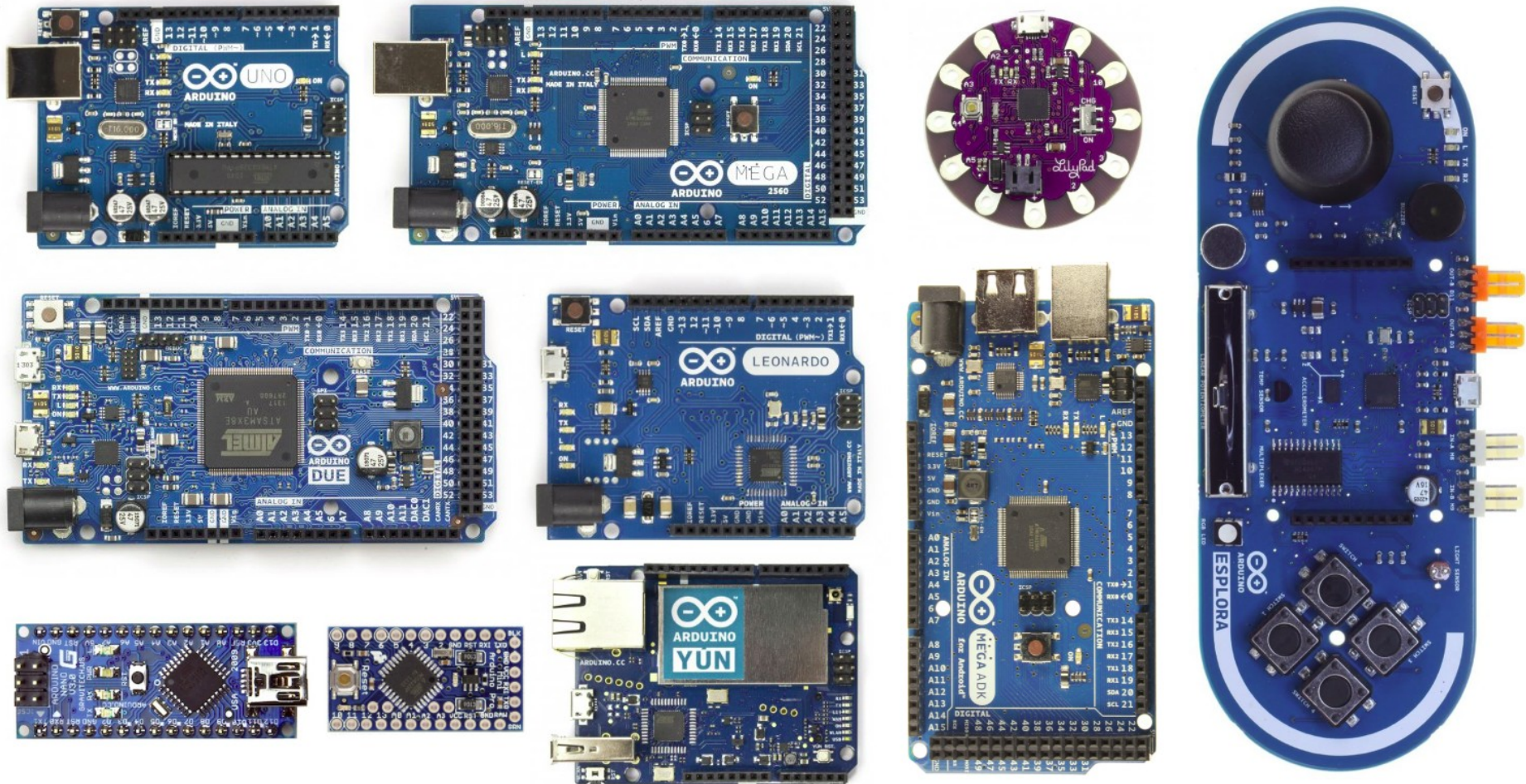


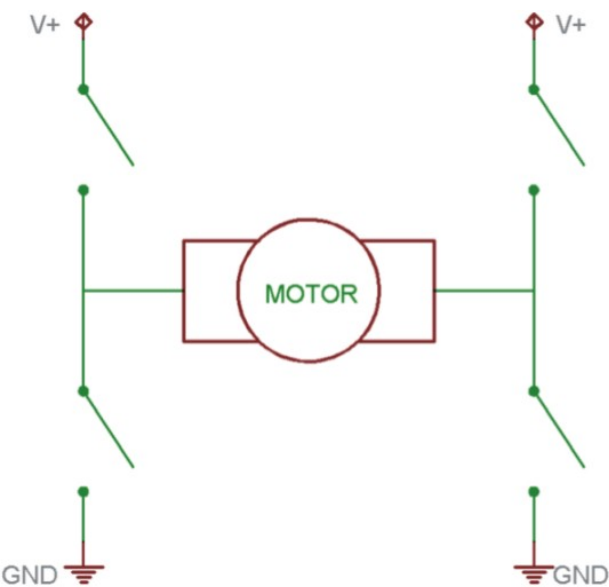


# Curso Arduino – Aula 7

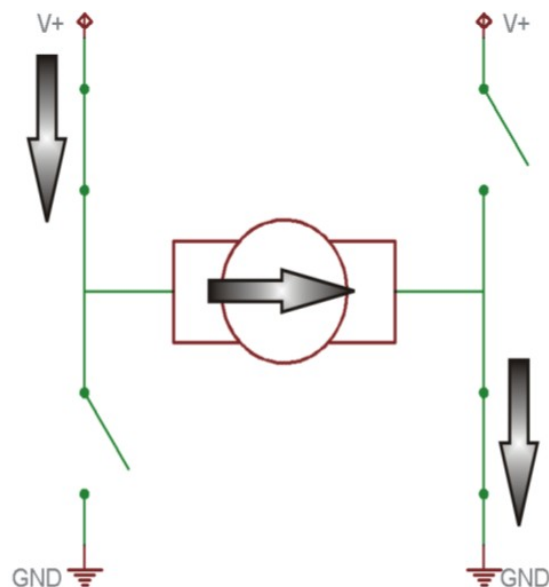


# Funcionamento da Ponte H

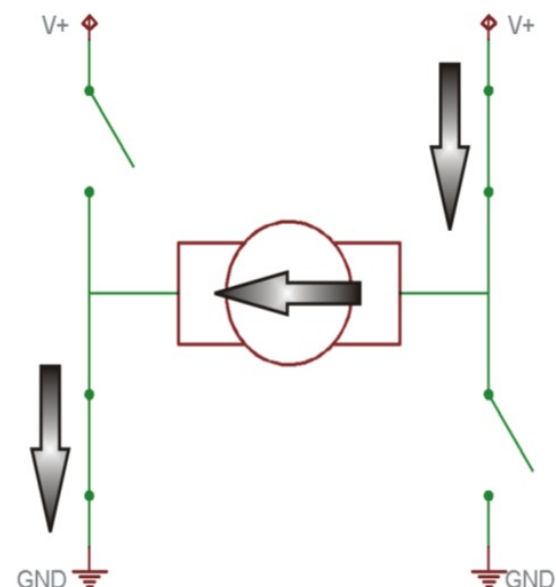
O funcionamento da ponte H é fácil de entender, ela consiste em quatro dispositivos de acionamento que faz o motor girar no sentido horário e anti-horário.



