4 Dof robot arm

Introduction

To build robotic system (at least like this project) it is not only programming and electrical field, but also required mechanical understating.

Kinematic is a subfield in classical mechanics that describes the motion of points, bodies (objects), and systems of bodies (groups of objects) without considering the forces that cause them to move.

Then what the correlation of this mechanics subfield study to programming robot? Programming a robot which mean we give a 'set of instruction' where it moves on the environment, with any kind of tools and variety of job, but still, the principle is basically constructing the movement trajectory of a robot, then later we can come up for the programming.

In robotics nomenclature "end effector" is the device at the end of a robotic arm.

The end effector means the last link (or end) of the robot. At this endpoint, the tools are attached.

In a wider sense, an end effector can be seen as the part of a robot that interacts with the work environment.

By understanding definition of End Effector from previous paragraph now let’s model the robot arm end effector position as cartesian coordinate system, because robot work in 3-dimensional space then its cartesian coordinate axis consist of x axis, y axis, and z axis.

In this case the robot arm is 4 DOF and the hand (end effector) is gripper module, so by modeling the position of This gripper on environment now we can have a method to determined where it moves in order to do a job we want.

Let say we want this robot arm to pick up object at certain position, that position is then defined by value of each x, y, z axis (the unit of axis value are same as distance unit, e.g. Mm) then how much the angle of each joint should rotate in order the end effector of the robot reach that point are solved by implementing some mathematical equation (geometric formula), this is what Inverse Kinematic method are, a method that converting cartesian coordinate of robot end effector position to polar coordinate (angle) of each joint.

Because each joint of this robot arm driven by electric motor (servo motor) then the value of angle each joint calculated from geometric equation is the value of how much each these servo motor on the system should rotate in order to make the gripper reach the position or move according to trajectory movement of the end effector (gripper) as we want.

