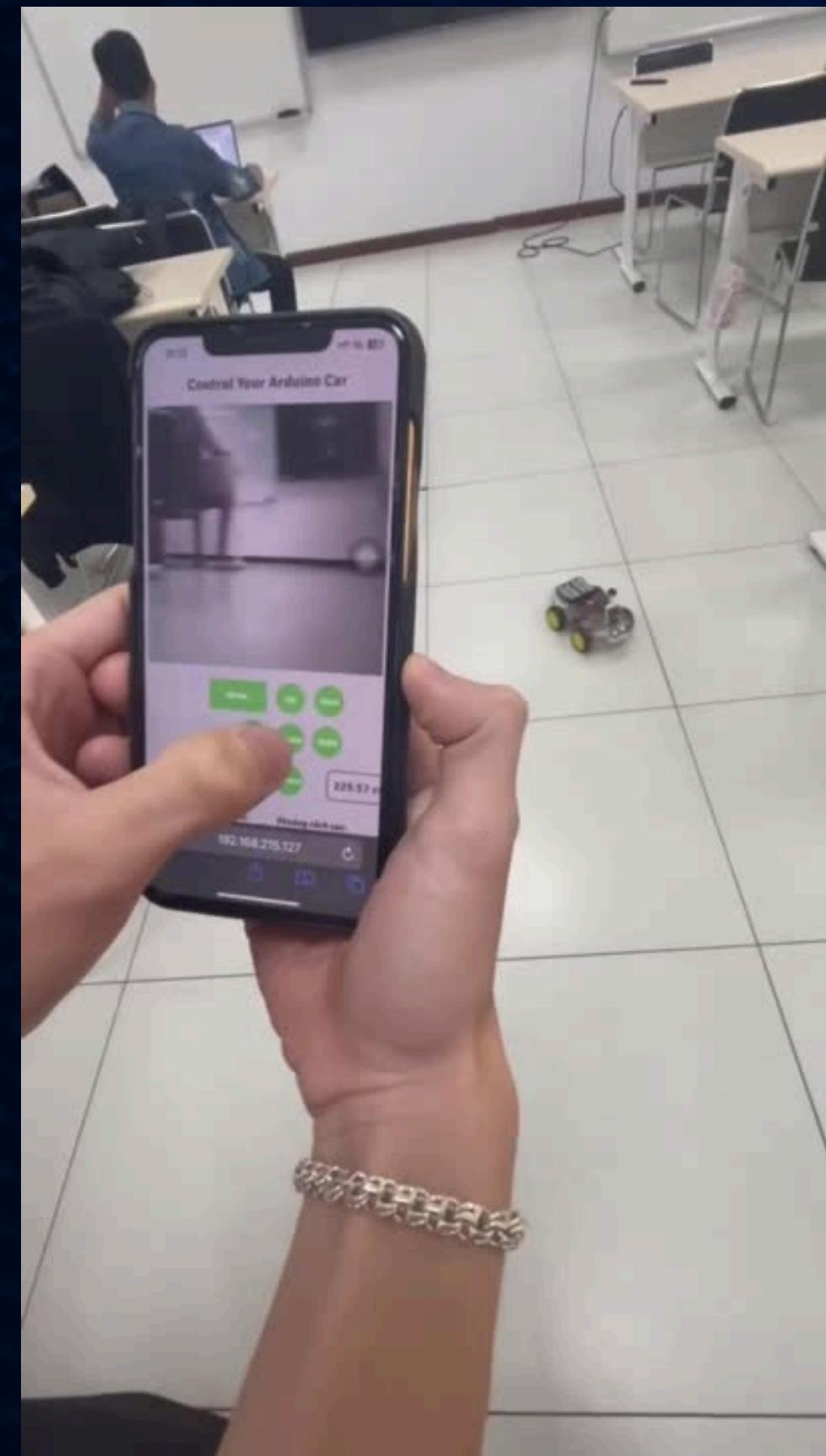


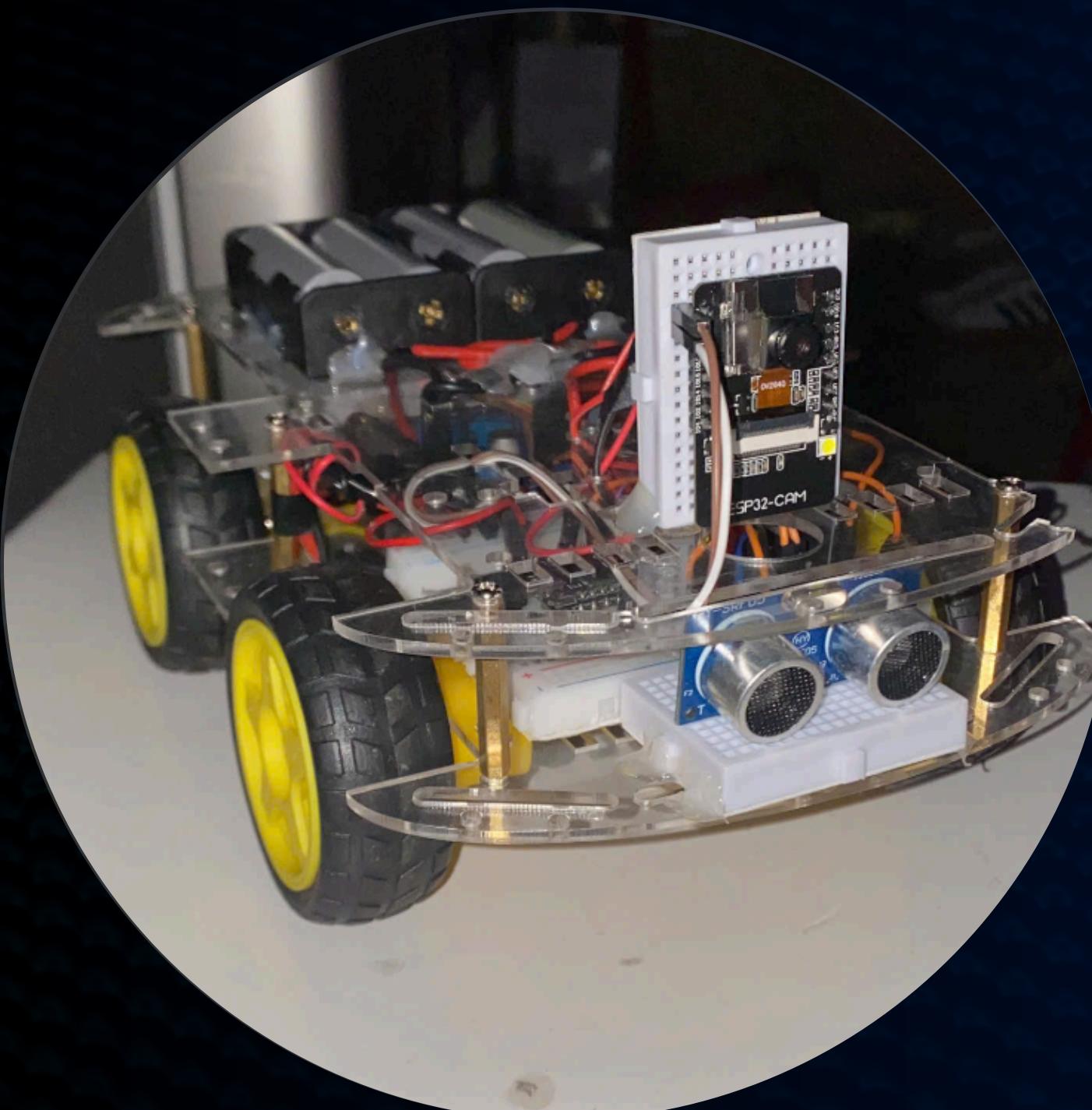


ARDUINO PROJECT

REMOTE CONTROL CAR WITH CAMERA

[Get Started](#)[Play Video](#)





Introduction

Hello teachers, today our group would like to introduce the project "Remote Control Car with Camera". This is a small, flexible car that can be controlled remotely via a laptop or phone. It combines distance sensors, encoder V2, ESP32-CAM, and the INA219 current sensor.

The project helps our group apply knowledge of programming and electronics, while also developing creativity in optimizing the car's features. We hope that this product will provide an interesting and useful experience for everyone in learning about automation technology.



