MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR

(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)

NAAC Accredited with A++ Grade

Project Report

on

Hand Gesture Controlled Robotic Arm

Submitted By:

Ayush Carpenter

0901AI211014

Chandramani Shukla

0901AI211019

Faculty Mentor:

Dr. Anshika Srivastava

Assistant Professor

CENTRE FOR ARTIFICIAL INTELLIGENCE

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE

GWALIOR - 474005 (MP) est. 1957

JULY-DEC. 2023

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR

(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)

NAAC Accredited with A++ Grade

CERTIFICATE

This is certified that **Ayush Carpenter** (0901AI211014) & **Chandramani Shukla** (0901AI211019) has submitted the project report titled **Gesture Controlled Robot System** under the mentorship of **Dr. Anshika Srivastava**, in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in **IT(AIR)** from Madhav Institute of Technology and Science, Gwalior.

Anshika 23/11/2023

Dr. Anshika Srivastava
Faculty Mentor

Centre for Artificial Intelligence

23/11/23

Dr. R. R. Singh
Coordinator
Centre for Artificial Intelligence

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR

(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)

NAAC Accredited with A++ Grade

DECLARATION

I hereby declare that the work being presented in this project report, for the partial fulfilment of requirement for the award of the degree of Bachelor of Technology in **IT(AIR)** at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of **Dr. Anshika Srivastava**, Center of Artificial Intelligence.

I declare that I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Ayush Carpenter
0901AI211014
Chandramani Shukla
0901AI211019
III Year
Centre for Artificial Intelligence

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE GWALIOR

(A Govt. Aided UGC Autonomous Institute Affiliated to RGPV, Bhopal)

NAAC Accredited with A++ Grade

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary/interdisciplinary project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on the AICTE Model Curriculum 2018), approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Centre for Artificial Intelligence**, for allowing me to explore this project. I humbly thank **Dr. R. R. Singh**, Coordinator, Centre for Artificial Intelligence, for his continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Dr. Anshika Srivastava**, Center of Artificial Intelligence, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

Ayush Carpenter
0901AI211014
Chandramani Shukla
0901AI211019
III Year
Centre for Artificial Intelligence

ABSTRACT

This A robotic system focused on the construction, design, and operation of a 4-degree-of-freedom robotic arm. This system encompasses the design, manufacture, and application of a versatile robotic arm with an emphasis on modularity, flexibility, and sensor-driven functionality. The primary objective of this robotic system is to enable intuitive control through human gestures.

The robotic arm integrates sensors to interpret hand movements captured by a gesture device worn by the user. Through wireless communication, the sensor records specific hand movements, translating them into precise control commands for the robotic arm. This intuitive interaction method allows users to control the arm's forward, backward, leftward, and rightward motions effortlessly.

The mechanism governing the movement of the robotic arm involves precise control of its four degrees of freedom. Each degree of freedom corresponds to a specific joint or segment, enabling the arm to mimic human-like motion. The system allows for the manipulation of the arm's joints, replicating the functionality of the human arm with modularity and precision.

This robotic arm, guided by hand gestures, offers enhanced control and maneuverability in navigating its environment. The integration of sensor-based control provides a seamless and responsive interaction, enabling precise and intricate movements in a wide range of applications. The modularity of the arm design ensures adaptability and ease of integration into various tasks and environments.

Overall, this system aims to revolutionize human-robot interaction by enabling intuitive and precise control of a versatile 4-degree-of-freedom robotic arm, expanding its utility across diverse fields and applications.

Keywords : Robotic system, 4-DOF arm, design, manufacture, modularity, flexibility, sensor-driven control, gesture device, wireless interaction, precise movement, human-like motion, maneuverability, sensor-based functionality.

सार:
यह एक रोबोटिक प्रणाली 4-डिग्री-ऑफ़-फ़्रीडम रोबोटिक भुजा के निर्माण, डिज़ाइन और संचालन पर केंद्रित है। यह प्रणाली मॉड्यूलरिटी, लचीलेपन और सेंसर-संचालित कार्यक्षमता पर जोर देने के साथ एक बहुमुखी रोबोटिक बांह के डिजाइन, निर्माण और अनुप्रयोग को शामिल करती है। इस रोबोटिक प्रणाली का प्राथमिक उद्देश्य मानवीय इशारों के माध्यम से सहज नियंत्रण को सक्षम करना है।

रोबोटिक भुजा की गति को नियंत्रित करने वाले तंत्र में इसकी स्वतंत्रता की चार डिग्री का सटीक नियंत्रण शामिल है। स्वतंत्रता की प्रत्येक डिग्री एक विशिष्ट जोड़ या खंड से मेल खाती है, जो हाथ को मानव जैसी गति की नकल करने में सक्षम बनाती है। सिस्टम हाथ के जोड़ों में हेरफेर की अनुमति देता है, मानव हाथ की कार्यक्षमता को मॉड्यूलरिटी और सटीकता के साथ दोहराता है।

हाथ के इशारों से निर्देशित यह रोबोटिक भुजा, अपने वातावरण में नेविगेट करने में बेहतर नियंत्रण और गतिशीलता प्रदान करती है। सेंसर-आधारित नियंत्रण का एकीकरण एक सहज और प्रतिक्रियाशील इंटरैक्शन प्रदान करता है, जो अनुप्रयोगों की एक विस्तृत श्रृंखला में सटीक और जटिल गतिविधियों को सक्षम बनाता है। आर्म डिज़ाइन की मॉड्यूलरिटी विभिन्न कार्यों और वातावरणों में अनुकूलनशीलता और एकीकरण में आसानी सुनिश्चित करती है।

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1: Project Overview

1.1 Introduction:

The Robotics is a current emerging technology in the field of science. As robots move away from industrial settings and closer into our lives, the question arises how to interact with these robots. A large part of interaction occur through hand gestures [1]. In fact Interpretation of human gestures by a computer is used for human-machine interaction [2]. The main purpose of gesture recognition research is to identify a particular human gesture and convey information to the user pertaining to individual gesture. Overall aim is to make the machine understand human body language [3], thereby bridging the gap between machine and human. Hand gesture recognition can be used to enhance human–computer interaction without depending on traditional input devices such as keyboard and mouse.

Robotic hand can be controlled remotely by hand gestures. Research is being carried out in this area for a long time. Hand gestures are extensively used for telerobotic control and application [4].Robotic systems can be controlled naturally and intuitively with telerobotic technologies [5]. It is used in agriculture industry as there is reduction in the traditional human harvesters who want to take up other jobs which pays good income. (As in one part of the world, the robot revolution in agriculture sector I already in progress, in India it is a difficult story). According to the report in huffintonpost.in [6], 55% of total population in India depends on farming whereas in US its only 2% because of heavy mechanization of agriculture. Several approaches have been developed for sensing hand movements and controlling robotic hand. The robotic hand, operated and controlled wirelessly with the help of hand gesture transmits signals to the robot through an auto device fixed on the gloves which is to be put on hands rather than controlling it manually through a conventional remote. The five finger hand combined with its integrated wrist and forearm has fourteen independent degree of freedom [7].With growing interest in using robots to perform above tasks, design of anthropomorphic robotic hands, the end effectors of robots, with desired dexterity and output power has become the focus of recent robotic research [8].Glove based technique is a well- known means of recognizing hand gestures. The Robot moves and acts in the manner depending on the gestures made by the fingers and hand from a distance. The robot moves in up, down, left or right directions and picks up objects from one place and keeps at another desired place as directed by the movements of fingers and hand. It is a Type –C Robot, programmable, Servo controlled with either point to point or continuous trajections.

1.2 Scope and Objectives:

The objectives of this project are shown as following:

1. Building a robotic arm of 4 degree of freedom.
2. To place into practice gesture control on the robotic arm.

3. To urge conversant in MPU 6050 sensor technology
4. To urge conversant in flex sensor

Gesture control method is going to be adopted during this development. With this gesture controlled robotic arm, the bomb disposal operation is going to be higher efficiency because the robot is often operating in faster and more intuitive way and no training is required.

When we use remote control with buttons or a joystick to control a robot the actions, we create are not that precise or smooth. Using hand gestures, we can give exact commands and replicate our hand movement into robotic actions. Our Objective in the proposed system is to make the system cheap and simple, so that it would be mass produced and can be used for various purposes. It will help to reduce human effort in controlling robotic systems using remotes and thereby providing a better and maximum efficiency at the output.

1.3 Project Features:

The Project is designed with a modular approach, allowing for ease of assembly, maintenance, and scalability. It suitable for deployment and also cost effective.

