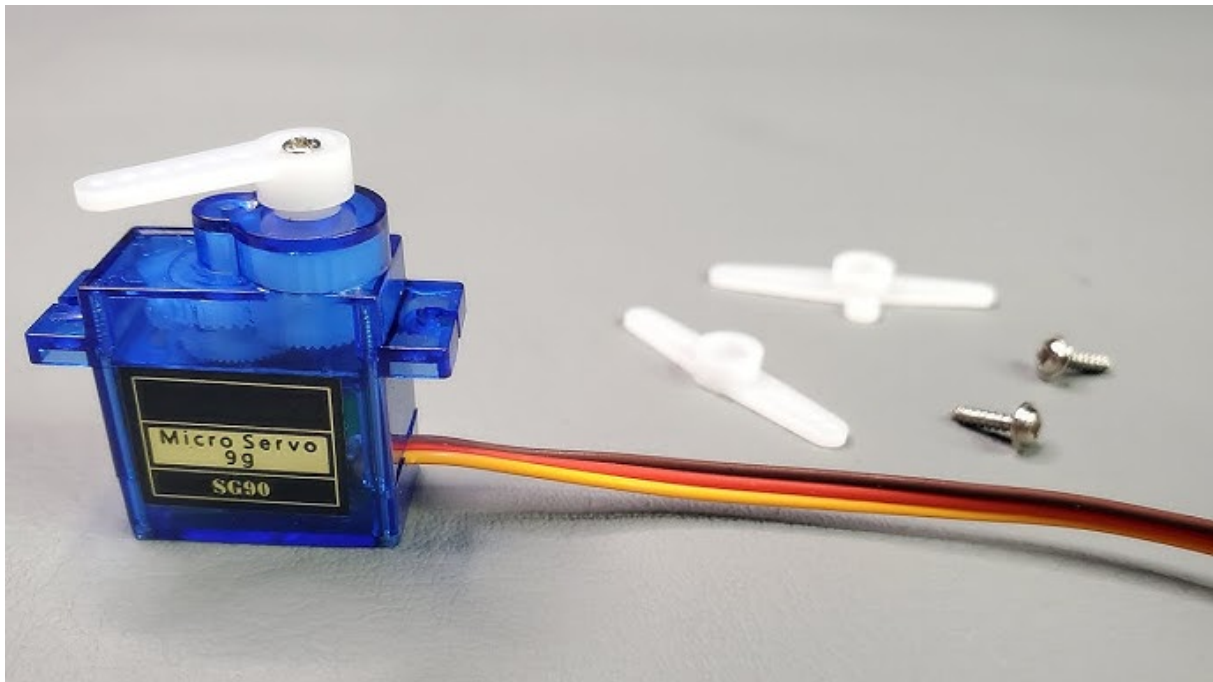


## Atelier: Introduction au Servo-Moteur avec l'Arduino



**Figure 1:** Moteur Servo

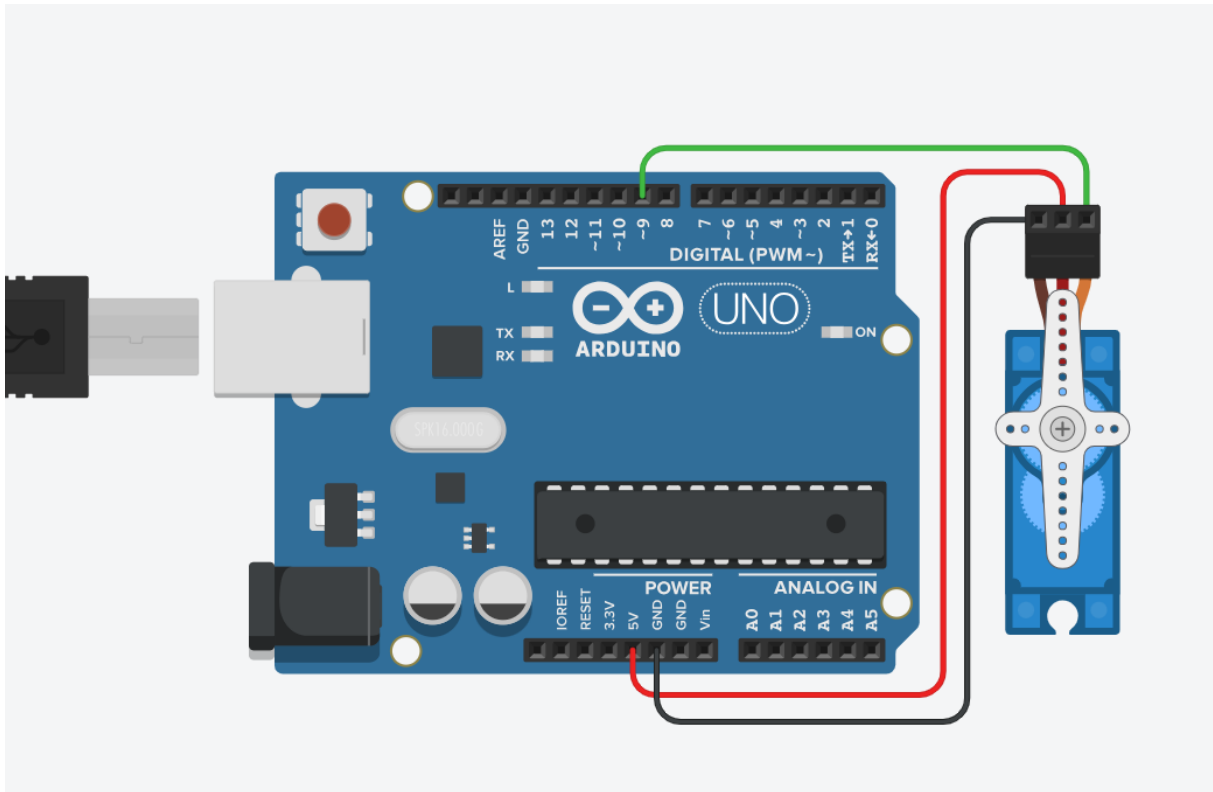
Dans cet atelier, nous allons connecter et programmer un servo-moteur pour qu'il fonctionne comme un métronome.

