Surface EMG Based Hand Gesture Signal Classification Using CNN for Control Of Software Robot

Mid-Review 3

GITAM (Deemed-to-be) University

AY 2021-25

Major Project
Project ID: Cs-6

Department of Electrical Electronics and Communication Engineering

Project Team:
- Avula Veera Siva Reddy
- Boya Rajesh
- Bhanu Siva Sai Kumar M

Project Mentor:
- Dr. Jaya Prakash Sahoo

Project In-charge:
- Dr. Ambar Bajpai

Objective and Goals

Objective

- To develop a hand gesture recognition model using surface EMG signals to control a software designed robot.
- Hand gestures through surface electromyography (sEMG) signals, which are then converted into commands for the control of robots.

Goals

Main Goals :

1. **Develop a Robot Gesture Classification System**:
    1. Capture and process Surface EMG signals to classify hand gestures with high accuracy.
    2. Implement signal processing techniques to filter noise and extract meaningful features from EMG data.
2. **Control a Software Robot Using Gestures**:
    1. Translate classified gestures into robot movements (e.g., Forward, Backward, Left, Right).
    2. Ensure real-time and reliable control of the robot via Wi-Fi communication

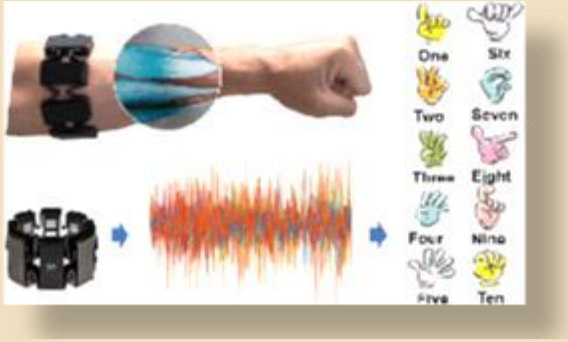
Abstract:

The development of a software robot controlled by a hand gesture recognition model is a modern approach to human-computer interaction. This project applies surface electromyography signals to interpret hand gestures, which are then translated into commands for robot control. The system is developed to perform real-time gesture-based control. The Main features include the seamless integration of the gesture recognition model with the logic of controlling the robot, thus achieving precise and responsive movements, such as moving forward, turning, stopping, or performing task-specific actions. Improving recognition accuracy and robustness under various conditions.

This project not only showcases the feasibility of sEMG-based control systems but also contributes to the rapidly increasing demand for adaptive and intelligent robotic solutions that could significantly enhance user experience and accessibility in real-world situations

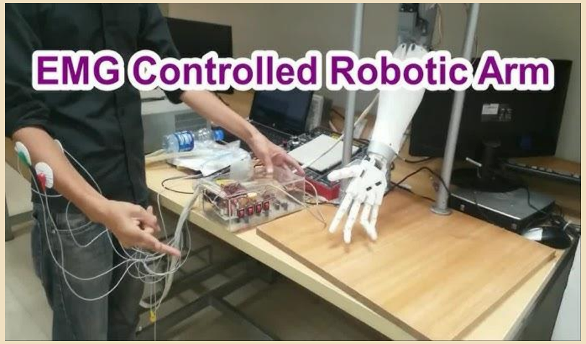
Applications of sEMG

Human machine interaction



Sign language recognition

Applications



EMG Controlled Robotic Arm

Virtual Reality (VR) and Gaming

Literature Survey (Improved post minor project)

Key Publications

1. **Paper 1**: "Surface EMG-Based Hand Gesture Recognition Using Machine Learning" – Focuses on ML algorithms for gesture classification.
2. **Paper 2**: "Real-Time Control of Robotic Systems Using EMG Signals" – Explores real-time robot control via EMG.
3. **Paper 3**: "Noise Reduction Techniques in EMG Signal Processing" – Discusses filtering methods for accurate signal analysis.
4. **Paper 4**: "Wireless Communication for IoT-Based Robotic Systems" – Covers Wi-Fi and IoT integration in robotics.