1. **Gesture-Based Control:** Control the robotic arm's movements using hand gestures detected by sensors, allowing intuitive interaction.
2. **4 Degrees of Freedom**: Enables precise control of four distinct joints or segments, mimicking human arm motion for versatile manipulation.
3. **Modularity and Flexibility:** Designed with interchangeable components or modules for adaptability to various tasks or environments.
4. **Wireless Communication:** Utilizes wireless technology for seamless communication between the gesture device and the robotic arm.
5. **Sensor Integration:** Integration of sensors for accurate movement tracking and precise control of the arm's degrees of freedom.
6. **Human-Robot Interaction:** Aimed at enhancing user experience through intuitive and responsive interaction with the robotic arm.
7. **Customizable Movement Sequences:** Allows users to program and execute custom movement sequences for specific tasks or applications.
8. **Maneuverability:** Offers precise and agile movement capabilities, enabling the arm to navigate and manipulate objects with precision.
9. **Adaptability:** Designed for easy integration into different setups or workflows, offering adaptability to various use cases.

10. **User-Friendly Interface:** Provides a user-friendly interface for setting up, calibrating, and controlling the robotic arm.
11. **Educational Value:** Offers a learning platform for robotics enthusiasts, students, or researchers to explore gesture-based control and robotics.
12. **Safety Features:** Includes safety measures such as emergency stop functionality or collision detection to prevent accidents during operation.
13. These features aim to highlight the capabilities and functionalities of the hand gesture-controlled 4-degree-of-freedom robotic arm manipulator, emphasizing its versatility, precision, and intuitive control mechanisms.

1.4 Feasibility:

The feasibility of a hand gesture-controlled 4-degree-of-freedom (4-DOF) robotic arm project can be assessed across various aspects:

Technical Feasibility:

- Technology Availability: Check the availability of sensors, actuators (servo motors or other mechanisms), and wireless communication modules suitable for gesture control.
- Integration Complexity: Assess the technical challenges in integrating sensor data with robotic arm movement commands accurately and responsively.
- Hardware and Software Compatibility: Ensure compatibility between selected hardware components and the software platform for programming the arm.

Financial Feasibility:

- Cost Analysis: Evaluate the cost of required components, including sensors, actuators, microcontrollers, and other associated hardware, against the project budget.
- Resource Availability: Consider expenses for development tools, prototyping materials, and any potential outsourcing costs.

Operational Feasibility:

- User Interaction: Assess the ease of use and intuitiveness of the hand gesture control system for users.
- Robotic Arm Capabilities: Determine if the 4-DOF design meets the intended requirements for maneuverability, precision, and task versatility.
- Adaptability: Evaluate the adaptability of the robotic arm to different environments or tasks.

Time Feasibility:

- Development Time: Estimate the time required for hardware assembly, sensor integration, programming, testing, and troubleshooting.
- Project Deadlines: Ensure alignment between project deadlines and the realistic development timeline.
- Legal and Ethical Feasibility:
- Regulatory Compliance: Consider any legal regulations or standards that the project might need to adhere to.
- Ethical Considerations: Address ethical concerns related to safety, privacy, and any implications of the technology's use.

Resource Availability:

- Skills and Expertise: Evaluate the team's skills or the need for hiring expertise in robotics, sensor integration, programming, and hardware assembly.
- Tools and Equipment: Ensure access to necessary tools and equipment required for assembly, testing, and validation.

By carefully examining these aspects, addressing potential challenges, and ensuring adequate resources, a feasibility study can help determine the viability and potential success of a hand gesture-controlled 4-DOF robotic arm project. It's essential to conduct thorough research and planning before committing to the project.

1.5 System Requirements:

1.5.1 Hardware Requirements:

1.5.1.1 Aurduino UNO Microcontroller:

Arduino UNO is a microcontroller board based on the **ATmega328P**. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The controller which is used in this project is ARDUINO UNO. It is a microcontroller board which contain ATmega328p processor [4]. The pin out section as follows

1) 14digitalI/Opins.(outofwhich16canbeusedasPWMoutputs)
2) 6 analoginputs.

Fig 1.1 Aurduino UNO

Summary :

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Analog Input Pins	6
Digital I/O Pins	14
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Clock Speed	16 MHz

Fig 1.2 Details of Aurduino

1.5.1.2 Bread Board:

A breadboard consists of plastic block holding a matrix of electrical sockets of a size suitable for gripping thin connecting wire, component wires or the pins of transistors and integrated circuits (ICs). The sockets are connected inside the board, usually in rows of five sockets.

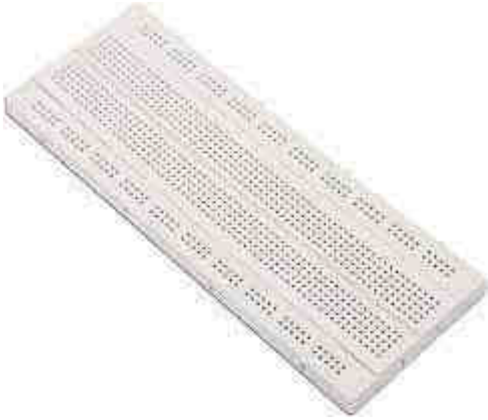


Fig 1.3 Breadboard

1.5.1.3 Jumper Wires:

Jumper cables is a smaller and more bendable corrugated cable which is used to connect antennas and other components to network cabling.

Fig 1.4 Jumper Wires

1.5.1.4 Servo Motor:

A **servo motor** is a type of motor that can rotate with great precision. Normally this type of motor consists of a control circuit that provides feedback on the current position of the motor shaft, this feedback allows the servo motors to rotate with great precision.

Fig 1.5 Servo Motor

These servo have 3 Pins –Vcc - Red wire

GND - Brown

PWM - yellow /Orange

Vcc is used to connect the servo with the 5V supply or greater (depends on the type of servo). It is required to make the servo motor work for a particular task.

GND is the ground pin.

PWM stands for pulse width modulation. This signal through this pin have a total symbol width of 20ms (50Hz).

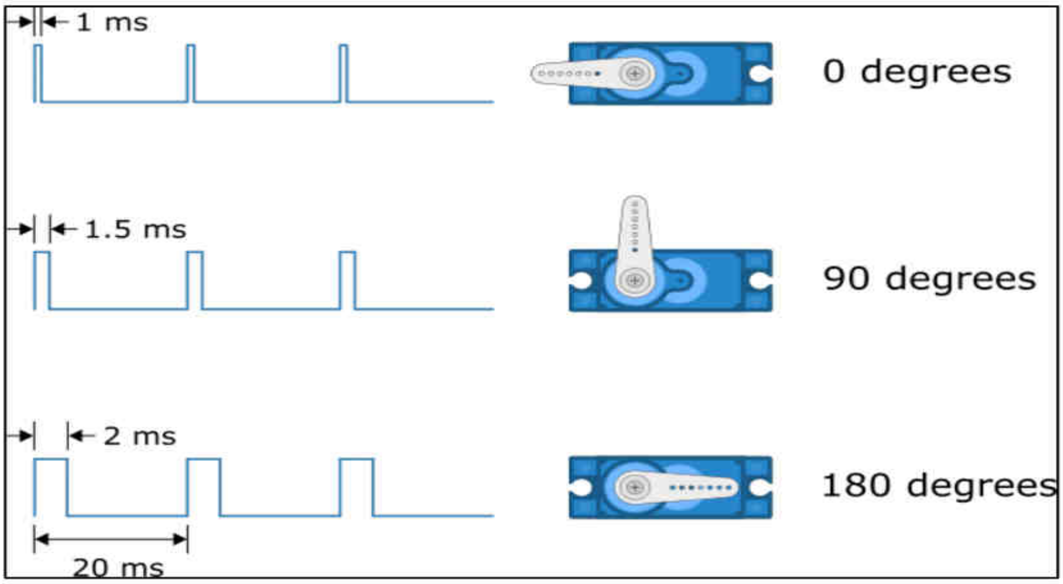
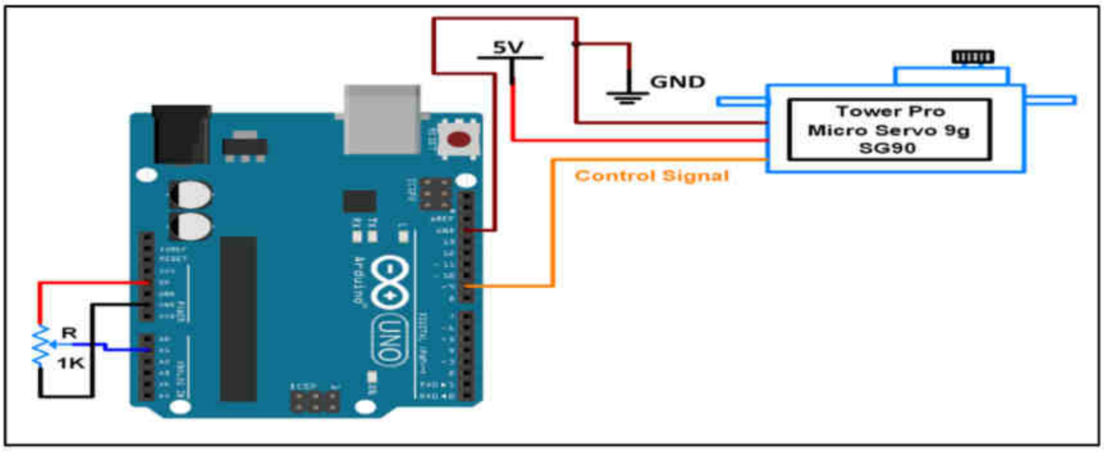


Figure 1.6 PWM signal

By controlling the PWM signal, we can easily control the position of the servo motor. The arduino has PWM pins connection through which it produces signal but the signal width is limited to 50Hz or 20ms.



