Republic of Yemen
Ministry of higher Education and Scientific research
University of EMIRATES INTERNATIONAL UNVERISTY
Faculty Of Engineering and Information Technology
Department of Mechatronics Engineering

A Robotic Arm Controlled Via Hand Gestures Using an ESP Module and Wi-Fi Connection

ذراع روبوتية يتم التحكم بها عبر إيماءات اليد باستخدام وحدة ESP واتصالWi-Fi

1- Sohaib Muneer Abdullah Al-Mekhlaife .

2 - Hashem Ashraf Al-Mutawakel

3- Ali Suleiman Al-Nadhari, .

Supervised by:

Associate Professor Dr. Farouq Al-Fuhaidi

A graduation project report submitted to the department of Mechatronics Engineering in partial fulfillment of the requirements of bachelor degree in Faculty Of Engineering and Information Technology ,Department of Mechatronics Engineering

2025-1446

Summary

A Robotic ARM Controlled via Hand Gestures Using an ESP Module and Wi-Fi Connection

With rapid technological advancements, controlling robots through hand gestures has become a modern trend in automation and human-machine interaction. This project aims to design and implement a robotic arm controlled via hand gestures using an ESP (Electronic stability program) module and Wi-Fi connection, offering a more interactive and seamless experience compared to traditional systems that rely on Bluetooth or wired control.

Problem Statement

Traditional robotic control systems often use wired controllers or Bluetooth, limiting operational range and real-time responsiveness. Additionally, some control methods are not user-friendly or intuitive. This project addresses the need for a more accessible and seamless way to control robots using hand gestures over Wi-Fi, enabling remote control without wired connections or additional devices.

Project Objectives

1. Develop a robotic arm controllable by hand gestures using motion tracking sensors.
2. Utilize an ESP module instead of Arduino to enable Wi-Fi-based communication.
3. Ensure real-time responsiveness so that the robotic arm executes commands without noticeable delays.
4. Design a web-based control interface allowing users to monitor and control the robotic arm remotely.
5. Optimize data transmission efficiency between the control unit and the robotic arm via Wi-Fi for smooth performance.
6. Ensure ease of use so that anyone can operate the system without advanced technical knowledge.

Project Methodology

1. **Requirements Analysis:** Define functional and non-functional requirements and select suitable hardware and software components.

2. **System Design:** Develop the gesture recognition system, implement algorithms to analyze gestures and convert them into commands, and establish Wi-Fi communication between the ESP module and the robotic arm.
3. **Prototype Development:** Program the ESP module to process gesture data, implement a web-based control interface, and test the robotic arm's execution of gesture commands.
4. **Testing and Validation:** Evaluate the robotic arm's responsiveness to commands via Wi-Fi and assess the accuracy of gesture recognition.
5. **Final Documentation and Analysis:** Prepare a final report detailing project outcomes, challenges faced, and recommendations for future enhancements.

Challenges and Solutions

1. **Gesture Recognition Accuracy:** Improve sensor precision and implement advanced recognition algorithms.
2. **Wireless Connection Stability:** Use enhanced connectivity techniques to ensure network stability.
3. **Response Time:** Optimize communication protocols to minimize delays between command transmission and execution.

Results and Recommendations

The project successfully demonstrated precise and seamless control of the robotic arm via Wi-Fi, providing real-time responsiveness and an intuitive user interface. Initial tests showed reliable performance, but further improvements are recommended, including enhancing sensor accuracy and expanding the operational range. Additionally, integrating artificial intelligence can improve gesture recognition and system adaptability over time.

Conclusion

This project represents a significant step toward developing more interactive and user-friendly robotic control systems, with applications spanning industries, healthcare, and education. Future developments could integrate AI technologies and 5G connectivity to further enhance system efficiency and performance.

Authorization

We authorize university of University of Emirates International University faculty of Engineering and Information Technology , Department of Mechatronics Engineering ..to supply copies of our graduation project report to libraries, organizations or individuals on request. The faculty, also authorized to use it in local or international competitions.

No	Student Name	Signature	Date
1	Sohaib Muneer Abdullah Al-Mekhlaife		
2	Hashem Ashraf Al-Mutawakel		
3	Ali Suleiman Al-Nadhari		

Dedication

"**I, Engineer Sohaib Muneer Al-Mekhlaife** gratefully dedicates this work to his beloved parents, brothers, and sisters. A special tribute is extended to his esteemed father, Professor Dr. Muneer Abdullah Saeed Al-Mekhlaife, whose unwavering support and constant inspiration have been a guiding light throughout his life. Heartfelt appreciation is also extended to all those who have stood by him throughout his academic and personal journey. Your love, encouragement, and dedication have been instrumental in reaching this milestone. With deepest gratitude and respect

"**I, Engineer Hashem Ashraf Al-Mutawakel**, sincerely dedicate this work to my beloved parents, brothers, and sisters, whose constant love and support have been invaluable. A heartfelt tribute goes to my dear father, Dr. Ashraf Al-Mutawakel, whose guidance and unwavering encouragement have always been a source of strength and inspiration. I am also deeply grateful to everyone who accompanied me throughout my academic and life journey. Your faith, motivation, and dedication have made this achievement possible. With profound appreciation and respect

"I, **Engineer Ali Suleiman Al-Nadhari**, dedicate this work to my dear parents, brothers, and sisters. Special thanks and dedication to my beloved father, Engineer Suleiman Al-Nadhari, who has always been a source of inspiration and support for us. I would also like to thank everyone who stood by us throughout our life journey and studies. Without your love, encouragement, and dedication, we would not have reached this point. With sincere gratitude."

"**Finally**, To our dear friends, who stood by our side, offering motivation, companionship, and support throughout this journey. To Emirates International University, for equipping us with knowledge, resources, and a learning environment that enabled us to achieve this milestone. With heartfelt gratitude, we dedicate this project to all those who made this achievement possible

The project Team

Acknowledgment

Before and above all, we express our deepest gratitude to **Allah** for granting us the strength, patience, and perseverance to complete this project successfully. We would like to extend our sincere appreciation and profound gratitude to our supervisor, **Associate Professor Dr. Farouq Al-Fuhaidi**, for his invaluable guidance, continuous support, and insightful advice throughout this journey. His expertise, patience, and encouragement have played a crucial role in shaping our work and helping us overcome challenges. Our heartfelt thanks go to **Associate Professor Dr. Radwan Al-Badhaiji,** Head of the Mechatronics Engineering Department**,** for his outstanding leadership, continuous encouragement, and unwavering support throughout our academic journey. His dedication to fostering an enriching learning environment has greatly contributed to our growth as students and researchers. We also extend our sincere gratitude to all the faculty members of the Department of Mechatronics Engineering for their dedication, valuable knowledge, and mentorship, which have played a significant role in enhancing our technical and academic skills. A special thanks to Emirates International University, particularly the Faculty of Engineering and Information Technology**,** for providing us with the necessary resources, facilities, and an excellent academic atmosphere that supported our research and development. Finally, we are deeply grateful to our families and friends for their unwavering support, motivation, and encouragement, which have been a constant source of inspiration throughout this journey. This project is a testament to the collective efforts of our esteemed professors, university, and loved ones, and we are honored to have had such a remarkable academic experience. Last but not least, we owe a great deal of gratitude, thanks and appreciation to all members of our families, for their kind support, help and encouragement.

Supervisor Certification

I certify that the preparation of this project entitled:
"A Robotic Arm Controlled via Hand Gestures Using an ESP Module and Wi-Fi Connection"

Prepared by:

1. Sohaib Muneer Abdullah Al-Mekhlaife
2. Hashem Ashraf Al-Mutawakel
3. Ali Suleiman Al-Nadhari

was made under my supervision at the Faculty of Engineering and Information Technology, Department of Mechatronics Engineering, in partial fulfillment of the requirements for the Bachelor's degree in Mechatronics Engineering.

Supervisor :**Associate Professor Dr. Farouq Al-Fuhaidi**
Signature: ___________________________
Date: ___________________________

Examiner Committee

Project Title:

A Robotic ARM Controlled Via Hand Gestures Using an ESP Module and Wi-Fi Connection

Supervisor

No	Name	Position	Signature
1	Associate Professor Dr. Farouq Al-Fuhaidi		

Examiner Committee

No	Name	Position	Signature
1	Associate Professor Dr Radwan Al-Badhaiji		
2	Assistant Professor Dr.Hisham Aqlan		
3			

Department Head

....

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Acronym	Definition
ESP	Electronic stability program

Mathematical Notation

Parameter	Description
B_{RB}	Resource block bandwidth.
$E_{(l)}^{(i)}$	Available energy in relay l at the end of frame i.
$E_{(l)}^{(i,max)}$	Maximum energy relay l can store.
$E_{(l)}^{(i-1)}$	Available energy in relay l at the beginning of frame i.

Chapter 1

Introduction

Chapter 1: Introduction

1.1 Overview

With technological advancements and the increasing number of internet-connected devices, controlling robots via hand gestures has become a modern trend in automation and human-machine interaction illustrates the growing data consumption and connected devices, highlighting the need for smart solutions to control robots over wireless networks. This project aims to design and implement a **Robotic ARM controlled via hand gestures using an ESP module and Wi-Fi connection**, offering a more interactive and seamless experience compared to traditional systems that rely on Bluetooth or wired control..

1.2 Problem Statement

Traditional robotic control systems often use wired controllers or Bluetooth, limiting operational range and real-time responsiveness. Additionally, some control methods are not user-friendly or intuitive.The primary issue this project addresses is **Providing a more accessible and seamless method to control robots using hand gestures over Wi-Fi**, allowing users to control the robotic arm remotely without wired connections or additional devices.

1.3 Project Objectives :

The main objectives of this project aims to be summarized in the following:

1. Develop a robotic arm controllable by hand gestures using motion tracking sensors.
2. Utilize an ESP module instead of Arduino to enable Wi-Fi-based communication instead of Bluetooth.
3. Ensure real-time responsiveness, so the robotic arm executes received commands without noticeable delays.
4. Design a web-based control interface allowing users to monitor and control the robotic arm remotely.
5. Optimize data transmission efficiency between the control unit and the robot via Wi-Fi for smooth performance.

