Dhirubhai Ambani Institute of Information and Communication Technology

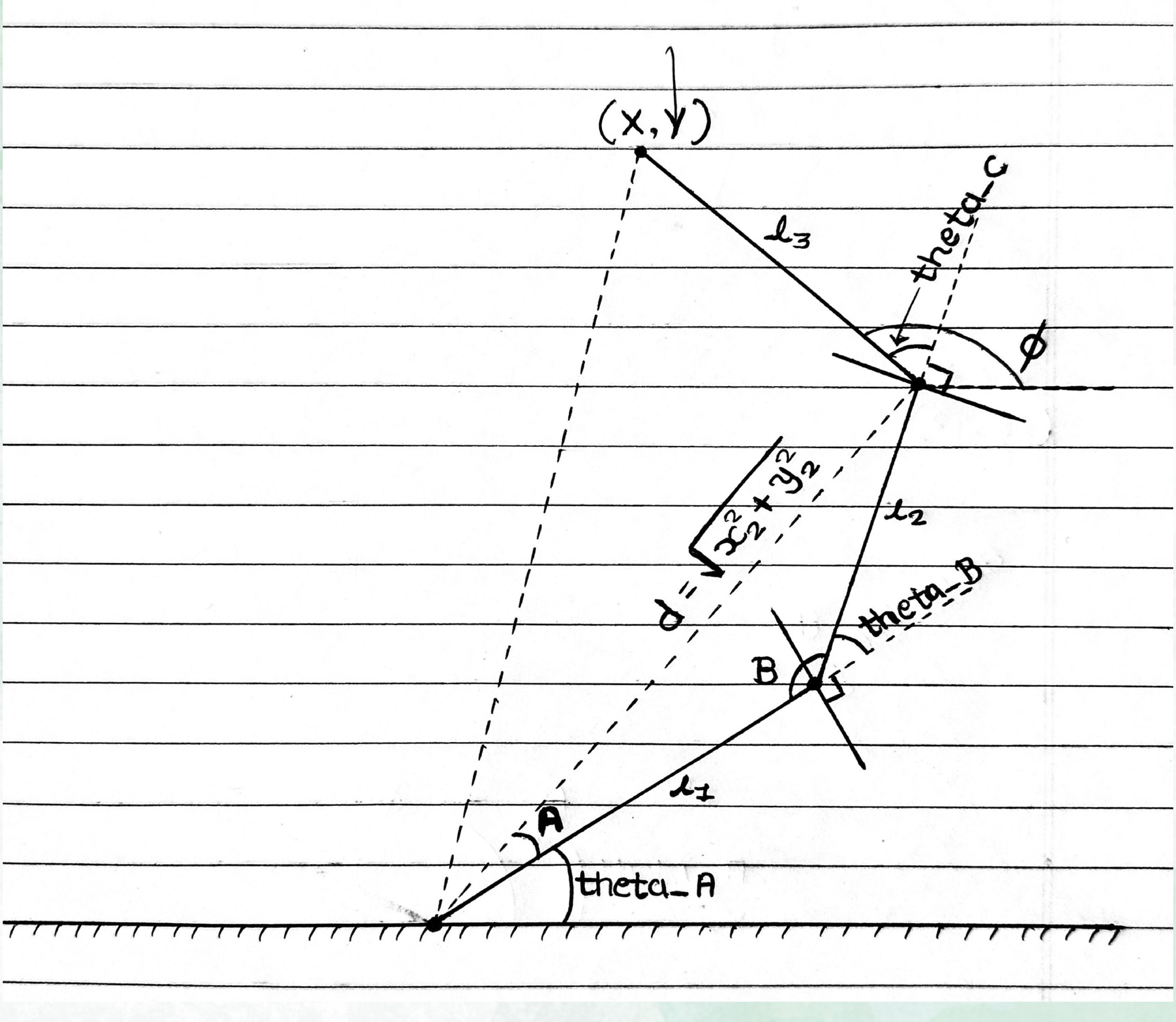
DA-IICT

Group Name : 4_NANO
(Group - 6)

Group Member

Darshan Gami	202201205
Priyank Ramani	202201497
Elvis Kotadiya	202201526
Priyansh Lathigara	202201449