Servo Interface with Arduino-UNO.

Fig 1.7 Circuit diagram

1.5.1.5 Gripper:

A gripper is something that grips things or makes it easier to grip things.

Fig 1.5 Gripper

1.5.1.6 Gyroscope MPU6050:

MPU6050 is a three-axis accelerometer and three-axis gyroscope Micro Electro-mechanical system (MEMS). It aids in the measurement of velocity, orientation, acceleration, displacement, and other motion-related features.

Fig 1.6 Gyroscope MPU6050

The pin configuration are as follows –

VCC - +5V is used for power supply.

GND - GND of Arduino.

SCL - Serial clock pulse for I2C communication.

SDA - Serial Data communication with I2C.

XDA - Connect to other I2C module for data.

XCL - Connect to other I2C module for clock.

AD0 - Pin can be used to vary default address.

INT - Interrupt pin for data

Interfacing with ARUINO-UNO

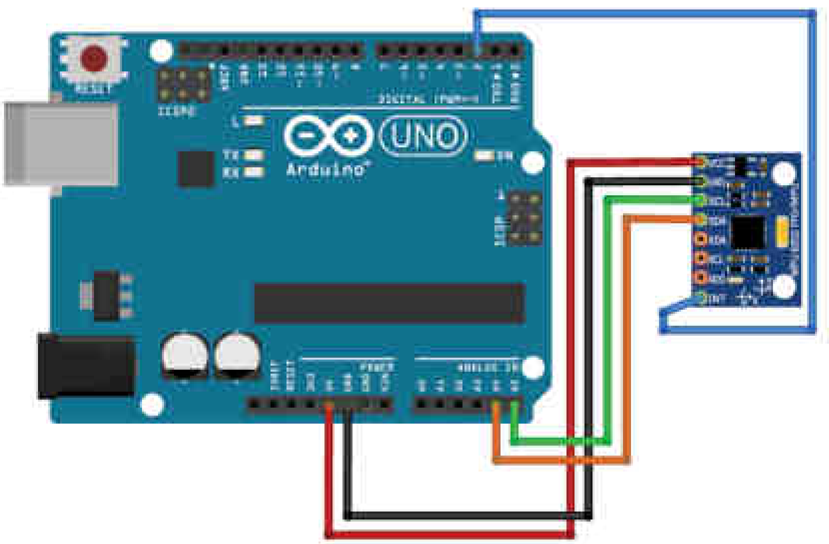


Fig 1.7 MPU6050 Circuit Diagram

1.5.1.7 Flex Sensor:

A flex sensor or bend sensor is a sensor that measures the amount of deflection or bending. Usually, the sensor is stuck to the surface, and resistance of sensor element is varied by bending the surface.

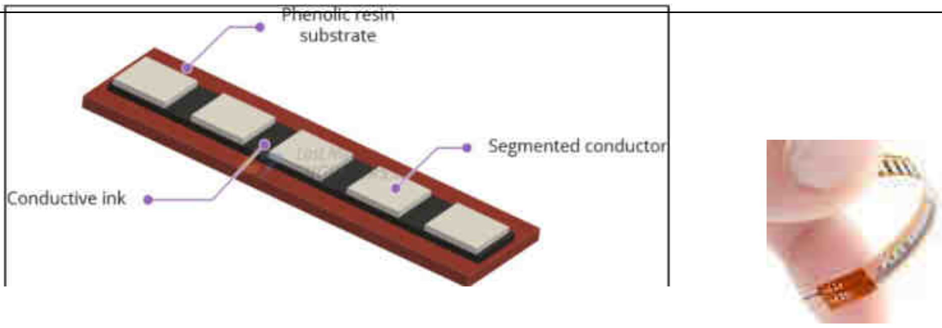


Fig 1.8 Component on Flex Sensor

Construction – flex sensor consist of phenolic resin substrate with conductive ink deposited on its surface. This conductive ink contain Segmented conductor on the top of it. It provides a flexible potentiometer in which resistance change upon bending.

Working – A resistance of the flex sensor will change according to the bend in the sensor.

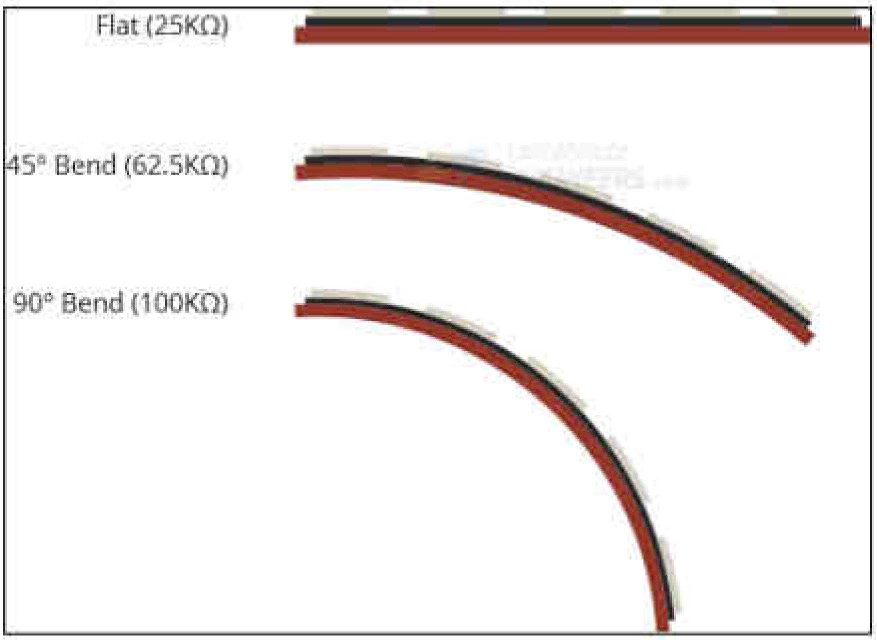


Fig 1.9 Bend Resistance

1.5.1.8 Robotic Arm Kit:

Designing a robotic arm using acrylic sheets, the degree of freedom (DoF) refers to the number of independent movements or axes the arm can control. A robotic arm made from acrylic sheets could typically have four degrees of freedom, allowing it to perform movements along four different axes

Fig 1.6 Robotic Arm Kit

1.5.1.9 Motor Driver PCA9685 12 IC Interface:

The PCA9685 servo driver module is used applications where large number of servo motors are used such as robot arm, hexapod and robots. It increases the number pf PWM output of your microcontroller. Using only two pins, you can control 16 free-running PWM outputs.

Fig 1.8 Motor Driver PCA9685

1.5.1.10 Robotic Manipulator:

A manipulator robot can be defined as an equipment consisting of a set of mechanical, electrical, and electronic components that are programmed to perform repetitive tasks automatically.

Fig 1.9 Manipulator

1.5.2 Software Requirements:

The Arduino Software (IDE) makes it easy to write code and upload it to the board offline. We recommend it for users with poor or no internet connection. This software can be used with any Arduino board.

