

Manuel d'utilisation du module MotoDriver1

Cher client,

Merci d'avoir acheté notre produit.

Veillez observer les instructions ci-dessous avant la première utilisation :



Utilisation avec un Arduino (ou compatible)

Connecter le module

Banchez le module aux broches de l'Arduino comme représenté ci-dessous :

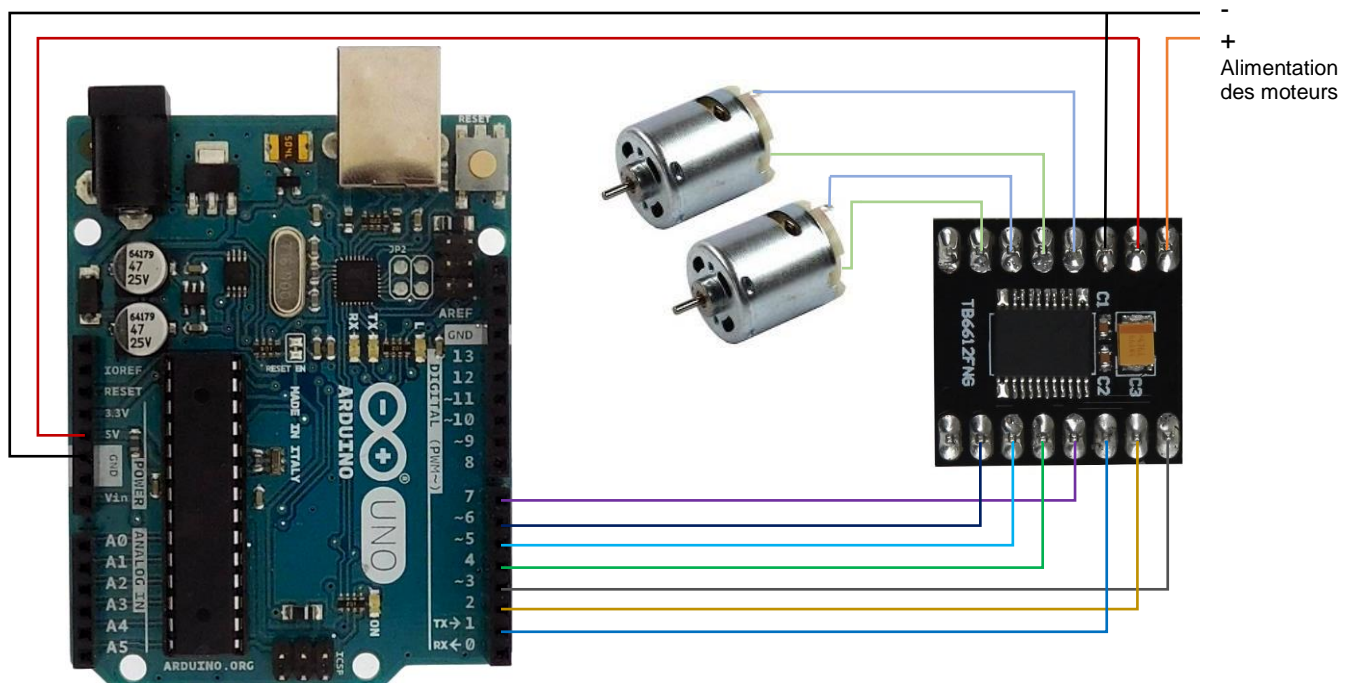
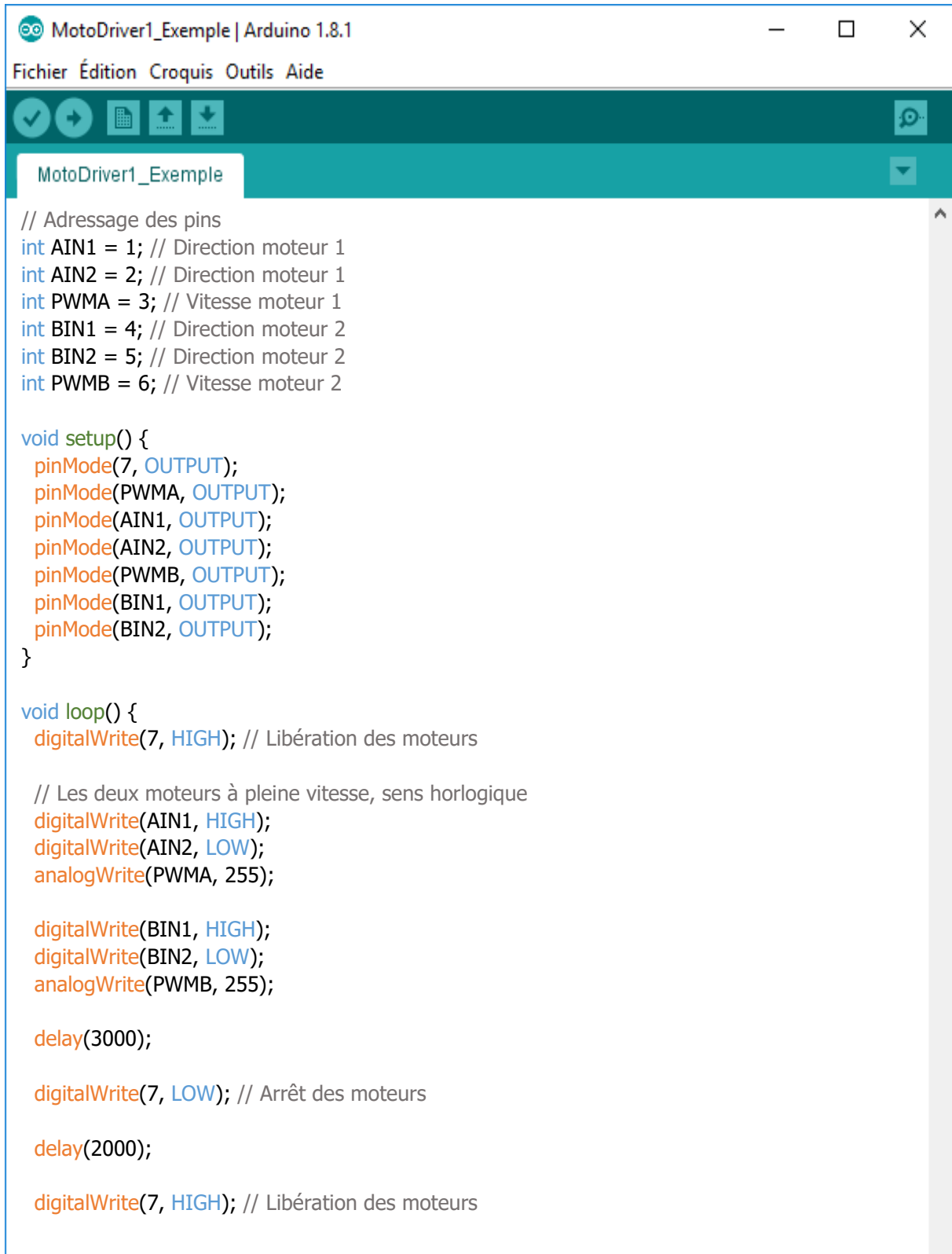


