



IMPLEMENTATION OF ROBOT JOGGING AND FORWARD KINEMATIC ANALYSIS USING 5 DEGREES OF FREEDOM ROBOTIC ARM

^[1]Raghav Kejriwal, ^[2] Mahesh Pradhan

^[1] Student, School of Mechanical Engineering, Dr. Vishwanath Karad MIT World Peace University, Kothrud, Pune-411038

^[2] Associate Professor, School of Mechanical Engineering, Dr. Vishwanath Karad MIT World Peace University, Kothrud, Pune-411038

Abstract: Articulated Manipulators or Robotic Arms, as we know them, are state of the art mechatronics equipment that is at the pinnacle of the automation technologies. They have revolutionized the production processes in various industries, especially in the automobile and electronics sector. Being so advanced, they are not accessible and affordable to many, who aspire to become a robotics engineer and study robotics. The purpose of this paper is centered around the design, development and analysis (using forward kinematics) of an Arduino based 5 degrees of freedom (d.o.f) pick and place robotic arm which is manufactured using 3D Printing technology and uses servo motors. However, the development of the robotic arm itself is not sufficient unless we understand how robots are taught to do the task they are intended and developed to perform. Therefore, this paper also focuses on the various methods of Robot Teaching also known as Robot Jogging and the development of a Teach Pendant which uses potentiometers and buttons as its core components. This enables the user to teach the robotic arm to perform a required task.

Keywords: *Arduino, Forward Kinematics, Mechanical Gripper, Robotic Arm, Robot Jogging, Robot Teaching, Teach Pendant, 3D printing.*

I. INTRODUCTION

The word Robot has been derived from the Czech word “robota” which means forced or slave labourer. Robotics is a science, which deals with the design, manufacturing and usages of robots. Robotics is the perfect example of a multidisciplinary approach to engineering. To tackle the escalating demands of a dynamic and competitive market, manufacturing methods used in this modern era should be able to reduce production costs and increase productivity. Automation is the only alternative to achieve these requirements. Automation can be categorized into Hard and Flexible Automation. The use of Robotic Arms comes under Flexible Automation. The articulated manipulator is the one that is most similar to the human arm. An articulated manipulator is commonly seen in robotic applications that require all kinds of complicated rotating and articulate motion. Common uses of articulated manipulators include processes such as welding, painting, packaging etc. Not only industrial automation but these robots either in the form of articulated manipulators or robots resembling living creatures such as a hexapod robot and even drones have found their applications in the Medical and Defense sectors as well. Robotics applications can be divided into two segments. Industrial and Service Robotics. Service Robots, as defined by the International Federation of Robotics (IFR), are those robots that are capable of functioning either semi or completely autonomously to carry out tasks that are related to the welfare and convenience of human beings. These tasks include medical surgical procedures, agricultural activities and defense applications such as defusing bombs or a hexapod robot, as a disguised Monitoring/Spying Robot [8].

The work aims at designing and developing a 5 d.o.f robotic arm by 3D printing technology, that uses a mechanical gripper for performing pick and place operation. It is an Arduino based robotic arm that serves as a controller for the servo motors which are responsible for enabling the degrees of freedom for the robotic arm. To control the robot, a control module needs to be developed which is called the Teach Pendant. It consists of 6 potentiometers, 5 potentiometers for controlling the respective servo motor located at each of the joints of the robotic arm and 1 for the gripper. There will be 4 buttons to perform “SAVE”, “RUN”, “PAUSE” AND “RESET” functions. This depicts the Robot Teaching Method. To analyze the motion and the behavior of the arm as it is moved through specific co-ordinates in space, Forward Kinematics is used. This analysis is achieved using the RoboAnalyser software.

II. LITERATURE REVIEW

A Robot arm is a type of mechanical arm, mostly programmable, which is meant to mimic the functions of a human arm and it is mainly used in different applications such as industrial, military and medical areas [2]. Kadirimangalam Jahnvi and Sivraj P. [1] have explained the design of a robotic arm to control the end-effector position using robot kinematics. The robot arm performs pick and place application and kinematic control of the robotic arm is done using MATLAB software. The ATmega2560 based Arduino microcontroller is used for communication between the MATLAB GUI and motors of the OWI robotic arm. Christos Tolis and George F. Fragulis [4] have done the modelling and control of 5 d.o.f robot arm using D-H parameters. Their study aims at formulating the forward and inverse kinematics algorithms of their proposed robotic arm for pick and place application. Calculations have been done in order to find out the torque requirements at each joint and to find out the Payload of the robotic arm. Sayali V., Poonam G., Kalyani D., Rasika S. [8] have developed a 3 d.o.f manipulator that is meant to depict the human wrist. In order to control all the 3 degrees of freedom, they have developed a mobile app from where they can control the motion of each joint as per the requirements and the communication between the app and the robot takes place with the help of Bluetooth HC-05 module and Arduino. Hooman Lee and Joongbae Kim [11] have researched the various Robot Teaching and their categorization. They have segmented the robot teaching technologies according to the level of programming skill and then divided them into two types according to the teaching layer. They have covered various teaching methods used in industries to teach robotic arms according to the operator skill level and the infrastructure available with the industry, which are very crucial aspects of the Robot Teaching Method. Samir Raval and Bhavesh Patel [12] have done research on the various end effectors and the gripping mechanism used in robotic arms in today's day and age. They have categorized the robotic arm grippers, talked about the grasping principles, grasping and releasing procedures and the aspects that are considered while designing and developing a gripper such as the grasping force of a gripper.

The above literature review helps to understand the significance of the various factors that are vital to the designing and the development of a robotic arm that provides both standalone operations for pick and place application and as a laboratory model for performing experiments to gain hands-on experience on robotic arms and for educational purposes.

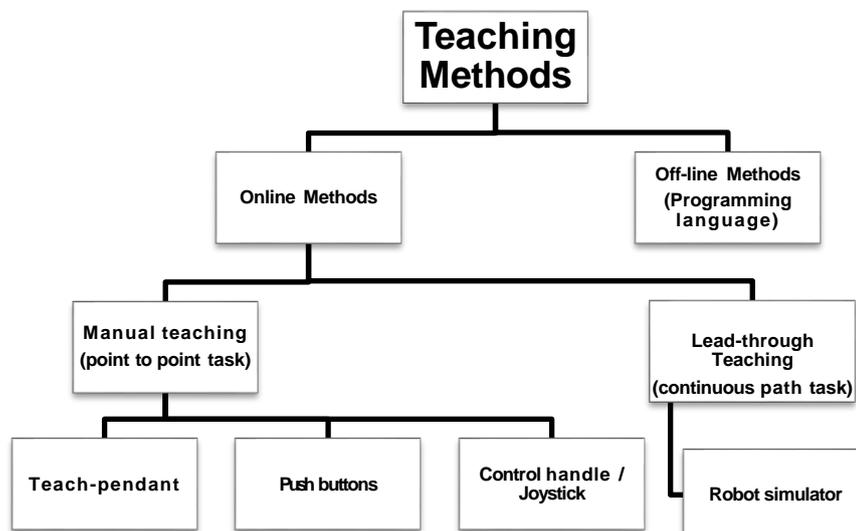
III. METHODOLOGY

1. Selection of Robot Teaching Method:

To extract the full potential of a robotic arm, it is essential to develop a method where the user is able to interact with the robot and control it. But there are some barriers to achieve this goal. The first obstruction is how the user can teach the robot in a simple yet effective way where the user can comfortably control the robot as per his/her will and can take full advantage of the technology used in the robot. The second being, how the robot learns the commands given by the user and responds to it. In order to provide necessary commands and instructions to the robotic arm, it is vital that we use the Robot Teaching Method. The process of teaching or programming a robot in order to carry out a specific task is known as Robot Jogging. It essentially means that we jog the robotic arm through a path or series of co-ordinates to do the intended task. [11]

These robot teaching methods depend upon:

- Type of robots used
- Configuration of the robotic arm
- Application of the robot
- Teaching Hardware available
- Programming language used
- The skill of the operator/teacher



The above hierarchy gives an overview of the various kinds of teaching methods used by industrial experts to teach robotic arms for industrial applications. As the accessibility of robots to unskilled operators is increasing, it is necessary that the methods we use to program and manage robotic arms should become easier. Manual programming methods are still dominant in industrial robotics because of accuracy, convenience and reliability. However, the limitations to such methods are that these methods are cumbersome and difficult for people new to such jobs. Robots are becoming integrated into our lives at an exponential rate and so is the number of people who are working in the development of such mechanisms. Thus, new methods are needed to be developed that serve the requirements of operators who are non-programmers so this method becomes more intuitive and more manageable for operators. From the above discussion and evaluating the limitations and advantages of various robot teaching methods, the most suitable and widely used method of Robot Jogging is the use of Teach Pendant. [11] [2][8]

Teach Pendant: A Teach Pendant also called a Teach Box is a handheld control and programming unit connected to a robot via a cable. They usually incorporate buttons, joystick and touchscreens (advanced teach pendants) all interfaced by a GUI. All teach pendants include an emergency stop button in case something unexpected happens. With the use of Teach Pendant the operator is able to move the robotic arm point by point while recording/saving those co-ordinates in the robot's memory to execute a cycle of operations. This way the operator is able to "Jog" the robot to the desired location. Not only positioning but the Teach Pendant can also control the speed of each joint of the arm which is necessary while test running through a new or modified routine. Once the operator has taught the robot and the test run is complete, the teach pendant is disconnected the robot can run at any desired speed [11]. ANSI standards recommend a speed of about 10 in/s while programming the robot.

