PRESENTED BY :

Yuchen Eric LIANG: yuchen.liang@connect.ust.hk
Yucheng Simon LUO: yluobl@connect.ust.hk

DC
MOTOR
CONTROL

OBJECTIVE

In this lab, you will learn how to control a motor remotely using your mobile phone.

LAB
OBJECTIVE
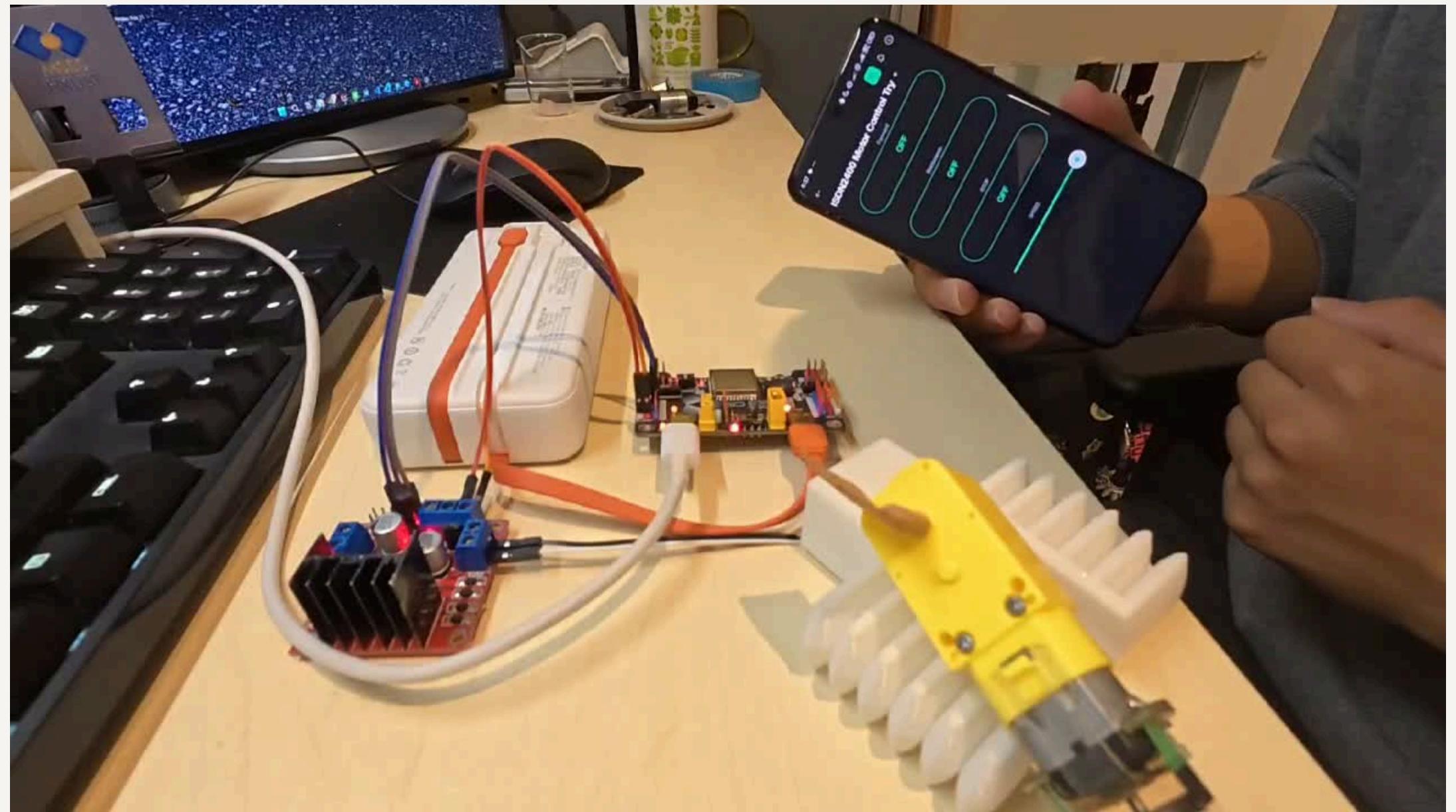
REVISION
BRUSHED
DC MOTOR

Motor performance characteristics
