# Unmanned Ground Vehicle (UGV) kit

## User Guide

భారతీయ సాంకేతిక విజ్ఞాన సంస్థ హైదరాబాద్
भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

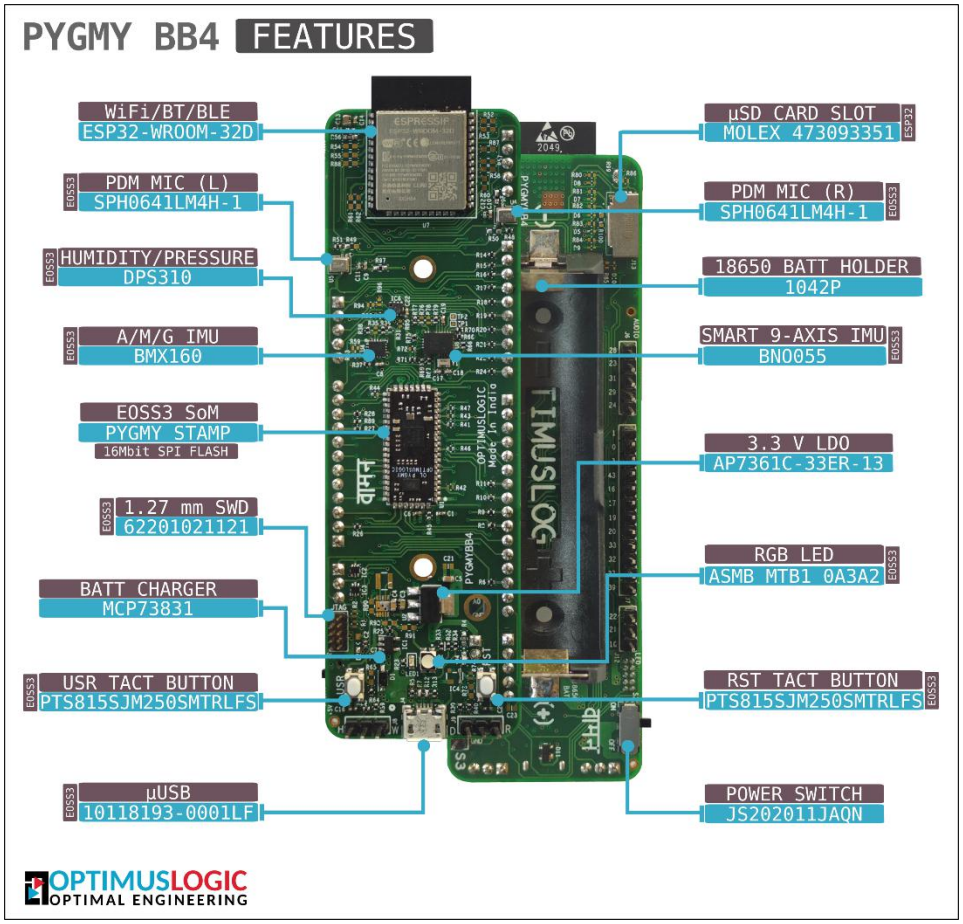
Tihan-IIT Hyderabad
NextGen Autonomous Navigation

# Table of Contents

1. Components and Specifications....3
1.1. Vaman (Pygmy BB4)....3
1.2. ESP 32....4
1.3. Arduino Uno.... **Error! Bookmark not defined.**
1.4. Dual-Motor Driver IC on breadboard.... **Error! Bookmark not defined.**
1.1. DC motors....5
1.2. Ultrasonic sensor....6
1.3. UGV frame/chassis....7
1.4. Batteries for powering various equipment....7
2. Assembling the UGV kit....8
3. Exercise Programs....11
3.1. Navigation using ESP32 and Android Phone....11
3.1.1. Connection diagram:....11
3.1.2. Wiring description....12
3.1.3. Codes (GitHub link)....12
3.2. Beacon tracking using ESP32....13
3.2.1. Connection diagram:....13
3.2.2. Wiring description....13
3.2.3. Codes (GitHub link)....14

