



# **SNS COLLEGE OF TECHNOLOGY**

**Coimbatore-35**

**An Autonomous Institution**

Accredited by NBA – AICTE and Accredited by NAAC – UGC  
with 'A++' Grade

Approved by AICTE, New Delhi & Affiliated to Anna  
University, Chennai

## **DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING**

### **19ECT213-IoT SYSTEM ARCHITECTURE**

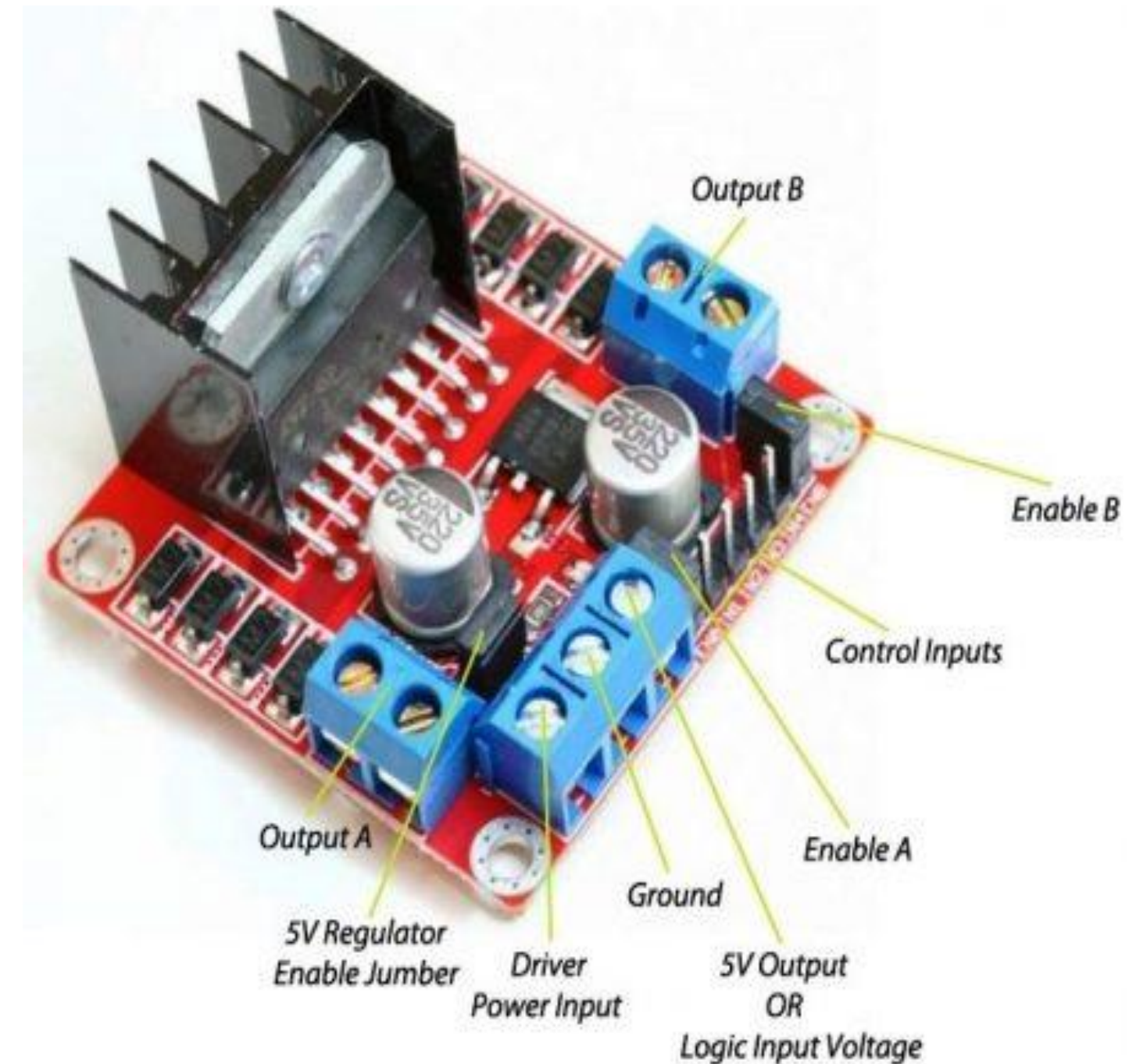
II YEAR/ IV SEMESTER

**UNIT 3 – MICROCONTROLLER AND INTERFACING TECHNIQUES FOR IoT DEVICES**



## Interface L298N DC Motor Driver Module with Arduino

- **Arduino** board or a **microcontroller** IO pins don't have enough current/voltage driving capability to drive a motor
- For driving the motor in both directions (clockwise and anti-clockwise) an H-Bridge is used
- L298N is an integrated monolithic circuit with dual H-Bridge.
- It can be used to rotate the motor in both directions and to control the speed of the motor using PWM technique.