I. Hardware (Basic)

**Động Cơ Giảm Tốc 3-9V
(2 trục)**



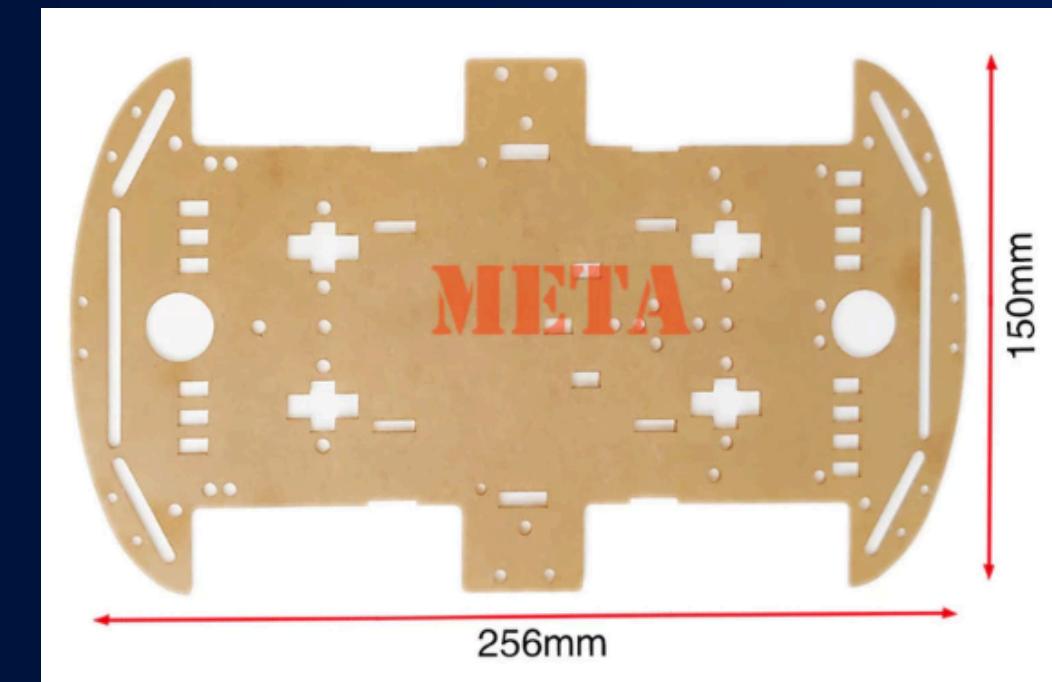
Bánh Xe Vàng



**2 Pin
2s2p**

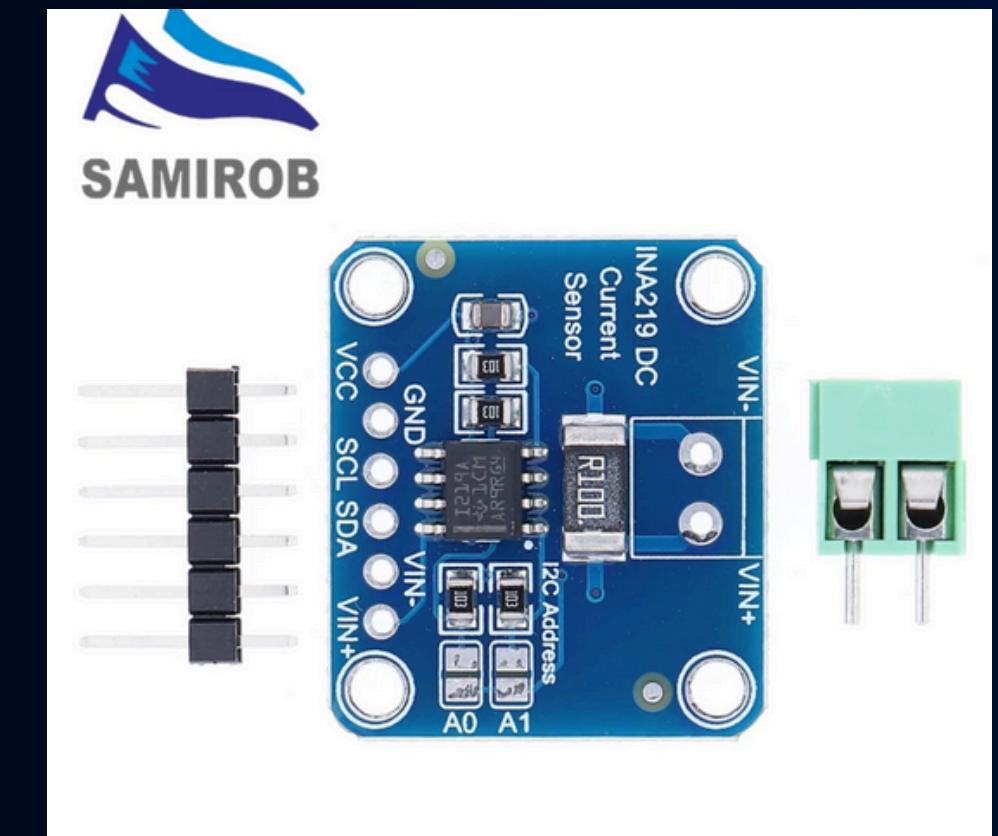
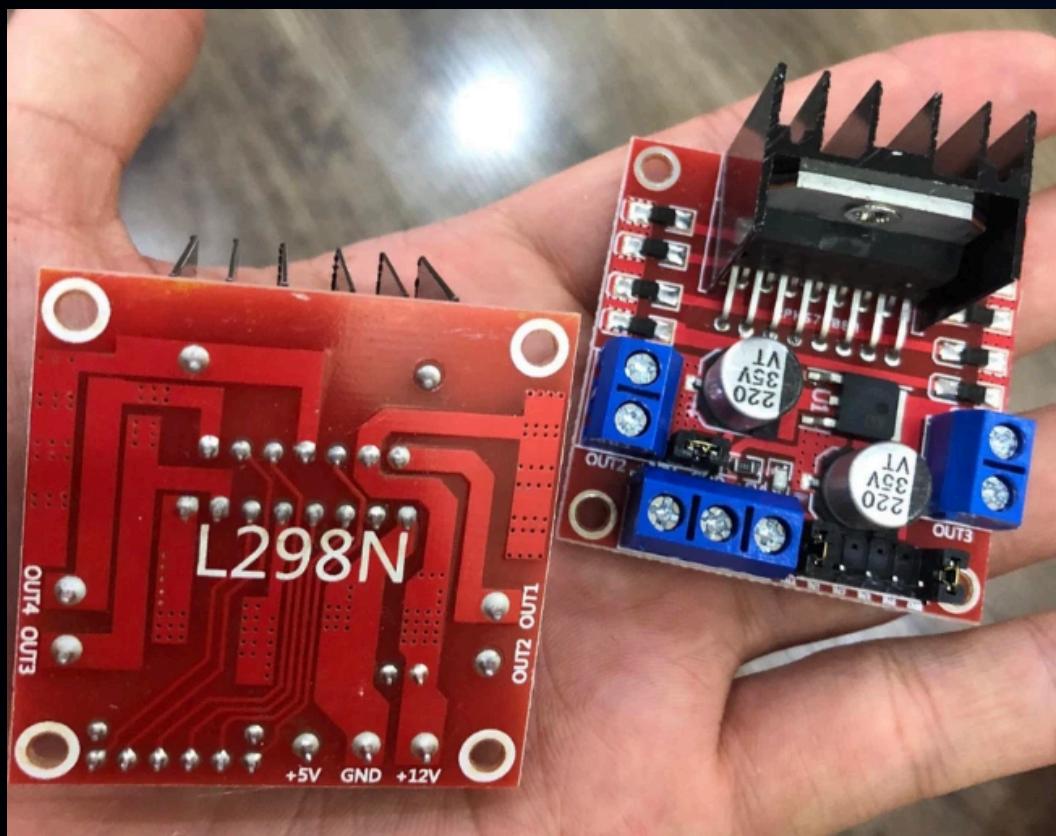
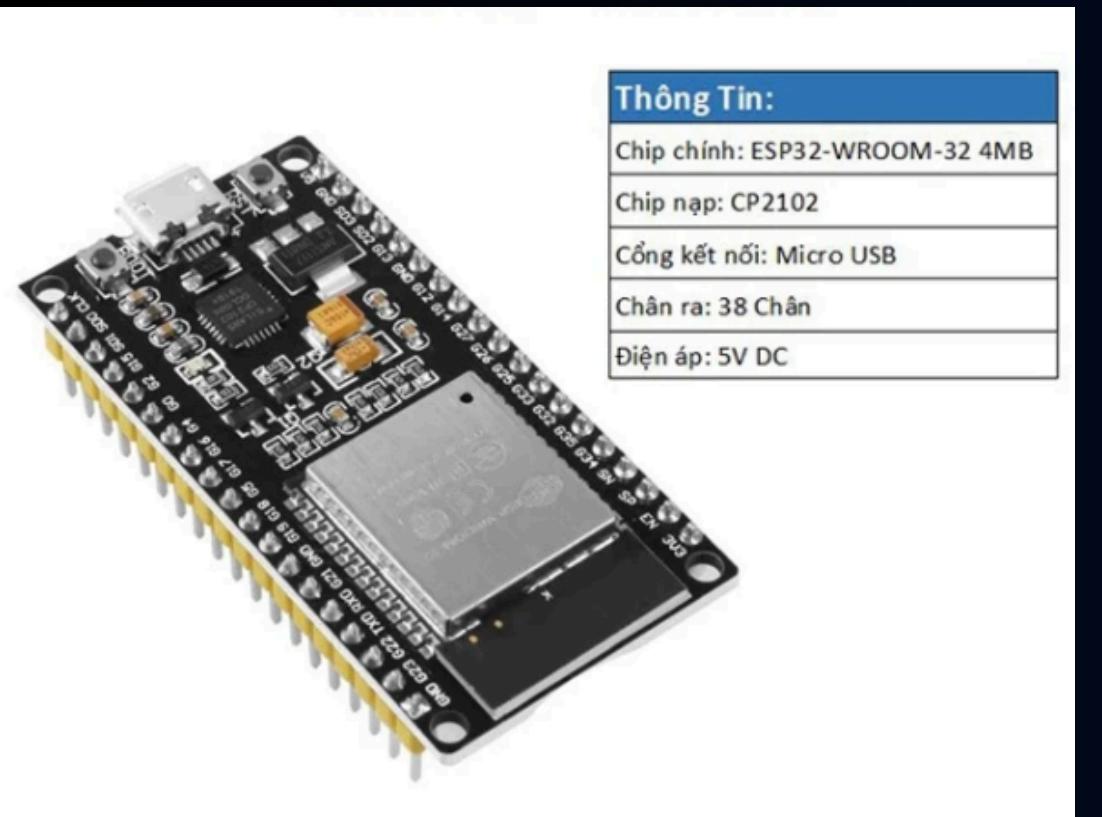


TALO SAFETY



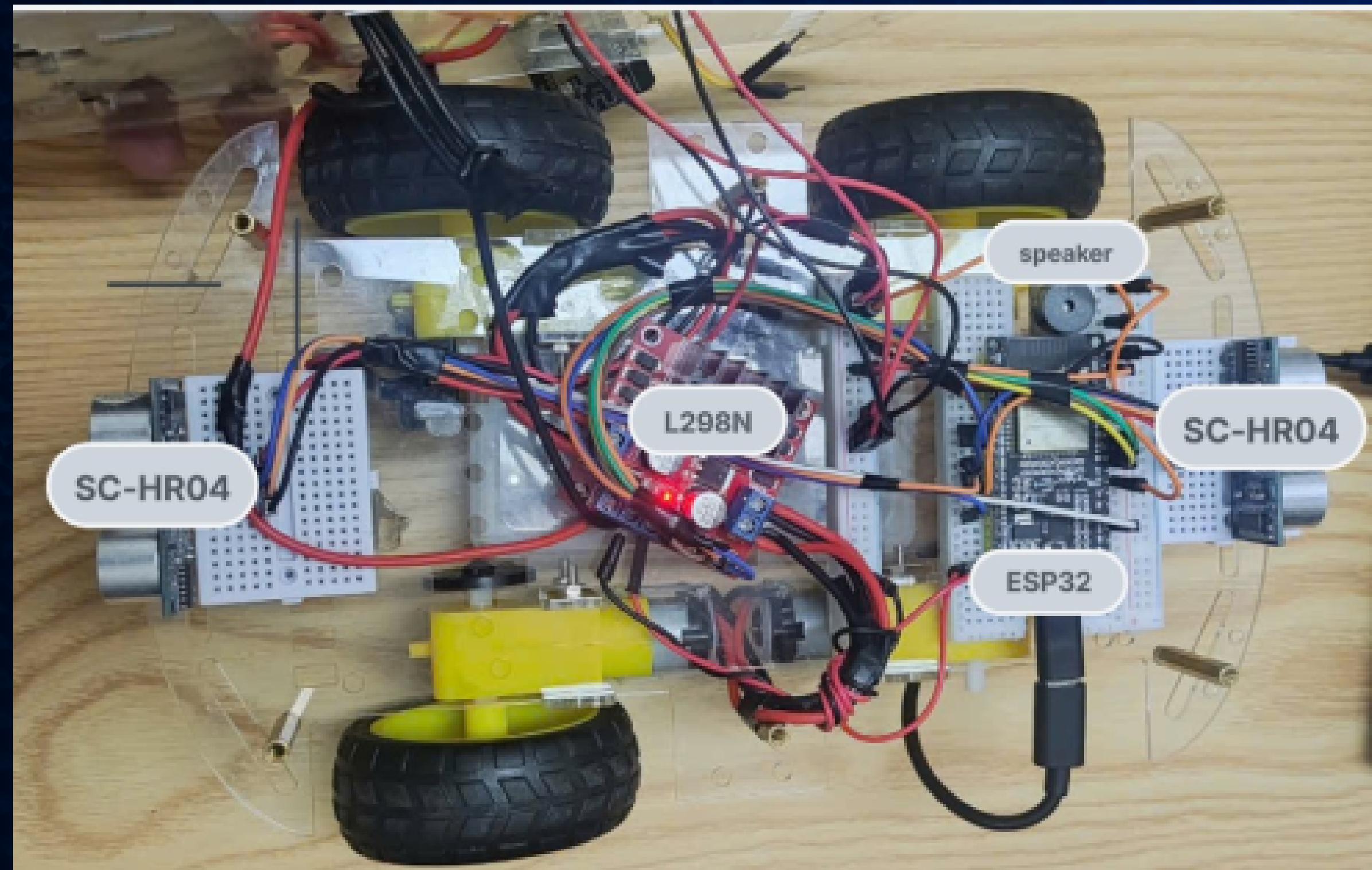
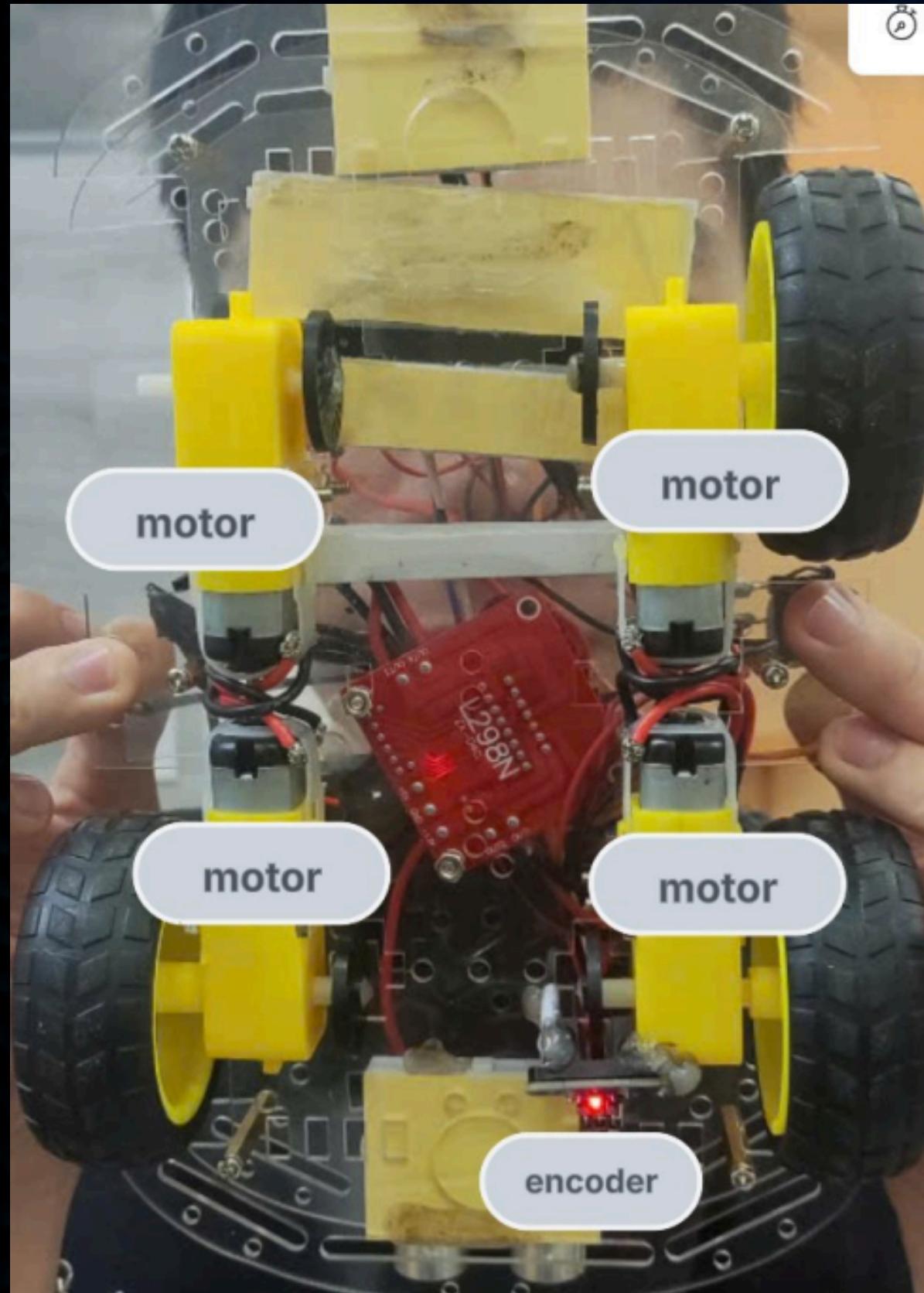