Diagram :



Code :

```
#include <Braccio.h>
#include <Servo.h>

Servo base;
Servo shoulder;
Servo elbow;
Servo wrist_rot;
Servo wrist_ver;
Servo gripper;

const float l1 = 12.5;
const float l2 = 12.5;
const float l3 = 7.15;

void IK(float X, float Y, float phi) {
        float x2;
        x2 = X - (l3 * cos(radians(phi)));
        float y2;
        y2 = Y - (l3 * sin(radians(phi)));

        float d = sqrt(pow(x2, 2) + pow(y2, 2));

        if ((l1 + l2) >= d)
        {
             float A = degrees(acos((pow(l1, 2) + pow(d, 2) - pow(l2, 2)) / (2 * l1 * d)));
             float theta_A = degrees(atan2(y2, x2)) - A;
             float B = degrees(acos((pow(l1, 2) + pow(l2, 2)- pow(d, 2)) / (2 * l1* l2)));
             float theta_B = 180 - B;
             float theta_C = phi - (theta_A + theta_B) ;

             Braccio.ServoMovement(20,0,int(theta_A),90+int(theta_B),90+int(theta_C),65,73);
        }
}
void setup() {
        Serial.begin(9600);
        delay(1000);
        Braccio.begin();
}
void loop() {
    float x =  //(input x);
    float y =  // (input y);
    float phi =  // (input phi);
    Braccio.ServoMovement(20, 0, 15, 180, 170, 0, 73);
    delay(1000);
    IK(x, y, phi);
    delay(5000);
}
```

Explanation :

- In this code, firstly we define lengths of links of our robot arm as follow l1=l2=12.5(cm), l3=7.15(cm).
- Now we define a function named IK in which we give arguments as coordinates of end effector of link-3 (X, Y) and the orientation angle(phi).
- From given end effector of link-3 we calculate the coordinates of link-2 where it ends.
- After this we find the distance of this link-2 from the origin.
- By finding this distance we check that our robot arm is capable to reach that coordinate or not.
- If it is possible than we find the angles A and B from given link's length by applying the cosine rule in triangle PQR. From this we find theta_A and theta_B.
- For theta_C we have to subtract (theta_A+theta_B) from phi.
- After doing this calculation of angles, we give these angles as input to the robot arm as following: Braccio.ServoMovement(20,0,int(theta_A),90+int(theta_B),90 +int(theta_C),65,73).
- In the looping function we set the coordinates and orientation angle and pass this into IK function.
- We are also setting up the robot arm to move to the default position and for getting the exact observation we give the delay also.

Procedure :

- First connect Arduino to the computer using cable and join Arduino with compatible shield.
- Now we give three different coordinates(X,Y) with respective orientation(phi).
- In first case we given X=0, Y=32.14 and phi=90 degree. From this we expect angles after hand calculations :

theta_A=88.38 degree, theta_B=3.25 degree,
theta_C=-1.63 degree.

From this angles input position is as follow:

(20 ,0,88.38,93.25,88.37,65,73).

- In second case X=12.5, Y=19.64 and phi=0 degree. From this we expect:

theta_A=42.7489 degree, theta_B=68.23 degree,
theta_C=-110.9729 degree.

From this angles input position is as follow:

(20,0,42.7489,158.23,-20.97,65,73).

- In third case X=8.83, Y=28.48 and phi=45 degree. From this we expect:

theta_A=61.90 degree, theta_B=36.15 degree,
theta_C=-54.47 degree.

From this angles input position is as follow:

