



Prof. Me. Hélio Esperidião

Motor DC e ponte H

O motor DC (direct current)

Também é chamado de motor CC (corrente contínua), devido ao fato de ser alimentado por corrente contínua.

Grande parte dos aparelhos eletrônicos possui recursos mecânicos automatizados constituídos por motores de corrente contínua.

Operação do motor DC

Um condutor, que é percorrido por uma corrente elétrica, é colocado em um campo magnético, ele experimenta uma força eletromagnética, gerando o torque e, conseqüentemente, o giro do eixo do motor.

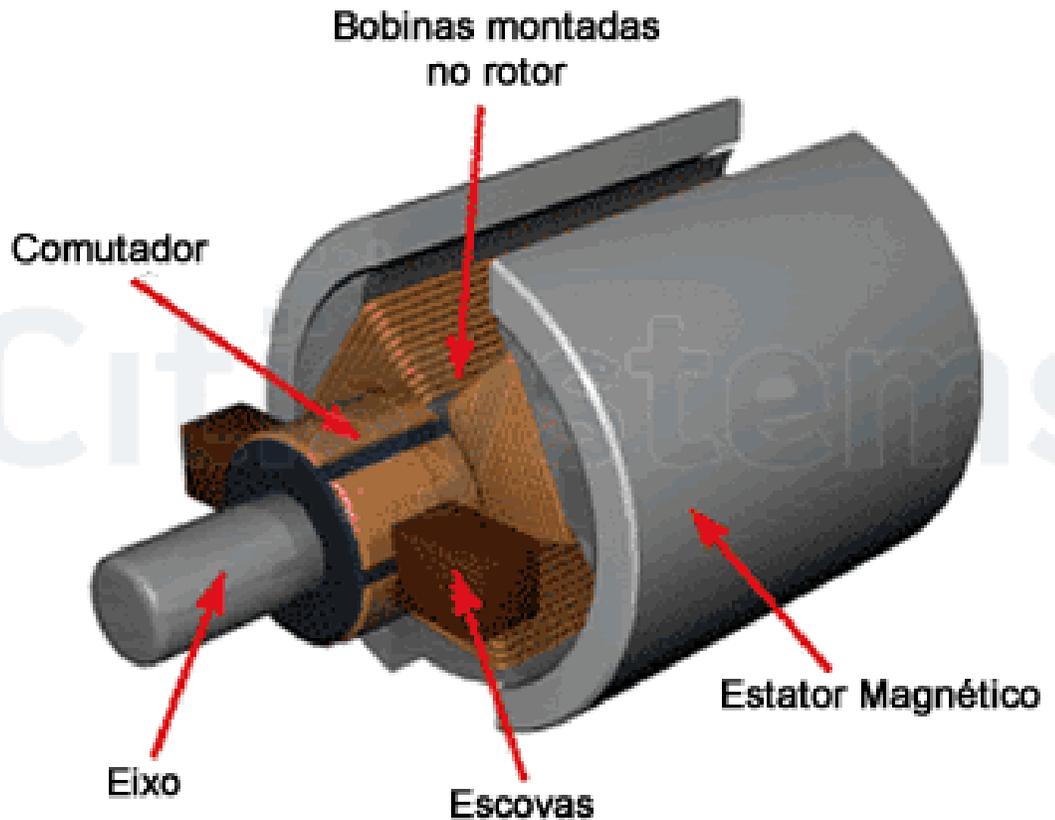
Desta forma, o **motor DC** funciona aproveitando as forças de atração e repulsão geradas por eletroímãs ou ímãs permanentes.

Velocidade

- A velocidade do **motor DC** pode ser controlada apenas variando a sua tensão.
- Ao escolher um **motor DC** para o seu projeto de robótica, é importante identificar algumas características, como: especificações de desempenho, requisitos de potência e tamanho.
- Considere também, o ambiente no qual o robô será utilizado.

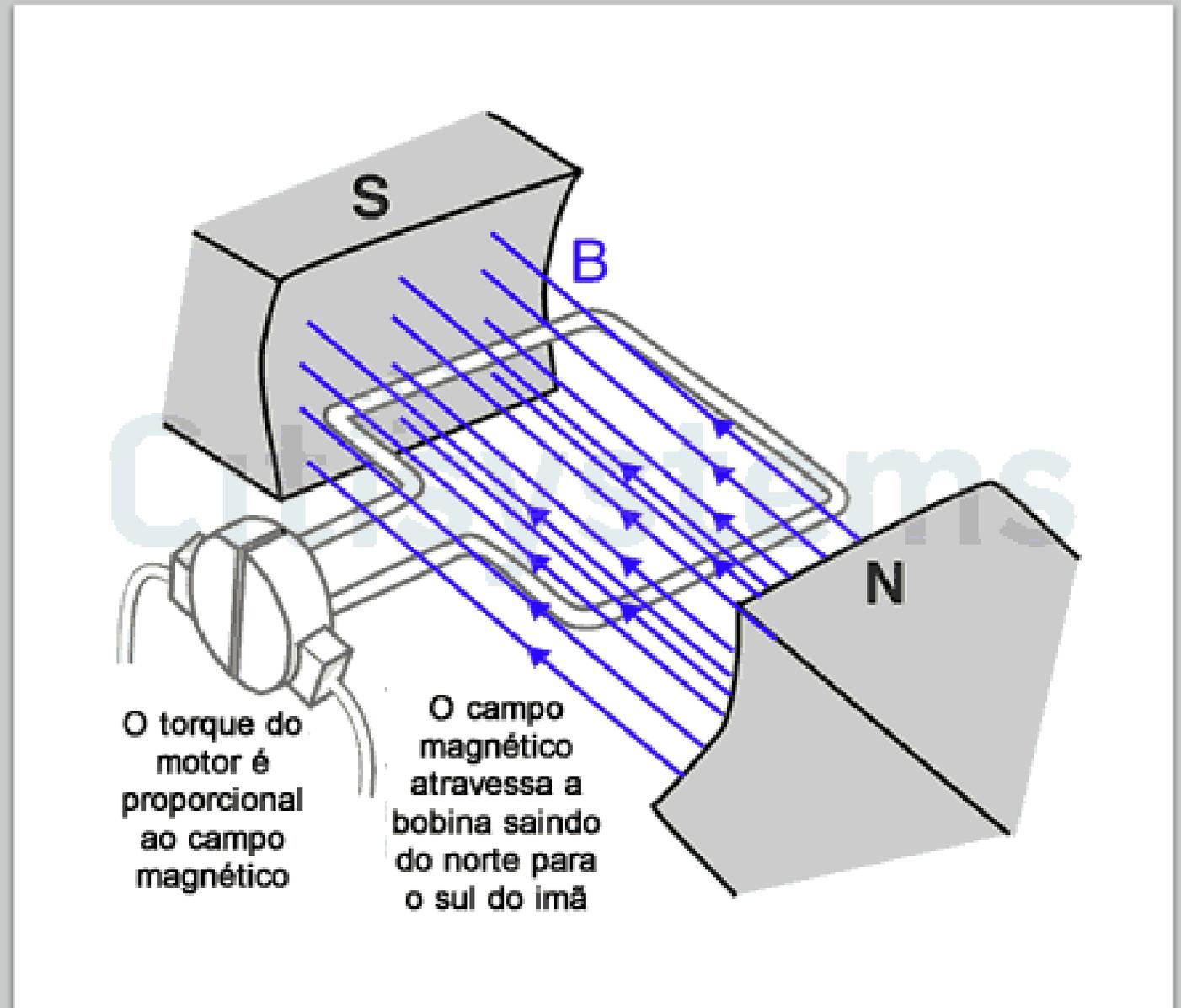
Funcionamento

- Um motor CC é composto por um eixo acoplado ao rotor que é a parte girante do motor.
- O estator é composto por um ímã e o comutador tem a função de transferir a energia da fonte de alimentação ao rotor.