Motor Parado

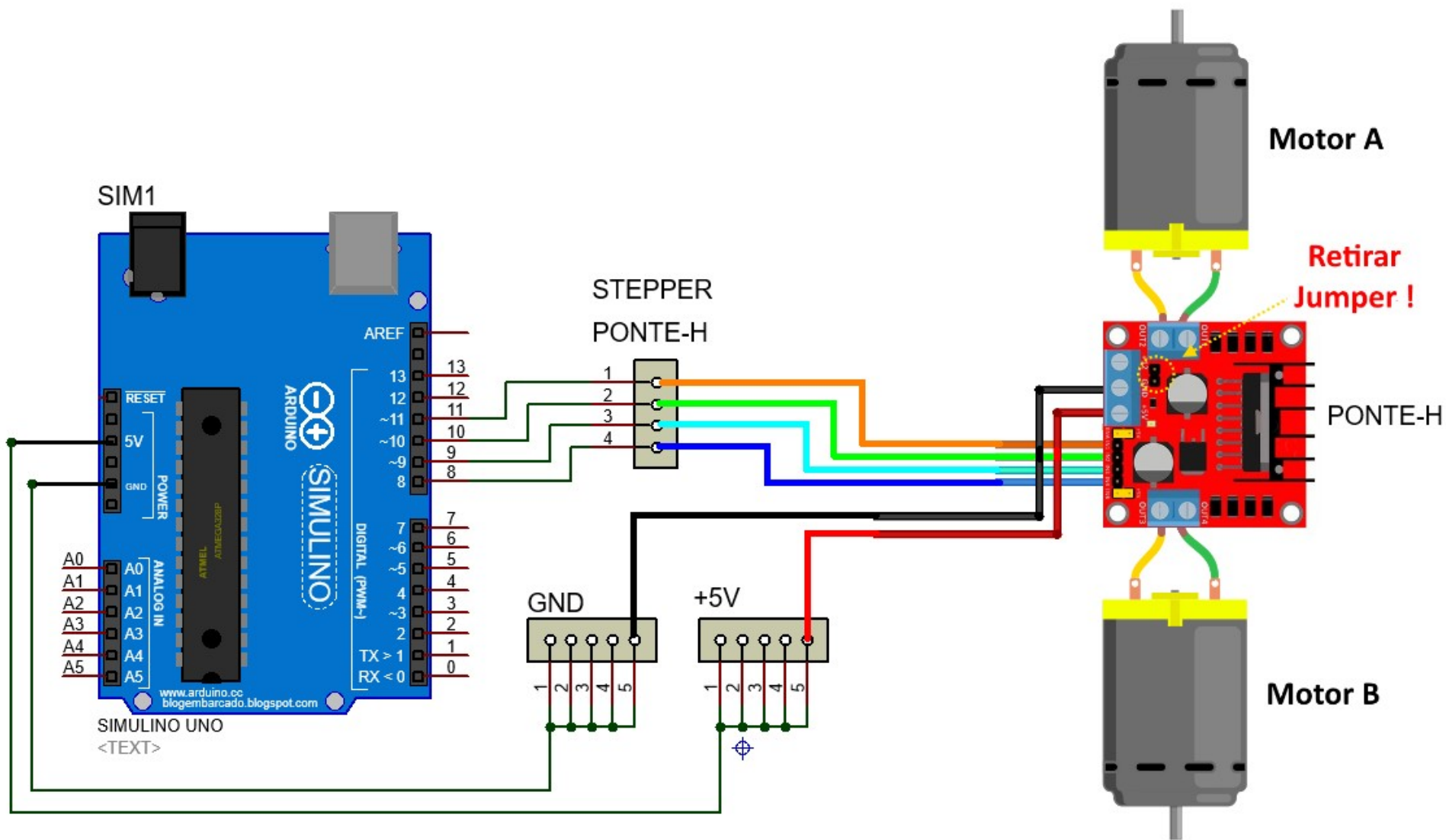


Motor no  
Sentido horário



Motor no  
Sentido  
Anti-horário

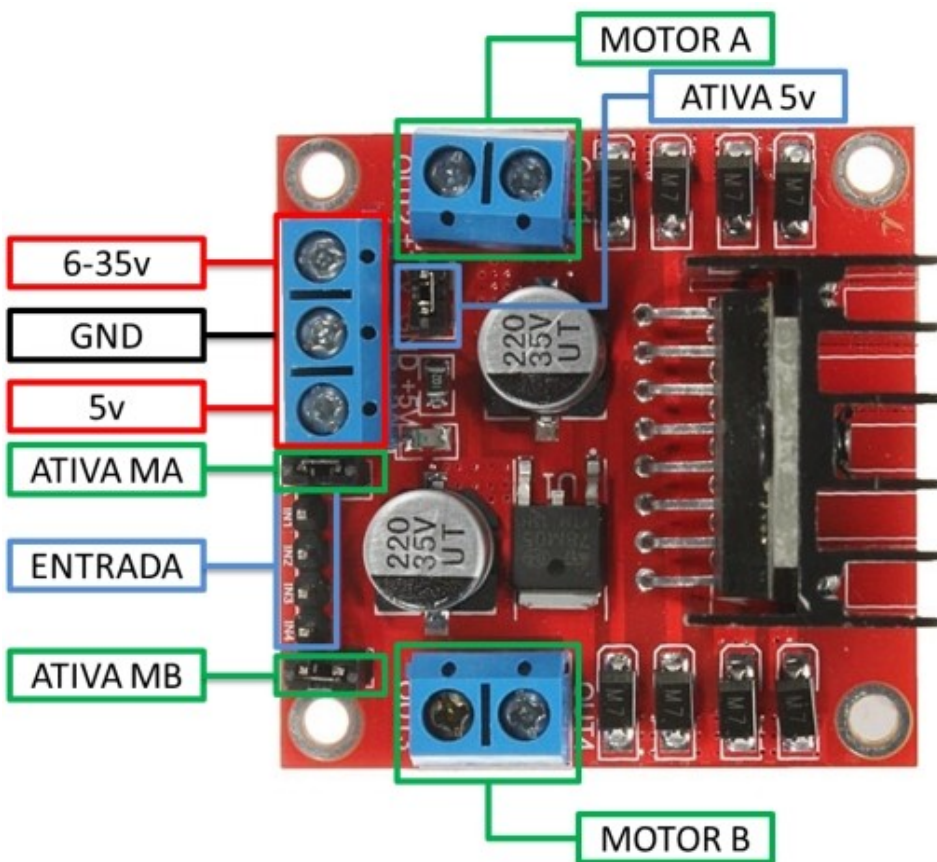
# Funcionamento da Ponte H





# Funcionamento da Ponte H

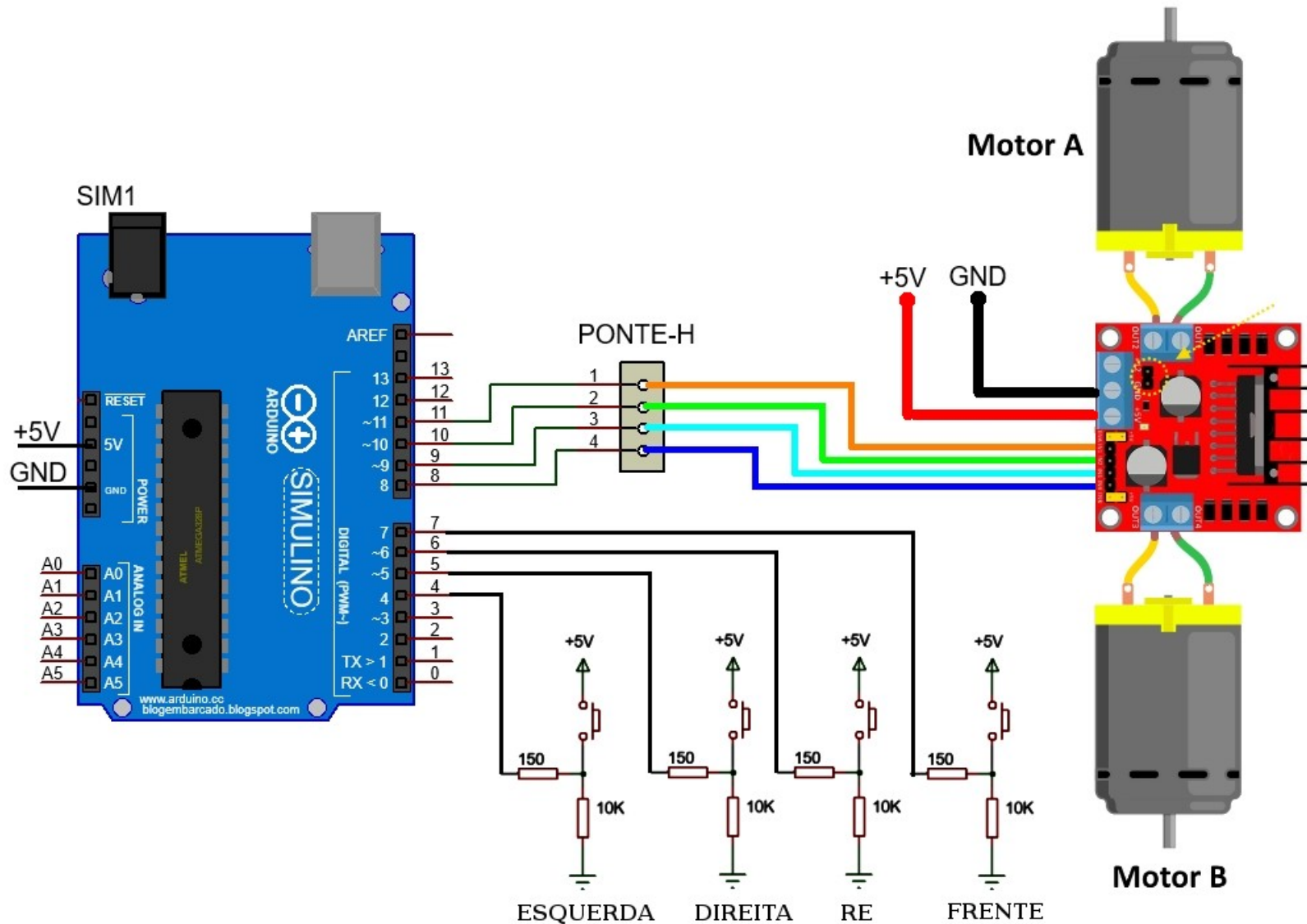
A seguir está os detalhes de entrada do drive da Ponte H e níveis de tensão para o acionamento dos motores.

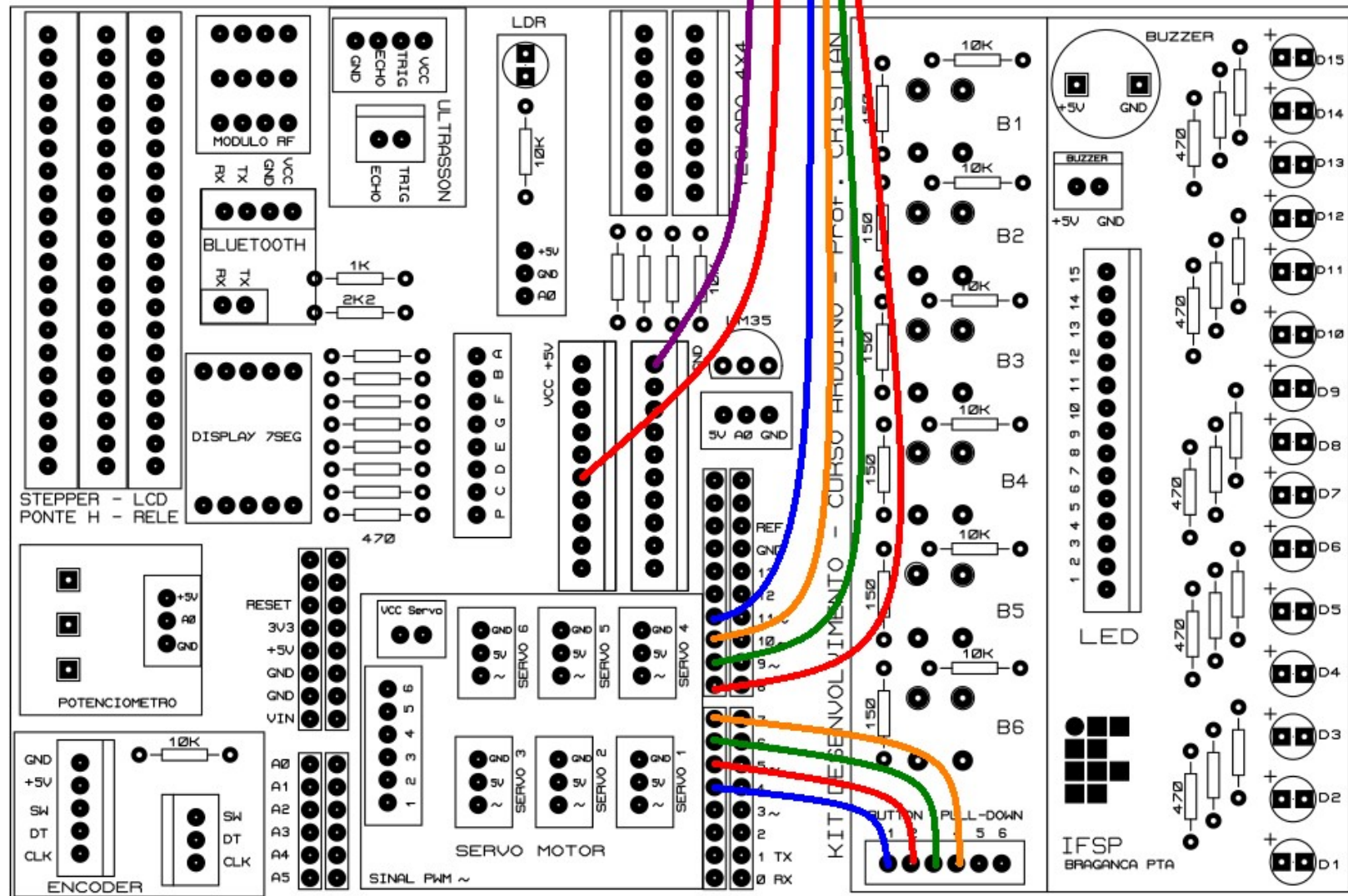


MOTOR A	IN1	IN2
HORÁRIO	5v	GND
ANTI-HORÁRIO	GND	5v
PONTO MORTO	GND	GND
FREIO	5v	5v

MOTOR B	IN3	IN4
HORÁRIO	5v	GND
ANTI-HORÁRIO	GND	5v
PONTO MORTO	GND	GND
FREIO	5v	5v

# Prática 26 – Acionamento Motores de Corrente Contínua via Ponte H





# Prática 26 – Acionamento Motores de Corrente Contínua via Ponte H

```
int IN1 = 8;
int IN2 = 9;
int IN3 = 10;
int IN4 = 11;
int botao_Frente = 7;
int botao_Re = 6;
int botao_Direita = 5;
int botao_Esquerda = 4;
```

```
void setup(){
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  pinMode(botao_Frente, INPUT);
  pinMode(botao_Re, INPUT);
  pinMode(botao_Direita, INPUT);
  pinMode(botao_Esquerda, INPUT);
}
```

```
void loop(){

  int Frente = digitalRead(botao_Frente);
  int Re = digitalRead(botao_Re);
  int Direita = digitalRead(botao_Direita);
  int Esquerda = digitalRead(botao_Esquerda);
```

```
  if (Frente == HIGH){
    digitalWrite(IN1, HIGH);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, HIGH);
    digitalWrite(IN4, LOW);
  }

  if (Re == HIGH){
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, HIGH);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, HIGH);
  }

  if (Direita == HIGH){
    digitalWrite(IN1, HIGH);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, LOW);
  }

  if (Esquerda == HIGH){
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, HIGH);
    digitalWrite(IN4, LOW);
  }

  if ((Frente && Re && Direita && Esquerda) == LOW){
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, LOW);
  }
}
```

