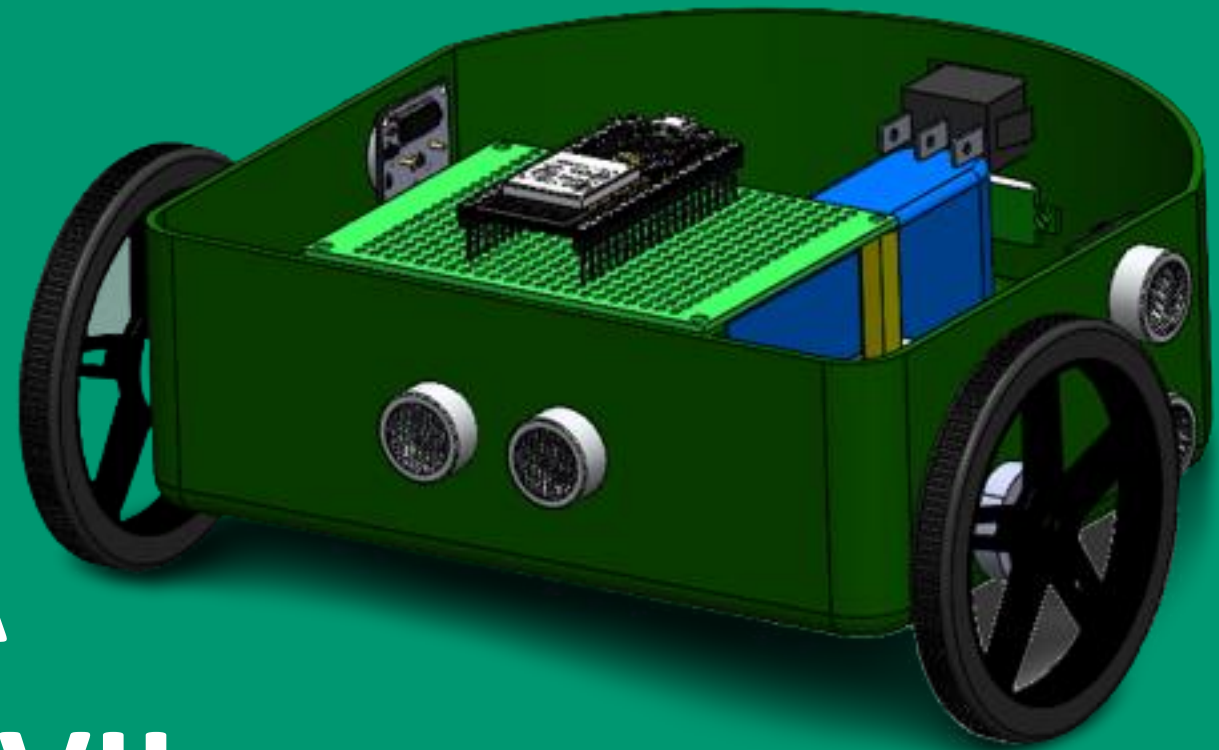




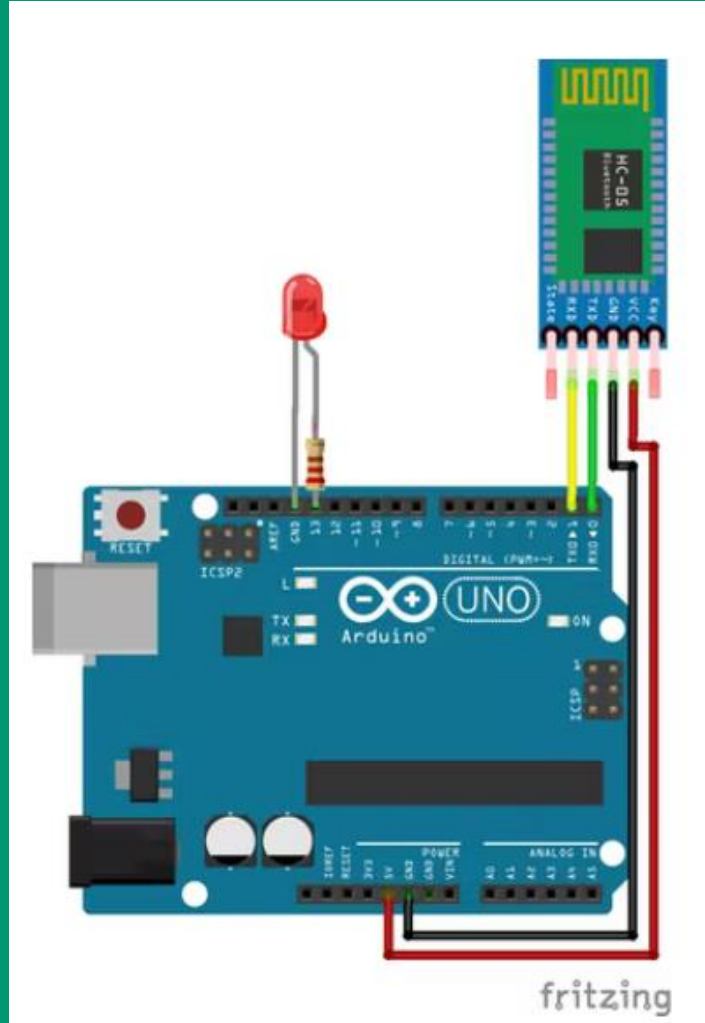
INTRODUCCIÓN A LA ROBÓTICA MÓVIL



M.I. Sergio Hernández Sánchez
sergio.hernandez.sanchez@itam.mx

Comunicación Bluetooth

Comunicación Bluetooth



Utiliza el Protocolo de Comunicación Serial

Distancia de alcance: 10 m

Comunicación Bluetooth - ejemplo

Ejemplo - _10-Bluetooth

```
#include "BluetoothSerial.h"