I. Hardware (Controller & Sensor)



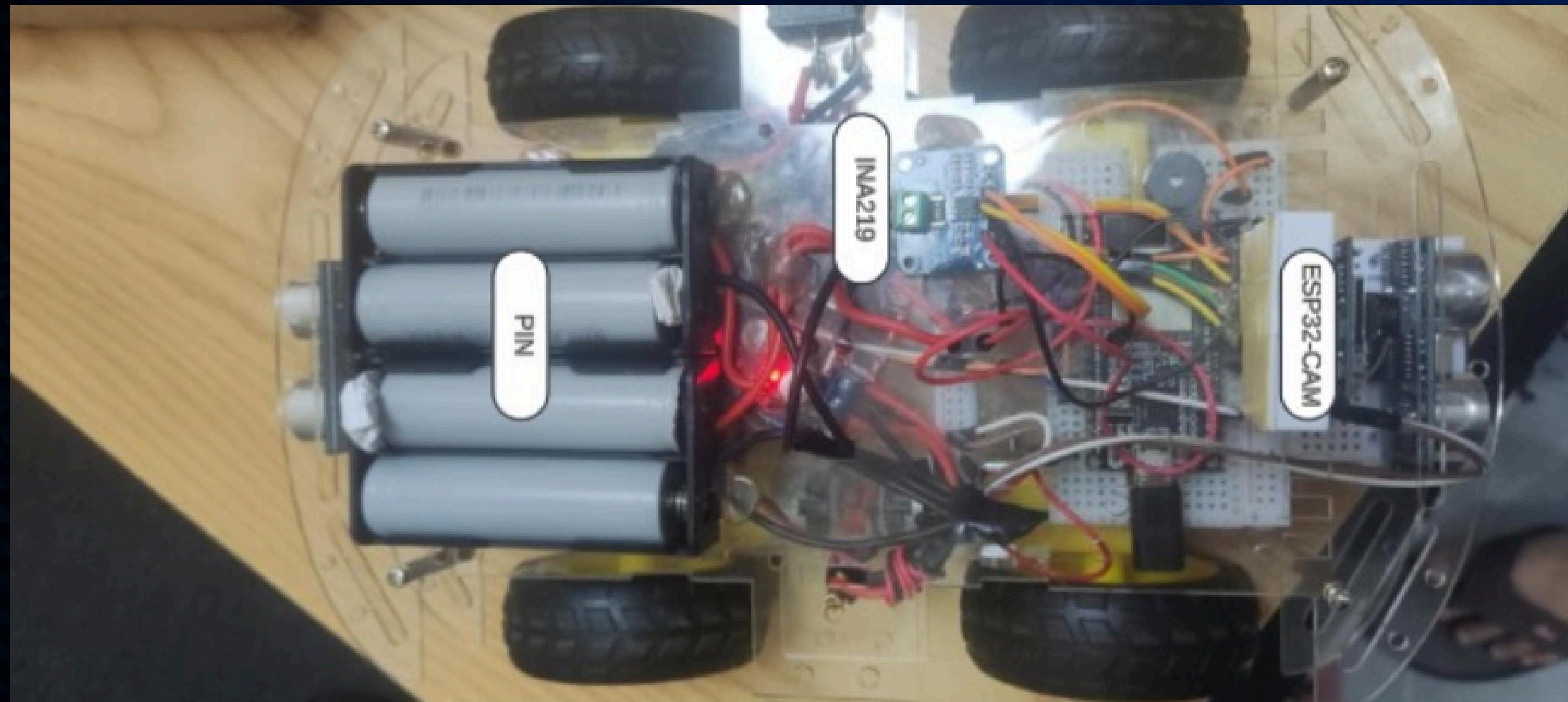


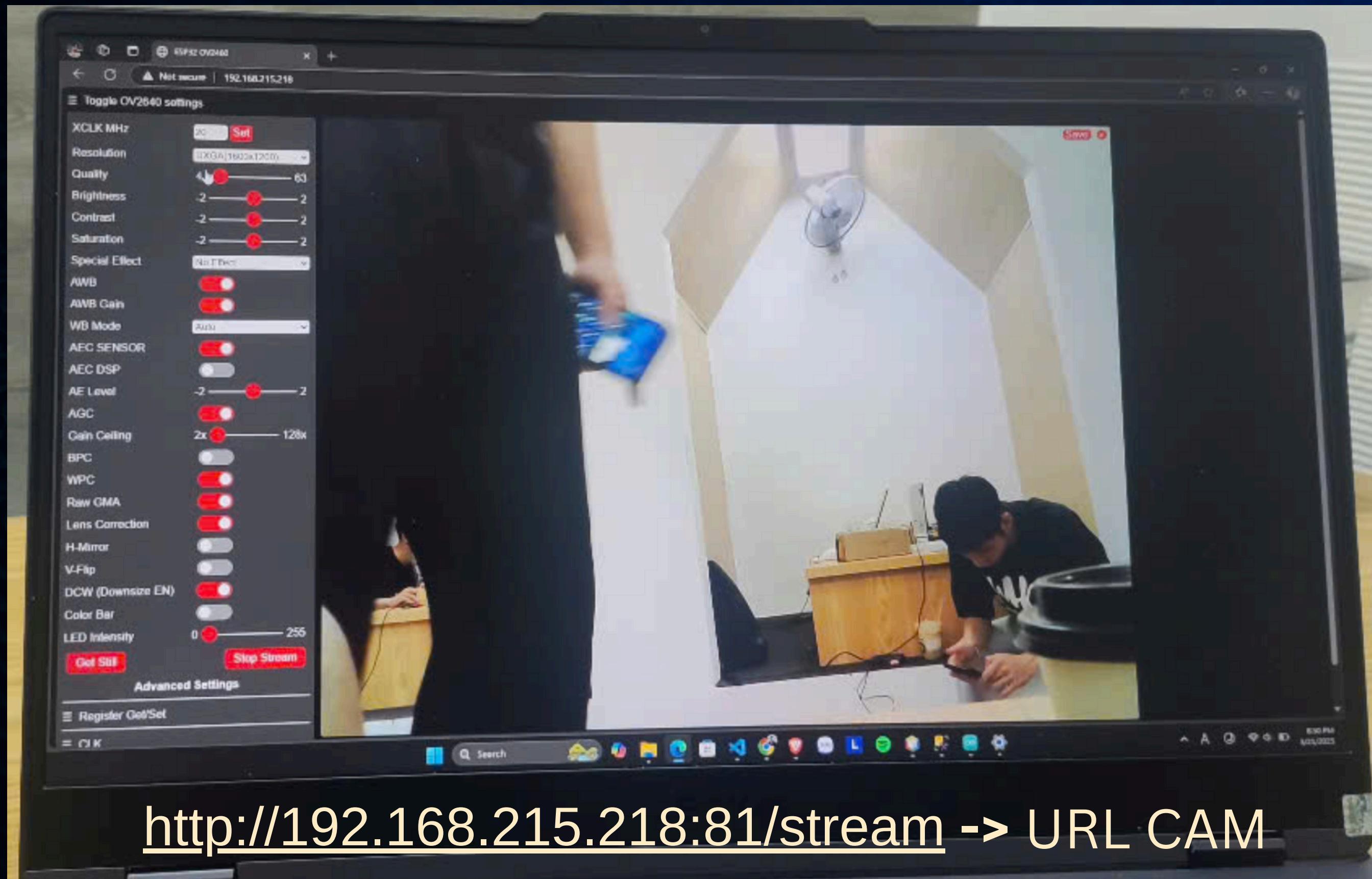
II. Model assembly.