Table de correspondance :

Arduino	Module MotoDriver
1	AIN1
2	AIN2
3	PWMA
4	BIN1
5	BIN2
6	PWMB
7	STBY
5V	VCC
GND	GND
Moteur 1	AO1 et AO2
Moteur 2	BO1 et BO2

Exemple de programme

L'exemple de code suivant (à copier dans l'IDE Arduino) fait tourner les moteurs à pleine vitesse dans un sens pendant 3 secondes, les arrête pendant 2 secondes, les fait tourner à 50% de leur vitesse dans l'autre sens, les arrête pendant 2 secondes et recommence :



```

MotoDriver1_Exemple | Arduino 1.8.1
Fichier Édition Croquis Outils Aide

MotoDriver1_Exemple

// Adressage des pins
int AIN1 = 1; // Direction moteur 1
int AIN2 = 2; // Direction moteur 1
int PWMA = 3; // Vitesse moteur 1
int BIN1 = 4; // Direction moteur 2
int BIN2 = 5; // Direction moteur 2
int PWMB = 6; // Vitesse moteur 2

void setup() {
  pinMode(7, OUTPUT);
  pinMode(PWMA, OUTPUT);
  pinMode(AIN1, OUTPUT);
  pinMode(AIN2, OUTPUT);
  pinMode(PWMB, OUTPUT);
  pinMode(BIN1, OUTPUT);
  pinMode(BIN2, OUTPUT);
}

void loop() {
  digitalWrite(7, HIGH); // Libération des moteurs

  // Les deux moteurs à pleine vitesse, sens horlogique
  digitalWrite(AIN1, HIGH);
  digitalWrite(AIN2, LOW);
  analogWrite(PWMA, 255);

  digitalWrite(BIN1, HIGH);
  digitalWrite(BIN2, LOW);
  analogWrite(PWMB, 255);

  delay(3000);

  digitalWrite(7, LOW); // Arrêt des moteurs

  delay(2000);

  digitalWrite(7, HIGH); // Libération des moteurs
  