# 1. Components and Specifications

## 1.1. Vaman (Pygmy BB4)




| SPECIFICATION FOR VAMAN CONTROLLER | | |
|---|---|---|
| **Make** | Optimus Logic | |
| **Model** | Vaman (Pygmy BB4) | |
| **Sr No** | **Description** | |
| | **General Data** | |
| 1 | Processors | E0SS3 SoM (ARM + FPGA dual processor) |
| 2 | Connectivity | Wi-Fi/BT/BLE connectivity with onboard ESP32 |
| 3 | Battery option | Yes |
| 4 | µSD card support | Yes (Connected to ESP32) |
| 5 | Onboard sensors | • BMX160 smart 9-Axis IMU<br>• BNO055 smart fusion sensor<br>• DPS310 provides Pressure, Humidity, and Temperature monitoring.<br>• PDM MIC (L) & (R) |
| 6 | Buttons | User and reset button for programming the board |
| 7 | Connector | Micro-USB Type-B connector for connection to PC during programming and debugging. |

## 1.2. ESP 32

| SPECIFICATION FOR ESP32 CONTROLLER | | |
|---|---|---|
| **Make** | Espressif Systems | |
| **Model** | ESP-WROOM-32 | |
| **Sr No** | **Description** | |
| | **General Data** | |
| 1 | Processors | • CPU: Xtensa dual-core (or single-core) 32-bit LX6 microprocessor, operating at 160 or 240 MHz and performing at up to 600 DMIPS<br>• Ultra-low-power (ULP) co-processor |
| 2 | Memory | • 320 KiB RAM, 448 KiB ROM |
| 3 | Connectivity | • Wi-Fi: 802.11 b/g/n<br>• Bluetooth: v4.2 BR/EDR and BLE (shares the radio with Wi-Fi) |
| 4 | Peripheral interfaces | • 34 × programmable GPIOs<br>• 12-bit SAR ADC up to 18 channels<br>• 2 × 8-bit DACs<br>• 10 × touch sensors (capacitive sensing GPIOs)<br>• 4 × SPI<br>• 2 × $I^2S$ interfaces<br>• 2 × $I^2C$ interfaces<br>• 3 × UART |
| 4 | Onboard sensors | • BMX160 smart 9-Axis IMU<br>• BNO055 smart fusion sensor<br>• DPS310 provides Pressure, Humidity, and Temperature monitoring.<br>• PDM MIC (L) & (R) |
| 6 | Power management | • Internal low-dropout regulator<br>• Individual power domain for RTC<br>• 5 µA deep sleep current<br>• Wake up from GPIO interrupt, timer, ADC measurements, capacitive touch sensor interrupt |
| 7 | Other features | • Motor PWM<br>• LED PWM (up to 16 channels)<br>• Hall effect sensor<br>• Ultra-low-power analog pre-amplifier |

DC motors

| SPECIFICATION FOR DC MOTORS | | |
|---|---|---|
| **Make** | Robo India | |
| **Model** | DC motor | |
| **Sr No** | **Description** | |
| | **General Data** | |
| 1 | Supply Voltage | 9V DC |
| 1 | Type | Brushed motor |
| 2 | Operating speed | Up to 300 RPM |
| 3 | Efficiency | 60-75 % with high starting torque |

