

Comment contrôler un servomoteur

Par Kilian Sanfins

Introduction

Un servomoteur est de manière général un moteur asservi en position, c'est à dire qu'un système mesure la position du moteur et le contrôle de telle sorte à ce qu'il atteigne la position demandée. Un vaste choix de dimensions et de puissances existe, mais nous allons nous concentrer sur le SG90 (ou sa version plus costaud le MG90). C'est en effet le servomoteur le plus abordable et celui communément trouvé au sein de modélisme ou de projet hobbyiste.

Son contrôle est d'une simplicité enfantine et c'est une solution rapide pour mettre en mouvement un mécanisme qui nécessite peu de couple et un positionnement précis dans une plage de 270°.

Avantage

- Facile d'utilisation
- Positionnement plutôt précis pour un système aussi compact et bon marché.
- Prix bas (en chine ou en volume on peut les trouver à 2€/pièce)

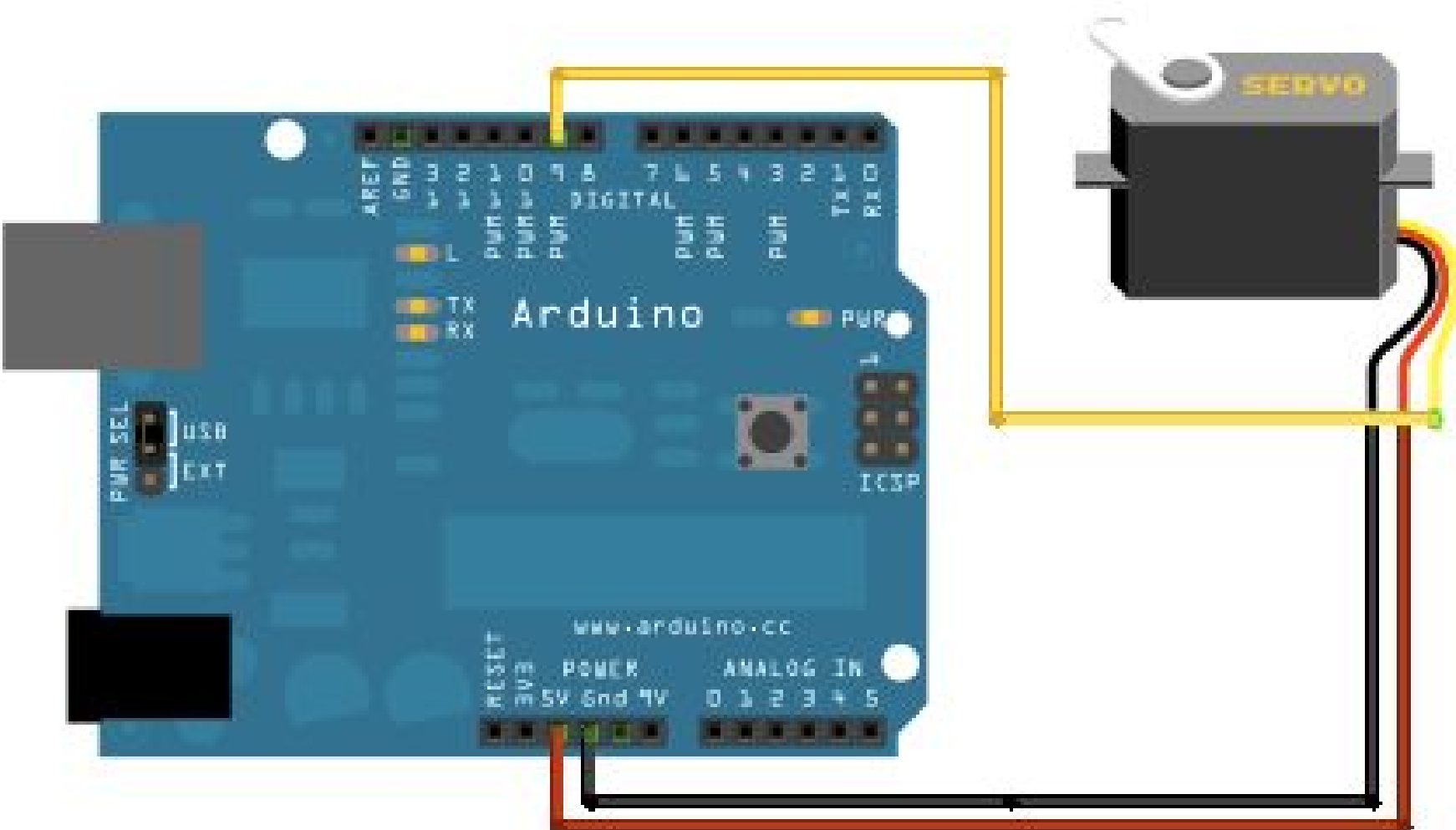
Désavantage

- Si l'on souhaite des servomoteurs plus puissant, le prix peut grimper très vite
- La course est généralement limité à ~270°. Il n'est donc pas possible de le faire tourner comme un moteur traditionnel. (Il existe bien sûr des servomoteurs 360°, mais le prix est différent. Attention, il existe aussi des servomoteurs asservi en vitesse, et non en position, ils font bien 360°, mais ils ne s'arrêteront pas forcément à la position désirée).

