

# TIPE Projet contrôle d'un moteur 24V

Projet d'un contrôle en tension d'un moteur en 24V et contrôle de celui si avec un potentiomètre

- [Montage Arduino pour le contrôle de celui ci](#)
  - [Outil nécessaire au test et au montage](#)
  - [Phase de test et de compréhension du H-Bridge et des commande arduino](#)

Montage Arduino pour le  
contrôle de celui ci

Montage Arduino pour le contrôle de celui ci

# Outil nécessaire au test et au montage

Carte Arduino uno

Transformateur 220 V AC en 24V DC

Moteur 24V DC 8000 RPM (838 rad/s)

H-Bridge: HiLetgo BTS7960

par la suite pour le contrôle manuelle de la monté en régime: Potentiomètre

pour le contrôle de la rotation à droite ou à gauche: Bouton poussoir`

# Phase de test et de compréhension du H-Bridge et des commande arduino

Tout d'abord on a la H-bridge suivant HiLetgo BTS7960 monté avec deux puce BTS7960B dont la documentation est la suivante:

<https://content.instructables.com/FYD/V3VU/HUL8W2ZW/FYDV3VUHUL8W2ZW.pdf>

À partir de cette documentation et d'un premier code:

```
/*.....  
BTS7960 Motor Driver Test  
Written By : Mohannad Rawashdeh  
Code for :  
https://www.instructables.com/member/Mohannad+Rawashdeh/  
*/  
int RPWM=5;  
int LPWM=6;  
// timer 0  
int L_EN=7;  
int R_EN=8;  
  
void setup() {  
  // put your setup code here, to run once:  
  for(int i=5;i<9;i++){  
    pinMode(i,OUTPUT);  
  }  
  for(int i=5;i<9;i++){  
    digitalWrite(i,LOW);  
  }  
  delay(1000);  
  Serial.begin(9600);  
}
```

```
void loop() {
```

```

// put your main code here, to run repeatedly:
Serial.println("EN High");
digitalWrite(R_EN,HIGH);
digitalWrite(L_EN,HIGH);
delay(1000);
for(int i=0;i<256;i++){
    analogWrite(RPWM,i);
// analogWrite(LPWM,255-i);
    delay(100);
}
delay(500);
for(int i=255;i>0;i--){
    analogWrite(RPWM,i);
// analogWrite(LPWM,255-i);
    delay(100);
}
delay(500);
Serial.println("EN LOW");
digitalWrite(R_EN,LOW);
digitalWrite(L_EN,LOW);
delay(1000);
for(int i=0;i<256;i++){
    analogWrite(RPWM,i);
    delay(100);
}
delay(500);
for(int i=255;i>0;i--){
    analogWrite(RPWM,i);
    delay(100);
}
delay(500);
}

```

ce code fait tourner trois fois le moteur en montée progressive de régime puis baisse de celui ci .

Ainsi, je reconnais donc le contrôle des rotation du moteur

```

void loop() {
// put your main code here, to run repeatedly:
Serial.println("EN High");
digitalWrite(R_EN,HIGH);
digitalWrite(L_EN,HIGH);
analogWrite(RPWM,255);
}

```

ce code permet tout simplement de faire tourner le moteur dans une rotation anti-trigo à plein régime

```
void loop() {  
  // put your main code here, to run repeatedly:  
  Serial.println("EN High");  
  digitalWrite(R_EN,HIGH);  
  digitalWrite(L_EN,HIGH);  
  analogWrite(LPWM,255);  
}
```

Quant à celui ci, il fait la même chose dans le sens trigo  
bien sur en posant les variable suivante en fonction des entrées:

```
int RPWM=10;  
int LPWM=11;  
// timer 0  
int L_EN=7;  
int R_EN=8;
```