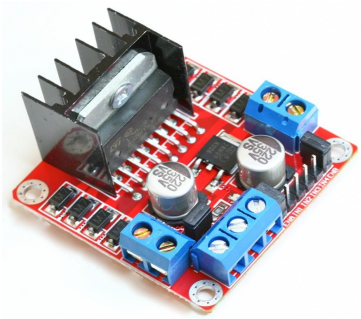
## 1.3. Ultrasonic sensor

| SPECIFICATION FOR ULTRASONIC CONTROLLER | | |
|---|---|---|
| **Make** | Arduino | |
| **Model** | HCSR04 | |
| **Sr No** | **Description** | |
| | **General Data** | |
| 1. | Input Voltage | 5V |
| 2. | Current Draw | 20mA (Max) |
| 3. | Digital Output | 5V |
| 4. | Digital Output | 0V (Low) |
| 5. | Working Temperature | -15°C to 70°C |
| 6. | Sensing Angle | 30° Cone |
| 7. | Angle of Effect | 15° Cone |
| 8. | Ultrasonic Frequency | 40kHz |
| 9. | Range | 2cm - 400cm |
| | **Dimensions** | |
| 10. | Length | 43mm |
| 11. | Width | 20mm |
| 12. | Height (with transmitters) | 15mm |
| 13. | Centre screw hole distance | 40mm x 15mm |
| 14. | Screw hole diameter | 1mm (M1) |
| 15. | Transmitter diameter | 8mm |

## 1.4. UGV frame/chassis

| SPECIFICATION FOR CHASIS | |
|---|---|
| | **General Data** |
| | • High-quality acrylic material<br>• Suitable for 2-wheel design.<br>• Supports universal Castor wheel<br>• All necessary screws & nuts |

## 1.5. L298 N motor driver



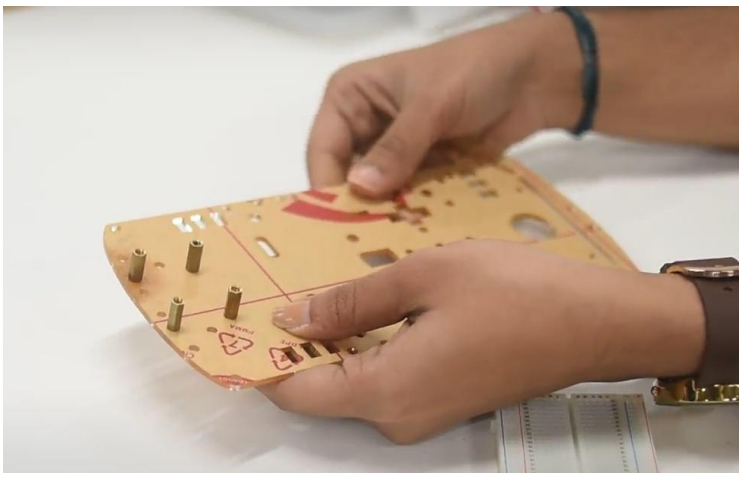
| SPECIFICATION FOR L298 N MOTOR DRIVER | |
|---|---|
| | **General Data** |
| | • Driver Model: L298N 2A<br>• Driver Chip: Double H Bridge L298N<br>• Motor Supply Voltage (Maximum): 46V<br>• Motor Supply Current (Maximum): 2A<br>• Logic Voltage: 5V<br>• Driver Voltage: 5-35V<br>• Driver Current:2A<br>• Logical Current:0-36mA<br>• Maximum Power (W): 25W<br>• Current Sense for each motor<br>• Heatsink for better performance<br>• Power-On LED indicator |
| | |

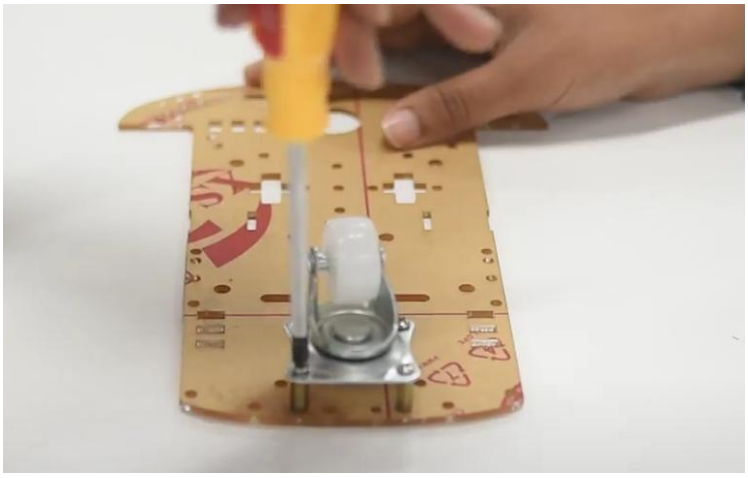
### 1.6. Batteries for powering various equipment