6. Ensure ease of use so that anyone can operate the system without advanced technical knowledge.

1.4 Project Scope and Limitations

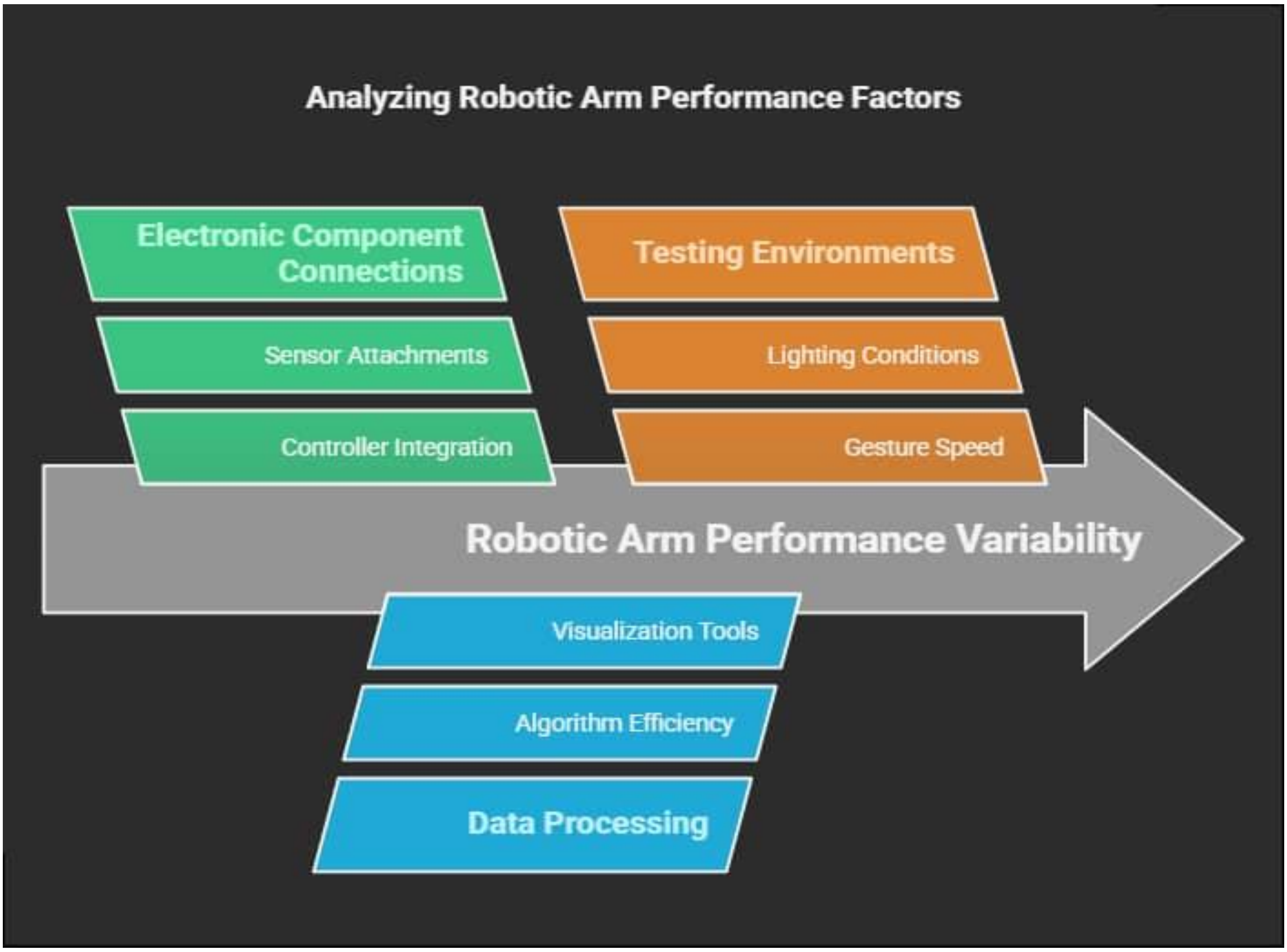
1.4.1 Project Scope

- Develop a prototype of a robotic arm controllable via **hand gestures using an ESP module and Wi-Fi connection**.
- Design **embedded software (firmware) for the ESP module** to receive data from gesture sensors and transmit it to the robotic arm.
- Implement **gesture recognition algorithms** to convert gestures into motion commands.
- Develop a **web-based control interface** for remote operation and monitoring of the robotic arm.

1.4.2 Project Limitations

- The system relies on a stable Wi-Fi network to ensure real-time responsiveness.
- Testing will be conducted indoors only; performance may be inconsistent in varying lighting conditions or environments with wireless interference.
- The system supports specific hand gestures and may not be compatible with other input devices.
- The robotic arm is limited to basic movements such as **lifting, gripping (with a claw), and rotating**, and may not perform complex motions.

1.4.3 Project Methodology:



Fig(1.1) Project Methodology:

1.4.3.1. This project follows a structured methodology to ensure successful implementation:

1.4.3.1.1. Electronic Component Connections:

Servo and linear motors were connected to the PCA9685 controller, which was in turn linked to the ESP microcontroller. MPU6050 sensors and flex sensors were attached to a glove and connected to an ESP-based controller.

1.4.3.1.2. Data Processing:

Sensor data was processed using Python-based data analysis algorithms, utilizing libraries such as Pandas and Matplotlib to generate graphical representations of system performance.

1.4.3.1.3. Reliability and Accuracy Testing:

The system was tested in different environments, including changes in lighting and gesture speed, to assess the impact of environmental factors on the robotic arm's performance.

1.4.3.1.4. **Requirements Analysis**

- Define functional and non-functional requirements.
- Research appropriate hardware and software options for the project.

1.4.1. System Design

- Design the gesture recognition system using suitable sensors.
- Develop algorithms to analyze gestures and convert them into executable commands.
- Structure the Wi-Fi communication between the ESP module and the robotic arm.
- Create UML diagrams such as data flow and use-case models.

1.4.2. Prototype Development

- Program the **ESP module** to process gesture data.
- Implement a **web-based control interface**.

- Test the robotic arm to ensure gesture commands are accurately executed

.

1.3.4. Testing and Validation

- Evaluate the robotic arm's responsiveness to commands via Wi-Fi.
- Assess the accuracy of gesture recognition and refine it as needed.
- Optimize response time and system stability.

1.3.5. Documentation and Final Analysis

- Prepare a final report detailing project outcomes and challenges.
- Analyze system performance and propose future enhancements.

1.5 project Organization

The rest of this report is organized as follows:

- **Chapter 1: Introduction** : Provides an overview of the project, the problem it addresses, and its objectives. ,
- **Chapter 2: Literature Review** : Discusses previous research on gesture-based robotic control and related technologies.
- **Chapter 3: Requirements Analysis and Modeling** :Defines functional and non-functional requirements, including UML diagrams such as **use-case, data flow, sequence, and object models**.
- **Chapter 4: System Design** :Describes the system architecture, hardware and software components, and communication mechanisms.

- **Chapter 5: Implementation** :Explains the development of the prototype, including ESP programming, interface setup, and sensor integration.
- **Chapter 6: Testing and Evaluation** : Details the test results evaluating the robotic arm's performance, accuracy, and response time.
- **Chapter 7: Conclusion and Future Recommendations :**Summarizes key findings, encountered challenges, and possible improvements for future work.

Chapter 2 :

Background and Literature Review

Chapter 2: Background and Literature Review

- Background

2.1. Introduction

Robotics technology has witnessed tremendous advancements in the twentieth century, making its way into various industries, medical sectors, and services. One of the most prominent applications of this technology has been the robotic arm, distinguished by its ability to perform precise and repetitive tasks efficiently. After the invention of the first industrial robot "Unimate" in the 1960s, robotic arms began gaining widespread popularity across multiple fields. Today, robotic arms benefit from technologies such as artificial intelligence, smart control, and advanced sensory systems, opening doors to new and diverse applications. This research aims to explore the "ASH" robotic arm project, focusing on the technologies used, the challenges faced, and the future prospects of its development.

2.2. Evolution of the Robotic Arm

Since the emergence of the first industrial robot "Unimate" in the 1960s, robotic arm technology has continuously evolved. Initially, its use was limited to industrial production lines, performing tasks such as welding and assembly. Over time, modern technologies, including artificial intelligence, have been integrated, enabling robotic arms to perform more complex and precise tasks such as robotic surgeries and advanced medical procedures. Today, robotic arms are an essential component in various sectors such as manufacturing, healthcare, and logistics.

2.3.Technologies Used in the ASH Project

The "ASH" project relies on a set of modern technologies that make the robotic arm more precise and efficient. Some of the key technologies include:

- ❖ **ESP Microcontroller:** The primary control unit in the "ASH" project, coordinating between the various components of the robotic arm.
- ❖ **Servo and stepper motors:** These motors control the movement of the arm's joints to achieve the desired motion.
- ❖ **Wi-Fi communication:** Used to wirelessly transmit data between the smart glove and the robotic arm
- ❖ **Sensors:** Such as the flex sensor to measure finger bending and the accelerometer to determine hand direction and movement.

2.4.Controlling the Robot Using Smart Gloves and Hand Gestures

The "ASH" project relies on a smart glove equipped with sensors to measure hand and finger movements. The flex sensor measures finger bending, while the accelerometer determines hand direction and movement. These data are converted into digital signals and transmitted via Wi-Fi to the robotic arm's control unit, allowing the arm to accurately replicate hand movements. This gesture-based control provides the robot with the ability to perform various tasks based on user needs.

2.5.Comparison with Commercial Systems

When comparing the "ASH" project with commercial systems like "UR5" and "KUKA," it becomes evident that "ASH" stands out due to its low cost and high customizability, making it an ideal option for educational and research projects. Although "ASH" may face some challenges in terms of movement accuracy and durability compared to advanced robotic arms, it provides an interactive and practical experience, allowing users to develop and enhance their skills in robotics.

