Ain Shams University

Faculty of Engineering

Mechatronics Engineering Department

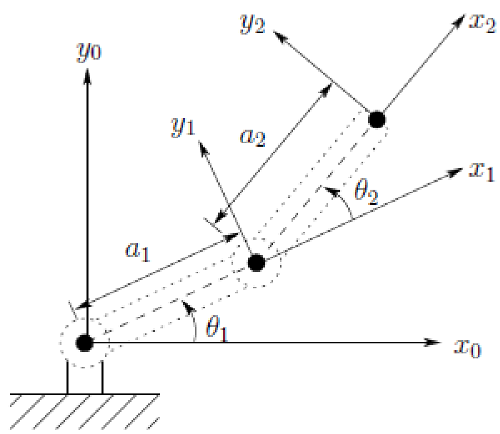
Industrial Robotics (MCT344s)

2-DOF robotic arm project

Name	ID
Alaa Elsayed Metwally Mohamed	1809361
Ahmed Yasser Ahmed Abdelhamid	1804679
Ali Abdelhakem Mahmod Hussin	1805347
Mohamed Emad Mohamed Hussien	1802245
Mohamed Emad Mohamed Moslem	1806770
Mohamed Abdulaziz Farrag Ahmed	1805858

List of contents:

1. Project description



- A 2-Degree-of-Freedom (DoF) Robot Arm simulation is developed to give us more freedom to determine the robot movement so that the trial and error cycle can be avoided . Furthermore, the results of the simulation can be applied to SCARA robot for practical implementation.

- For a servo plotter to work, each motor must resist the torque from the opposite motor.

- An ideal servo should hold its position if it experiences an external torque ... in practice there is a slight movement or “dead-band” until the servo resists.

- Inverse kinematics requires complex mathematical analysis, especially in the higher degree of freedom (DoF). Taking an example of 2-DoF forward kinematics analysis depicted with a kinematic trigonometric diagram on figure 1. The equation to calculate the end effector point are given in equation 1 through equation 4

$$x_1 = L_1 \cos\theta_1 \tag{1}$$

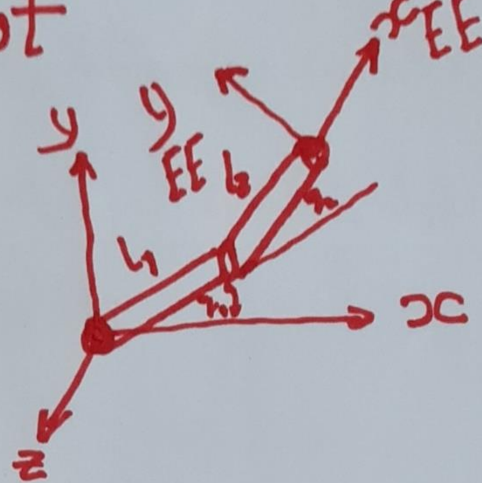
$$y_1 = L_1 \sin\theta_1 \tag{2}$$

$$x_2 = x_1 + L_2 \cos(\theta_1 + \theta_2) \tag{3}$$

$$y_2 = y_1 + L_2 \sin(\theta_1 + \theta_2) \tag{4}$$

2. Control analysis for equations

RR Robot



Fwd Pos.

$$x_{EE} = L_1 \cos q_1 + L_2 \cos(q_2+q_1)$$

$$y_{EE} = L_1 \sin q_1 + L_2 \sin(q_2+q_1)$$

Fwd velocity $\quad q_1 = \theta_1 \quad q_2 = \theta_2$

$$\dot{x}_{EE} = \frac{\partial x}{\partial q_1} * \frac{\partial q_1}{\partial t} + \frac{\partial x}{\partial q_2} * \frac{\partial q_2}{\partial t}$$

$$= [-L_1 \sin q_1 - L_2 \sin(q_1+q_2)]\, \dot{\theta}_1$$

$$[-L_2 \cos(q_1+q_2)] * \dot{\theta}_2$$

$$\dot{y}_{EE} = \frac{\partial y}{\partial q_1} * \frac{\partial q_1}{\partial t} + \frac{\partial y}{\partial q_2} * \frac{\partial q_2}{\partial t}$$

$$= [L_1 \cos q_1 + L_2 \cos(q_1+q_2)]\, \dot{q}_1 + \{L_2 \cos(q_1+q_2)\, \dot{q}_2$$

$$\begin{bmatrix} \dot{x}_{EE} \\ \dot{y}_{EE} \end{bmatrix} = \begin{bmatrix} -l_1 Sq_1 - l_2 S(q_1+q_2) & -l_2 C(q_1+q_2) \\ l_1 Cq_1 + l_2 C(q_1+q_2) & l_2 C(q_1+q_2) \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}$$