# Prática 27 – Controle de Motores com Ponte-H via Bluetooth

Agora, vamos controlar a direção via bluetooth para movimentar o carro nas direções:

F = frente;

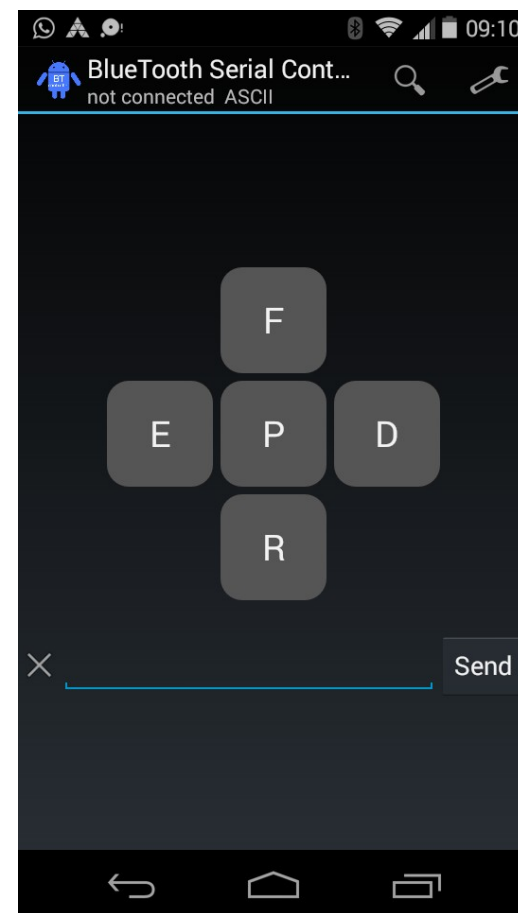
R = ré;

D = direita;

E = esquerda;