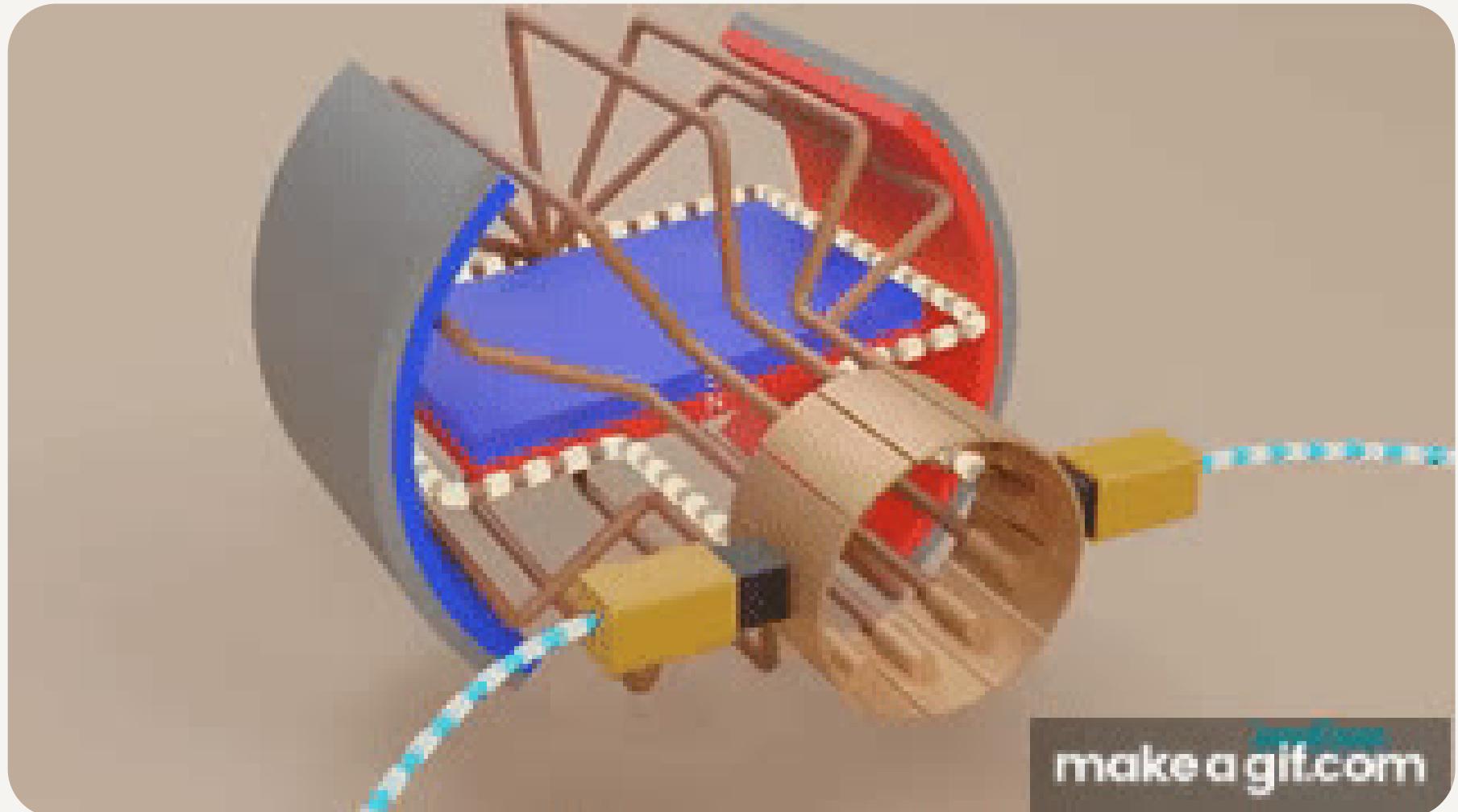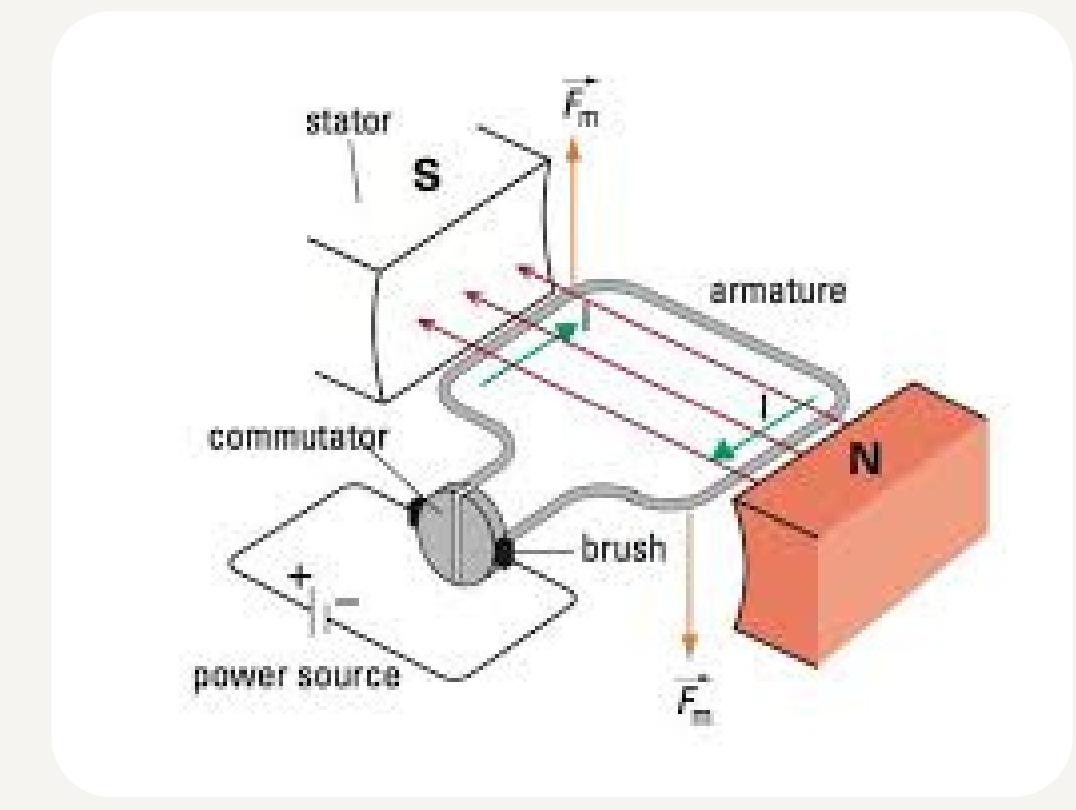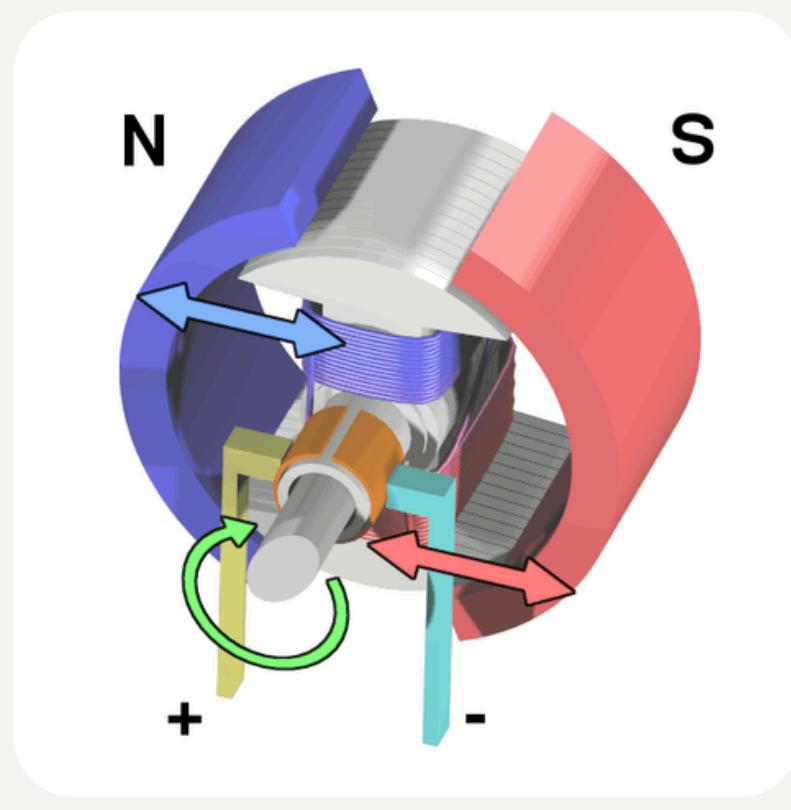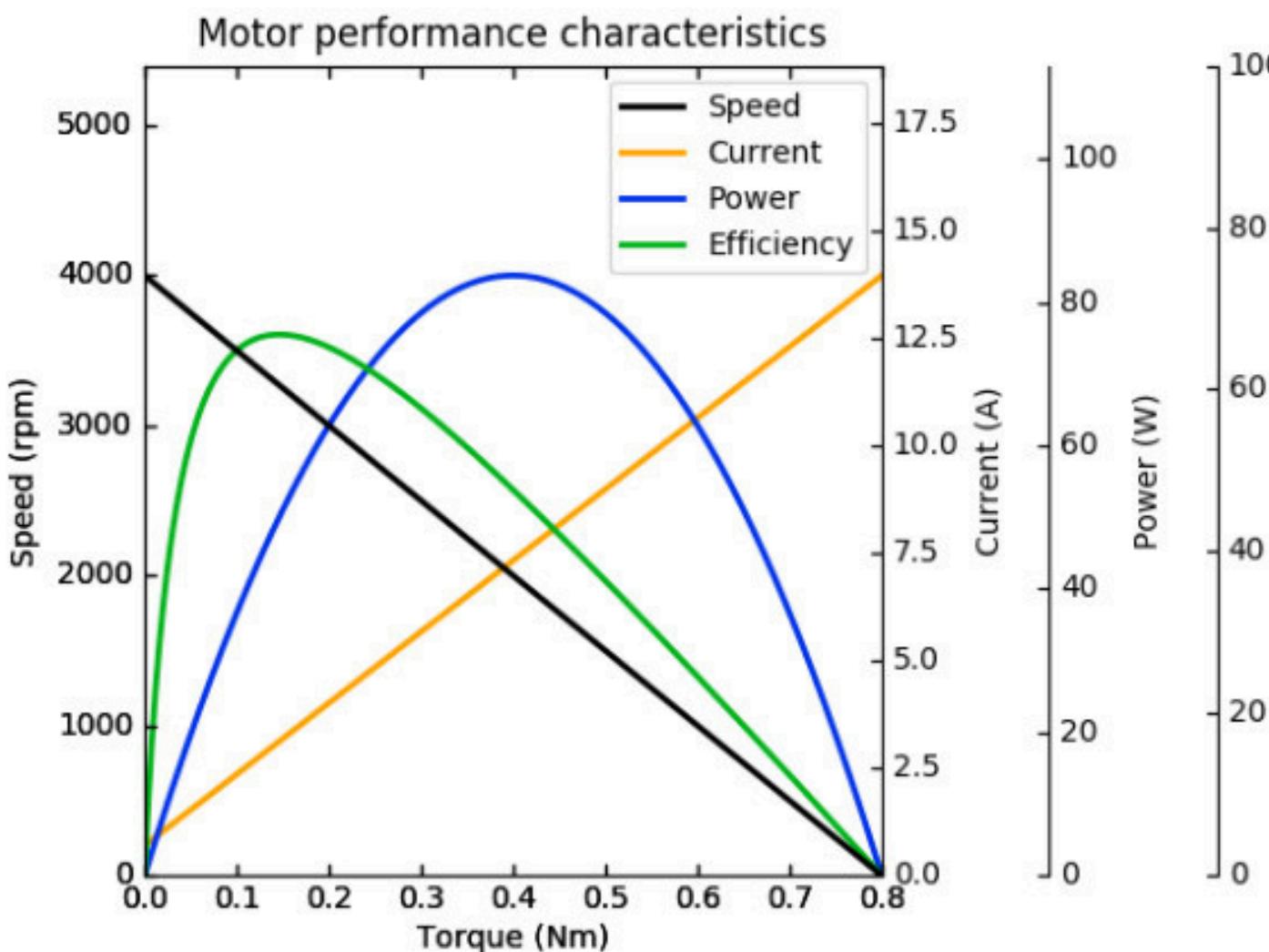
FARS-550 (voltage, current, torque, etc free customization)

典型应用 Typical Applications:		轴长 Shaft Length A(mm)
空气压缩机、电动工具、电动按摩器	Air compressor, Cordless power tools, Massager	Customizable