II. Model assembly.





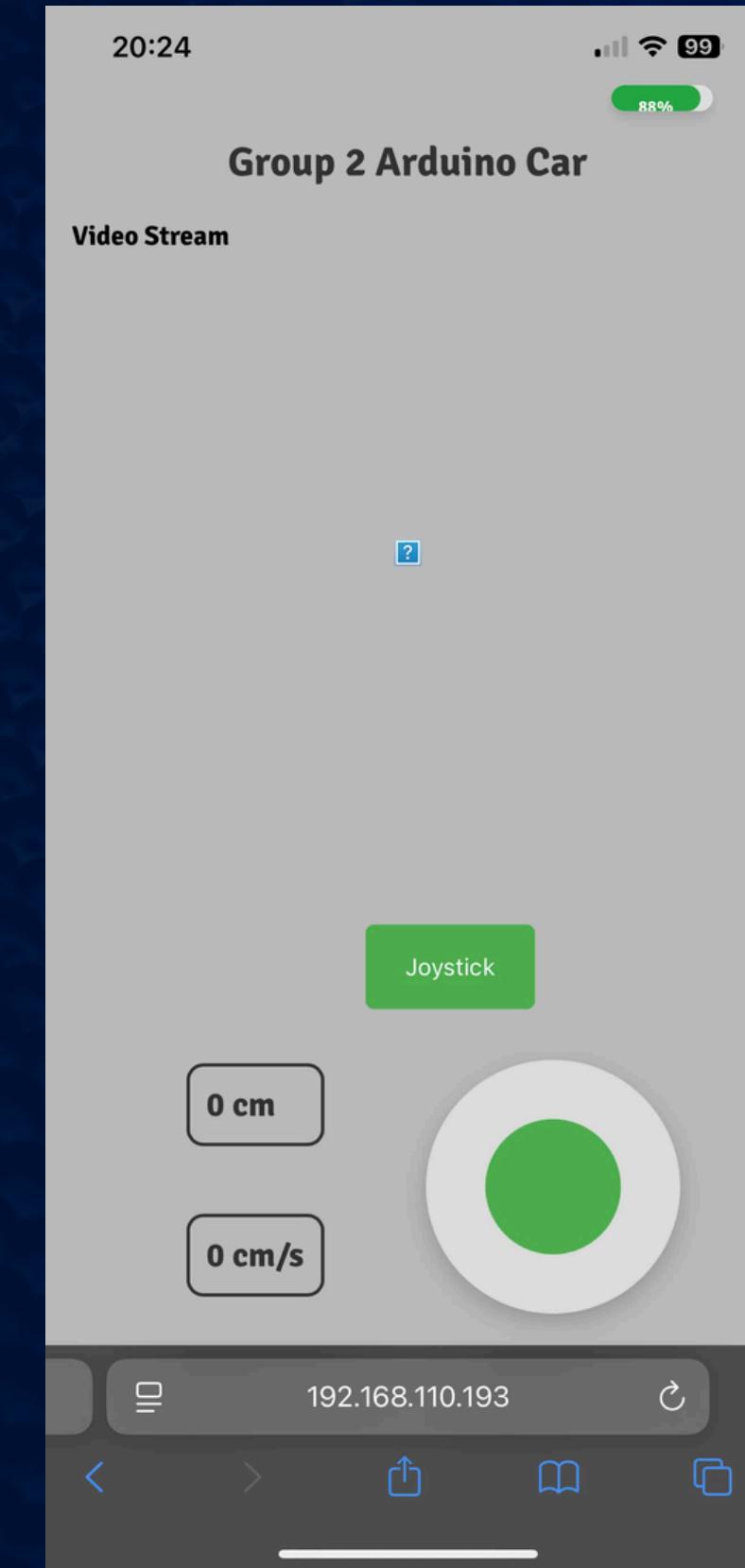
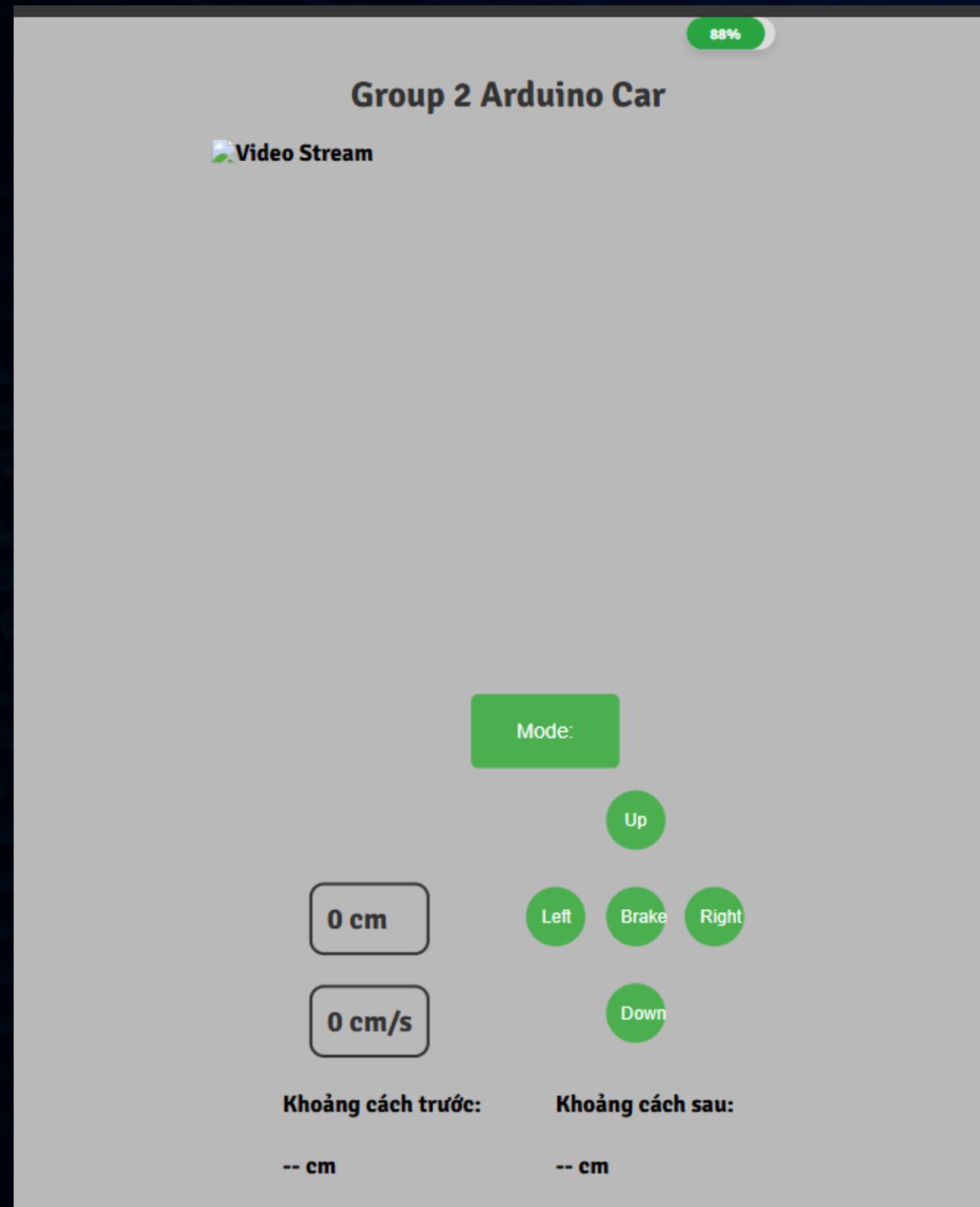


II. Model assembly.

```
51 void setup() {
52     Serial.begin(115200);
53     mySerial.begin(115200, SERIAL_8N1, RX_PIN, TX_PIN);
54
55     // Cấu hình camera
56     camera_config_t config;
57     config.ledc_channel = LEDC_CHANNEL_0;
58     config.ledc_timer = LEDC_TIMER_0;
59     config.pin_d0 = Y2_GPIO_NUM;
60     config.pin_d1 = Y3_GPIO_NUM;
61     config.pin_d2 = Y4_GPIO_NUM;
62     config.pin_d3 = Y5_GPIO_NUM;
63     config.pin_d4 = Y6_GPIO_NUM;
64     config.pin_d5 = Y7_GPIO_NUM;
65     config.pin_d6 = Y8_GPIO_NUM;
66     config.pin_d7 = Y9_GPIO_NUM;
67     config.pin_xclk = XCLK_GPIO_NUM;
68     config.pin_pclk = PCLK_GPIO_NUM;
69     config.pin_vsync = VSYNC_GPIO_NUM;
70     config.pin_href = HREF_GPIO_NUM;
71     config.pin_sccb_sda = SIOD_GPIO_NUM;
72     config.pin_sccb_scl = SIOC_GPIO_NUM;
73     config.pin_pwdn = PWDN_GPIO_NUM;
74     config.pin_reset = RESET_GPIO_NUM;
75     config.xclk_freq_hz = 20000000;
76     config.pixel_format = PIXFORMAT_JPEG;
77
78     // Đặt độ phân giải thành 320x240 (QVGA) và chất lượng ảnh
79     config.frame_size = FRAMESIZE_QVGA; // 320x240
80     config.jpeg_quality = 30; // Chất lượng trung bình, tốc độ nhanh hơn
81     config.fb_count = 1; // Giảm số buffer để tiết kiệm RAM
82
83     esp_err_t err = esp_camera_init(&config);
84     if (err != ESP_OK) {
85         Serial.printf("X Lỗi khởi tạo camera: 0x%x", err);
86         return;
87     }
```