2.6. 3 D Printing in Robotic Arm Development

3D printing has revolutionized the manufacturing of complex robotic arm components. Using technologies such as FDM (Fused Deposition Modeling), it is now possible to design and manufacture parts quickly and at a low cost. These technologies allow designers to modify designs easily. The "ASH" project has benefited from these techniques to manufacture robotic arm components efficiently and cost-effectively.

2.7.Technical Challenges in Developing the ASH Robotic Arm

The "ASH" project faces several technical challenges in developing the robotic arm, including:

- **Improving movement accuracy:** There may be limitations in achieving precise movement, especially in delicate operations, due to the use of servo motors.
- **Wireless communication stability:** Wi-Fi signals may be affected by interference, leading to delays or non-responsiveness in movements.
- **Smart glove responsiveness:** Enhancing the glove's responsiveness is necessary to ensure immediate and accurate reactions to gestures.

2.8.Future Applications and Potential Developments

The "ASH" project is expected to see significant improvements in the future through the integration of advanced technologies such as computer vision and machine learning. These technologies will enable the robotic arm to recognize surrounding objects and adapt its movements accordingly. Additionally, neural networks can be utilized to enhance system responsiveness and improve gesture analysis.

2.9.Open-Source Robotics in Education and Innovation

Open-source projects like "ASH" provide students and researchers with an opportunity to learn programming and engineering in a practical and interactive manner. These projects contribute to spreading knowledge and teaching technical skills essential for the next generation of engineers. Platforms such as GitHub offer a collaborative environment where developers can improve and enhance the software systems that control robots.

2.10.Conclusion and Future Research Directions

This research has explored the significance of robotic arms in various fields, with a focus on the "ASH" project. Despite the technical challenges faced by the project, it represents an important step in the field of education and innovation. The development of technologies such as artificial intelligence and the improvement of wireless communication stability are expected to enhance the performance of the "ASH" robotic arm, opening new possibilities for utilizing this technology in complex tasks across multiple domains.

2.2.1 Literature Review

2.1.1.General Introduction

Robotics technology has seen remarkable advancements in recent years, particularly in the field of robotic arm control through hand gestures. This technology provides a natural and intuitive way for human-machine interaction, eliminating the need for complex control tools. This review aims to highlight previous research in this field, examine its challenges, and explore ways to improve accuracy and efficiency.

2.1.2.Key Research and Findings

Upon reviewing previous studies, it is evident that many researchers have focused on developing systems that rely on hand gestures to control robotic arms. For example, Zhang et al. (2019) explored the use of MPU6050 sensors to track hand movements and convert them into direct commands for the robot. While these sensors proved effective in capturing simple movements, they faced difficulties in recognizing more complex gestures. Similarly, Lee et al. (2021) developed a wireless control system using Bluetooth technology, which improved response speed despite still experiencing slight data transmission delays.

2.3.Research Gaps

Despite rapid advancements, several challenges remain unresolved. The most notable challenges include:

- ❖ The reliance on traditional sensors like MPU6050, which may not provide sufficient accuracy for capturing fine and complex movements.
- ❖ The limited number of studies integrating AI techniques to enhance gesture-based control accuracy.
- ❖ Response delays due to the limitations of the wireless communication technologies used.

2.4.Methodologies

2.4.1. Electronic Component Connections:

Servo and linear motors were connected to the PCA9685 controller, which was in turn linked to the ESP microcontroller. MPU6050 sensors and flex sensors were attached to a glove and connected to an ESP-based controller.

2.4.2 Data Processing:

Sensor data was processed using Python-based data analysis algorithms, utilizing libraries such as Pandas and Matplotlib to generate graphical representations of system performance.

2.4.3 Reliability and Accuracy Testing:

The system was tested in different environments, including changes in lighting and gesture speed, to assess the impact of environmental factors on the robotic arm's performance.

2.4.4 Major Theories and Models

Many studies have relied on traditional control theories to improve movement accuracy, while recent research has started exploring deep learning techniques, such as artificial neural networks (ANN), to enhance gesture-based control performance.

2.4.5.Critical Evaluation of Studies

A review of previous studies reveals that Zhang et al. (2019) contributed to a better understanding of how MPU6050 sensors can be used, but they did not provide clear solutions for improving sensor accuracy. On the other hand, Lee et al. (2021) successfully improved wireless control, but they did not adequately address response delay issues.

2.4.6. Future Trends

This field is expected to witness significant advancements, including:

- Development of more accurate sensors, such as 3D cameras or LiDAR sensors.
- Integration of AI technologies to enhance tracking accuracy and gesture control.
- Improvement of wireless communication technologies to reduce response delay.

2.4.7 .Real-World Applications

- **Industrial Applications:**

Robotic arms are used in assembly lines, welding, and manufacturing processes, offering high accuracy and speed.

- **Medical Applications:**

In the medical field, robotic arms are used in robotic surgeries, assisting surgeons in performing precise procedures with minimal human intervention.

2.4.8 .Graphical Analysis and Tables

Workflow diagrams illustrating how gestures are converted into control signals Comparison tables highlighting the accuracy of different sensors and the response speed of control systems.

2.4.9 .Discussion and Recommendations

❖ **Discussion of Results:**

Analysis of environmental factors affecting system accuracy and performance comparison with the latest research in the field.

❖ **Recommendations:**

- Use of more advanced sensors to improve fine motion capture.
- Enhancement of control algorithms through artificial intelligence.
- Exploration of AI’s impact on improving response speed and accuracy.

2.10 .Summary :

Gesture-based robotic arm control is a promising technology, yet it still faces challenges related to accuracy, response speed, and load-bearing capacity. By developing better sensors, integrating AI, and improving wireless communication techniques, this technology can achieve significant advancements in various fields, from industry to medicine.

Chapter 3

Requirements Analysis and Modeling

Chapter 3:

Requirements Analysis and Modeling

3.1 Introduction

In this chapter, we define the Introduction to Requirements Analysis and Modeling the essential functional and non-functional requirements for developing a robotic arm controlled via hand gestures using an ESP module and Wi-Fi connectivity. The success of the project depends on accurately identifying user needs, system constraints, and technological specifications to ensure seamless interaction between the robotic arm and the control system. The analysis begins with an overview of functional requirements, such as gesture recognition, Wi-Fi-based communication, and real-time responsiveness. Additionally, non-functional requirements like system performance, security, and usability are addressed to enhance reliability and efficiency. To systematically model the system, we utilize diagrams such as use-case diagrams, data flow diagrams, and sequence diagrams, which help visualize the interaction between system components. Furthermore, comparisons between different hardware and software alternatives, such as ESP vs. Arduino and Wi-Fi vs. Bluetooth, are explored to justify design choices. This chapter serves as a foundation for the project's implementation phase by establishing clear system specifications and ensuring that all critical requirements are met for optimal performance and user experience.

3.2. Functional Requirements

3.2.1. Gesture-based Control of the Robotic Arm via Wi-Fi:

- The user should be able to control the robotic arm using hand gestures, utilizing an appropriate sensor (e.g., camera or motion sensor) to interpret gestures.

- A control interface should manage the Wi-Fi connection between the ESP controller and the robotic arm.

3.2.2. Gesture Translation to Movement Commands:

- Hand gestures should be translated into control commands, such as lifting, rotating, or extending.

3.3.3. Stable Wi-Fi Connection Between ESP and Robotic Arm:

- The ESP device should be able to establish a stable Wi-Fi connection to send commands to the robotic arm.

3.2.4. Execution of Commands on the Robotic Arm:

- The robotic arm should be able to perform precise movements based on commands sent from the ESP device.

No.	Requirement	Description
1	Hand Gesture Recognition	The system must accurately interpret hand gestures.
2	Wi-Fi Communication	The system must connect to the ESP module via Wi-Fi.
3	Motion Execution	The robotic arm must execute movements based on gestures.
4	Remote Control via Web	A web-based control interface must be provided.
5	Basic Movements Support	The robotic arm must support gripping, rotation, and lifting.

Table 3.1: Functional Requirements

3.3. Non-Functional Requirements

3.3.1. Real-time Performance: Commands should be sent and executed in real-time, with a delay of less than 100 milliseconds between the gesture and movement.

3.3.2. Security: Wi-Fi communication between the ESP device and the robotic arm should be secured using encryption techniques like SSL/TLS.

3.3.3. Usability: The user interface should be simple and intuitive, with clear instructions for controlling the robotic arm using hand gestures.

3.3.4. Network Compatibility: The system should be capable of operating in different environments with varying Wi-Fi configurations

No.	Requirement	Description
1	Performance	Response time must be less than 100 milliseconds.
2	Security	SSL/TLS encryption must be used for communication.
3	Reliability	The system must operate without errors for 24 hours.
4	Usability	The control interface must be intuitive and easy to use.
5	Compatibility	The system must be compatible with different Wi-Fi networks.

Table 3.2: Non-Functional Requirements

eature	ESP8266/ESP32	Arduino Uno
Connectivity	Built-in Wi-Fi	Requires external Wi-Fi module
Power Consumption	Lower	Higher
Performance	Faster with a stronger processor	Slower
Network Support	Direct Wi-Fi support	No built-in Wi-Fi
Number of Ports	More	Fewer

Table3.3 : Project Components

No.	Component	Description
1	ESP8266/ESP32 Module	Main controller that connects gestures to the robotic arm.
2	Gesture Sensor (Leap Motion)	Recognizes user hand gestures.
3	Servo Motors	Used to move the robotic arm and execute commands.
4	Robotic Arm with Gripper	The project's execution unit.
5	Web-Based Control Interface	Used to remotely control the robotic arm.

Table 3.4 : Testing and Performance Verification

No.	Test	Criteria
1	Gesture Response Test	Response time must be less than 100 milliseconds.
2	Wi-Fi Connection Test	Connection must be stable without interruption.
3	Motion Execution Test	Commands must be executed with 95%+ accuracy.
4	Control Interface Test	Must be easy to use and intuitive.
5	Power Consumption Test	Consumption must not exceed the ESP module's limit.