```

```
// Les deux moteurs à vitesse 1/2, sens anti-horlogique
digitalWrite(AIN1, LOW);
digitalWrite(AIN2, HIGH);
analogWrite(PWMA, 128);

digitalWrite(BIN1, LOW);
digitalWrite(BIN2, HIGH);
analogWrite(PWMB, 128);

delay(3000);

digitalWrite(7, LOW); // Arrêt des moteurs

delay(2000);
}
```

Utilisation avec un Raspberry Pi

Connecter le module

Branchez le module aux broches du Raspberry Pi comme représenté ci-dessous :

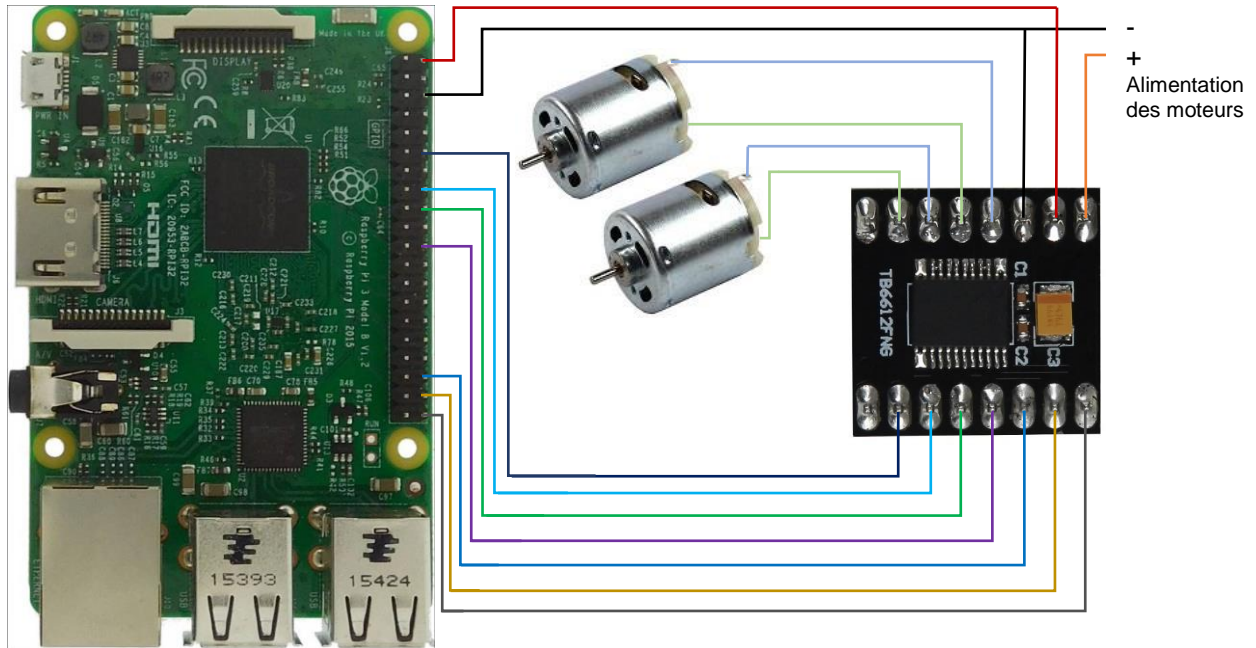


Table de correspondance :

Raspberry Pi	Module MotoDriver
GPIO 21	PWMA
GPIO 20	AIN2
GPIO 16	AIN1
GPIO 25	STBY
GPIO 24	BIN1
GPIO 23	BIN2
GPIO 18	PWMB
5V	VCC
GND	GND
Moteur 1	AO1 et AO2
Moteur 2	BO1 et BO2

Prérequis

Les explications et programmes qui suivent ont été créés et testés pour un Raspberry utilisant le système d'exploitation Raspbian.

