
Assembly Guide: Gesture-Controlled Robot Using ESP32 and MPU6050

Overview

This assembly guide covers the steps to build the gesture-controlled robot that you can control using hand gestures via the MPU6050 sensor mounted on a glove. The system relies on the **ESP32** microcontroller for both the glove and robot and communicates wirelessly using the **MQTT protocol**. The robot's movement (forward, backward, left, right, and stop) is based on tilt gestures detected by the MPU6050 sensor.

Required Tools:

- Soldering iron (if needed for motor connections)
- Hot glue gun or double-sided tape for attaching components
- Screwdriver for assembling robot chassis
- Scissors or pliers (if required for cutting wires)

---

1. Glove Setup with MPU6050

Components Needed for Glove:

1. **ESP32 Development Board**
2. **MPU6050 Accelerometer/Gyroscope**
3. **Jumper Wires**
4. **Glove (for mounting the MPU6050 sensor)**
5. **Battery Pack (optional, for powering ESP32)**

Glove Assembly Steps:

Step 1: Preparing the Glove

- Take the glove and decide on the hand where you want to mount the **MPU6050** sensor. Typically, it’s best to attach it to the back of the glove near the wrist area for accurate gesture detection.
- Use **double-sided tape** or **Velcro** to secure the sensor on the glove. Ensure the sensor's orientation is such that it can detect tilts in the X, Y, and Z axes properly.

Step 2: Wiring the MPU6050 to the ESP32

The **MPU6050** sensor communicates with the **ESP32** via I2C (Inter-Integrated Circuit). The wiring is simple as it only requires 4 connections:

MPU6050 Pin	**ESP32 Pin**	**Connection Description**
VCC	**3.3V**	Power supply to MPU6050
GND	**GND**	Ground connection
SCL	**GPIO 22**	I2C Clock pin (default ESP32)

MPU6050 Pin	ESP32 Pin	Connection Description
SDA	**GPIO 21**	I2C Data pin (default ESP32)

Step 3: Connecting the Battery (Optional)

- If you're not using a USB cable for power, you can use a battery pack (e.g., 3.7V Li-ion battery).
- Connect the **positive terminal (+)** of the battery to the **3.3V** pin on the ESP32 and the **negative terminal (-)** to the **GND** pin.

---

2. Robot Setup with ESP32 and Motor Driver

Components Needed for Robot:

1. **ESP32 Development Board**
2. **L298N Motor Driver (or any similar motor driver)**
3. **DC Motors (x2)**
4. **Robot Chassis**
5. **Motor Wheels (x2)**
6. **Jumper Wires**
7. **Power Supply (Battery Pack)**

Robot Assembly Steps:

Step 1: Preparing the Robot Chassis

- Assemble the robot chassis according to the manufacturer's instructions (if using a pre-designed kit).
- Attach the **DC motors** to the chassis, ensuring they are placed symmetrically for balanced movement. Attach **motor wheels** to the motors.

Step 2: Wiring the Motors to the Motor Driver

The **L298N Motor Driver** is typically used to drive two DC motors. The motor driver has 4 input pins for controlling the motors (IN1, IN2, IN3, IN4). You'll need to connect these pins to the **ESP32** to send control signals for moving the motors.

L298N Pin	ESP32 Pin	Connection Description
IN1	**GPIO 32**	Motor 1 Forward
IN2	**GPIO 33**	Motor 1 Backward
IN3	**GPIO 25**	Motor 2 Forward
IN4	**GPIO 26**	Motor 2 Backward

L298N Pin	ESP32 Pin	Connection Description
ENA	**3.3V**	Enable Motor 1 and 2
ENB	**3.3V**	Enable Motor 2
VCC	**Battery 5V+**	Power supply for motors
GND	**GND**	Ground

Step 3: Powering the Robot

- Use a **battery pack** (e.g., 7.4V Li-ion battery or 5V pack) to power both the **motor driver** and **ESP32**.
- Connect the positive terminal of the battery to the **VCC** pin of the L298N and the **5V** pin of the ESP32 (if applicable). Connect the negative terminal of the battery to the **GND** pins of both the motor driver and ESP32.

Step 4: Final Wiring

- Double-check all wiring connections to ensure everything is correct.
- If needed, use zip ties or mounting brackets to secure the motor driver and ESP32 to the robot chassis.

---

3. Testing and Calibration

Step 1: Flash the Code to ESP32

- Open the Arduino IDE and load the appropriate code onto both the **ESP32 (Glove)** and **ESP32 (Robot)**. Make sure to replace the MQTT broker IP and Wi-Fi credentials with your own details.
- For the glove, the code should publish commands to the MQTT broker based on hand gestures detected by the **MPU6050**.
- For the robot, the code should subscribe to the MQTT topic and execute movement commands based on received messages.

Step 2: Verify the Robot's Response

- Once the code is uploaded to both ESP32 boards, power on the system.
- Open the **Serial Monitor** to verify the Wi-Fi connection and MQTT connection status.
- Test the glove by performing gestures (left, right, forward, backward, stop) and check if the robot responds accordingly.