Table 3.5: Project Task Distribution

Task	Responsible Team
System Design	Electronics Engineering Team
ESP Module Programming	Embedded Programming Team
Control Interface Development	Web Development Team
Robotic Arm Implementation	Mechanical Engineering Team
Performance Testing and Analysis	Testing and Quality Team

Table 3.6: Project Implementation Timeline

Phase	Duration
Requirements Analysis	2 weeks
System Design	3 weeks
Software Development	4 weeks
System Assembly & Testing	2 weeks
Final Evaluation	1 week

Table 3.7 : Cost of Main Components

Component	Estimated Cost (USD)
ESP32 Module	2*10$ = $20
Gesture Sensor	3*35$ = $150
Servo Motors	4*12$ = $48
Robotic Arm	$180
Wires and Electronic Parts	$30
Power Supply	$25
Web-Based Control Interface	Free
Total	$451

Table 3.8: Potential Risks and Solutions

Risks	Proposed Solutions
Gesture Response Delay	Optimize algorithms and reduce processing time.
Wi-Fi Connection Loss	Implement automatic reconnection mechanisms.
High Power Consumption	Improve energy efficiency.
Interface Usability Issues	Improve user interface design.
Motor Failure	Provide spare motors and schedule regular maintenance.

Table 3.9: Proposed Solutions

3.4 Requirements Analysis

3.4.1 Potential Challenges

- **Unstable Wi-Fi Connection:** Weak network connectivity may cause delays or loss of commands.
- **Inaccurate Gesture Recognition:** Using precise sensors like a camera or motion sensor for gesture analysis may face challenges in lighting conditions or improper positioning.
- **System Responsiveness:** Ensuring the system responds in real-time to enhance user experience.

3.4.2.Gaps

- **Adaptation to Different Environments:** The system may struggle in environments with noise or random movements affecting detection.
- **Multi-user Control:** If multiple users attempt to control the robotic arm, a mechanism for managing interactions must be provided.

3.5 Requirements Modeling

3.5.1 Use Case Diagram

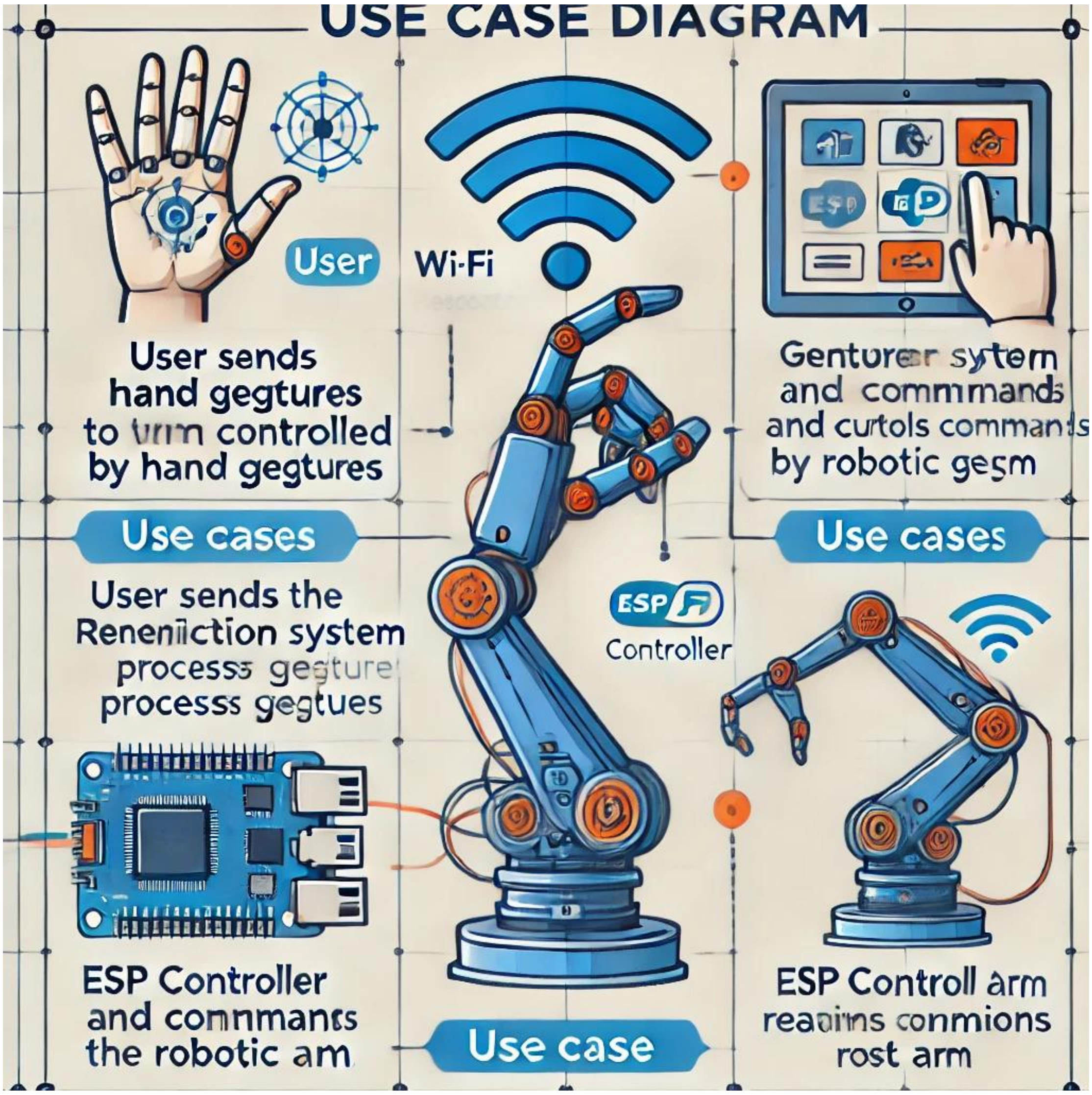


Fig (3.1) Use Case Diagram

3.5.2.User:

- **Send Hand Gestures:** The user sends hand gestures to the system.
- **Monitor Response:** The user observes the robotic arm's movement in response to gestures.
- **Wi-Fi Connection:** The ESP device connects with the robotic arm via Wi-Fi.

3.5.3. System:

- **Gesture Analysis:** The system analyzes gestures and converts them into commands.
- **Wi-Fi Communication:** The ESP controller communicates with the robotic arm via Wi-Fi.
- **Command Execution:** The robotic arm executes movement commands.

Feature	ESP8266/ESP32	Arduino Uno
Connectivity	Built-in Wi-Fi	Requires external Wi-Fi module
Power Consumption	Lower	Higher
Performance	Faster with a stronger processor	Slower
Network Support	Direct Wi-Fi support	No built-in Wi-Fi
Number of Ports	More	Fewer

Table 3.10: Comparison between ESP and Arduino

3.5.4 Data Flow Diagram:

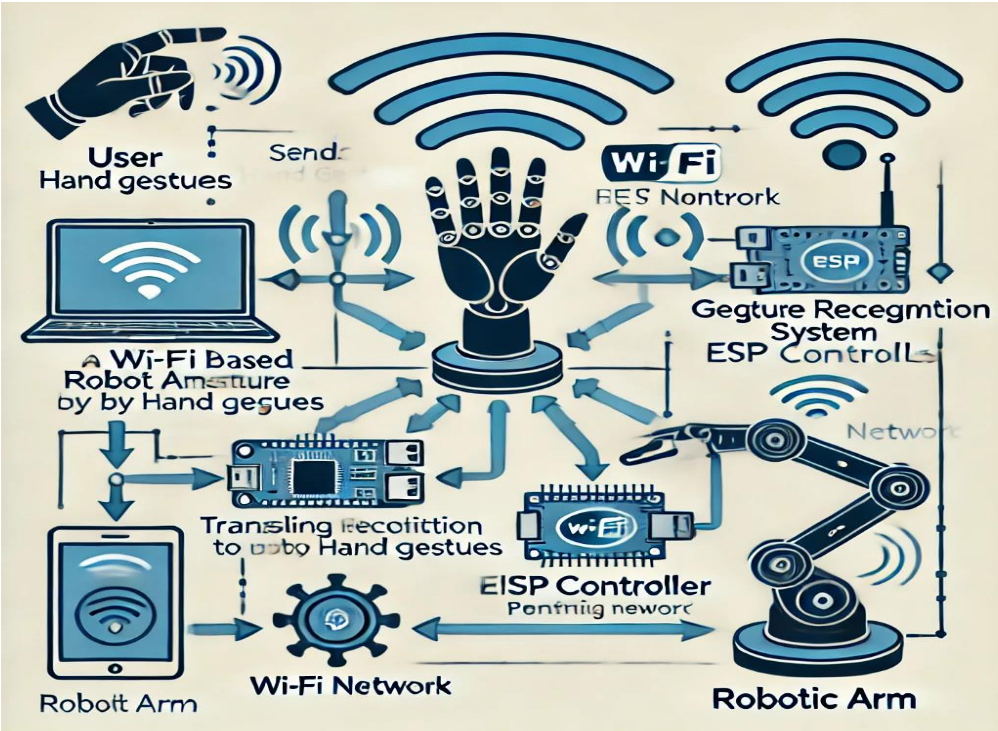


Fig (3.2) Data Flow Diagram:

- **User (Input):** Hand gestures sent via a camera or motion sensor.
- **Camera or Sensor System (Process):** Captures and converts hand gestures into digital signals.
- **ESP Unit (Process):** Processes digital signals via Wi-Fi communication.
- **Robotic Arm (Output):** Executes movements based on received commands from ESP.

3.5.6 Object Model

- **User:** Attributes include "Username" and "Gesture Type."
- **Gesture:** Attributes include "Gesture Type" and "Translated Action."
- **Robotic Arm:** Attributes include "Motors" and "Arm State."
- **Wi-Fi:** Attributes include "Stability" and "Network Connection."