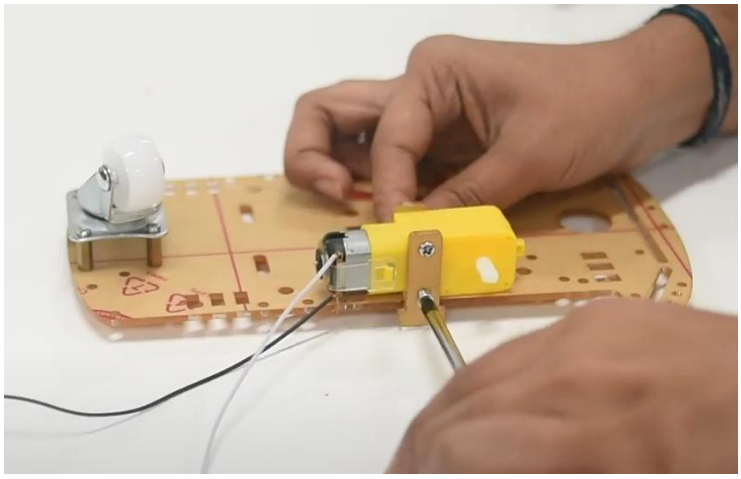

| SPECIFICATION FOR BATTERIES | | |
|---|---|---|
| | Li-Po Battery | |
| 1 | Voltage supply | 11.2 V |
| 2 | Rechargeable | Yes |
| 3 | Relevant components | For DC motors |

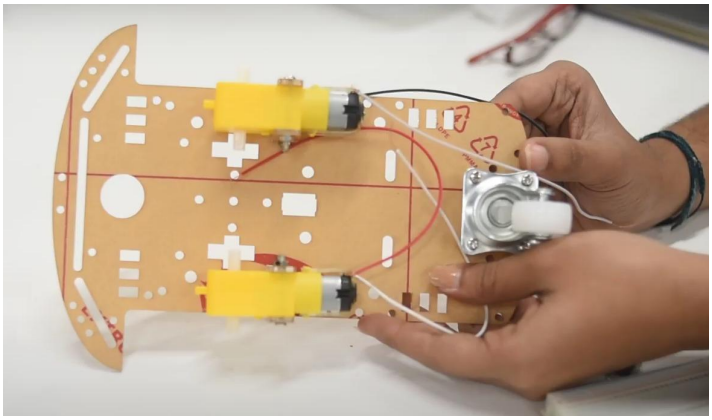
## 2. Assembling the UGV kit

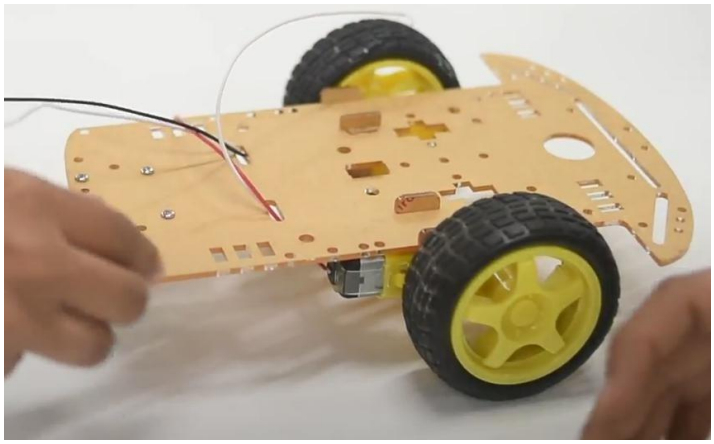
- Step 1: Assemble the Chassis using the provided nuts/screws, Wheels, and parts.



