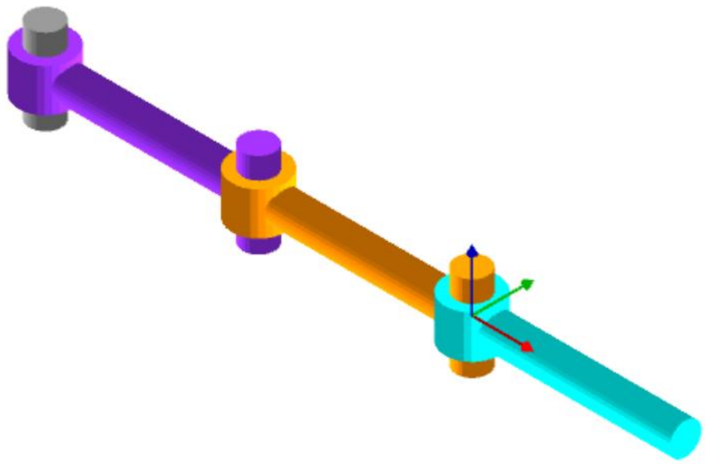
Link twist = 0, so the frame doesn't make an angle between the (Z)Axisi-1 and (Z)Axisi



- **Joint 3**

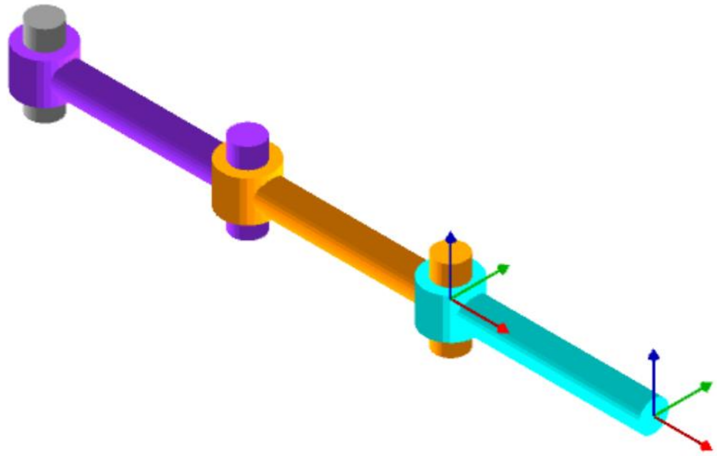
Joint offset = 0, so the frame of the axis doesn't move in Z-direction

Joint angle = 0 to 90 (variable)



Link length = 0.2m so the axis translates 0.2m along X-direction

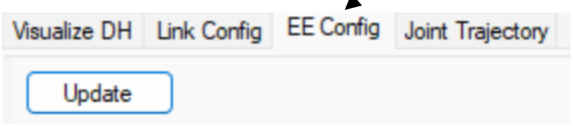
Link twist = 0, so the frame doesn't make an angle between the (Z)Axisi-1 and (Z)Axisi

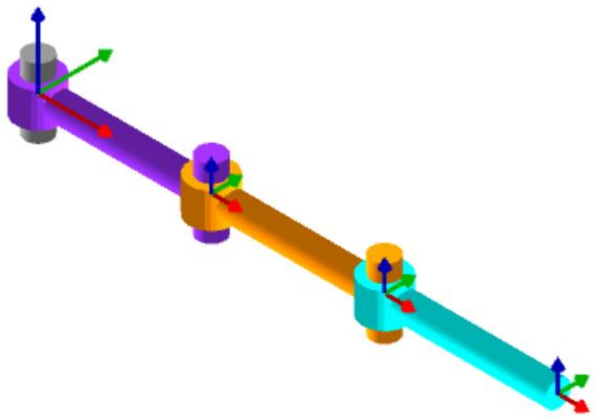


End Effector Configuration

- ### Initial Configuration

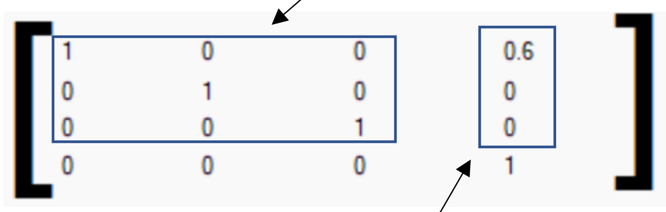
To obtain the end effector configuration we need to tap on **EEConfig** which is in the bottom of the right side and then we need to tap on update.