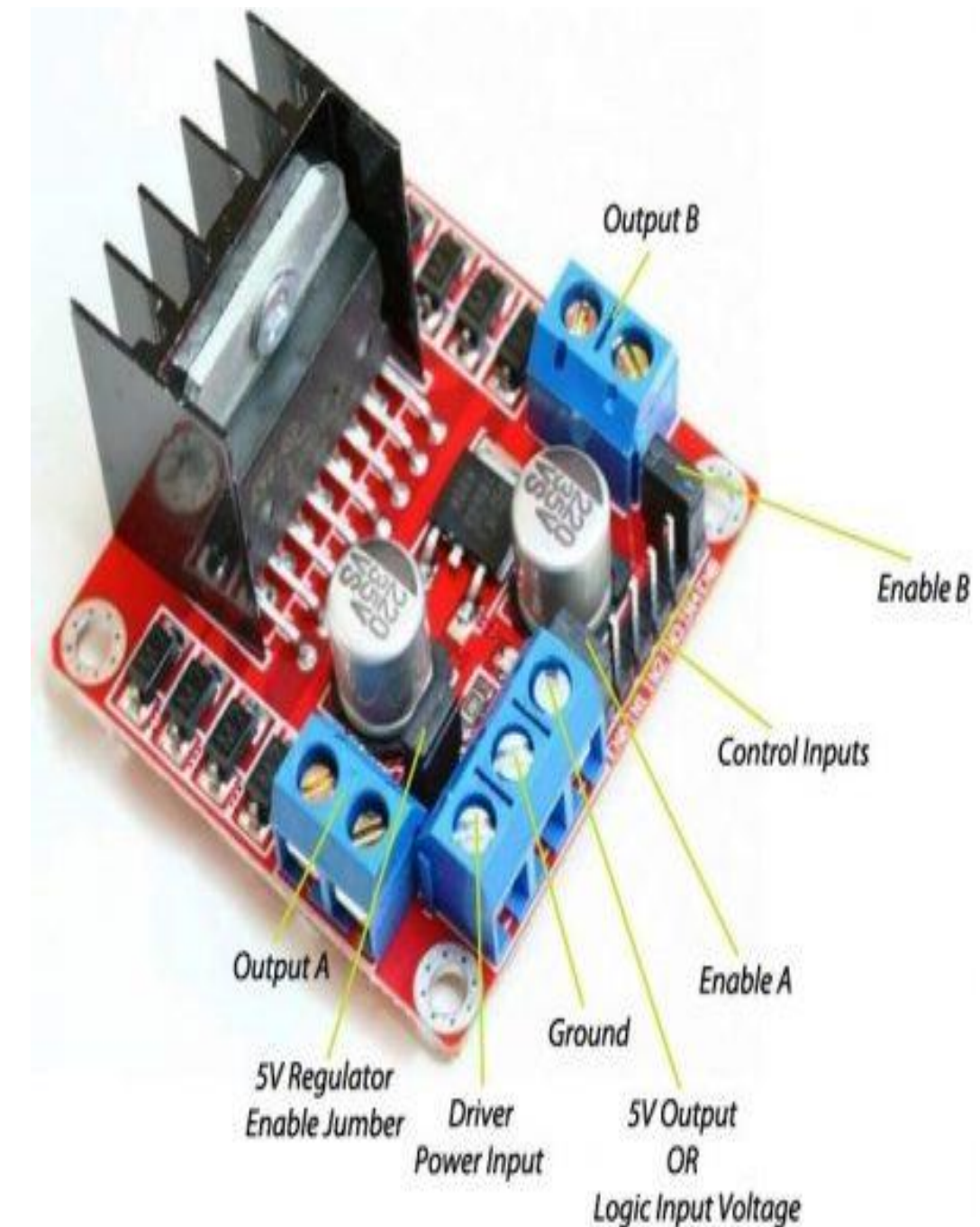




# Interface L298N DC Motor Driver Module with Arduino

## Specifications

- Output A, Output B – To connect two motors.
- Driver Power Input – Board can accept 5V to 35V which will act as the power supply to motor and internal 5V voltage regulator (if it is enabled using jumper).
- GND – Common Ground
- 5V Output / Logic Input Voltage – This pin will give 5V output if the on-board regulator is enabled using the jumper, other we should give the logic supply (5V in the case of Arduino Uno) to this pin.
- IN1, IN2, IN3, IN4 – H-Bridge control inputs which can be used to control direction of motors.
- Enable A and Enable B pins are used for enabling each bridge or for controlling the speed of the motors using PWM.





# Interface L298N DC Motor Driver Module with Arduino

## Controlling a DC Motor

In order to have a complete control over DC motor, we have to control its speed and rotation direction. This can be achieved by combining these two techniques.