Inv Velocity

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = J^{-1} \begin{bmatrix} \dot{x}_{EE} \\ \dot{y}_{EE} \end{bmatrix}$$

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \frac{1}{|J|} * \begin{bmatrix} l_2 C(\theta_1+\theta_2) & l_2 C(\theta_1+\theta_2) \\ -l_1 C\theta_1 - l_2 C(\theta_1+\theta_2) & l_1 S\theta_1 + l_2 S(\theta_1+\theta_2) \end{bmatrix} \begin{bmatrix} \dot{x}_{EE} \\ \dot{y}_{EE} \end{bmatrix}$$

Fwd Acc

$$\dot{x}_{EE} = J\ddot{q} + \dot{J}\dot{q}$$

$$\begin{bmatrix} \dot{x}_{EE} \\ \dot{y}_{EE} \end{bmatrix} = J \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} -l_2 S(\theta_1+\theta_2)\dot{\theta}_1 - l_2 S(\theta_1+\theta_2)\dot{\theta}_2 \; \cdots \; l_2 S(\theta_1+\theta_2)[\dot{\theta}_1 + \dot{\theta}_2] \\ [-l_1 S\theta_1 + \tfrac{1}{2} S(\theta_1+\theta_2)]\dot{\theta}_1 + l_2 S(\theta_1+\theta_2)\dot{\theta}_2 \; \cdots \; l_1 C\theta_1 + l_2 C(\theta_1+\theta_2)\dot{\theta}_1 + l_2 C(\theta_1+\theta_2)\dot{\theta}_2 \end{bmatrix} * \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}$$