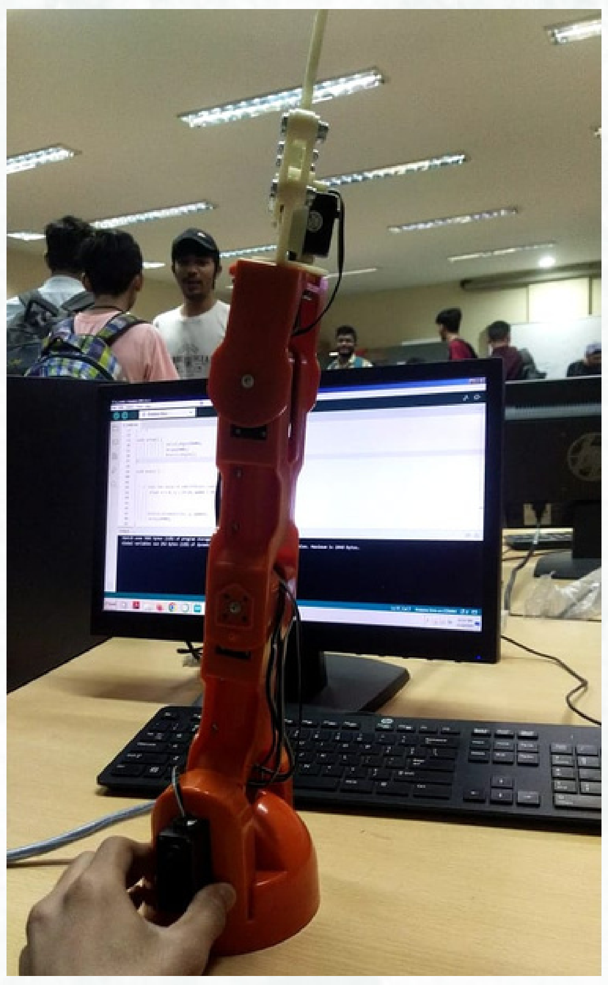
(20,0,61.90,126.75,35.53,65,73).

- Upload the codes as following values to the Arduino Uno via the connected cable.

Observation:

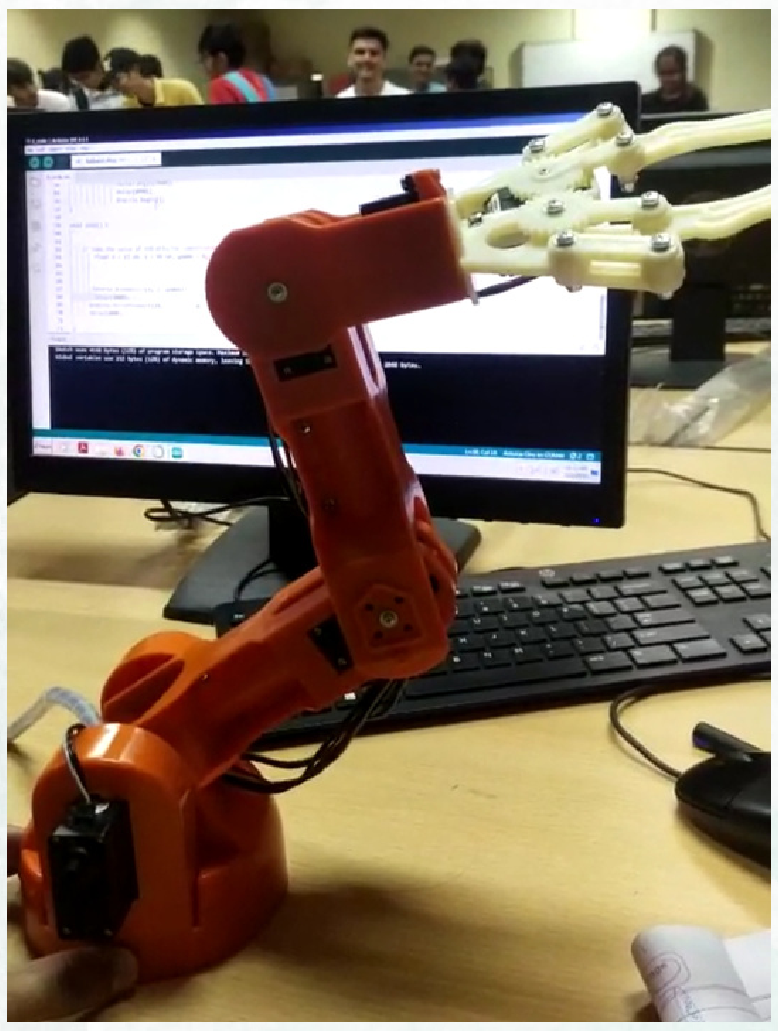
- Intially as given in code for all conditions robot arm will take its default position as (20, 0, 15, 180, 170, 0, 73) after delay of 1 second.