- PWM – For controlling speed
- H-Bridge – For controlling rotation direction

### PWM – For controlling speed

The speed of a DC motor can be controlled by varying its input voltage. A common technique for doing this is to use PWM (Pulse Width Modulation)

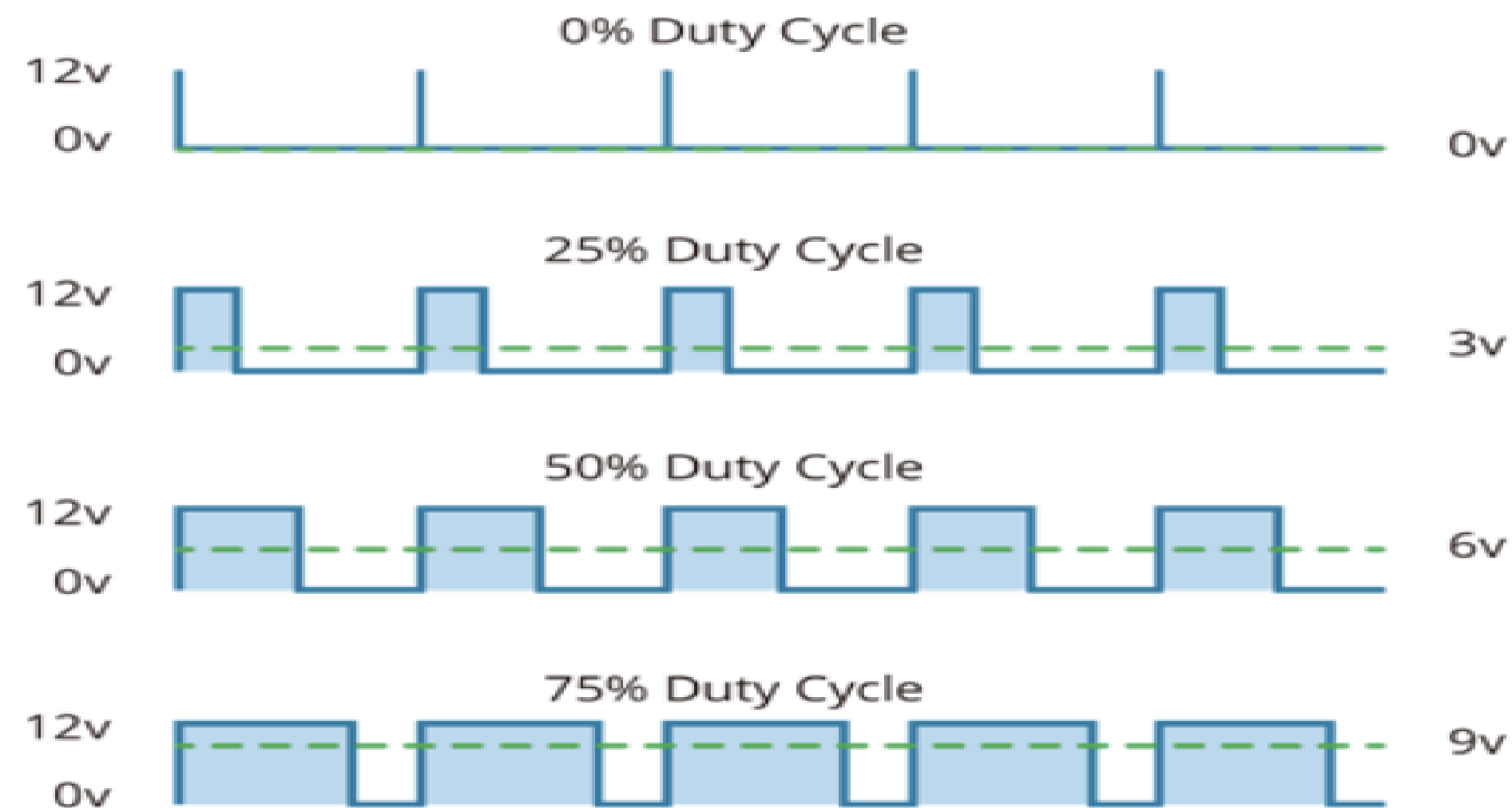
PWM is a technique where average value of the input voltage is adjusted by sending a series of ON-OFF pulses.



The average voltage is proportional to the width of the pulses known as **Duty Cycle**.

The higher the duty cycle, the greater the average voltage being applied to the dc motor(Hig Speed) and the lower the duty cycle, the less the average voltage being applied to the dc motor(Low Speed).

Below image illustrates PWM technique with various duty cycles and average voltages.