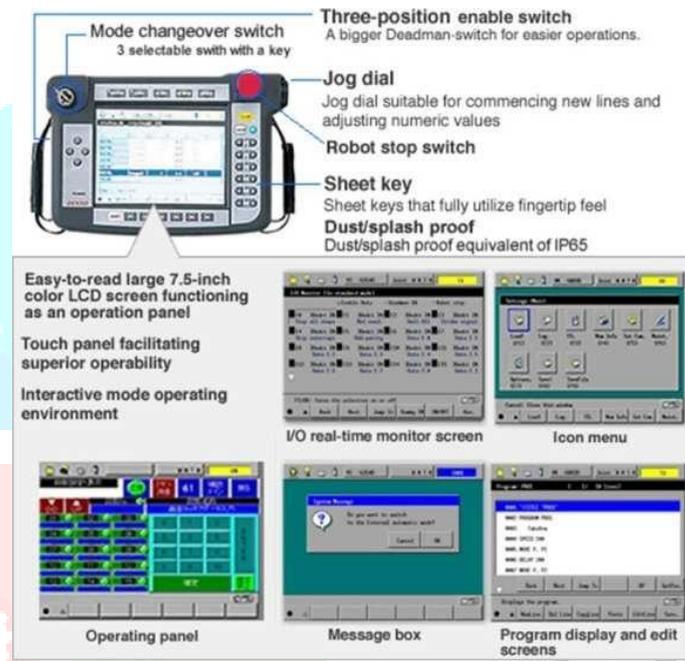


Fig.1. Teach Pendant developed by Denso Robotics [11]

2. Selection of the Gripper/End effector:

An End Effector (EE) is a device that is fixed to the wrist of the Spatial manipulator so as to hold materials, parts or tools to perform a series of tasks such as welding, cutting, painting etc. EE can be either a Gripper that is used to pick, hold or carry objects or the EE can be in the form of a tool such as a Spot Gun for welding or Spray Gun for painting. The proposed model uses a Gripper. The gripper is a very important part of the manipulator because with the help of gripper only, the robotic arm can interact with its environment. Grippers are specifically designed for the kind of object they are handling and so they don't have a standard design. [12]

The factors that are taken into consideration while designing a gripper are:

- The orientation of the object to be handled w.r.t the gripper
- Nature of the object to be handled by the gripper
- Gripping force required
- Specific Operation that is needed to be performed while carrying the object

Considering the above factors, 2-Jaw Electro-Mechanical gripper is the most suitable type of gripper for the operation performed by the proposed model which is the Pick and Place operation. But we also need to take into consideration the complexity of the grasping action. Grasping action not only depends upon the nature and the type of object to be handled but also depends upon the process of handling, positioning and the process of releasing the object. Fig.2. describes the grasping process of the proposed gripper.

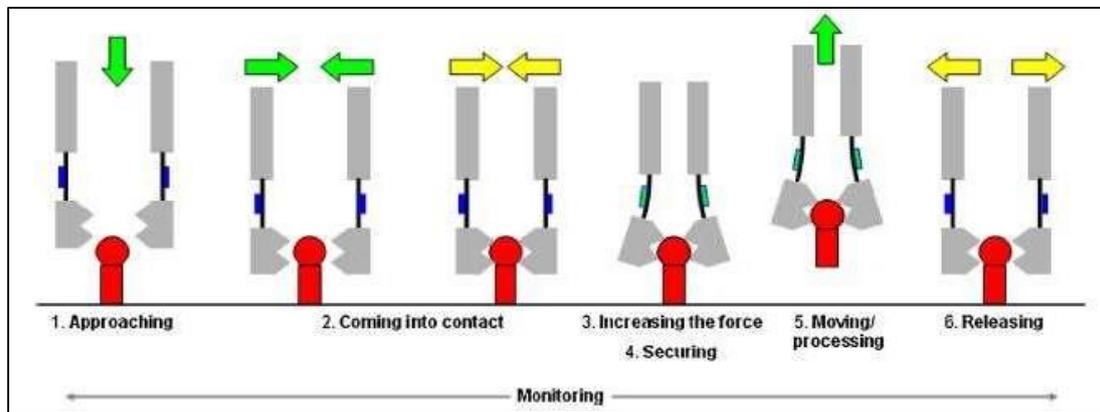


Fig.2. Process of grasping of object by the gripper [12]

The process shown in Fig.4. can be described as:

- 1) Approach: The first step of the grasping process which is the positioning the gripper near the object to be handled.
- 2) Contact: If the gripper is a jaw/mechanical/ vacuum gripper, it is the next step where the gripper attaches/ holds the object. In the case of an electromagnetic gripper, since the grasping process is contactless, the object remains in the force field set up by the gripper.
- 3) Adjustment of the gripping force: The gripping force should be appropriate. If not adequate, the part may slip and get damaged. If the gripping force is more than required, the object may get deformed. The right amount of gripping force is necessary because if the object is slipping while the arm is carrying it, it will introduce a redundant degree of freedom. This is not desirable when the object needs to be handled in a specific orientation. Therefore, the gripper should grasp the object in such a manner that the object becomes a part of the gripper while carrying it.
- 4) Processing: Moving the object to the desired location where it's needed to be dropped.
- 5) Release: The step where the gripper releases the object. Depending upon the nature of an object, either it is released under gravity or the robot is programmed in a manner such that the object is placed carefully on the shop floor or the conveyor belt. [12]

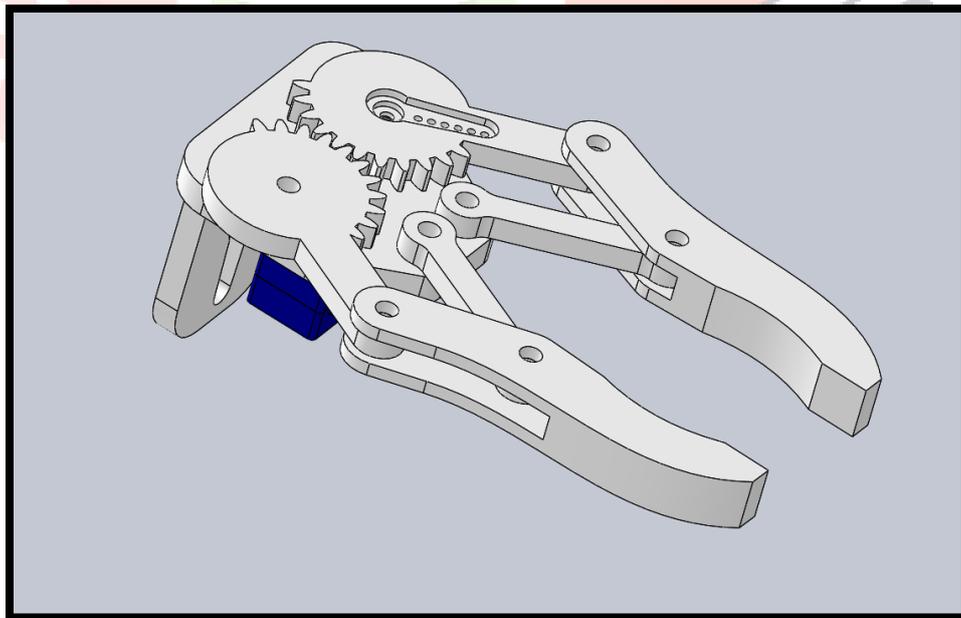


Fig.3. Proposed Gripper

Fig.3. shows the CAD model of the 2-Jaw Electro-Mechanical gripper designed using SolidWorks. The above gripper has been categorized as an Electro-Mechanical gripper because it uses an electrical device i.e. a servo motor as it's actuator and gears for opening and closing of both jaws at the same time. The gripper uses a SG90 servo motor as the actuator. The SG90 servo motor provides sufficient torque to overcome friction and provide adequate gripping force for various objects.

3. Objective of the System:

The design and the development of the robotic arm is carried out according to the pick and place operation. The robotic arms used for such operations can relieve the user from the tedious task of sorting. It can also be used for handling heavy loads where earlier labour was required to carry out this task in industries. Prolong exposure to such activities pose a risk to the health of the workers. Eliminating human error, the system emerges as a highly efficient and precise tool for such tasks. Therefore, using robotic arms in industries helps in reducing processing time, mistakes, money, manpower, wastage of resources etc. and increase productivity, worker safety, quality of production etc. [7] [10]. Hence an effort has been made to develop a scaled-down model of the industrial robotic arms.