外形尺寸 DIMENSION
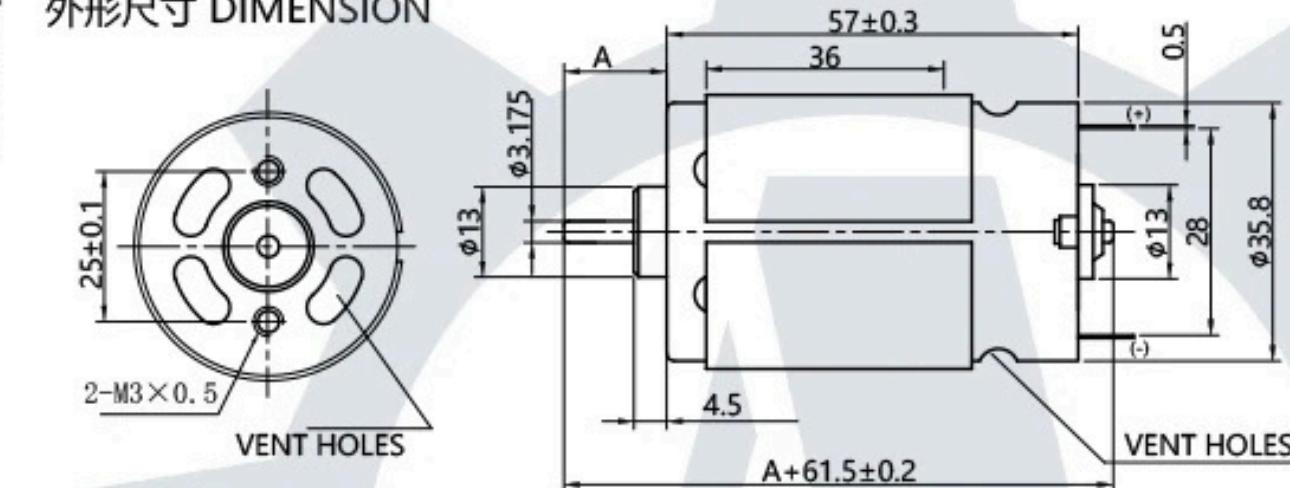
WEIGHT:220g(APPROX)

UNIT: MILLIMETERS

Parameter	Units	Value
Nominal voltage	V	24
No-load speed	rpm	4000
No-load current	A	0.7
Rated speed	rpm	3270
Rated torque	Nm	0.15
Rated current	A	3.1
Stall torque	Nm	0.80
Starting current	A	14.0
Max power	W	84
Max efficiency	%	67
Terminal resistance	Ω	1.7
Torque constant	Nm/A	0.057
Speed constant	rpm/V	167
Back-emf constant	V/rpm	0.006

微型直流有刷电机性能曲线图
Micro brush DC electric motor performance curve

FARS-550SH-6529R 12V
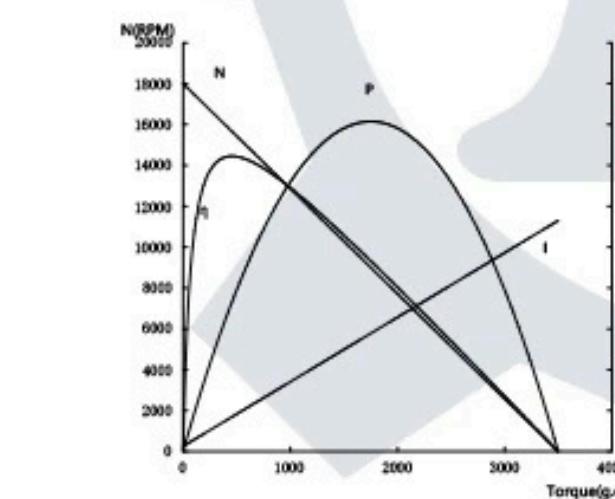
FARS-550PH-27130R 24V
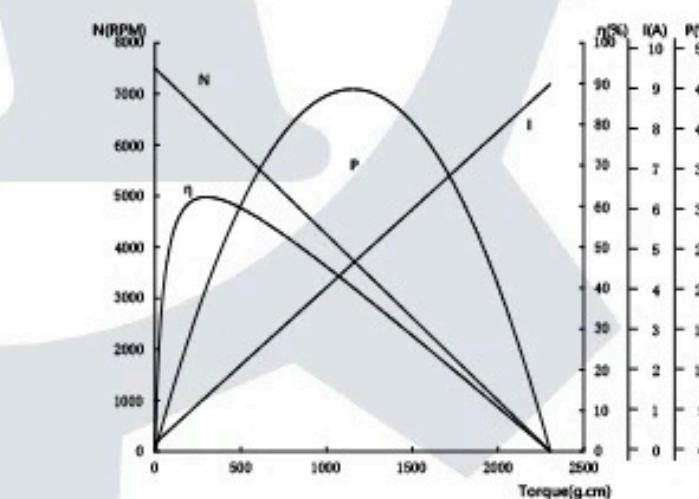
微型有刷直流电机性能参数表 Micro brush DC electric motor data table

型号 Model	电压 Voltage		空载 No Load		额定负载 RATED LOAD					堵转 Stall		
	适用范围 Operating range	额定 Rated	转速 Speed	电流 Current	转速 Speed	电流 Current	扭矩 Torque		输出功率 Output	效率 Efficiency	扭矩 Torque		电流 Current
	VDC		RPM	A	RPM	A	mN.m	gf.cm	W	%	mN.m	gf.cm	A
FARS-550SH-6529R	6.0-14.0	12	18000	1.25	15669	8.4	44.40	452.9	72.86	72.3	343.00	3498.6	56.5
FARS-550PH-27130R	18.0-28.0	24	7500	0.2	6527	1.34	29.30	298.86	20.03	62.3	225.49	2300	9