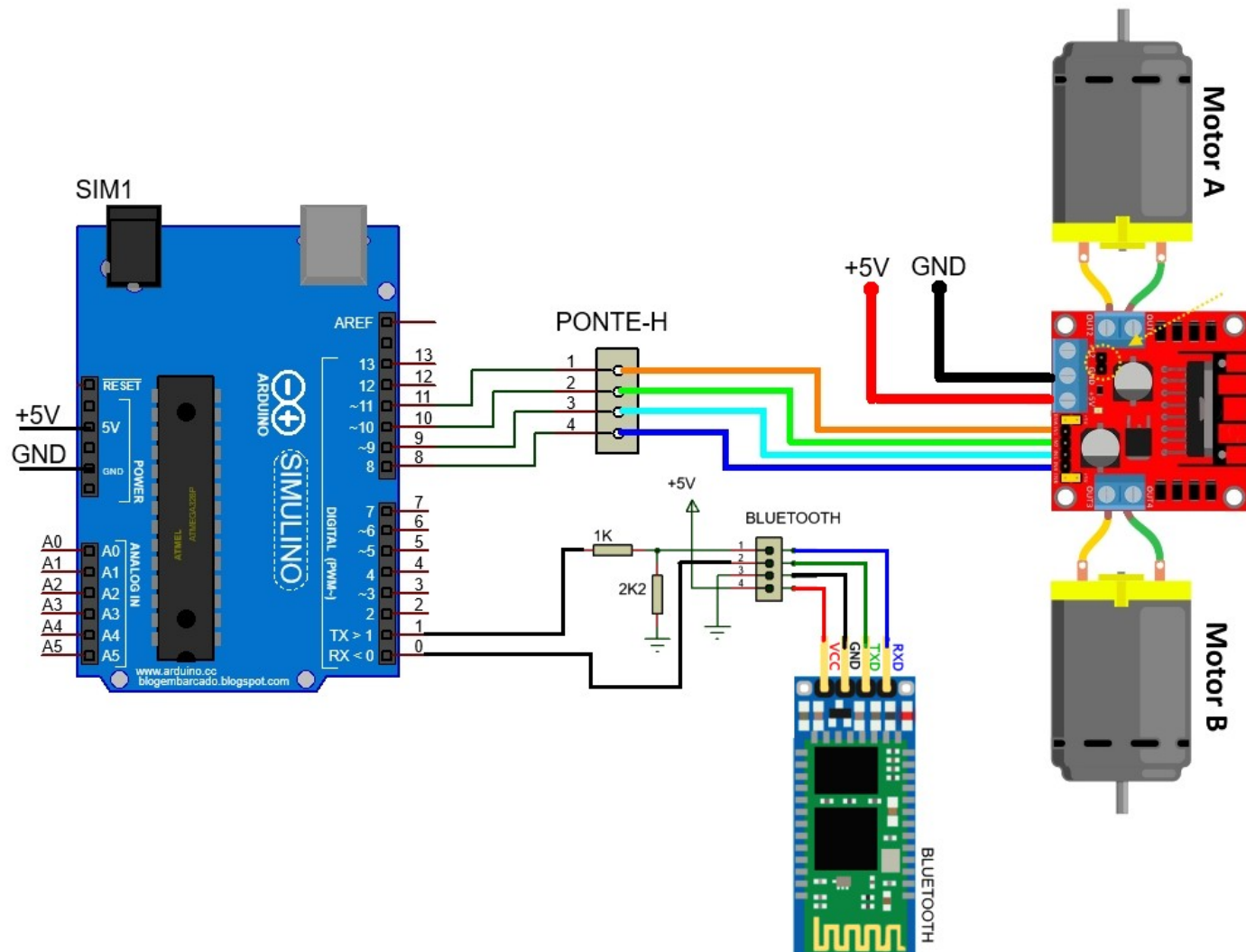
P = pare

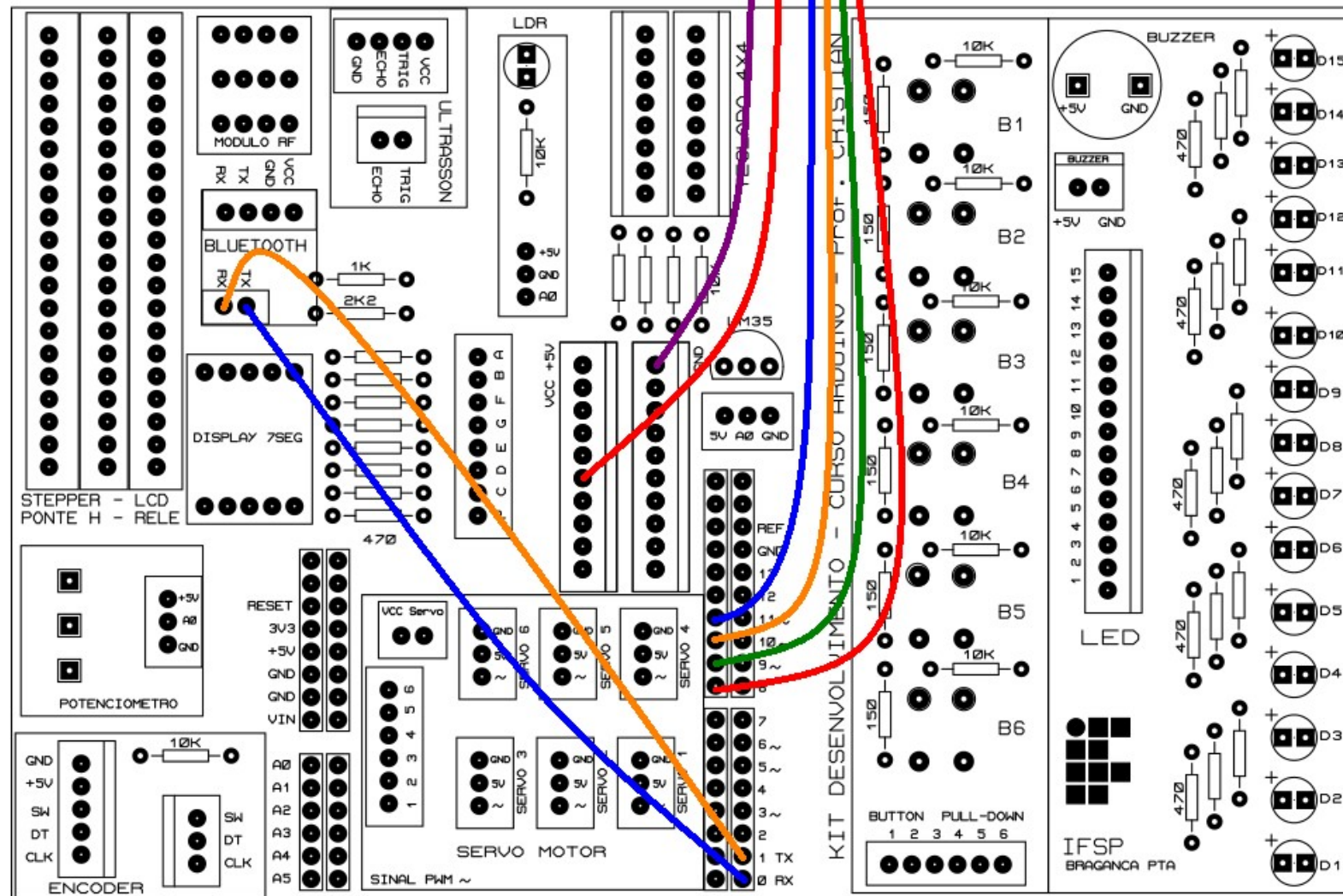
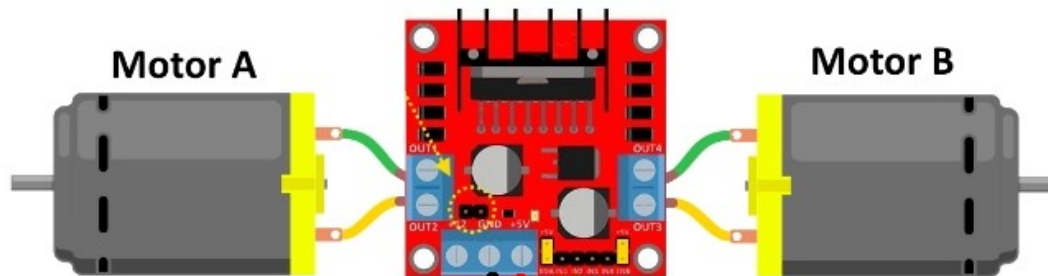
Primeiro realize os movimentos pelo Monitor Serial.





# Prática 27 – Controle de Motores com Ponte-H via BlueTooth





```

//Definicoes pinos Arduino ligados
//a entrada da Ponte H
int IN1 = 8;
int IN2 = 9;
int IN3 = 10;
int IN4 = 11;

char dados;

void setup()
{
    //Define os pinos como saida
    pinMode(IN1, OUTPUT);
    pinMode(IN2, OUTPUT);
    pinMode(IN3, OUTPUT);
    pinMode(IN4, OUTPUT);

    Serial.begin(9600);
}

void loop(){
    while(Serial.available() > 0) {
        dados = Serial.read();
        //Caso seja recebido o caracter F
        if (dados == 'F') {
            //Gira o Motor A e B no sentido horario
            digitalWrite(IN1, HIGH);
            digitalWrite(IN2, LOW);
            digitalWrite(IN3, HIGH);
            digitalWrite(IN4, LOW);
        }
    }
}

```

```

if (dados == 'R') {
    // o Motor A e B no sentido antihorario
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, HIGH);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, HIGH);
}

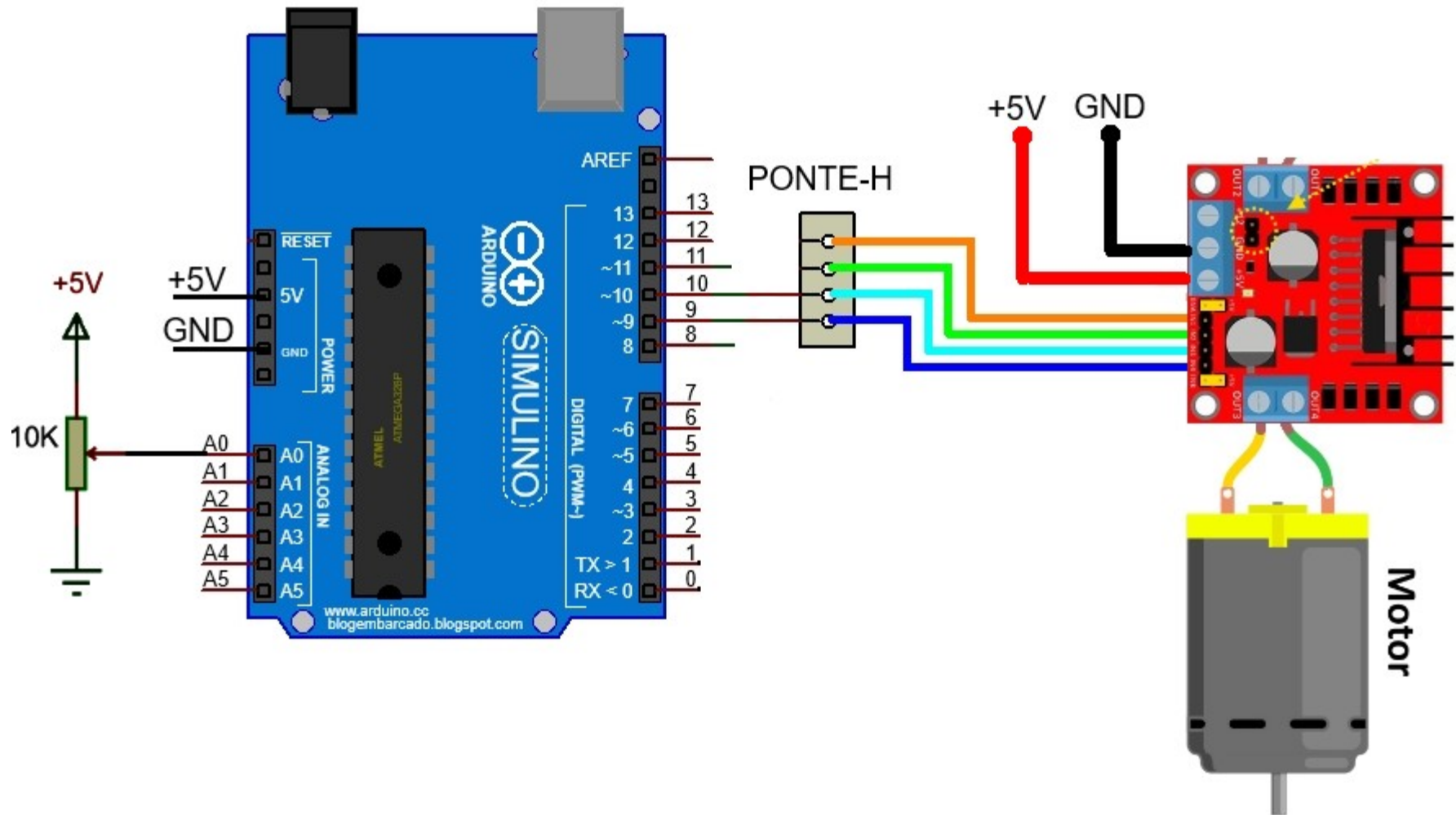
if (dados == 'D') {
    //Gira o Motor para direita
    digitalWrite(IN1, HIGH);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, LOW);
}

if (dados == 'E') {
    //Gira o Motor para esquerda
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, HIGH);
    digitalWrite(IN4, LOW);
}

if (dados == 'P') {
    //Para os Motor A e B
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, LOW);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, LOW);
}
}
}
}