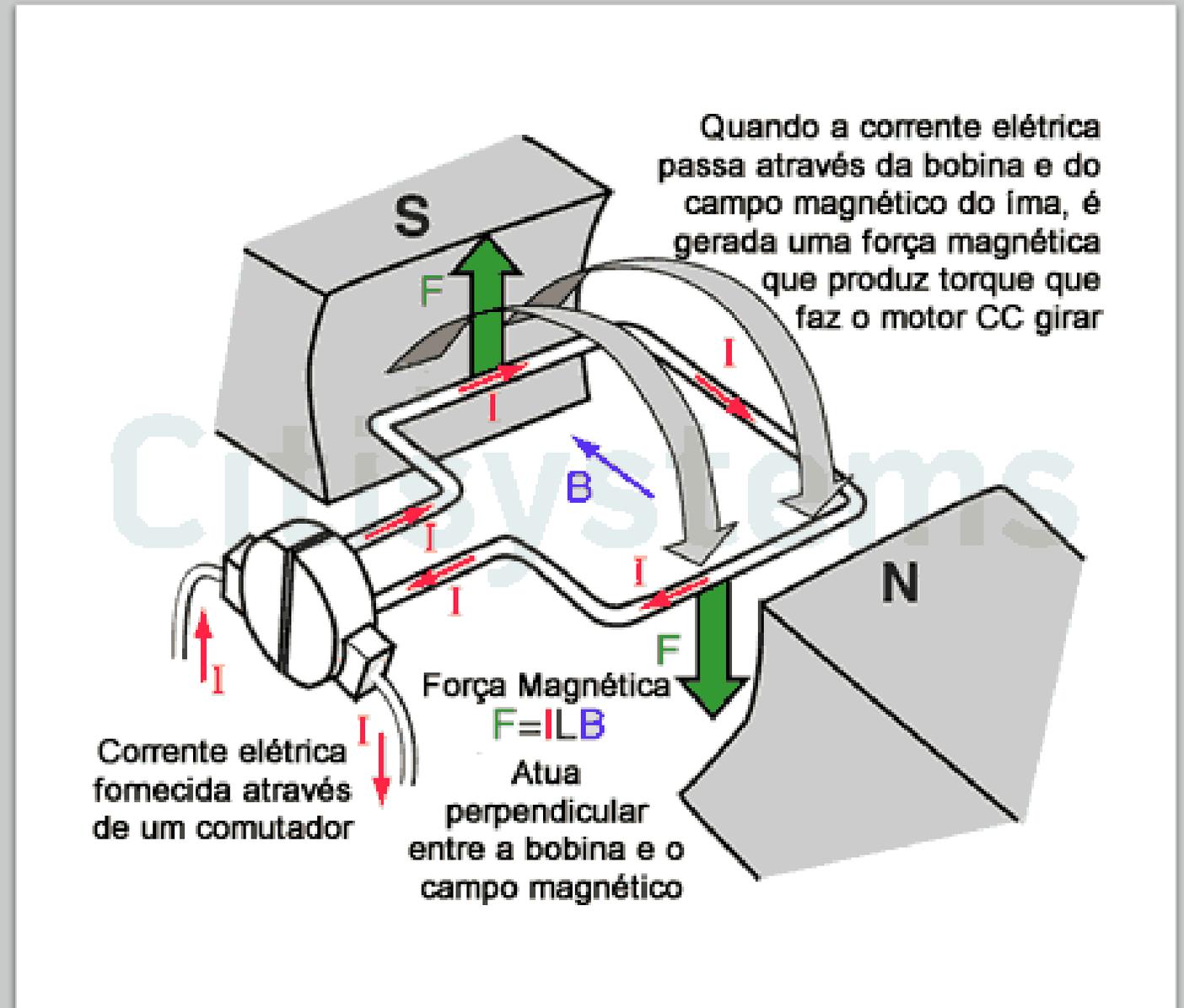
O Campo Magnético

- O campo magnético é gerado entre os polos norte e sul do ímã e possui um sentido partindo do norte para o sul.
- O torque que vai impulsionar a bobina e por sua vez, o rotor será proporcional ao campo magnético entre os ímãs.



O Princípio de Funcionamento

- o estator é constituído pelos ímãs (norte e sul) e o rotor é representado por uma bobina que é alimentada pelo comutador em que circula uma corrente I .
- O princípio básico de funcionamento do motor CC é o seguinte: “Sempre que um condutor conduzindo uma corrente elétrica (em vermelho) é colocado em um campo magnético (em azul), este condutor experimenta uma força mecânica (em verde)”, gerando o torque e o giro do eixo do motor.



Ponte H – O que é e como funciona?

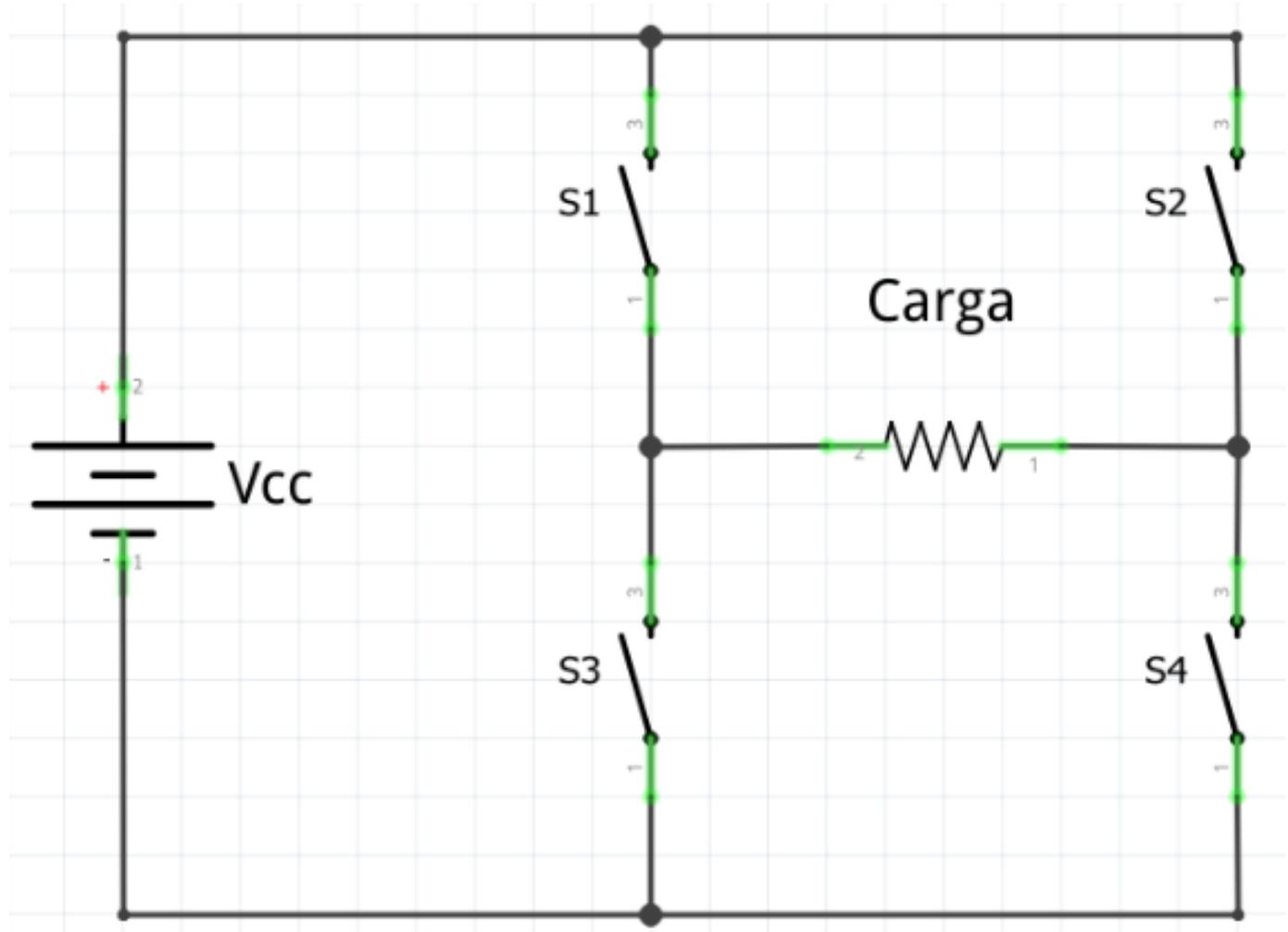
O microcontrolador é um dispositivo muito simples e frágil, com ele não é possível alimentar cargas que consomem muita corrente.

