Development of a Wi-Fi Enabled Robotic Arm with Real-Time Web-Based Control

Mr. Ajit J. Sevara[1], Ms. Pari R. Bamania[2], Mr. Hiren M. Bhatt[3], Mr. Mohit A. Lakhwani[4]

[1,2]Student, B. Tech Information Technology, Atmiya University
[3,4]Assistant Professor, B. Tech Information Technology, Atmiya University

Abstract
This paper discusses the design and development of a low-cost six-degree-of-freedom (6-DOF) robotic arm named ArmX-6 that is designed by laser-cutting MDF sheets for the structures. Five MG996R servo motors are used for the base (rotate), shoulder, elbow, and wrist, and one MG90S servo motor is used to control the gripper. At the heart of the system is an ESP32 microcontroller, which allows for wireless control with a custom-built, real-time web interface built using HTML, CSS, and JavaScript. WebSocket is used for low-latency communication between the microcontroller and the web interface. Overall, the proposed solution shows promise for providing jitter-free and responsive motion control (without the need for high-frequency switching), which is an effective approach in terms of positioning motors for small-scale automation tasks, prototyping, and applications requiring cost-effective spatial manipulation.

Keywords: 6-DOF Robotic Arm, Wireless Communication, Web-based Control, Esp32

1. Introduction

Robotic arms are widely used in automation, education, and research because of its commerciality, accuracy and agility. Many off-the-shelf designs for robotic arms can be costly, restricting their use for students, researchers, and hobbyists. This gap can be filled by a low-cost implementation of a robotic arm controlled wirelessly providing a good alternative for multiple use cases.

This paper presents the design and development of ArmX-6, a six degrees-of-freedom robotic arm. The robotic arm has been constructed using laser cut MDF sheets for the structural components. Five MG996R servo motors are used for base rotation and movement at shoulder, elbow, and wrist respectively; one MG90S servo motor is used for gripper control. The ESP32 microcontroller is used as the core of this system, which wirelessly controls the system in real time through a web interface created.

The web interface will be in HTML, CSS, and JavaScript, which gives an interactive platform for users. To ensure low-latency and jitter-free communication, WebSocket technology is implemented for bidirectional data transfer between the microcontroller and the web interface. High-frequency switching shall not be needed by this approach so that smooth and responsive motion control can be achieved.

The objectives of this research include:

1. Building a cheap and modular 6-DOF robotic arm using easily accessible materials.
2. Setting up a real-time wireless control system with a web-based GUI.
3. Improving communication delay using WebSocket for effective data transfer.
4. Making an easy solution for DIY and educational robotics applications.

The remaining parts of this article are organized as follows: Section 2 undergoes related works in robotic control systems. Section 3 presents the methodology and describes system design together with hardware and software components. Section 4 discusses the results and evaluates the performance of the proposed system. Section 5 includes short biographies about the authors. Finally, section 6 provides the references quoted throughout the study.

2. Related Work

The development of robot arms is increasing due to advances in microcontrollers, such as ESP32, and the increase of online control technologies. Several remarkable projects illustrate these trends. Suhaeb and Risal (2024) developed an ESP32 -web server that allows real -time control and monitoring of robotic arms. It emphasizes the importance of light weight interfaces to provide user control [1]. Also Benitez et al. (2020) developed a robotic arm for remote education in the Covid-19 pandemic [2].

Whelan (2020) developed a 3D -printed robotic arm with servo motors such as MG996R and MG90S, making it accessible and easy to recreate for educational and prototype development purposes [3]. These works illustrate the practical in the combination of ESP32, Servo Motors and web technologies for user-interactive robot systems.

In addition, Brown (2019) underwent different control algorithms for robotic arms, emphasizing the importance of precise servo activation and accurate signaling to obtain responsive and reliable motion control [6]. Davis (2021) emphasized the importance of online interfaces in industrial automation, emphasizing their scalability, flexibility and efficiency for external robot operations [7].

Based on these works, the current project ARMX-6, a robot arm with 6 degrees of freedom (6-DOF), web socket communication and a responsive graphic user network interface presents. This includes real -time interaction with the robot arm and increases the degree of freedom compared to previous systems.

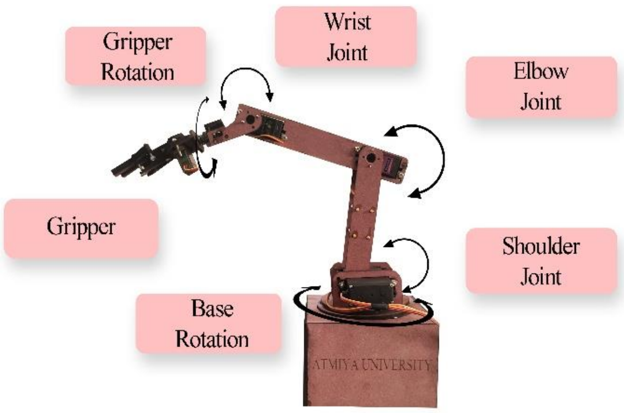
3. Methodology

The robotic arm was developed using a combination of Computer-Aided Design (CAD) for designing, laser cutting, microcontrollers programming, and web development technique.

3.1 Design and Fabrication

The mechanical design of robotic arm was created using AutoCAD software. An arm consists of several components, shoulder to elbow joint of 15 cm, elbow to wrist joint of 12 cm, and wrist to gripper assembly of 3 cm. The iris made of Medium Density Fiberboard (MDF) because it offers the best value and strength. Laser cutting was used to cut MDF sheets to the exact dimensions specified in the CAD designs. The base, shoulder, and elbow joints were equipped with MG996R servo motors to supply the required torque to lift and align. More precise control was needed for the rotation and gripping actions therefore Miniature MG90S servo motors were fitted to the gripper. The robotic arm with labelled components is shown in Figure 1.