After tapping on update we will get the initial end effector configuration or transformation matrix

Rotation Matrix (There is no rotation of the end effector)

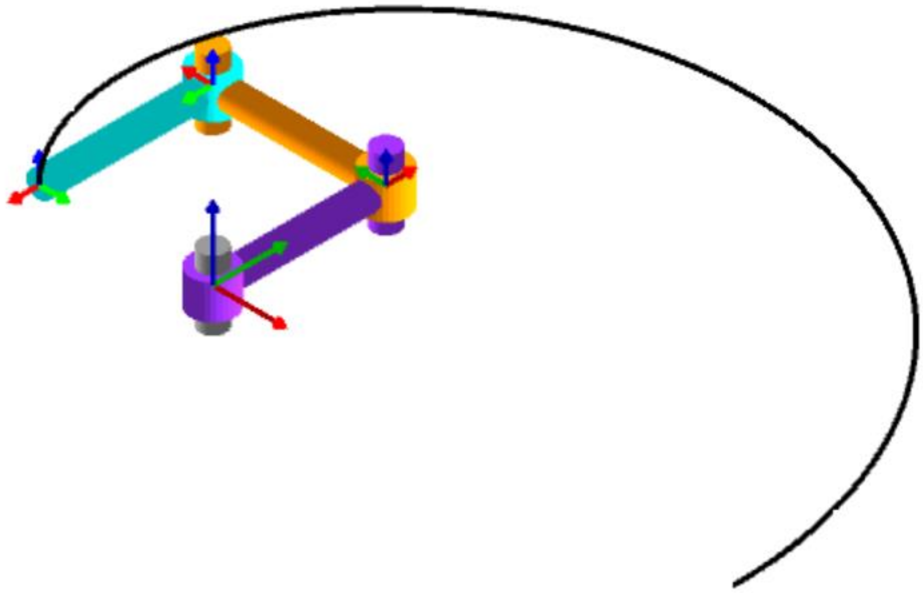


Translation Matrix (End effector is 0.6m distance away from the base frame at initial configuration)

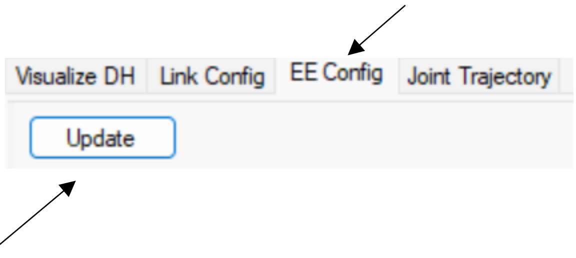
- **Final Configuration**

Now to obtain the final end effector configuration we need to move the robot's end effector to the final position by applying the forward kinematics of the given join parameters data.

After applying the Forward Kinematics, the robot comes into the desired end position.



To obtain the final end effector configuration we need to tap on **EEConfig** which is in the bottom of the right side and then we need to tap on update.



After tapping on the update, we will get the final end effector configuration or transformation matrix

Rotation Matrix (The final frame is 270 degrees oriented about the Z axis w.r.t base frame)

$$\begin{bmatrix} 0 & 1 & 0 & -0.2 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Translation Matrix (End effector is -0.2m distance away from the base frame at initial configuration)

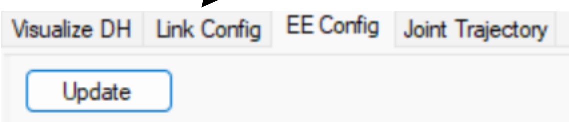
Link Configuration

We know that

- ⇨ transformation matrix of link 3 w.r.t base frame = transformation matrix of link 1 w.r.t base frame X transformation matrix of link 2 w.r.t 1 X transformation matrix of link 3 w.r.t 2

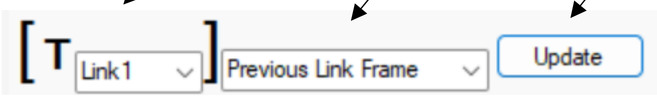
- ⇨ ${}^{0}T_3 = {}^{0}T_1 x {}^{1}T_2 x {}^{2}T_3$

To operate the link configuration, we need to tap on **Link Config**

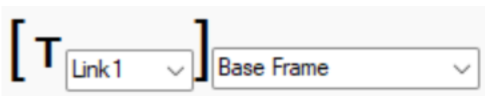


After that, we must select the main frame and reference frame accordingly, and then we need to update them to get the Transformation matrix.

