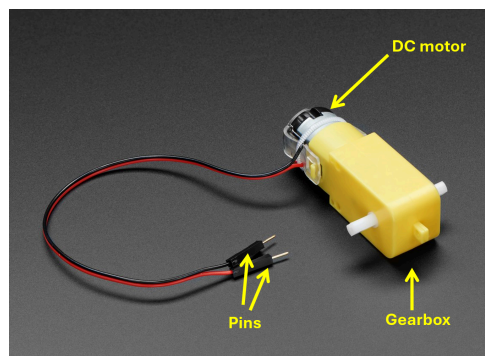# Motors

This workshop aims to familiarise yourself with using the L298N module to independently control up to 2 motors. These will be a key component of your robot in the competition and allow for a lot of versatility and control.

Materials
- 1x Arduino Uno + Cable
- 2x DC Motors
- 1x L298N Motor Driver
- Jumper Wires
- Usb trigger board and power bank
- L298N H-bridge motor driver
- HM10 Bluetooth module

DC motor

Every DC motor is connected to a gearbox. The DC motors have a voltage rating of 3V-6V. This means a voltage of 3-6V is required to be connected across the terminals to power the motor.

These motors can be powered directly from the Arduino. Although this will work the Arduino has a limited amount of current it can provide. To fix this issue it is possible to use an external power source (power bank) and control it with the arduino. We will start by powering it with the Arduino. Connect the red cable to 5V and black cable to GND. This will run the motor in the clockwise direction. Connecting in the opposite direction will cause the motor to spin counterclockwise.

By connecting the power supply directly to the motor, we can make the motor spin at a constant speed. To do more interesting things like controlling the speed and direction of the spinning motor, we need to use a L298N module.

Understanding the L298N module
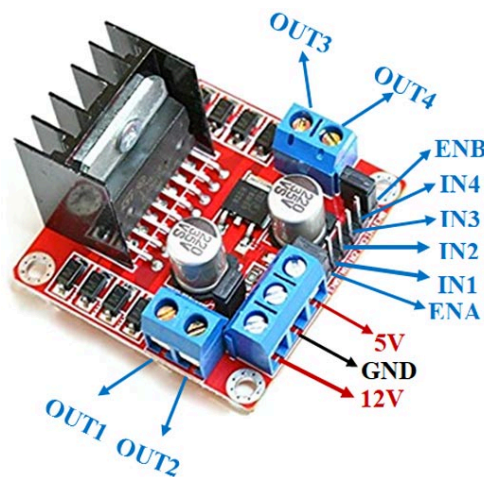
This image shows the pinout of the L298N. A description of each Pin can be found below.

ENA	Enables PWM signal for Motor 1 to control speed	Connect to PWM pin
IN1 & IN2	Control the direction of spin for Motor 1	Connect to Digital pin
OUT1 & OUT 2	Output Pins for Motor 1	Connect to Motor 1
ENB	Enables PWM signal for Motor 2 to control speed	Connect to PWM pin
IN3 & IN4	Control the direction of spin for Motor 2	Connect to Digital pin
OUT3 & OUT4	Output Pins for Motor 2	Connect to Motor 2
5V	Output voltage	To power other

		devices. Eg Arduino
GND	Ground	DC Power ground
12V	Input voltage to drive motors	Batteries/DC power positive

To summarise, when using the L298N module to control a motor:
1. Set the speed by sending a PWM signal between 0-255 to EN
2. Set the direction of the motor using the IN pins. One should be set high and the other low
3. Make sure to check that the two pins above ground and 12V are connected by a jumper (we are not connecting anything to it so need to make sure that we ground the power)

Controlling the motor using Arduino Uno

Open up the Arduino program on your computer and plug in your Arduino. Make sure that *Tools* -> *Board Type* is set to 'Arduino Uno' and *Tools* -> *Port* lists the Arduino that you just attached. Use the following code to test your motor.

```cpp
C/C++
// Define the motor control pins for Motor A
#define ENA  // Enable pin for motor A (PWM)
#define IN1  // Input 1 for motor A
#define IN2  // Input 2 for motor A

// Define the motor control pins for Motor B
#define ENB  // Enable pin for motor B (PWM)
#define IN3  // Input 1 for motor B
#define IN4  // Input 2 for motor B

void setup() {
  // Set all the motor control pins to output for Motor A
  // Write code here :

  // Set all the motor control pins to output for Motor B
  // Write code here :

  // Initial motor states (Stationary)
  // Write code here :
}

void loop() {
  // Test Motor A: Rotate forward at full speed
```

```
delay(2000);  // Run for 2 seconds
// Test Motor A: Stop

delay(1000);  // Stop for 1 second
// Test Motor A: Rotate backward at half speed

delay(2000);  // Run for 2 seconds
// Test Motor A: Stop

delay(1000);
// Test Motor B: Rotate forward at full speed

delay(2000);
// Test Motor B: Stop

delay(1000);

// Test Motor B: Rotate backward at half speed

delay(2000);
// Test Motor B: Stop

delay(1000);
}
```

ENA_PWM goes from LOW (0) to 255 (HIGH). The codes control the motor to spin in one direction for 0.5 seconds, and turn it off for 0.5 seconds. The table below shows the settings to control the motor in different ways.

EN	IN1	IN2	Function
H	H	H	Fast motor stop
H	H	L	Turn one way
H	L	H	Turn the other way
H	L	L	Fast motor stop
L	X	X	Fast motor stop

(H = High, L = Low, X = High or Low)

Activity:

Have a go at building this system for yourself. Use the kits provided and experiment with making a wheel system that goes

forward, backwards and rotates. Think about how you can rotate the robot.

You have to short the pin above the voltage source.

Remember to utilise the demonstrators as well as internet sources if you get stuck.

Next Power 2 motors and Arduino using battery, completely disconnect from the laptop after uploading the code (useful for the competition)

Control motor based on Ultrasonic sensor

Obstacle avoidance relies on reading the environment through sensors and stopping before the robot makes contact with obstacles. Alter your code such that the motor is running so long as the ultrasonic sensor does not detect an obstacle close by.