4. Proposed System:

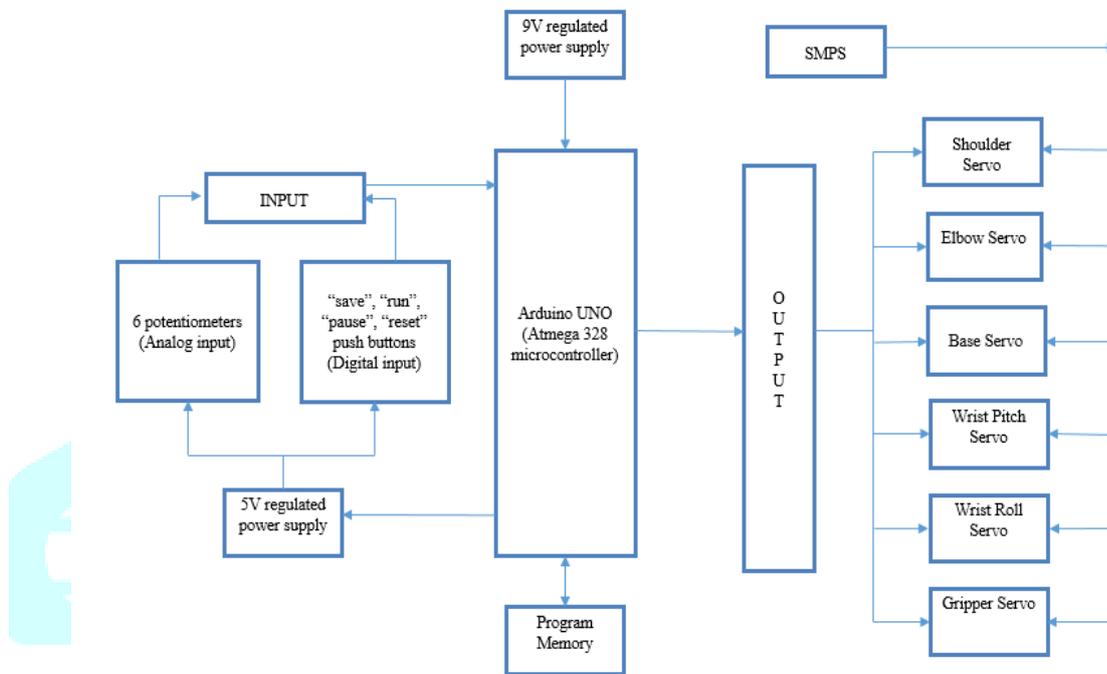


Fig.4. Block Diagram of the System

4.1. System Components

- Potentiometers:

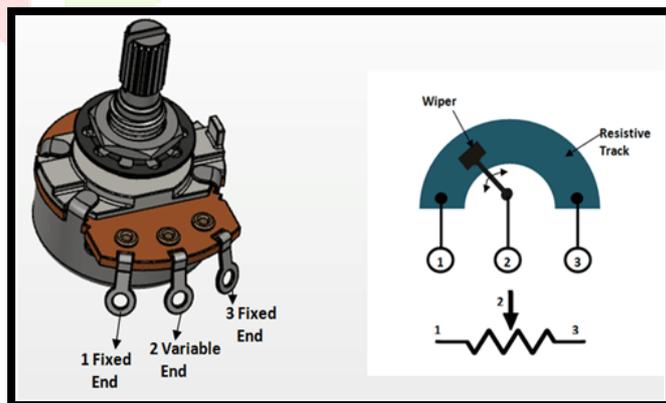


Fig.5. Potentiometer

The system consists of 6 10k-ohm potentiometers, as shown in Fig.5, for controlling the servo motors present at each joint of the robot and for opening and closing of the gripper jaws. Turning the knobs of the potentiometers will rotate their respective servo motors. Potentiometers being analog devices, they provide analog input to the Arduino and hence they are connected to the analog pins (A0-A5) of the Arduino. The pin1 and pin3 of the potentiometer are the positive and negative terminals respectively and the pin2 is connected to the analog pin of the Arduino. Turning the knob moves the wiper thus varying the resistance between the pin1 and pin2. The potentiometer must be operated at 5V and is therefore provided 5V from the 5V pin of Arduino.

- Push Buttons:

There are four buttons used in the Teach Pendant. The buttons perform the following function:

1. SAVE: Saves the current position of the servo motors as adjusted by the potentiometers.
2. RUN: Makes the robotic arm to run in the automatic mode.
3. PAUSE: Pauses the robotic arm as and when required.
4. RESET: Resets the robotic arm and erases the previously stored positions.

These are 4 pin buttons that are connected to the digital pins of Arduino (D0-D13). They need to be operated at 5V and hence they are provided 5V from the 5V pin of Arduino. In order to limit the current through the buttons to prevent them from damaging and to make the default state of the digital pin, to which the button is connected, LOW i.e. 0V a 220-ohm resistor is used in the Pull Down configuration. The buttons along with the potentiometers together make the Teach Pendant.

- Arduino UNO:

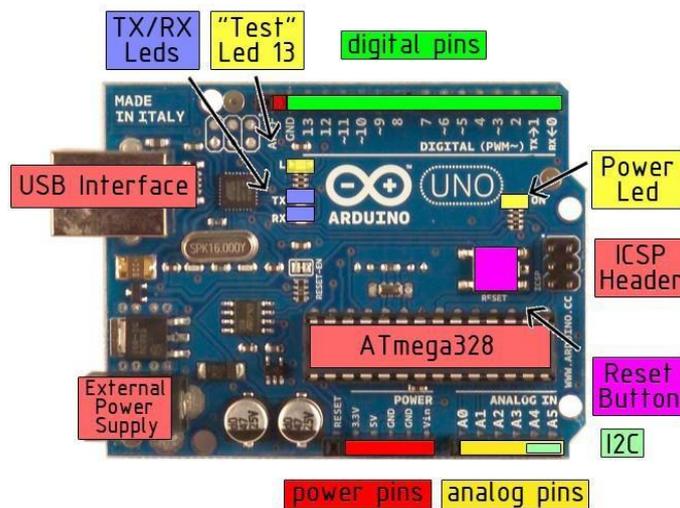


Fig.6. Arduino Uno Board [8]

Fig.6 shows the various pins and ports of the Arduino UNO board. It is a microcontroller based on ATmega328. It consists of 14 digital I/O pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. The Uno differs from other boards because it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega8U2 programmed as a USB-to-serial converter. According to the specifications given below, the Arduino is powered by a standard 9V alkaline battery through the External power supply jack. The coding for the program has been done using the C language using Arduino IDE and uploaded in the program memory of Arduino. [8]

Technical Specification:

Microcontroller: ATmega328
 Operating Voltage: 5V
 Input Voltage: (recommended) 7-12V
 Input Voltage: (limits) 6-20V
 Digital I/O Pins: 14 (of which 6 provide PWM output)
 Analog Input Pins: 6
 DC Current per I/O Pin: 40 mA
 DC Current for 3.3V Pin: 50 mA
 Flash Memory: 32 KB (0.5 KB of it is used by bootloader)
 SRAM: 2 KB
 EEPROM: 1 KB
 Clock Speed: 16 MHz

- Servo Motors:

The selection of motor depends upon the following factors:

1. Torque required
2. Speed required
3. Size of the motor
4. Noise during operation
5. Power consumption
6. Precision and accuracy during operation
7. Cost of the motor

Since the major requirement was the torque, precision and low cost, Servo motor was suitable for the application. The main feature of the servo motor is the ability to control the shaft position with high accuracy and repeatability. In industrial type servo motors, the position feedback sensor is a high precision Encoder, but in the small RC, as used in this system, the position sensor is usually a small potentiometer. The current position of the shaft is recorded by the potentiometer and is fed back to the error detector (control circuit board) where it is compared to the target position. Then depending upon the error, the controller resets the current position of the shaft to match with the target position of the shaft.

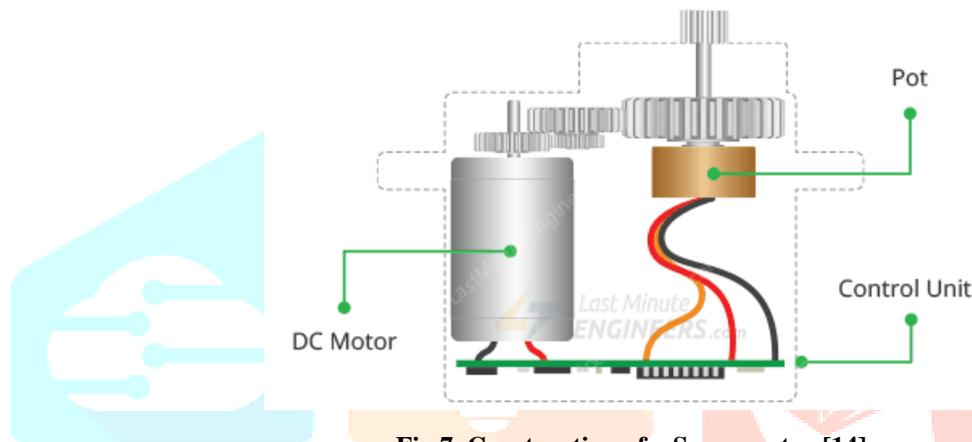


Fig.7. Construction of a Servo motor [14]

Fig.7 shows the arrangement of diff components inside the servo motor. It uses a small DC motor mated to a gearbox for torque amplification. The final driven gear is mounted on the shaft of the potentiometer which is the sensing device of the feedback circuit. In the Control Unit, this potentiometer voltage is compared to the voltage coming from the signal/yellow wire. The control unit uses an integrated H-Bridge which allows the motor to rotate in either direction until the two signals reach a difference of zero. Servo motor is controlled by sending a series of pulses via the signal line. The frequency of these pulses should be 50Hz. The width of the pulse determines the angular position of the servo shaft. Since the servo motor requires Pulse Width Modulation (PWM) for its operation, we can use only D3, D5, D6, D9, D10, D11 pins of Arduino for connecting the signal lines of the motors to the Arduino. One physical limitation of these servo motors is that they can only rotate between 0-180 degrees [2]. The servo motors used in this experiment are MG995 and SG90.