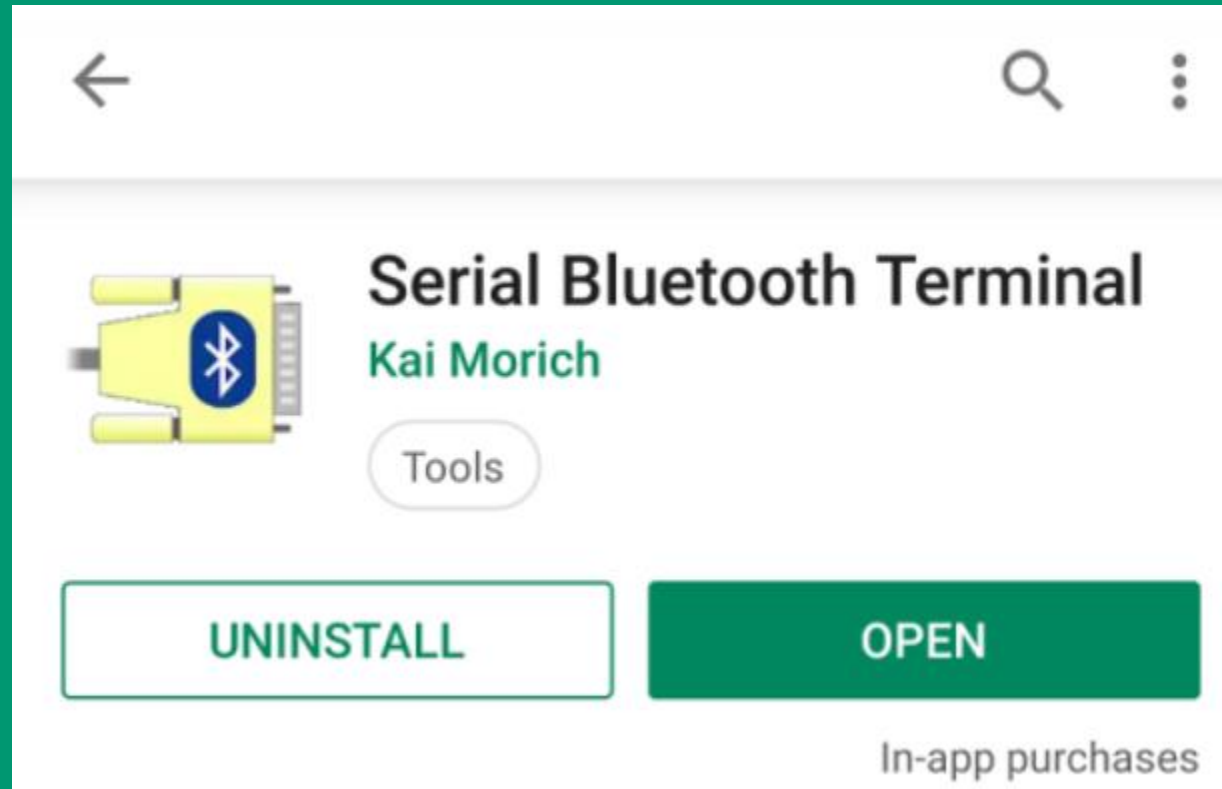
#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig` to and enable it
#endif

BluetoothSerial SerialBT;

void setup() {
  Serial.begin(115200);
  SerialBT.begin("ESP32_hersa"); //Bluetooth device name
}

void loop() {
  if (Serial.available()) {
    SerialBT.write(Serial.read());
  }
  if (SerialBT.available()) {
    Serial.write(SerialBT.read());
  }
  delay(20);
}
```