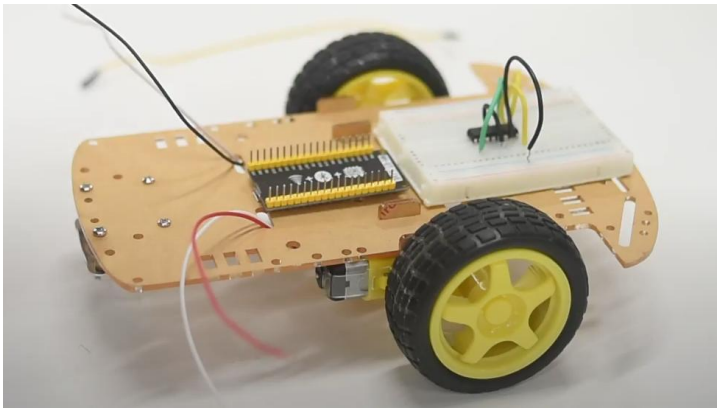


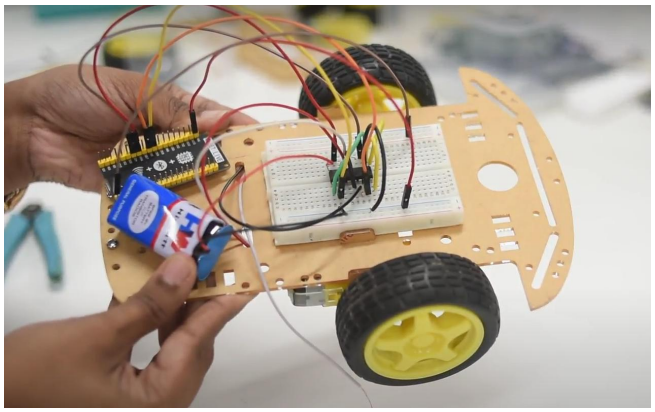


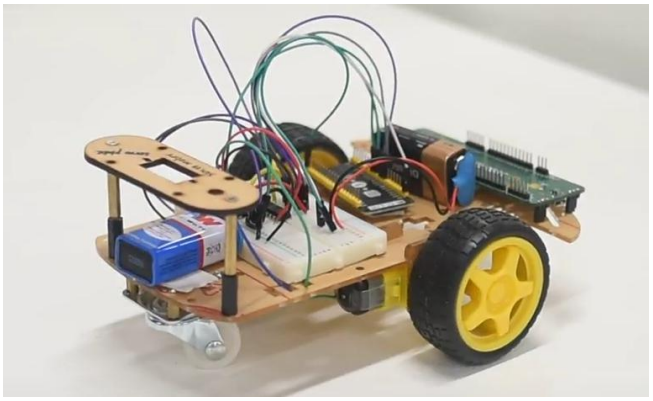




- Step 2:
  - Fix the Vaman controller and ESP32 on the chassis.
  - Fix the Dual motor driver IC along with a small breadboard on the chassis.
  - Fix the Li-Po battery on the chassis and insert AA batteries in the battery holder.

