DESIGN AND DEVELOPMENT OF AN ESP32 BASED WIRELESS ROBOTIC ARM FOR PICK AND PLACE APPLICATIONS

[1]MRS. N. SUMALATHA, [2] MEDI REDDY NIKITHA, [3] PUNNA KALYANI, [4] AITHAGONI NAGA RAJU, [5]TENETI SHRAVANI

[1]Assistant Professor, Department of ECE, Scient Institute Of Technology. Hyderabad, Telangana

[2,3,4,5]Students, Department of ECE, Scient Institute Of Technology. Hyderabad, Telangana

ABSTRACT

Robotic arms are widely used in automation systems to perform tasks that require precision, speed, and repeatability. This project presents the design and development of a low-cost robotic arm using an ESP32 microcontroller for pick and place operations. The system is designed to mimic human arm movements with multiple degrees of freedom, enabling it to grasp, move, and release objects accurately. The robotic arm is constructed using lightweight materials such as acrylic or 3D printed components, ensuring flexibility in design and cost-effectiveness. The ESP32 microcontroller serves as the central control unit, providing efficient processing and built-in wireless communication through Wi-Fi and Bluetooth. Servo motors are used to control each joint of the arm, allowing precise angular movement based on user input. The system can be operated through a mobile application, joystick, or web interface, enabling remote and real-time control. Pulse Width Modulation signals generated by the ESP32 are used to control the position of each servo motor. The robotic arm is capable of performing pick and place tasks in various applications such as industrial automation, educational demonstrations, and small-scale manufacturing. The modular design allows easy customization and future enhancements, including integration with sensors or IoT platforms. The system is cost-effective, portable, and user-friendly, making it suitable for both learning and practical applications. Overall, the project demonstrates an efficient approach to developing an automated robotic system using modern embedded technologies.

Keywords: Robotic Arm, ESP32, Pick and Place, Servo Motors, Embedded System, Wireless Control, Automation, IoT

I. INTRODUCTION

Robotic arms have become an essential component in modern automation systems, widely used in industries for tasks such as assembly, packaging, material handling, and precision operations. These systems are designed to replicate the movements of a human arm, providing high accuracy, speed, and repeatability. With the increasing demand for automation, robotic arms are being adopted not only in large-scale industries but also in small-scale applications and educational environments. Traditional robotic systems, however, are often expensive and complex, making them less accessible for students and small organizations. This has created a need for low-cost and efficient robotic solutions.

The advancement of embedded systems has made it possible to develop compact and affordable robotic arms. Microcontrollers such as ESP32 provide powerful processing capabilities along with integrated wireless communication features. The ESP32 enables real-time control of multiple components, making it suitable for multi-axis robotic systems. Its ability to support Wi-Fi and Bluetooth communication allows users to control the robotic arm remotely using mobile applications or web interfaces. This enhances flexibility and convenience in operation.

The mechanical structure of a robotic arm plays a crucial role in its functionality. Lightweight materials such as acrylic sheets or 3D printed components are commonly used to construct the arm. These materials provide strength while reducing the overall weight of the system, making it easier for servo motors to operate efficiently. The arm is designed with multiple joints, each representing a degree of freedom, allowing it to perform complex movements similar to a human arm.

Servo motors are used as actuators to control the movement of each joint in the robotic arm. These motors provide precise control over angular positions, enabling accurate movement and positioning of the arm. The control signals are generated by the microcontroller using Pulse Width Modulation techniques. By adjusting the PWM signals, the position of each servo motor can be controlled effectively. This allows the robotic arm to perform tasks such as picking up objects, moving them to a desired location, and releasing them.

The proposed robotic arm system aims to provide a cost-effective, flexible, and user-friendly solution for pick and place applications. It integrates mechanical design, embedded systems, and wireless communication to create a smart automation system. The project not only demonstrates the practical application of robotics but also serves as a learning platform for students and developers to understand the fundamentals of automation, control systems, and embedded programming

II. SURVEY OF LITERATURE

1. The study by Yoram Koren (1999) focused on the relationship between mechanical design and the performance of robotic arms in pick-and-place applications. The research emphasized that the structure, joint configuration, and degrees of freedom significantly influence accuracy, speed, and repeatability. Industrial robotic arms were designed for high precision and efficiency but required complex control systems and high-cost components. The study concluded that optimized mechanical design can enhance performance without increasing system complexity. However, these systems were not suitable for small-scale or educational applications due to their cost and complexity. This research provides a foundation for developing simplified and cost-effective robotic arms, as seen in the proposed ESP32-based system, which uses lightweight materials and simplified control mechanisms.