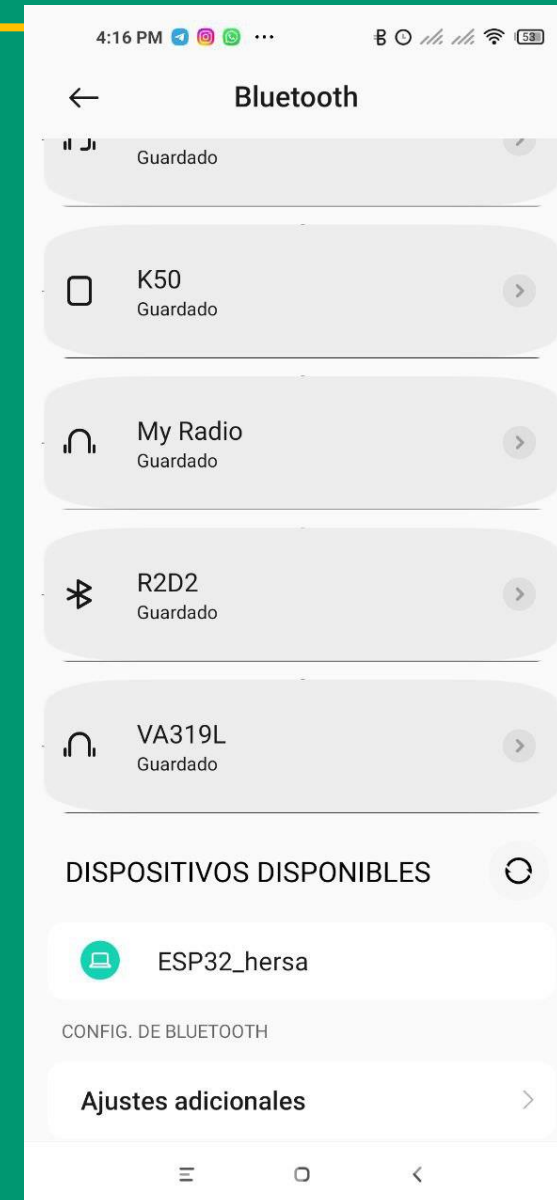
Comunicación Bluetooth - demostración



Desde su configuración vincularse al dispositivo ESP32_NombreAsignado

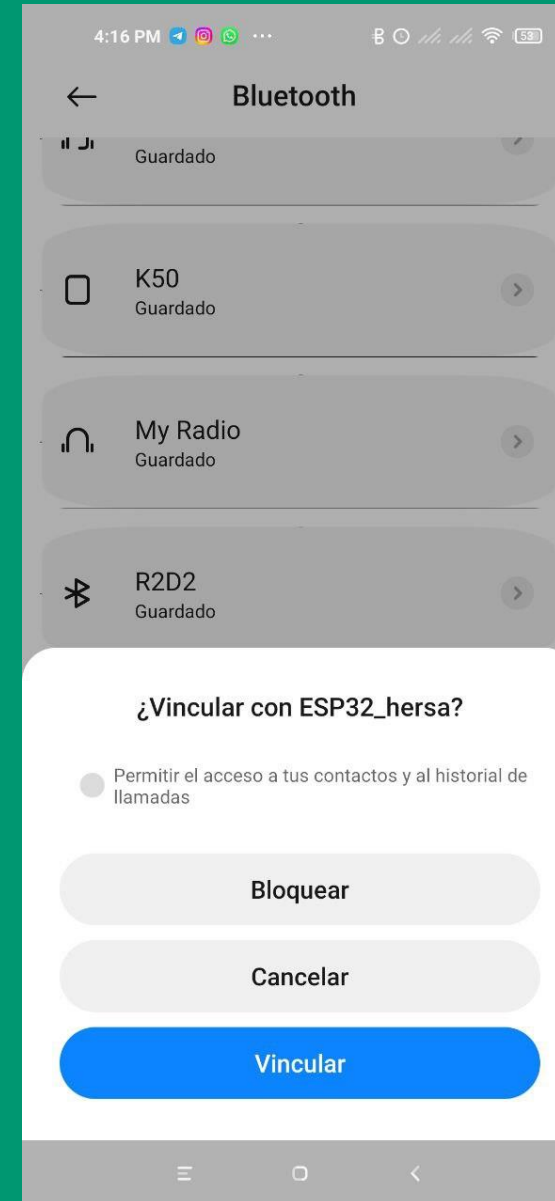
Comunicación Bluetooth – Conexión Android

Entrar a la configuración del Bluetooth desde su dispositivo Android y buscar “Dispositivos Adicionales”



Comunicación Bluetooth – Conexión Android

Vincularse con la “ESP32_nombre” y salir de configuración



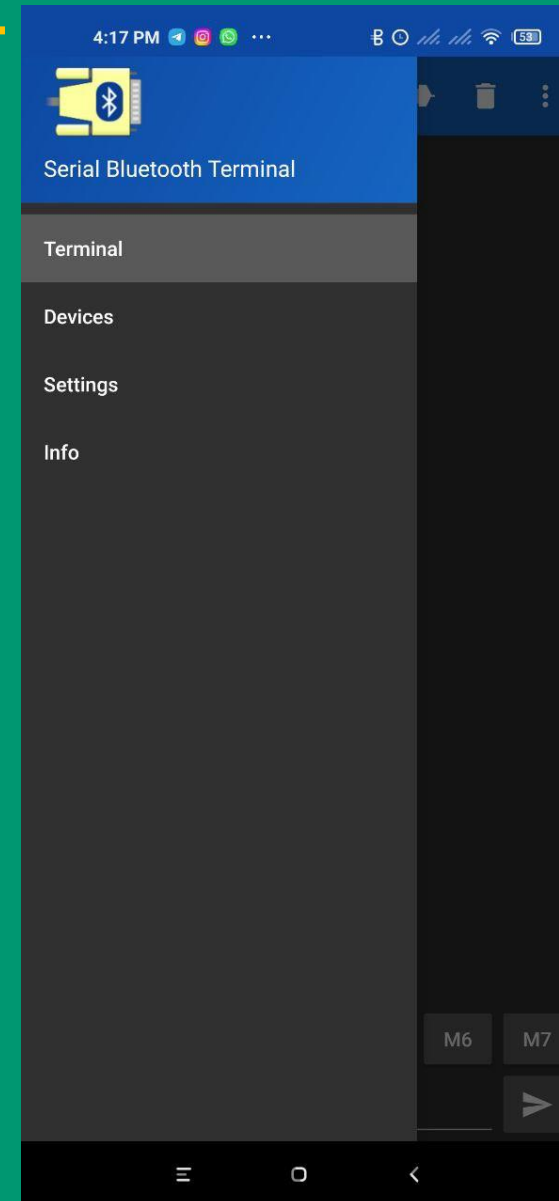
Comunicación Bluetooth – Conexión Android

Entrar a la app Serial Bluetooth
Terminal y seleccionar el botón con las
tres líneas del lado superior izquierdo