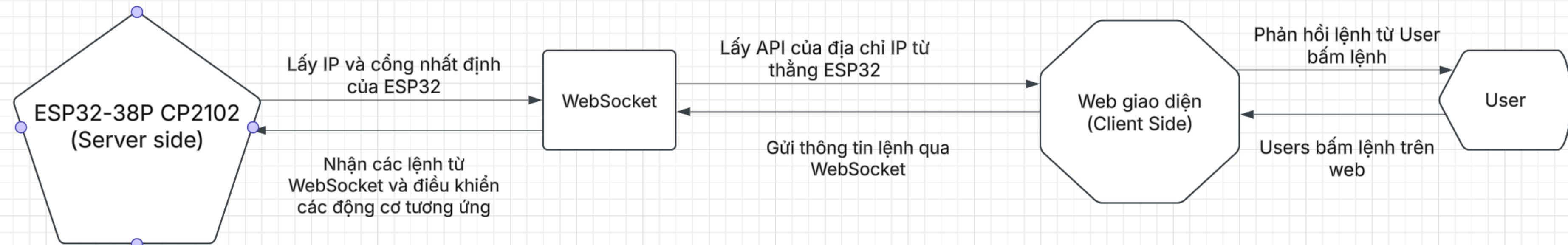


III. Software (Control Car)



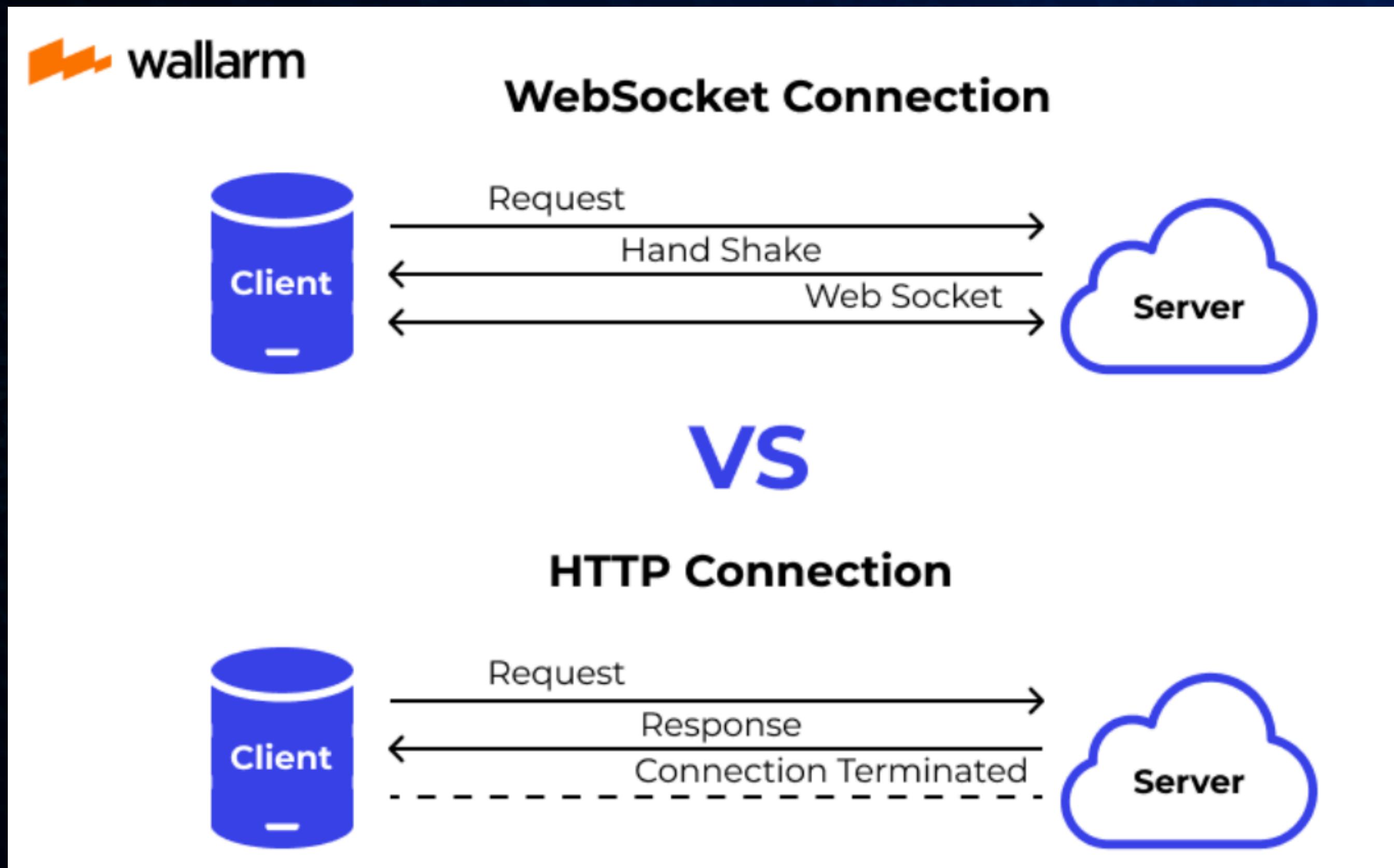


III. Software (Control Car)



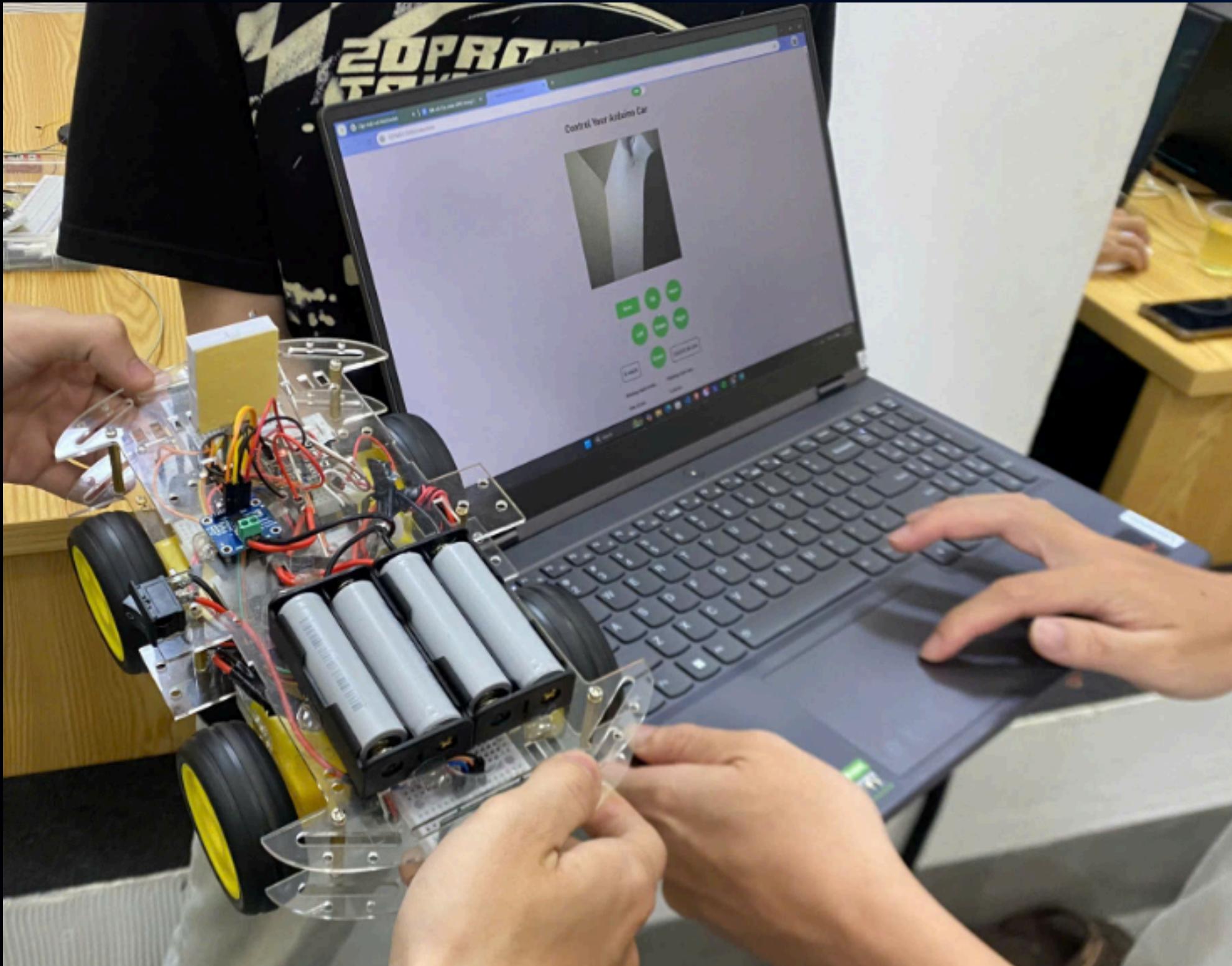


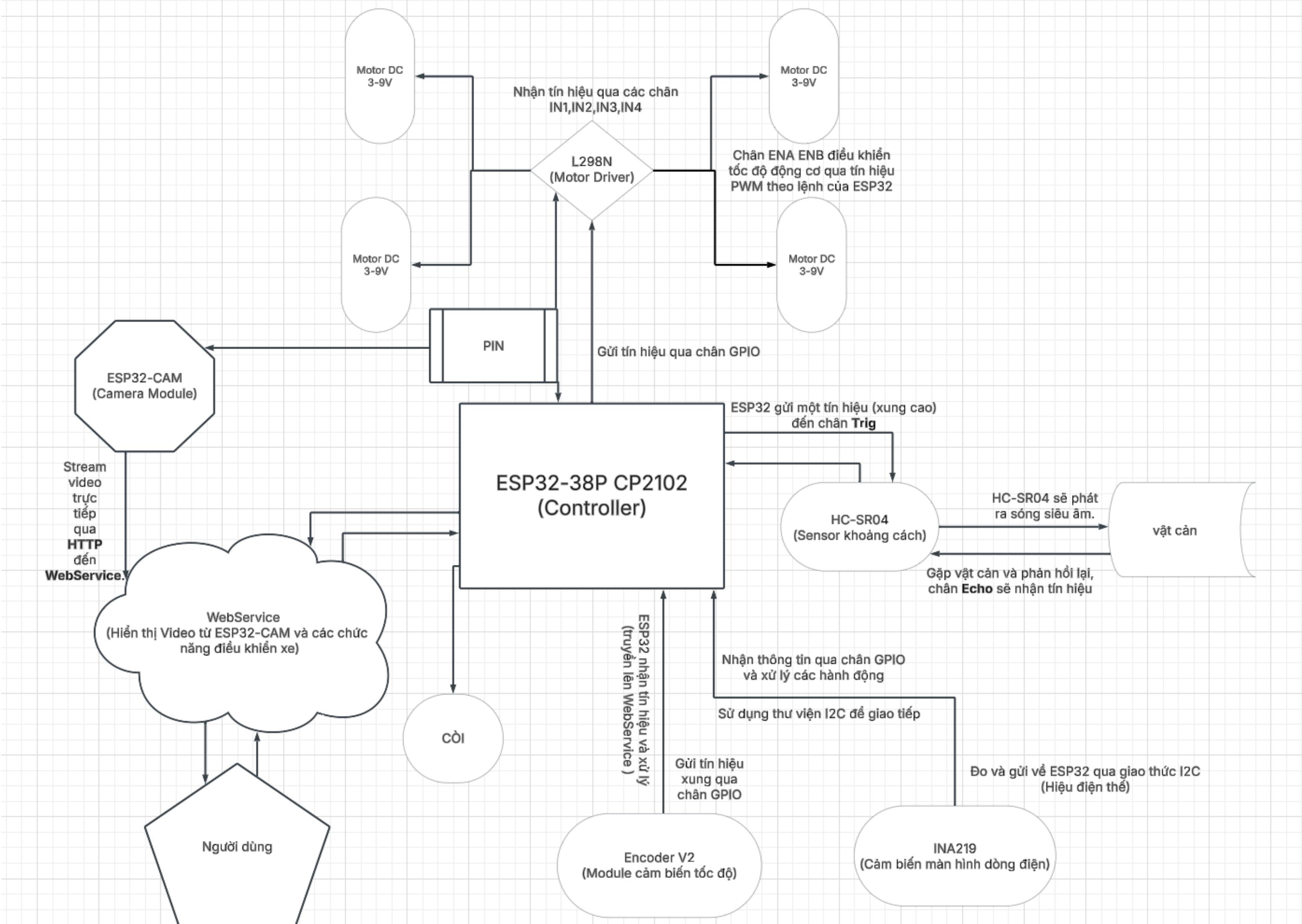
III. Software (Control Car)