Para isso é necessário criar um circuito externo

No caso de motores, existe um circuito específico chamado ponte h.

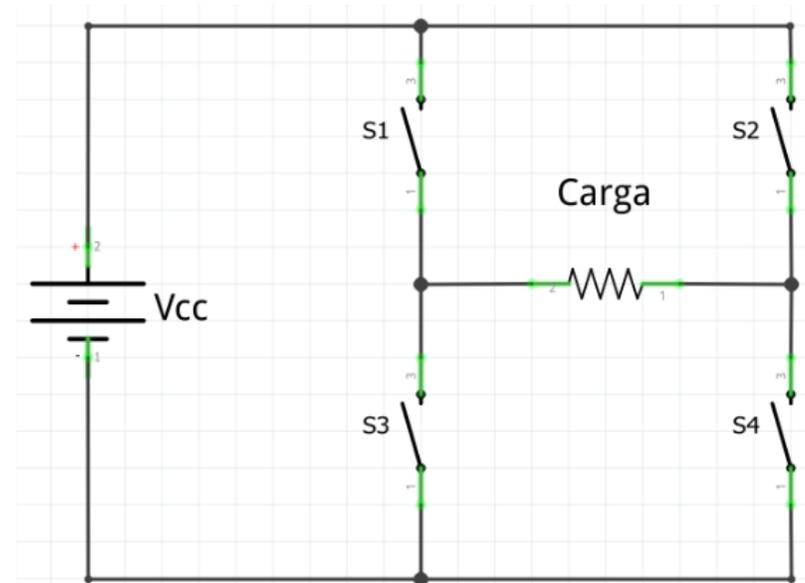
O que é?

- Acionando a chave **S1** e **S4** a corrente flui em um sentido;
- Acionando a chave **S2** e **S3**, a corrente flui no sentido contrário;
- Acionando **S1** e **S2** ou **S3** e **S4** nada acontece.



Parte de chaves

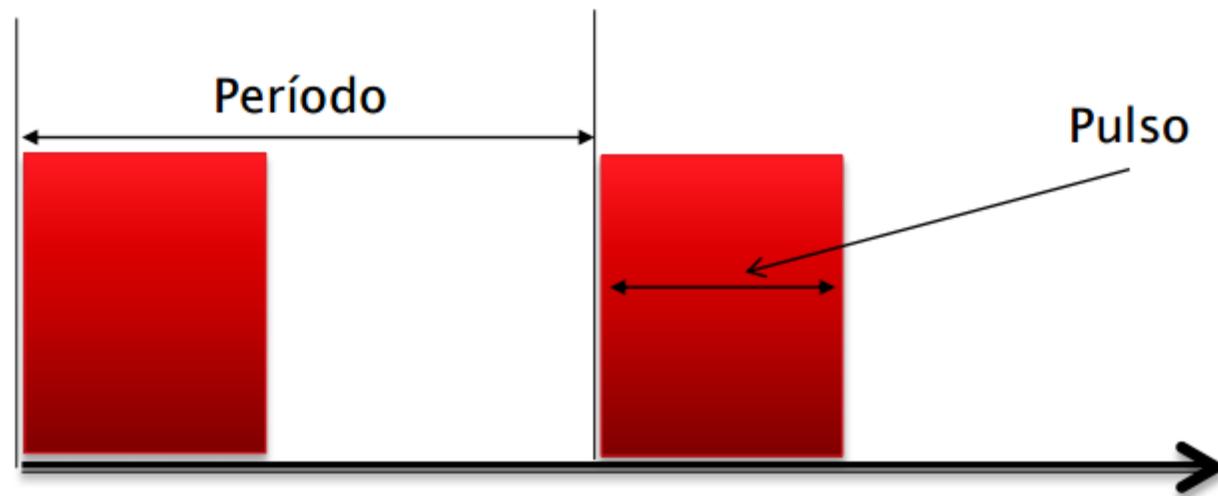
- A principal diferença é que, na ponte h, nós mantemos um par de chave fixo (S1 e S4 ou S2 e S3). Isto é, a corrente só fluirá em um sentido ao longo do funcionamento da carga. É claro que, se você quiser alterar o sentido de rotação do motor, outro par de chaves deve ser acionado.
- É possível também fazermos o controle da tensão em cima do motor de modo a controlar sua velocidade. Abaixo explicarei como isso é feito.



PWM (Pulse Width Modulation)

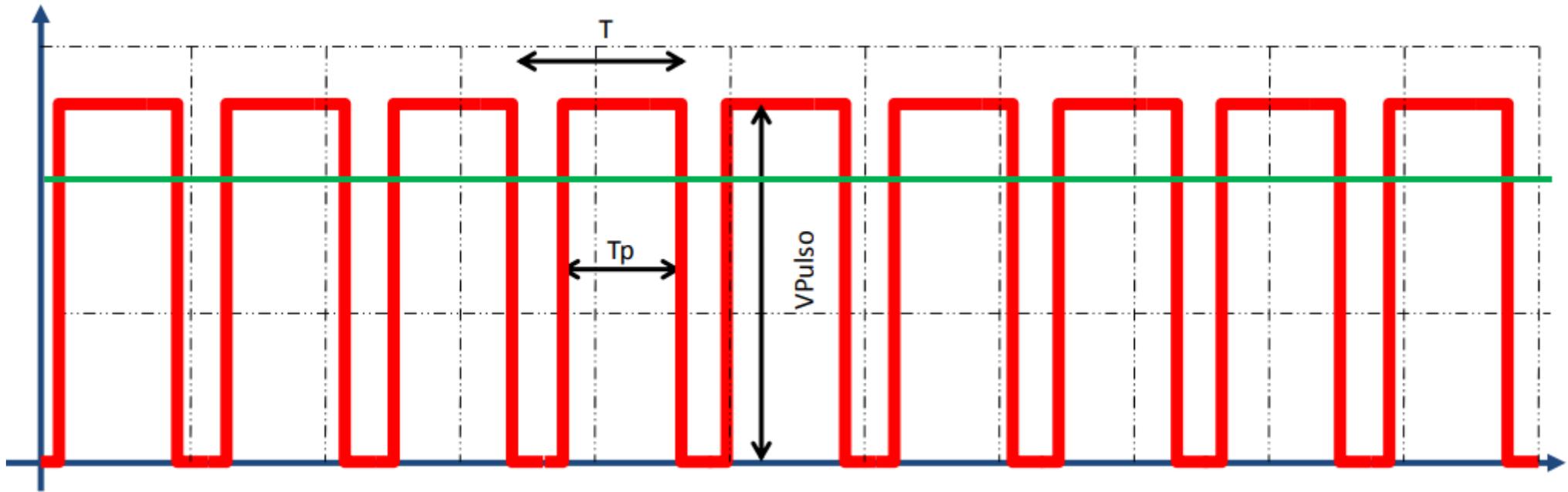
- PWM (Pulse Width Modulation – Modulação por Largura de Pulso) é uma técnica para obter resultados analógicos por meios digitais.
- Essa técnica consiste na geração de uma onda quadrada em uma frequência muito alta em que pode ser controlada a porcentagem do tempo em que a onda permanece em nível lógico alto.
- Esse tempo é chamado de Duty Cycle (Ciclo de trabalho) e sua alteração provoca mudança no valor médio da onda, indo desde 0V (0% de Duty Cycle) a 5V (100% de Duty Cycle) no caso do Arduino.