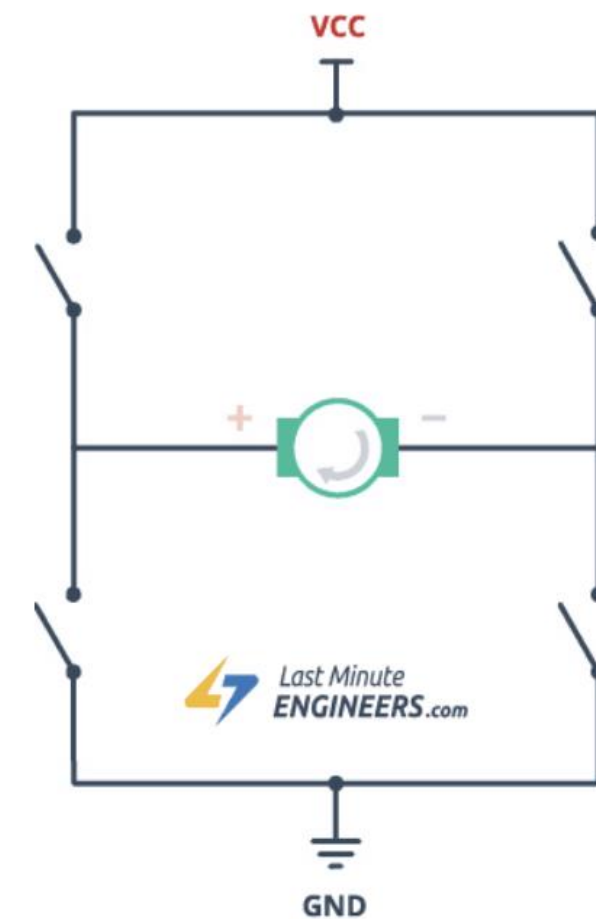




## H-Bridge – For controlling rotation direction

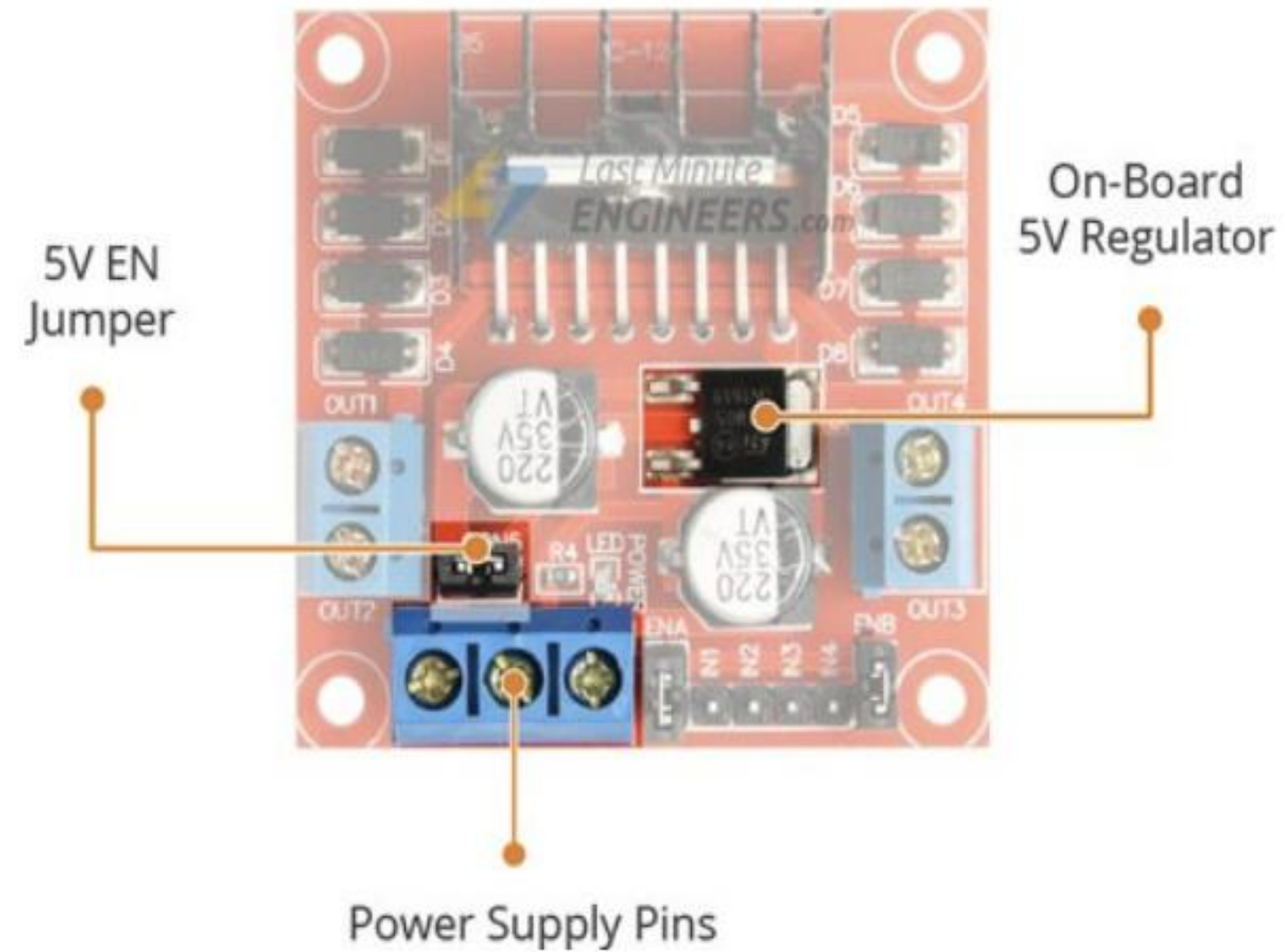
The DC motor's spinning direction can be controlled by changing polarity of its input voltage. A common technique for doing this is to use an H-Bridge.

An H-Bridge circuit contains four switches with the motor at the center forming an H-like arrangement.