- Technical specifications of MG995 Servo motor:

1. Weight: 55g
2. Stall torque: 9.4kg/cm (4.8v)
3. 11kg/cm (6v) Operating voltage: 4.8~ 6V
4. Current draw at idle: 10MA
5. No load operating current draw: 170MA
6. Stall current draw: 1200MA

- Technical specifications of SG90 Servo motor:

1. weight: 9g
2. Stall torque: 1.8kg/cm (4.8v); 2.2kg/cm (6V)
3. Operating voltage: 4.8 - 6V
4. Current draw at idle: 5MA
5. No load operating current draw: 60MA
6. Stall current draw: 700MA

According to the above specifications, the maximum current (stall current) drawn by all servo motors together at full load will be:

Total Current drawn = $(1.2A \times 4) + (0.7A \times 2) = 6.2A$

Therefore, the maximum current that all the servos together can draw is 6.2A. However, Arduino cannot handle such a huge amount of current if all the servos are powered via Arduino. Thus, we need a separate power source for the servo motors that can supply up to 10A of current at 5V. For this purpose, Switch Mode Power Supply (SMPS) is used that is rated at 5V 10A.

- 3D Printed Parts

The system uses a total of 10 different parts that are 3D printed using Fused Deposition Modeling (FDM) technology. It is a layer additive manufacturing process that uses production-grade thermoplastic filaments to produce both prototypes and end-use parts. This technology is known to accurately produce feature details and has an excellent strength to weight ratio. Thin thread-like spools of thermoplastic and support materials are used to create each cross-section of the part.

Objects are made by heating the thermoplastic filament to its melting point and extruding the thermoplastic layer by layer hence it is a layer by layer process. The printer can extrude a different material that will serve as support material during the printing process. This support material can, later on, be removed after the completion of the printing process by either removing them physically or by dissolving them in a suitable solvent.

Some printing specifications used for printing the parts in this model are:

- 1) Material used: Polylactic Acid (PLA)
- 2) Infill %: 20%
- 3) Layer Height: 200 micron
- 4) Supports: Required
- 5) Printing speed: 60mm/sec
- 6) Shell Thickness: 1.2mm
- 7) Wall count: 3

4.2 Process description

When the setup is started, the robotic arm will move to its HOME position automatically. After a delay of 5seconds (adjustable through the code) the operator can turn the potentiometers depending upon which link needs to be moved. After the links are moved by the desired angles or after "Jogging" the arm to a particular position the operator will press the SAVE button to save the current angular positions of all the servo motors.

The correct way to carry out the pick and place operation properly is to divide the operation into the following processes and save the position of the robot arm at the end of each process as discussed below:

1. Approach,
2. Contact
3. Pick
4. Move to destination
5. Drop/Release

After saving all the positions, to run the robotic arm automatically, the operator should press the RUN button. The robot will continue to run in cycles of the above mentioned processes until the PAUSE button is pressed at any stage the operator desires to. During the automatic operating stage of the robot, only one servo moves at a time and the speed of the servo motor can be adjusted by the operator through the code. This ensures the control of the operator over the robotic arm in case something wrong occurs. It is ensured that during the operating condition of the arm no interference is caused in the functioning of the robot if the potentiometers are used in between or the save button is pressed in between. If something goes wrong or the robot is not functioning properly the operator can press the RESET button to reset the entire system and start over again. At any stage, the operator can check the status of the robot through the display (the laptop/PC/external display) and can check how many positions have been saved by the operator.

IV. DESIGN AND EXPERIMENTAL SETUP

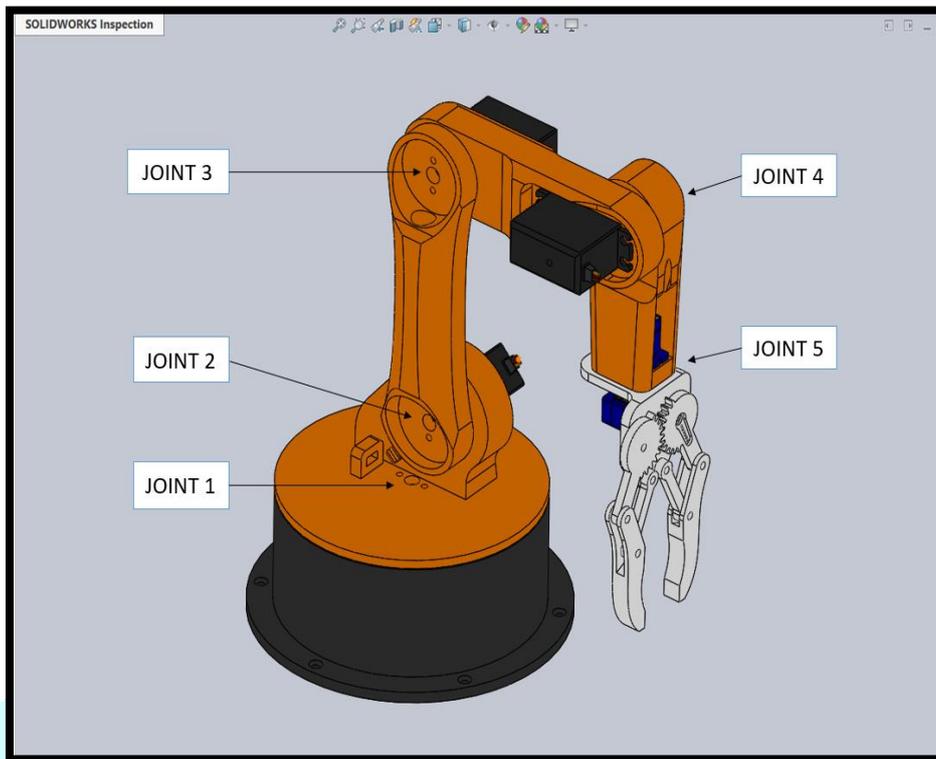


Fig.8. CAD model of the proposed system

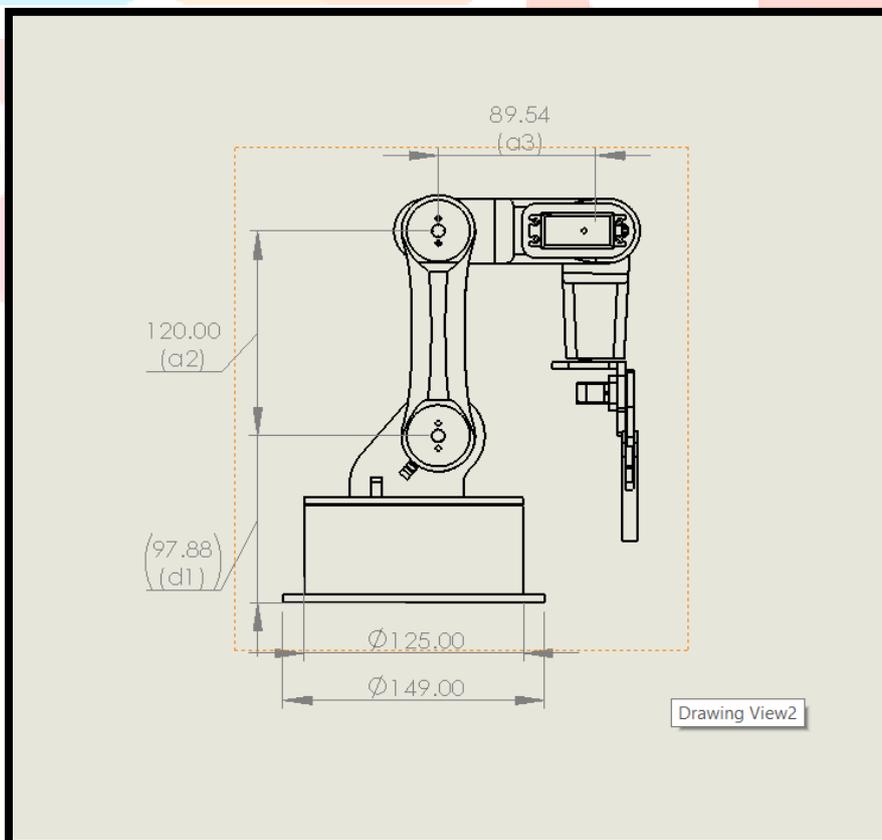


Fig.9. Side View
(All measurements are in mm)