Período vs Pulso



Exemplo de PWM

- T_p : time pulse
- V_{pulso} : voltage pulse



pwm

- É um sinal de frequência constante e largura de pulso (ciclo ativo ou duty cycle) variável.
- A tensão média equivalente de um sinal PWM é:

$$V_M = \frac{1}{T} \int_0^T V(t) dt \rightarrow \text{onde } T \text{ é o período do sinal.}$$

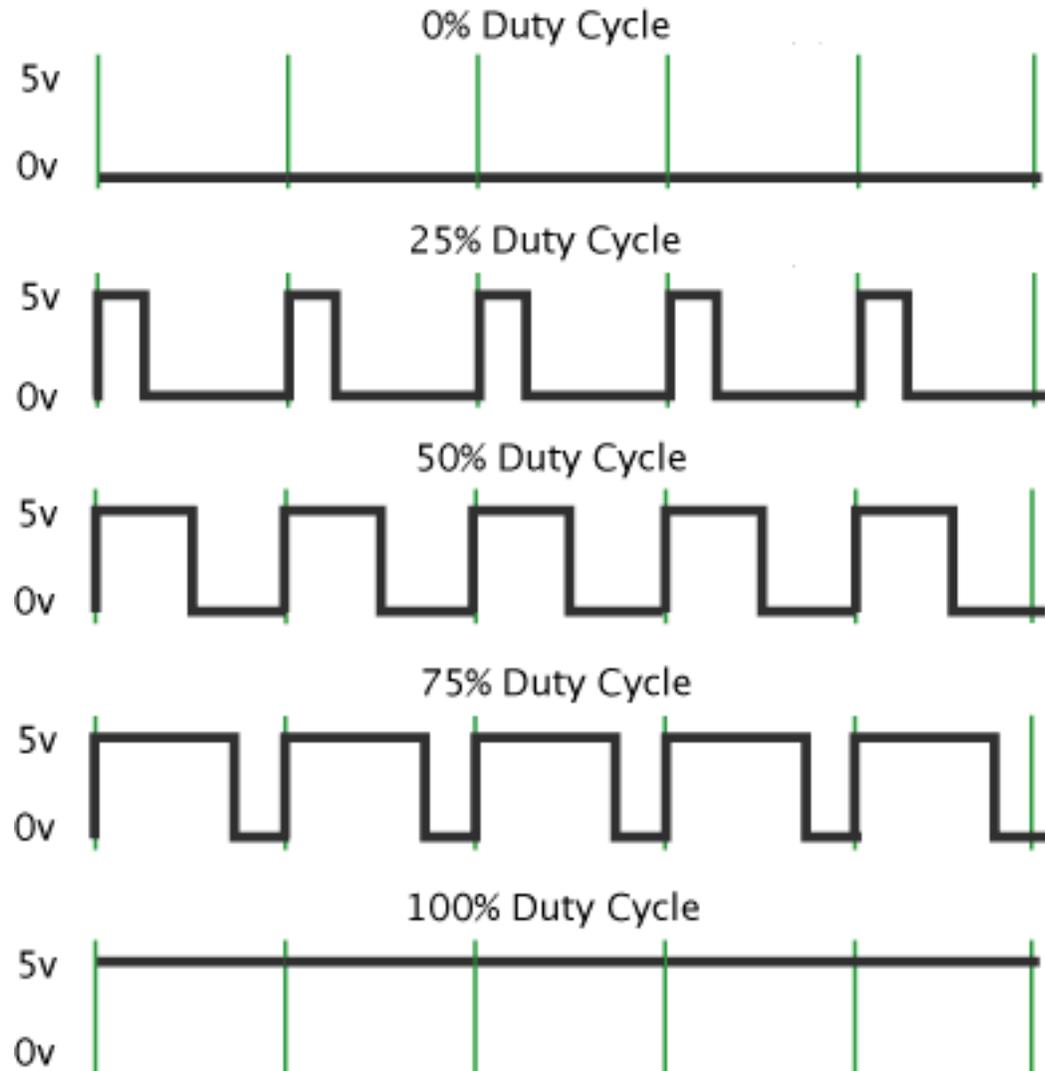
- Se um sinal PWM ao longo do tempo é:

$$V(t) = \begin{cases} V_{pulso}, & 0 \leq t \leq t_p \\ 0, & t_p \leq t \leq T \end{cases} \rightarrow \text{onde } t_p \text{ e } V_{pulso} \text{ são a duração e a tensão do pulso em nível alto.}$$

Intensidade da lâmpada:



Pulse Width Modulation



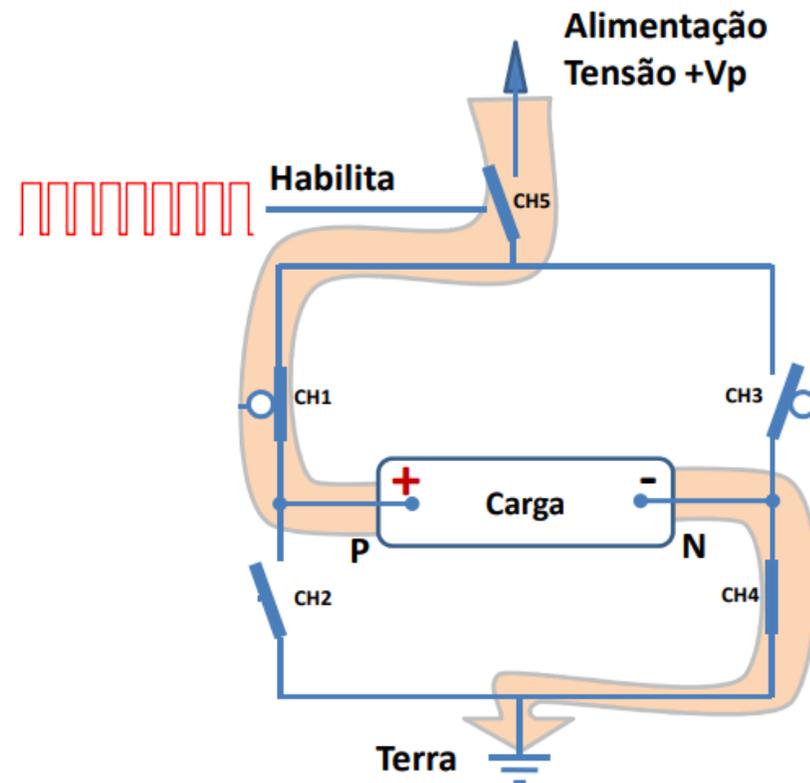
Duty Cycle(8Bits)

- O valor do Duty Cycle pode variar de acordo com a resolução.
- Trabalhando com 8 bits seu valor vai de 0 (0%) a 255 (100%).
- 8 bits = $2^8 = 256$
- 9 bits = $2^9 = 512$
- 10bits = $2^{10} = 1024$
- 12 bits = $2^{12} = 4096$