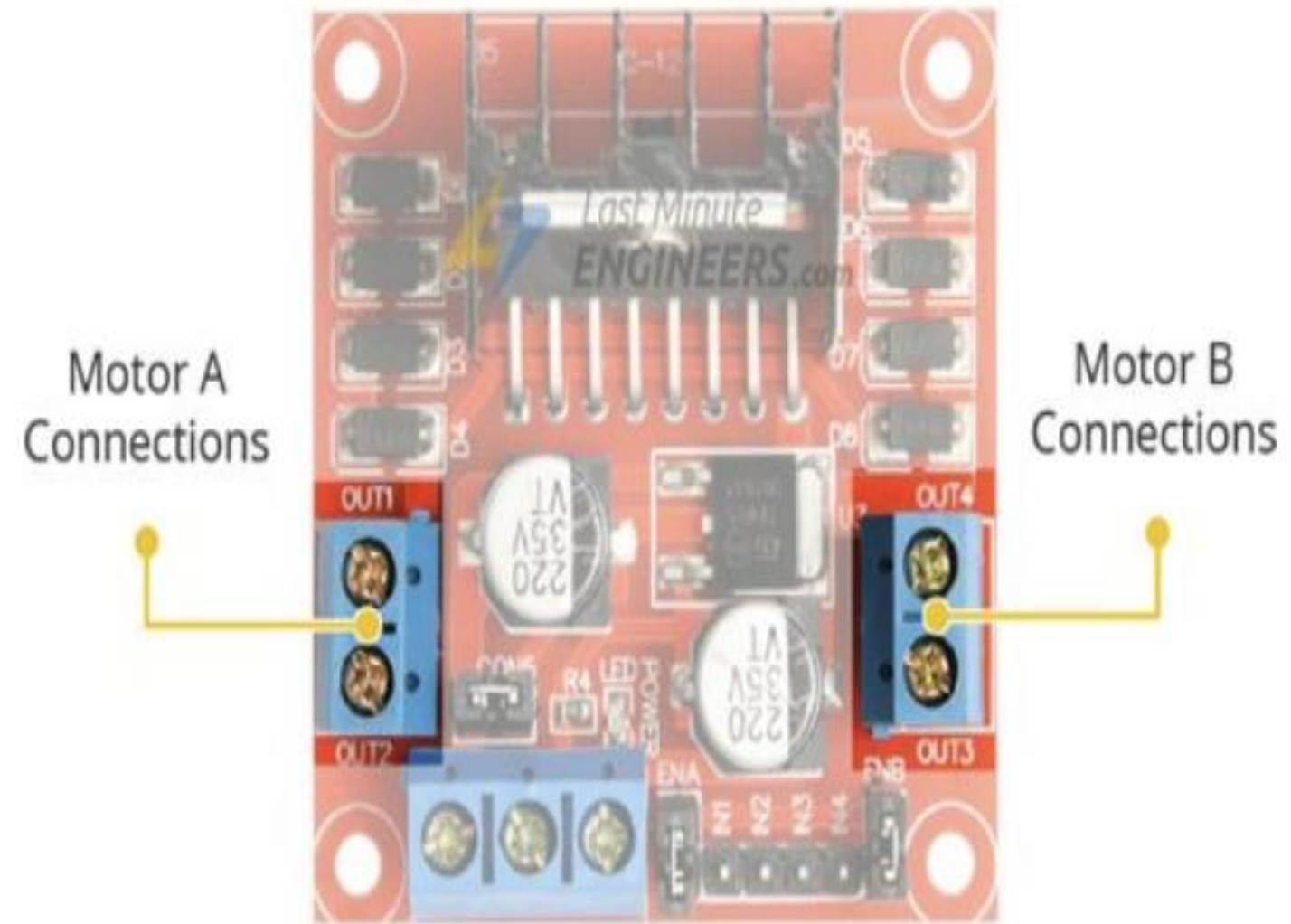
# Power Supply



The L298N motor driver module is powered through 3-pin 3.5mm-pitch screw terminals. It consists of pins for motor power supply( $V_s$ ), ground and 5V logic power supply( $V_{ss}$ ).



# Output Pins

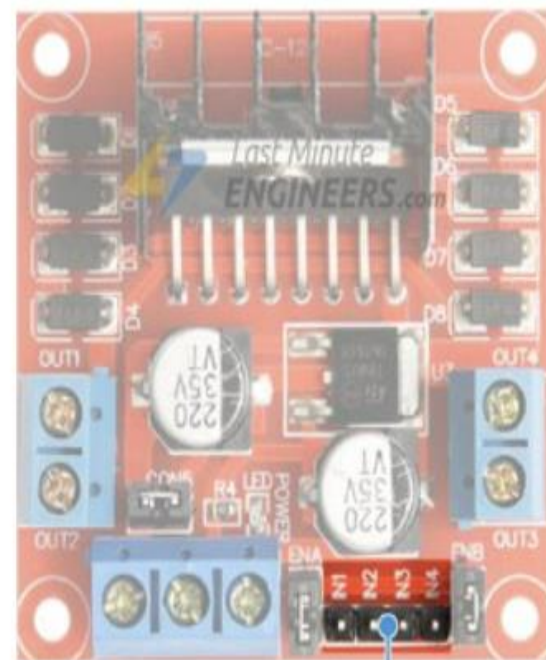




## Control Pins

For each of the L298N's channels, there are two types of control pins which allow us to control speed and spinning direction of the DC motors at the same time viz. Direction control pins & Speed control pins.