```

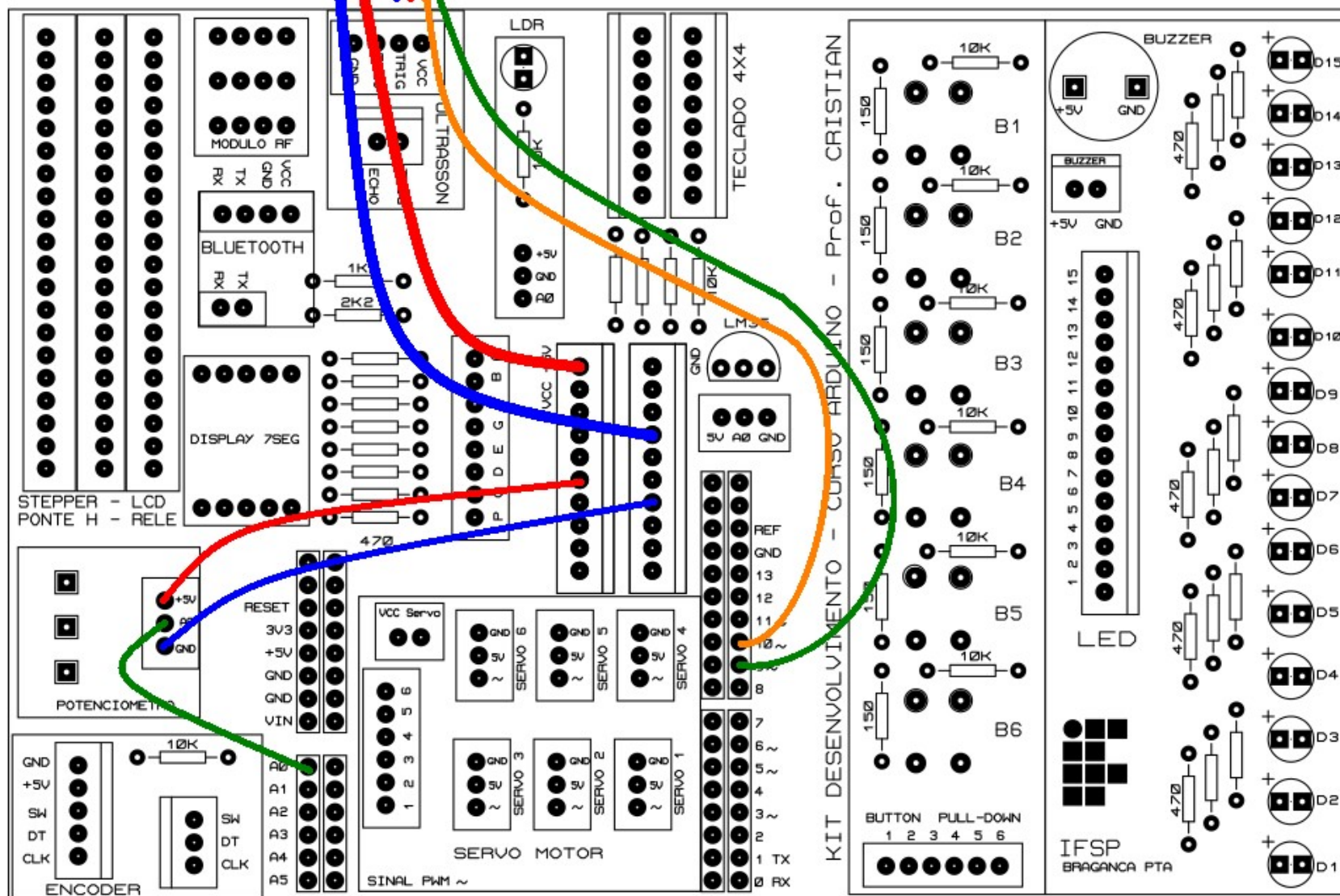
# Prática 28 – Controle de Velocidade de Motores via PWM







Motor



# Prática 28 – Controle de Velocidade de Motores via PWM

```
int IN1 = 10;
int IN2 = 9;

void setup(){
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  Serial.begin(9600);
}

void loop(){

  int pot = analogRead(A0);
  float pwm = map(pot,0,1023,0,255);

  analogWrite(IN1,pwm);
  //digitalWrite(IN1,HIGH);
  digitalWrite(IN2,LOW);

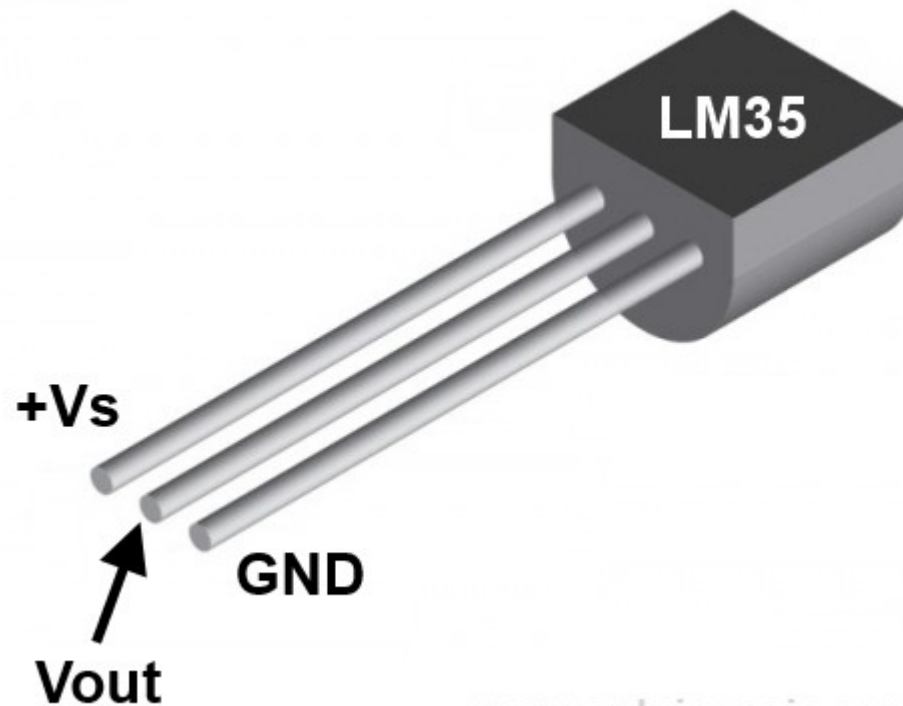
  Serial.print("Saida PWM = ");
  Serial.println(pwm);
  delay(500);
}
```