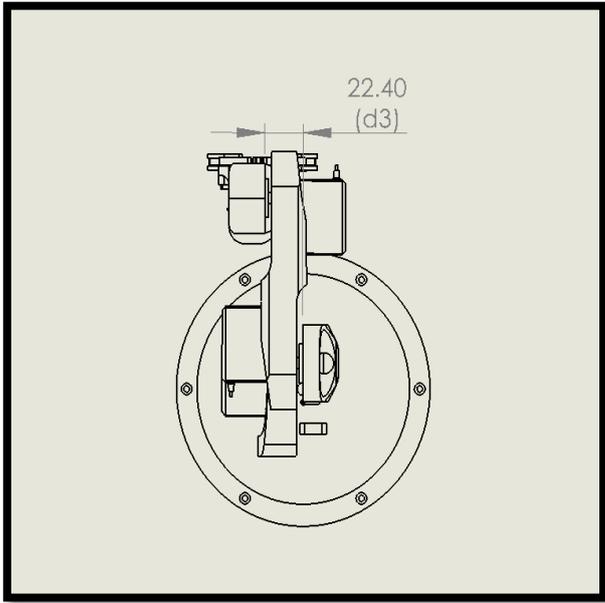


Fig.10. Top View

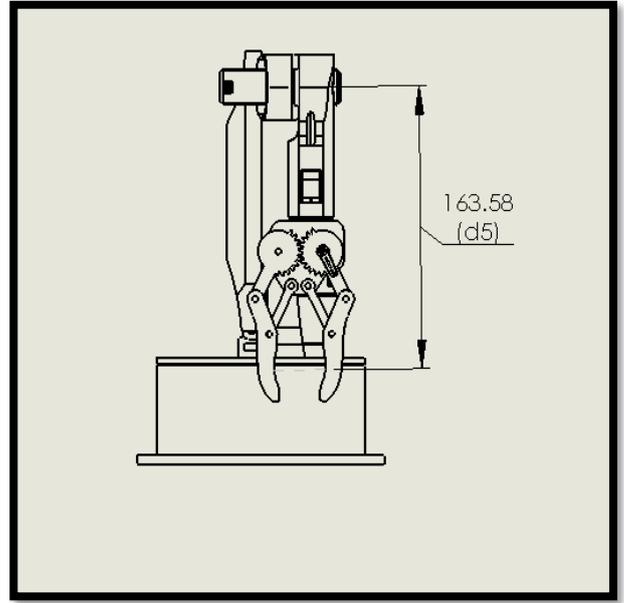


Fig.11. Front View

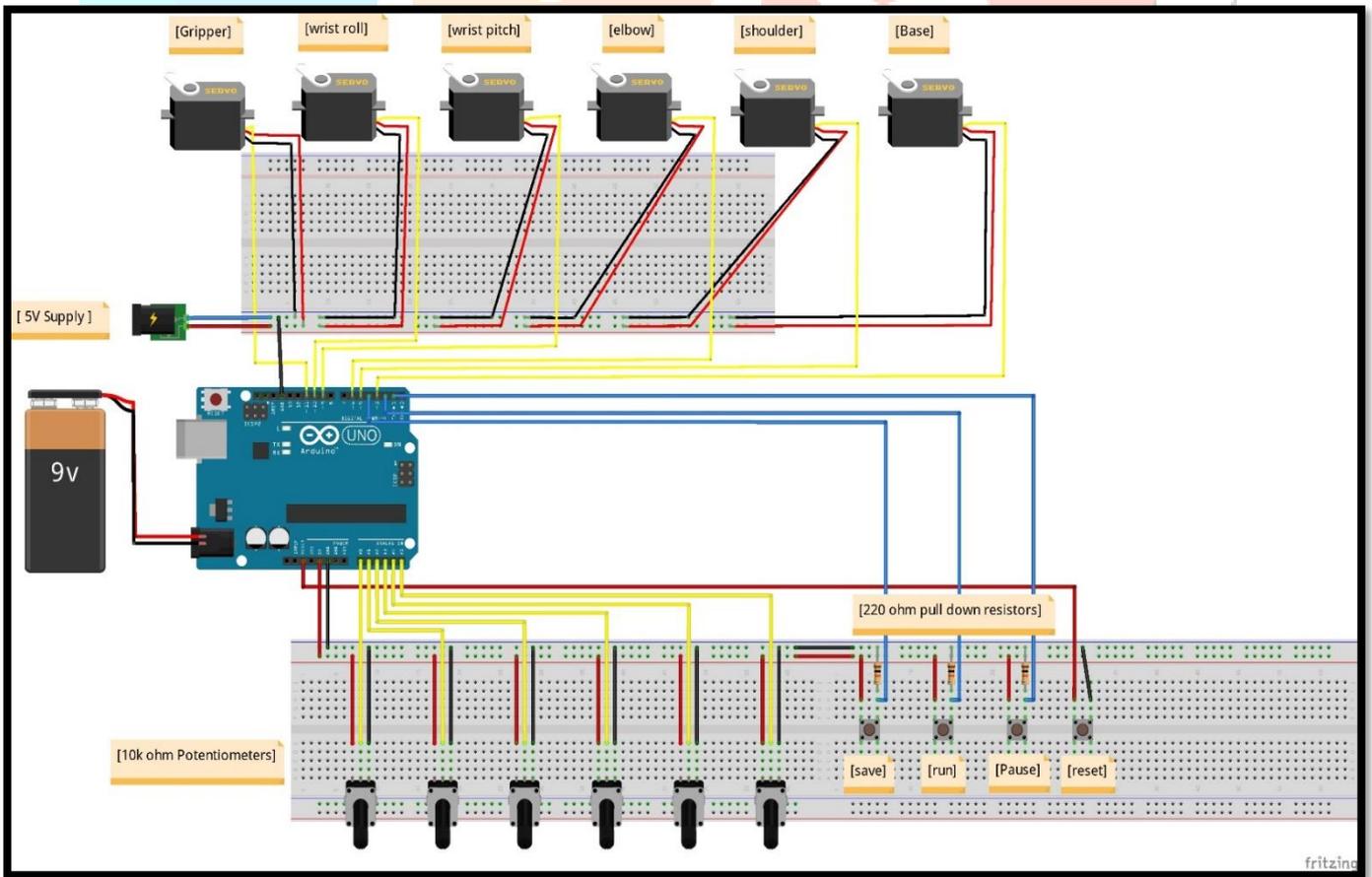


Fig.12. Connections



Fig.13. Working Model

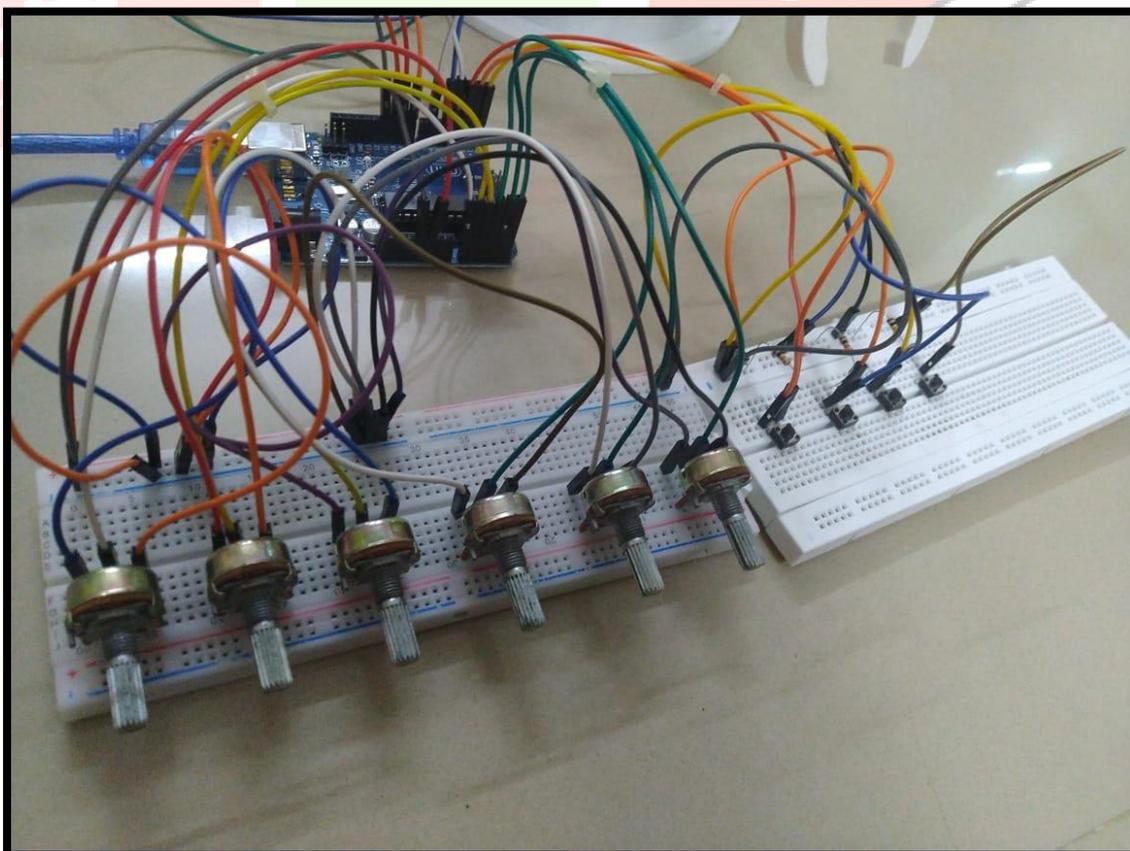


Fig.14. Proposed Teach Pendant

V. ANALYSIS

The kinematic analysis of a robotic arm can be done by two approaches i.e. Forward and Inverse Kinematics. Forward kinematics will determine where the End Effector (EE) will lie in 3 dimensional space and what will be its orientation with respect to the base co-ordinate frame when each joint of the robotic arm is moved by specific angles i.e. the values of Joint Angles are known and co-ordinates and orientation of EE are derived. Forward kinematics is therefore defined as the transformation from Joint space to Cartesian space. Since, in the program, we have to input the angles by which the shafts of the servo motors have to rotate i.e. Joint Angle (θ_i) is known therefore, the analysis has been done by Forward Kinematics to analyze the movement of the arm when it is moved from one point to another in space. [9][3]

• Denavit-Hartenberg (D-H) Parameters

In this study, the standard D-H convention and methodology has been used to derive the parameters and kinematics for the proposed model. Denavit-Hartenberg algorithm helps to find the position and orientation of end-effector with respect to the base as shown in Table1.