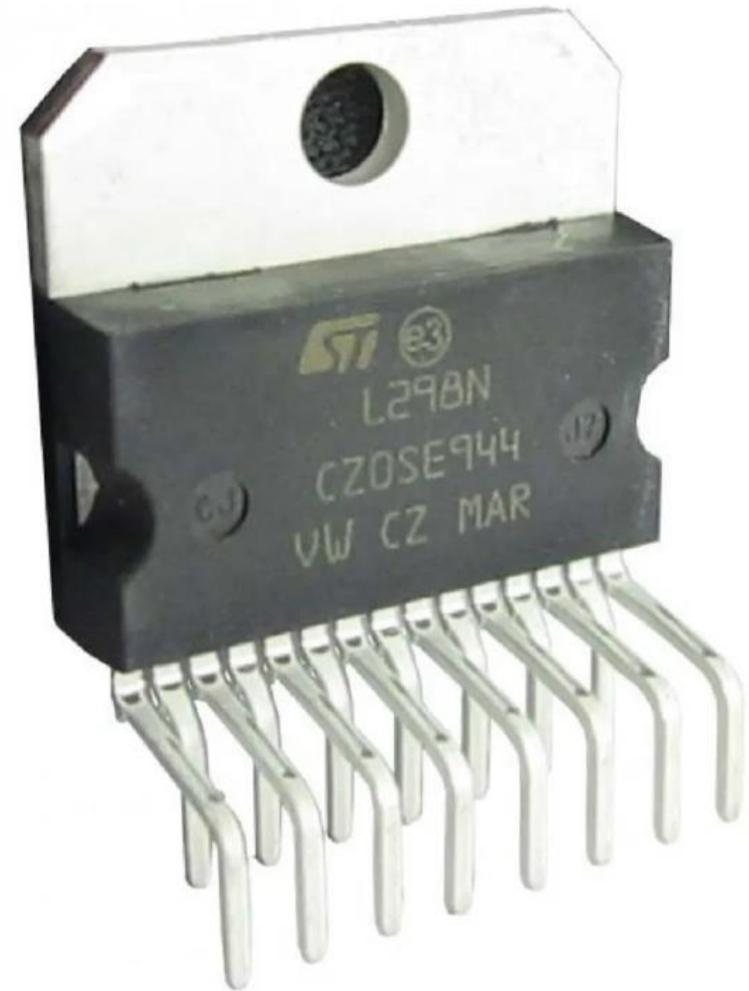
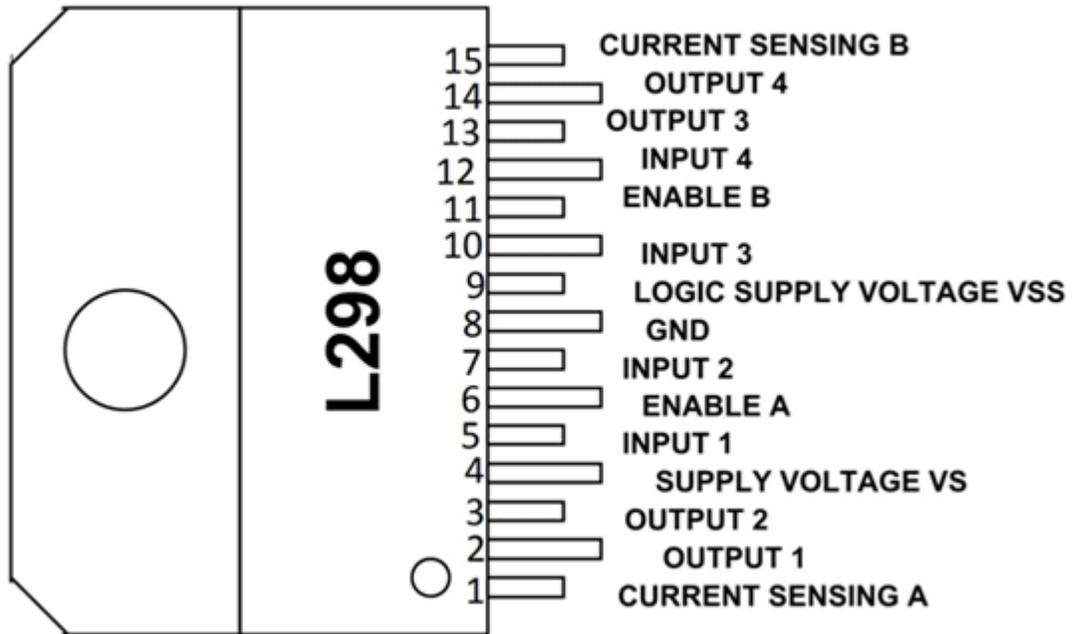
PWM na prática

- Observe que CH5 é ligada e desligada de acordo com sinal pwm.
- Habilitar e desabilitar CH5 em “alta” frequência provoca um comportamento integral.

$$V_M = \frac{1}{T} \int_0^T V(t) dt$$

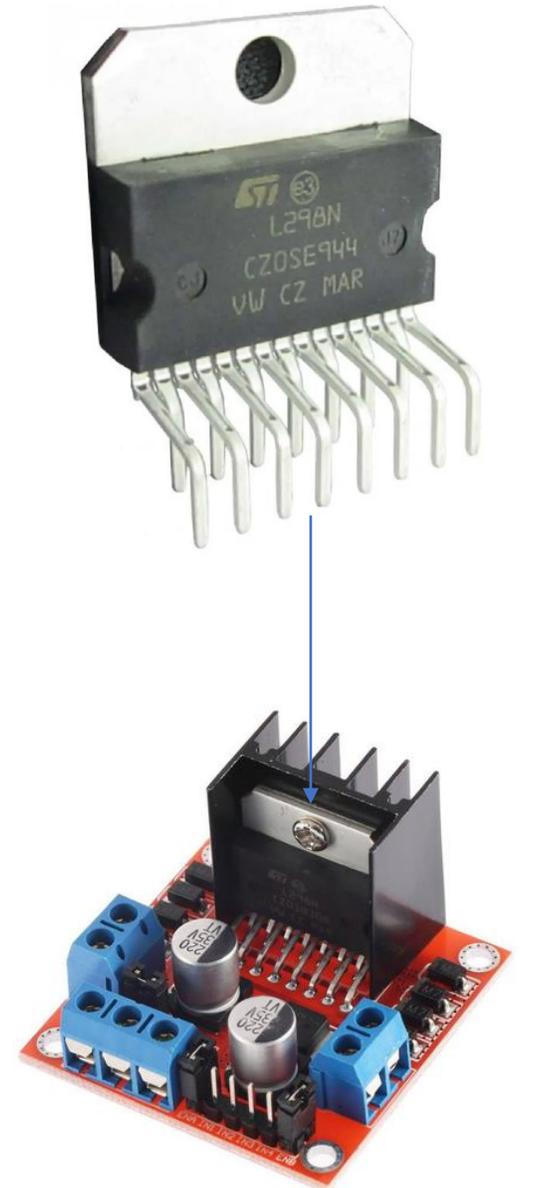


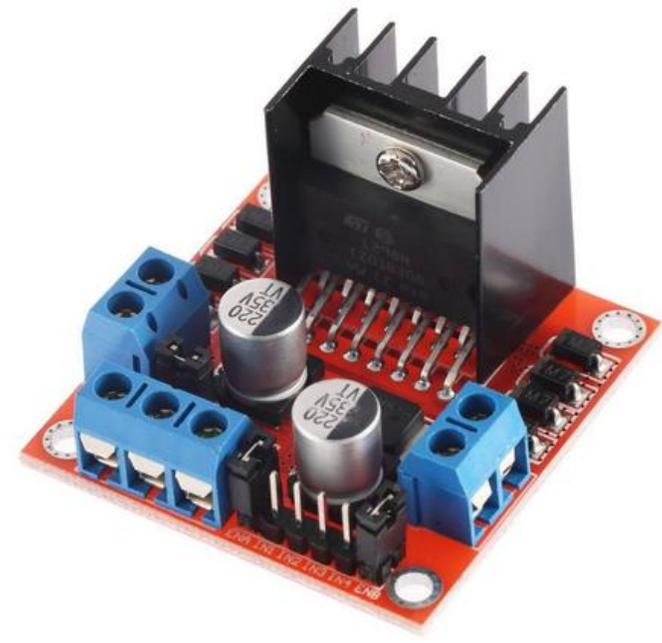
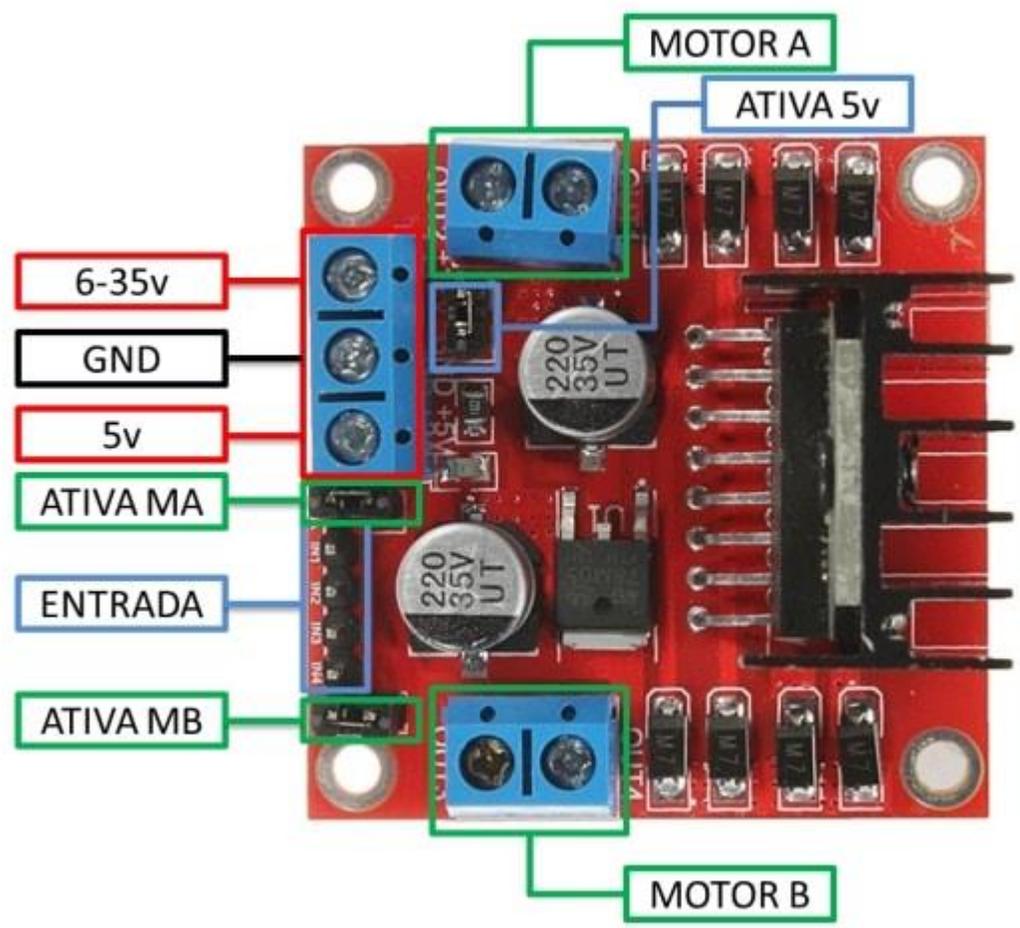
L298N – pin out