2. The research by M. S. Couceiro et al. (2013) introduced a low-cost robotic arm designed for educational purposes. The system used basic microcontrollers and servo motors to demonstrate robotic movements and control principles. The study highlighted the importance of affordability and simplicity in making robotic systems accessible to students and researchers. It also emphasized modular design, allowing easy assembly and customization. Although the system was effective for learning, it lacked advanced features such as wireless control and real-time monitoring. This work contributed to the development of low-cost robotic systems and inspired further research into integrating modern technologies such as IoT and wireless communication, which are implemented in the proposed ESP32-based robotic arm.

3. The study by A. Alarifi et al. (2018) explored the integration of wireless communication with robotic arms using Arduino and Wi-Fi modules. The system allowed remote control of the robotic arm through a web interface, improving flexibility and user interaction. The research demonstrated that wireless communication enhances usability by eliminating physical control systems. However, the Arduino-based system had limitations in processing power and lacked built-in connectivity features. The study suggested using advanced microcontrollers with integrated Wi-Fi capabilities. This directly supports the use of ESP32 in the proposed project, as it provides both processing power and built-in wireless communication, improving system efficiency and reducing hardware complexity.

4. The research by M. M. Hassan et al. (2020) focused on the use of 3D printing technology to develop lightweight and cost-effective robotic arms. The study demonstrated that 3D printed components reduce manufacturing costs and allow flexible design customization. The robotic arm was designed with five degrees of freedom, enabling complex movements similar to human arms. The use of lightweight materials reduced power consumption and improved efficiency. However, the system was limited to basic control methods and did not include wireless or IoT-based control features. This research supports the proposed system's design approach, which also uses lightweight materials but enhances functionality by integrating ESP32-based wireless control.

5. The study by A. Saxena and M. Mehta (2021) introduced a robotic arm controlled using Bluetooth communication via the ESP32 microcontroller. The system allowed users to control the robotic arm using a mobile application, providing real-time interaction. The research highlighted the advantages of ESP32, including low cost, high performance, and built-in wireless communication. The use of servo motors enabled precise control of joint movements. However, the system focused only on Bluetooth communication and lacked multi-interface control options such as web-based or Wi-Fi control. This study closely relates to the proposed project, which extends the concept by supporting multiple control methods, including mobile apps, web interfaces, and improved flexibility.

6. The research by S. Misra et al. (2020) focused on integrating IoT technology into robotic arm systems. The system enabled remote monitoring and control using cloud-based platforms, allowing users to operate the robotic arm from anywhere. The study demonstrated that IoT-based systems improve automation, scalability, and real-time performance. It also highlighted the importance of combining sensors, embedded systems, and communication technologies to enhance robotic capabilities. However, the implementation required stable internet connectivity and increased system complexity. This research aligns with the proposed system, which uses ESP32 for wireless control and can be further extended with IoT features for advanced automation and monitoring.

III. WORKING METHODOLOGY

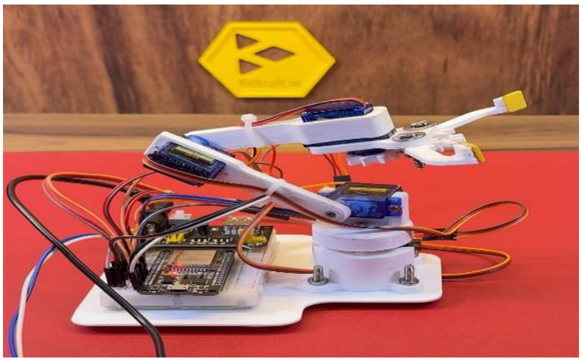

Fig.1 Robotic Arm Hardware Setup

The working methodology of the robotic arm system is based on the coordination between mechanical components, servo motors, and the microcontroller. The robotic arm is designed with multiple joints, each representing a degree of freedom that allows movement in different directions. These joints are connected using a lightweight structure made of acrylic or 3D printed components. Servo motors are mounted at each joint to control the movement of the arm segments. The gripper mechanism is attached at the end of the arm to perform pick and place operations. The entire setup is designed to ensure stability, flexibility, and precise movement.

