**WolfieWeb Robotics & AI**

Beginner Build: Raspberry Pi 4 / Pi 5 Robot Car

Build it. Wire it. Code it in Python. Add obstacle avoidance.

# Printable Tutorial

## Build a Beginner Robot Car with Raspberry Pi 4 or Raspberry Pi 5

This guide walks you from zero to a working Python-controlled robot car with simple obstacle avoidance. It is designed for beginners and uses common, easy-to-find parts.

Date: December 17, 2025

# What You Will Build

A small 2WD or 4WD robot car that can drive forward/back/turn, and stop/turn away when it sees an obstacle using an ultrasonic distance sensor.

## Sections

1) Parts list and shopping links

2) Build the chassis

3) Power plan (important)

4) Wiring the motor driver (L298N or TB6612FNG)

5) Wiring the ultrasonic sensor safely

6) Install Raspberry Pi OS and set up software

7) Run the motor test program

8) Add obstacle avoidance

9) Troubleshooting

# 1) Parts List and Shopping Links

| Part | Notes | Shopping links (examples) |
|---|---|---|
| Raspberry Pi 4 or Raspberry Pi 5 | Either works. Pi 5 is faster. | Raspberry Pi resellers: raspberrypi.com/products |
| Power supply | Pi 4: official 15W USB-C. Pi 5: official 27W USB-C PD recommended. | Official Pi 27W PSU: raspberrypi.com/products/27w-power-supply/<br>Adafruit listing: adafruit.com/product/5814 |
| microSD card (32GB+) | Use a name brand A1/A2 card. | PiShop SD cards: pishop.us/.../sd-cards/<br>Guide: Tom's Hardware microSD roundup |
| Robot car chassis + DC gear motors | 2WD is simplest; 4WD is fine. | Example 4WD kit: amazon.com/dp/B07F759T89 |
| Motor driver (choose one) | TB6612FNG runs cooler than L298N. L298N is very common. | TB6612FNG (SparkFun): sparkfun.com/rob-14450<br>L298N board: makerfabs.com/l298n-motor-driver-board.html |
| Ultrasonic sensor (HC-SR04) | Needs 3.3V-safe echo to Pi. | Adafruit: adafruit.com/product/3942<br>Amazon example: amazon.com/dp/B01COSN7O6 |
| Resistors for echo divider | 1k + 2k (or 1k + 1.8k) works. | Any electronics resistor kit |
| Motor battery pack | 6xAA NiMH is beginner-safe. (6-9V motor supply). | RobotShop AA holder: robotshop.com/products/6-aa-battery-holder-enclosed-with-switch |
| Jumper wires + small breadboard | Male-female wires help a lot. | Any starter jumper kit |

Tip: Start with L298N if you want the easiest wiring to follow. Choose TB6612FNG if you want cooler/quieter motor control.

# 2) Build the Chassis

Assemble the chassis, mount the motors, wheels, caster (if included), and battery holder. Mount the Raspberry Pi on standoffs or a top plate so it does not short against metal.

