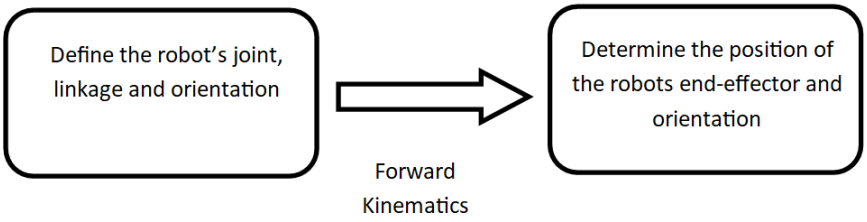
Forward Kinematics of RRPR Robot

Written by: Andres Cuenca

Date: 06/18/2020

Intro/Background

Forward kinematics is applied to determine the orientation and position of the end-effector of any robot. Below is a straight forward concept of forward kinematics.



In this case I worked closely with the RRPR robot. Below is the robot that I designed on SolidWorks CAD.

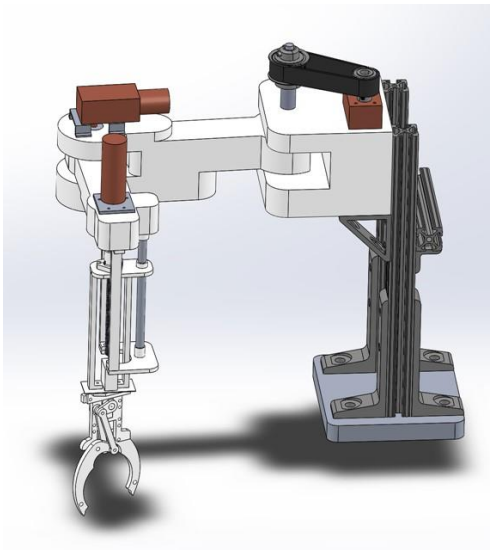


Figure 1: SCARA Robot designed by Andres Cuenca.

Figure 2: Forward Kinematics Diagram.

Joints that revolve, will rotate relative to the z-axis. Joints that are prismatic, will move in a linear motion, with respect to the z-axis. The general rule is to begin with the linkage at the base of the robot and move to the next joint until the end-effector of the robot is reached. Below is a derived Denavite-Hartenburg table, which each row will represent a 4x4 matrix for each joint of the robot.

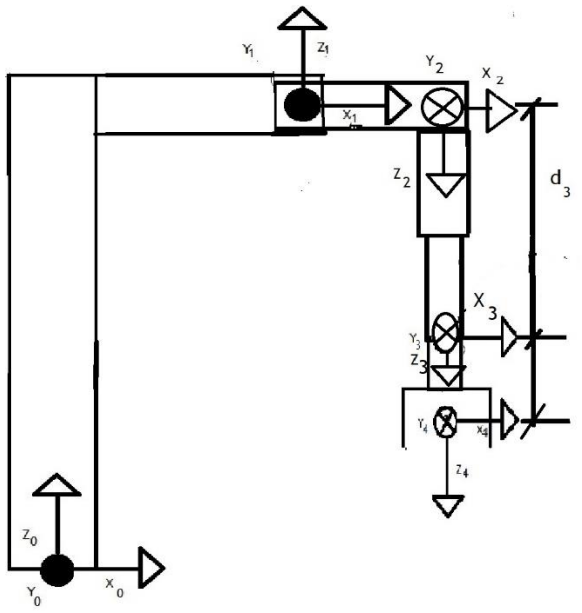


Figure 3: Setting up the coordinates of each joint of the RRPR robot.

	i-1	ai-1	di	i
1	0	0.150 m	0.036 m	1*
2	180	0.125 m	0.03 m	2*
3	0	0 m	d3*	0
4	0	0 m	0.02 m	4*

Table 1: Denavite-Hartenburg table derive from the figure 3.

The following parameters are defined as the following:

- α_{i-1}: angle measure from zi-1 to zi measured about xi-1 (Link twist) - constant
- a_{i-1}: distance from zi-1 to zi measured along xi-1 (Link length) - constant
- d_i: distance from xi-1 to xi measured along zi (Link offset) - prismatic joint, this is variable

- θ_i: angle from xi-1 to xi measured about zi (Joint angle) - revolute joint, this is variable

First, second and fourth row represent the revolute joints of the RRPR robot. The third row represents the prismatic linkage robot.

Finally, a matlab script was written to compute the forward kinematics, which requires properly setting up the matrix of each joint and multiplying all 4x4 matrix to compute for the final transformation matrix.