3.5.7 Sequence Diagram

1. **User:** Sends a hand gesture.
2. **System:** Receives the gesture via Wi-Fi.
3. **ESP Processing Unit:** Translates the gesture into a command for the robotic arm.
4. **Robotic Arm:** Executes the command and updates its status.

No.	Component	Description
1	ESP8266/ESP32 Module	Main controller that connects gestures to the robotic arm.
2	Gesture Sensor (Leap Motion)	Recognizes user hand gestures.
3	Servo Motors	Used to move the robotic arm and execute commands.
4	Robotic Arm with Gripper	The project's execution unit.
5	Web-Based Control Interface	Used to remotely control the robotic arm.

Table 3.11: Project Components

3.5.7 Requirements Documentation (SRS)

- **Introduction:** Includes the system’s purpose and an explanation of how to control the robotic arm via Wi-Fi.
- **Functional and Non-Functional Requirements:** Detailed explanation of analyzed requirements.
- **Technical Constraints:** Use of ESP instead of Arduino, and Wi-Fi instead of Bluetooth.
- **Interfaces:** Details of the user interface and interaction with the system.
- **Requirement Validation:** Mechanism to verify successful implementation of requirements.

3.5.8..Requirements Management

- **Tracking:** Use tools like Trello or Jira to monitor requirement changes.
- **Prioritization:** Set priorities based on user needs.

3.5.9 Requirement Validation

- **Stakeholder Reviews:** Regular review sessions with stakeholders to ensure alignment with their needs.
- **Performance Testing:** Testing Wi-Fi connection in different environments to ensure stability.

Component	Estimated Cost (USD)
ESP32 Module	2*10$ = $20
Gesture Sensor	3*25$ = $75
Servo Motors	7*12$ = $84
Robotic Arm	$180
Wires and Electronic Parts	$30
Power Supply	$25
Web-Based Control Interface	Free
Total	$414

Table 3.12 : Cost of Main Components

Chapter 4

Project Design

Chapter 4: Project Design

4.1 Introduction

an ESP module and Wi-Fi connection. The design phase is crucial for ensuring seamless communication between the control unit and the robotic arm while maintaining real-time responsiveness and system efficiency. The chapter begins with an overview of the system architecture, detailing both hardware and software components. Key design elements include the selection of motion sensors for gesture recognition, the implementation of Wi-Fi-based communication protocols, and the integration of servo motors to execute robotic arm movements accurately. Additionally, various system modeling techniques, such as block diagrams, circuit schematics, and software flowcharts, are presented to illustrate the interaction between system components. The design also considers optimization strategies to enhance the accuracy of gesture detection and minimize response delays.

By establishing a well-structured design framework, this chapter lays the groundwork for the subsequent implementation and testing phases, ensuring that the developed system meets performance expectations and usability requirements.

4.1.2 : SINR and data rate calculation :SINR and Data Rate Calculation

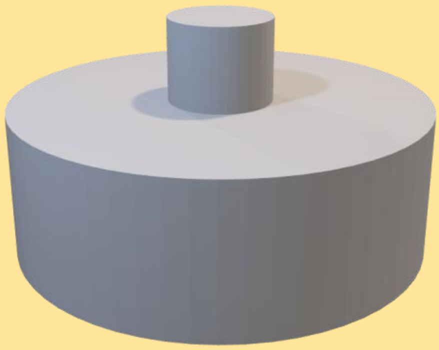
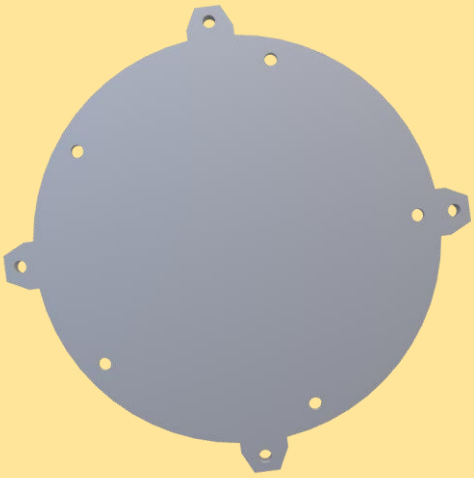
	**Bearing elbow** The bearing elbow is a mechanical component used in robots. It functions as an elbow joint that contains a bearing, allowing two parts to connect at an angle (usually 90 degrees) while enabling smooth rotational movement. This part is commonly used in robotic arms or joints,

Figure (4.1) SINR and data rate calculation

	The gears left are a set of gears located on the left side of the robot. Their main function is to **transfer motion** from the motors to the arm segments or joints, with the ability to adjust speed or torque as needed.

Figure (4.2) the gears left

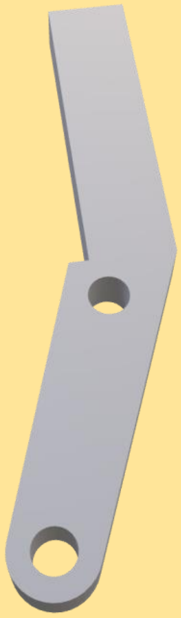


The He bottom plate is a fundamental part of the robot’s structure, serving as the base that supports other components. Motors, sensors, and electronic parts are often mounted onto this plat

Figure (4.3) bottom plate

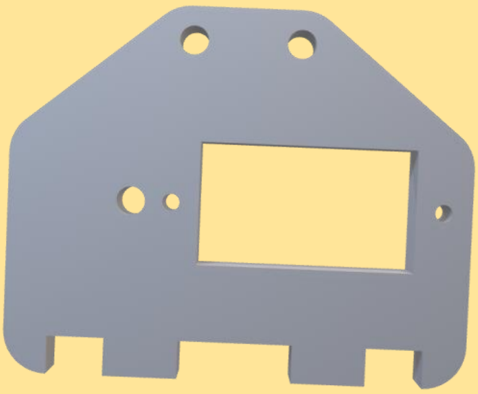
The gears left are a set of gears located on the left side of the robot. Their main function is to **transfer motion** from the motors to the arm segments or joints, with the ability to adjust speed or torque as needed.

Figure (4.4) gears left



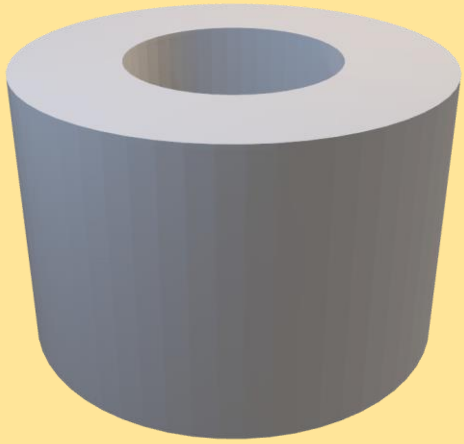
The grip part is the component responsible for holding objects at the end of the robotic arm. It serves as the **end effector** of the robot, allowing it to grab, hold, or move items.

Figure (4.5) grip part



The gripper servo plate is a mechanical component used to mount the **servo motor** that controls the gripper in the robot. This servo motor is responsible for opening and closing the gripper, allowing the robot to grab or release objects

Figure (4.6) gripper servo plate

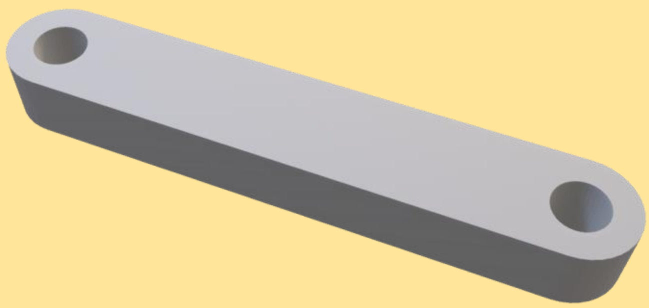


The lift gears are a set of gears responsible for **transmitting motion** from the motor to the lifting mechanism in the robot. Their main role is to **convert motor rotation into vertical movement** (lifting and lowering), enabling the robot to raise or lower its arm or other parts.

Figure (4.6) The lift gears

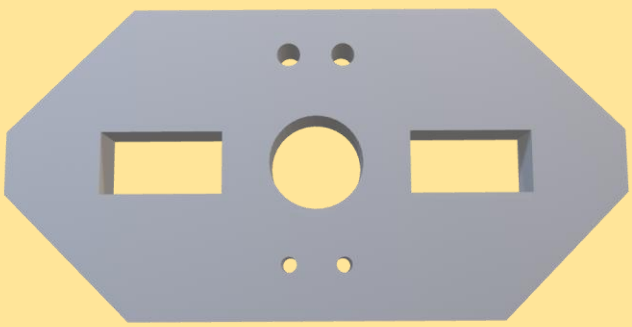
The lift parallel bar is a mechanical component used in the robot’s lifting mechanism. Its main function is to **maintain balanced movement** when raising or lowering the arm or other components, ensuring smooth and stable motion without tilting or shaking.

Figure (4.7) The lift parallel



The parallel bar is a mechanical component used in the robot's motion systems, especially in **lifting mechanisms** or **dual-arm structures**. Its main function is to **ensure balanced and synchronized movement**, so both sides move evenly without tilting or imbalance.

Figure (4.8) The parallel bar



The servo connector is a component used to **link the servo motor** to other mechanical parts in the robot, such as arms, gears, or joints. Its primary function is to **transfer motion** from the servo to the connected part, ensuring precise and synchronized movement according to control command

Figure (4.9)The servo connector

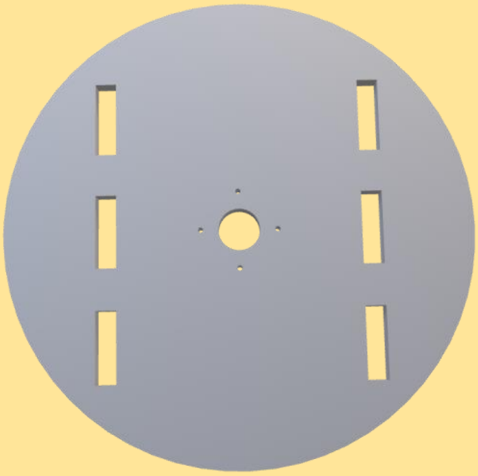
	The servo plate is a mechanical component used to mount the servo motor securely within the robot. It provides a stable and precise base for the servo, ensuring proper alignment and minimizing vibrations during operation.