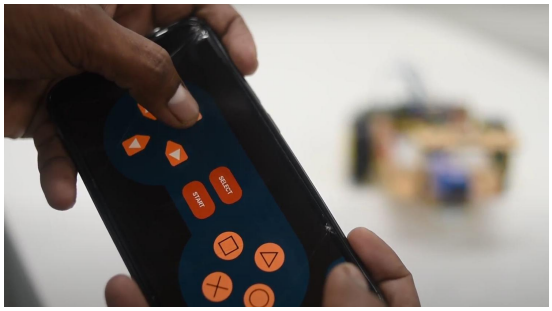
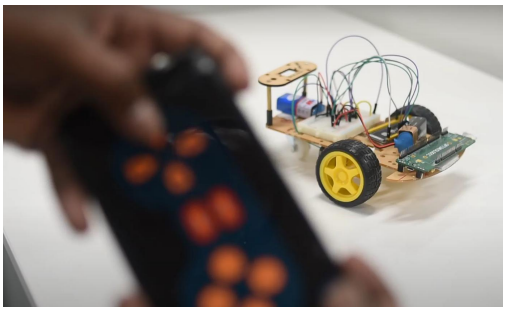






- Step 5:
  - Refer to the Exercise programs (section 3) for guide on wiring connection and programming of the controllers for the desired application.
  - Connect the battery supply and turn on the power to various equipment.

- Step 6:
  - Download the “dabble” application from the play store on an Android phone.
  - Using dabble application, connect to the ESP32 on the UGV kit using Bluetooth connection.
  - Control the navigation of the UGV kit using the GUI controls on the dabble application.

