

ServoSix (PCB seulement)

La carte Servo Six sert à simplifier le processus de connexion de jusqu'à six servomoteurs à un Raspberry Pi ou un Arduino.

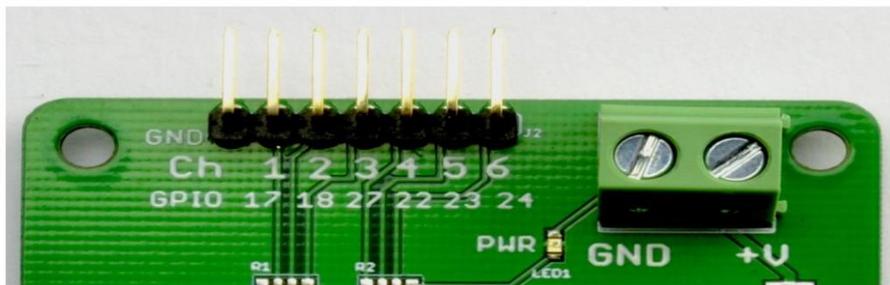
Caractéristiques :

- Bornes à vis pour l'alimentation électrique du servomoteur
- Protection contre l'inversion de polarité pour l'alimentation du servomoteur.
- Condensateur 470µF 16V pour l'alimentation du servo.
- Résistances de limitation de courant 470Ω pour les lignes de commande des servos (pour protéger les broches GPIO).
- Témoin lumineux d'alimentation

Connexion d'une carte Servo Six à un Raspberry Pi

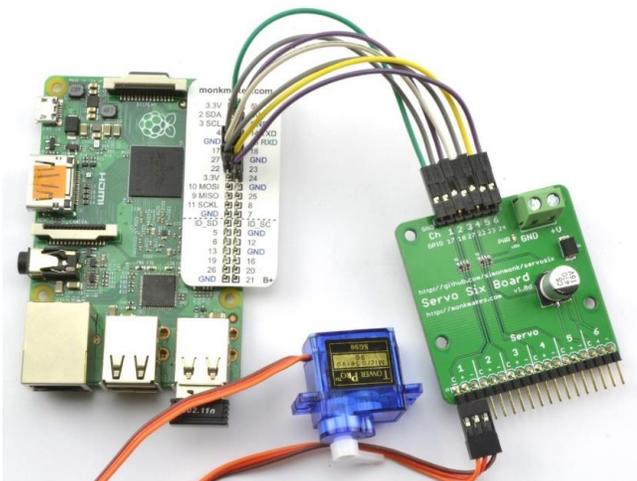
Pour connecter une carte ServoSix à un Raspberry Pi, vous avez besoin d'un fil de liaison femelle-femelle pour connecter la masse (GND) du Raspberry Pi à GND de la carte ServoSix. Pour chaque servo (jusqu'à six) que vous souhaitez contrôler, vous avez également besoin d'un fil de liaison femelle-femelle reliant la broche de contrôle de ce servo sur la carte ServoSix à l'une des broches GPIO du Raspberry Pi.

Pour vous aider, la carte ServoSix comporte une broche GPIO à utiliser avec une broche de commande particulière.



Ainsi, en regardant l'image ci-dessus, la broche la plus à gauche est GND et doit être connectée à une broche GND sur le Raspberry Pi. La broche suivante (broche de commande 1) doit être connectée à GPIO17 sur le Raspberry Pi, la broche de commande 2 sur le ServoSix à GPIO18 et ainsi de suite.

L'alimentation des moteurs doit être fournie séparément à l'aide de la borne à vis située à droite.



Logiciel (Raspberry Pi)

La manière la plus simple de contrôler des servos depuis Python est d'utiliser la bibliothèque officielle Raspberry gpio zero. Celle-ci fournit une classe très pratique appelée AngularServo.

Voici un exemple qui devrait vous aider à démarrer. Connectez un servo au canal 1 comme indiqué ci-dessus.

```
from gpiozero import AngularServo
from guizero import App, Slider

servo = AngularServo(17, min_pulse_width=0.5/1000, max_pulse_width=2.5/1000)

def slider_changed(angle) :
    servo.angle = int(angle)

app = App(title='Servo Angle', width=500, height=150)

slider = Slider(app, start=-90, end=90, command=slider_changed, width='fill', height=50)

slider.text_size = 30

app.display()
```

Lorsque vous exécutez le programme, une fenêtre s'ouvre avec un curseur. Déplacez le curseur vers la gauche et la droite et le servo devrait suivre.



Veuillez noter que nous ne maintenons plus activement la bibliothèque ServoSix.

Logiciel (Arduino)

Lorsque vous utilisez le Servo Six avec un Arduino, vous pouvez simplement utiliser la bibliothèque standard Arduino Servo.

La présentation suivante suppose qu'un servo est attaché au canal 1 et que le canal 1 est relié à la broche 2 de l'Arduino. Utilisez le moniteur série pour entrer un angle entre 0 et 180 degrés.

```
#include <Servo.h>

const int servoControlPin = 2 ;

Servo servo ;

void setup() {
    servo.attach(servoControlPin) ;
    Serial.begin(9600) ;
    Serial.println("Angle en degrés (0 à 180)" ) ;
}

void loop() {
    if (Serial.available()) {
        int angle = Serial.parseInt() ;
        servo.write(angle) ;
    }
}
```