  - In first case we given input X=0,Y=32.14 and phi=90 degree, our output will display as following: theta_A=88.51 degree, theta_B=3.24 degree, theta_C=-1.62 degree. From this our robot arm will take position as (20,0,88.51,93.24,88.38,65,73) which is shown in diagram.



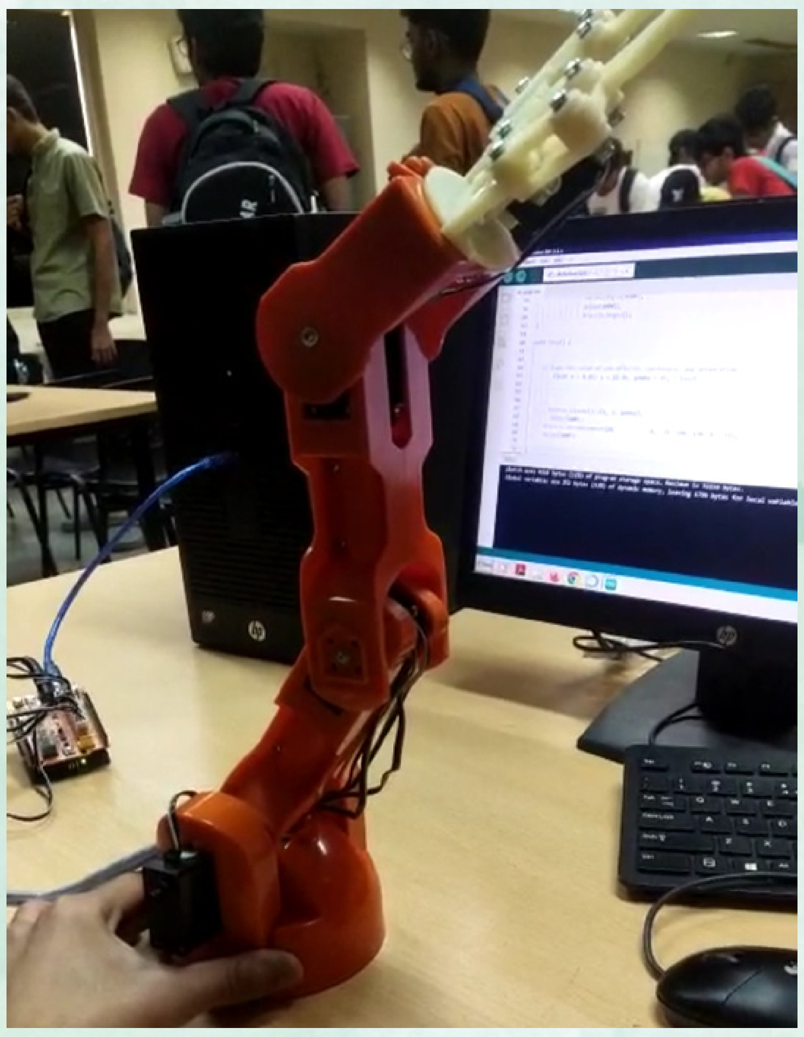
Observation - 1

  - In second case we given input X=12.5, Y=19.64 and phi=0 degree, our output will display as following: where theta_A=39.29 degree, theta_B=71.01 degree, theta_C=-20.30 degree. From this our robot arm will take position as (20,0,39.29,161.01,-20.30,65,73) which is shown in diagram.



Observation - 2

  - In third case as we given input X=8.83, Y=28.48 and phi=45 degree, our output is display as following: theta_A=62.51degree,theta_B=36.75 degree,theta_C=-54.24 degree. From this our robot arm will take position as (20,0,62.51,126.75,35.76,65,73) which is shown in diagram.



Observation - 3

- After delay of 5 second, we shown that this process is continuously repeating.
- We observe that these parameters are nearly same as we calculate previously with hand calculation.

Result :

For given end effector coordinates and orientation we successfully get angles using inverse kinematics concepts and accurate positions for robot arm as we calculated.

Video Link : Tap here