DC MOTOR
WITH
GEARBOX

REVISION
H-BRIDGE

VCC

+ −

Last Minute
ENGINEERS.com

GND

12V 5V Gnd

L293D

Motor1 Input

Motor2 Input

16 VCC1

8 VCC2

2 INPUT1
7 INPUT2
1 ENABLE1

OUTPUT1 3
OUTPUT2 6

10 INPUT3
15 INPUT4
9 ENABLE2

OUTPUT3 11
OUTPUT4 14

4 GND
5 GND
12 GND
13 GND

1N4001

1N4001

1N4001

1N4001

Motor1

Motor2
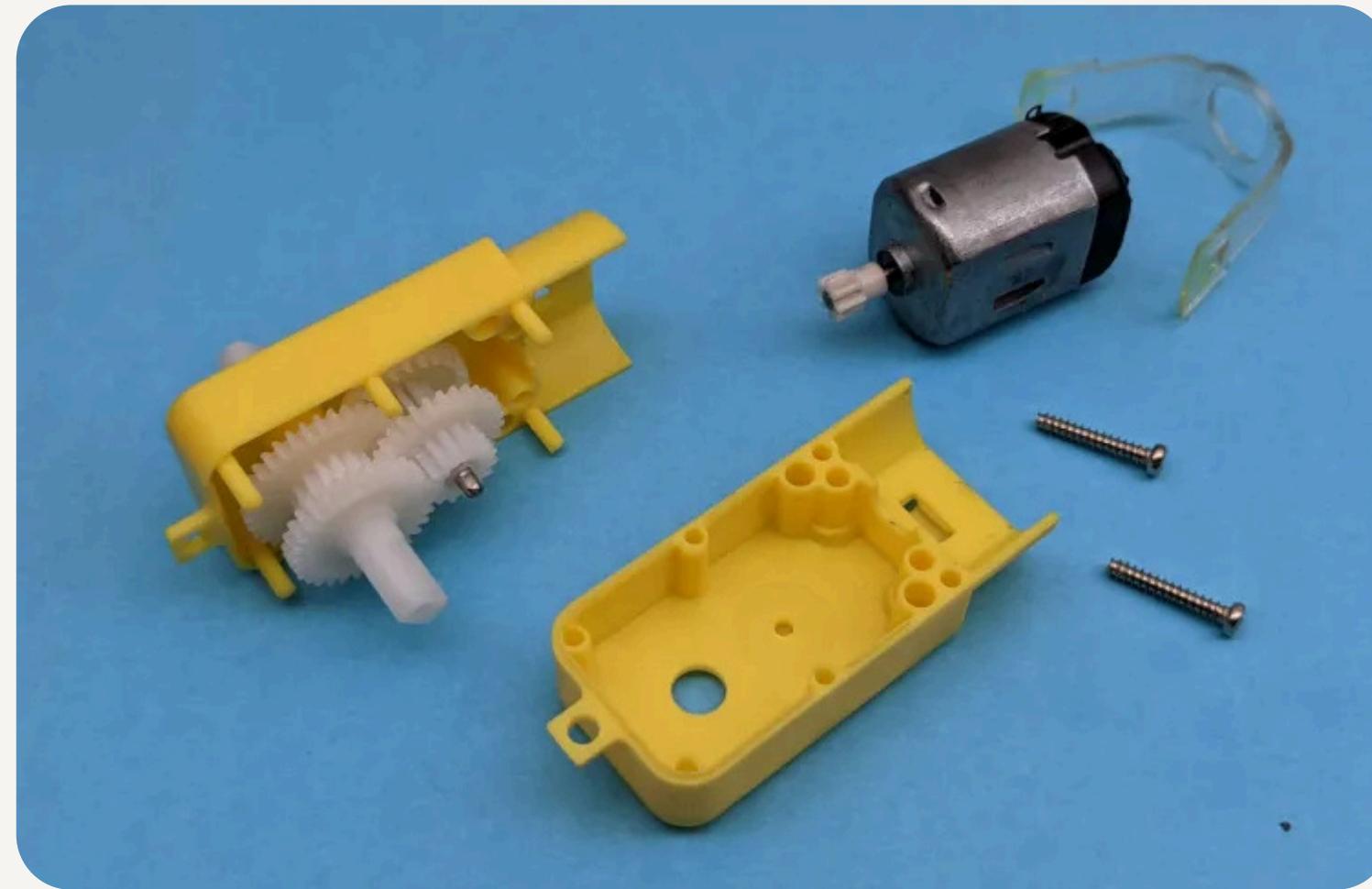