Main frame Reference frame Gives the Transformation matrix



- **Link 1 w.r.t Base frame**



$$\begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0.2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- **Link 2 w.r.t 1**

$[T_{\text{Link2}}]$ Previous Link Frame

$$\begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0.2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- **Link 3 w.r.t 2**

$[T_{\text{Link3}}]$ Previous Link Frame

$$\begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0.2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- **Link 3 w.r.t base frame**

$[T_{\text{Link3}}]$ Base Frame Update

$$\begin{bmatrix} 0 & 1 & 0 & -0.2 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Hence,

$$\begin{bmatrix} 0 & 1 & 0 & -0.2 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} ({}^{0}T_{3})$$

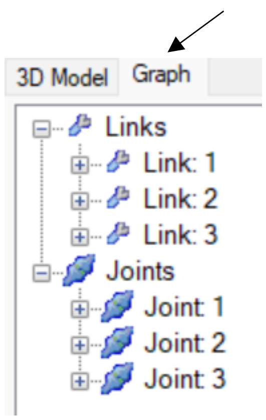
$$\Updownarrow$$

$$({}^{0}T_{1}) \qquad ({}^{1}T_{2}) \qquad ({}^{2}T_{3})$$

$$\begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0.2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0.2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0.2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Graphs

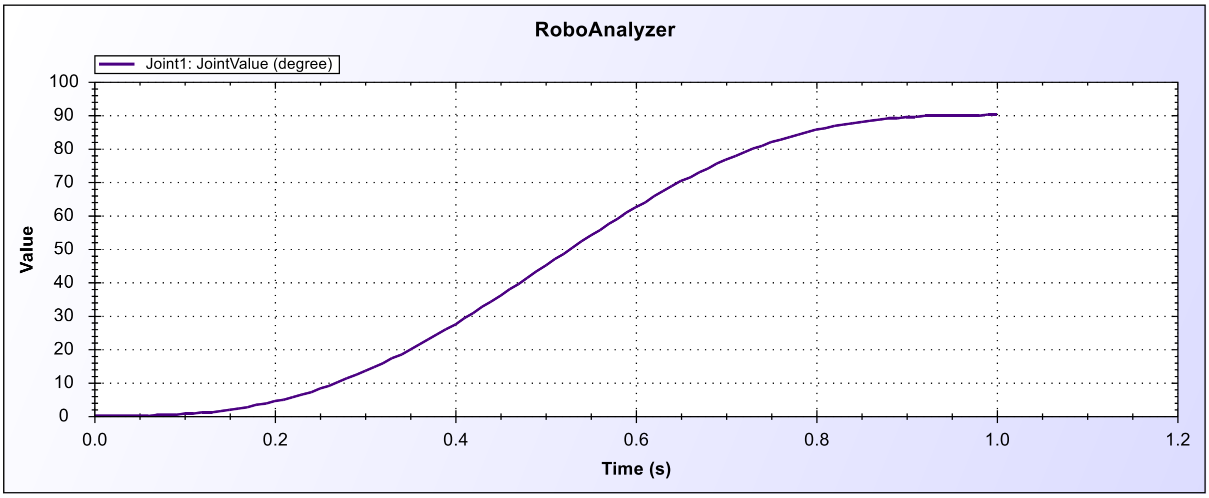
To get the different graphs of joints (joint angle, joint velocity, etc..) and links in robo analyzer 1st we must tap on the **FKin** option, and then we need to tap on the **Graph** option which is on the top right side. Then we need to select the different options which we see in the below image to access different graphs