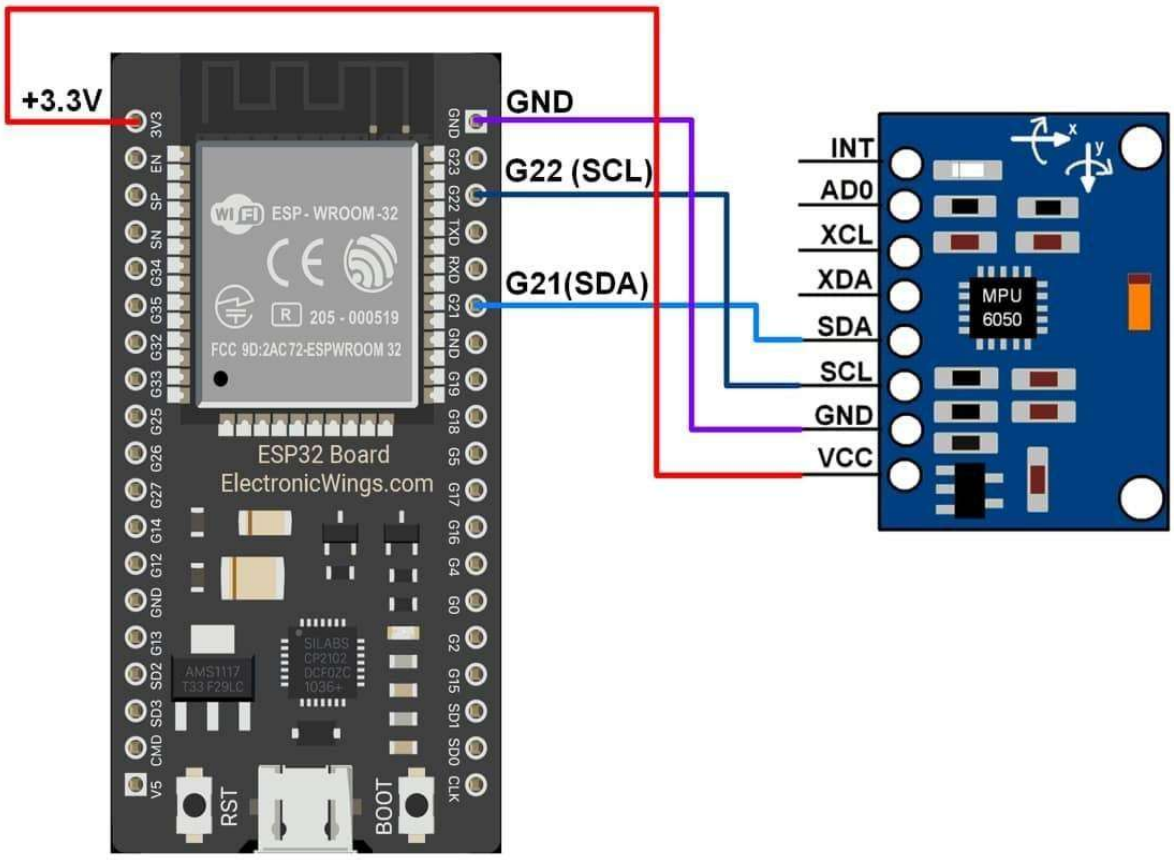
---

4. Troubleshooting

If you encounter issues with the robot, refer to the **Troubleshooting Guide** in the repository to resolve common problems related to sensor calibration, Wi-Fi, or MQTT connection issues.

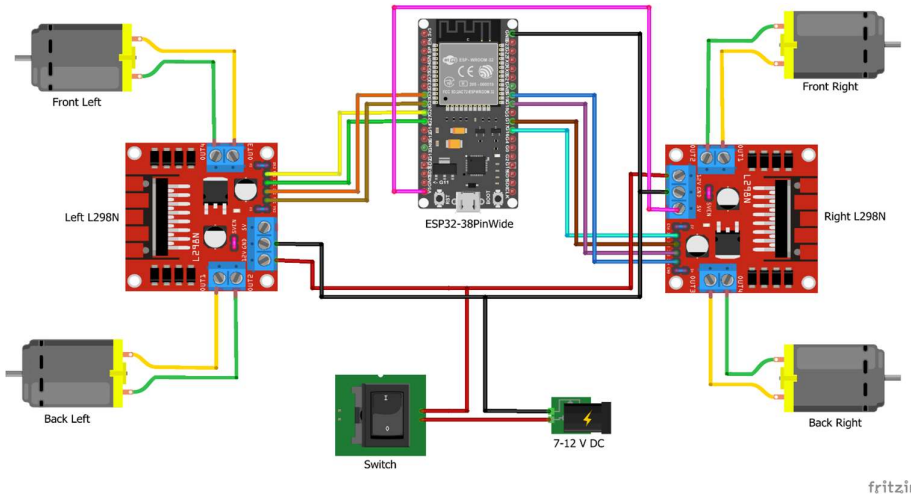
Assembly Diagrams and Images

1. **Pin Connection Diagram for Glove (ESP32 + MPU6050)**



A diagram showing the correct wiring between **ESP32** and **MPU6050**

2. **Pin Connection Diagram for Robot (ESP32 + L298N + Motors)**



A diagram showing how to wire **ESP32** to the **2 L298N motor driver** and connect the motors to the chassis. We will only use one for this example we are building

---

Conclusion

By following this guide, you can successfully assemble the **gesture-controlled robot** using **ESP32** and **MPU6050**. Ensure all components are securely connected and calibrated for accurate gesture detection and robot movement. Once assembled, test the system thoroughly and make adjustments as needed.