# Sensor de Temperatura - LM35



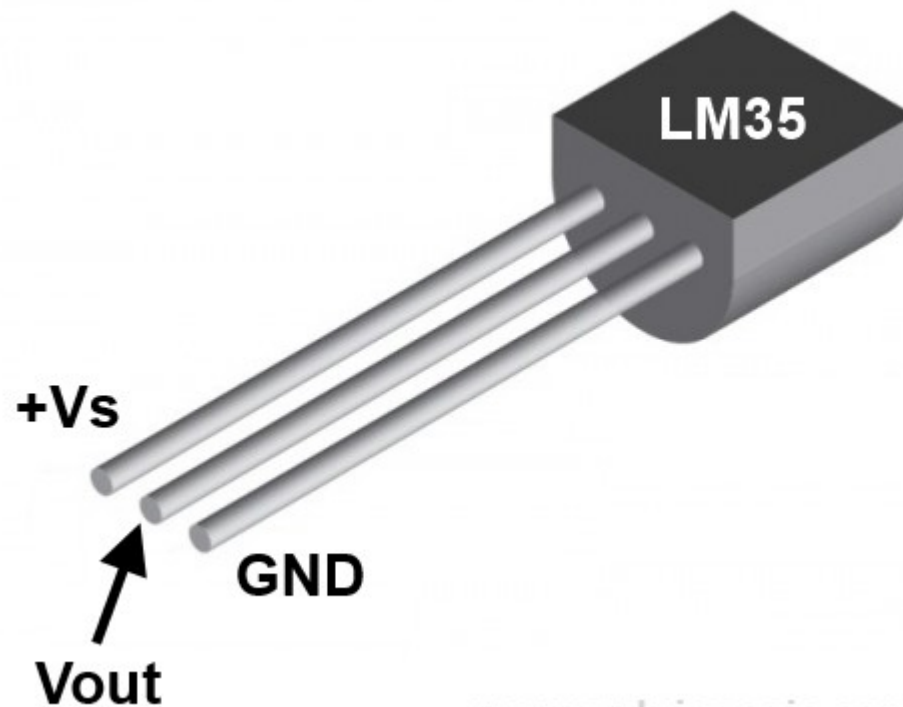
O LM35 é um sensor de temperatura que varia a tensão de saída (**pino Vout**) de acordo com a temperatura medida na superfície do sensor.

No Arduino, usamos uma das entradas analógicas para ler esta tensão do LM35 e calcular a temperatura.



# Sensor de Temperatura - LM35

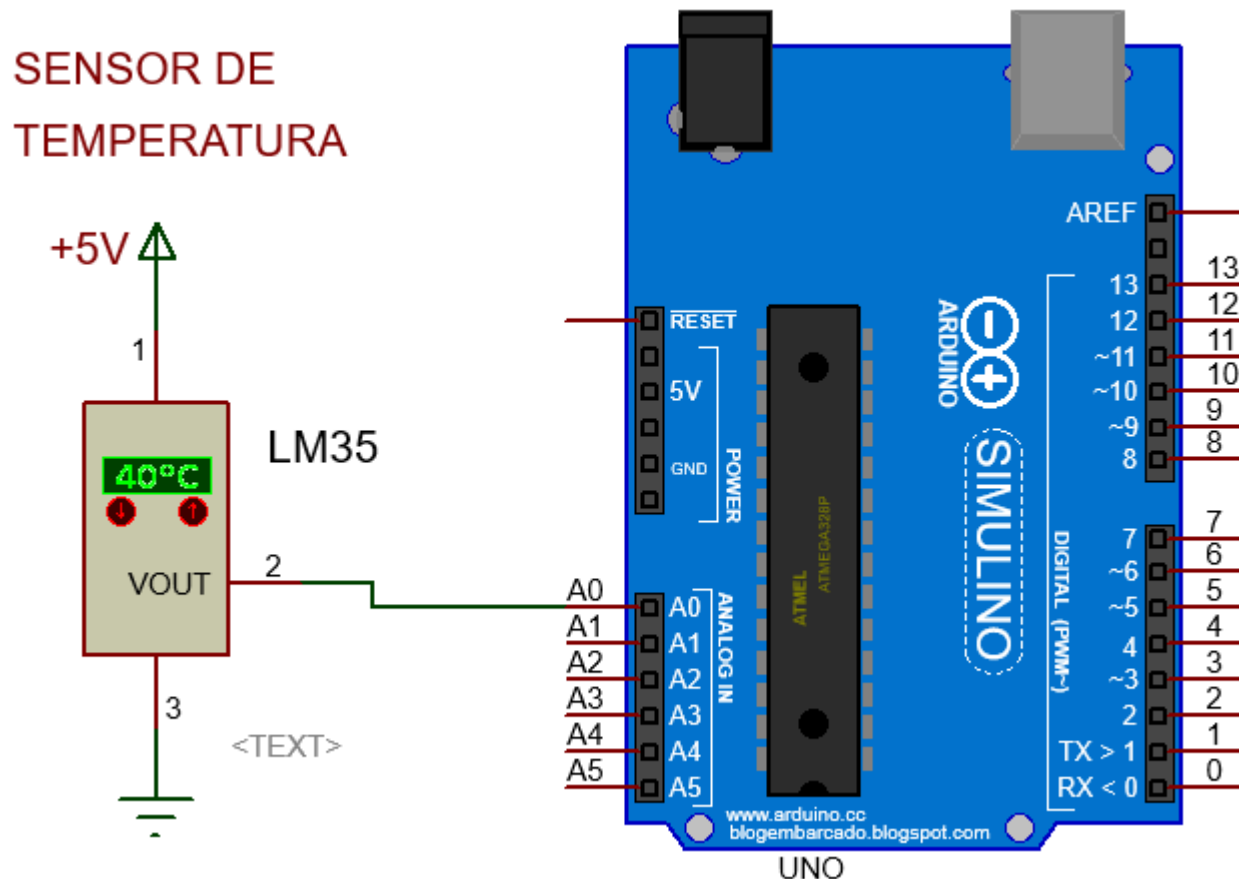
O LM35 tem um range de operação de  $-55^{\circ}\text{C}$  à  $150^{\circ}\text{C}$  com uma precisão de  $0,5^{\circ}\text{C}$  e possui um fator de escala linear de  $+10\text{mV}/^{\circ}\text{C}$  podendo trabalhar de  $4\text{V}$  à  $30\text{V}$ .