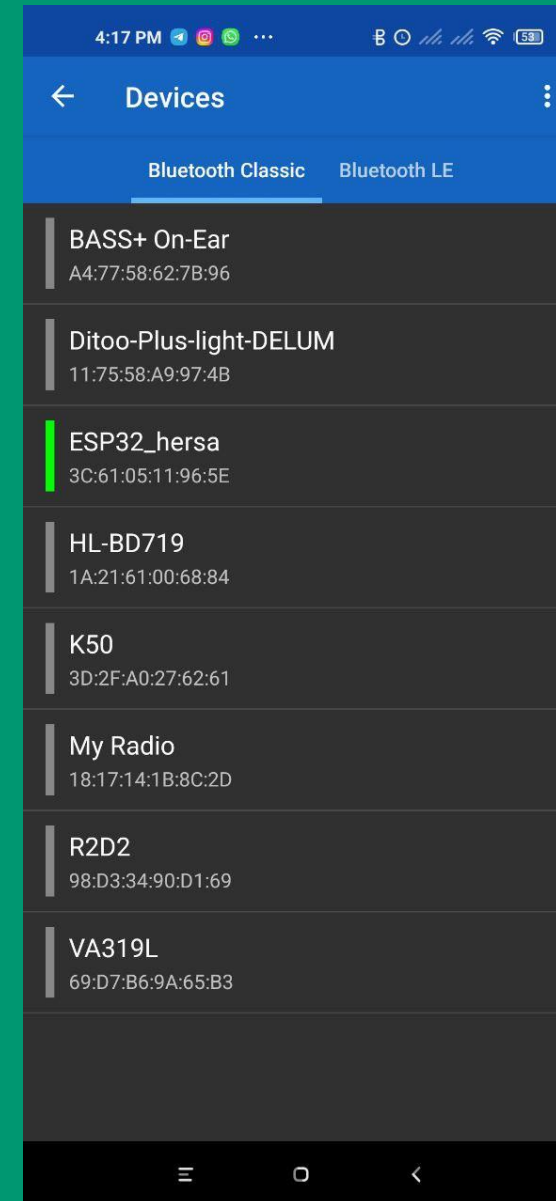
Comunicación Bluetooth – Conexión Android

Seleccionar “Devices”



Comunicación Bluetooth – Conexión Android

Seleccionar la ESP32, vinculada anteriormente



Comunicación Bluetooth – Conexión Android

Lo regresará a la pantalla de inicio, dirá conectando hasta que se logre realizar correctamente



Comunicación Bluetooth – ejemplo2

Ejemplo - _11-BT_LED

```
#include "BluetoothSerial.h"

#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig` to enable it
#endif

BluetoothSerial SerialBT;

char dato;

void setup() {
  Serial.begin(115200);
  SerialBT.begin("ESP32_hersa"); //Bluetooth device name
  pinMode(2, OUTPUT);
}
```

Comunicación Bluetooth – ejemplo2

```
void loop() {  
  if (SerialBT.available()) {  
    dato = SerialBT.read();  
  
    if (dato == '1') {  
      digitalWrite(2, HIGH);  
      Serial.println("Encendido");  
    }  
  
    if (dato == '0') {  
      digitalWrite(2, LOW);  
      Serial.println("Apagado");  
    }  
  }  
  delay(20);  
}
```

Comunicación Bluetooth – Encendido de LED's

Ejemplo - _12-BT_4LEDs

```
#include "BluetoothSerial.h"

#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig` to enable it
#endif

#define LED_BLUE 2
#define LED_GREEN 33
#define LED_RED 32
#define LED_YELLOW 18

BluetoothSerial SerialBT;

char dato;

void setup() {
  Serial.begin(115200);
  SerialBT.begin("ColmiBoT"); //Bluetooth device name
  pinMode(LED_BLUE, OUTPUT);
  pinMode(LED_GREEN, OUTPUT);
  pinMode(LED_RED, OUTPUT);
  pinMode(LED_YELLOW, OUTPUT);
}

void loop() {
  if (SerialBT.available()) {
    dato = SerialBT.read();
```

Comunicación Bluetooth – Encendido de LED's

```
if (dato == '0'){
  digitalWrite(LED_BLUE,HIGH);
  digitalWrite(LED_GREEN,LOW);
  digitalWrite(LED_RED,LOW);
  digitalWrite(LED_YELLOW,LOW);
  Serial.println("AZUL");
  SerialBT.println("AZUL");
}

if (dato == '1'){
  digitalWrite(LED_BLUE,LOW);
  digitalWrite(LED_GREEN,HIGH);
  digitalWrite(LED_RED,LOW);
  digitalWrite(LED_YELLOW,LOW);
  Serial.println("VERDE");
  SerialBT.println("VERDE");
}

if (dato == '2'){
  digitalWrite(LED_BLUE,LOW);
  digitalWrite(LED_GREEN,LOW);
  digitalWrite(LED_RED,HIGH);
  digitalWrite(LED_YELLOW,LOW);
  Serial.println("ROJO");
  SerialBT.println("ROJO");
}
```

```
if (dato == '3'){
  digitalWrite(LED_BLUE,LOW);
  digitalWrite(LED_GREEN,LOW);
  digitalWrite(LED_RED,LOW);
  digitalWrite(LED_YELLOW,HIGH);
  Serial.println("AMARILLO");
  SerialBT.println("AMARILLO");
}

if (dato == '4'){
  digitalWrite(LED_BLUE,LOW);
  digitalWrite(LED_GREEN,LOW);
  digitalWrite(LED_RED,LOW);
  digitalWrite(LED_YELLOW,LOW);
  Serial.println("APAGADOS");
  SerialBT.println("APAGADOS");
}

}
delay(100);
}
```