Step 1: List of Parts Needed

5 Servo Motor Metal Gear MG996R

metal gripper

2- L-Bracket

4- U-bracket

5-sevo motor bracket

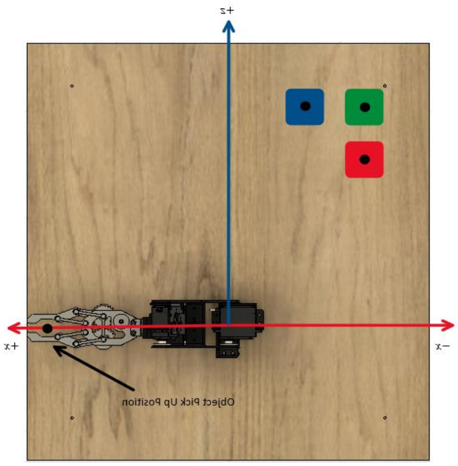
power supply

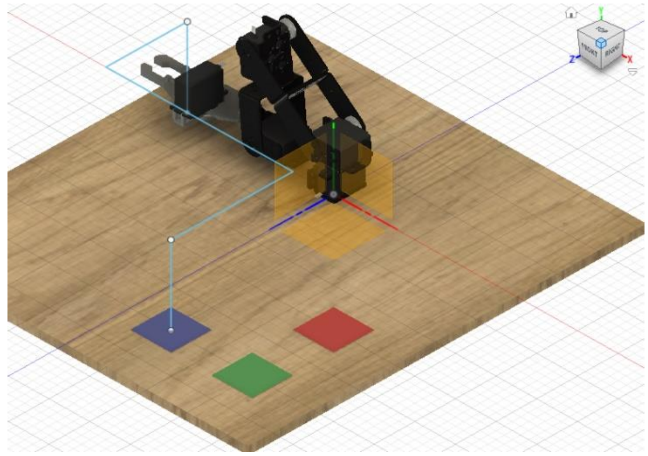
Robot base

Arduino uno

power cable

Step2: How It Works





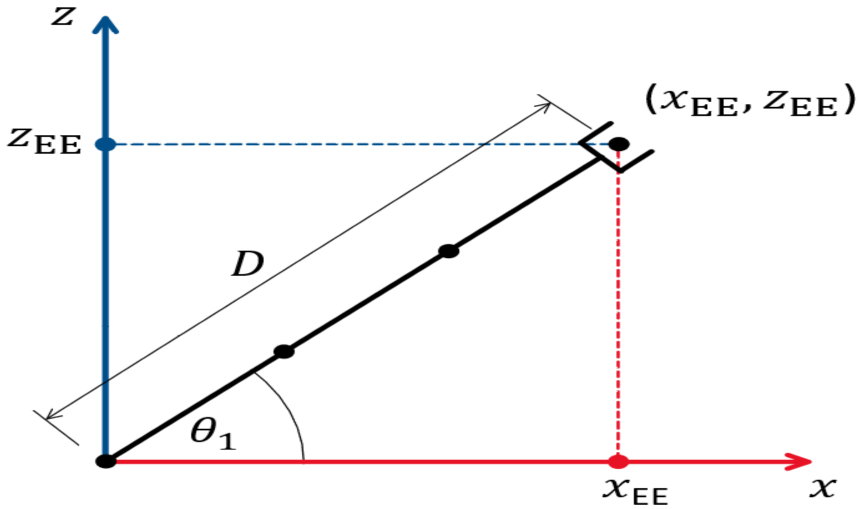
Before assembling the parts or component it is better to understand what exactly this robot arm is supposed to do.

To make this robot arm as pick and place object sorter based on color then first locate or determined position of where the object is coming and need to pick up, then where it should be moved according on its quality. The position where the robot grip and pick up the object is where the sensors are placed, HC-SR04 ultrasonic sensor detect the presence of object and The MATLAB will send zero or one to the arm Arduino, if it was one, the robot arm moves that object to designated location, and if it was zero the robot arm moves that object to another designated location.

The vertical direction are y axis, x and z axis are the surface, the z is 'forward' direction, x is 'side' direction. Zero point for y axis is the surface, x and z zero point are where center of first joint. First joint is rotational joint which perpendicular to its surface, or in other word first joint is rotational at y axis.

By modeling the movement of robot end effector with cartesian coordinate system general overview writing program for operating the robot is you just need to change the value of x, y, and z position of the end effector. Sounds easy, right? But programming servo motor with Arduinos Servo. Write () you need to input degree value how much the servo you want to rotate. So that's why we need implement some geometric formulas that could convert position (given value of x, y, and z) of end effector then calculate to give output value of angle each joint should rotate. Then we should go to next step to understanding some Kinematics formulas.

Step 3: The Inverse Kinematics (Part 1)



$$D = \sqrt{(x_{EE})^2 + (z_{EE})^2}$$

$$\frac{z_{EE}}{x_{EE}} = tan(\theta_1)$$

$$\theta_1 = tan^{-1}\left(\frac{z_{EE}}{x_{EE}}\right)$$

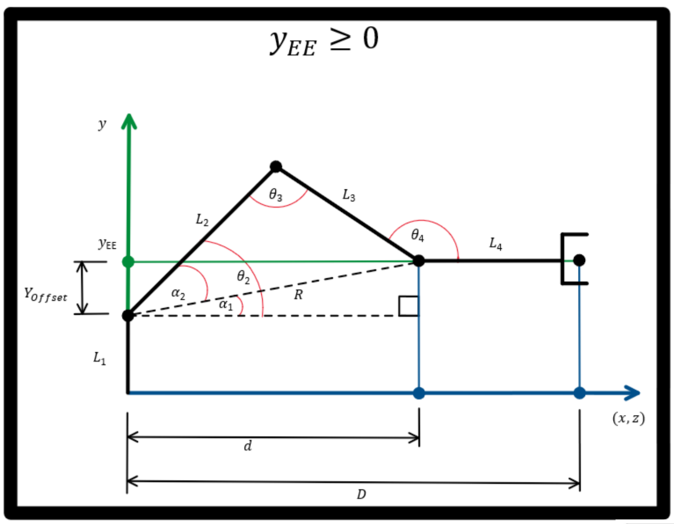
Given angle value each joint of group of bodies or mechanical system then calculated to give output of position coordinates where the end effector of the system is expected is called 'Forward Kinematics'.