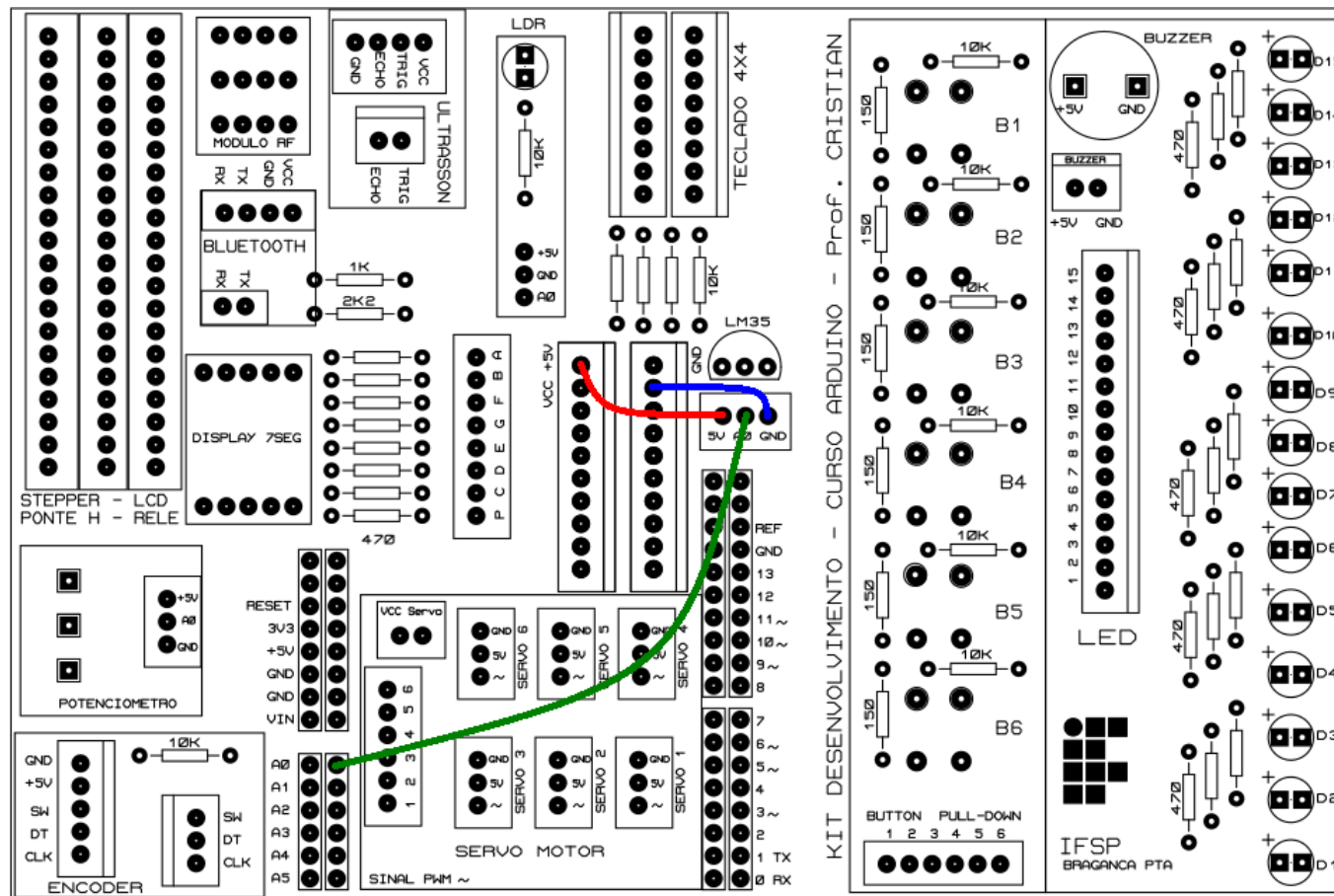
# Prática 29 - Sensor de Temperatura com LM35

Na placa didática do curso, já existe um sensor de temperatura LM35, ligue-o a entrada analógica A0.



# Prática 29 - Sensor de Temperatura com LM35

Na placa didática do curso, já existe um sensor de temperatura LM35, ligue-o a entrada analógica A0.



# Prática 29 - Sensor de Temperatura com LM35



```
void setup(){
  Serial.begin(9600); }

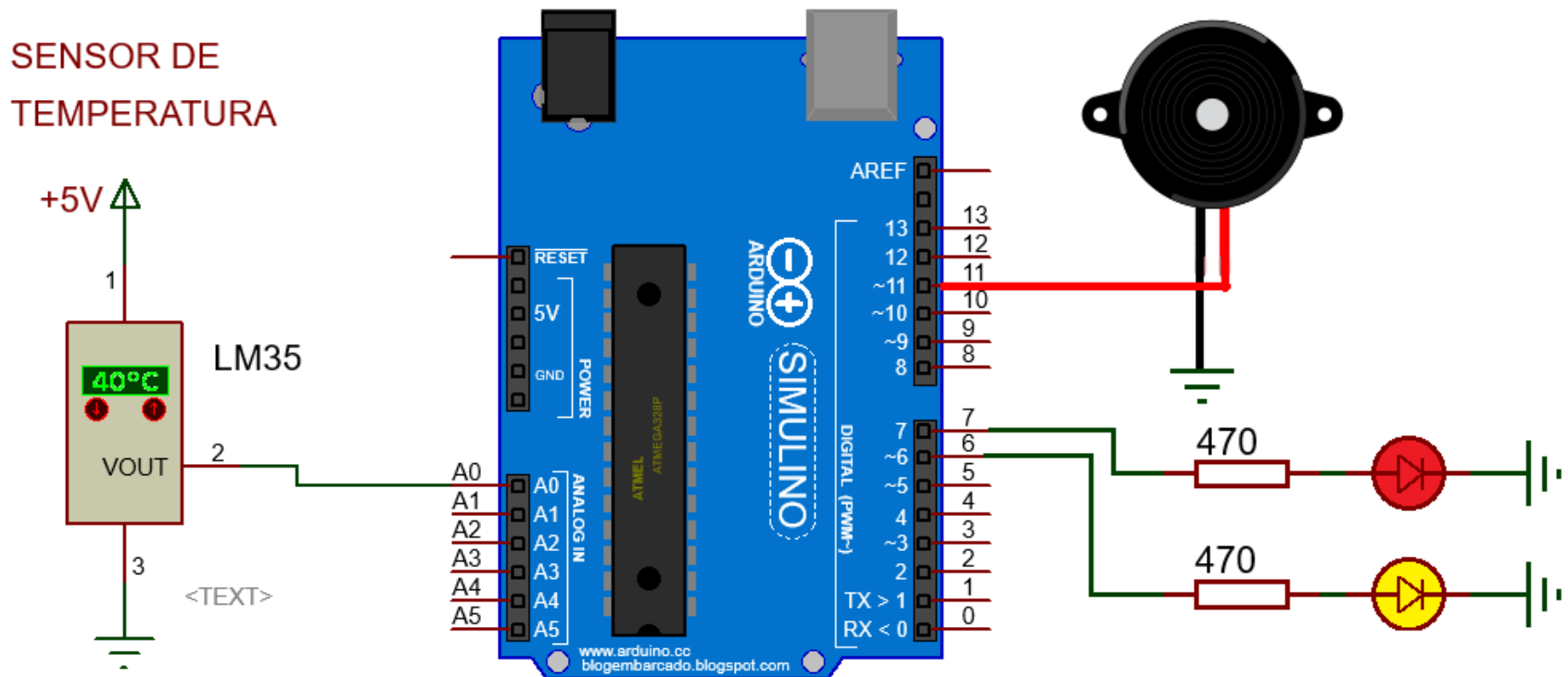
void loop(){
  //temp. do sensor lida em volts
  float sense=analogRead(A0);

  //valor em graus celsius
  float temp=((sense*5)/(1023))/0.01;

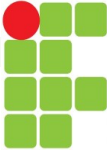
  Serial.print("Temperatura = ");
  Serial.print(temp);
  Serial.println(" Graus Celsius");
  delay(1000);
}
```

# Prática 30 - Sensor de Temperatura com Alerta de Alarme

Nesse programa o sensor mede a temperatura e aciona um alarme sonoro e pisca as luzes dos LEDs vermelho e amarelo, caso a temperatura ultrapasse 32°C.







# Prática 30 - Sensor de Temperatura com Alerta de Alarme

```
void setup(){
  Serial.begin(9600);
  pinMode(11,OUTPUT); //Pino do buzzer
  pinMode(7,OUTPUT);
  pinMode(6,OUTPUT);
}

void loop(){
  //temp. do sensor lida em volts
  float sense=analogRead(A0);

  //valor em graus celsius
  float temp=((sense*5)/(1023))/0.01;
```

```
  if (temp > 32){
    tone(11, 4000, 50);
    digitalWrite (7,HIGH);
    delay(200);
    tone(11, 2600, 50);
    digitalWrite (6,HIGH);
    digitalWrite (7,LOW);
    delay(200);
    digitalWrite (6,LOW);
  }

  else{
    delay(400);
  }

  Serial.print("Temperatura = ");
  Serial.print(temp);
  Serial.println(" Graus Celsius");
}
```