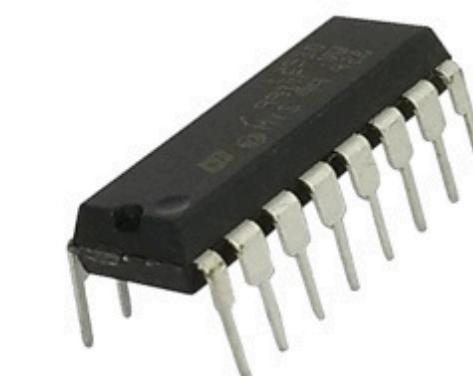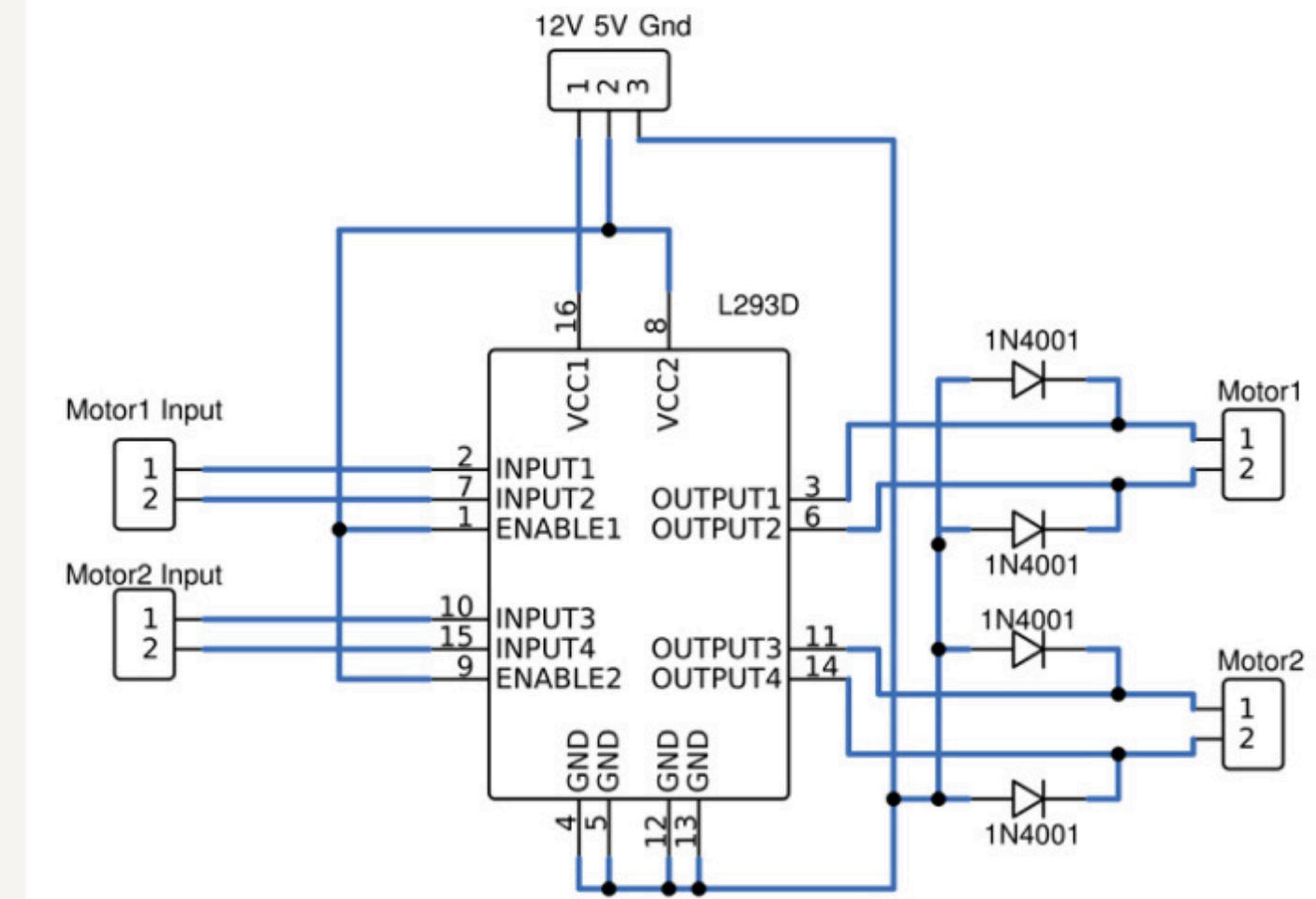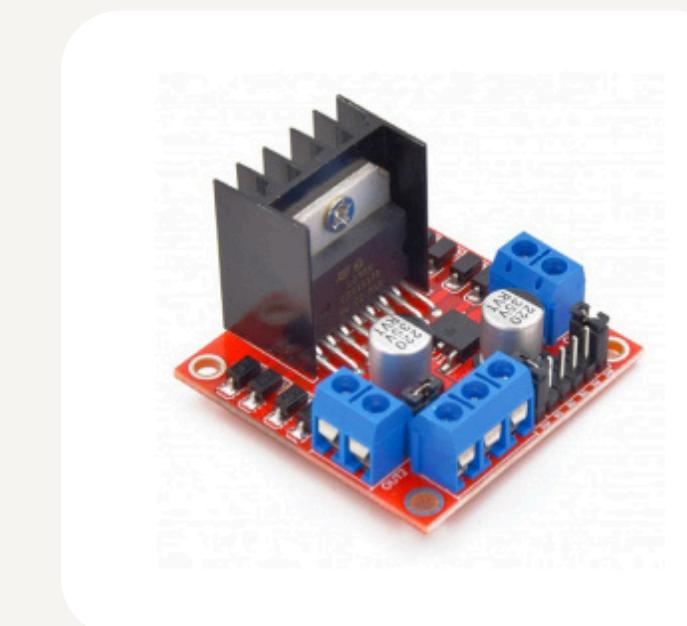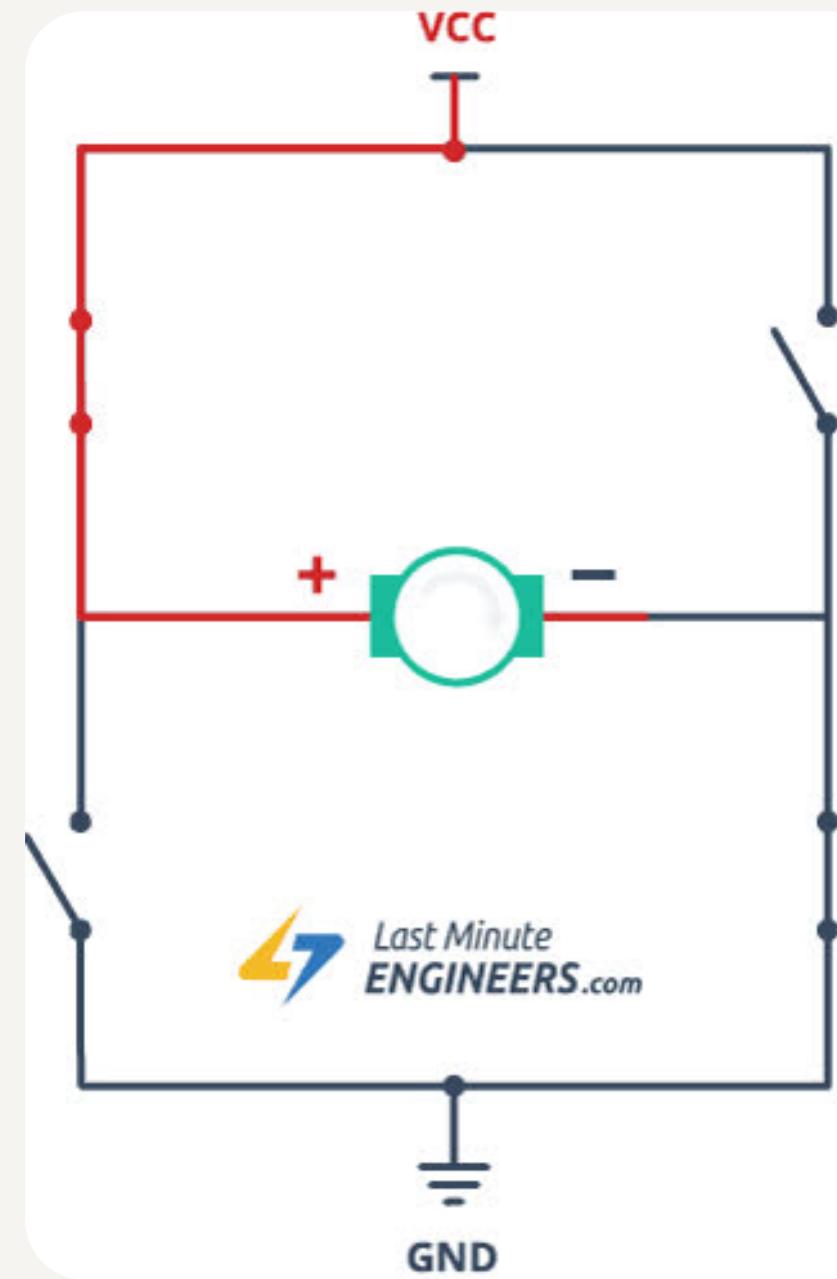
REVISION
ENCODER
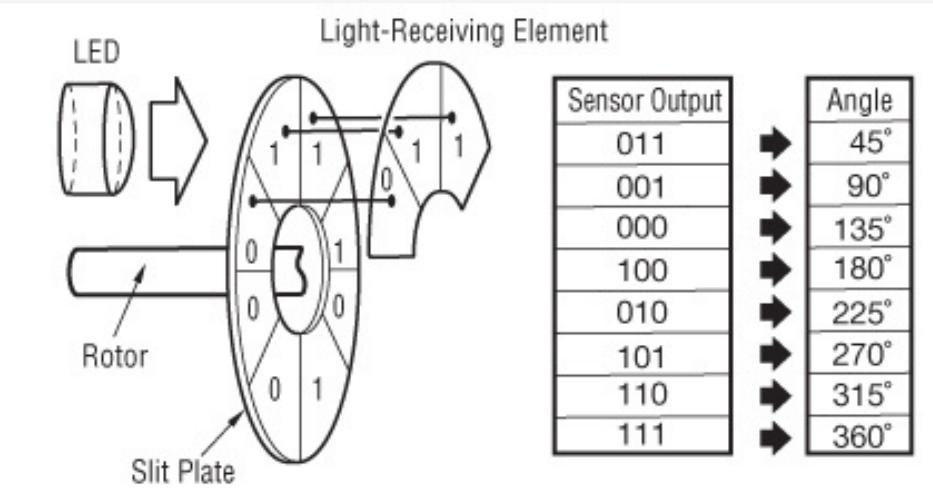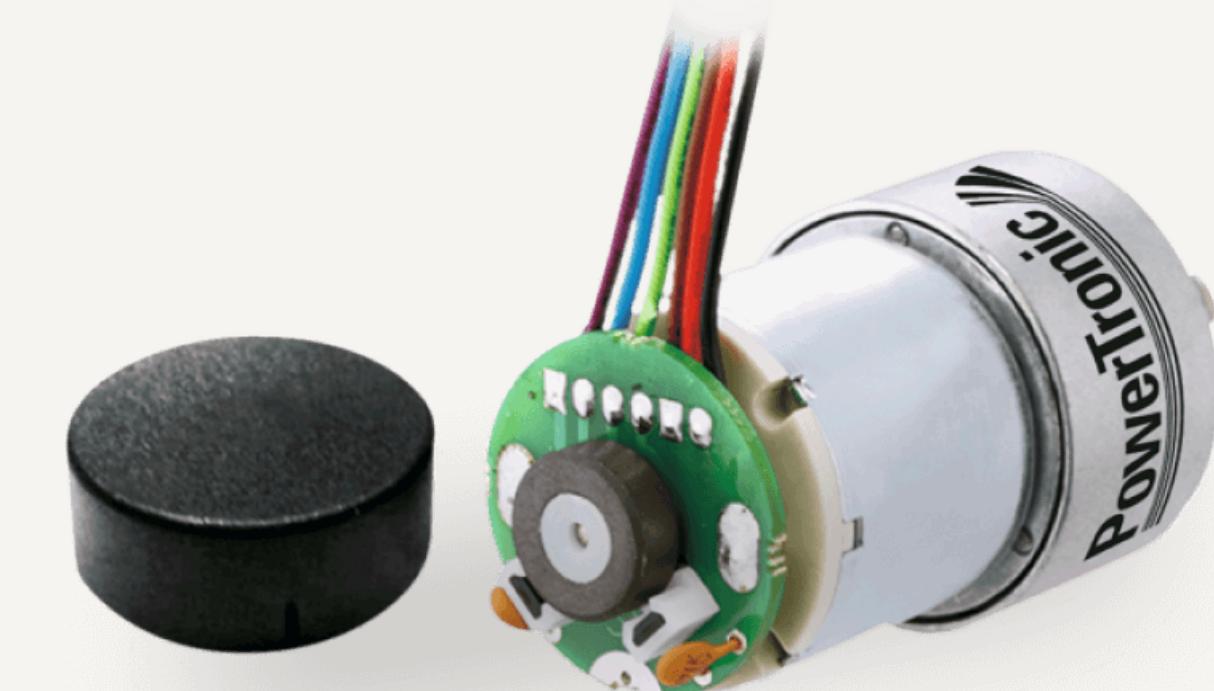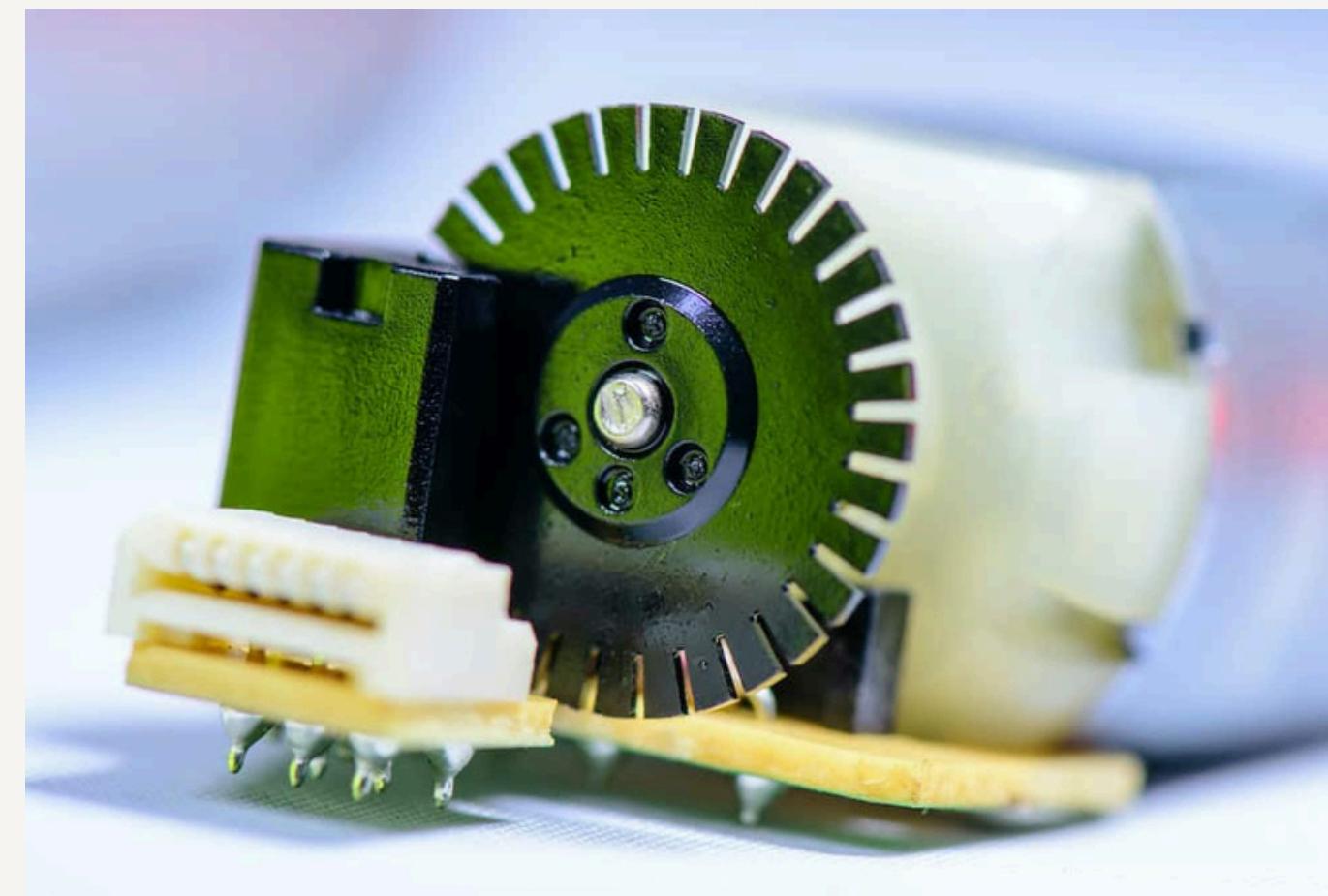
Sensor Output		Angle
011	➡	45°
001	➡	90°
000	➡	135°
100	➡	180°
010	➡	225°
101	➡	270°
110	➡	315°
111	➡	360°