# 3. Exercise Programs

## 3.1. Navigation using ESP32 and Android Phone

### 3.1.1. Connection diagram:

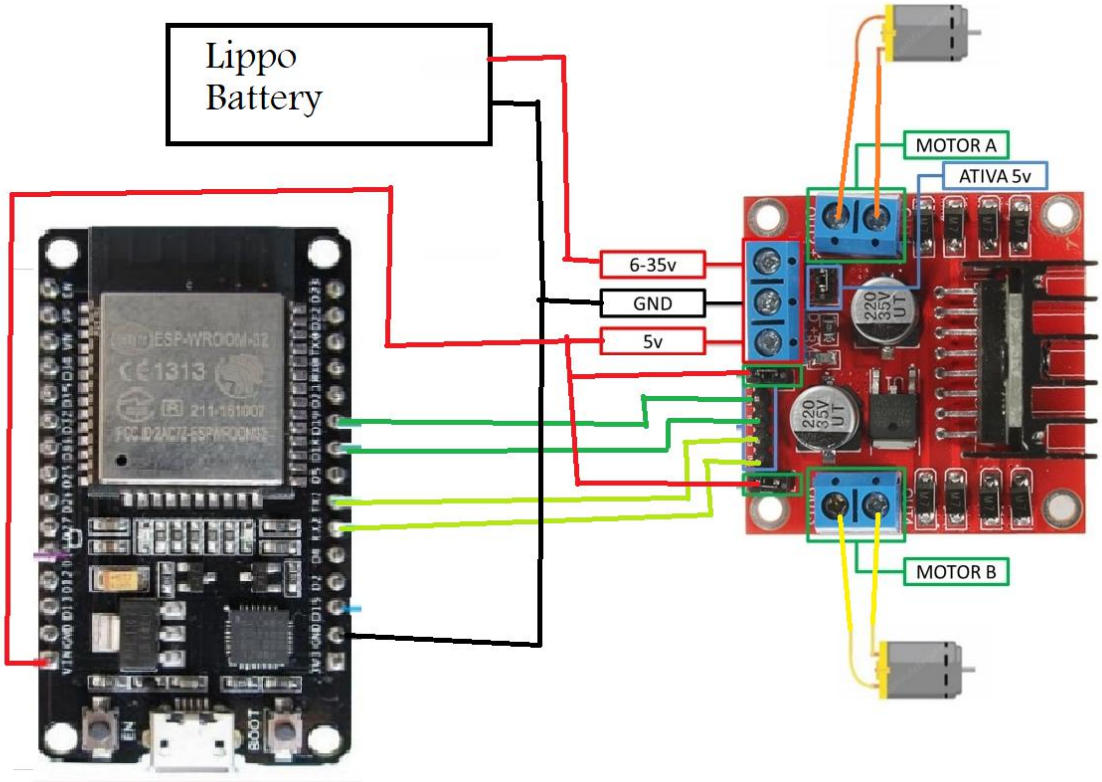



Figure 1 - Connection diagram

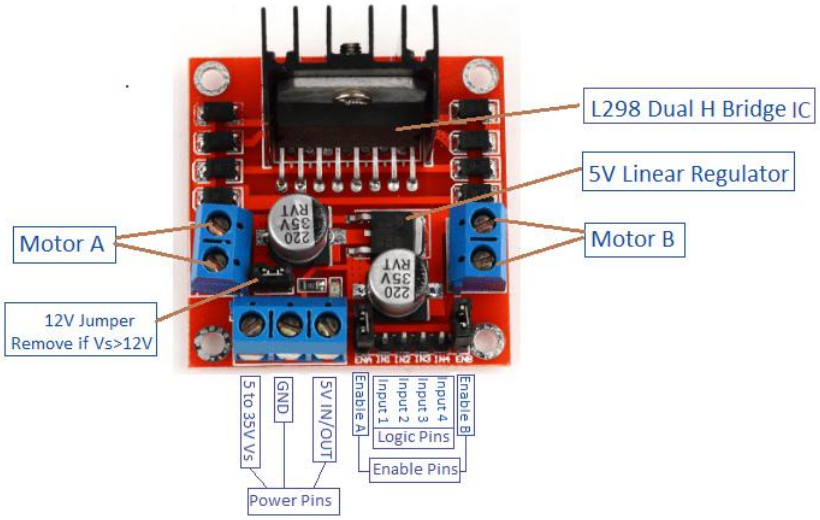



Figure 2 - Pin diagram of L298 N Motor driver

### 3.1.2. Wiring description

- Refer the pin diagram of L298 N motor driver
- Connect VCC1 pin to Vin pin on ESP32.
- The input and enable pins (ENA, IN1, IN2, IN3, IN4 and ENB) of the L298 N are connected to six ESP32 digital output pins (14, 16, 17, 18, 19 and 15).
- Connect one motor across the OUT1 & OUT2 pins of driver and the other motor across the OUT3 & OUT4 pins of driver.
- Connect external 9V battery to the VCC2 pin of L298N motor driver.
- Connect external GND pin battery to the GND pin of L298 N motor driver
- Go to Arduino IDE and Write the following code:

https://github.com/sachinomdubey/Projects/blob/main/Autonomous%20Navigation/ESP32/IDE/UGV_navigation_using_android_phone/Codes/UGV_navigation_using_android_phone.ino

- Click on Compile and Upload the code to" DOIT ESP32 DEVKITV1".
- Now open Dabble app search for Bluetooth devices and Select" MyEsp32" and connect it.
- After connecting click on Gamepad Icon and you will get control panel.

### 3.1.3. Codes (GitHub link)

https://github.com/sachinomdubey/Projects/tree/main/Autonomous%20Navigation/ESP32/IDE/UGV_navigation_using_android_phone/Codes