### Direction Control Pins



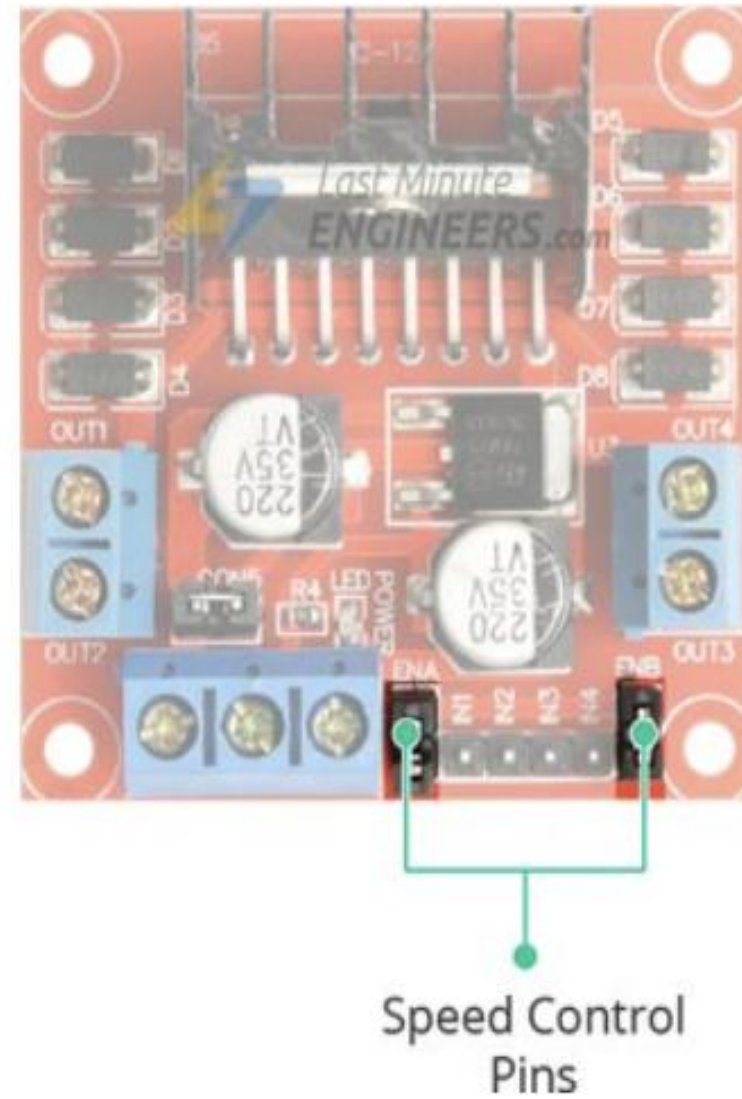
Direction Control Pins

The spinning direction of a motor can be controlled by applying either a logic HIGH(5 Volts) or logic LOW(Ground) to these inputs. The below chart illustrates how this is done.

Input1	Input2	Spinning Direction
Low(0)	Low(0)	Motor OFF
High(1)	Low(0)	Forward
Low(0)	High(1)	Backward
High(1)	High(1)	Motor OFF