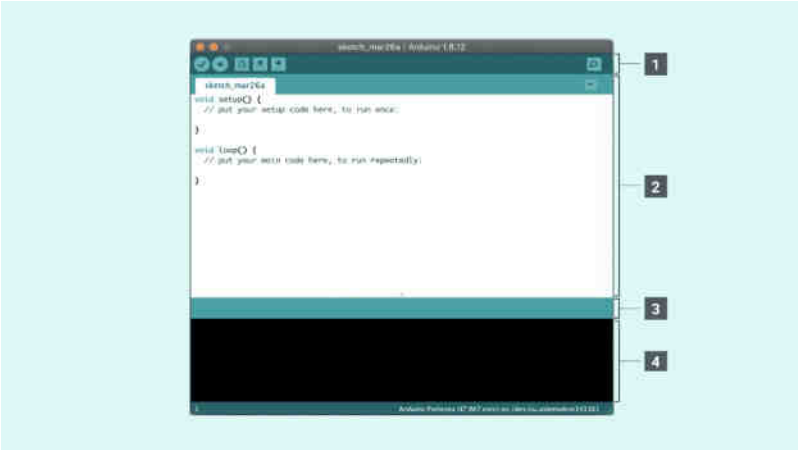


Fig 1.10 Aurduino IDE

Chapter 2: Literature Review

Literature Review: Hand Gesture-Controlled 4-DOF Robotic Arm Manipulator

Introduction:

Gesture control has emerged as a compelling interface for human-robot interaction, offering intuitive and natural means to command robotic systems. This literature review examines the advancements, challenges, and applications of hand gesture-controlled 4-degree-of-freedom (4-DOF) robotic arm manipulators. The review encompasses studies in gesture recognition, robotic arm design, sensor integration, and human-robot interaction to unveil the state-of-the-art developments and pave the way for future innovations.

Gesture Recognition in Robotics:

Research on gesture recognition techniques for robotics spans various methodologies, including vision-based, sensor-driven, and machine learning approaches. Vision-based systems leverage cameras to interpret hand movements, while sensor-driven techniques utilize accelerometers and gyroscopes for precise gesture tracking. Recent advancements in machine learning algorithms have enhanced the accuracy and versatility of gesture recognition in controlling robotic systems.

Robotic Arm Design and Mechanisms:

The design and mechanisms of 4-DOF robotic arms play a pivotal role in enabling intricate movements and precise manipulation. Kinematics, dynamics, and control algorithms form the foundation for achieving human-like motion in robotic arms. Various mechanisms, such as articulated joints and servo-driven actuators, provide the required degrees of freedom for the arm, allowing diverse and precise movements in different applications.

Sensor Integration for Gesture-Controlled Robotics:

Integration of sensors into gesture-controlled robotic systems is crucial for accurate interpretation of hand movements. Accelerometers, gyroscopes, and other sensors are integrated with control systems to convert gesture inputs into precise commands for the robotic arm. Efforts are focused on enhancing sensor accuracy, reducing latency, and improving the robustness of sensor-based control interfaces.

Human-Robot Interaction through Gesture Control:

Studies evaluating human-robot interaction (HRI) using gesture control highlight the importance of user experience and interface intuitiveness. Gesture-based interfaces provide a natural and intuitive means for users to interact with robotic systems. User satisfaction, efficiency, and ease of use are key parameters in evaluating the effectiveness of gesture-controlled interfaces in HRI scenarios.

Case Studies and Implementations:

Real-world implementations of hand gesture-controlled 4-DOF robotic arm manipulators showcase their applicability in diverse domains. Applications range from industrial assembly lines to healthcare, showcasing the adaptability and versatility of gesture-controlled robotic arms. Case studies emphasize successes in task efficiency, precision, and user acceptance, while acknowledging challenges in calibration and gesture recognition accuracy.

Emerging Trends and Future Directions:

Emerging technologies, such as artificial intelligence-driven gesture recognition and haptic feedback systems, hold promise for enhancing gesture-controlled robotic arms. Future research directions focus on improving gesture recognition accuracy, integrating multi-modal interfaces, and addressing ergonomic challenges for prolonged use.

Conclusion:

The reviewed literature underscores the significance of hand gesture-controlled 4-DOF robotic arm manipulators in revolutionizing human-robot interaction. Advancements in gesture recognition, robotic arm design, sensor integration, and user-centric interfaces pave the way for innovative applications across industries. Future research aims to refine existing technologies and explore novel approaches for enhancing the capabilities and usability of gesture-controlled robotic arms.

This literature review provides an overview of the key themes, advancements, and challenges in the domain of hand gesture-controlled 4-DOF robotic arm manipulators, highlighting their significance in shaping the future of human-robot interaction and robotics applications.

2.1 Framework Architecture:

Robotic Arms are one of the fascinating engineering creations and it is always fascinating to watch these things tilt and pan to get complex things done just like a human arm would. These robotic arms can be commonly found in industries at the assembly line performing intense mechanical work like welding, drilling, painting, etc., recently advanced robotic arms with high precision are also being developed to perform complex surgical operations.

The Flex sensor is used to control the gripper servo of Robotic Arm and the MPU6050 is used for the movement of robotic in X and Y-axis. If you do not have a printer, you can also build your arm with simple cardboard as we built for our Arduino Robotic Arm Project. For inspiration, you can also refer to the Record and Play Robotic Arm that we built earlier using Arduino.

MPU6050 is based on Micro-Mechanical Systems (MEMS) technology. This sensor has a **3-axis accelerometer, a 3-axis gyroscope, and an in-built temperature sensor**. It can be used to measure parameters like Acceleration, Velocity, Orientation, Displacement, etc.

The framework architecture of a hand gesture-controlled 4-degree-of-freedom (4-DOF) robotic arm project typically involves several interconnected components working together to enable gesture recognition and precise control of the robotic arm. Here's an outline of the framework architecture:

Gesture Input Interface:

Sensor Devices: Include sensors like accelerometers, gyroscopes, or vision-based cameras to capture hand movements or gestures.

Data Acquisition: Capture and process data from sensors to interpret specific hand gestures accurately.

Gesture Recognition Algorithms: Implement algorithms to recognize and interpret hand gestures based on sensor data.

Control System:

Gesture-to-Motion Conversion: Translate recognized gestures into corresponding robotic arm movements.

Kinematics and Dynamics Processing: Calculate the required joint angles or movements for the 4-DOF robotic arm based on the recognized gestures.

Control Algorithms: Use control algorithms to generate precise commands for each degree of freedom in the robotic arm.

Robotic Arm Mechanism:

Actuation Mechanisms: Utilize actuators (such as servo motors) at each joint of the robotic arm to facilitate movement.

Joint Control: Control the movement of each joint based on the commands received from the control system.

Feedback Mechanism: Implement feedback mechanisms to ensure accurate positioning and feedback on the arm's current state.

Communication and Integration:

Wireless Connectivity: Enable wireless communication between the gesture input interface and the control system to ensure real-time interaction.

Sensor-Controller Integration: Establish a seamless integration between the sensor data acquisition, gesture recognition, and control system modules.

Hardware Integration: Ensure compatibility and proper integration of sensors, microcontrollers, actuators, and communication modules.

User Interface and Interaction:

User Feedback: Provide visual or haptic feedback to users indicating successful gesture recognition and corresponding robotic arm movement.

Interface Design: Develop an intuitive and user-friendly interface for users to initiate and interact with the robotic arm using gestures.

System Validation and Testing:

Calibration and Testing: Perform calibration of sensors, actuators, and control algorithms for accuracy and reliability.

Simulation and Validation: Conduct simulations or validation tests to ensure precise and responsive control of the robotic arm based on gesture inputs.

Future Expansion and Adaptability:

Modular Design: Ensure a modular design that allows for easy upgrades or modifications to accommodate additional degrees of freedom or new functionalities.

Compatibility and Scalability: Design the framework to be compatible with emerging technologies and adaptable to different robotic arm configurations.

This framework architecture outlines the key components and their interactions within a hand gesture-controlled 4-DOF robotic arm project. Implementing this framework requires careful integration, testing, and optimization of each component to achieve accurate and responsive gesture-based control of the robotic arm.