Cas d'utilisation

- Mettre en mouvement un système facilement et pour un prix dérisoir
- Le mouvement doit être précis et ne dépasse pas la plage de 0-270°
- Le couple requis est relativement faible (~2kg.cm max)

Schéma électrique



Code

```
/*
 * Ce programme a pour but de contrôler un servomoteur
 * en lui faisant parcourir un aller et retour en utilisant
 * la bibliothèque Servo.
 * Ecrit par BARRAGAN <http://barraganstudio.com>
 * et modifié par Scott Fitzgerald http://www.arduino.cc/en/Tutorial/Sweep
 * Commentaire ajouté par Kilian Sanfins
 */

// On ajoute la bibliothèque Servo
#include <Servo.h>

Servo myservo; // on crée un objet servo

void setup() {
  // On "attache" à notre objet servo le fil signal de notre servomoteur (pin 9 ici)
  myservo.attach(9);
}

void loop() {
  for (int pos = 0; pos <= 180; pos += 1) { // On incrémente la valeur de 0 à 180° en
pas de 1°
    myservo.write(pos);          // On envoie la commande au servomoteur
    delay(15);                   // On attend 15 ms
  }

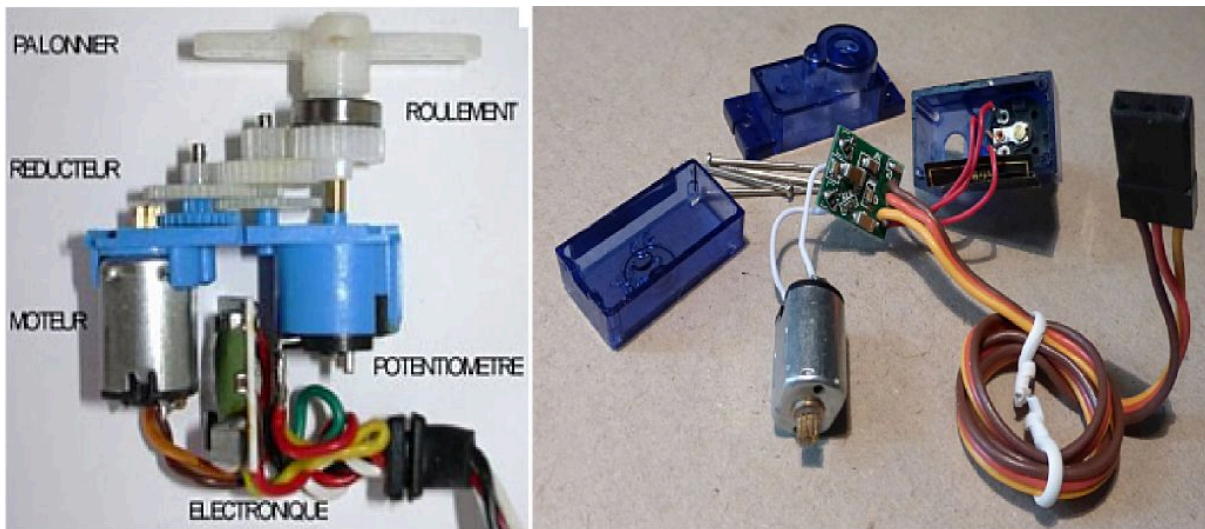
  // On réitère la même chose dans l'autre sens
  for (int pos = 180; pos >= 0; pos -= 1) { // On va de 180° à 0°
    myservo.write(pos);
    delay(15);
  }
}
```

Explication

Comment fonctionne un servomoteur?

Un servomoteur est constituée de 4 parties:

- un moteur à courant continu
- un réducteur
- un potentiomètre
- un circuit de contrôle



Le moteur est connecté au réducteur, le potentiomètre est attaché à l'axe de sortie et le circuit de contrôle alimente le moteur et lit la valeur du potentiomètre.