MICRO CONTROLLER
ARDUINO ESP32
STM32

| Feature	Arduino	ESP32 (What we are using)	STM32
Processor	8-bit ATmega (Most common)	32-bit Dual-Core Xtensa LX6	32-bit ARM Cortex-M
Clock Speed	16-20 MHz	Up to 240 MHz	Up to 480 MHz
Wireless	None (requires add-on)	Built-in WiFi & Bluetooth	None (requires add-on)
Memory	2-8KB RAM, 32KB Flash	520KB RAM, 4MB Flash	Up to 512KB RAM, 2MB Flash
Ease of Use	Very Easy	Moderate	Complex
Cost	Low	Low	Moderate
Best For	Beginners, Basic Projects	IoT Projects, Wireless Applications	Industrial, Professional Use
Programming	Arduino IDE, Simple C++	Arduino IDE, ESP-IDF, Multiple Frameworks	Multiple IDEs, C/C++

MICRO CONTROLLER
ARDUINO ESP32
STM32

ANTENNA

ESP-WROOM-32

CP2102

VOLTAGE REGULATOR

BOOT

RESET/EN

USB INPUT

IOT PLATFORM
BLYNK

Blynk Server

No laptop required

Blynk app

Blynk Libraries

Internet Access

LAB
ACTIVITIES

INSTALL ARDUINO IDE

Setup the environment for coding ESP32 on Arduino

SETUP BLYNK (WEB & PHONE)

Setup the BLYNK web app and phone app.