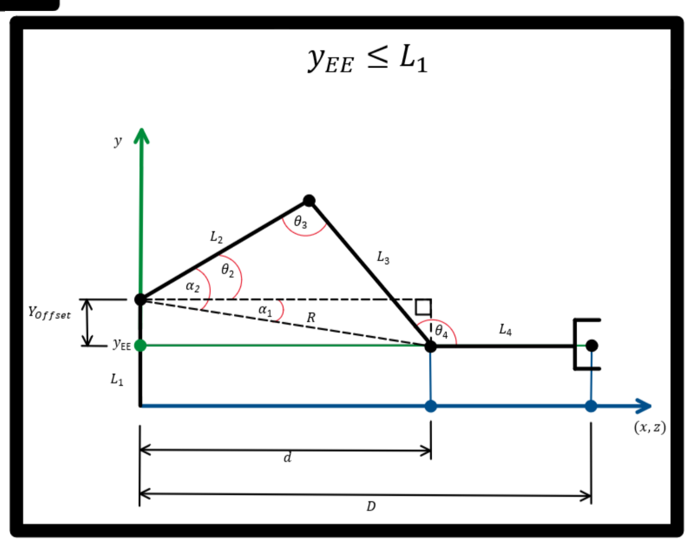
Forward kinematics of course sounds not suitable to operate a robotic system, we want the opposites, determined the path, or trajectory of robot end effector where to move, then calculate how much angle each joint should rotate. Because when programming a robot, we need to send 'signal' from controller board to the actuator, the signal contains value of information how much angle servo motor rotate that drive the robot joints, hence why we need Inverse Kinematic.

In this project, the inverse kinematic equation will use a geometric method to determine the joint angles of the robot arm to reach desired end effector position. For reducing the complexity, we reduce the problem from 3D to 2D point of view. Now, let’s assume we look at the robot arm from above, which represent plane (x, z).

To make the gripper (end effector) reach desired coordinate position the first joint should rotate at variable declared as "Theta 1", beside the first joint rotation, the arm needs to span to make end effector reach desired coordinate position. Let's symbolize variable how far the arm needs to span as "D". Calculation for Theta 1 and D is shown on picture above.

Step 4: Inverse Kinematic (Part 2)





$$\boldsymbol{y_{EE} \geq L_1}$$

$$d = D - L_4$$
$$Y_{Offset} = y_{EE} - L_1$$
$$R = \sqrt[2]{(d)^2 + \left(Y_{Offset}\right)^2}$$
$$\frac{d}{R} = cos(\alpha_1)$$
$$\alpha_1 = cos^{-1}\left(\frac{d}{R}\right)$$
$$(L_3)^2 = (L_2)^2 + (R)^2 - 2 \times L_2 \times R \times cos(\alpha_2)$$
$$\alpha_2 = cos^{-1}\left(\frac{{L_2}^2 + R^2 - {L_3}^2}{2 \times L_2 \times R}\right)$$
$$\theta_2 = \alpha_1 + \alpha_2$$
$$R^2 = (L_2)^2 + (L_3)^2 - 2 \times L_2 \times L_3 \times \cos(\theta_3)$$
$$\theta_3 = cos^{-1}\left(\frac{{L_2}^2 + {L_3}^2 - R^2}{2 \times L_2 \times L_3}\right)$$
$$\theta_4 = 180° - \{[180° - (\alpha_2 + \theta_3)] - \alpha_1\}$$

In the previous section we have implemented formula from geometric principle which can be used to calculate the rotation angle of joint 1 (Theta 1), now we are analyzing another 2D viewpoint to determine the equation to find the other three joint angles (Theta 2, Theta 3, and Theta 4). In short, how much angle joint 2, 3, and 4 needed to rotate to make the arm form span from length that calculated before (D).

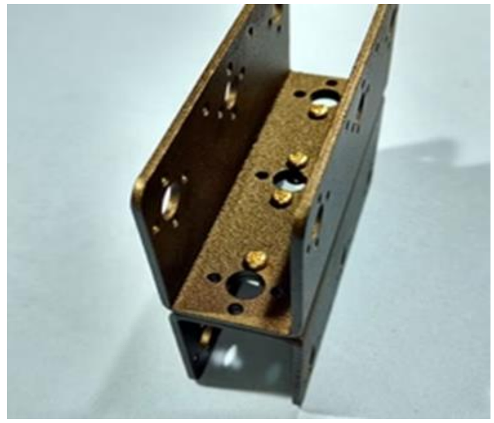
Calculation to find Theta 2 (joint 2 rotation angle) and Theta 4 (joint 4 rotation angle) when position of gripper (end effector) lower than joint 2 position in y axis direction a bit requires different formula than if end effector position higher than joint 2 position.

This inverse kinematic formula only for to make link 4 (L4) parallel to the ground/surface, but since this robot is 4 Degree of Freedom (4dof) so there is flexibility to not only determined position where its end effector should move/reach but also the orientation of the end effector, but it require different formula.