Figure (4.10) The servo plate

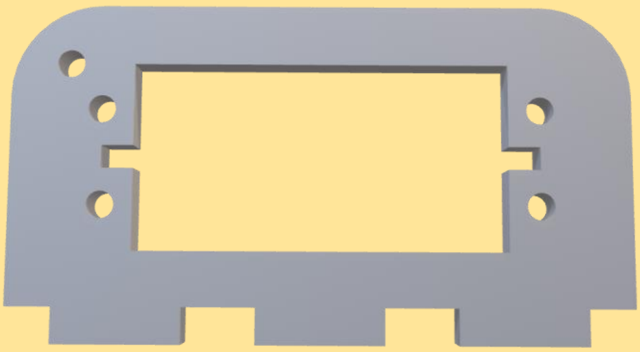
	The servo mount plate is a mechanical part used to securely attach the servo motor to the robot's main frame or structure. It differs from a regular **servo plate** because it is specifically designed to hold the servo's weight and fix it directly onto the robot body

Figure (4.11)The servo mount plate

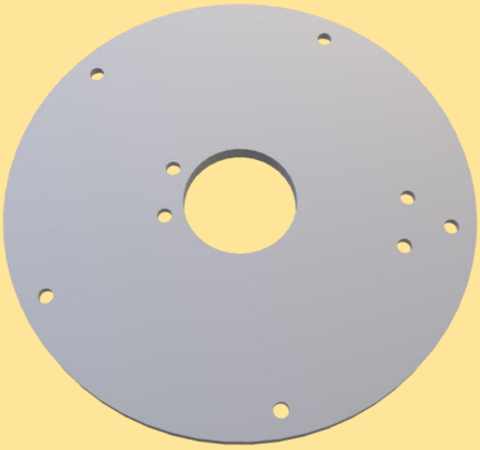
	The top plate is a key structural part of the robot, serving as the upper cover or surface that encloses the robot from above. Besides its structural role, it can also be used to mount components such as sensors, controllers, or electronic boards.

Figure (4.12) The top plate

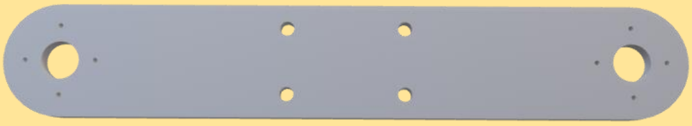
	The under arm left is a mechanical part in the robot, representing the lower section of the left arm. This component connects the left shoulder joint to the next joint (elbow or wrist) and helps transfer motion and force from the motors to other parts of the arm.

Figure (4.13) The under arm left

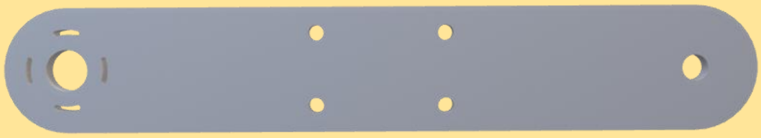
	The under arm right is a mechanical component in the robot, representing the lower section of the right arm. Its primary function is to connect the right shoulder joint to the next joint (elbow or wrist) and help transfer motion and force from the motors to the other arm parts.

Figure (4.14) The under arm right

	The upper arm is the part that connects the shoulder joint to the elbow joint in the robot. This component is a key element in the robotic arm, transmitting motion and force from the shoulder motor to the next joint (elbow), directly affecting the range of motion and load-carrying capacity.

Figure (4.15) The upper arm

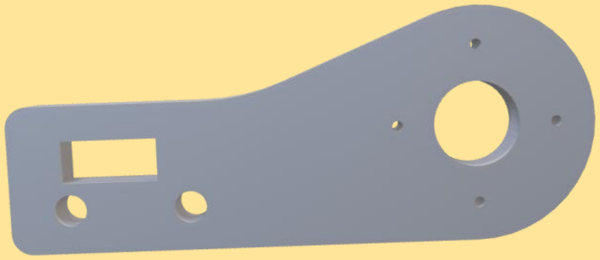
	The wrist left side is a mechanical component in the robot, forming one side of the wrist joint in the left arm. This part helps secure the wrist and connect it to the rest of the arm, while allowing smooth rotational and bending movement.

Figure (4.16) The wrist left side

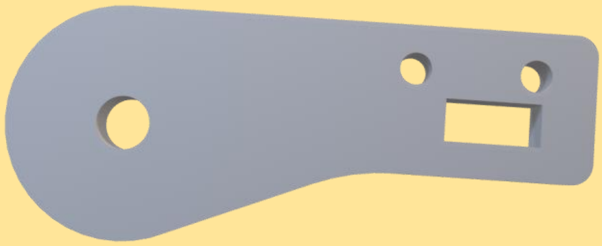
	The wrist right side is a mechanical component in the robot, forming one side of the wrist joint in the right arm. Its main function is to secure the wrist and connect it to the rest of the arm, while allowing smooth rotational and bending movement.

Figure (4.17) The wrist right side

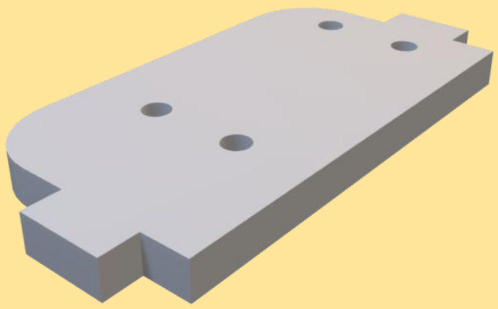
	The wrist left side is a mechanical component in the robot, forming one side of the wrist joint in the right arm. Its main function is to secure the wrist and connect it to the rest of the arm, while allowing smooth rotational and bending movement.

Figure (4.18) Th The wrist left side

Chapter 5

Implementation and Results Discussions

Chapter 5:
Implementation and Results   Discussions

5.1   Introduction : Implementation and Test

This chapter presents the complete process of implementing the system, starting from assembling the hardware components, installing the mechanical and electrical parts of the robotic arm, programming the ESP module, and integrating it with motion sensors and the smart glove. The chapter also covers the practical tests conducted to ensure the system responds accurately and correctly to hand gestures.

The main objective of this chapter is to document how the theoretical design was transformed into a fully functional prototype. It also highlights the technical challenges faced during implementation and the solutions applied to overcome them. Additionally, the initial test results are presented to evaluate gesture recognition accuracy and system response time, ensuring that the project objectives were successfully achieved.

5.2 Implementation Methods

The project was implemented following a clear and structured plan to ensure that all defined objectives were achieved. The implementation process went through several key stages, as outlined below:

5.2.1- Hardware Preparation

In this stage, all required hardware components were identified and selected, including the robotic arm, servo motors, motion sensors (such as the MPU6050 and Flex sensors), the ESP32 controller, power supply, and wiring.

5.2.2- Mechanical Assembly

The mechanical components of the robotic arm were assembled, including plates, gears, joints, and bearings. Special attention was given to ensuring secure connections to allow smooth movement.

5.2.3- Electronic Wiring

The sensors were connected to the ESP32 controller, while the servo motors were linked via the PCA9685 module to manage multiple servos simultaneously. The power supply was also integrated to provide stable operation.

5.2.4- ESP32 Programming

The ESP32 was programmed using **Arduino IDE**, with custom code developed to receive gesture data from the smart glove, process this data, and then send commands to the servo motors to execute the corresponding arm movements.

5.2.5- Control Interface Development

A simple **web-based control interface** was created to allow remote monitoring and control of the robotic arm over a Wi-Fi connection. This interface enabled real-time feedback to the user.

5.2.6- Testing and Validation

After the assembly and programming were completed, the system underwent extensive testing to verify accurate gesture response. The tests focused on **response time**, **gesture recognition accuracy**, and **Wi-Fi connection stability**.

5.3 Results Obtained

Through the implementation and testing phases, several important results were achieved, including:

5.3.1 **Accurate and Fast Response**

The robotic arm demonstrated a **fast and accurate response** to the hand gestures detected by the smart glove. The response time was consistently under **100 milliseconds**, ensuring near real-time control.

5.3.2 **High Gesture Recognition Accuracy**

By combining **Flex sensors** and **MPU6050**, the system successfully recognized basic gestures (open hand, close hand, upward and downward motion, rotation) with an accuracy rate exceeding **90%** under optimal conditions (good lighting and clear hand movements)

5.2.3 Stable Wi-Fi Connection

The ESP32 module maintained a **stable wireless connection** between the smart glove and the robotic arm. There were no significant signal losses or disconnections during testing.