CODE ESP32

Code the ESP32

CONNECT WIRES
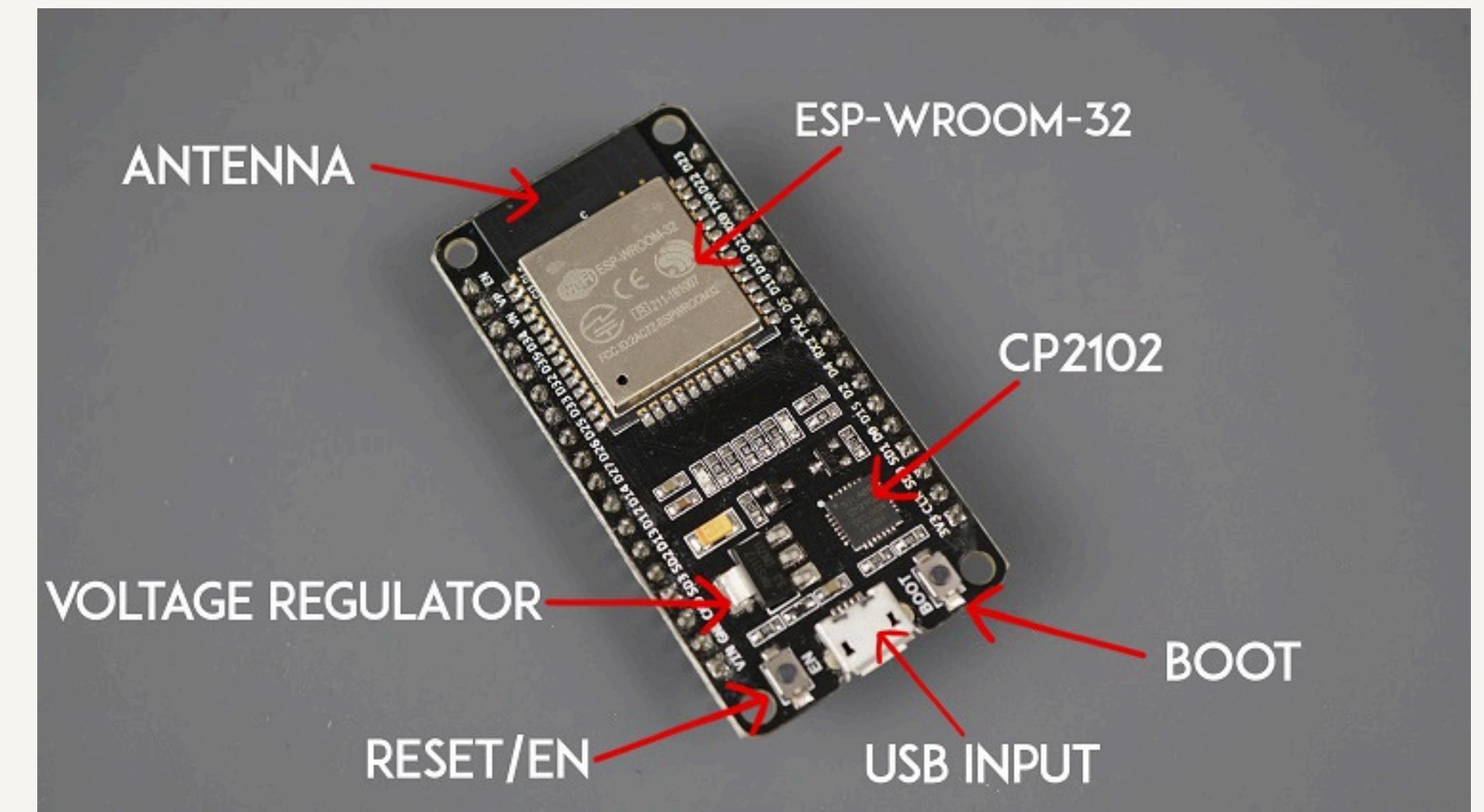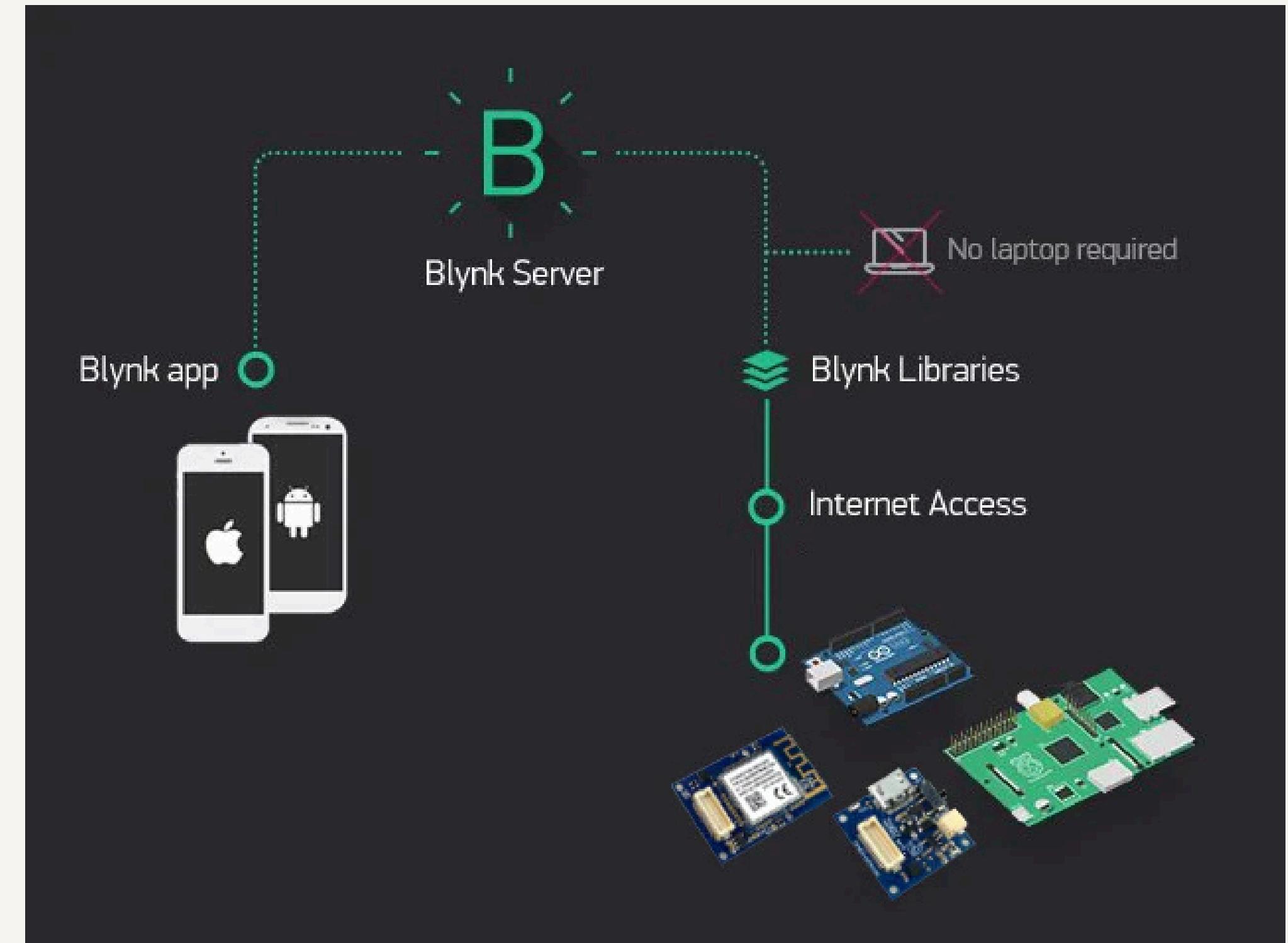
Connect all the necessary wires between ESP32 board and the motors
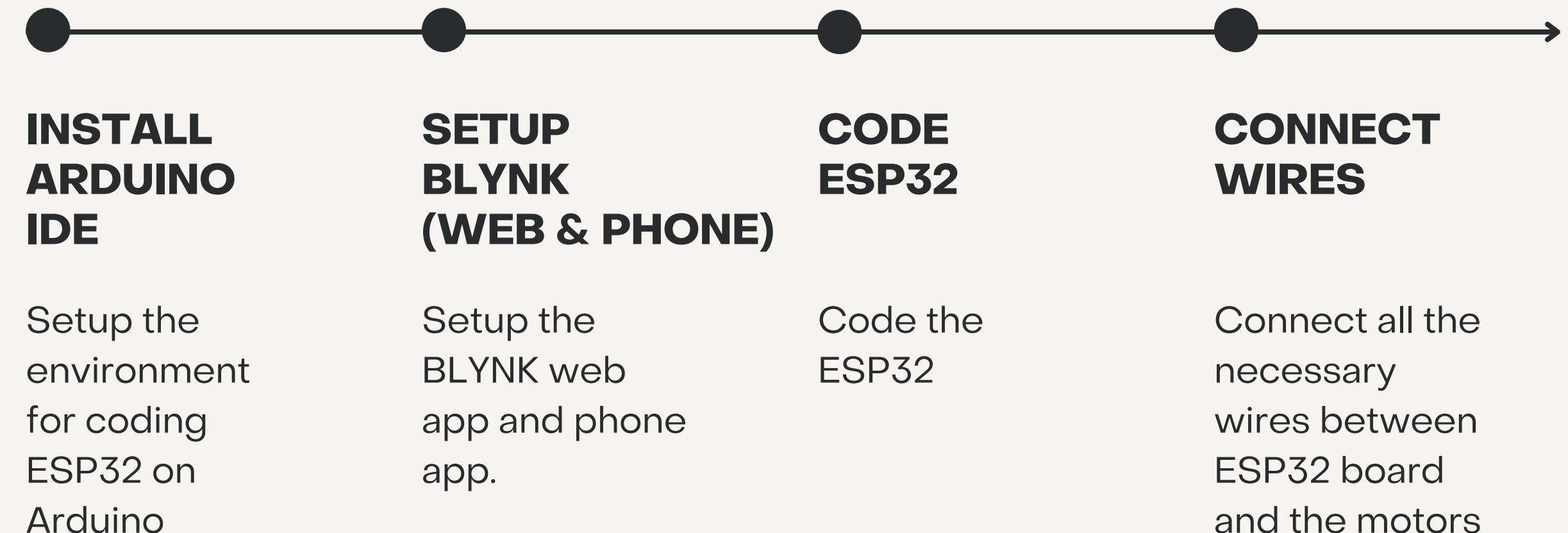
SETUP
ARDUINO IDE ENVIRONMENT

software

SETUP
BLYNK
(WEB)

SETUP
BLYNK
(PHONE APP)

CODING
ESP32

```cpp
#define BLYNK_PRINT Serial

#define BLYNK_TEMPLATE_ID " YOUR TEMPLATE ID " // Template ID in Developer Zone -> Template -> Firmware Configuration
#define BLYNK_TEMPLATE_NAME " YOUR TEMPLATE NAME " // Template Name in Developer Zone -> Template -> Firmware Configuration
#define BLYNK_AUTH_TOKEN " YOUR TOKEN " // Authorization Token in Devices -> Device

#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

char auth[] = BLYNK_AUTH_TOKEN;

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = ""; // Input the name of the WIFI here (You can use your phone's hotspot)
char pass[] = ""; // Input the Passwork of the WIFI here

int Speed_Pin ;
//Motor Driver Pins
const int IN1 = 47;
const int IN2 = 48;
int motorSpeed = 0;
bool stateIN1 = 0;
bool stateIN2 = 0;
```

```
void setup ()
{
  Serial.begin(115200);
  // Motor
  pinMode(IN1, OUTPUT);  // Set the pin as OUTPUT
  pinMode(IN2, OUTPUT);  // Set the pin as OUTPUT

  Blynk.begin(auth, ssid, pass);

}
void loop ()
{
  Blynk.run();

}
```

```
BLYNK_WRITE(V0) // Forward
{
  int pinValue1=param.asInt();
  if (pinValue1==HIGH)
  {
    analogWrite(IN1, motorSpeed);
    analogWrite(IN2, 0);
    stateIN1 = 1;
    stateIN2 = 0;
  }
  if (pinValue1==LOW)
  {

    analogWrite(IN1, 0);
    analogWrite(IN2, 0);
    stateIN1 = 0;
    stateIN2 = 0;

  }
}
```

```
BLYNK_WRITE(V1) // Backward
{
  int pinValue2=param.asInt();
  if (pinValue2==HIGH)
  {
    analogWrite(IN1, 0);
    analogWrite(IN2, motorSpeed);
    stateIN1 = 0;
    stateIN2 = 1;
  }
  if (pinValue2==LOW)
  {

    analogWrite(IN1, 0);
    analogWrite(IN2, 0);
    stateIN1 = 0;
    stateIN2 = 0;

  }
}
```

```
BLYNK_WRITE(V3) // Speed
{
  int Motor_Speed=param.asInt();
  Serial.println(Motor_Speed);
  motorSpeed = Motor_Speed;
  Serial.println(motorSpeed);
  if (stateIN1 == 1){
    analogWrite(IN1, motorSpeed);
  }
  if (stateIN2 == 1){
    analogWrite(IN2, motorSpeed);
  }
}
```

```
BLYNK_WRITE(V2) // STOP
{
  int pinValue3=param.asInt();
  if(pinValue3==HIGH)
  {
    analogWrite(IN1, 0);
    analogWrite(IN2, 0);
    stateIN1 = 0;
    stateIN2 = 0;
  }
}
```