Joint Angle Graphs

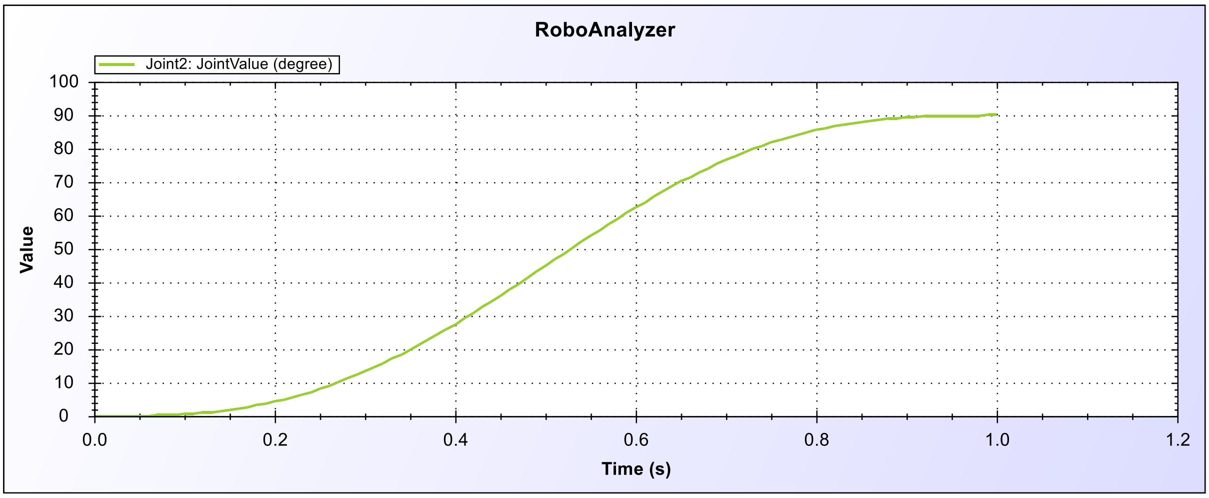
- ### Joint 1

The joint angle of joint 1 is variable as time increases the movement of the joint starts from 0 degrees and ends at 90 degrees within 1sec and the graph will be a nonlinear curve.



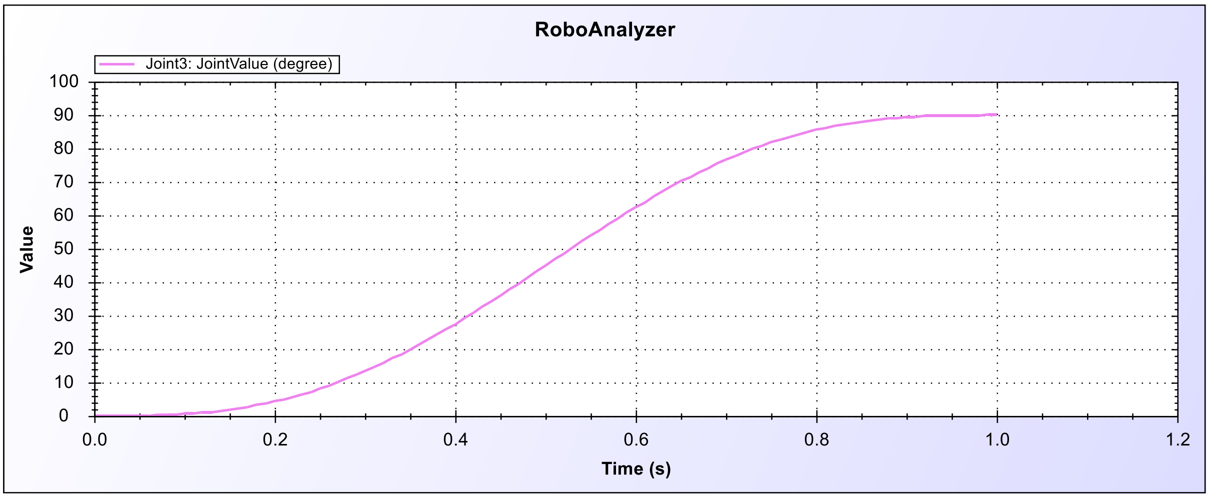
- ### Joint 2

The joint angle of joint 2 is variable as time increases the movement of the joint starts from 0 degrees and ends at 90 degrees within 1sec and the graph will be a nonlinear curve.