Key Resources – Whitepaper| Application Notes | Datasheet| Others

- **Whitepapers**:
  - "EMG Signal Processing for Gesture Recognition" .
  - "Wi-Fi Communication Protocols for IoT Devices"
- **Application Notes**:
  - MPU6050 Datasheet and Application Guide.
  - ESP8266 Wi-Fi Module Configuration Notes.
- **Datasheets**:
  - EMG Sensor Datasheet.
  - Motor Driver IC Datasheet.
- **Others**:
  - Online tutorials and forums for ESP8266 and MPU6050

Existing Implementations:

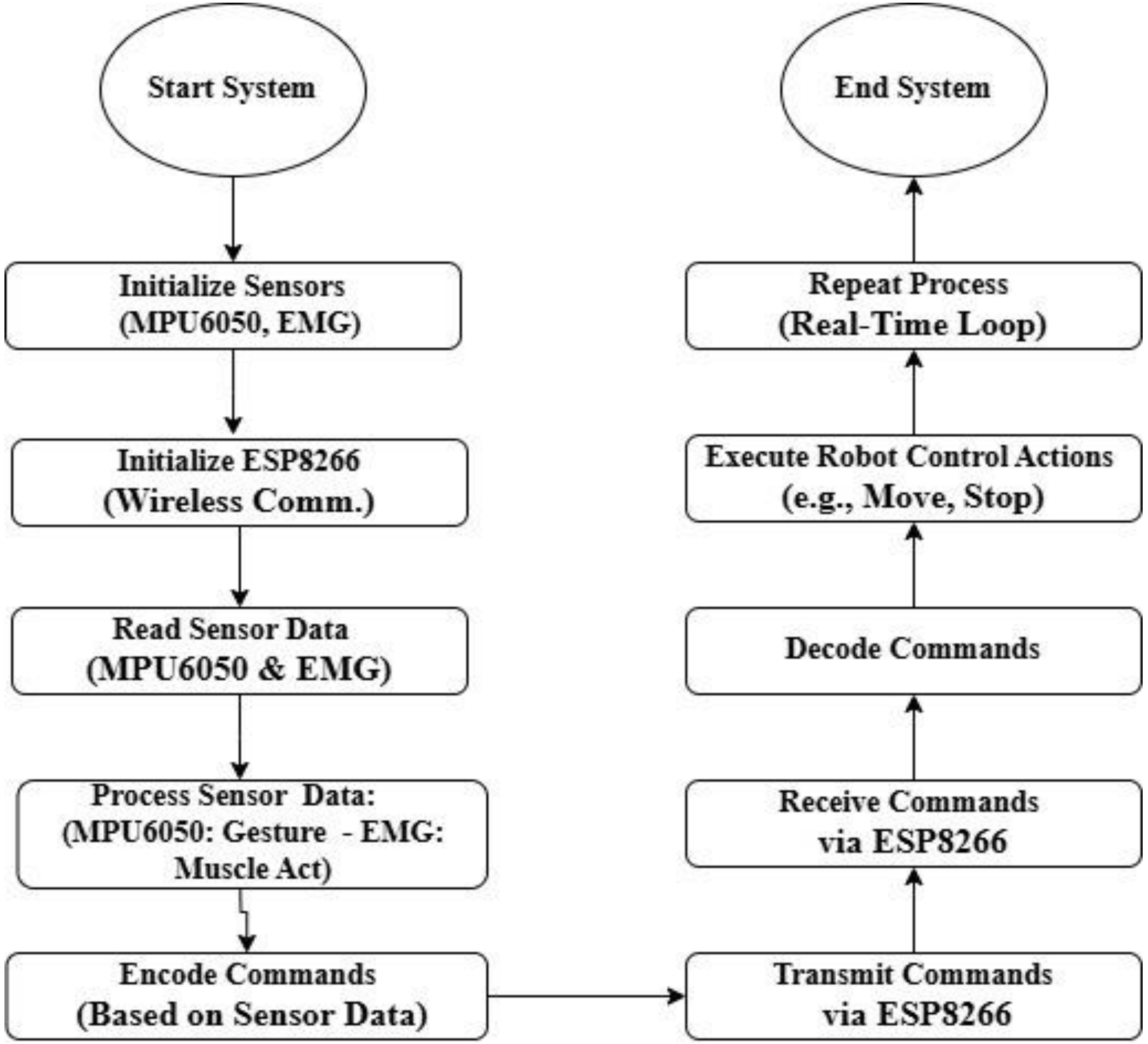
- **Products**:
  - Myo Armband (Thalmic Labs) – EMG-based gesture control.
  - OpenBCI – Open-source EMG and EEG systems.
- **Open Source**:
  - GitHub Repo: "EMG-Based-Robot-Control" – Python and Arduino-based implementation.
  - GitHub Repo: "Gesture-Recognition-Using-MPU6050" – MPU6050 gesture detection.
- **GitHub Projects**:
  - "ESP8266-Robot-Control" – Wi-Fi-based robot control.
  - "EMG-Signal-Processing" – EMG signal filtering and classification.