## Speed Control Pins

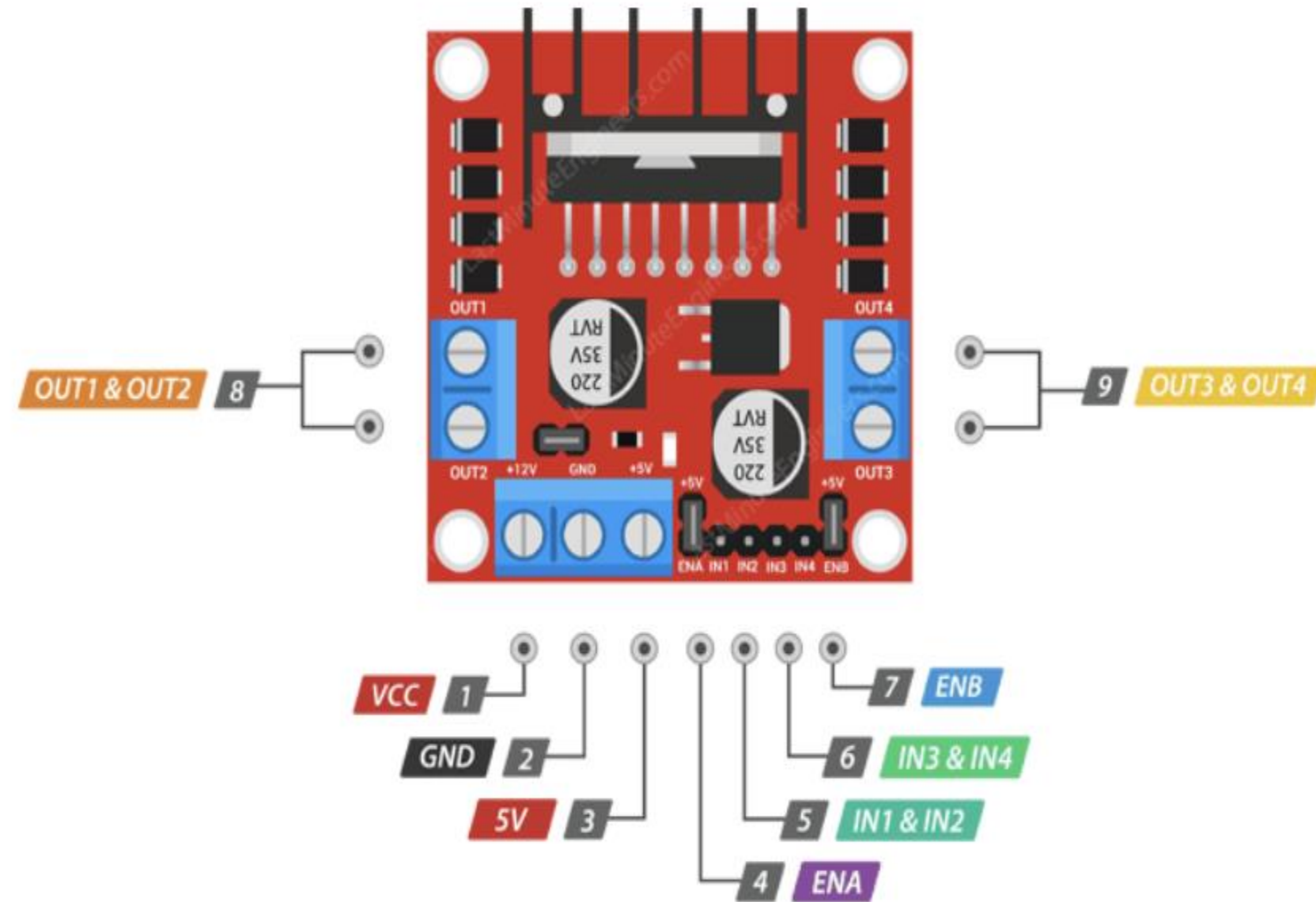


The speed control pins viz. **ENA** and **ENB** are used to turn the motors ON, OFF and control its speed.



# L298N Motor Driver Module Pinout

Before diving into hookup and example code, let's first take a look at its Pinout.





**VCC** pin supplies power for the motor. It can be anywhere between 5 to 35V. Remember, if the 5V-EN jumper is in place, you need to supply 2 extra volts than motor's actual voltage requirement, in order to get maximum speed out of your motor.

**GND** is a common ground pin.

**5V** pin supplies power for the switching logic circuitry inside L298N IC. If the 5V-EN jumper is in place, this pin acts as an output and can be used to power up your Arduino. If the 5V-EN jumper is removed, you need to connect it to the 5V pin on Arduino.

**ENA** pins are used to control speed of Motor A. Pulling this pin HIGH(Keeping the jumper in place) will make the Motor A spin, pulling it LOW will make the motor stop. Removing the jumper and connecting this pin to PWM input will let us control the speed of Motor A.



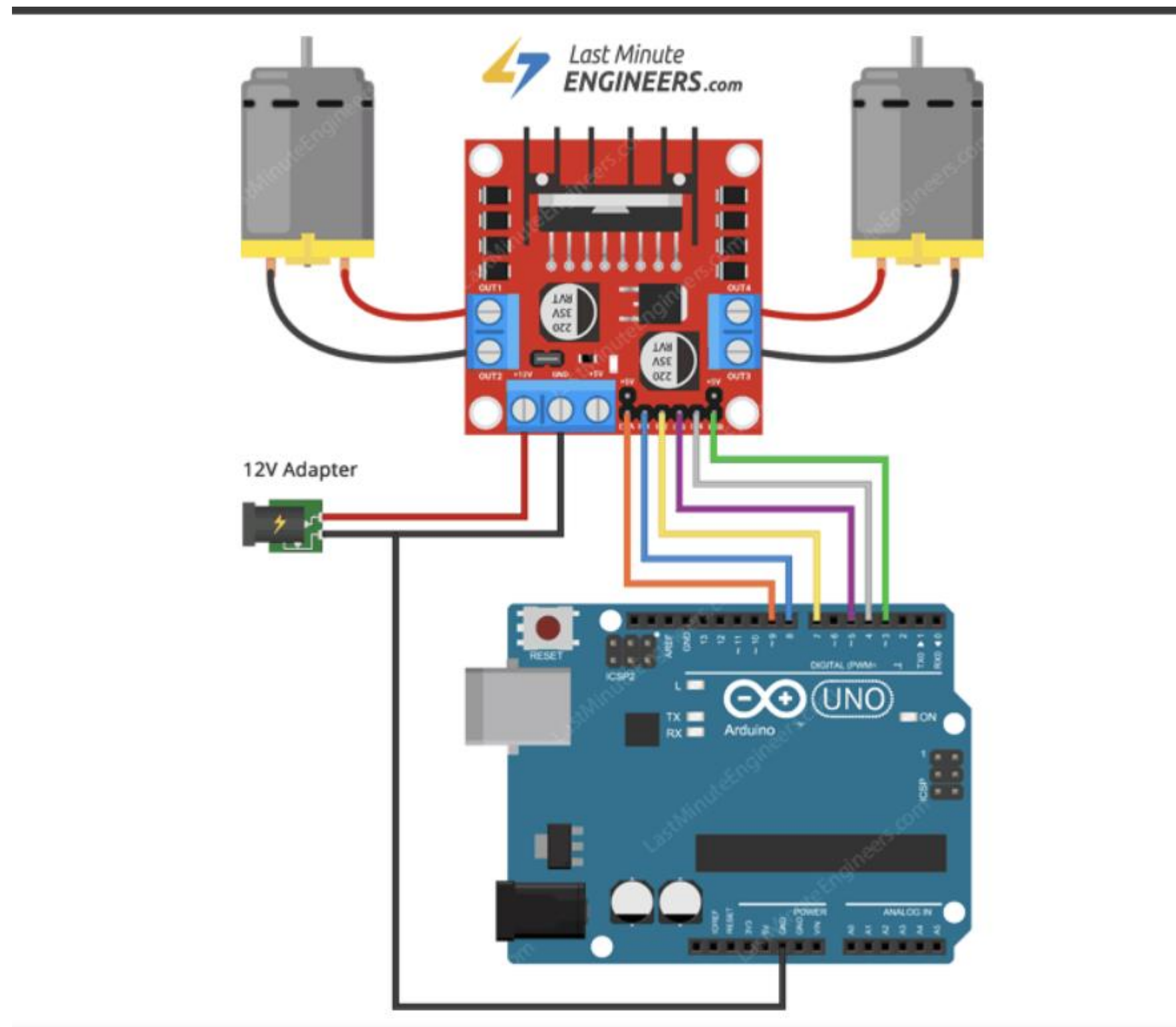
**IN1 & IN2** pins are used to control spinning direction of Motor A. When one of them is HIGH and other is LOW, the Motor A will spin. If both the inputs are either HIGH or LOW the Motor A will stop.