- ## Joint 3

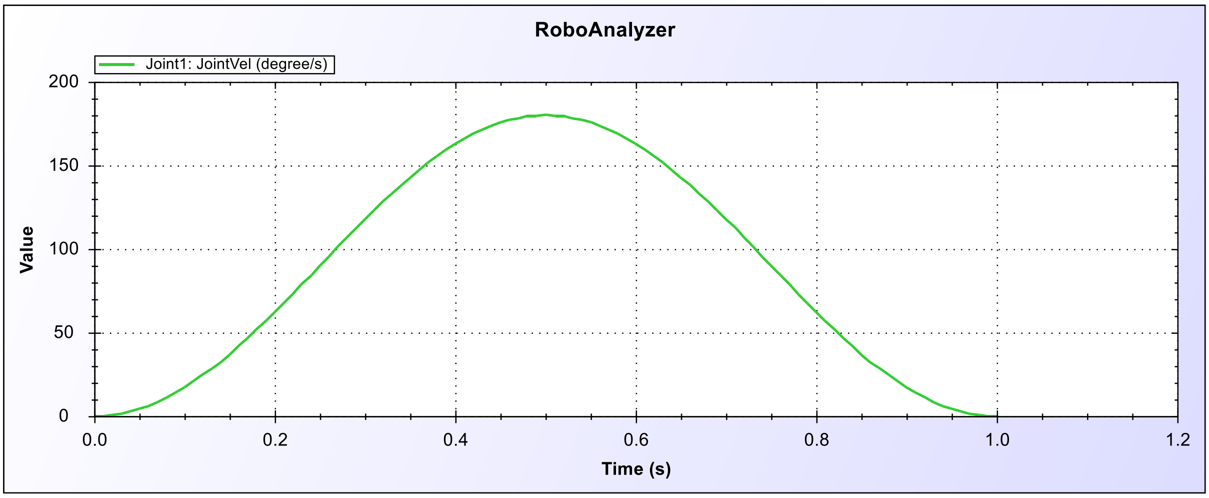
The joint angle of joint 3 is variable as time increases the movement of the joint starts from 0 degrees and ends at 90 degrees within 1sec and the graph will be a nonlinear curve.



Joint Velocity Graphs

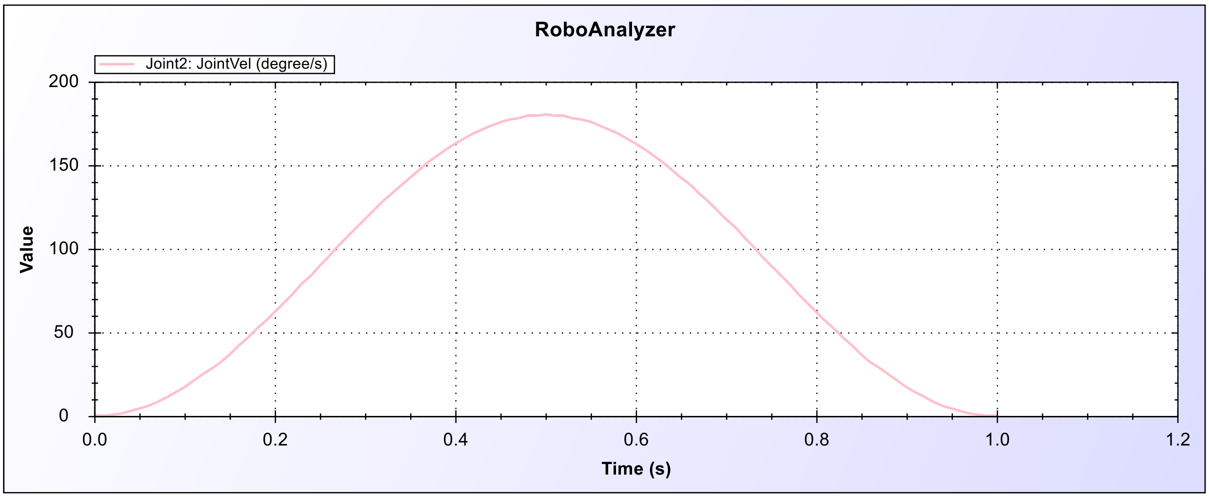
- ## Joint 1

The joint velocity of joint 1 is increasing as time increases up to half of the total period of settling time and then the velocity of the joint starts decreasing gradually within 1sec, and the graph will be an exponential curve.