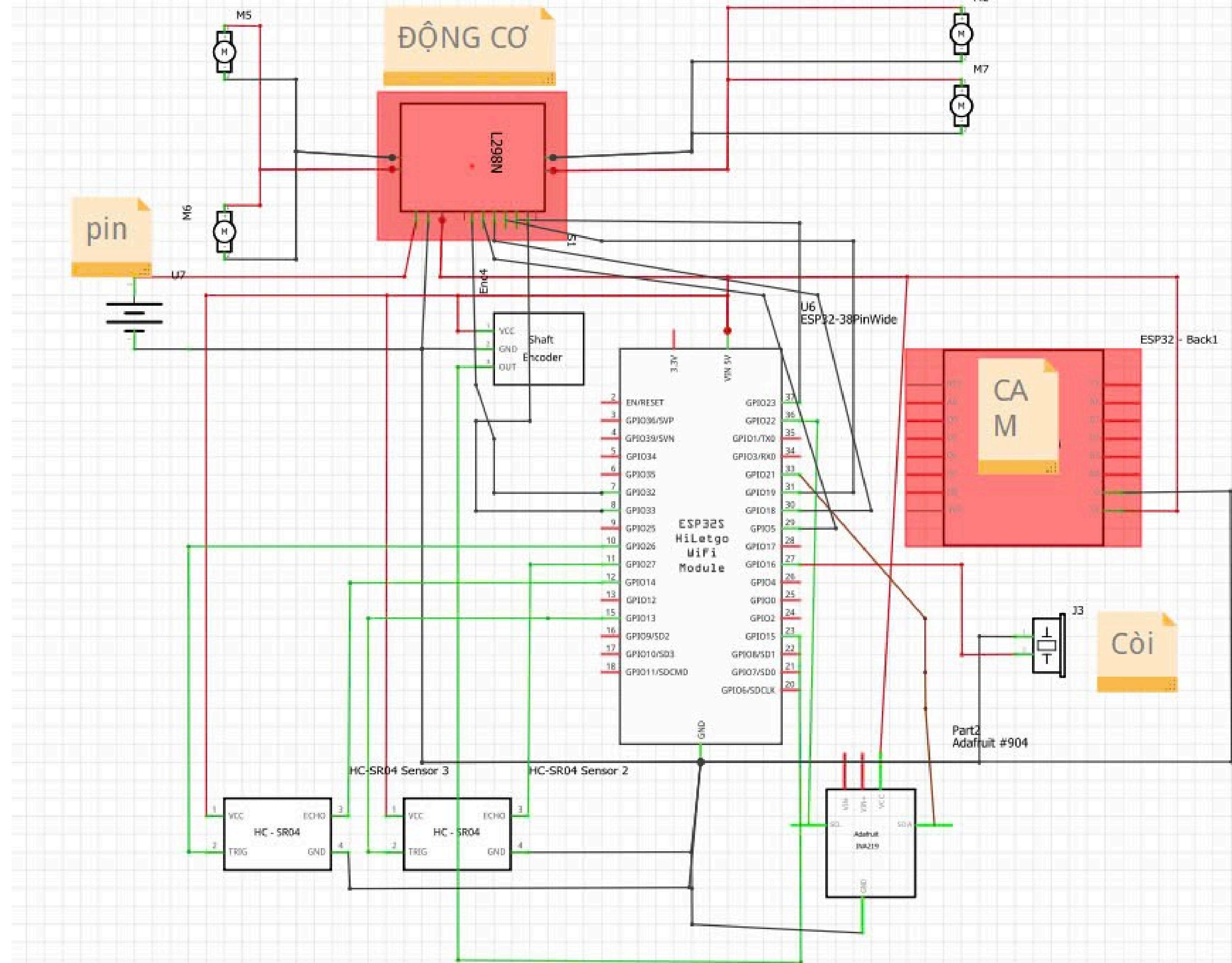


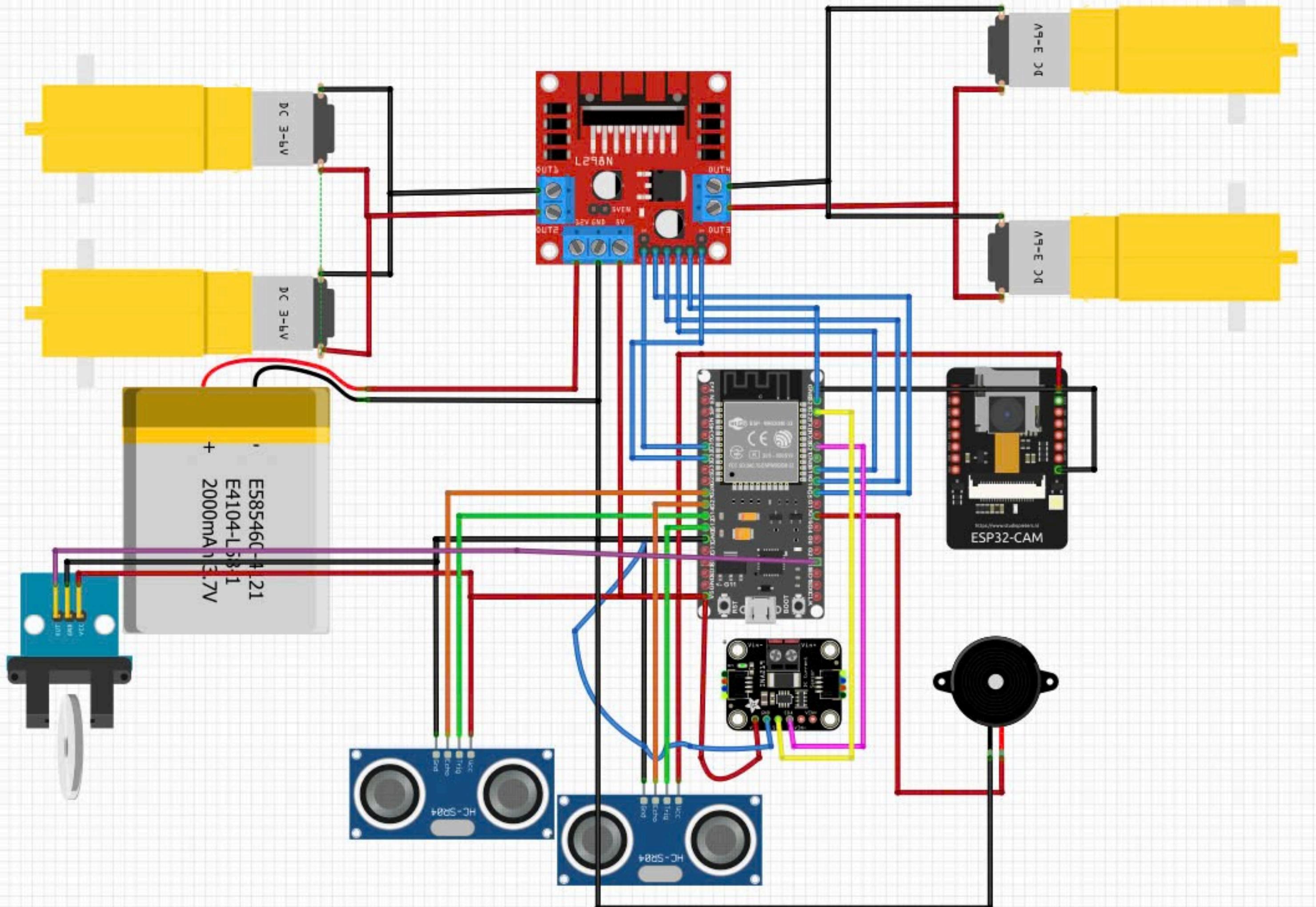


IV. Product









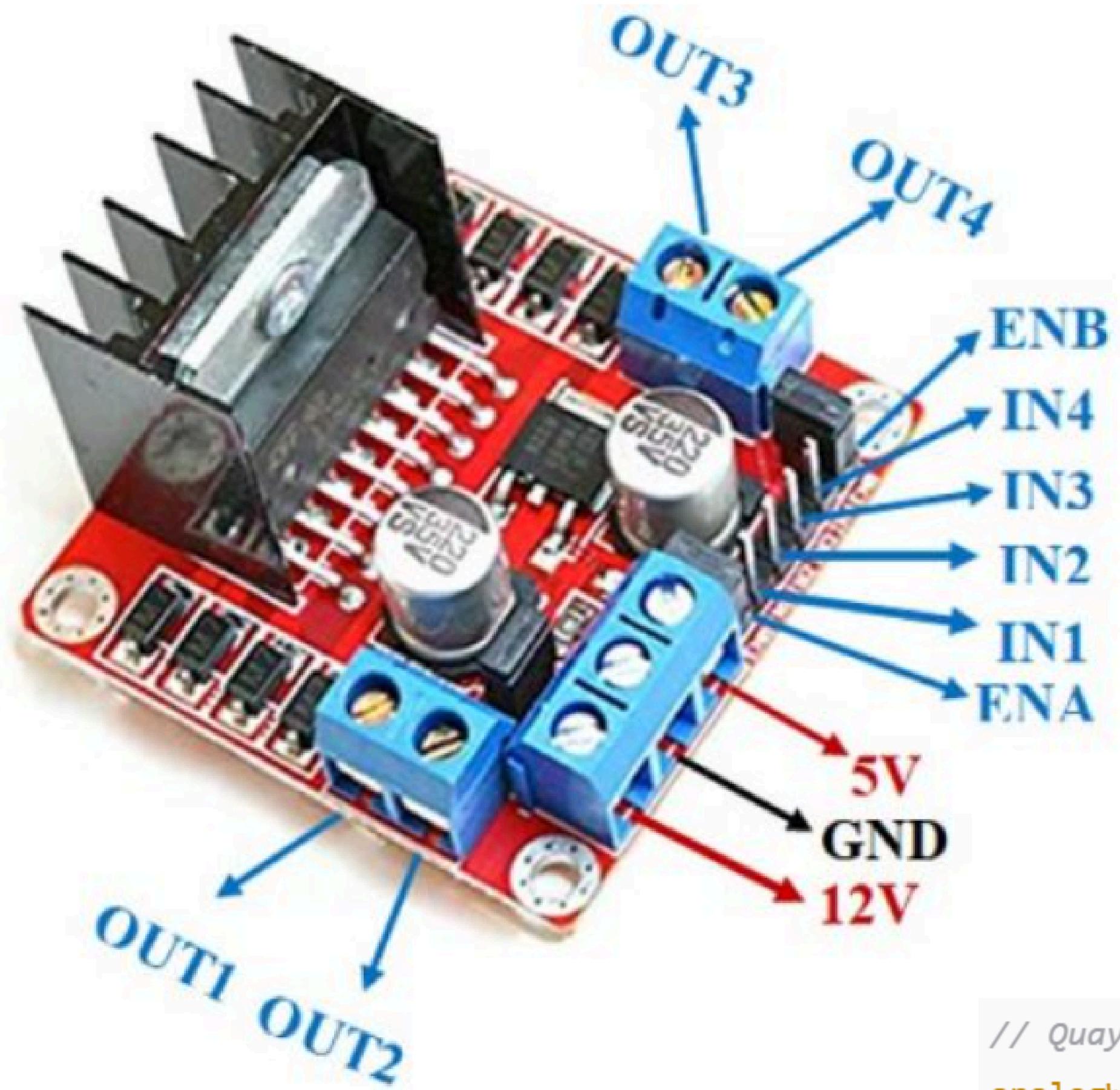


Flow Chart

		3.3V	1
RESET	EN	2	
ADC0	GIOP36	3	
ADC3	GIOP39	4	
ADC6	GIOP34	5	
ADC7	GIOP35	6	
TOUCH9	ADC4	GIOP32	7
TOUCH8	ADC5	GIOP33	8
DAC1	ADC18	GIOP25	9
DAC2	ADC19	GIOP26	10
TOUCH7	ADC17	GIOP27	11
TOUCH6	ADC16	GIOP14	12
TOUCH5	ADC15	GIOP12	13
		GND	14
TOUCH4	ADC14	GIOP13	15
RX1	FLASH D2	GIOP9	16
TX1	FLASH D3	GIOP10	17
		GND	18
		Vin 5V	19



38	GND	
37	GIOP23	VSPI MOSI
36	GIOP22	I2C SCL
35	GIOP1	TX0
34	GIOP3	RX0
33	GIOP21	I2C SDA
32	GND	
31	GIOP19	VSPI MISO
30	GIOP18	VSPI SCK
29	GIOP5	VSPI SS
28	GIOP17	TX2
27	GIOP16	RX2
26	GIOP4	ADC10
25	GIOP0	ADC11
24	GIOP2	ADC12
23	GIOP15	ADC13
22	GIOP8	FLASH D1
20	GIOP7	FLASH D0
20	GIOP6	FLASH CK