## 3.2. Beacon tracking using ESP32

### 3.2.1. Connection diagram:

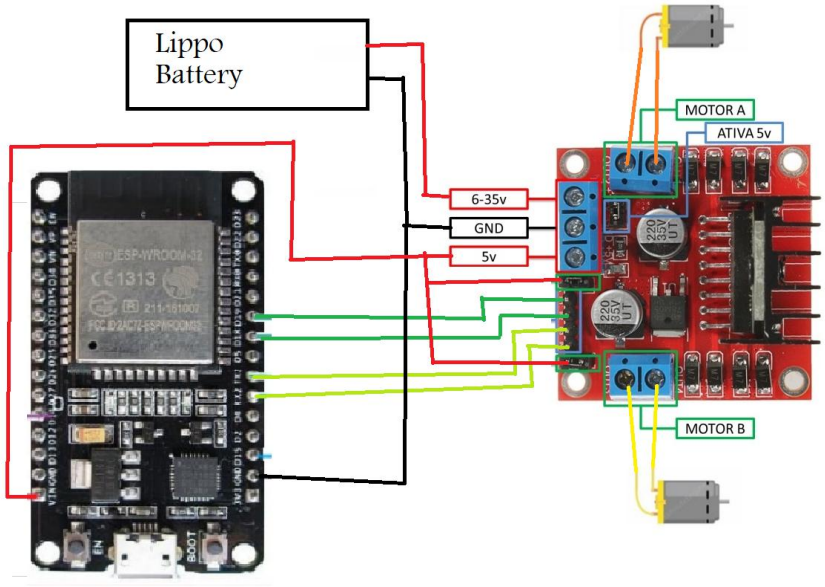



Figure 1 - Connection diagram

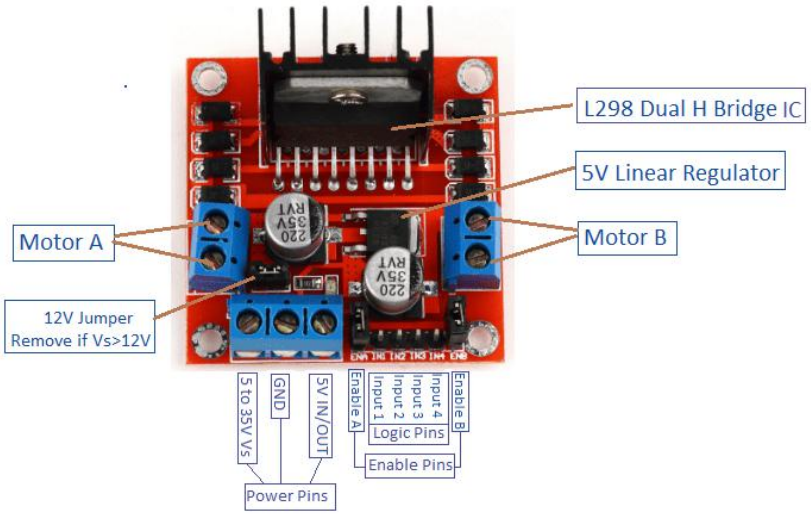



Figure 2 - Pin diagram of L298 N Motor driver

### 3.2.2. Wiring description

- Refer the pin diagram of L298N motor driver
- Connect VCC1 pin to 5V pin on ESP32.
- The input and enable pins (IN1, IN2, IN3, IN4) of the L298N are connected to six ESP32 digital output pins (16, 17, 18, 19).
- Connect the ENA and ENB to the 5V pin of EPS32
- Connect one motor across the OUT1 & OUT2 pins of driver and the other motor across the OUT3 & OUT4 pins of driver.

- Connect external 9V battery to the VCC2 pin of L298 N motor driver .
- Connect external GND pin battery to the GND pin of L298 N motor driver .
- Go to Arduino IDE and Write the following code:

https://github.com/sachinomdubey/Projects/blob/main/Autonomous%20Navigation/ESP32/IDE/Beacon_tracking/navsuccess_edit.ino

- Note : Before compiling the code remember to modify the code in order to connect to your mobile hotspot. For this, change the SSID and password to desired values in the code.
- Click on Compile and Upload the code to ”DOIT ESP32 DEVKITV1”.
- Turn on the mobile hotspot on your android phone whose SSID was entered in the code. Keep the mobile phone at a particular location.
- The ESP32 will connect with the hotspot over Wi-Fi and the subsequent code will guide allow ESP32 to navigate to the mobile phone’s location by measuring the received signal strength.

### 3.2.3. Codes (GitHub link)

https://github.com/sachinomdubey/Projects/blob/main/Autonomous%20Navigation/ESP32/IDE/Beacon_tracking/navsuccess_edit.ino