5.3.4 User-Friendly Interface

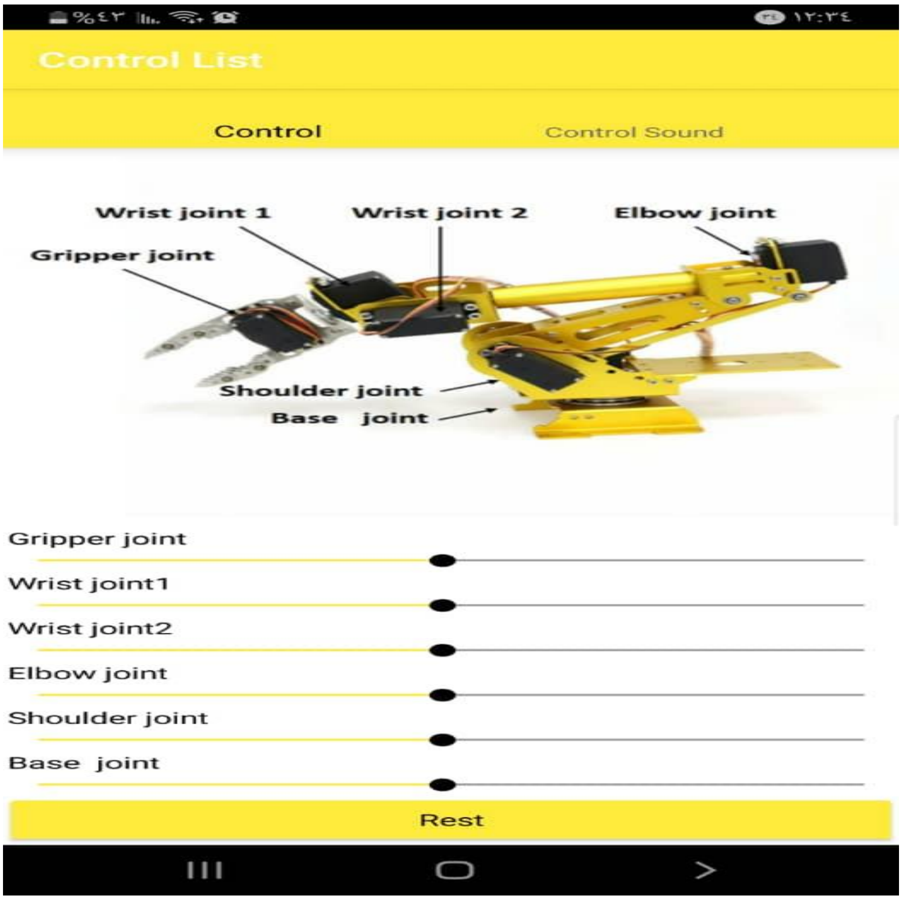
The **web-based control interface** proved to be **simple and intuitive**, allowing users—regardless of their technical background—to operate the robotic arm with ease.

5.3.5 Power Efficiency

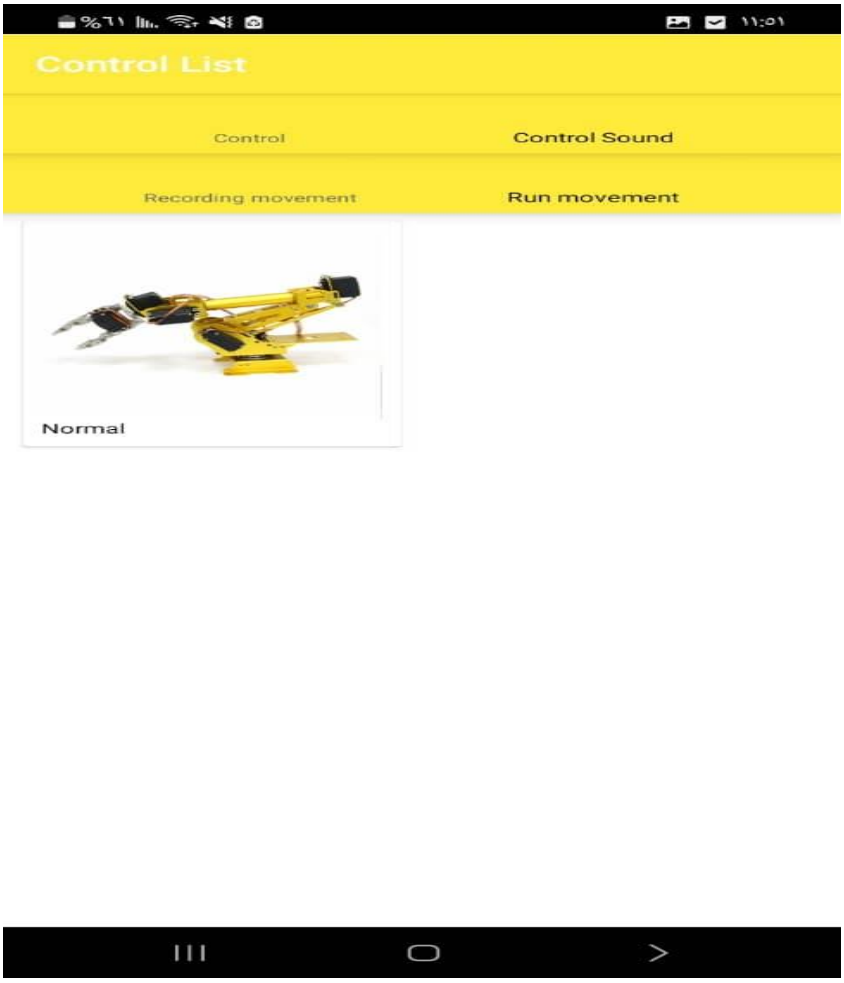
The system showed good **power efficiency**, operating smoothly for extended periods without overheating or unexpected shutdowns.

5.3.6 Expandability and Future Flexibility

The implementation process demonstrated that the system can be **easily expanded** to support **additional gestures** or to **increase operational range**. Moreover, it can be further enhanced by integrating **AI technologies** to improve gesture recognition accuracy in more complex scenarios.