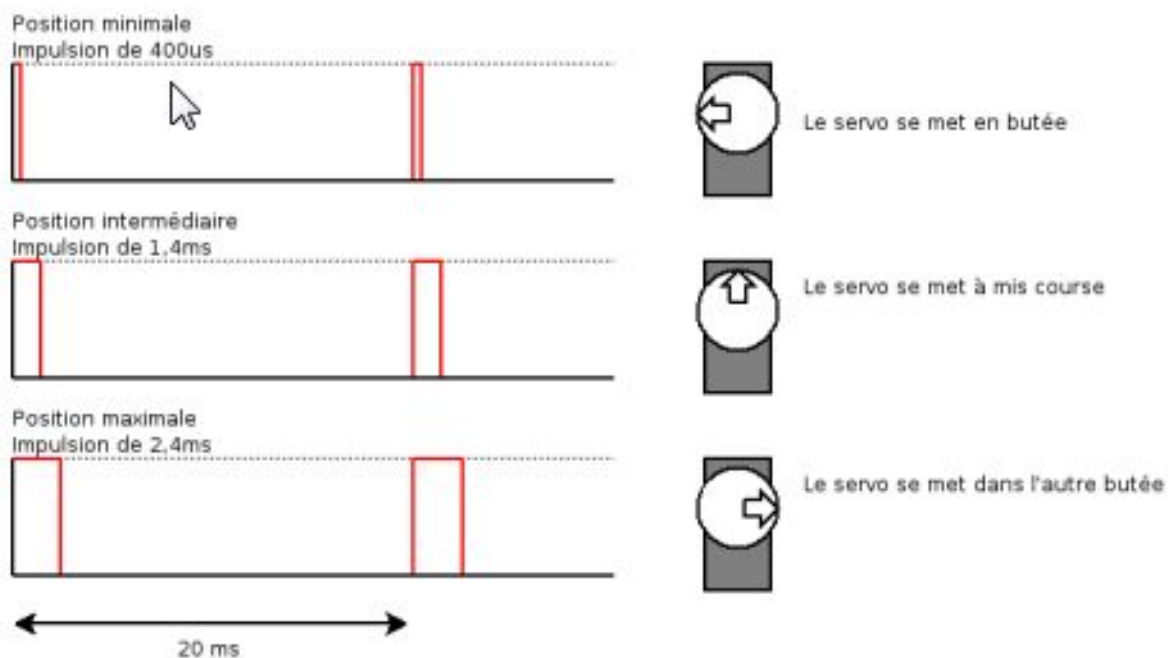
Le circuit de contrôle reçoit également des informations du microcontrôleur qui lui indique la position à laquelle doit se placer l'axe.

Le moteur est alors alimenté et au travers du réducteur, l'axe tourne ainsi que le potentiomètre. En lisant régulièrement la valeur du potentiomètre, le circuit peut moduler la tension au borne du moteur pour placer exactement l'axe à la position souhaitée.

Le protocole

Le protocole de contrôle d'un servomoteur est plutôt simple. Il consiste en une impulsion d'une durée variable toutes les 20 ms, la durée de l'état haut encode la position souhaitée du servomoteur.

Il n'y a pas de vraie norme sur ce protocole mais grâce au monde du modélisme, les constructeurs fabriquent des servomoteurs qui sont interchangeables (ceci est surtout vrai au niveau des hobbyistes).



Le schéma ci dessus est plutôt explicite. Cependant en pratique implémenter directement ce genre de protocole est assez inefficace, on préférera plutôt l'utilisation d'une bibliothèque comme "Servo", qui l'implémente en utilisant des mécanismes plus avancés de l'arduino.

Ou encore, il est possible d'utiliser un circuit comme le PCA9685 qui permet de contrôler jusqu'à 16 servomoteurs en parallèle (à condition que l'alimentation électrique soit suffisante!) et ceux sans utiliser de ressources ni de pins autre que ceux de la communication de la liaison i2c!

Le code utilisé pour contrôler cette puce est un peu hors de portée de ce tutoriel, mais je laisse dans les sources à approfondir un lien dirigeant vers Adafruit.com qui explique très bien en détail toute la procédure et le code!

Sources pour approfondir

- Site officiel Arduino, avec un exemple de servomoteur , en anglais
<https://www.arduino.cc/en/tutorial/sweep>
- Blog en français sur lequel j'ai récupéré les images, mais sans vrai information supplémentaire
<http://pierrecaulet.free.fr/Projets/Servomoteur/servomoteur.html>
- Adafruit.com est un revendeur de composants electronique qui fournit des tutoriels pour certaines cartes, très bien détaillé, en anglais
<https://learn.adafruit.com/16-channel-pwm-servo-driver?view=all>