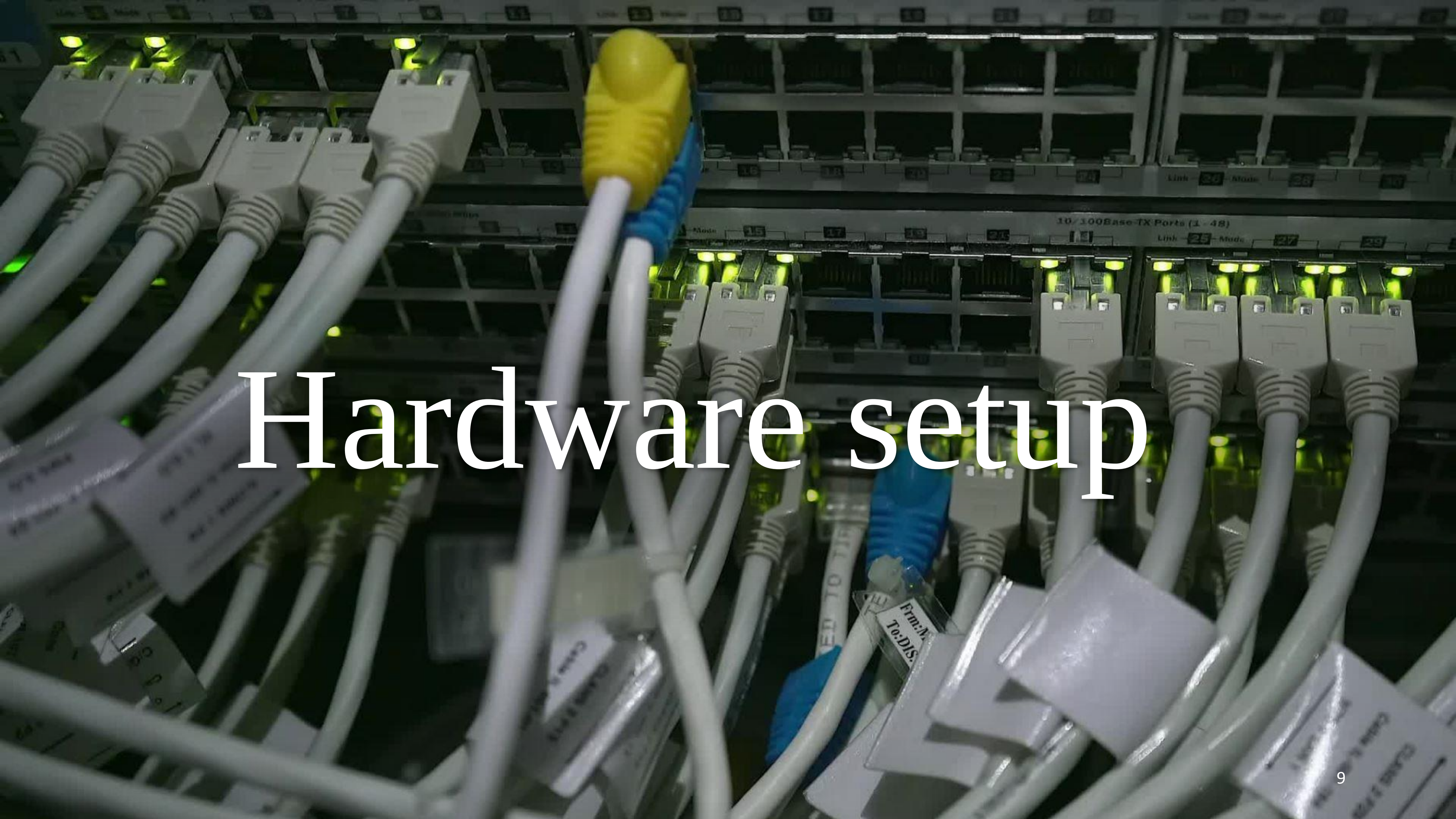
Key Resources

- **Whitepapers:**
  - "EMG Signal Processing for Gesture Recognition" .
  - "Wi-Fi Communication Protocols for IoT Devices" .
- **Application Notes:**
  - MPU6050 Datasheet and Application Guide.
  - ESP8266 Wi-Fi Module Configuration Notes.