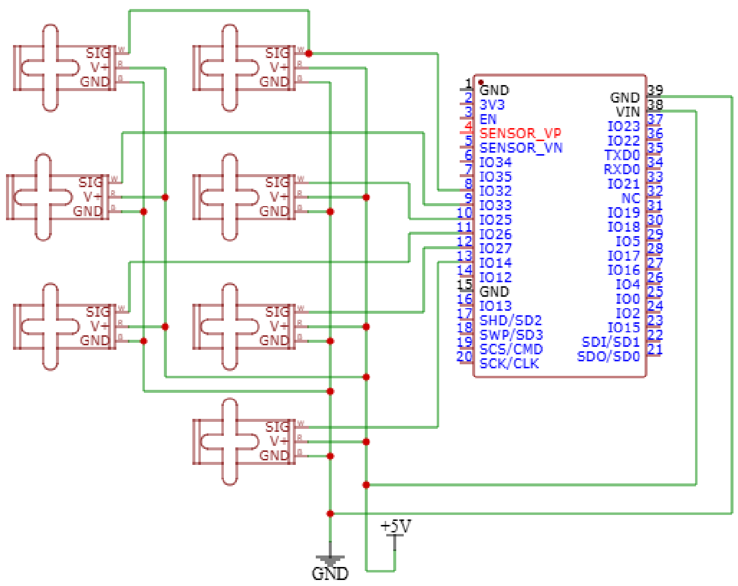
Figure 1: Robotic Arm with Labeled Components



3.2 Electronics and Control

The robotic arm is managed with the help of an ESP32 microcontroller, which provides adequate processing energy for wireless fidelity (Wi-Fi) connectivity and real-time control. 5V 10A external power delivery was used to ensure strong power distribution to Sarvo Motors (Garcia, 2020) [8]. ESP32 programmed the use of the Arduino Integrated Development Environment (IDE), which takes advantage of its large libraries for manipulation and WebSocket interactions in the servos (Li et al., 2022) [4]. Code turned into writing to deal with consumer input to operate the angles of the joint. The circuit diagram of a control system is shown in Figure 2.

Figure 2: Circuit Diagram of Control System



3.3 Web Interface and Communication

A web-based interface was developed to provide remote control and monitoring of the robotic arm using a Hypertext Markup Language (HTML), Cascading Style Sheet (CSS), and JavaScript. WebSocket API was used to facilitate bipartite and real -time communication between browser (client) and ESP32 Microcontroller (Server). The user interface includes sliders to control the angles of six joints of the robotic arm, as well as buttons for actions such as opening or closing the gripper. Users are provided with real-time visual response, allowing them to look at changes (Kim et al., 2023) [5].

4. Results

This section presents the experimental results obtained from testing the ArmX-6 robotic arm. The key performance metrics evaluated include accuracy and repeatability.

4.1 Accuracy Testing

The accuracy of the robotic arm changed into assessed with the aid of measuring its ability to attain predefined target factors within its workspace. A series of 20 randomly selected factors were programmed, and the arm was advised to move to each point ten instances. The real function of the quit effector was measured, and the mistake between the commanded and real positions become calculated. The average accuracy achieved became 2 mm with a preferred deviation of 0.5 mm.

4.2 Web Interface

The web interface response capacity was measured by recording time delay between a user input (for example, moving sliding control) and the corresponding movement of the robotic arm. The average response time was 150 milliseconds, which indicates real -time control features. The tests were performed using Google Chrome and Mozilla Firefox browsers.

Table 1: Accuracy Results

Target Point	Average Error(mm)	Standard Deviation(mm)
1	1.8	0.4
2	2.1	0.6
3	1.9	0.5
4	2.2	0.5
5	2.0	0.4

5. References

1. Suhaeb S., Risal A., "Implementation of ESP32-Based Web Host For Control and Monitoring of Robotic Arm", Journal of Embedded Systems, Security and Intelligent Systems, 2024, 5 (3), 249–254.
2. Benitez V.H., Mendoza J.A., Rosas J.M., Tlelo-Cuautle E., "Design of an affordable IoT open-source robot arm for online teaching of robotics courses during the pandemic contingency", HardwareX, 2020, 8, e00158. https://doi.org/10.1016/j.ohx.2020.e00158.
3. Whelan J., "3D Printed Robotic Arm", Instructables, 2020. https://www.instructables.com/3D-Printed-Robotic-Arm-2/
4. Li M., Chen L., Zhao K., Wang Q., "Development of a Low-Cost Open-Source Robotic Arm Based on Microcontroller", International Journal of Robotics and Automation, 2022, 15 (2), 78–85.
5. Kim D., Park S., Lee J., "A Web-Based Interface for Remote Control of a Robotic Arm", Journal of Internet Technology, 2023, 24 (1), 112–120.
6. Brown L.K., "A Review of Control Algorithms for Robotic Arms", Journal of Robotics, 2019, 36 (4), 221–235.
7. Davis P.R., "Web-Based Control Systems for Industrial Automation", International Journal of Automation and Control, 2021, 18 (2), 89–102.

8. Garcia A.M., "The Use of Low-Cost Microcontrollers in Robotics Applications", Journal of Microcontroller Engineering, 2020, 42 (1), 45–58.