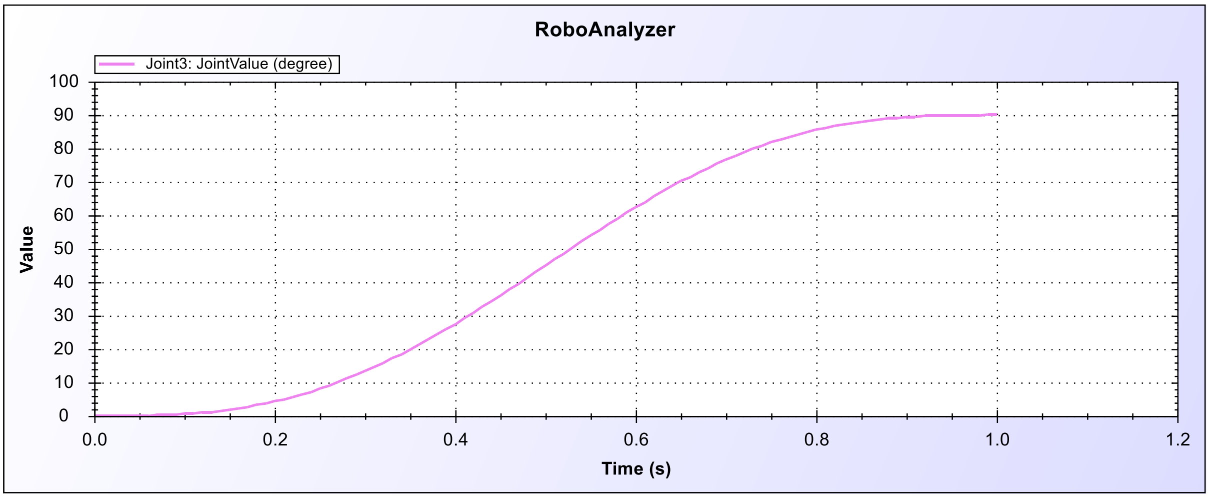
- **Joint 2**

The joint velocity of joint 2 is increasing as time increases up to half of the total period of settling time and then the velocity of the joint starts decreasing gradually within 1sec, and the graph will be an exponential curve.



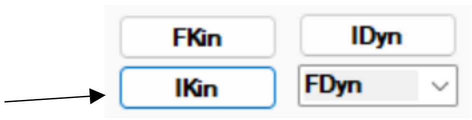
- **Joint 3**

The joint velocity of joint 3 is increasing as time increases up to half of the total period of settling time and then the velocity of the joint starts decreasing gradually within 1sec, and the graph will be an exponential curve.

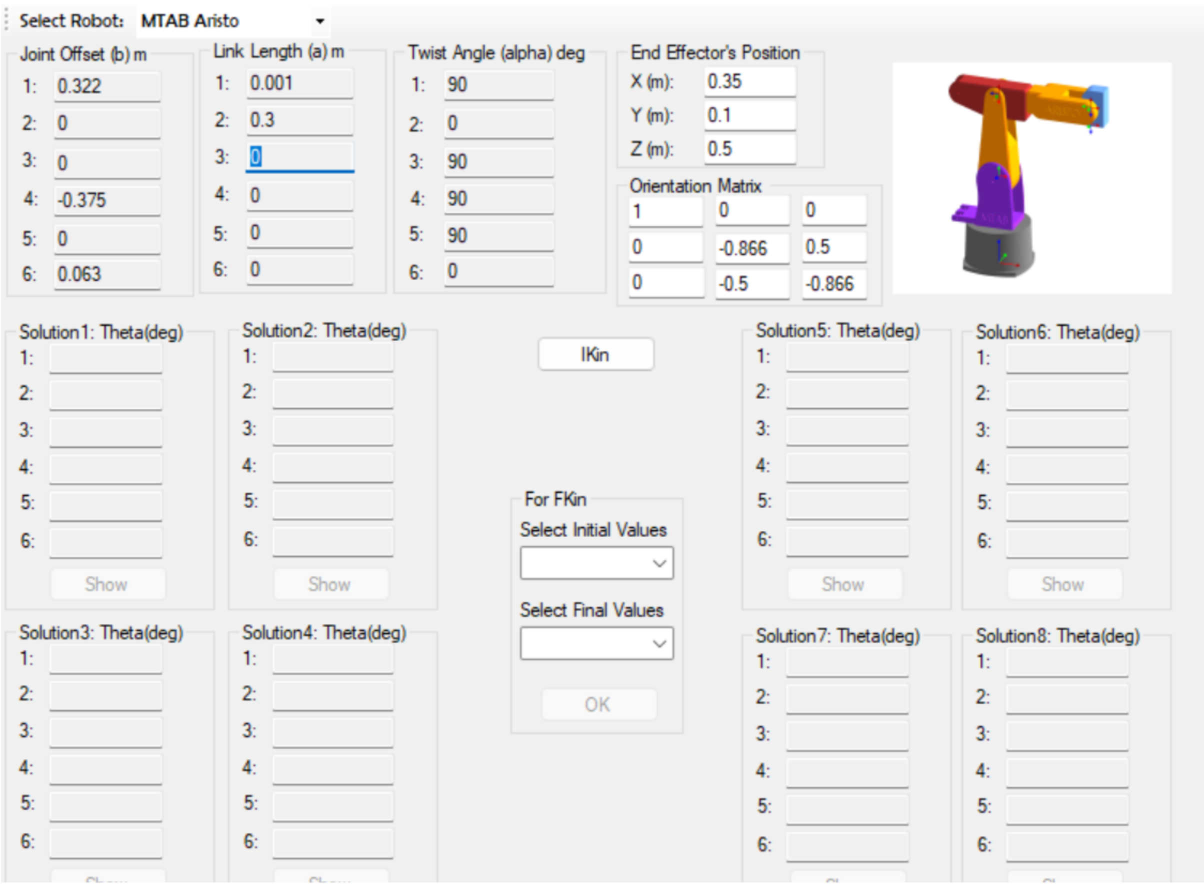


Inverse Kinematics:

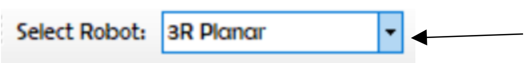
To tackle Inverse Kinematics in robo analyzer we need to tap on **IKin** option which is on the top right side.



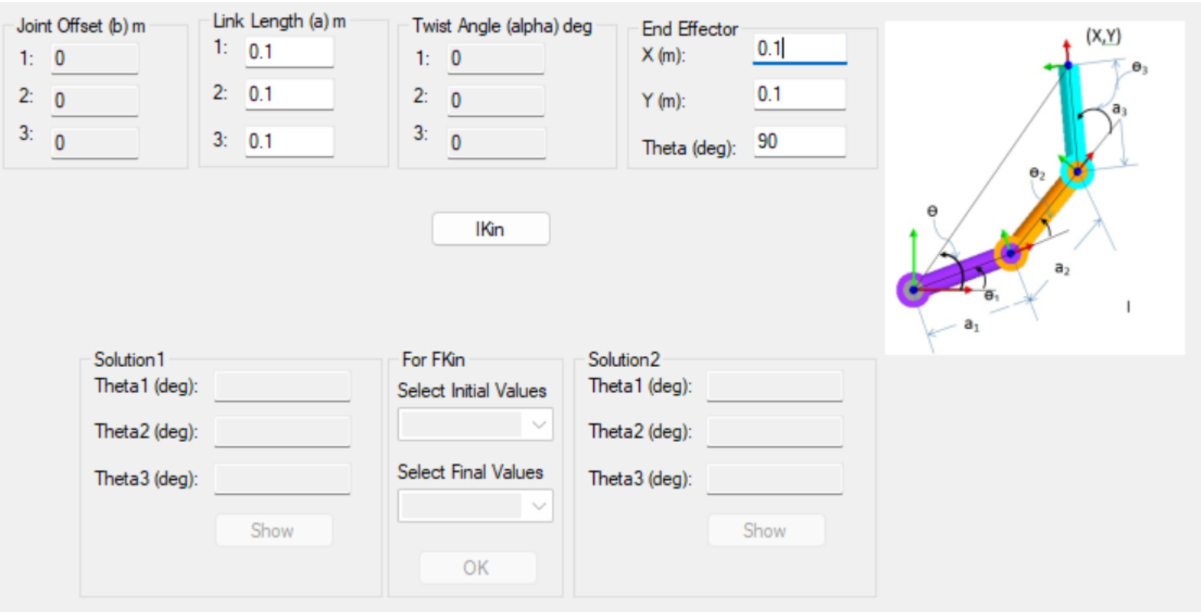
Then you will see the Inverse Kinematics page opened in a new tab



Then on top, there is an option to select the robot

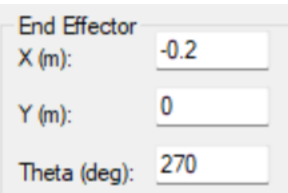


Then the tab will change to our selected robot parameters

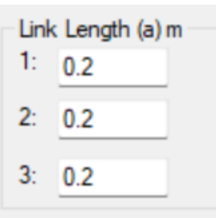


Then we need to type the end effector position and orientation values and link length values to get their respective joint angle values