Frame	Joint Angle (θ_i) (degrees)	Link Offset(d_i) (mm)	Twist Angle (α_i) (degrees)	Link Length (a_i) (mm)
1	θ_1	97.88	90	0
2	θ_2	0	0	120
3	θ_3	-22.4	180	89.5
4	θ_4	0	90	0
5	θ_5	163.58	0	0

Table.1. D-H parameters

Fig.15. shows the Kinematic Diagram of the robotic arm made according to the D-H parameters. **It is important to note that the frames 3 and 4 coincide because the Z axis of the joints 4 and 5 intersect each other and the link offset is negligible. Hence, the link offset is taken as 0.**

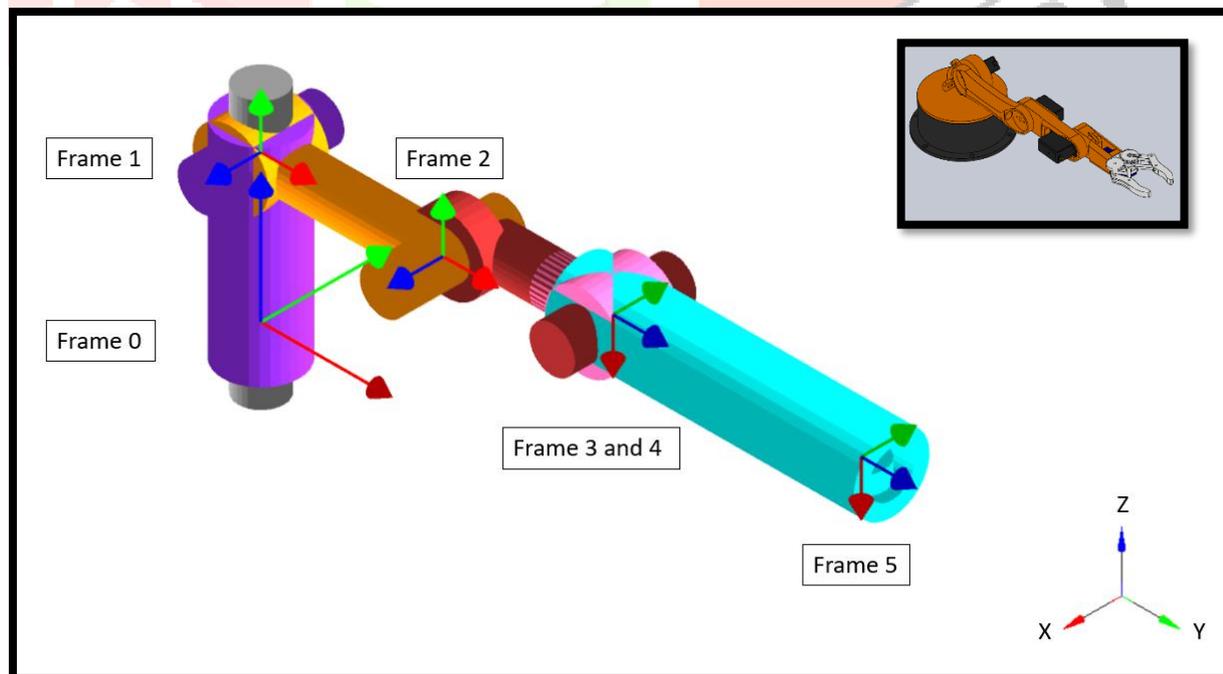


Fig.15. Kinematic Diagram made using RoboAnalyzer

- **Forward Kinematics**

Based on the D-H parameters, the Homogeneous Transformation Matrix (HTM) for transformation of the Frame $i-1$ to i is given by:

$${}^{i-1}T_i = ROT(Z, \theta_i) TRANS(Z, d_i) ROT(X, \alpha_i) TRANS(X, a_i)$$

Where:

$$ROT(Z, \theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$ROT(X, \alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$TRANS(Z, d) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$TRANS(X, a) = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

Substituting the matrices (1), (2), (3) and (4) in the above equation we get:

$${}^{i-1}T_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

Therefore, the HTM of the End Effector frame w.r.t Base frame for the proposed robotic arm will be given by:

$${}^0_5T = {}^0_1T {}^1_2T {}^2_3T {}^3_4T {}^4_5T$$

$${}^0_1T = ROT(Z, \theta_1) TRANS(Z, 97.88) ROT(X, 90) TRANS(X, 0)$$

$$= \begin{bmatrix} \cos\theta_1 & 0 & \sin\theta_1 & 0 \\ \sin\theta_1 & 0 & -\cos\theta_1 & 0 \\ 0 & 1 & 0 & 97.88 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

$${}^1_2T = ROT(Z, \theta_2) TRANS(Z, 0) ROT(X, 0) TRANS(X, 120)$$

$$= \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & 120\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & 120\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

$${}^2_3T = ROT(Z, \theta_3) TRANS(Z, -22.4) ROT(X, 180) TRANS(X, 89.5)$$

$$= \begin{bmatrix} \cos\theta_3 & \sin\theta_3 & 0 & 89.5\cos\theta_3 \\ \sin\theta_3 & -\cos\theta_3 & 0 & 89.5\sin\theta_3 \\ 0 & 0 & -1 & -22.4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

$${}^3_4T = ROT(Z, \theta_4) TRANS(Z, 0) ROT(X, 90) TRANS(X, 0)$$

$$= \begin{bmatrix} \cos\theta_4 & 0 & \sin\theta_4 & 0 \\ \sin\theta_4 & 0 & -\cos\theta_4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

$${}^4_5T = ROT(Z, \theta_5) TRANS(Z, 163.58) ROT(X, 0) TRANS(X, 0)$$

$$= \begin{bmatrix} \cos\theta_5 & -\sin\theta_5 & 0 & 0 \\ \sin\theta_5 & \cos\theta_5 & 0 & 0 \\ 0 & 0 & 1 & 163.58 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

Multiplying the matrices (6), (7), (8), (9), (10) we get:

$${}^0_5T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & l_x \\ r_{21} & r_{22} & r_{23} & l_y \\ r_{31} & r_{32} & r_{33} & l_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (11)$$

Where:

$$r_{11} = -\sin(\theta_5) \sin(\theta_1) + \cos(\theta_5) \cos(\theta_1) \cos(\theta_2 + \theta_3 - \theta_4) \quad (a)$$

$$r_{12} = -\sin(\theta_1) \cos(\theta_5) - \sin(\theta_5) \cos(\theta_1) \cos(\theta_2 + \theta_3 - \theta_4) \quad (b)$$

$$r_{13} = -\sin(\theta_2 + \theta_3 - \theta_4) \cos(\theta_1) \quad (c)$$

$$r_{21} = \sin(\theta_1) \cos(\theta_5) \cos(\theta_2 + \theta_3 - \theta_4) + \sin(\theta_5) \cos(\theta_1) \quad (d)$$

$$r_{22} = -\sin(\theta_5) \sin(\theta_1) \cos(\theta_2 + \theta_3 - \theta_4) + \cos(\theta_5) \cos(\theta_1) \quad (e)$$

$$r_{23} = -\sin(\theta_1) \sin(\theta_2 + \theta_3 - \theta_4) \quad (f)$$

$$r_{31} = \sin(\theta_2 + \theta_3 - \theta_4) \cos(\theta_5) \quad (g)$$

$$r_{32} = -\sin(\theta_5) \sin(\theta_2 + \theta_3 - \theta_4) \quad (h)$$

$$r_{33} = \cos(\theta_2 + \theta_3 - \theta_4) \quad (i)$$

$$l_x = -22.4 \sin(\theta_1) - 89.5 \sin(\theta_2) \sin(\theta_3) \cos(\theta_1) - 163.58 \sin(\theta_2 + \theta_3 - \theta_4) \cos(\theta_1) + 89.5 \cos(\theta_1) \cos(\theta_2) \cos(\theta_3) + 120 \cos(\theta_1) \cos(\theta_2) \quad (j)$$

$$l_y = -89.5 \sin(\theta_1) \sin(\theta_2) \sin(\theta_3) - 163.58 \sin(\theta_1) \sin(\theta_2 + \theta_3 - \theta_4) + 89.5 \sin(\theta_1) \cos(\theta_2) \cos(\theta_3) + 120 \sin(\theta_1) \cos(\theta_2) + 22.4 \cos(\theta_1) \quad (k)$$

$$l_z = 120 \sin(\theta_2) + 89.5 \sin(\theta_2 + \theta_3) + 163.58 \cos(\theta_2 + \theta_3 - \theta_4) + 97.88 \quad (l)$$