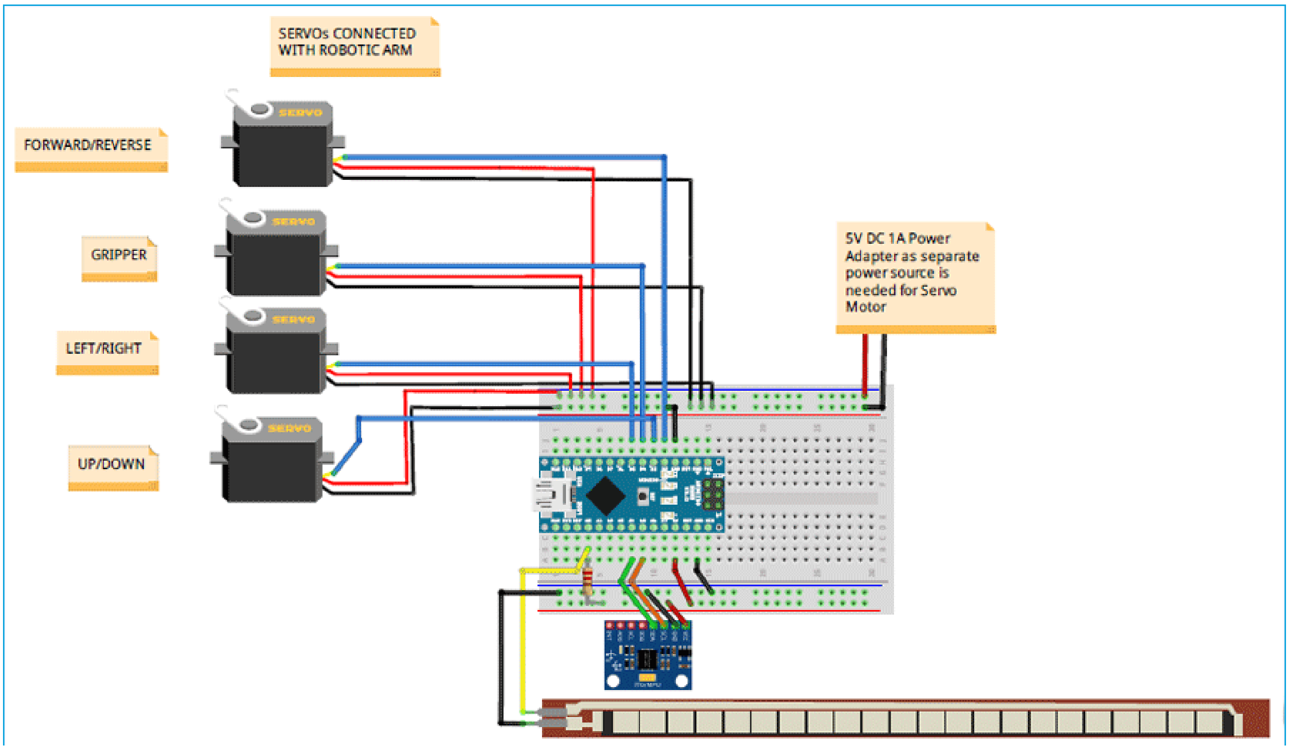


Fig 2.1 Circuit Diagram

Chapter 3: Preliminary design

It involves choosing appropriate gesture recognition methods, including sensor types, image processing techniques, and machine learning models for accurate interpretation. Planning the control mechanism incorporates algorithms to translate recognized gestures into specific arm movements. Designing the robotic arm entails specifying degrees of freedom, actuators, and end effectors aligned with intended tasks. Considerations also encompass user interfaces, integration plans, safety protocols, scalability, and compliance standards. This phase serves as a blueprint, outlining fundamental components and interactions, guiding subsequent development, prototyping, and testing phases for the gesture-controlled robotic arm system.

3.1 Theoritical Background of Robotic Arm:

Robotic is defined as the study, design, and use of robotic systems for manufacturing. With the rise in manufacturing industrial activities, a robotic arm is invented to help various industries to perform a task or work instead of using manpower. Robots are generally used to perform unsafe, hazardous, highly repetitive, and unpleasant tasks. Robots can perform material handling, assembly, arc welding, resistance welding, machine tool load and unload function, painting, and spraying, etc. It is very useful because it possesses high precision, intelligence, and endless energy levels in doing work compared to human beings. For example, a robotic arm is widely used in assembling or packing lines by lifting small objects with a repetitive motion that humans could not bear to do in a long period of time. The light material lifting task can be done by the robotic arm efficiently and time-saving because it is not restricted by fatigue or health risks that man might experience. Robots and humans share a common feature. Humans and mechanical robots—as opposite as they may seem, they, in fact, share the same underlying structure of links (bones) and joints. The basic skeleton of industrial robots, which is made up of mainly robotic arms, is a combination of links and joints. Relating it to a human body, parts that can freely bend and move about, such as the elbow and shoulder, are the joints, and the bones connecting those joints are equivalent to a robot's links. The principle of moving joints and transmitting power through the links is common in both humans and robots.

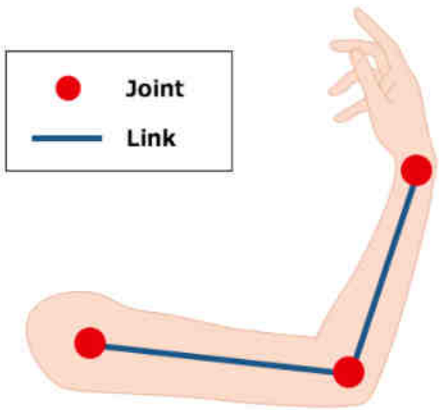


Fig 3.1 A human elbow and shoulder are joints representing the joints

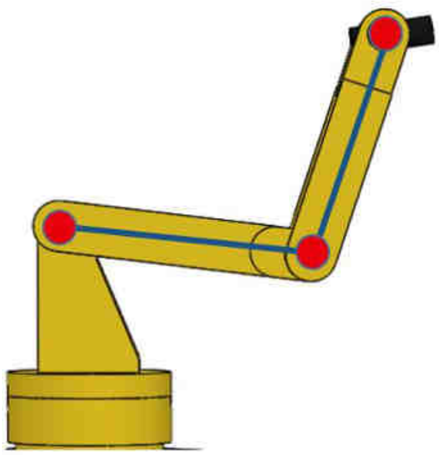


Fig 3.2 Robotic Arm

3.2 Robotic Arm:

A robotic arm is a programmable mechanism comprising two or more segments linked by means of joints into a kinematic chain. Each joint in the chain is a servo or another motor providing either rotational or linear displacement of the segments. The number of linkages in the structure defines how many freedom degrees (DOF) it has—typically, ranging from two to the human arm maximum of seven.

3.2.1 Types of Robotic Arm:

Serial Link: Serial manipulators are the most common industrial robots and they are designed as a series of links connected by motor-actuated joints that extend from a base to an end-effector. Often they have an anthropomorphic arm structure described as having a "shoulder", an "elbow", and a "wrist".

Cartesian robot/ Gantry robot: Used for pick and place work, application of sealant, assembly operations, handling machine tools, and arc welding. It's a robot whose arm has three prismatic joints, whose axes are coincident with a Cartesian coordinator.

Cylindrical robot: Used for assembly operations, handling at machine tools, spot welding, and handling at diecasting machines. It's a robot whose axes form a cylindrical coordinate system.

Spherical robot / Polar robot: Used for handling machine tools, spot welding, diecasting, fettling machines, gas welding, and arc welding. It's a robot whose axes form a polar coordinate system.

SCARA robot: Used for pick and place work, application of sealant, assembly operations, and handling machine tools. This robot features two parallel rotary joints to provide compliance in a plane.

Articulated robot: Used for assembly operations, diecasting, fettling machines, gas welding, arc welding, and spray painting. It's a robot whose arm has at least three rotary joints.

Parallel robot: One use is a mobile platform handling cockpit flight simulators. It is a robot whose arms have concurrent prismatic or rotary joints.

Anthropomorphic robot: It is shaped in a way that resembles a human hand, i.e. with independent fingers and thumbs.

3.2.2 Robot and Human motion comparison:

The following figure shows a comparison between robot and human movement. The 1st to the 3rd axes are the waist and the arm, and the 4th to the 6th axes are from the wrist up to the fingertips. The first three axes carry the wrist to a specific position, and the next three axes move the wrist freely. This 6-axis construction allows robots to move freely like humans can.