L298N

**Điều chỉnh tốc độ
Điện áp và dòng điện**

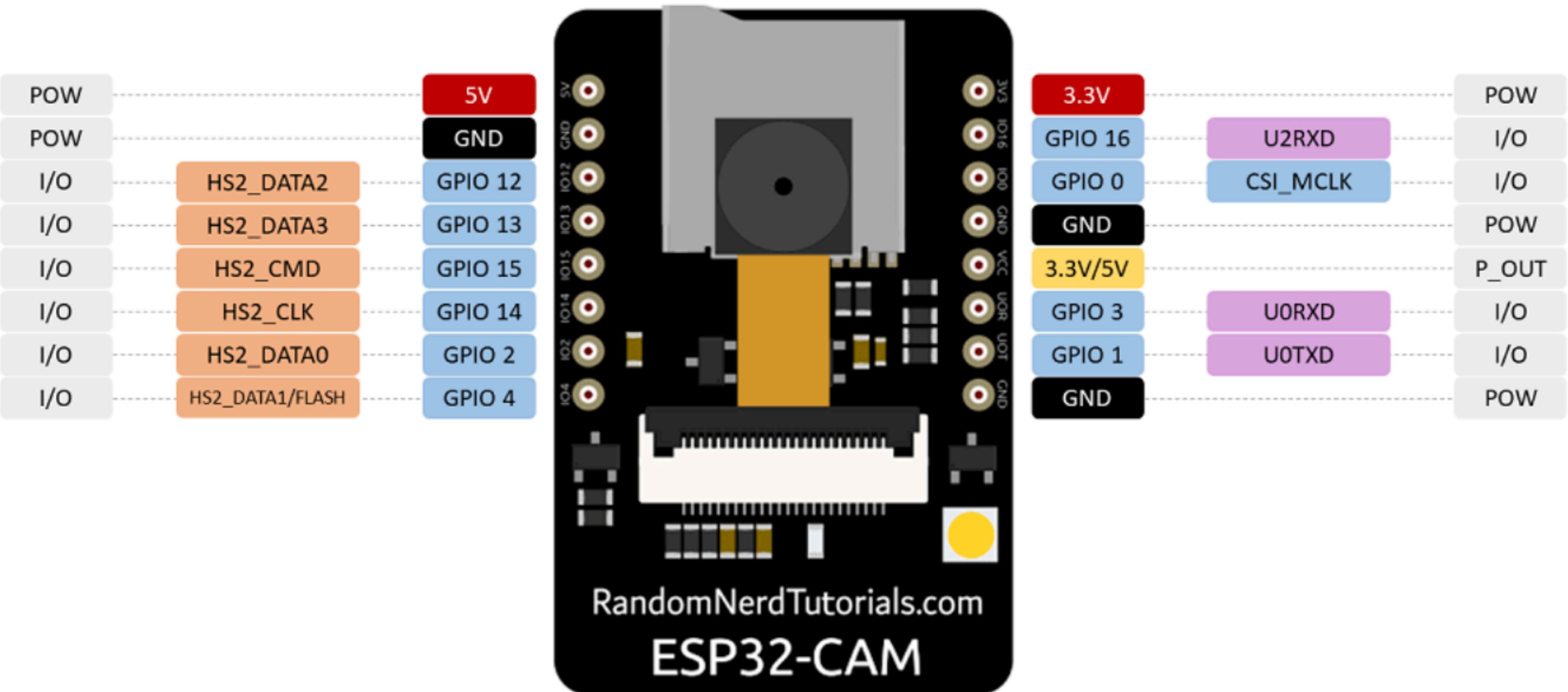
**Điều khiển bằng tín hiệu logic
Điều chỉnh chiều quay**

```
#define IN1 5 // Điều khiển bánh trái
#define IN2 18 // Điều khiển bánh trái
#define IN3 19 // Điều khiển bánh phải
#define IN4 23 // Điều khiển bánh phải
```

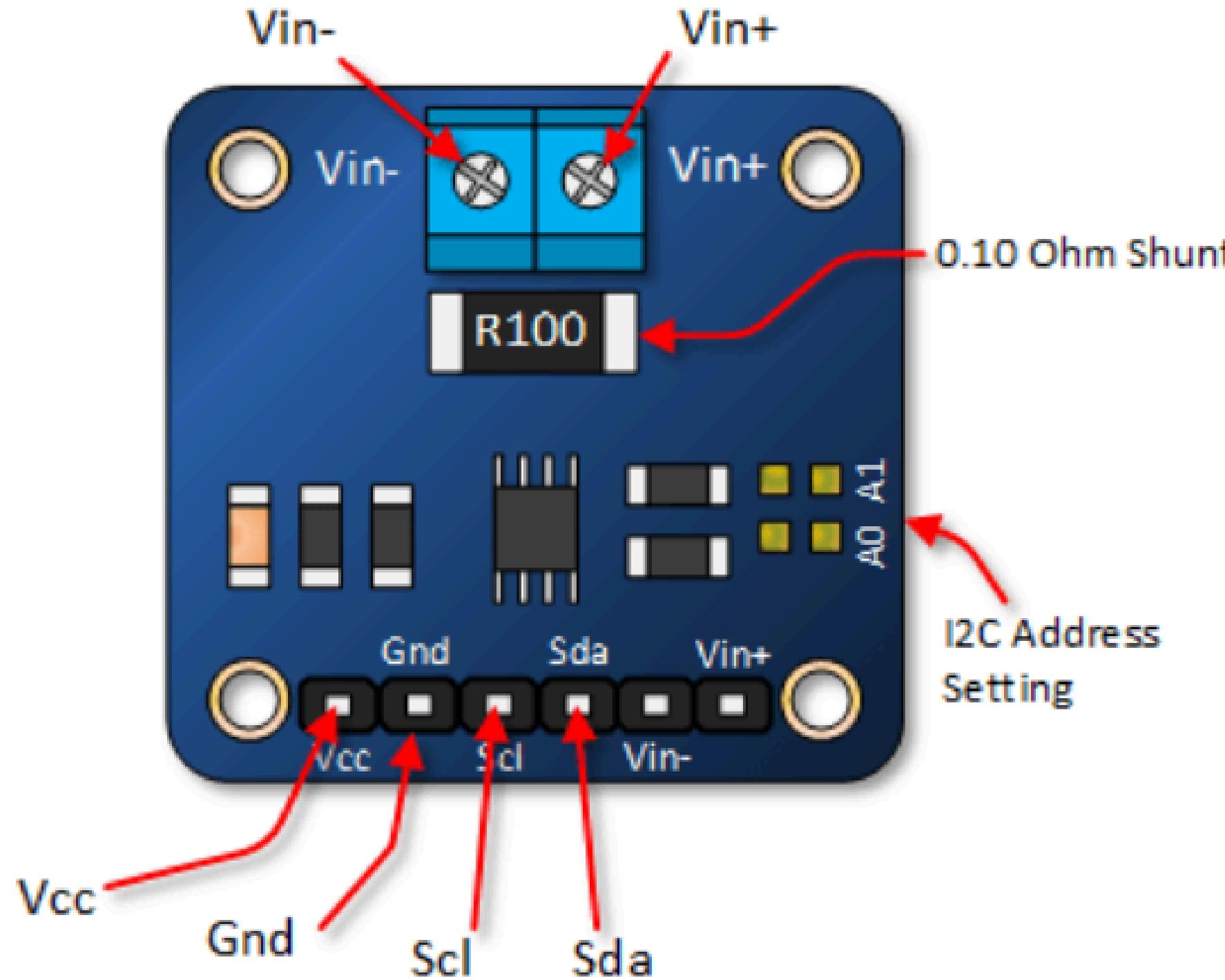
Code động cơ trái quay tiến
`digitalWrite(IN1, LOW);
digitalWrite(IN2, HIGH);`

```
#define ENA 32 // PWM điều khiển động cơ 1
#define ENB 33 // PWM điều khiển động cơ 2
```

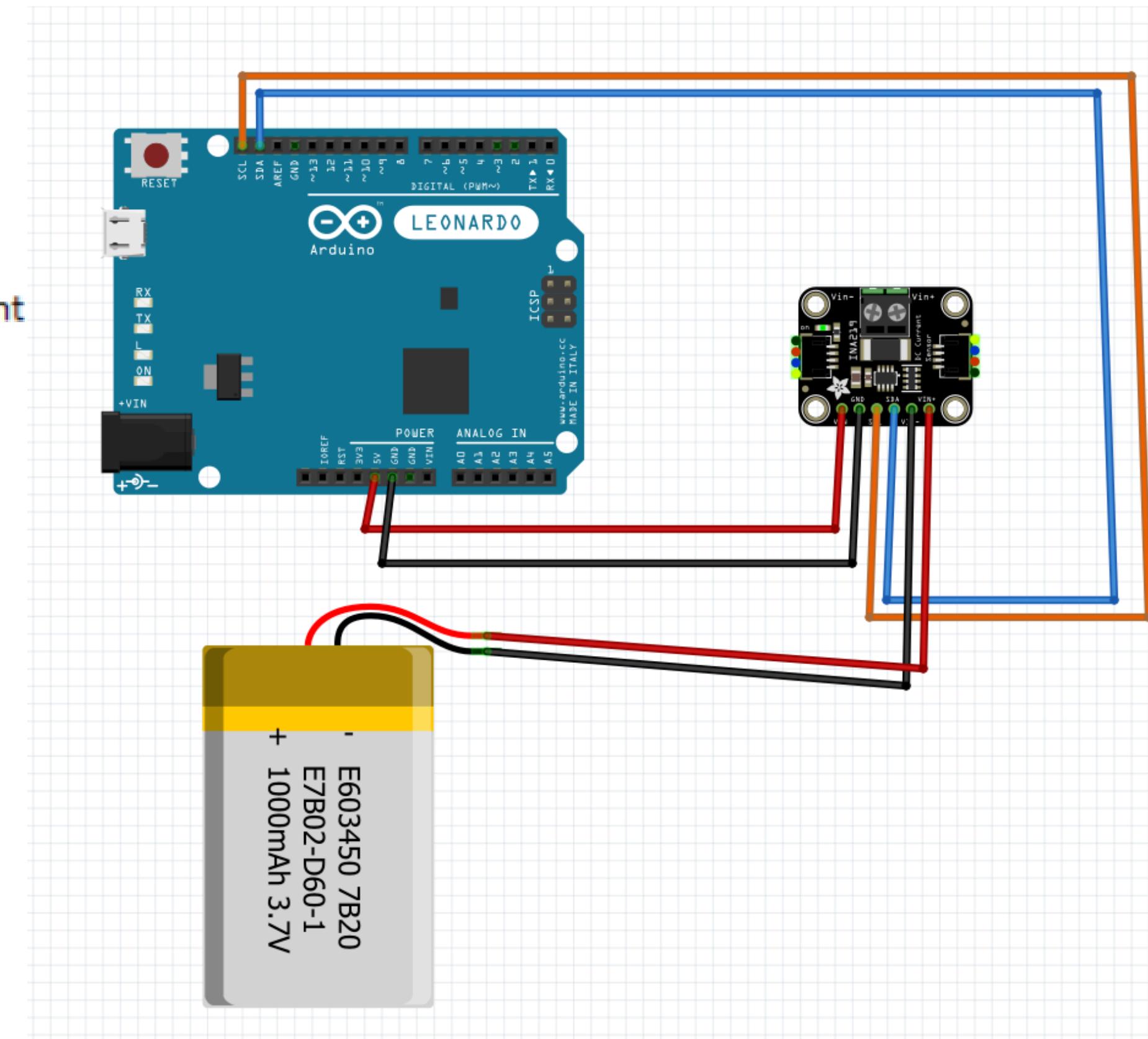
```
// Quay động cơ trái và phải với tốc độ 50% (Duty Cycle = 128)
analogWrite(ENA, 128); // 50% tốc độ động cơ trái
analogWrite(ENB, 128); // 50% tốc độ động cơ phải
```