Now the robotic arm will be moved between two points/positions. Let's say the initial position of the robot, as shown in Fig.19, is defined by the values of the Joint Angles (θ_i) as:

Position 1

- $\theta_1 = 0$
- $\theta_2 = 0$
- $\theta_3 = 0$
- $\theta_4 = 90$
- $\theta_5 = 0$

Substituting the values of Joint angles in equations (a), (b),..... (l), the HTM becomes:

$${}^0_5T = \begin{bmatrix} 0 & 0 & 1 & 373.08 \\ 0 & 1 & 0 & 22.4 \\ -1 & 0 & 0 & 97.88 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The co-ordinates and orientation of the end effector (frame 5) w.r.t base (frame 0) can be defined as:

- $x = l_x = 373.08$
- $y = l_y = 22.4$
- $z = l_z = 97.88$
- $\theta_x = \tan^{-1} \left(\frac{r_{32}}{r_{33}} \right) = 0$
- $\theta_y = \tan^{-1} \left(\frac{-r_{31}}{\sqrt{r_{11}^2 + r_{21}^2}} \right) = 90$ (frame 0 is rotated by 90° about its y axis to achieve frame 5)
- $\theta_z = \tan^{-1} \left(\frac{r_{21}}{r_{11}} \right) = 0$

Now let's say the robotic arm is moved to a new position/point for which the values of Joint Angles (θ_i) are:

Position 2

- $\theta_1 = 0$
- $\theta_2 = 90$
- $\theta_3 = 0$
- $\theta_4 = 150$
- $\theta_5 = 0$

Therefore, substituting the above Joint Angle values in equations (a), (b),..... (l), HTM becomes:

$${}^0_5T = \begin{bmatrix} 500 & 0 & 866.025 & 141.664 \\ 0 & 1 & 0 & 22.4 \\ -866.025 & 0 & 500 & 389.17 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The co-ordinates and orientation of the end effector (frame 5) w.r.t base (frame 0) can be defined as:

- $x = l_x = 141.664$
- $y = l_y = 22.4$
- $z = l_z = 389.17$
- $\theta_x = \tan^{-1} \left(\frac{r_{32}}{r_{33}} \right) = 0$

- $\theta_y = \tan^{-1} \left(\frac{-r_{31}}{\sqrt{r_{11}^2 + r_{21}^2}} \right) = 59.99$ (frame 0 is rotated by 59.99° about its y axis to achieve frame 5)
- $\theta_z = \tan^{-1} \left(\frac{r_{21}}{r_{11}} \right) = 0$

VI. RESULTS

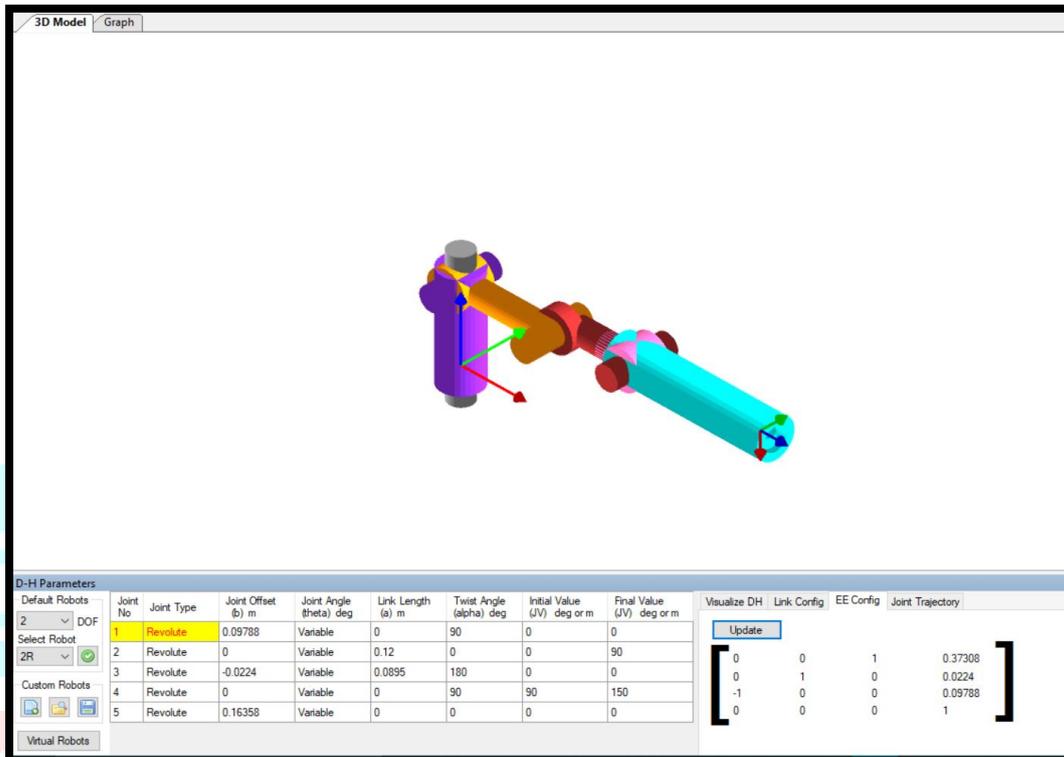


Fig.16. Position 1

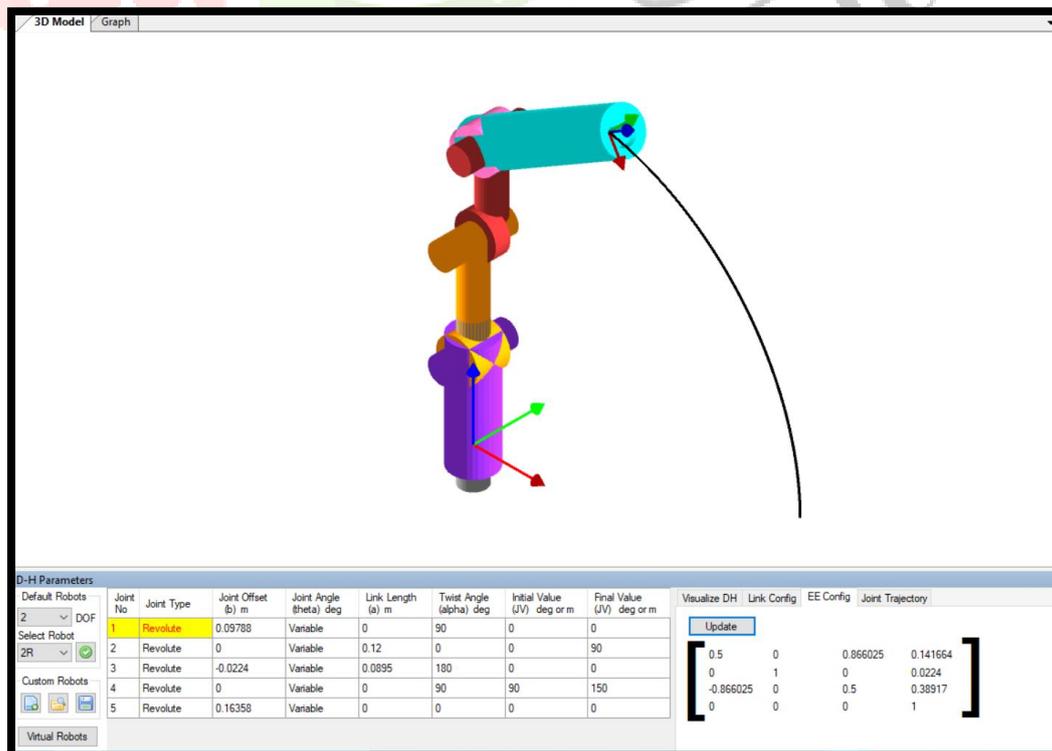


Fig.17. Position 2

In order to verify the HTM obtained for position 1 and position 2 of the robotic arm, a similar skeletal model has been made in the RoboAnalyzer software and the arm is moved from Position1 to Position 2 as shown in Fig.16 and 17.

The above figures also show the HTM for the EE in Position 1 and Position 2. Comparing the HTM got from the RoboAnalyzer software and the one derived in the above sections using Forward Kinematics, we can say that the analysis was accurate and successful. Since only the joints 2 and 4 were in motion, the graphs below show the variation of joint value, joint velocity and joint acceleration of joints 2 and 4 while moving from position 1 to position 2.