Project Plan

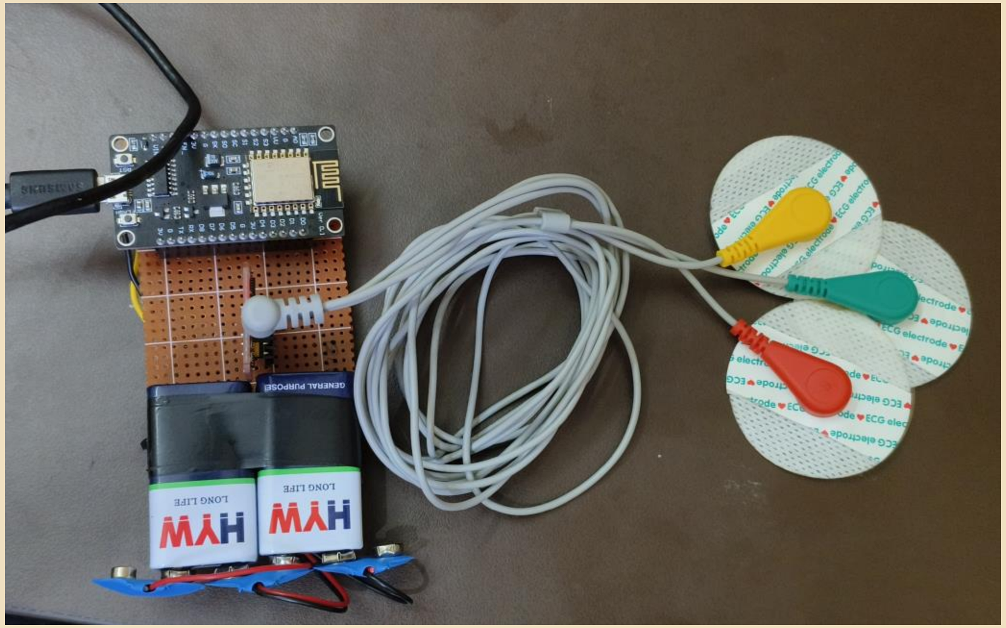
Block Diagram



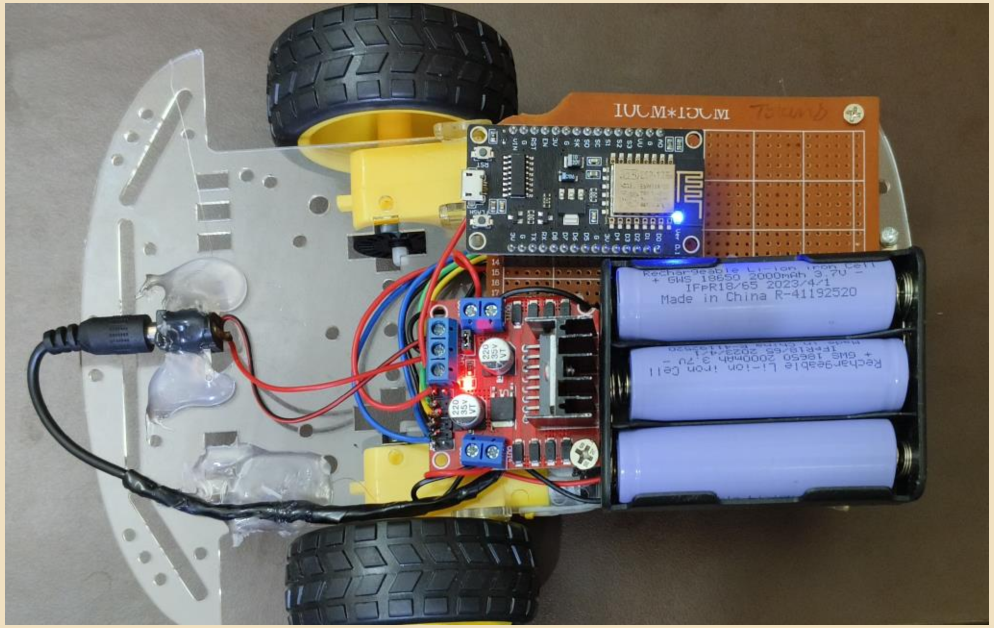


Hardware setup

Transmitter Setup

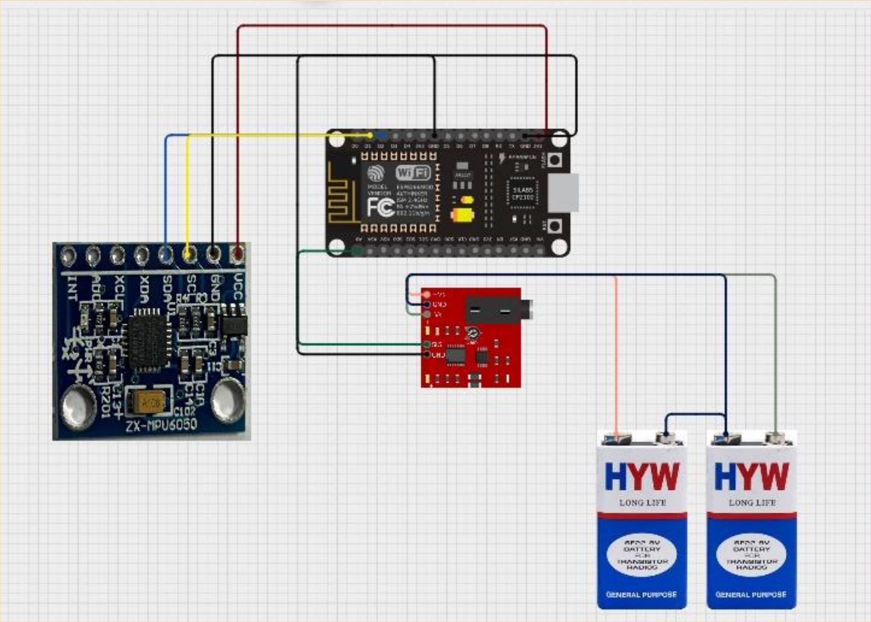


Receiver Setup

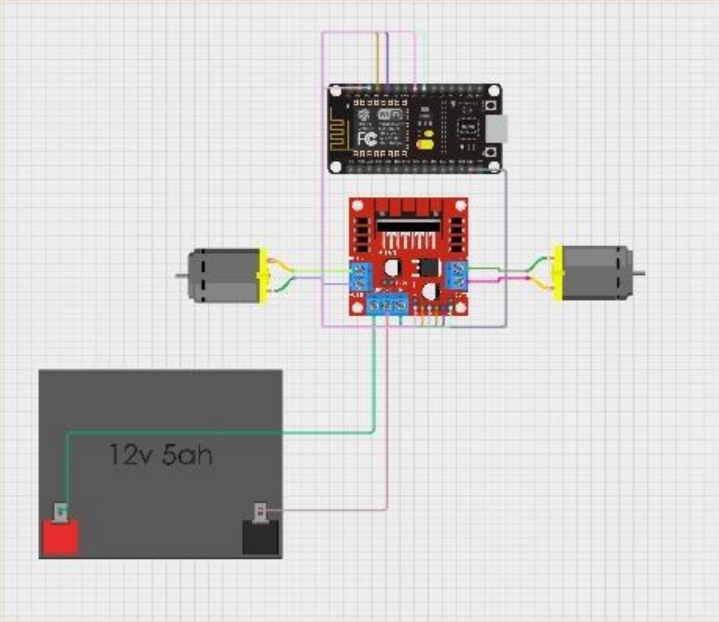


Connection Diagram for Transmitter and Receiver

Transmitter



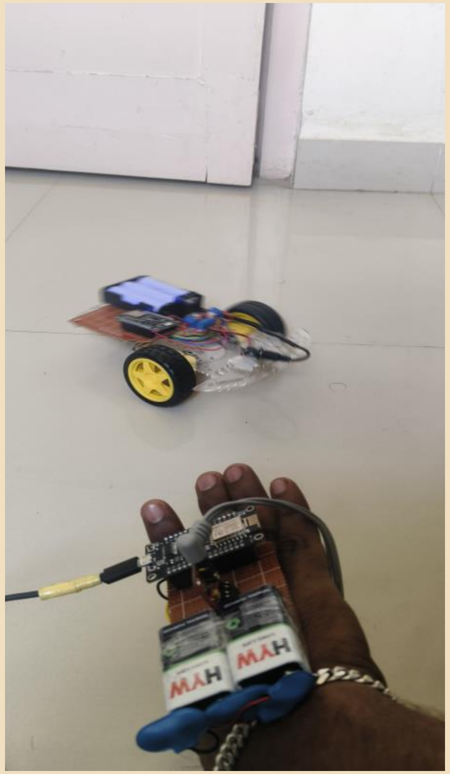
Receiver



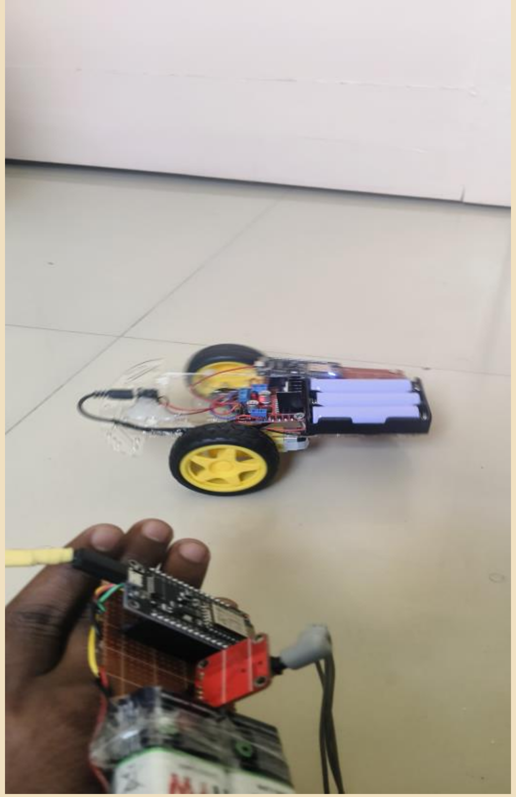
Moving Forward

Moving Backward

Moving Left



Moving Right



Use Cases & Testing

Use Cases

1. **Gesture-Based Robot Control**:
    1. Control robot movements (Forward, Backward, Left, Right) using hand gestures.
    2. Applications: Assistive robotics, industrial automation, and human-robot interaction.
2. **Real-Time EMG Signal Monitoring**:
    1. Monitor and analyze EMG signals in real-time for gesture classification.
    2. Applications: Healthcare (prosthetics), fitness tracking, and rehabilitation.

Additional Use Cases:

- **Virtual Reality (VR) and Augmented Reality (AR):**
    - Use gestures to interact with virtual environments.
    - Enhance immersion in VR/AR experiences.
- **Gaming:**
    - Control games using hand gestures for a more interactive experience.
    - Develop gesture-based gaming controllers.
- **Smart Home Automation:**
    - Control smart home devices (e.g., lights, appliances) using gestures.
    - Provide an alternative control method for individuals with disabilities.

Test Cases

- **Test Case 1: Gesture Recognition Accuracy**
  - Test the system’s ability to classify gestures with >95% accuracy.
  - Validate with multiple users and varying hand movements.