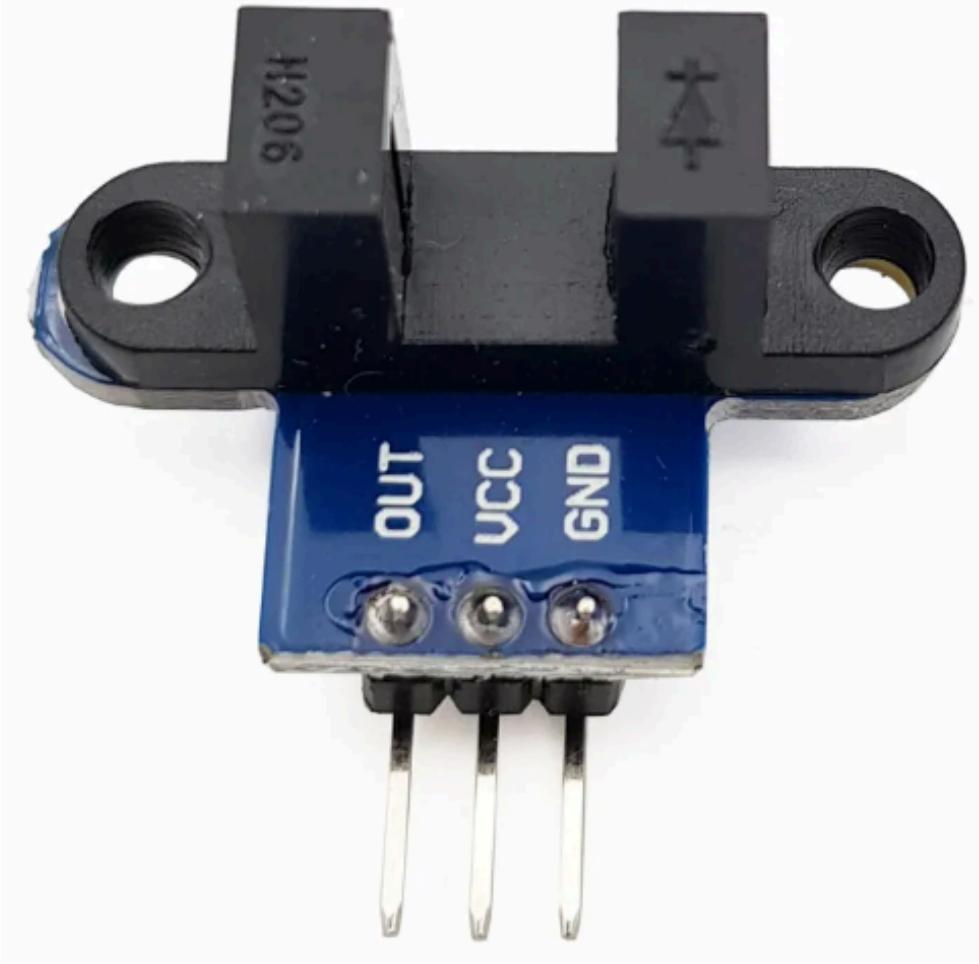
ESP32-CAM



INA219



Sơ đồ kết nối



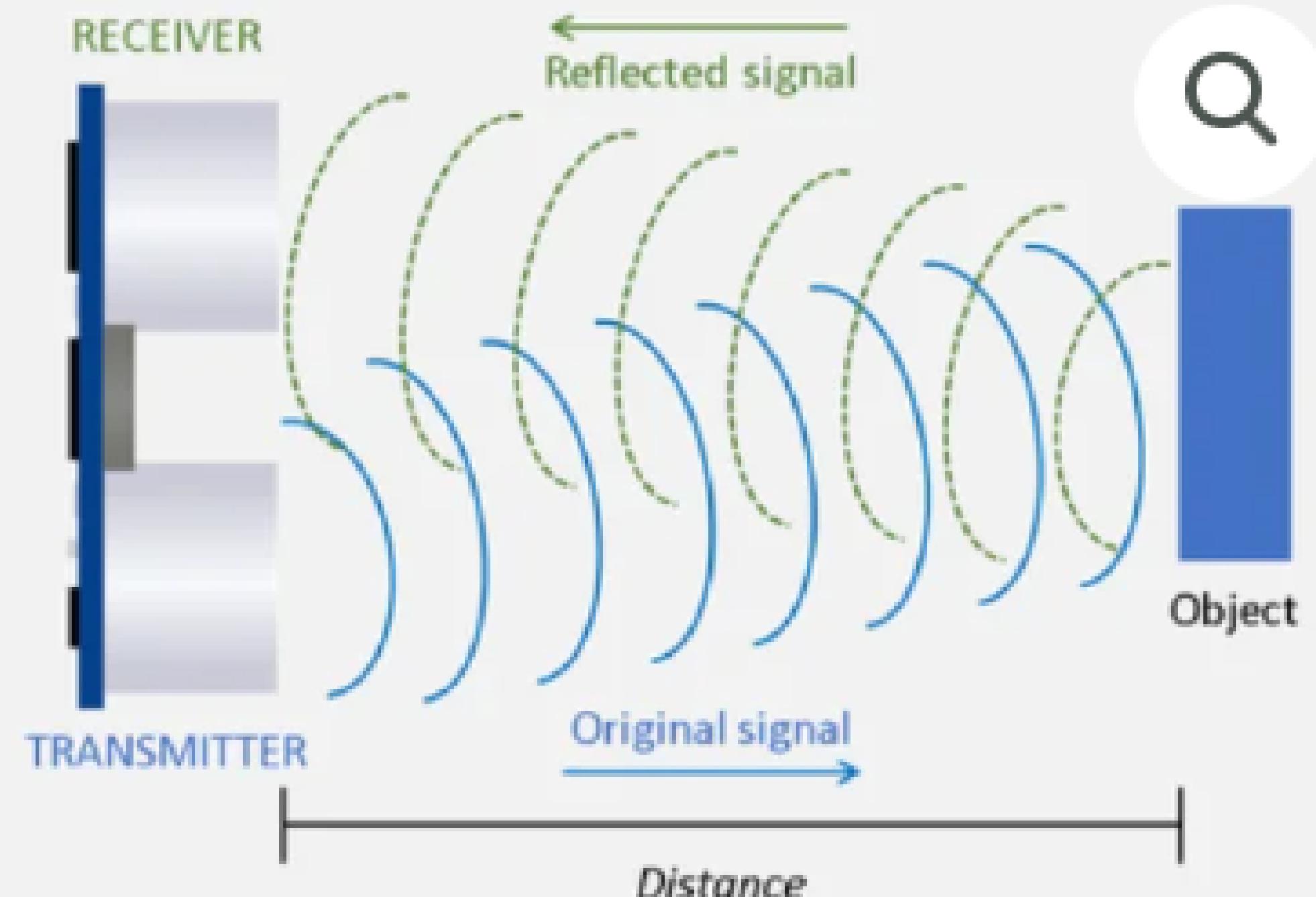
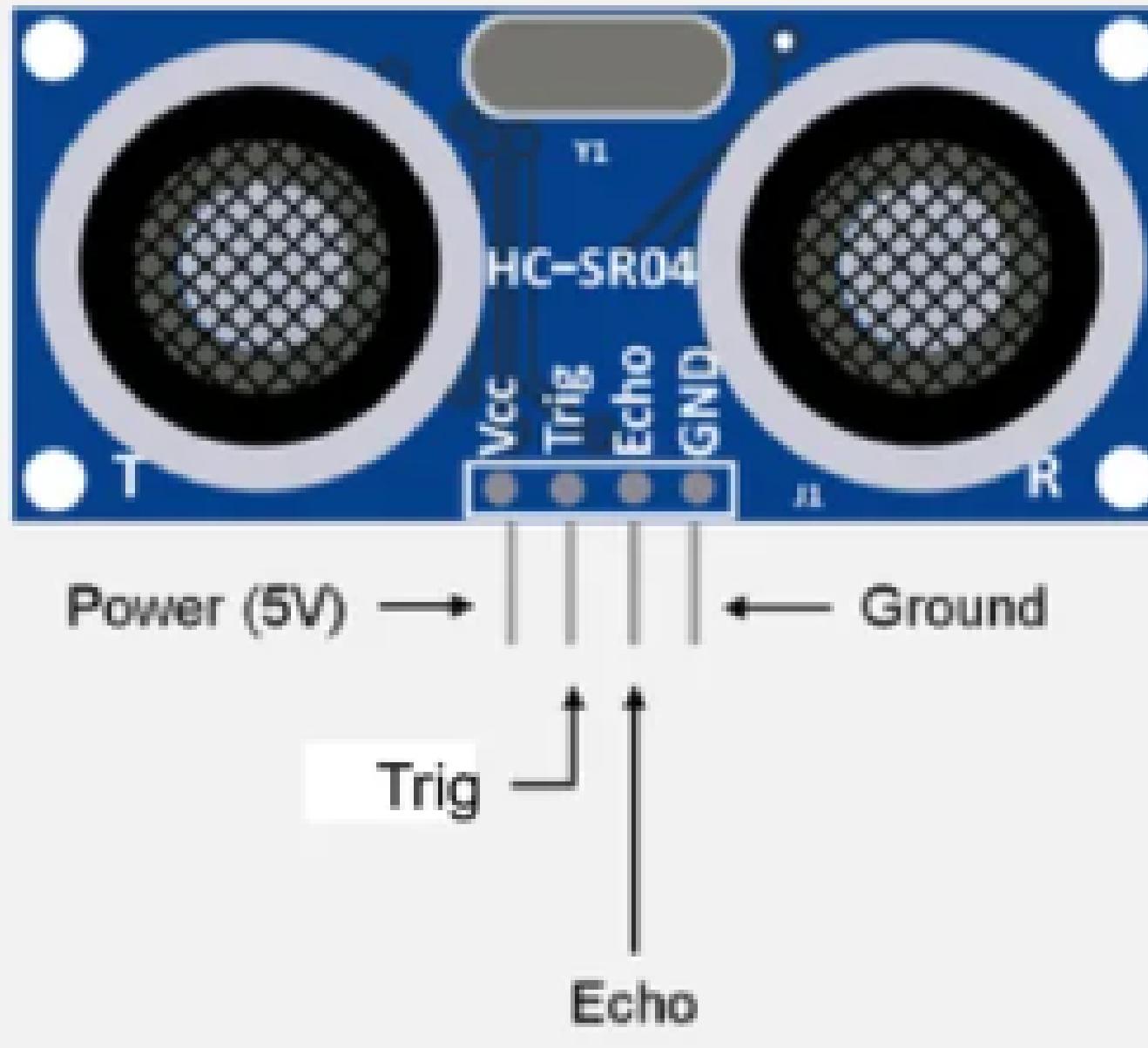
Chân Encoder V2

Tốc độ và Quãng đường:

- **Tốc độ (V)** được tính dựa trên số vòng quay trong một khoảng thời gian (ví dụ 1 giây). Công thức tính: $V = \frac{\text{Số vòng quay} \times \text{Chu vi bánh xe}}{\text{Thời gian}}$.
 - **Quãng đường (S)** là tổng số vòng quay nhân với chu vi bánh xe, và được tính theo công thức: $S = \text{Số vòng quay} \times \text{Chu vi bánh xe}$.
-
- Encoder đếm số xung và tính toán số vòng quay. (Trong 1 thời gian)
 - Dựa vào số vòng quay và chu vi bánh xe, tính được quãng đường di chuyển và tốc độ của xe trong thời gian thực.



Bánh răng



$$\text{Distance} = \frac{\text{Time}}{2} \times \text{Speed of Sound}$$

- **Time** là thời gian từ khi phát sóng đến khi nhận lại.
- **Speed of Sound** là vận tốc của sóng siêu âm trong không khí (khoảng 343 m/s ở nhiệt độ phòng).



THANK YOU

FOR YOUR ATTENTION