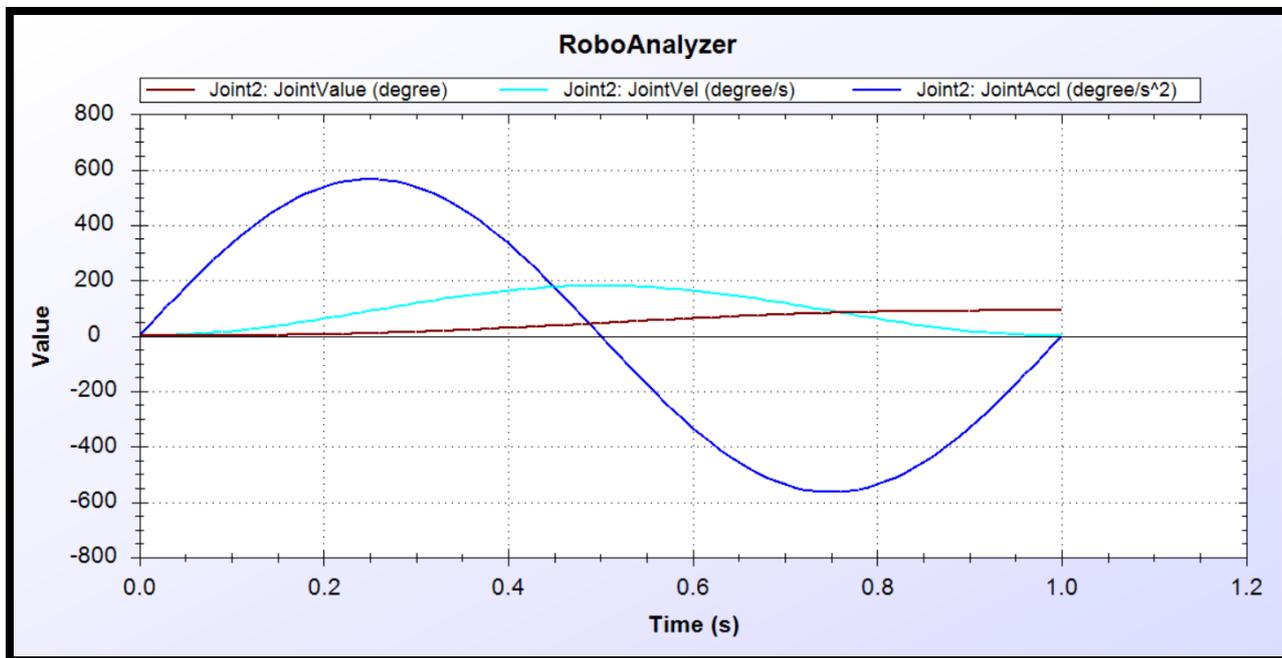


Fig.18. Joint 2

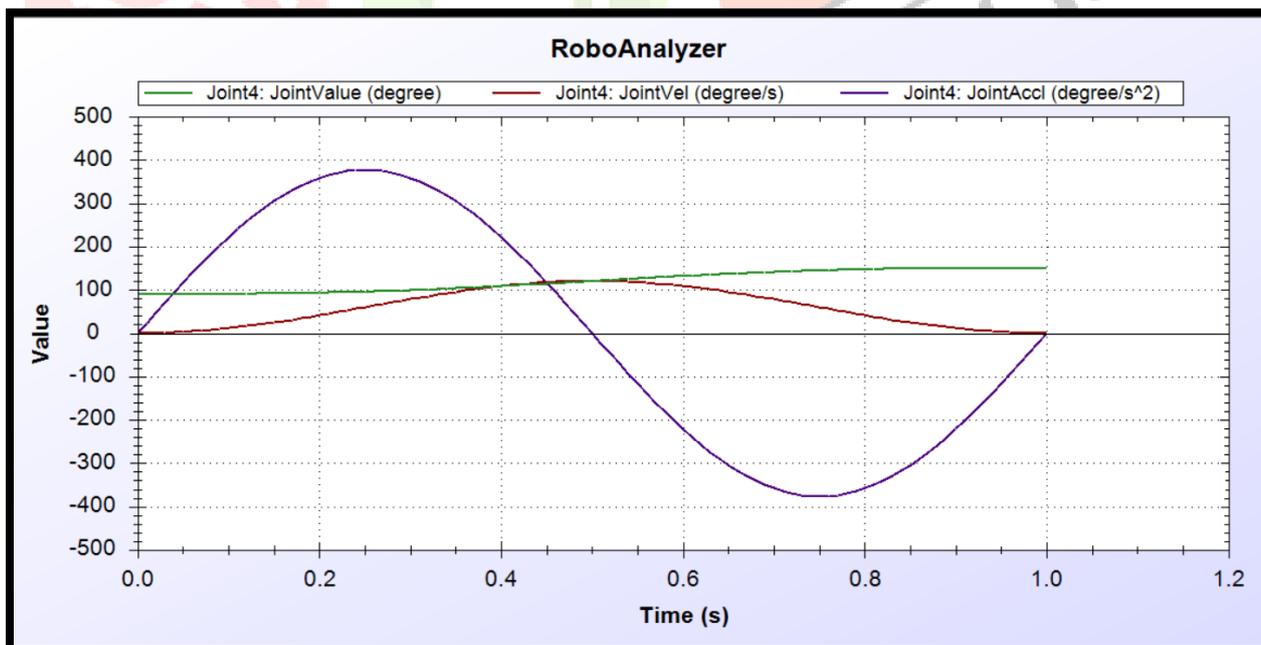


Fig.19. Joint 4

VII. CONCLUSION

The aim of this study was to build a working robotic arm model with a similar approach used by professionals to build actual robotic arms. This paper covered the basic methodology used for designing and developing the robotic arms which are used in industries for production and manufacturing. The paper has covered the various teaching methods employed by professionals in the industries for teaching and controlling robots. By analyzing the various Robot teaching methods, we have developed a teach pendant that can conveniently control the 5 degrees of freedoms Robotic Arm developed using 3D printing which helps to understand the fundamentals of Robot Jogging. By understanding the various types of grippers and end effectors used in this industry, we were able to choose and develop the gripper which was most suitable for the pick and place operation. A complete analytical solution has been formulated using Forward Kinematics for the proposed model and the mathematical model is prepared and solved for different position and orientation of the end effectors. The Homogeneous Transformation Matrices obtained were verified using the RoboAnalyzer software and the results of the analysis were satisfactory.

VIII. REFERENCES

1. Kadirimangalam Jahnavi, Sivraj P, Teaching and Learning Robotic Arm Model, International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICT), 2017, July, Kannur, India.
2. Ms. Rasika Yenorkar, Prof. Dr.U. M. Chaskar, GUI Based Pick and Place Robotic Arm for Multipurpose Industrial Applications, Proceedings of the Second International Conference on Intelligent Computing and Control Systems (ICICCS), 2018, June, Madurai, India.
3. Daniel Constantin, Marin Lupoae, Cătălin Baciuc, Dan-Ilie Buliga, Forward Kinematic Analysis of an Industrial Robot, NEW DEVELOPMENTS in MECHANICS and MECHANICAL ENGINEERING, 2015, ISBN: 978-1-61804-288-0, p. 90-95.
4. Christos Tolis, George F. Fragulis, An experimental mechatronic design and control of a 5 DOF Robotic arm for identification and sorting of different sized objects, Western Macedonia Univ. of Applied Sciences, Kozani, Hellas, November, 2017.
5. Nimisha Limaye, Siddharth Verma, 5 Degrees of Freedom Robotic Arm controlled using PS/2 Mouse, International Journal of Engineering and Technical Research (IJETR), 2014, Vol. 2, Issue-11, p. 201-204.
6. Avi Kumar R, Papiya Mandal, Preethi K S, Ranjitha N, Nishkala U, Haptic Based Tele-picking Robotic Arm, International Journal of Engineering Research & Technology (IJERT), 2019, Vol. 7, Issue 10, p. 1-4.
7. Abhiraj Bhalerao, Kunal Chopade, Prasad, Jitendra Gaikwad, Pick and Place Robotic ARM using PLC, International Journal of Engineering Research & Technology (IJERT), 2019, Vol. 8, Issue 08, p. 667-670.
8. Miss.Sayali V. Vanjari, Miss.Poonam Ghewande, Miss.Kalyani Deshmukh, Miss. Rasika Sawale, ROBOTIC ARM, International Journal of Creative Research Thoughts (IJCRT), 2018, Vol. 6, Issue 1, p. 1142-1162.
9. Vivek Deshpande, P. M. George, KINEMATIC MODELLING AND ANALYSIS OF 5 DOF ROBOTIC ARM, International Journal of Robotics Research and Development (IJRRD), 2014, Vol. 4, Issue 2, p. 17-24.
10. Harish K, Megha D, Shuklambari M, Amit K, Chaitanya K Jambotkar, Pick and Place Robotic Arm Using Arduino, International Journal of Science, Engineering and Technology Research (IJSETR), 2017, Vol. 6, Issue 12, p. 1568-1573.
11. Hooman Lee, Joongbae Kim, A survey on Robot Teaching: Categorization and Brief Review, Applied Mechanics and Materials, Vol. 330, 2013, doi:10.4028/www.scientific.net/AMM.330.648, p. 648-656.
12. Samir Raval, Bhavesh Patel, A Review on Grasping Principle and Robotic Grippers, International journal of Engineering development and research IJEDR, 2016, Vol. 4, Issue 1, p. 483-490.
13. B.O. Omijeh, R. Uhumwangho, Design Analysis of a Remote Controlled "Pick and Place" Robotic Vehicle, International Journal of Engineering Research and Development (IJERD), 2014, Vol. 10, Issue 5, p. 57-68.
14. <https://lastminuteengineers.com/servo-motor-arduino-tutorial/>