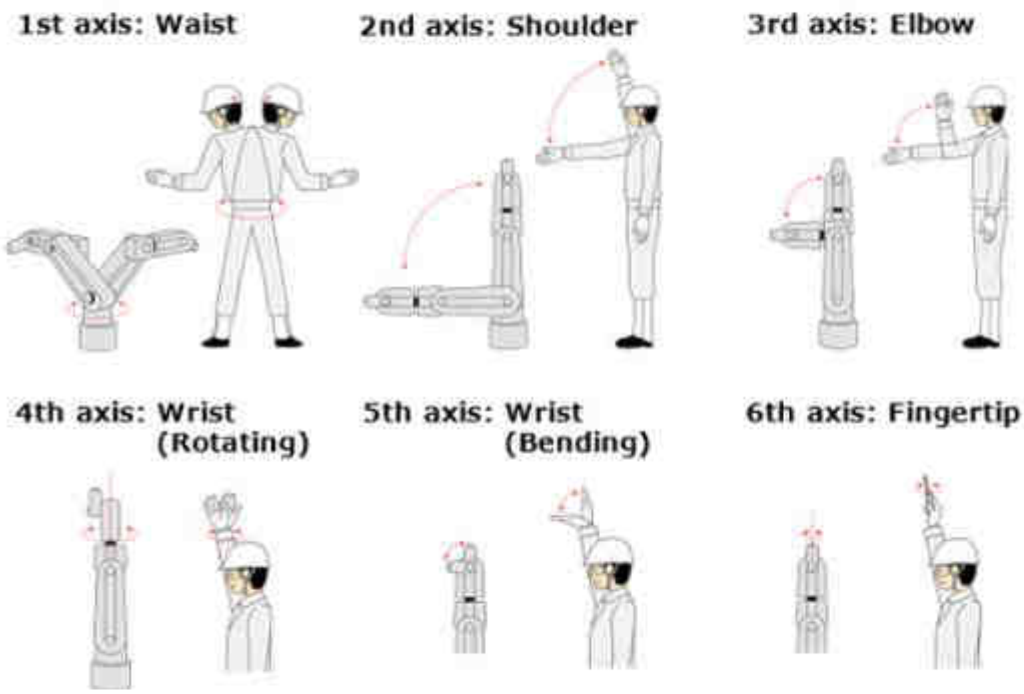


Fig 3.3 Overview of robot and human motion comparision

3.3 Kinematic Analysis and Design of Robotic Arms:

As the basis for the motion control of the robotic arm, the kinematic analysis of the robotic arm is the study of the relationship between the motion of the robotic arm in each coordinate system [15,16]. Before carrying out the kinematic analysis of the robotic arm, the arm is first modeled in Matlab using the D-H model with the following basic parameters: theta is the joint angle, d is the linkage offset [17,18], a is the linkage length, and alpha is the linkage torsion angle.

The basic joint model can be found in Figure 2. Once the model has been established, the workspace of the joint model can be determined by the Monte Carlo method [19,20]. In this section, the Monte Carlo method is used to solve the workspace as follows [21]: firstly, random variables are generated for each joint, and a random set of joint space vectors are generated for the robotic arm, which is calculated by using 10,000 points. Secondly, the kinematic positive solution is calculated and mapped from the joint space to the end workspace (Cartesian coordinate system), and finally, the result is plotted in Figure 3. It can be seen from Figure 3 that the result of the arm motion contains any coordinate in the 3D space, which shows that the designed robotic arm can reach any specified point in the actual work. After the arm has been modeled, a kinematic analysis of the arm is carried out, including both positive and negative kinematics.

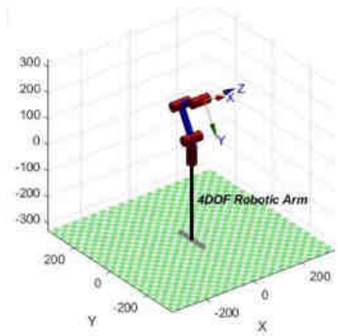


Fig 3.4 Robotic Arm Joint model

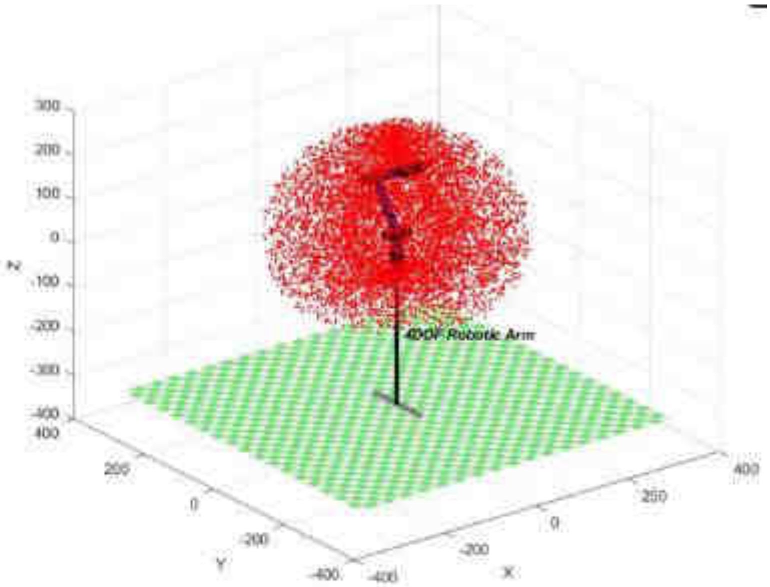


Fig 3.5 Robotic Arm Workspace

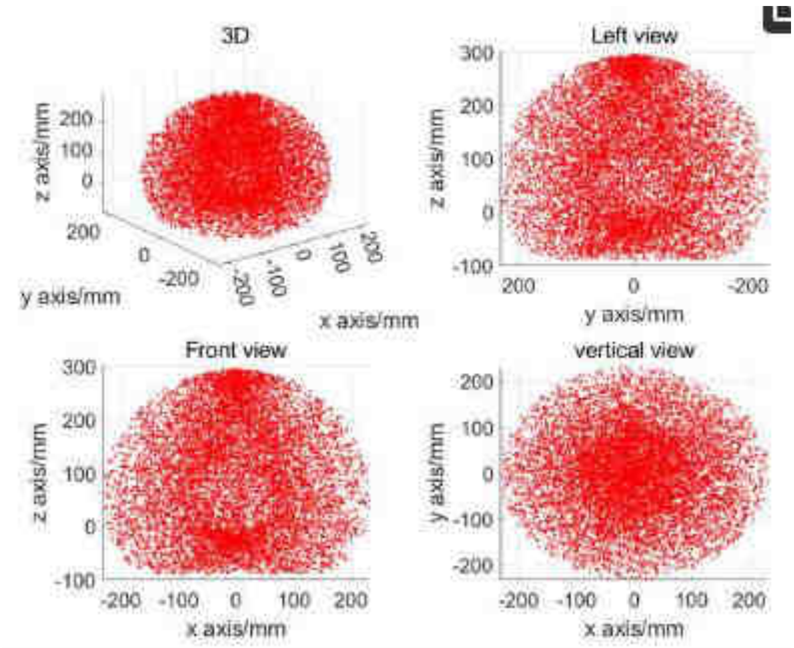


Fig 3.6 The movement space of the robotic arm

3.4 Control Mechanism and Methodology:

The control mechanism of a robotic arm orchestrates precise movements by interpreting user commands or system inputs. It comprises algorithms, microcontrollers, and software coordinating the arm's actuators and joints. These algorithms translate recognized gestures or received commands into specific joint angles and motions, enabling the arm to execute desired actions accurately. Control mechanisms ensure synchronization between the arm's degrees of freedom, regulating the sequence and speed of movements for optimal performance. Additionally, feedback systems within the control mechanism validate executed actions, maintaining accuracy and adjusting for any discrepancies during operation. This crucial component governs the arm's responsiveness, precision, and adaptability, playing a pivotal role in realizing intuitive and efficient human-robot interaction.

3.2.1 Methodology:

Proposed methodology aims to build a robotic hand which efficiently translate the hand gestures into the movement of robotic hand (made of any material). Movement of hand in the specified direction will transmit a command to the robot which will then move in a specific direction. Proposed technique to control robotic hand using hand gesture is divided into 2 subparts:

- Transmitter section
- Receiving section

Figure 1 shows the flow diagram of the whole system, i.e. performing hand gesture identification and robot control. On the basis of the gesture identified, the robotic hand shows the same gestures as performed by the hand.

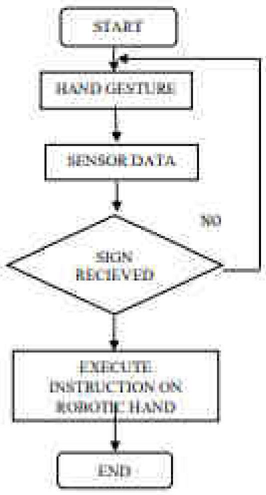


Fig 3.7 Flow diagram for robot system

The Transmitting section consists of one Arduino Nano, 7 flex sensors and one RF transmitter module. The Receiver section consists of Arduino Mega, 8 Servo Motors and one RF Receiving module. Here we will require two 5V power supplies which will be applied to both sections. The Arduino Nano will read the analog output values from the flex sensors and convert the analog values into digital values. The digital values will be processed by the Arduino Nano and will be sent to the RF transmitter which is received by the Receiver and will be processed at the receiver end which drives the motor to the particular direction. Figure 3.8 shows the whole system block diagram for controlling robotic hand thorough wires.

3.2.2 Robotic Glove:

Robotic glove holds the circuitry which controls the robotic hand. It consists of Arduino which is programmed in such a way that it transfers the required data with the help of a transmitter Module. At the same time the Flex sensor is doing its job by sending the degree of movement of the finger to the Arduino Nano. The processed values are then transmitted from the Module (NRF Transmitter) to the robotic hand. The module takes the feedback from the hand and sends the new processed signals to it.