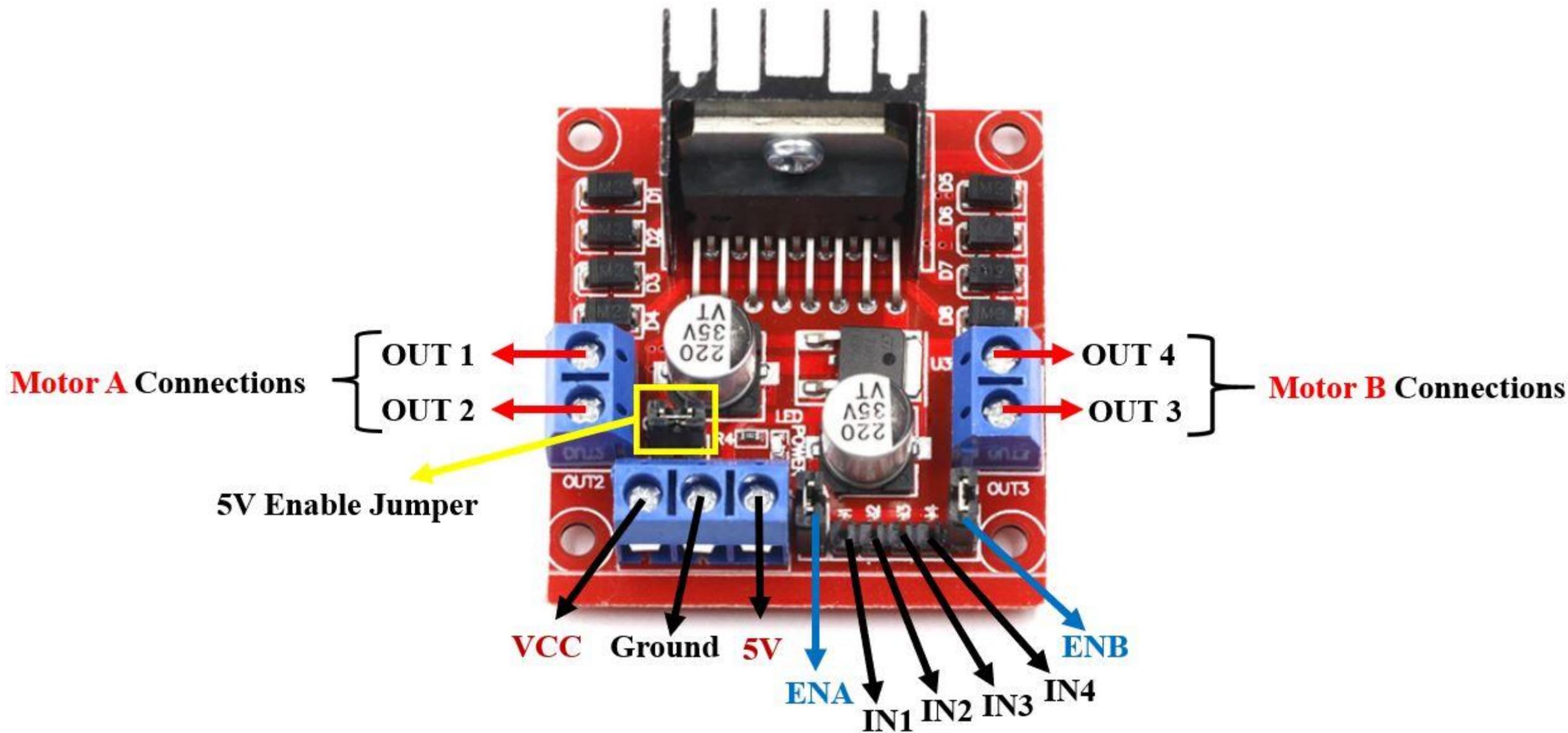


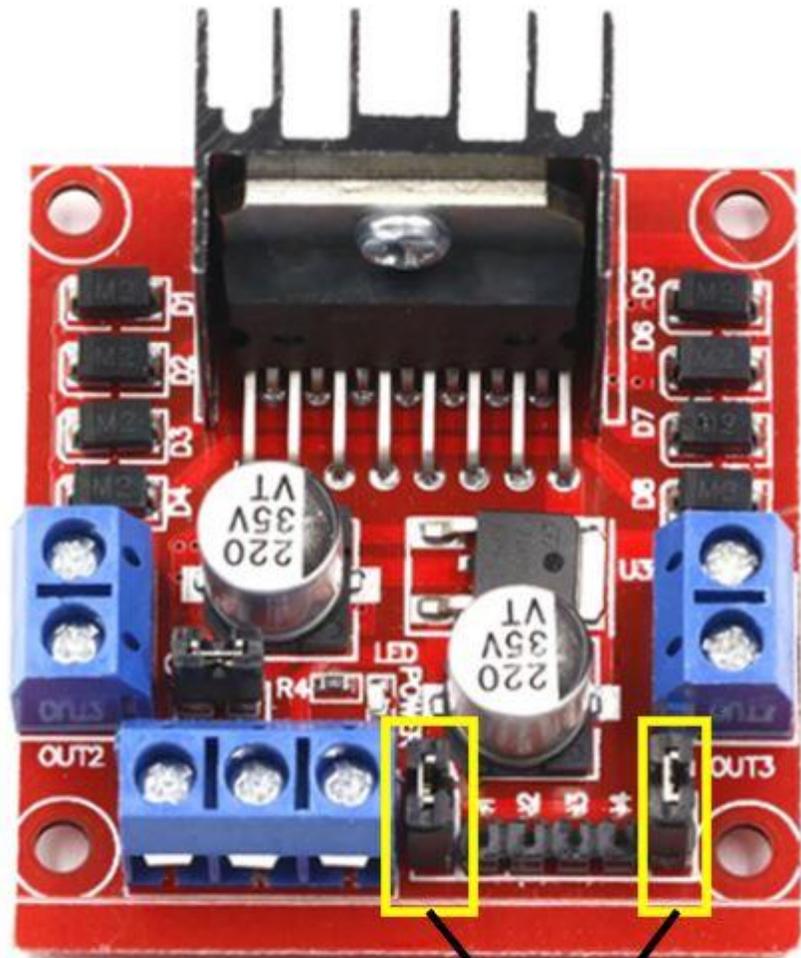
Kit - Ponte H L298N

- Tensão de Operação: 4~35v
- Chip: ST L298N
- Controle de 2 motores DC ou 1 motor de passo
- Corrente de Operação máxima: 2A por canal ou 4A max
- Tensão lógica: 5v
- Corrente lógica: 0~36mA
- Limites de Temperatura: -20 a +135°C
- Potência Máxima: 25W
- Dimensões: 43 x 43 x 27mm
- Peso: 30g