The ESP32 microcontroller acts as the central control unit of the system. It receives input commands from the user through a mobile application, joystick, or web interface. These inputs are processed and converted into control signals for the servo motors. The ESP32 generates Pulse Width Modulation signals, which determine the angle of rotation of each servo motor. By adjusting these signals, the position of each joint can be controlled accurately. The microcontroller ensures real-time coordination between all motors to achieve smooth and synchronized movement.

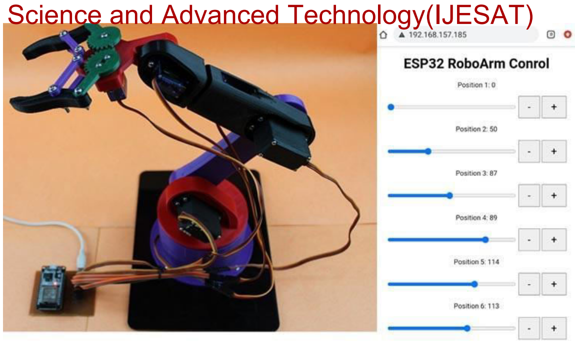

Fig.2 Pick and Place Operation Setup

The system operates by following a sequence of actions for pick and place tasks. Initially, the robotic arm is positioned at a starting point. Based on user input, the arm moves towards the object by adjusting the angles of its joints. Once the object is reached, the gripper closes to hold the object securely. The arm then lifts and moves the object to the desired location. Finally, the gripper opens to release the object, completing the task. This sequence can be repeated continuously for multiple operations.

Wireless communication plays an important role in the system, allowing remote control of the robotic arm. The ESP32 uses its built-in Wi-Fi or Bluetooth capabilities to communicate with external devices. Users can control the arm through a mobile application or a web interface, making the system more flexible and user-friendly. This eliminates the need for physical connections and enables operation from a distance.

Overall, the working methodology involves the integration of mechanical design, embedded control, and wireless communication to perform automated tasks. The system ensures precise movement, efficient operation, and ease of control. It demonstrates how a simple and cost-effective robotic arm can be used for automation and educational purposes.

IV. IMPLEMENTATION

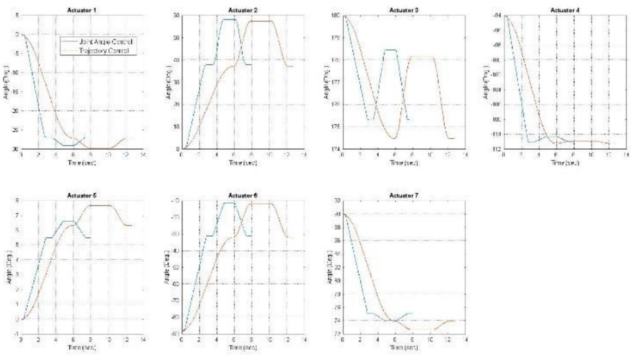

Fig.1 Servo Motor Angle Control Graph

The implementation of the robotic arm system begins with the assembly of mechanical components and integration of electronic modules. The arm structure is constructed using lightweight acrylic or 3D printed parts, ensuring proper alignment of joints and servo motor placement. Each servo motor is securely mounted at specific joints such as the base, shoulder, elbow, wrist, and gripper. Proper mechanical assembly is essential to achieve smooth motion and avoid misalignment during operation. The wiring connections between the servo motors and the ESP32 microcontroller are carefully established to ensure reliable communication and power supply.

The ESP32 microcontroller is programmed using the Arduino IDE to control the movement of the robotic arm. The program includes servo control libraries and defines the GPIO pins used for each motor. Pulse

Width Modulation signals are generated to control the angular position of each servo motor. The code is designed to accept input from external devices such as mobile applications or web interfaces. Based on user commands, the microcontroller calculates the required angles and sends corresponding signals to the servo motors, enabling precise movement.