Nous verrons comment contrôler la vitesse de balancement du moteur avec une variable et comment adapter le système à différents besoins.

### Matériaux Nécessaires

- 1x carte Arduino Uno
- 1x Servo-moteur (SG90)
- Fils de connexion
- Câble USB pour Arduino

## Partie 1 : Configurer le Circuit



**Figure 2:** Configuration du Circuit

- **Fil Rouge (Alimentation)** : Connectez-le à la broche 5V de l'Arduino.
- **Fil Noir ou Marron (Masse)** : Connectez-le à une broche GND de l'Arduino.
- **Fil Jaune ou Orange (Signal)** : Connectez-le à la broche digitale 9 de l'Arduino.

## Partie 2 : Programmer l'Arduino

Ouvrez l'IDE Arduino. Nous utiliserons un sketch simple pour contrôler le mouvement du servo.

### Étape 1: Inclure la Bibliothèque Servo

Ajoutez cette ligne au début de votre code pour utiliser la bibliothèque Servo :

```
1 #include <Servo.h>
```

### Étape 2: Créer un Objet Servo

Créez un objet `myServo` pour contrôler le servo-moteur :

```
1 Servo myServo;
```

### Étape 3: Configurer le Servo

Dans la fonction `setup()`, attachez le servo à une broche spécifique de l'Arduino :

```
1 void setup() {  
2     myServo.attach(9); // Attache le servo à la broche 9  
3 }
```

### Étape 4: Boucle de Mouvement du Servo

Utilisez la fonction `loop()` pour déplacer le servo entre deux positions :

```
1 void loop() {  
2     myServo.write(0); // Déplacer le servo à 0 degrés  
3     delay(1000);      // Attendre 1 seconde  
4     myServo.write(180); // Déplacer le servo à 180 degrés  
5     delay(1000);      // Attendre 1 seconde  
6 }
```

Puisque le `loop()` est exécuté en boucle, le servo se déplacera de 0 à 180 degrés et reviendra à 0 degrés toutes les 1000 millisecondes.

### Étape 5: Créer une Variable pour changer le Délai

Créez une variable `delayTime` pour contrôler la vitesse du métronome :

```
1 int delayTime = 1000;
```

Modifiez la fonction `loop()` pour utiliser cette variable :

```
1 void loop() {  
2   myServo.write(0);  
3   delay(delayTime); // Utiliser la variable delayTime  
4   myServo.write(180);  
5   delay(delayTime); // Utiliser la variable delayTime  
6 }
```

Changer la valeur de `delayTime` permet de contrôler la vitesse du métronome sans modifier le reste du code.

### Partie 3: Questions d'Exploration

1. Modifiez la valeur de `delayTime`. Comment cela affecte-t-il le métronome ?
2. Comment modifier le code pour lire la valeur du potentiomètre et l'utiliser pour contrôler `delayTime` ?
3. Au lieu que notre servomoteur agisse comme un métronome, comment pourriez-vous utiliser un potentiomètre pour contrôler la position du servo entre 0 et 180 degrés ?