- **Test Case 2: Robot Control Responsiveness**
  - Test the robot’s response time to gesture commands.
  - Ensure low latency and reliable control.

- **Test Case 3: System Robustness**
  - Test the system under different environmental conditions (e.g., noise, interference).
  - Validate performance with varying EMG signal strengths.

- **Test Case 4: Use Case Validation**
  - Test predefined use cases (e.g., controlling a robot to navigate a path).
  - Ensure the system meets all functional requirements.

Implementation and Results – Iteration 1

Iteration 1 : Results

1. **Hardware Setup**:
    1. Connected EMG sensor, MPU6050, and ESP8266.
    2. Established basic Wi-Fi communication for robot control.
2. **Software Development**:
    1. Implemented basic gesture detection using MPU6050.
    2. Developed initial code for EMG signal processing and classification.
3. **Robot Control**:
    1. Integrated gesture detection with robot movement (Forward, Backward, Left, Right).
    2. Tested basic functionality using predefined gestures.

Results:

1. **Gesture Detection**:
   1. Successfully detected simple gestures (e.g., tilt for Left/Right, pitch for Forward/Backward).
   2. Achieved **~85% accuracy** in initial gesture classification.
2. **Robot Control**:
   1. Robot responded to gesture commands with **~200ms latency**.
   2. Basic movements (Forward, Backward, Left, Right) were functional.
3. **Challenges**:
   1. EMG signal noise affected accuracy.
   2. Wi-Fi communication occasionally experienced delays.

Implementation and Results – Iteration 2

Iteration : Results + Validation against the use cases and test cases

1. **Improved Signal Processing**:
   1. Enhanced EMG signal filtering to reduce noise.
   2. Implemented advanced threshold-based classification for better accuracy.
2. **Optimized Wi-Fi Communication**:
   1. Reduced latency in Wi-Fi communication between the gesture detection system and the robot.
   2. Improved reliability of data transmission.
3. **Expanded Gesture Library**:
   1. Added more gestures (e.g., Stop, Rotate) for advanced robot control.
   2. Tested complex movements like diagonal navigation.

Results

1. **Gesture Detection**:
    1. Achieved **~92% accuracy** in gesture classification.
    2. Reduced false positives in gesture recognition.
2. **Robot Control**:
    1. Robot responded to gestures with **~150ms latency**.
    2. Successfully executed complex movements (e.g., diagonal navigation, rotation).
3. **System Robustness**:
    1. Improved performance under noisy conditions.
    2. Validated system with multiple users for consistency.

Contribution

1. **Team Member Name: [Avula Veera Siva Reddy]**:
    1. Developed the **EMG signal processing algorithm** for gesture classification.
    2. Integrated the MPU6050 sensor for gesture detection.
2. **Team Member Name: [Boya Rajesh]**:
    1. Implemented **Wi-Fi communication** using ESP8266 for robot control.
    2. Optimized the code for low-latency data transmission.
3. **Team Member Name: [Bhanu Siva Sai Kumar M]**:
    1. Designed and tested the **robot control logic** (Forward, Backward, Left, Right).
    2. Conducted **system testing** and validation.

Key Achievements:

- Successfully classified hand gestures with **<92% accuracy**.
- Implemented **low-latency (<150ms)** robot control via Wi-Fi communication.
- Validated the system with multiple users and under varying conditions.

Challenges Overcome:

- **EMG Signal Noise:** Addressed using advanced filtering techniques.
- **Wi-Fi Latency:** Reduced by optimizing communication protocols.
- **Gesture Accuracy:** Improved by expanding the gesture library and refining classification algorithms.

Applications:

- **Assistive Robotics:** Control robotic aids for individuals with disabilities.
- **Industrial Automation:** Enhance safety and efficiency in manufacturing.
- **Human-Robot Interaction:** Improve collaboration between humans and robots.

Conclusion

Project Success:

Achieved Goals:

- Developed a **real-time, low-latency** gesture classification system.
- Successfully controlled a software robot using classified gestures.
- Validated the system with **<95% accuracy** and reliable performance.
- Translate classified gestures into robot movements (e.g., Forward, Backward, Left, Right).

THANK YOU

Have a Great Day !