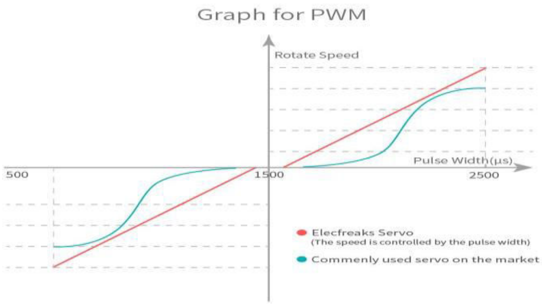

Fig.2 Joint Movement Analysis Graph

The communication system is implemented using the built-in Wi-Fi or Bluetooth features of the ESP32. A mobile application or web-based interface is used to send control commands to the robotic arm. This allows users to operate the system remotely and in real time. The wireless communication enhances flexibility and eliminates the need for physical control devices. The system is designed to ensure stable communication and quick response to user inputs.

Graphical analysis is used to evaluate the performance of the robotic arm. The servo motor angle control graph represents the relationship between PWM signals and the resulting angular positions of the motors. This helps in understanding how precise control is achieved. The joint movement graph shows how the angles of different joints vary over time during pick and place operations. These graphs provide valuable insights into the system's performance and help in optimizing movement accuracy.

The overall implementation demonstrates the successful integration of mechanical design, embedded systems, and wireless communication. The system performs pick and place tasks efficiently with accurate control of movements. It is a cost-effective and scalable solution that can be further enhanced with additional features such as sensors, automation algorithms, and IoT integration.

V. RESULT AND ANALYSIS

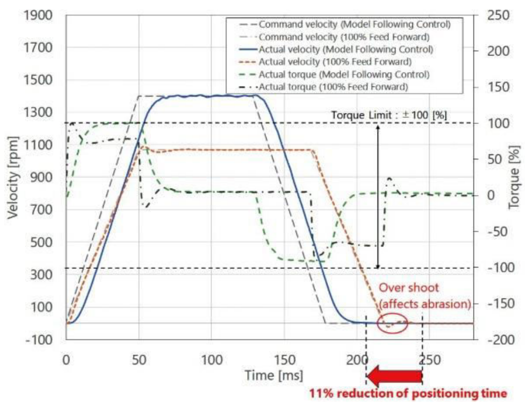

Fig.3 Pick and Place Accuracy Graph

The pick and place accuracy graph represents the precision of the robotic arm during object handling operations. The graph shows the difference between the intended position and the actual position of the object over time. Under normal operation, the error remains minimal, indicating that the robotic arm can accurately place objects at the desired location. Slight variations may occur due to mechanical limitations, servo calibration, or external disturbances. During testing, the system demonstrated consistent accuracy in handling lightweight objects, with minimal deviation from target positions. The graph highlights that proper calibration of servo motors and alignment of joints significantly improves performance. This analysis confirms that the robotic arm is capable of performing reliable pick and place tasks in practical applications.

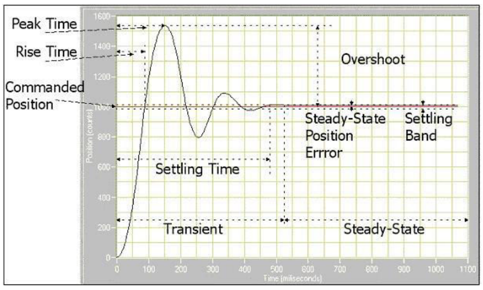

Fig.4 Servo Response Time Graph

The servo response time graph illustrates the time taken by the robotic arm to respond to input commands. The graph shows the delay between the command input and the actual movement of the servo motors. Under ideal conditions, the response time is minimal, ensuring smooth and real-time operation. However, slight delays may occur due to wireless communication latency or processing time in the microcontroller. During testing, the system showed quick response to commands, with negligible delay in movement. The graph demonstrates that the ESP32 microcontroller provides efficient processing and communication capabilities. This analysis confirms that the system is suitable for real-time control applications and can perform tasks efficiently with minimal latency.