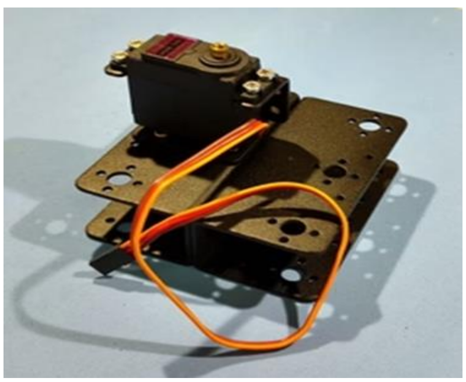
Anyway, there is 6 Degree of Freedom 'law' of rigid body movement in three-dimensional space, there are; translational move along axis x, y, z (3 dof), and rotational move along axis x, y, z (3 dof), so total is six. This robot end effector capable all three-axis direction translational move freedom, and one rotational freedom, so hence why it is 4 dof robot.

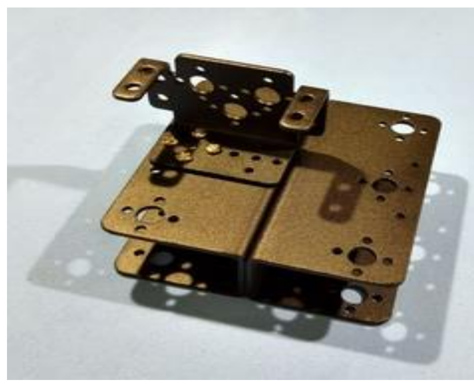
Step5: Assembling the robot arm.