Motores CD



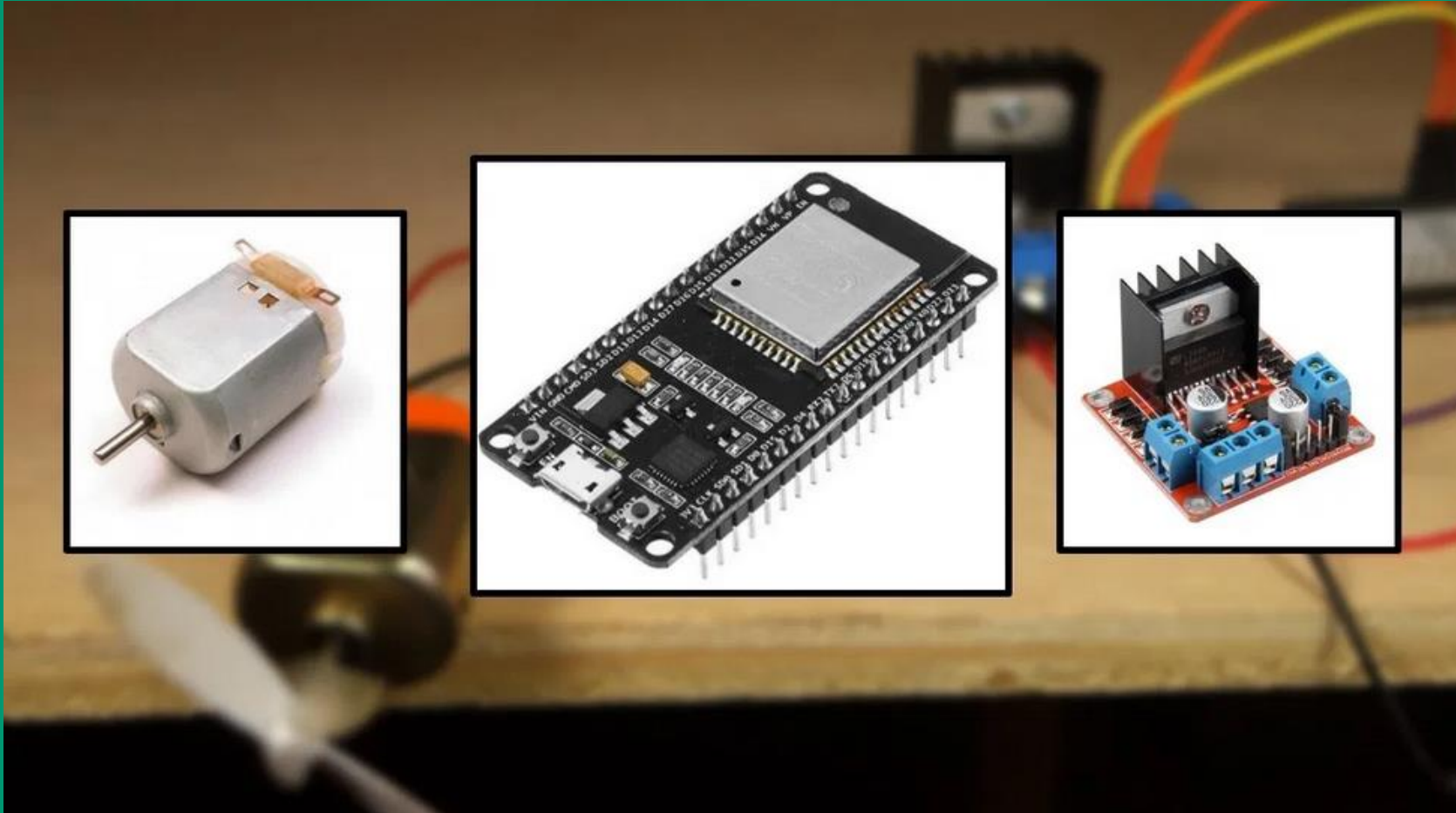
Servomotores



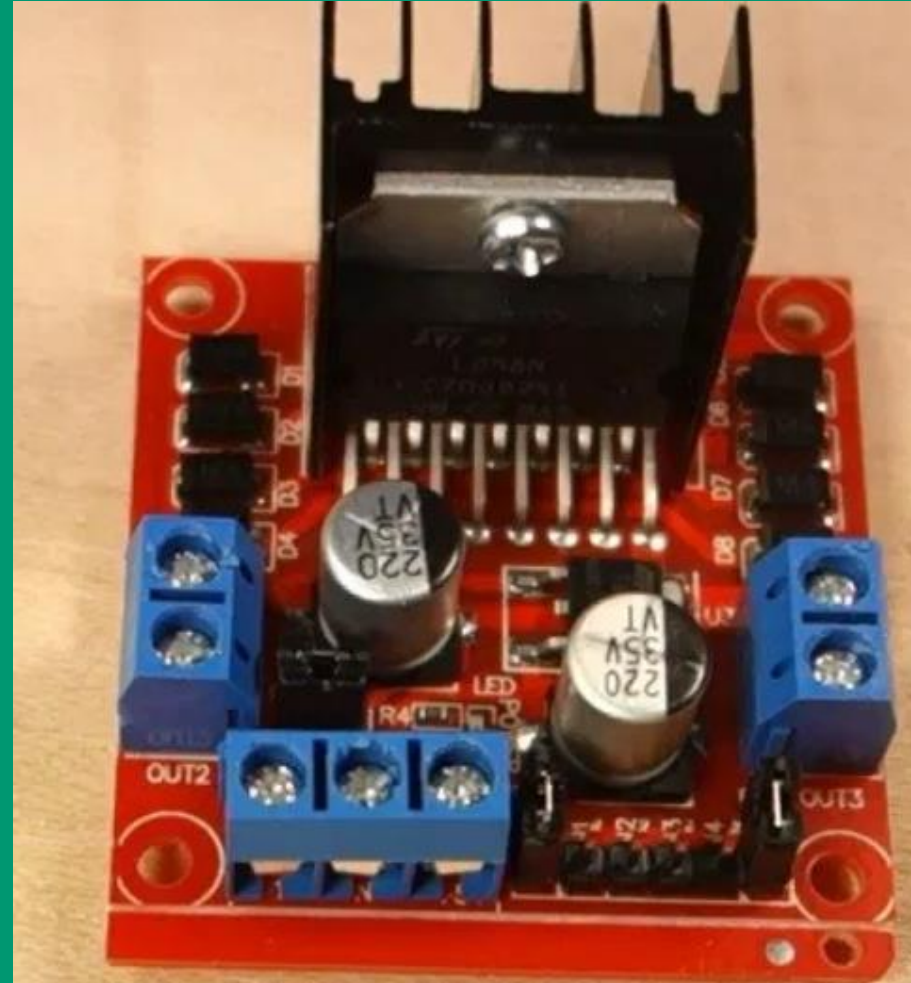
Motores paso a paso



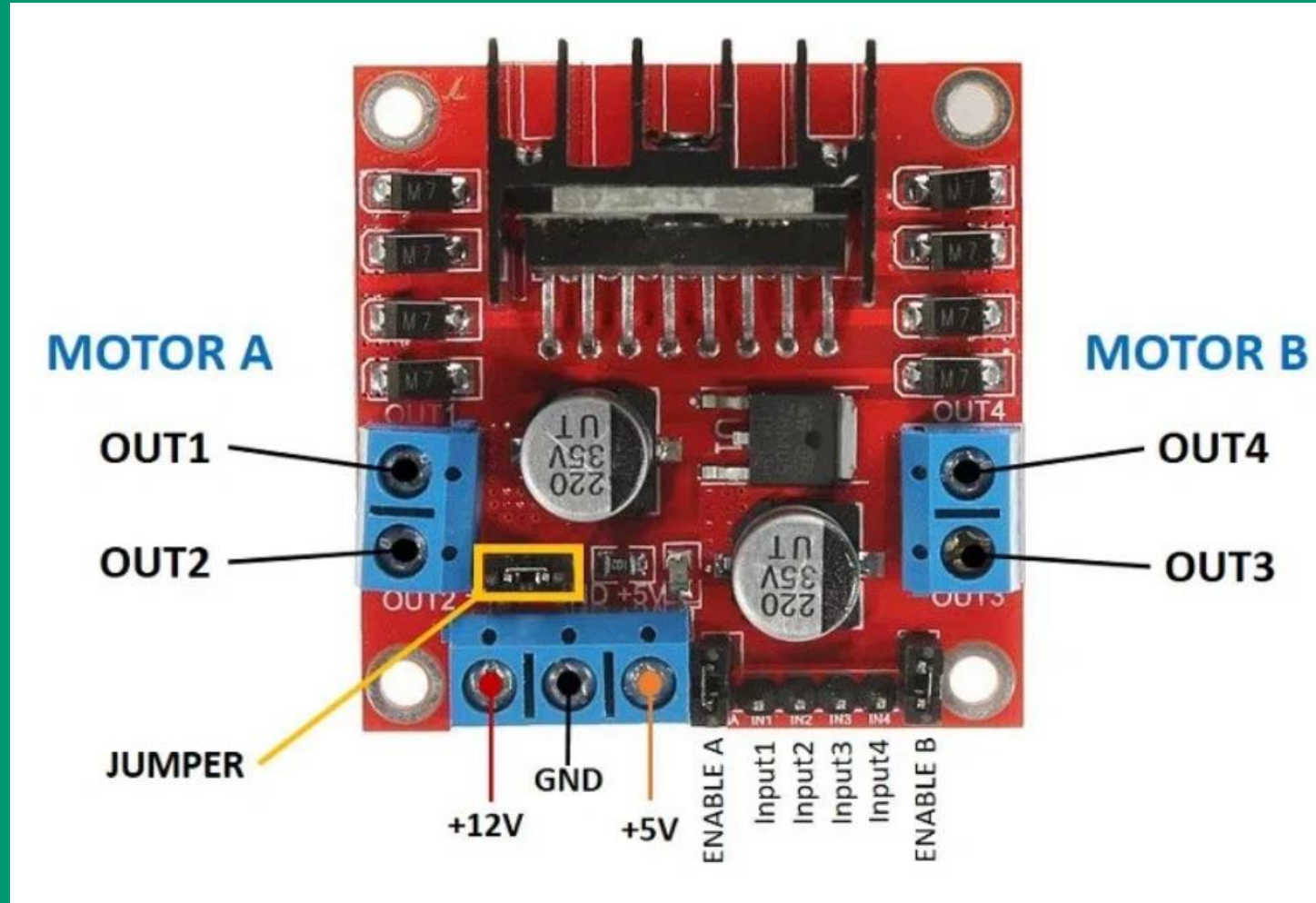
Control de un motor de CD



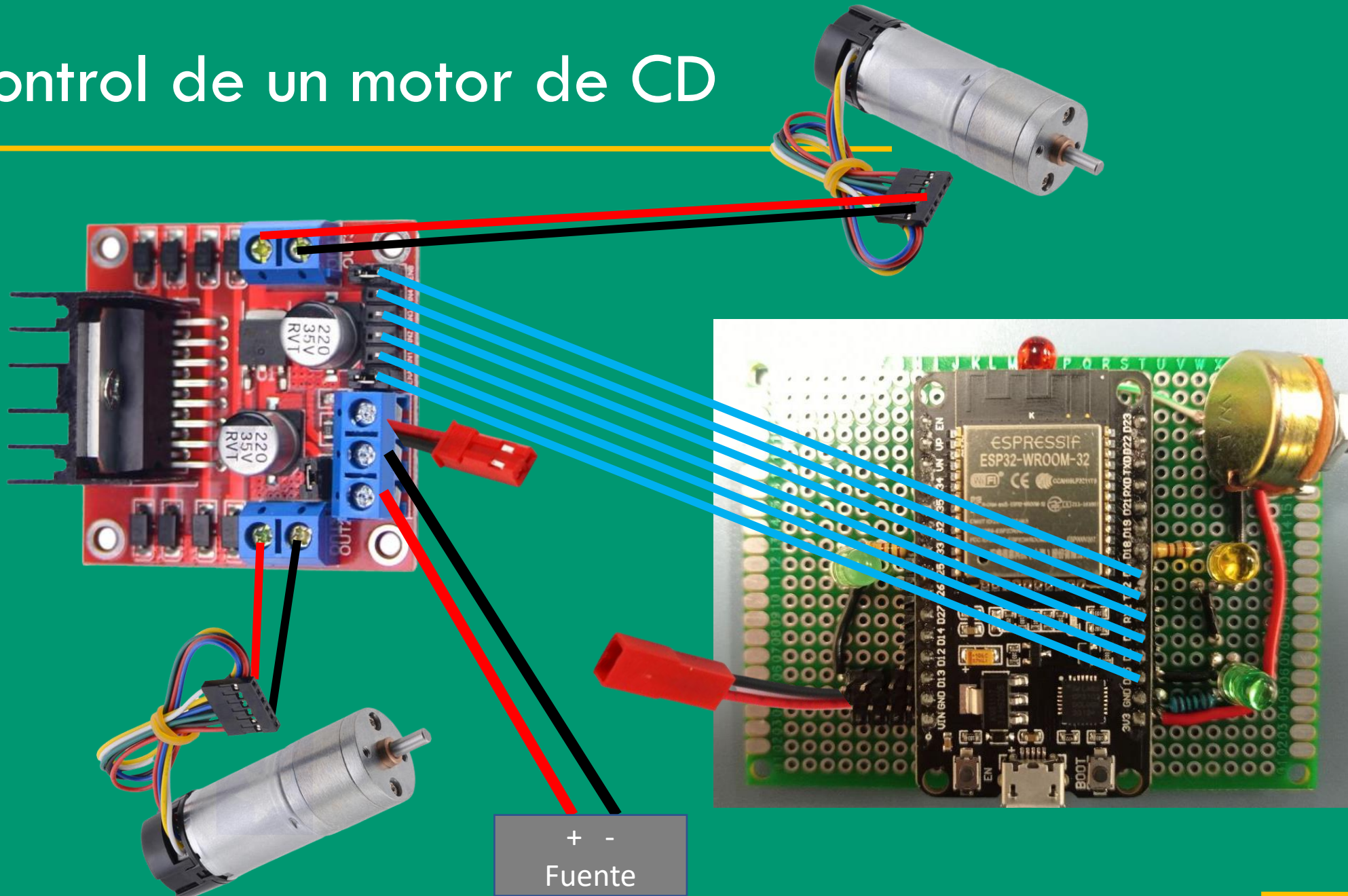
Control de un motor de CD - L298N



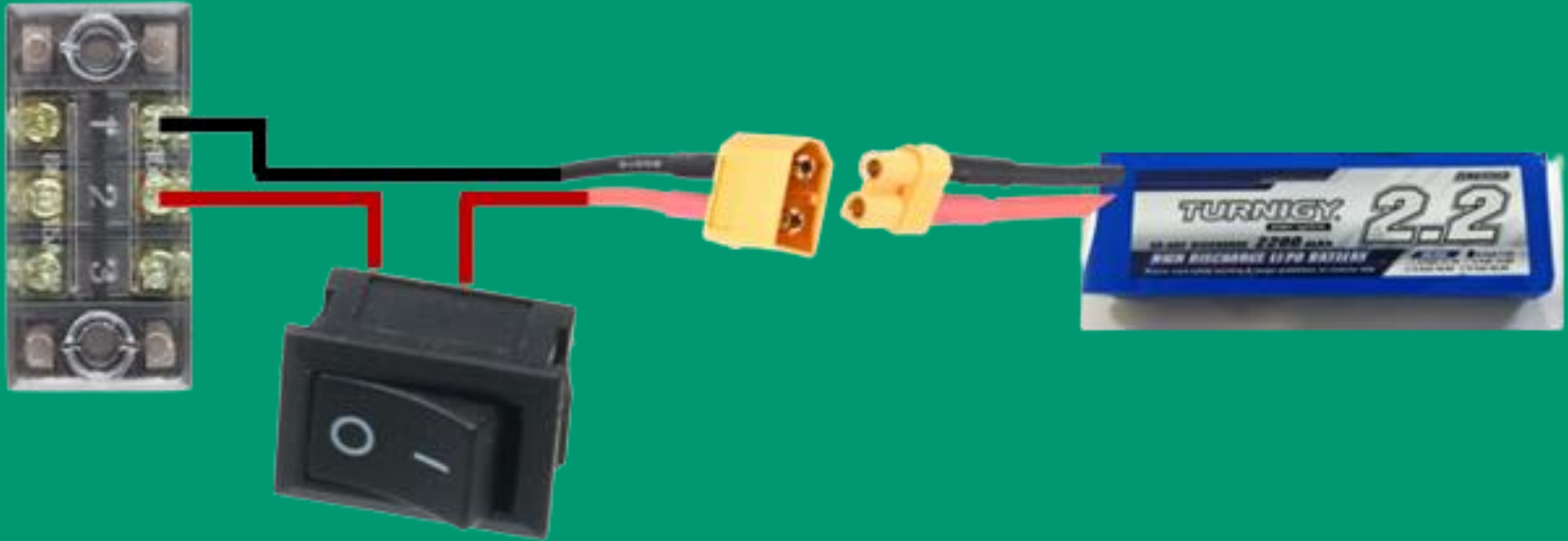
Control de un motor de CD - L298N



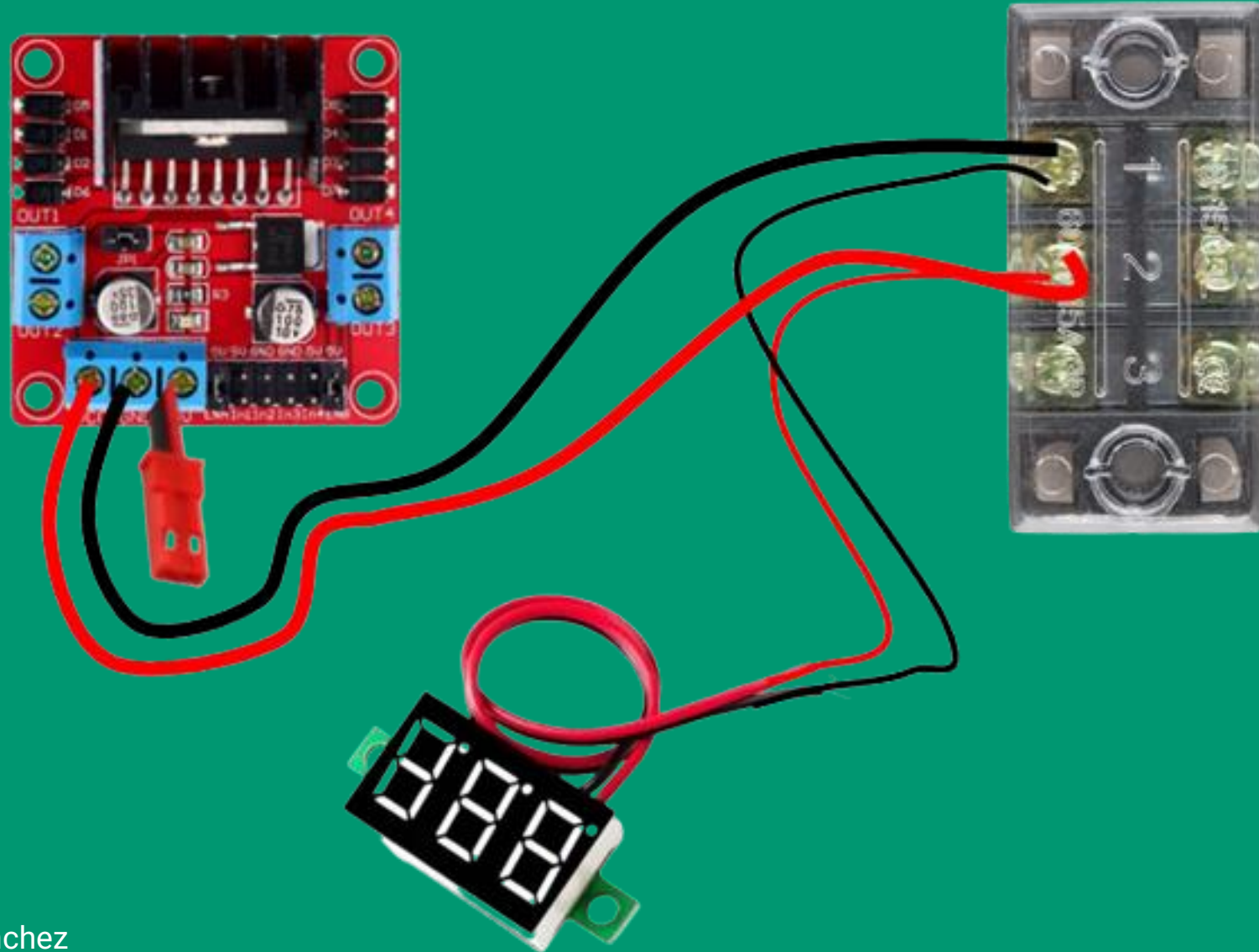