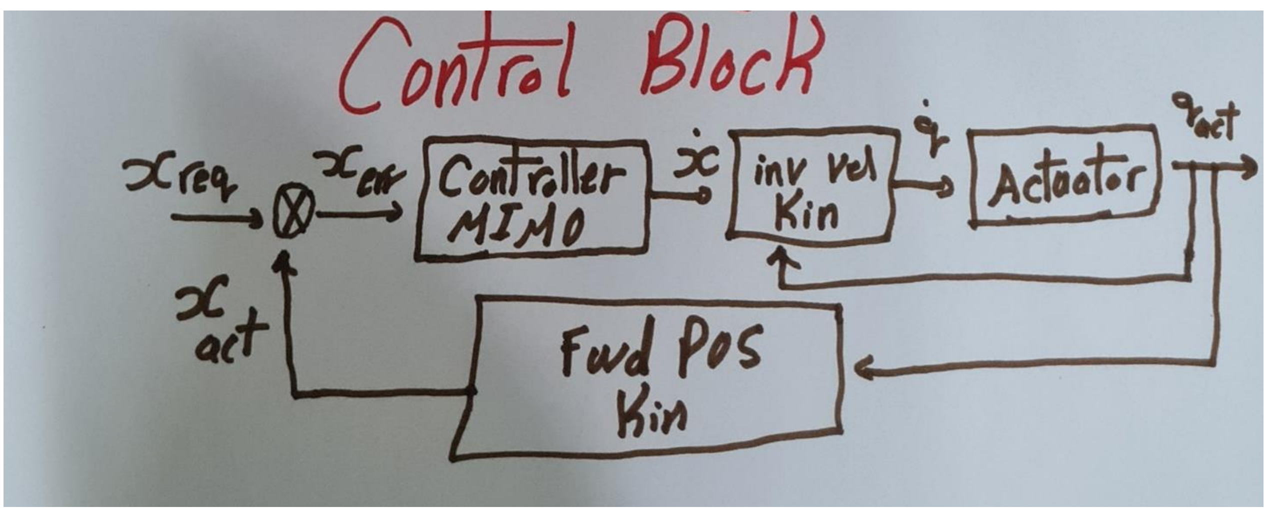
Inv Acc

$$\ddot{x}_{EE} = J\ddot{q} + \dot{J}\dot{q}$$

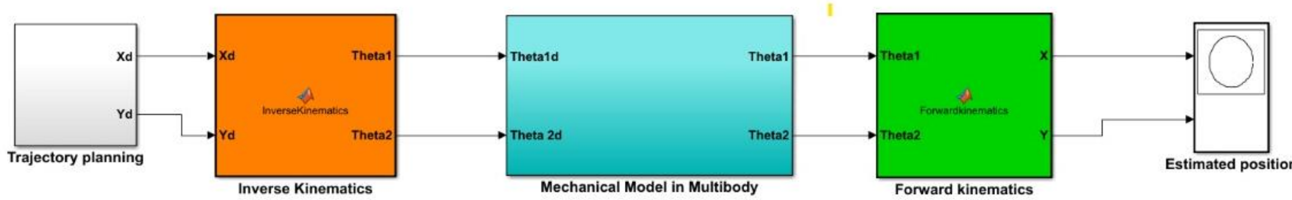
$$\Longrightarrow \ddot{q} = J^{-1}\left[\ddot{x}_{EE} - \dot{J}\dot{q}\right]$$

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} = \frac{1}{|J|} \begin{bmatrix} L_2 C(\theta_1+\theta_2) & L_2 C(\theta_1+\theta_2) \\ -L_1 C\theta_1 - L_2 C(\theta_1+\theta_2) & L_1 S\theta_1 + L_2 S(\theta_1+\theta_2) \end{bmatrix} * \left(\begin{bmatrix} \ddot{x}_{EE} \\ \ddot{y}_{EE} \end{bmatrix} - \dot{J} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \right)$$

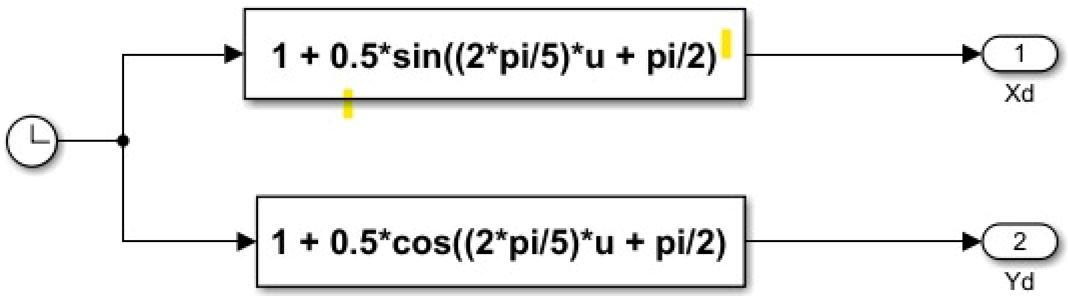
3. The Control Block Diagram :