Un guide d'installation est disponible ici :

<http://www.gotronic.fr/userfiles/www.gotronic.fr/files/Raspberry/GuideRPI.pdf>

Un guide de démarrage est disponible ici :

<http://www.gotronic.fr/userfiles/www.gotronic.fr/files/Raspberry/Raspberry%20Joyit.pdf>

Installation des bibliothèques

Pour installer les bibliothèques nécessaires, entrez les commandes ci-dessous dans LXTerminal :

Bibliothèque GPIO :

```
sudo apt-get install python-pip python-dev build-essential
sudo pip install RPi.GPIO
```

Bibliothèque Python :

```
sudo apt-get install python-imaging
```

Exemple de programme

Le programme suivant affiche à l'écran le numéro du bouton pressé.

Créez un nouveau fichier vierge *MotoDriver.py* en entrant la commande suivante :

```
sudo nano MotoDriver.py
```

Et copier les lignes suivantes dans ce fichier :

```
#!/user/bin/python
# coding=utf-8
import RPi.GPIO as GPIO
import time
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)

# Definition des variables
pwma=21
ain1=20
ain2=16
bin1=24
bin2=23
pwmb=18
stby=25

# Initialisation des broches
GPIO.setup(ain1, GPIO.OUT)
GPIO.setup(ain2, GPIO.OUT)
GPIO.setup(pwma, GPIO.OUT)
GPIO.setup(bin1, GPIO.OUT)
GPIO.setup(bin2, GPIO.OUT)
GPIO.setup(pwmb, GPIO.OUT)
GPIO.setup(stby, GPIO.OUT)

# Initialisation du PWM
PWMA = GPIO.PWM(pwma, 50)
PWMB = GPIO.PWM(pwmb, 50)
PWMA.start(0)
PWMB.start(0)
```

```
# Programme principal
print "Programme test MotoDriver1 [Appuyez sur Ctrl + C pour terminer le test]"

try:
    while True:
        GPIO.output(stby, GPIO.HIGH)
        print("Les deux moteurs à pleine vitesse, sens horlogique")
        GPIO.output(ain1, GPIO.HIGH)
        GPIO.output(ain2, GPIO.LOW)
        PWMA.ChangeDutyCycle(100)
        GPIO.output(bin1, GPIO.HIGH)
        GPIO.output(bin2, GPIO.LOW)
        PWMB.ChangeDutyCycle(100)
        time.sleep(3)

        print("Stop des moteurs")
        GPIO.output(stby, GPIO.LOW)
        time.sleep(2)

        GPIO.output(stby, GPIO.HIGH)
        print("Les deux moteurs à vitesse 1/2, sens anti-horlogique")
        GPIO.output(ain1, GPIO.LOW)
        GPIO.output(ain2, GPIO.HIGH)
        PWMA.ChangeDutyCycle(50)
        GPIO.output(bin1, GPIO.LOW)
        GPIO.output(bin2, GPIO.HIGH)
        PWMB.ChangeDutyCycle(50)
        time.sleep(3)

        print("Stop des moteurs")
        GPIO.output(stby, GPIO.LOW)
        time.sleep(2)

# remise en place de tous les GPIO
except KeyboardInterrupt:
    GPIO.cleanup()
```

Sauvegardez le fichier (*Ctrl+O et Enter*) et quittez le fichier (*Ctrl+X*).

Vous pouvez exécuter le programme :

```
sudo python MotoDriver.py
```

Ctrl+C pour arrêter le programme

GO TRONIC

ROBOTIQUE ET COMPOSANTS ÉLECTRONIQUES

Si vous rencontrez des problèmes, merci de nous contacter par courriel à :

sav@gotronic.fr



Coordonnées du fabricant :

JOY-IT®

service@joy-it.net

+49 (0)2845 9360 – 50