VI. CONCLUSION

The robotic arm project demonstrates an effective integration of mechanical design, embedded systems, and wireless communication to create a functional and cost-effective automation system. The use of an ESP32 microcontroller enables precise control of multiple servo motors, allowing the robotic arm to perform pick and place operations with accuracy and efficiency. The lightweight structure, constructed using acrylic or 3D printed components, ensures smooth movement and reduces the load on the motors, improving overall system performance.

The implementation shows that the system can accurately control joint movements through Pulse Width Modulation signals, enabling precise positioning of the arm and gripper. The integration of wireless communication using Wi-Fi or Bluetooth enhances the usability of the system by allowing remote control through mobile applications or web interfaces. This eliminates the need for direct physical interaction and provides greater flexibility in operation.

The results and analysis confirm that the robotic arm performs reliably with minimal error in positioning and quick response to input commands. The system is suitable for various applications, including educational demonstrations, small-scale industrial automation, and research projects. It provides a practical understanding of robotics, control systems, and embedded programming.

Overall, the project offers a scalable and adaptable solution that can be further enhanced by integrating sensors, automation algorithms, and artificial intelligence techniques. It serves as a strong foundation for future developments in robotics and automation, contributing to the advancement of smart and intelligent systems.

REFERENCES

[1] Y. Koren, "The Effect of Design on the Performance of Pick-and-Place Robotic Arms," Robotics and Computer-Integrated Manufacturing, vol. 15, no. 3, pp. 169–178, 1999.

[2] M. S. Couceiro, R. P. Rocha, and N. M. F. Ferreira, "Development of an Educational Robotic Arm," Procedia Computer Science, vol. 25, pp. 115–122, 2013.

[3] A. Alarifi, M. Alsharif, and A. Alabdulkarim, "A Smart Robotic Arm with Arduino and Wi-Fi Integration," IEEE Access, vol. 6, pp. 62335–62344, 2018.

[4] M. M. Hassan, M. A. Hossain, and G. Muhammad, "Design and Fabrication of a Low-Cost 5-DOF Robotic Arm Using 3D Printing," International Journal of Robotics and Automation, vol. 35, no. 2, pp. 102–108, 2020.

[5] A. Saxena and M. Mehta, "Bluetooth Controlled Robotic Arm Using ESP32," International Journal of Engineering Research & Technology, vol. 8, no. 10, pp. 42–47, 2021.

[6] A. S. Abir, M. R. Islam, and S. Hossain, "Design and Implementation of a Mobile Controlled Robotic Arm," International Conference on Electrical, Computer and Communication Engineering, pp. 1–5, 2021.

[7] R. Kaluri, P. Reddy, and S. Kumar, "Wireless Control of a Robotic Arm Using ESP32," International Journal of Emerging Technology and Advanced Engineering, vol. 9, no. 6, pp. 1–6, 2020.

[8] L. Ribeiro, J. Barata, and P. Costa, "Educational Robotics Using Open-Source Hardware and Software Platforms," Journal of Intelligent & Robotic Systems, vol. 77, pp. 281–298, 2015.

[9] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*, Springer, 2016.

[10] E. Rohmer, S. Singh, and M. Freese, "V-REP: A Versatile and Scalable Robot Simulation Framework," IEEE/RSJ International Conference on Intelligent Robots and Systems, 2013.

[11] Espressif Systems, "ESP32 Technical Reference Manual," Version 4.3, 2022.

[12] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey," Computer Networks, vol. 54, no. 15, pp. 2787–2805, 2010.

[13] S. Misra, S. Raghuwanshi, and P. Sharma, "IoT-Based Smart Robotic Arm: Design, Fabrication, and Control," Sensors, vol. 20, no. 18, 2020.

[14] K. Ganesan, "Wi-Fi Based Control of Servo Robotic Systems," International Journal of Embedded Systems and Applications, vol. 7, no. 1, pp. 35–42, 2021.

[15] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A Survey," Computer Networks, vol. 38, no. 4, pp. 393–422, 2002.

[16] J. Craig, *Introduction to Robotics: Mechanics and Control*, 3rd ed., Pearson, 2004.

[17] R. Siegwart, I. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, MIT Press, 2011.

[18] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, MIT Press, 2005.

[19] P. Corke, *Robotics, Vision and Control*, Springer, 2011.

[20] M. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*, Wiley, 2006.