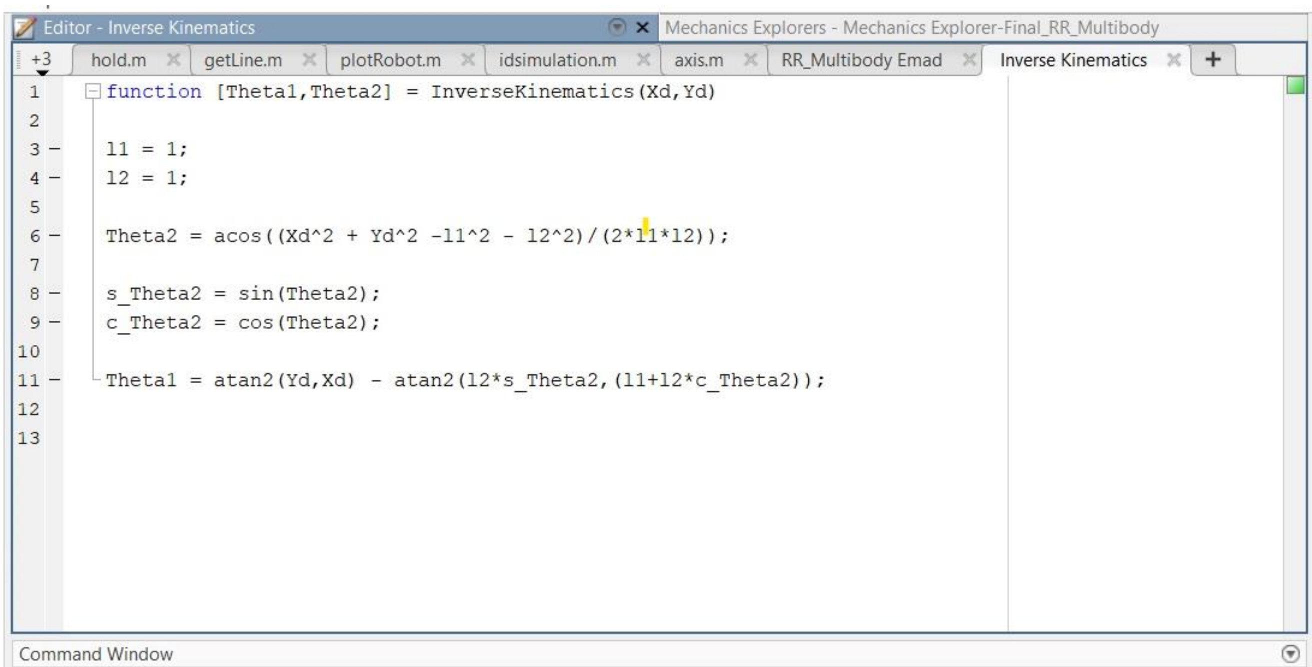
4. MATLAB Simulation



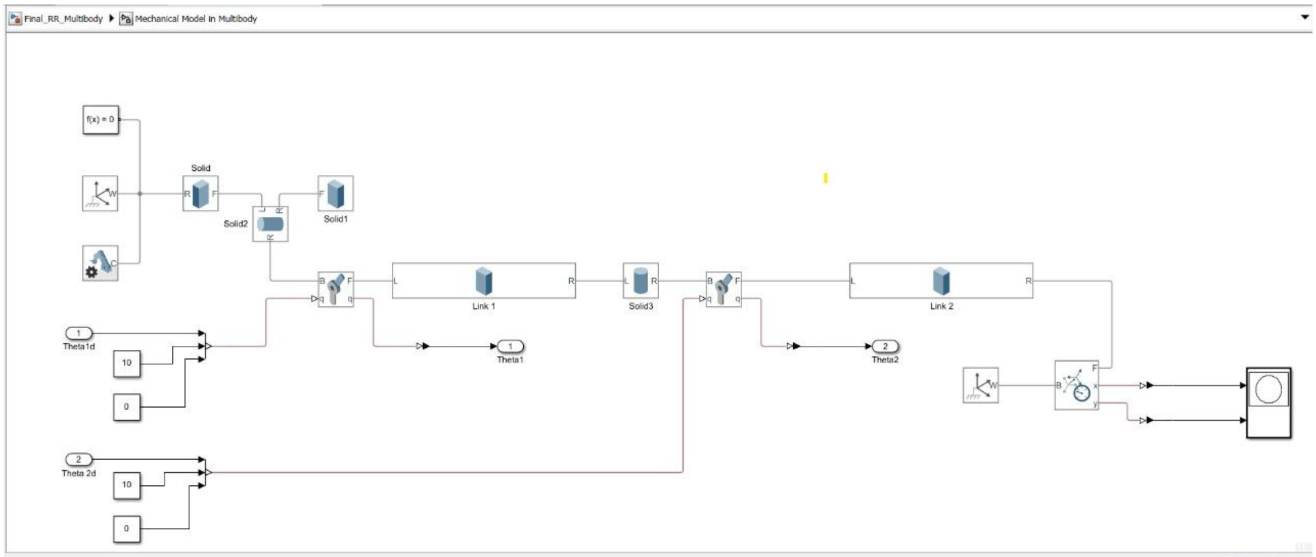
4.1 Trajectory planning



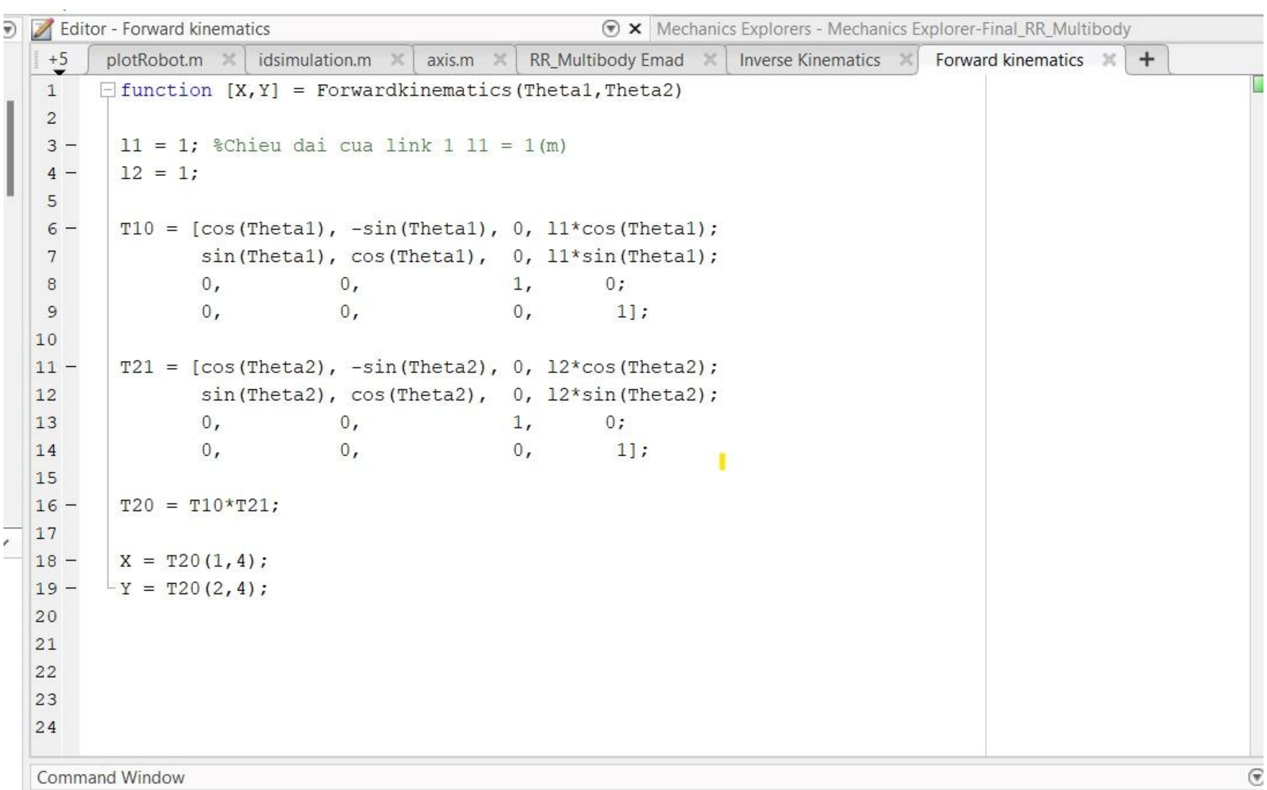
4.2 Inverse Kinematics



4.3 Mechanical model in multibody

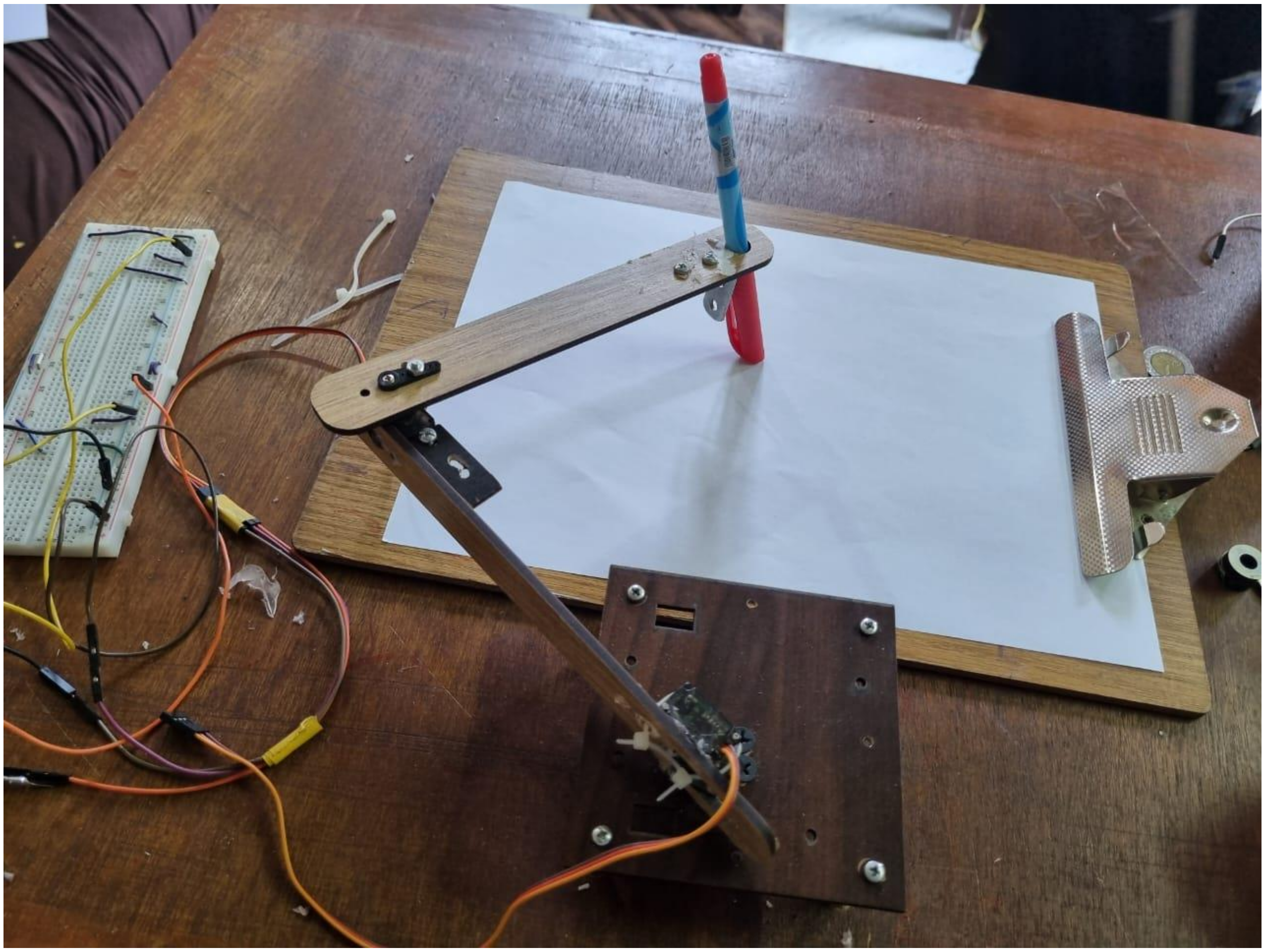


4.4 Forward Kinematics

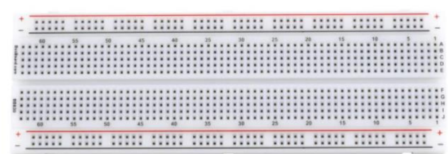
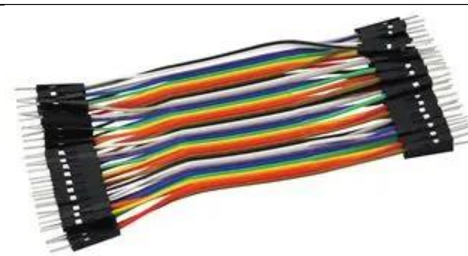



```
function [X,Y] = Forwardkinematics(Theta1,Theta2)

l1 = 1; %Chieu dai cua link 1 l1 = 1(m)
l2 = 1;

T10 = [cos(Theta1), -sin(Theta1), 0, l1*cos(Theta1);
       sin(Theta1), cos(Theta1),  0, l1*sin(Theta1);
       0,           0,            1,      0;
       0,           0,            0,       1];

T21 = [cos(Theta2), -sin(Theta2), 0, l2*cos(Theta2);
       sin(Theta2), cos(Theta2),  0, l2*sin(Theta2);
       0,           0,            1,      0;
       0,           0,            0,       1];

T20 = T10*T21;

X = T20(1,4);
Y = T20(2,4);
```

5. Overall real hardware system picture



6. Components

Name & Its Function	Picture
1 .Arduino uno : is able to read inputs and turn it into an output	
2. Servo motors : it transmits the rotation movement by angle	
3. Breadboard : allows for easy and quick creation of temporary electronic circuits or to carry out experiments with circuit design.	
4. Links : Two 100-mm links used to transmit the rotation movement of motors into writing commands by connecting between them	
5. Wires : used to create the circuit of the system and connect the components with each other	

7. Reference

https://www.instructables.com/Servo-Plotter/