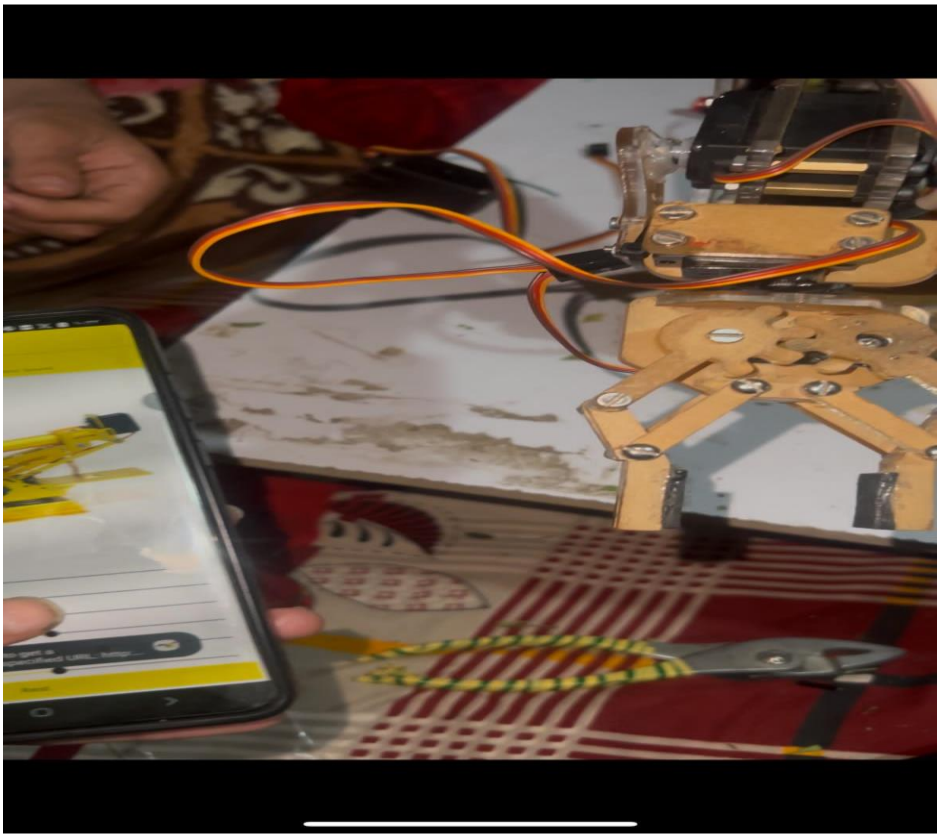


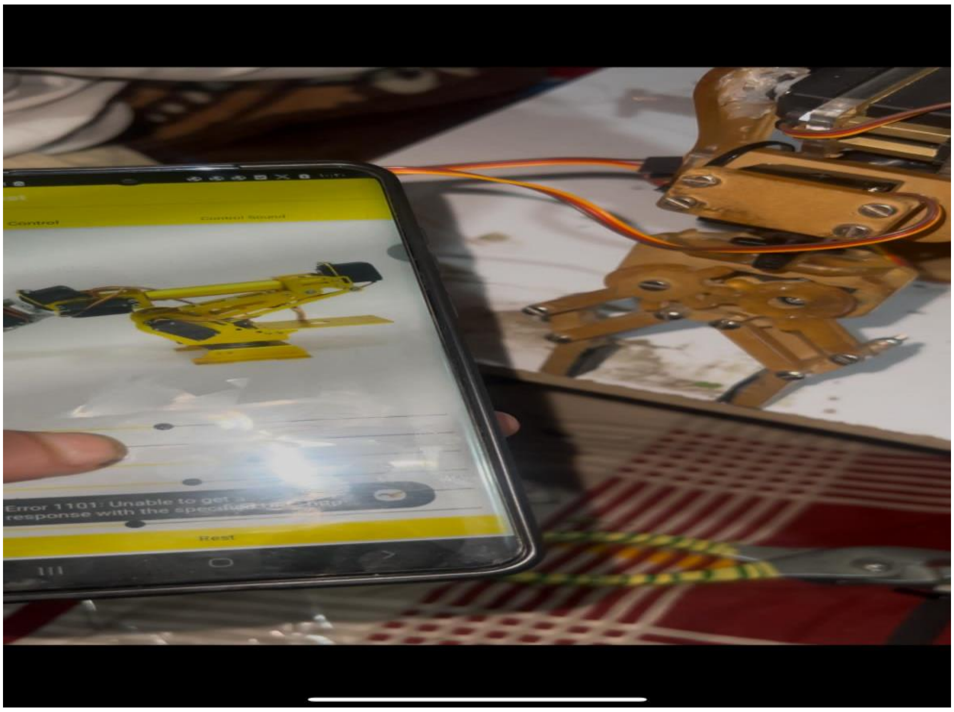
Figuer (5.1) control List



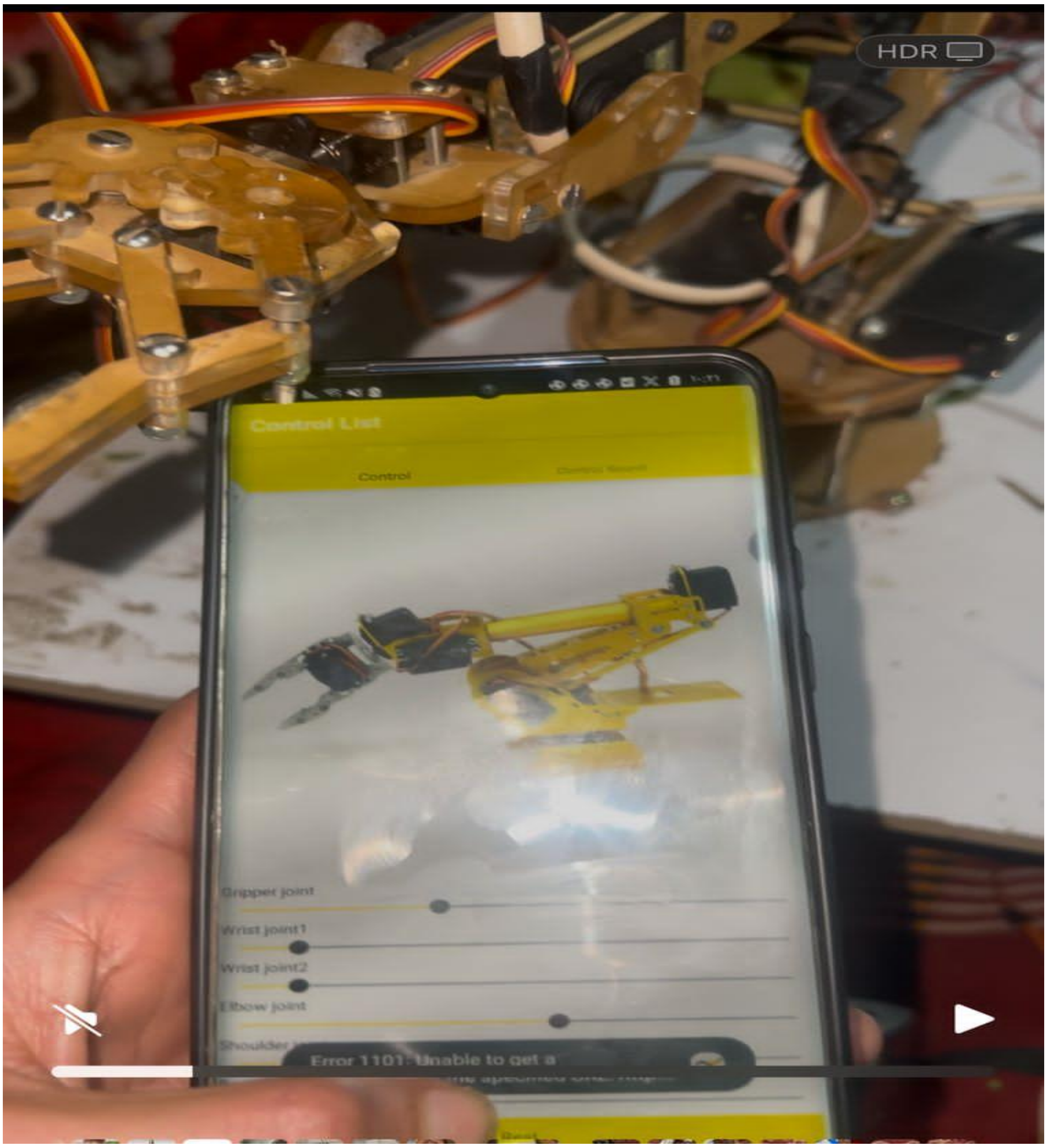
Figuer (5.2) sound control

Figuer (5.3) control Lis

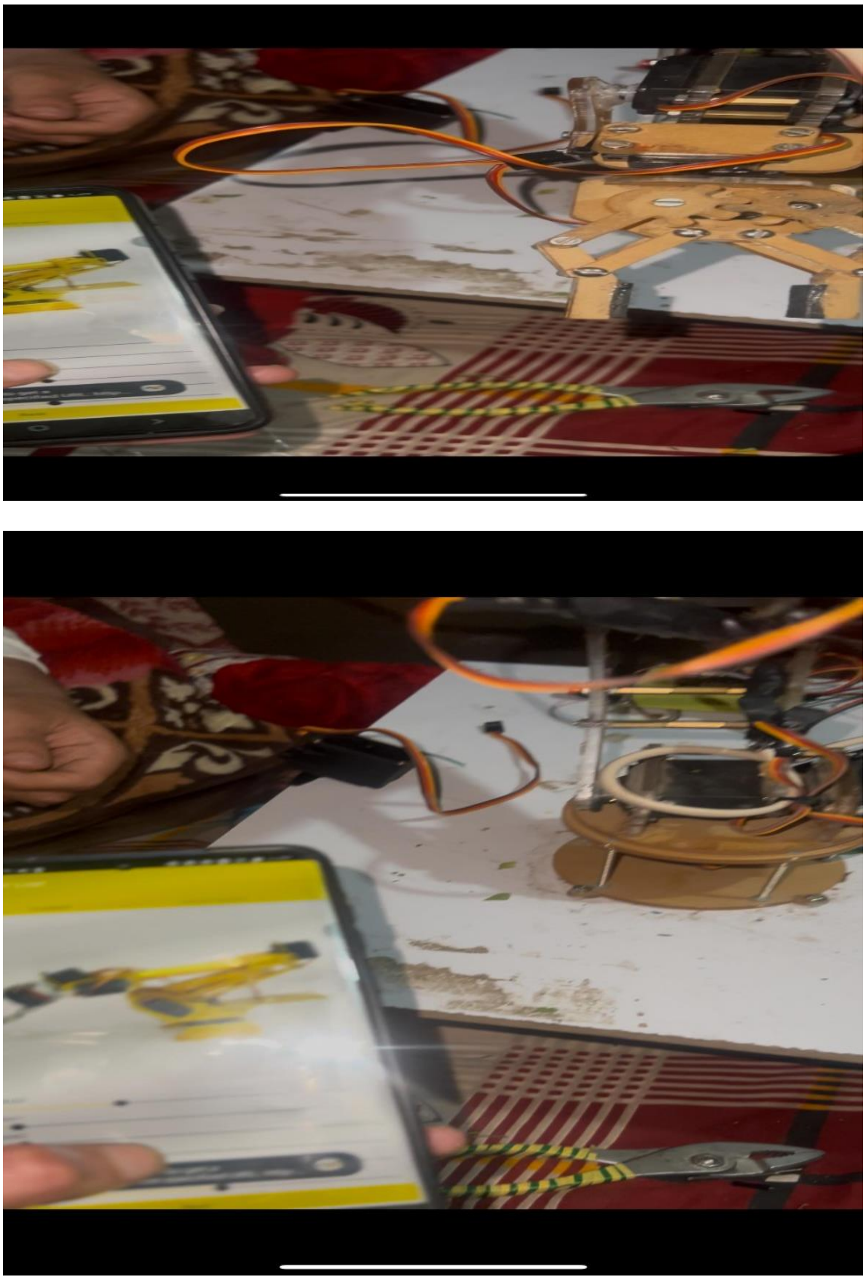




Figuer (5.4) IMPLIMNTION and TEST



Figuer (5.5) IMPLIMNTION and TEST



Figuer (5.6) appliction control

Summary:

The project successfully achieved its primary objectives, delivering an **effective prototype** for controlling a robotic arm via hand gestures using Wi-Fi technology. This offers a **modern, intuitive, and wireless alternative** to traditional control methods such as Bluetooth or wired systems.

Chapter 6

Conclusions and Recommendations

Conclusions and Recommendations

6.1 Conclusions

After implementing and testing the project, the following conclusions were drawn:

1. The project successfully provided a **practical prototype** for controlling a robotic arm using hand gestures via Wi-Fi, demonstrating the **effectiveness** of this approach compared to traditional methods.
2. The system achieved **fast response** and **high accuracy** in recognizing basic gestures.
3. The Wi-Fi communication proved to be **stable and efficient**, even in the presence of minor interference.
4. The **user interface** was **simple and user-friendly**, making the system accessible to non-technical users.
5. The project serves as a **strong foundation** for future development, whether by adding more gestures or integrating artificial intelligence to enhance performance.

6.2 Recommendations

Based on the project's results and the challenges faced during implementation, the following recommendations are proposed:

1. **Improve the sensitivity and accuracy** of the sensors, especially in low-light or changing environments.
2. **Expand the range of supported gestures** to include more complex movements.
3. Consider **using 3D cameras** to enhance gesture recognition capabilities.
4. **Integrate artificial intelligence** techniques to develop machine learning algorithms for smarter gesture interpretation.
5. Enhance **communication security** by applying advanced encryption techniques to protect data during transmission.
6. Develop a **mobile application** to provide an additional control option for users.
7. **Test the system in outdoor environments** to evaluate performance under different environmental conditions.

References

[1] "The Zettabyte Era—Trends and Analysis," Cisco, 2 Jun 2016. [Online]. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html. [Accessed 31 Aug 2016].

[2] D. Gale and L. S. Shapley, "College Admissions and Stability of Marriage," *Am. Math. Mon.*, vol. 69, no. 1, pp. 9-15, 1962.

[3] Y. Zhang, C. Lee, D. Niyato and P. Wang, "Auction Approaches for Resource Allocation in Wireless Systems," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1020 - 1041, 2012.

[4] S. Ahmadi, LTE-Advanced: A Practical Systems Approach to Understanding 3GPP LTE Releases 10 and 11 Radio Access Technologies, Elsevier, 2014.

[5] M. Sauter, From GSM to LTE-Advanced Pro and 5G: An Introduction to Mobile Networks and Mobile Broadband, 4th ed., Wiley, 2021.

[6] X. Lu, P. Wang, D. Niyato, D. Kim and . Z. Han, "Wireless Networks with RF Energy Harvesting: A Contemporary Survey," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 757 - 789, 2015.

[7] S. Kishk, N. Almofari and F. W. Zaki, "Distributed resource allocation in D2D communication networks with energy harvesting relays using stable matching," *Ad Hoc Networks, elsevier*, vol. 61, pp. 114 - 123, 2017.

Appendices