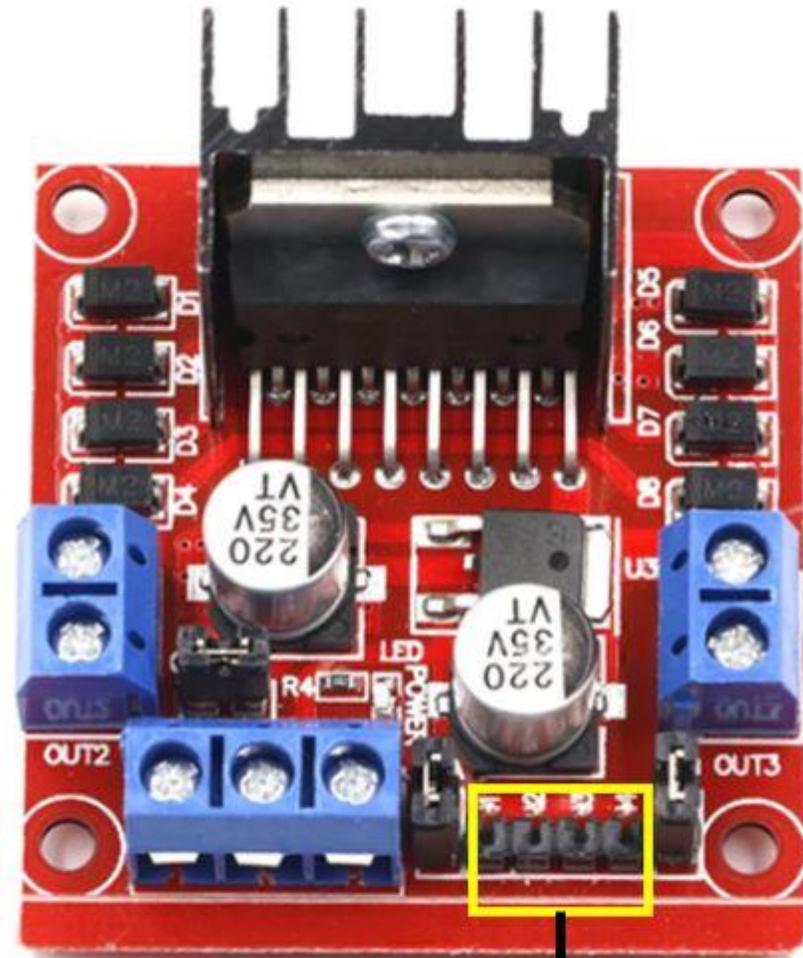






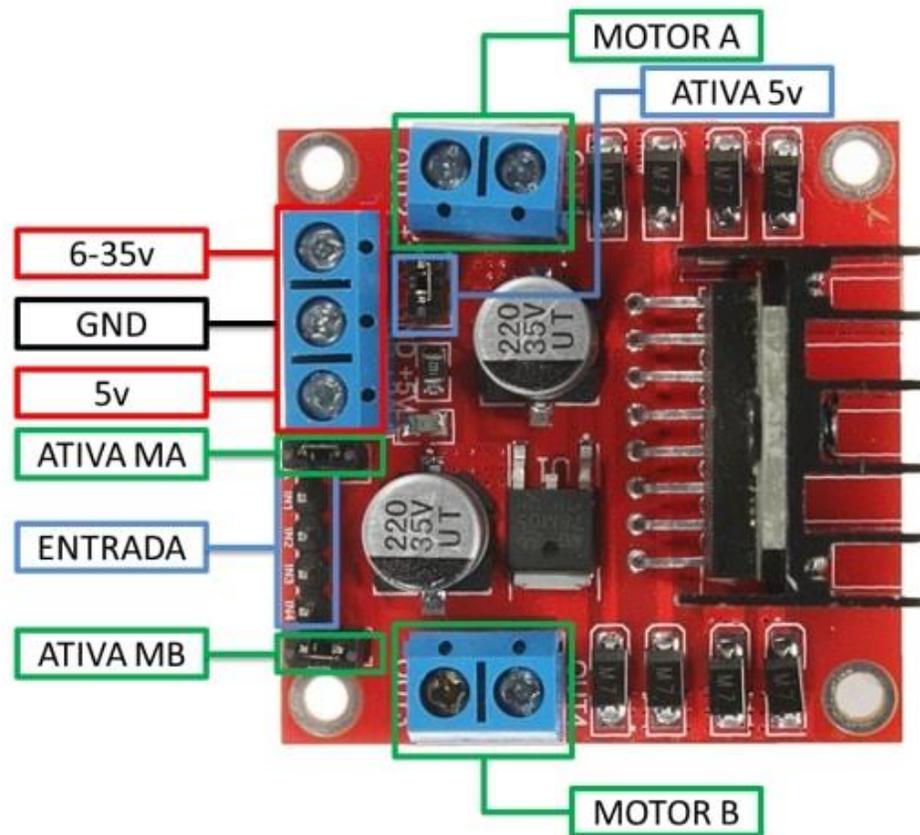


**Speed Control
(ENABLE) PINS**



**Direction Control
(INPUT) PINS**

Descrição das portas

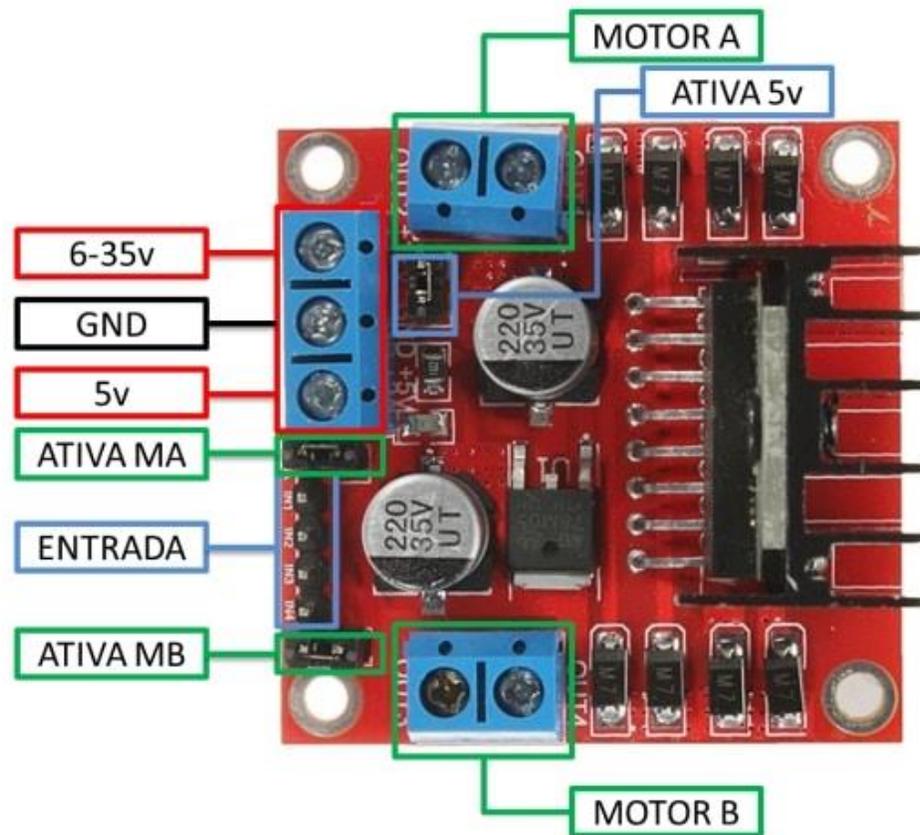


(Motor A) e (Motor B) se referem aos conectores para ligação de 2 motores DC ou 1 motor de passo

(Ativa MA) e (Ativa MB) – são os pinos responsáveis pelo controle PWM dos motores A e B. Se estiver com jumper, não haverá controle de velocidade, pois os pinos estarão ligados aos 5v. Esses pinos podem ser utilizados em conjunto com os pinos PWM do Arduino