Fig 3.8 Gesture Control Glove

3.2.3 Robotic Hand:

It is the main part where implementation of the program from the robotic glove takes place. It consists of total of 8 Servos, connected in such a way that it provides 3 DOF's (Degrees of Freedom) to the system. A microcontroller Arduino inputs the values from the module and sends the data accordingly to the servos. Both the circuitry and base are clipped upon the common base, broad and thick enough to improve the stability. 5 servos are attached for controlling the movement of fingers. Two more servos are for wrist movements and therefore to control the overall movement of hand one servo is attached to the base. As mentioned above, the robotic hand mimics the movement of glove worn by the user, when the glove is tilted in the forward direction or any such direction, the arm spontaneously follows the suit.

Fig 3.9 Gesture Control Hand

3.2.4 Circuit Diagram:

Gesture Sensor to Microcontroller: Connect the output of the gesture sensor (e.g., camera) to the input pins of the microcontroller (e.g., Arduino). Use appropriate interfaces or communication protocols (e.g., I2C, UART, or GPIO).

Microcontroller to Motor Drivers: Connect the microcontroller to motor driver modules. Motor drivers control the power and direction of the servo motors.

Motor Drivers to Servo Motors: Connect the output pins of the motor drivers to the input pins of the servo motors. Ensure proper wiring to control each motor's movement.

Power Supply: Provide adequate power to the system, ensuring it meets the requirements of the microcontroller, sensors, motor drivers, and servo motors. Use voltage regulators or appropriate power sources for stability.

Control Logic: Program the microcontroller to interpret gesture data and convert it into motor control signals. Implement logic to translate recognized gestures into specific motor movements (e.g., Arduino code for gesture recognition and servo motor control).

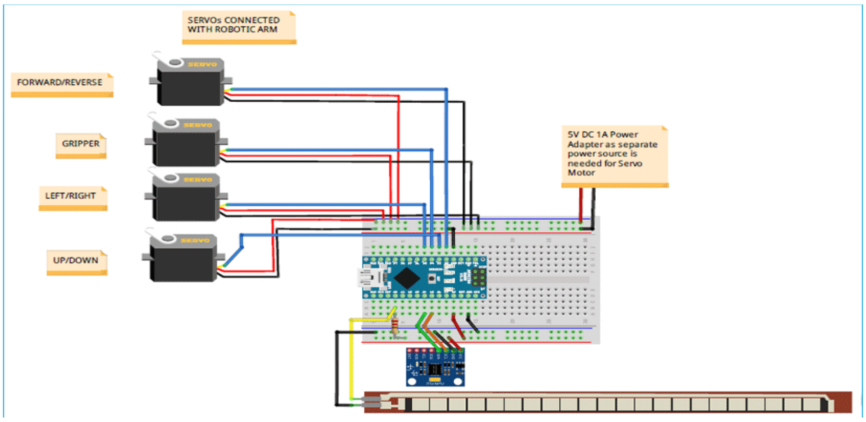


Fig 3.10 Circuit Diagram

3.2.5 Working of Robotic Arm:

Finally, upload the code to Arduino.

1. Now move the hand down to move the robotic arm forward and move up to move the robotic arm up.

2. Then tilt the hand left or right to turn the robotic arm left or right.

3. Bend the flex cable attached with the hand glove's finger to open the gripper and then release it to close it.

Fig 3.5 Working of Robotic Arm

3.5 Programming for Robotic Arm:

1. First, include the necessary library files. Wire.h library is used for I2C communication between Arduino Nano & MPU6050 and servo.h for controlling servo motor.

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

2. Next, the objects for the class servo is declared. As we use four servo motors, four objects such as servo_1, servo_2, servo_3, servo_4 are created.

```
Servo servo_1;
Servo servo_2;
Servo servo_3;
Servo servo_4;
```

3. Next, the I2C address of MPU6050 & the variables to be used is declared.

```
const int MPU_addr=0x68;        //MPU6050 I2C Address
int16_t axis_X,axis_Y,axis_Z;
int minVal=265;
int maxVal=402;
double x;
double y;
double z;
```

4. Next in the void setup, a baud rate of 9600 is set for Serial communication.

```
  Wire.begin();                      //Initilize I2C Communication
  Wire.beginTransmission(MPU_addr); //Start communication with MPU6050
  Wire.write(0x6B);                  //Writes to Register 6B
  Wire.write(0);                     //Writes 0 into 6B Register to Reset
  Wire.endTransmission(true);        //Ends I2C transmission
```

5. Next in the void loop function, again establish I2C connection between the MPU6050 and Arduino Nano and then start to read the X, Y, Z-Axis data from the register of MPU6050 and store them in corresponding variables.

```
  Wire.beginTransmission(MPU_addr);
  Wire.write(0x3B);                  //Start with regsiter 0x3B
  Wire.endTransmission(false);
  Wire.requestFrom(MPU_addr,14,true);  //Read 14 Registers
                        axis_X=Wire.read()<<8|Wire.read();
                        axis_Y=Wire.read()<<8|Wire.read();
                        axis_Z=Wire.read()<<8|Wire.read();
```

After that, map the min and max value of the axis data from the MPU6050 sensor in the range of -90 to 90.

```
                        int xAng = map(axis_X,minVal,maxVal,-90,90);
                        int yAng = map(axis_Y,minVal,maxVal,-90,90);
                        int zAng = map(axis_Z,minVal,maxVal,-90,90);
```

Then use the following formula to calculate the x, y, z values in terms of 0 to 360.

```
x= RAD_TO_DEG * (atan2(-yAng, -zAng)+PI);
y= RAD_TO_DEG * (atan2(-xAng, -zAng)+PI);
                        z= RAD_TO_DEG * (atan2(-yAng, -xAng)+PI);
```

Then read the flex sensor Analog output data at the Arduino Nano's A0 pin and according to the digital value of the flex sensor set the servo angle of the gripper. So if the flex sensor data is greater than 750 the servo motor angle of the gripper is 0 degree and if less than 750 it is 180 degrees.

```
int gripper;
int flex_sensorip = analogRead(A0);
if(flex_sensorip > 750)
      {
        gripper = 0;
      }
      else
      {
        gripper = 180;
      }
 servo_3.write(gripper);
```

Then the **movement of MPU6050 on the X-axis from 0 to 60** is mapped in terms of 0 to 90 degrees for the servo motor's Forward/Reverse motion the Robotic arm.

```
  if(x >=0 && x <= 60)
{
   int mov1 = map(x,0,60,0,90);
   Serial.print("Movement in F/R = ");
   Serial.print(mov1);
   Serial.println((char)176);
   servo_1.write(mov1);
}
```

And the **movement of MPU6050 on the X-axis from 250 to 360** is mapped in terms of 0 to 90 degrees for the servo motor's UP/DOWN motion Robotic arm.

```
else if(x >=300 && x <= 360)
{
   int mov2 = map(x,360,250,0,90);
   Serial.print("Movement in Up/Down = ");
   Serial.print(mov2);
   Serial.println((char)176);
   servo_2.write(mov2);
}
```

M**ovement of MPU6050 on the Y-axis from 0 to 60** is mapped in terms of 90 to 180 degrees for the servo motor's Left Movement of the Robotic arm.

```
if(y >=0 && y <= 60)
 {
    int mov3 = map(y,0,60,90,180);
    Serial.print("Movement in Left = ");
    Serial.print(mov3);
    Serial.println((char)176);
    servo_4.write(mov3);
 }
```

Movement of MPU6050 in the Y-axis from 300 to 360 is mapped in terms of 0 to 90 degrees for the servo motor's Right Movement of the Robotic arm.

```
 else if(y >=300 && y <= 360)
 {
    int mov3 = map(y,360,300,90,0);
    Serial.print("Movement in Right = ");
    Serial.print(mov3);
    Serial.println((char)176);
    servo_4.write(mov3);
 }
```

Chapter 4: Final Analysis and Design

4.1 Result:

Finally, upload the code to Arduino.