CODING
ESP32

CONNECT WIRES
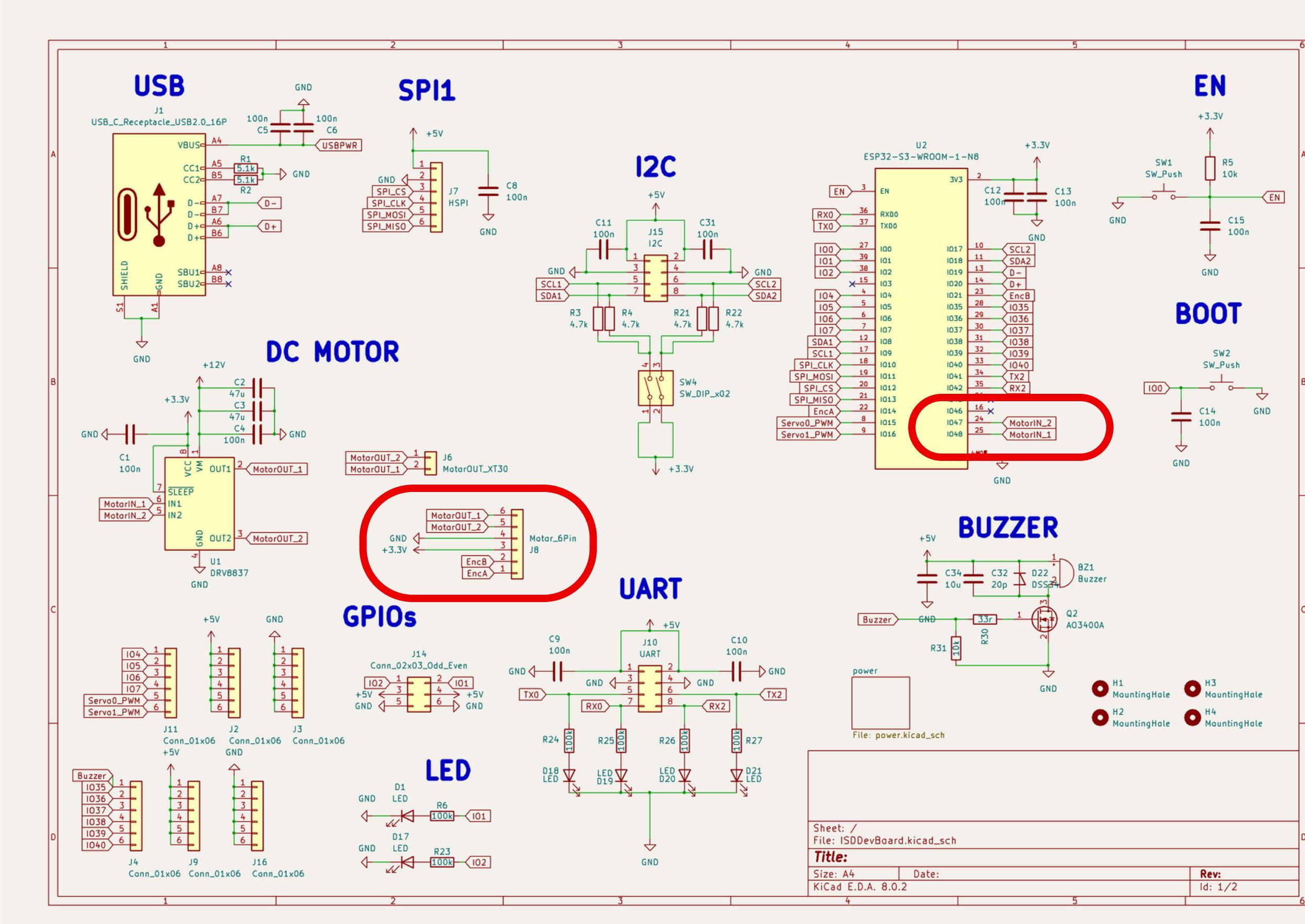
CONNECT WIRES
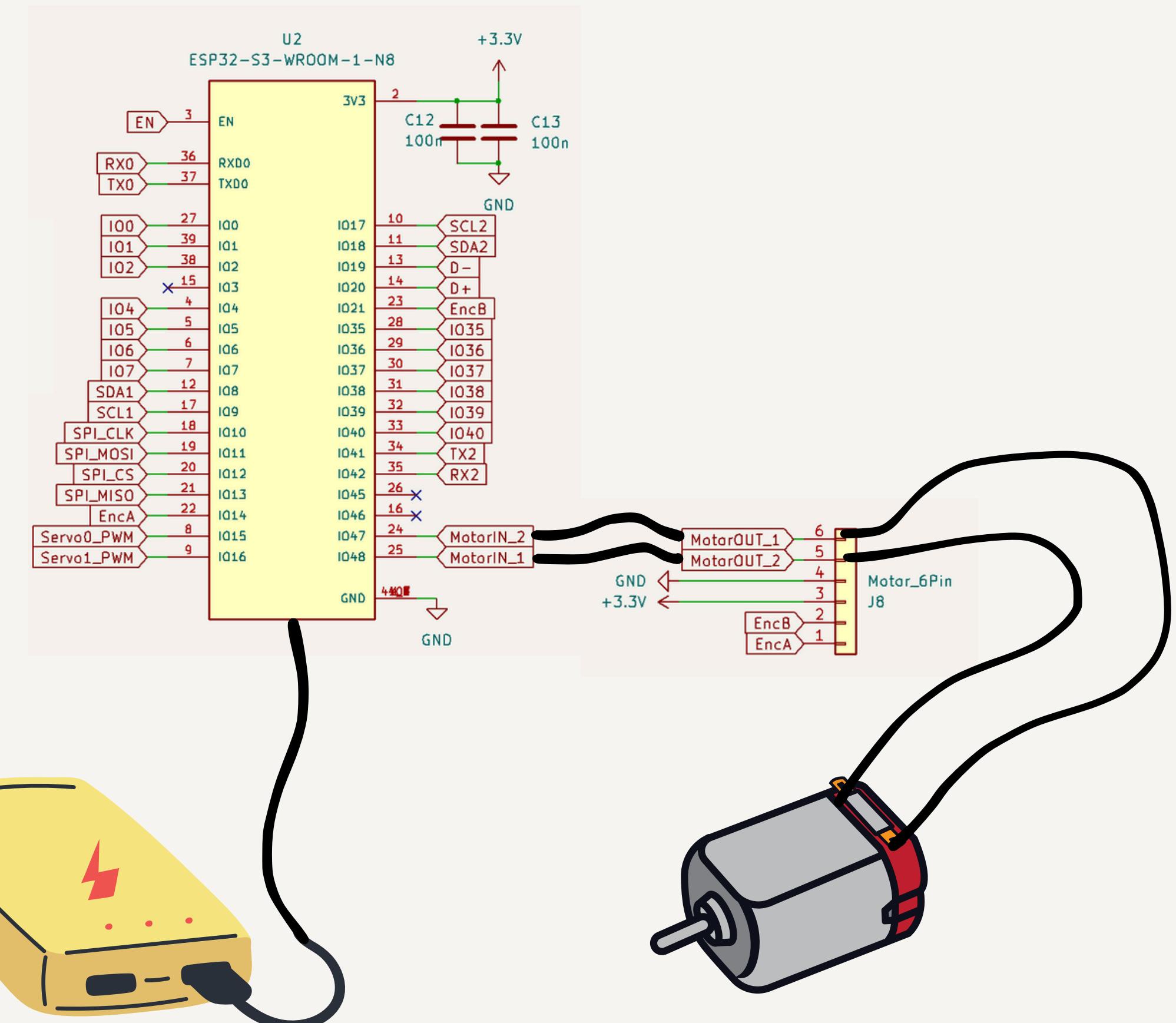
DC MOTOR CONTROL
ASSIGNMENT

DUE MAR 14TH FRI 23:59

- A video showing the system working. It should include:
  - The motor is controlled by phone app.
  - The motor can spin forward and the speed can be controlled.
  - The motor can spin backward and the speed can be controlled.

Detailed instruction are on Lab Notion Website, link can be found on Canvas

https://hkust-isdn2400.notion.site/dc-motor-lab