1. The base



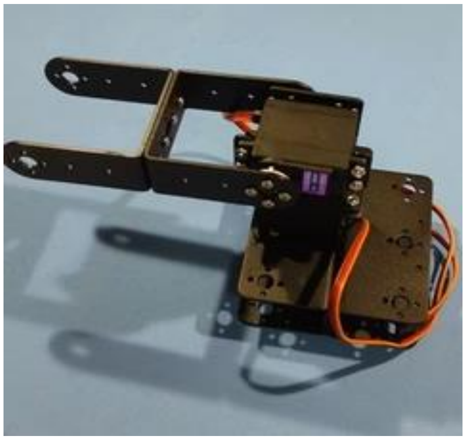
2. First joint

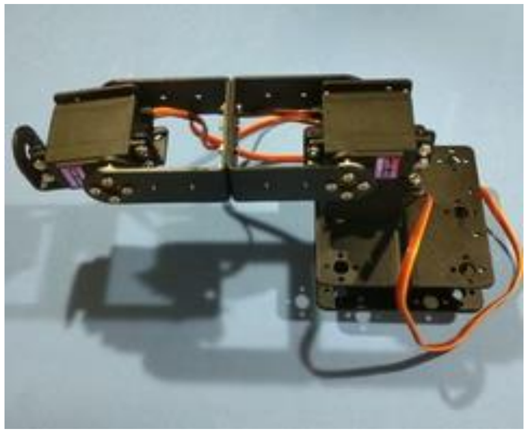




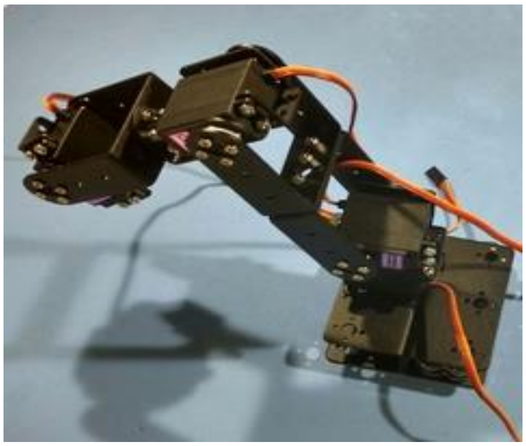
3. Second joint

4. Third joint



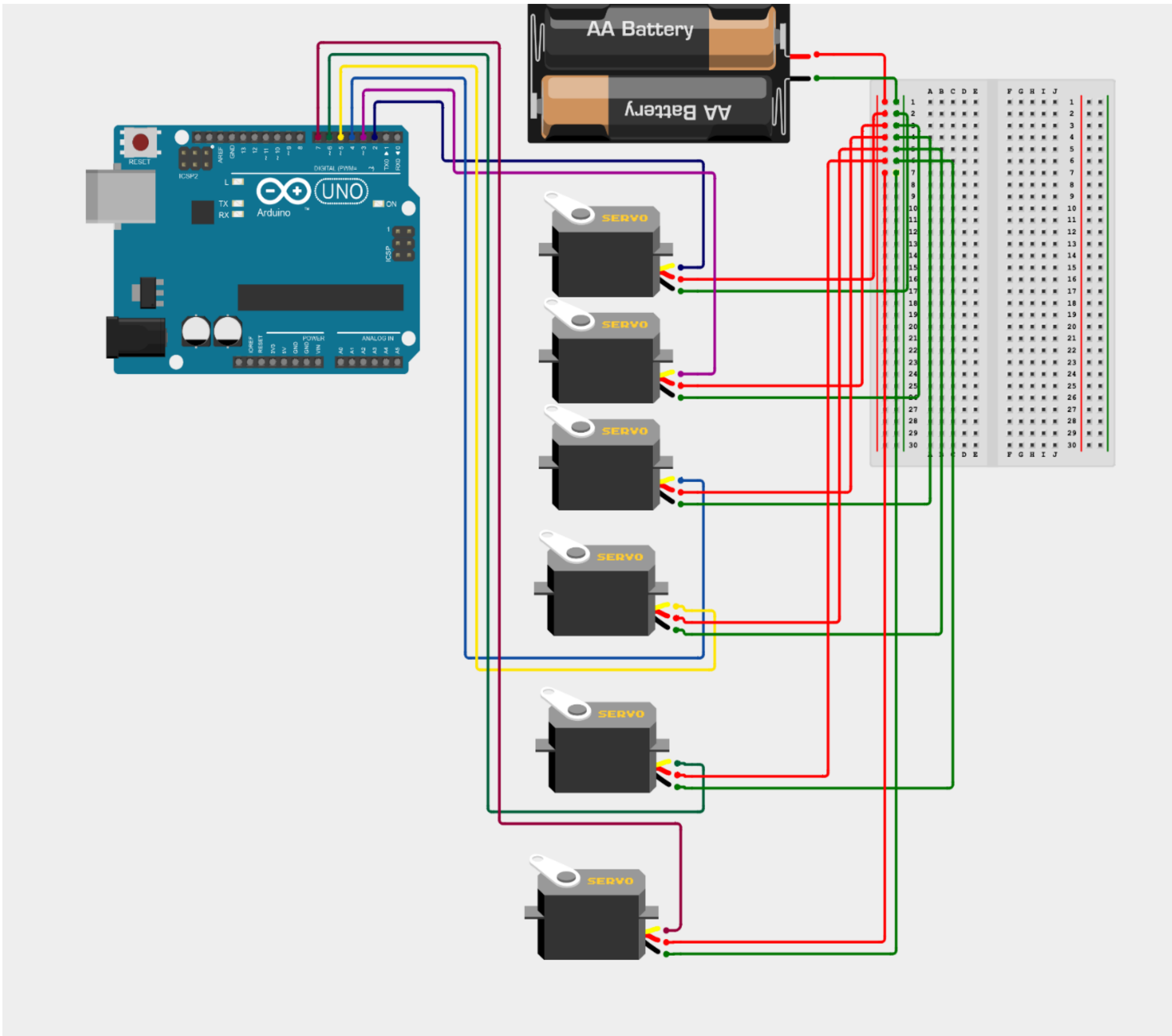


5. Fourth joint



Completed four degree of freedom robot arm.

Step 6: Wiring



- AC-DC 12V Power Supply.

  If you see a terminal (Philips head style bolt) that parallel to "L N" label that the input electric from your building wall socket, so you need 3 wires, 1 to L, 1 to N, and 1 other for ground, the ground terminal is beside the L and N terminal. Next to the right terminal of L N and Ground, is the output terminal. Electricity from this terminal is in form of DC current. "COM" label terminal is the negative and "V+" terminal is the positive.

- HC-SR04 ultrasonic sensor.

  There are 4 pin in this module, GND, Echo, Trig, and Vcc.
- Output terminal of XL4015 converter connect to VCC and GND, VCC is positive GND is negative. Echo pin connect to pin 7 of Arduino UNO board, Trig pin connect to pin 8 of Arduino UNO board.
- Servo motors.

  Servo motor have 3 wires, black (or brown), red, and yellow.
- Red and black are where power comes in, which mean positive output terminal of PR15 converter connect to red wire, negative output terminal of PR15 connect to black or brown wire.
- Yellow wire is 'signal' wire from controller (Arduino board). Yellow wire for joint 1 connect to pin 9, for joint 2 connect to pin 5, for joint 3 connect to pin 10, for joint 4 connect to pin 6, and servo to drive grippe connect to pin 3.

Step 7: code

Appendix C