1. Now move the hand down to move the robotic arm forward and move up to move the robotic arm up.

2. Then tilt the hand left or right to turn the robotic arm left or right.

3. Bend the flex cable attached with the hand glove's finger to open the gripper and then release it to close it.

4.1.1 Output:

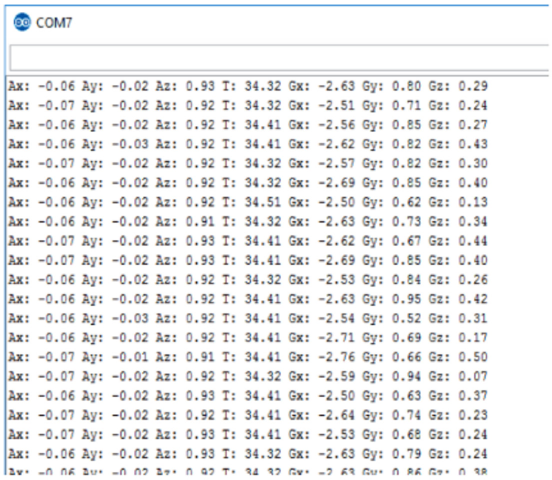


Fig 4.1 Output window of MPU6050

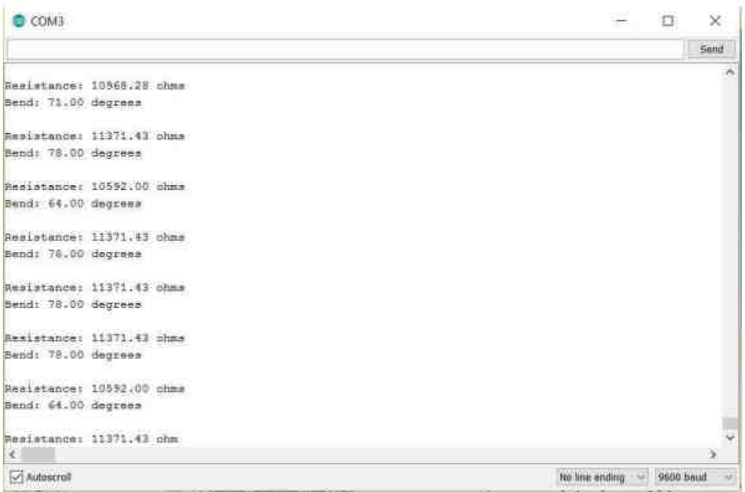


Fig 4.2 Output window flex sensor

4.1.2 Real time Working of robotic arm:

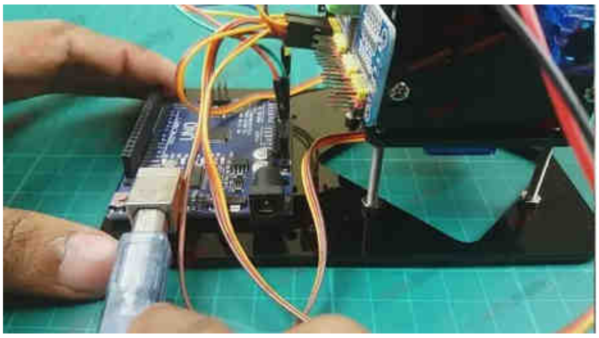


Fig 4.3 Side view representing the circuit

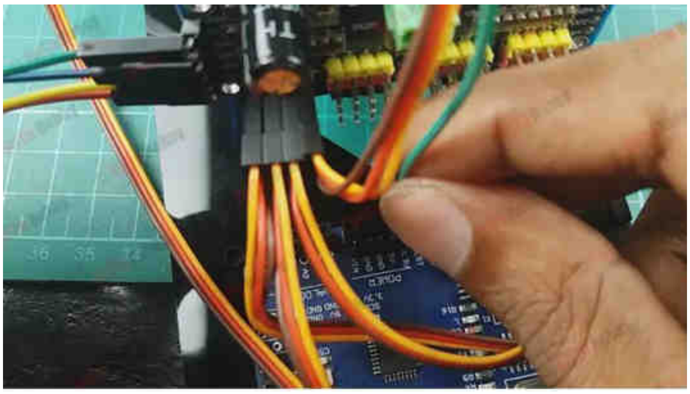


Fig 4.4 Top view representing the gyroscope connection

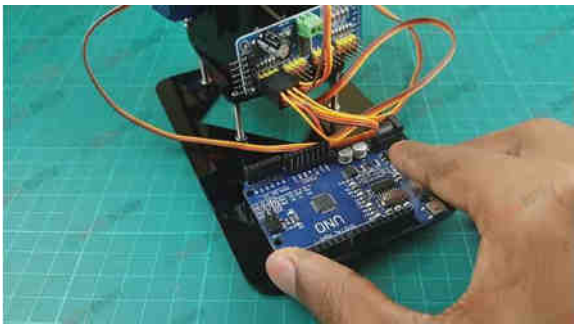


Fig 4.4 Front view of the robotic arm

Fig 4.5 Hand Glove

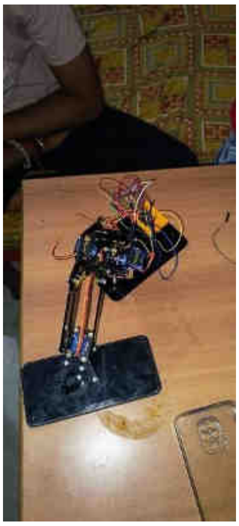


Fig 4.6 Top view of the Robotic Model

Fig 4.7 Top view of glove and Arm

4.2 Result Analysis:

In analyzing the results of a gesture-controlled robotic arm system, several key aspects are considered to evaluate its functionality, accuracy, and user interaction.

4.2.1 Gesture Recognition Accuracy:

Assess the system's ability to accurately interpret and classify hand gestures. Measure recognition rates, false positives, and false negatives to determine the reliability of gesture interpretation. Analyze the system's performance across various gestures and environmental conditions.

4.2.2 Robotic Arm Movement Precision:

Evaluate the robotic arm's ability to execute movements corresponding to recognized gestures. Measure the accuracy of joint angles, end-effector positioning, and the arm's responsiveness to gesture commands. Conduct tests for different tasks and complexities to gauge the arm's precision.

4.3 Application:

4.3.1 Manufacturing and Industry: In assembly lines and manufacturing plants, gesture-controlled robotic arms can streamline processes by allowing workers to control machinery with intuitive hand gestures.

4.3.2 Healthcare and Rehabilitation: Gesture-controlled robotic arms find applications in assisting individuals with limited mobility or motor impairments. These systems can aid in rehabilitation exercises, prosthetic limb control, or supporting daily tasks, providing greater independence and control for individuals with disabilities.

4.3.3 Education and Research: Gesture-controlled robotic arms serve as valuable tools in educational settings, enabling students and researchers to explore robotics, human-machine interaction, and artificial intelligence.

4.4 Problem Faced:

Achieving consistent and accurate gesture recognition poses a significant challenge. Variations in lighting conditions, hand orientations, and background clutter can hinder the system's ability to interpret gestures reliably.

Coordinating various components, including sensors, processors, and actuators, while ensuring real-time synchronization can be complex. Latency issues or communication delays between the gesture recognition module and the robotic arm can affect responsiveness.
Translating recognized gestures into precise arm movements requires intricate mapping algorithms. Aligning the interpreted gestures with specific joint angles and motions might present challenges, especially for complex gestures or multi-step commands.

4.5 Limitation:

Gesture Recognition Challenges: Variability in lighting conditions, background clutter, or hand positions can hinder accurate gesture recognition. Complex or subtle gestures may be challenging to interpret consistently, impacting the system's reliability.
Limited Gesture Vocabulary: Systems may have constraints in recognizing a broad range of gestures. Limited gesture vocabulary restricts the variety of commands users can input, potentially limiting the versatility of interactions.
Hardware and Processing Limitations: Processing power and sensor capabilities may limit the system's real-time responsiveness and accuracy. Delays or lags in gesture recognition and arm movement execution can diminish user experience and precision.
Environmental Interference: External factors like noise, interference, or varying environmental conditions might disrupt sensor readings or gesture recognition algorithms, impacting the system's performance and reliability.
User Learning Curve: Mastering the gestures required for precise control of the robotic arm may present a learning curve for users. Complex or non-intuitive gestures could make the system less accessible or require additional training for users to operate effectively.

4.6 Conclusion:

In conclusion, the development of a gesture-controlled robotic arm represents a significant stride in human-machine interaction, yet challenges persist. The system showcases promise in offering intuitive control through hand gestures, fostering accessibility and versatility across various domains.
Throughout this endeavor, notable achievements include advancements in gesture recognition, precision in robotic arm movements, and strides in user interface design. However, limitations in gesture recognition accuracy, hardware constraints, and user learning curves pose ongoing challenges.

The system's success in real-world applications hinges on addressing these limitations. Future advancements should prioritize enhancing gesture recognition reliability, optimizing hardware for seamless performance, and refining user interfaces for intuitive control. Additionally, fostering adaptability to diverse environments and expanding gesture vocabularies will broaden its practical applicability.

References

The major reference of our project is internet and we use the following websites:

1. Circuit Digest
2. Research Gate
3. Marker Pro
4. Dhepekar, Puja, and Yashwant G. Adhav. "Wireless robotic hand for remote operations using flex sensor." 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT). IEEE, 2016.
5. Neto, Pedro, J. Norberto Pires, and A. Paulo Moreira. "Accelerometer-based control of an industrial robotic arm." RO-MAN 2009- The 18th IEEE International Symposium on Robot and Human Interactive Communication. IEEE, 2009.