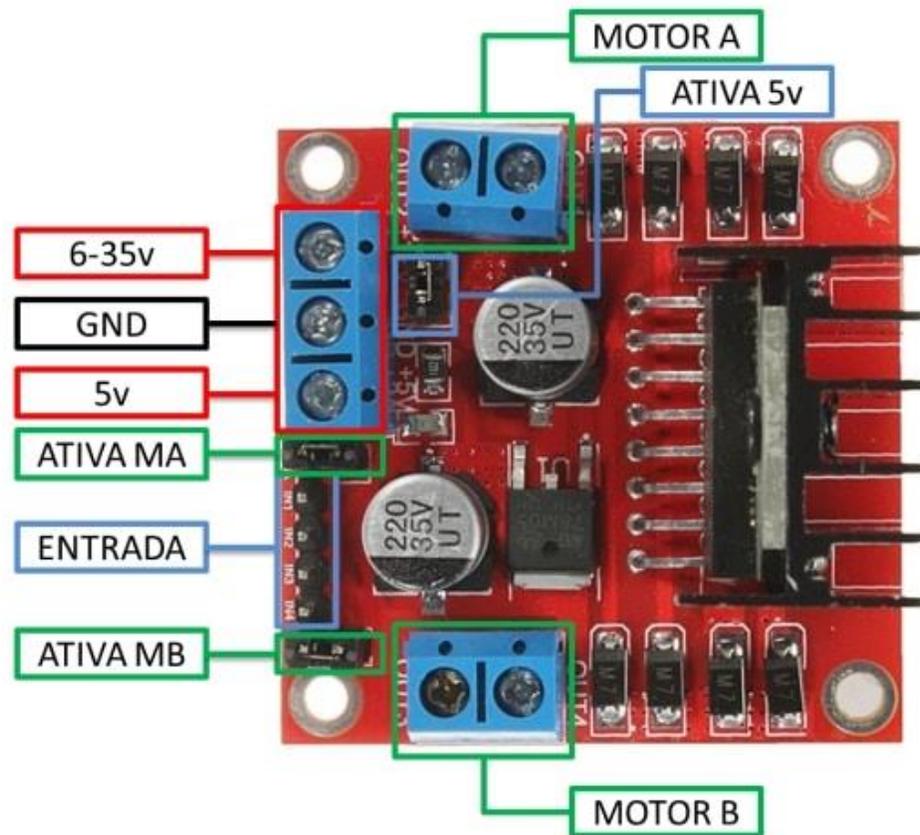
(6-35v) e (GND) – Aqui será conectada a fonte de alimentação externa quando o driver estiver controlando um motor que opere entre 6-35v. Por exemplo se estiver usando um motor DC 12v, basta conectar a fonte externa de 12v neste pino e (GND).

Descrição das portas

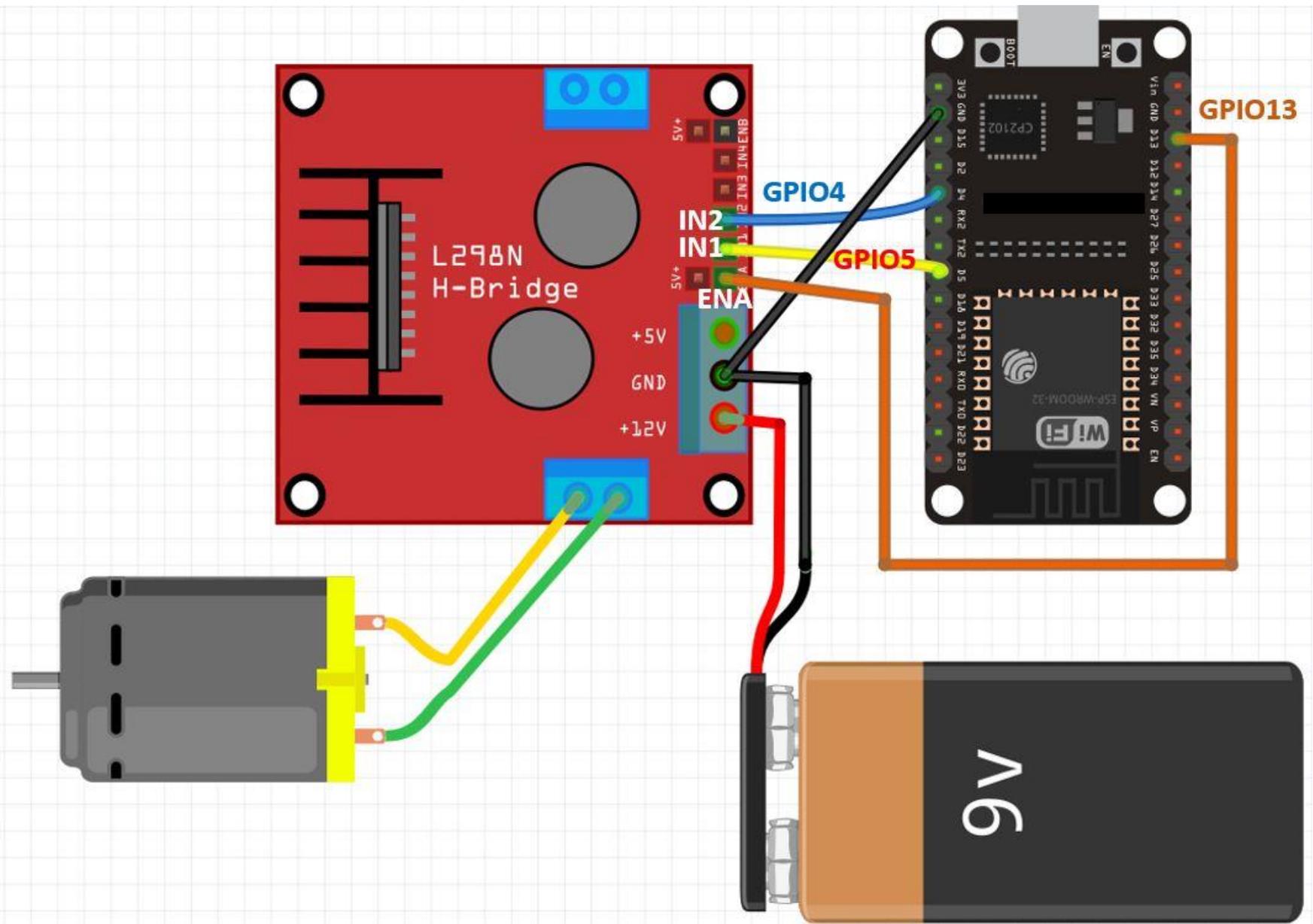


- (Ativa 5v) e (5v) – Este Driver Ponte H L298N possui um regulador de tensão integrado. Quando o driver está operando entre 6-35V, este regulador disponibiliza uma saída regulada de +5v no pino (5v) para um uso externo (com jumper), podendo alimentar por exemplo outro componente eletrônico.
- Não alimente o pino de (5v) com +5v se estiver controlando um motor de 6-35v e jumper conectado, isto danificará a placa. O pino (5v) somente se tornará uma entrada caso esteja controlando um motor de 4-5,5v (sem jumper), assim poderá usar a saída +5v.

Descrição das portas



- (Entrada) – Este barramento é composto por IN1, IN2, IN3 e IN4. Sendo estes pinos responsáveis pela rotação do Motor A (IN1 e IN2) e Motor B (IN3 e IN4).



```
int ENA = 13;
int IN1 = 5;
int IN2 = 4;
const int frequency = 500;
const int pwm_channel = 0;
const int resolution = 8;
void setup() {
  pinMode(ENA, OUTPUT);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
  ledcSetup(pwm_channel, frequency, resolution);
  ledcAttachPin(ENA, pwm_channel);
}
void loop() {
  mudarDirecao();
  delay(1000);
  mudarVelocidade();
  delay(1000);
}
```

mudarDirecao()

```
void mudarDirecao() {  
    ledcWrite(pwm_channel, 255);  
    digitalWrite(IN1, HIGH);  
    digitalWrite(IN2, LOW);  
    delay(5000);  
    digitalWrite(IN1, LOW);  
    digitalWrite(IN2, HIGH);  
    delay(5000);  
    digitalWrite(IN1, LOW);  
    digitalWrite(IN2, LOW);  
}
```

mudarVelocidade()

```
void mudarVelocidade() {  
    digitalWrite(IN1, LOW);  
    digitalWrite(IN2, HIGH);  
    for (int i = 0; i < 256; i++) {  
        ledcWrite(pwm_channel,i);  
        delay(20);  
    }  
    for (int i = 255; i >= 0; --i) {  
        ledcWrite(pwm_channel,i);  
        delay(20);  
    }  
    digitalWrite(IN1, LOW);  
    digitalWrite(IN2, LOW);  
}
```