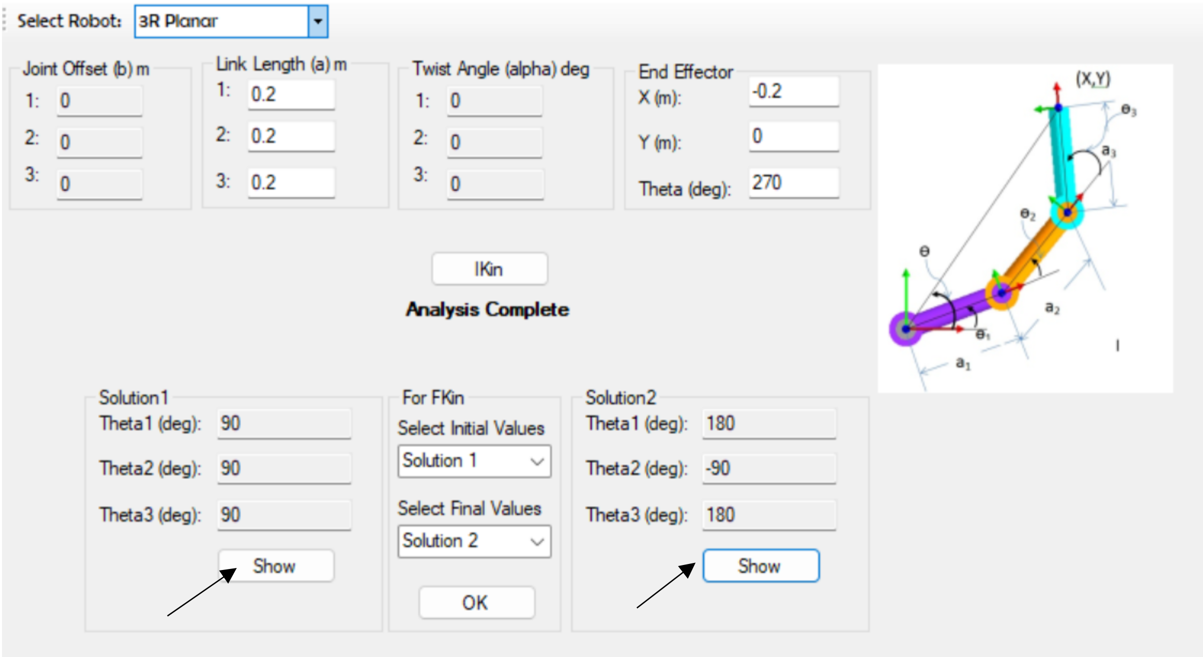
effector position and orientation



link lengths



After that, we must tap on **IKin** option to complete the analysis and after completion of the analysis, we will get one or more than one solution.



Here we got two solutions

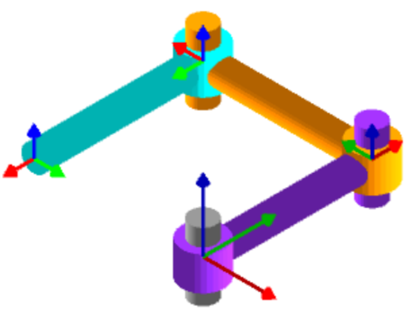
By tapping on the show option, we can see the model of the robot and its end effector position and orientation which will match our given values.

Solution1
Theta1 (deg): 90

Theta2 (deg): 90

Theta3 (deg): 90

End effector position of Solution 1

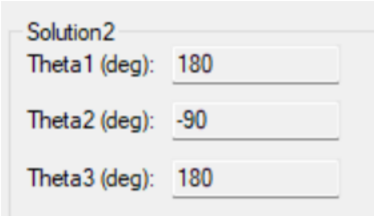


Final End-Effector Configuration of solution 1

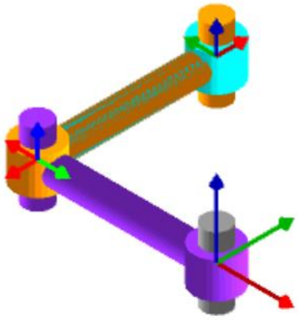
$$\begin{bmatrix} 0 & 1 & 0 & -0.2 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

End effector Oriented by 270 degrees

End effector Coordinates



End effector position of Solution 2



Final End-Effector Configuration of solution 2

$$\begin{bmatrix} 0 & 1 & 0 & -0.2 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

End effector Oriented by 270 degrees

End effector Coordinates

<u>Conclusion</u>:

Kinematic analysis of the 3DOF 3R robot was simulated using Robo Analyzer software.

At the end of this project, we came to a clear understanding of Robo Analyzer Software.

By using Robo Analyzer we obtained forward kinematics of the end effector of the robot for the given joint parameters.

By using graphs in robo analyzer software we analyzed the change in joint angles with respect to time and also the change in joint velocities with respect to time.

By inputting the link lengths and desired coordinates and angle of the end effector we obtained solutions for inverse kinematics and calculated DH parameters theoretically and that obtained matrix of the end effector was matched with the updated matrix of DH parameters in Robo Analyzer.

Thus, the kinematic analysis of the 3DOF 3R robot is completed.

-------------------------xxxxxxxxx----------------------