## Checklist

- Motors bolted tight

- Wheels fully seated

- Battery holder secured

- Pi mounted with spacers

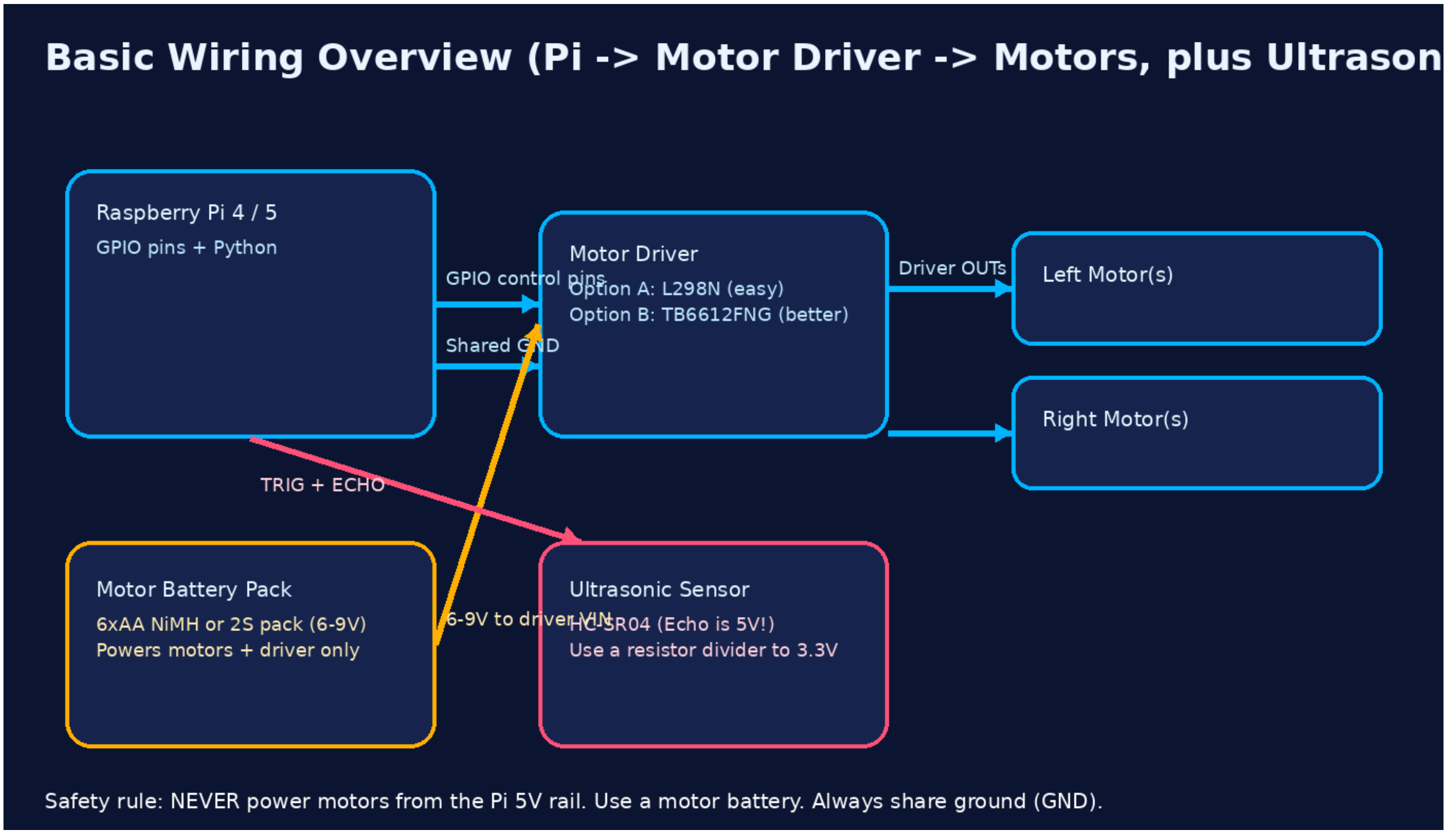
- Wires cannot rub wheels

# 3) Power Plan (Read This)

You need two power systems:
1) Raspberry Pi power: a proper USB-C supply (Pi 5: official 27W PD recommended).
2) Motor power: a separate battery pack that feeds the motor driver.

Rule: do NOT power motors from the Pi 5V pins. Motors create electrical noise and can reboot or damage the Pi.

## 4) Wiring Overview


Basic Wiring Overview (Pi -> Motor Driver -> Motors, plus Ultrason

Safety rule: NEVER power motors from the Pi 5V rail. Use a motor battery. Always share ground (GND).

Your exact pin numbers can be changed, but keep them consistent in your code. The important parts are: shared ground (GND) and separate motor power.

# 5) Suggested Pin Mapping (Works on Pi 4 and Pi 5)

| Function | GPIO (BCM) | Notes |
|---|---|---|
| Motor IN1 | 17 | Direction control |
| Motor IN2 | 18 | Direction control |
| Motor IN3 | 22 | Direction control |
| Motor IN4 | 23 | Direction control |
| Ultrasonic TRIG | 24 | Output from Pi |
| Ultrasonic ECHO | 25 | Input to Pi (must be 3.3V-safe) |
| GND | Any GND pin | All grounds must connect together |

**Echo safety:** HC-SR04 echo outputs 5V. Use a resistor divider so the Pi sees about 3.3V. Example: 2k from ECHO to Pi GPIO, and 1k from Pi GPIO to GND.

# 6) Software Setup

- Flash Raspberry Pi OS (64-bit) using Raspberry Pi Imager.

- Boot the Pi, finish setup, and connect to Wi-Fi (or Ethernet).

- Update packages: sudo apt update && sudo apt full-upgrade -y

- Install gpio tools (usually already present): sudo apt install -y python3-pip

- Enable I2C only if you use an I2C motor HAT (optional).

## 7) Motor Test Program (Python)

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)

IN1, IN2, IN3, IN4 = 17, 18, 22, 23
for p in (IN1, IN2, IN3, IN4):
  GPIO.setup(p, GPIO.OUT)

def stop():
  for p in (IN1, IN2, IN3, IN4):
    GPIO.output(p, 0)

def forward():
  GPIO.output(IN1, 1); GPIO.output(IN2, 0)
  GPIO.output(IN3, 1); GPIO.output(IN4, 0)

try:
  forward(); time.sleep(2)
  stop(); time.sleep(1)
finally:
  stop(); GPIO.cleanup()
```

Save as robot_test.py and run: python3 robot_test.py. If it drives the wrong way, swap the motor wires on one side.

## 8) Add Obstacle Avoidance (Simple Logic)

Basic loop: read distance; if under 20 cm, stop and turn; otherwise drive forward.

## 9) Troubleshooting

- **Robot does not move:** check motor driver power and that grounds are shared.

- **Pi reboots:** motors stealing power/noise **-** keep motor power separate, shorten wires, and ensure a solid ground.

- **Distance reads 0 or nonsense:** echo wiring/voltage divider wrong; confirm TRIG and ECHO pins match code.

- **Turns are weird:** one motor wired backward **-** swap the motor leads on that side.

WolfieWeb tip: Once this works, you can add a camera and control the robot from a phone browser.