**IN3 & IN4** pins are used to control spinning direction of Motor B. When one of them is HIGH and other is LOW, the Motor B will spin. If both the inputs are either HIGH or LOW the Motor B will stop.

**ENB** pins are used to control speed of Motor B. Pulling this pin HIGH(Keeping the jumper in place) will make the Motor B spin, pulling it LOW will make the motor stop. Removing the jumper and connecting this pin to PWM input will let us control the speed of Motor B.

**OUT1 & OUT2** pins are connected to Motor A.

**OUT3 & OUT4** pins are connected to Motor B.





## Description

- As mentioned above L298N contains two pairs output which are connected to a pair of DC motors.
- The positive of battery is connected to the Power Input of the L298N module and negative is connected to GND.
- The 5V pin of the driver is connected to the Vin pin of the Arduino to power the Arduino board.
- Input 1 and Input 2 pins are used to control the direction of Motor 1 is connected to pin 13, pin 12 of the Arduino respectively.
- Input 3 and Input 4 pins are used to control the direction of Motor 2 is connected to pin 11, pin 10 of the Arduino respectively.
- Enable A and Enable B are connected to the pin 9 and pin 3 of Arduino, which are used to control the speed of motors using PWM.



```
// Motor A connections
int enA = 9;
int in1 = 13;
int in2 = 12;
// Motor B connections
int enB = 3;
int in3 = 11;
int in4 = 10;
void setup()
{
  pinMode(enA, OUTPUT);
  pinMode(enB, OUTPUT);
  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  pinMode(in3, OUTPUT);
  pinMode(in4, OUTPUT);
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
}
```





```
void loop()
{
directionControl();
delay(1000);
speedControl();
delay(1000);
}
void directionControl()
{ // Set motors to maximum speed
// PWM value ranges from 0 to 255
analogWrite(enA, 255);
analogWrite(enB, 255);
// Turn on motor A & B
digitalWrite(in1, HIGH);
digitalWrite(in2, LOW);
digitalWrite(in3, HIGH);
digitalWrite(in4, LOW);
delay(2000);
digitalWrite(in1, LOW);
digitalWrite(in2, HIGH);
digitalWrite(in3, LOW);
digitalWrite(in4, HIGH);
delay(2000);
}
```





```
// Turn off motors
digitalWrite(in1, LOW); digitalWrite(in2, LOW);
digitalWrite(in3, LOW); digitalWrite(in4, LOW);
}
void speedControl()
{
// Turn on motors
digitalWrite(in1, LOW);
digitalWrite(in2, HIGH);
digitalWrite(in3, LOW);
digitalWrite(in4, HIGH);
for (int i = 0; i < 256; i++)
{ analogWrite(enA, i);
analogWrite(enB, i);
delay(20);
}
for (int i = 255; i >= 0; --i)
{ analogWrite(enA, i);
analogWrite(enB, i);
delay(20); }
// Now turn off motors
digitalWrite(in1, LOW); digitalWrite(in2, LOW);
digitalWrite(in3, LOW); digitalWrite(in4, LOW); }
```





## Explanation

- In the setup section all the motor control pins are declared as output and are pulled LOW to make both motors OFF by default.
- In the loop function, two user-defined functions are called within a time delay of one second.
- directionControl() function is defined for rotating both motors in clockwise, anti-clockwise direction and stop for 2 seconds each.
- speedControl() is defined to vary the speed of the motor by varying PWM from 0 to 100% (0 – 255) duty cycle.

<https://electrosome.com/interfacing-l298n-motor-driver-arduino-uno/>

[https://www.youtube.com/watch?v=9Qjs\\_dmVJZw](https://www.youtube.com/watch?v=9Qjs_dmVJZw)