Control de un motor de CD



Conexión batería - switch - terminal



Conexión terminal - voltmetro - L298N



Control de un motor de CD

Ejemplo - _13-Control_Motor

```
#define EnML 13
#define ML_A 12
#define ML_B 14

// Configurando los parámetros de PWM
const int frecuencia = 8000;
const int canal = 0;
const int resolucion = 12;
int vel = 4095;

void setup() {
  pinMode(EnML, OUTPUT);
  pinMode(ML_A, OUTPUT);
  pinMode(ML_B, OUTPUT);

  ledcSetup(canal, frecuencia, resolucion);

  ledcAttachPin(EnML, canal);

  ledcWrite(canal, vel);

  Serial.begin(115200);
}
```

```
void loop() {
  Forward();
  delay(2000);

  Stop();
  delay(1000);

  Back();
  delay(2000);

  Stop();
  delay(1000);
}
```

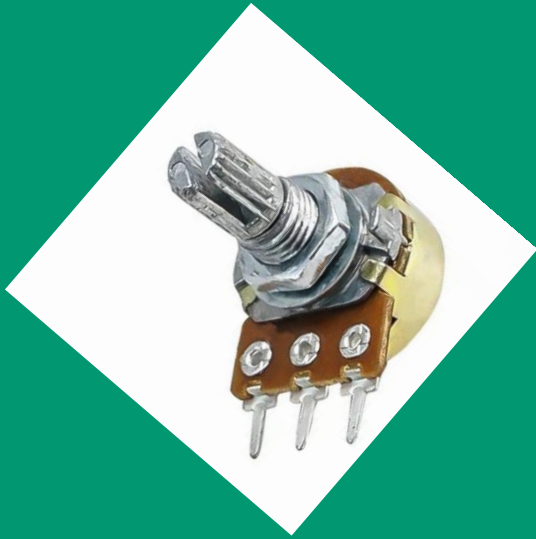
```
void Forward() {
  digitalWrite(ML_A, LOW);
  digitalWrite(ML_B, HIGH);
  Serial.println("Forward");
}

void Back() {
  digitalWrite(ML_A, HIGH);
  digitalWrite(ML_B, LOW);
  Serial.println("Back");
}

void Stop() {
  digitalWrite(ML_A, LOW);
  digitalWrite(ML_B, LOW);
  Serial.println("Stopped");
}
```


Control de un motor de CD

Ejemplo - _14-Control_Motor_Vel



0%  100%

Voltaje



ADC



PWM



0%  100%

Velocidad