



# Le Cerveau - Arduino capacitif

Une sculpture animée grâce à un Arduino en mode capacitif et un moteur pas à pas

 Difficulty **Medium**

 Duration **6 hour(s)**

 Categories **Art, Decoration, Electronics, Robotics**

 Cost **150 EUR (€)**

## Contents

Introduction

Video overview

Step 1 - Fabrication du capteur capacitif

Step 2 - Bibliothèque Capacitive Sensor

Step 3 - Câblage et réglage du driver TB6600

Step 4 - Quelques explications du code Arduino


Comments

## Introduction

Dans le cadre d'une résidence d'artiste, le Fablab de Saint Laurent du Maroni a été amené à collaborer avec un artiste camerounais Dieudonné Fokou afin d'animer certaines de ses sculptures. Cette oeuvre baptisée "le Cerveau" utilise la technique de l'arduino capacitif et un moteur pas à pas. Nous détaillerons ici que le code et le câblage électronique, le reste relevant plus de la soudure à l'arc et de l'artiste.

## Materials

## Tools

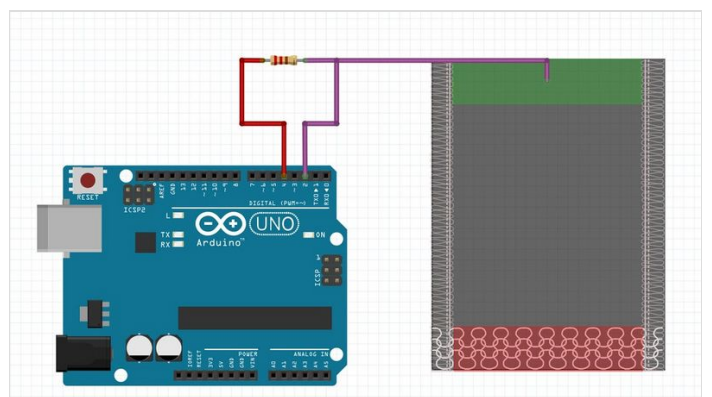
 Le\_Cerveau\_-\_Arduino\_capacitif\_Thebrain\_vf.ino

---

## Step 1 - Fabrication du capteur capacitif

Lorsqu'on place notre main sur ou près de la structure métallique, une partie du crâne va s'ouvrir révélant son contenu. La sculpture fait office de capteur capacitif, elle envoie un signal à notre **Arduino** dès qu'on la touche par comparaison de temps entre l'impulsion qu'elle reçoit sur une entrée réceptrice et l'impulsion de sortie qu'elle envoie sur une autre.

Une **résistance de 10 MOhm** (au minimum 1 MOhm) sera nécessaire pour obtenir cet effet et sera placée entre 2 pins de l'Arduino. Un **moteur pas à pas** de type **Nema 17** activera l'ouverture et la fermeture du crâne et sera piloté par un très bon **driver TB6600**.



---

## Step 2 - Bibliothèque Capacitive Sensor

Pour transformer l'Arduino en capteur capacitif, il faut tout d'abord installer la bibliothèque de détection capacitive `CapacitiveSensor` depuis le logiciel Arduino. Extraire les fichiers dans `Documents/Arduino/libraries` puis redémarrer l'Arduino.

Pour le contrôle du moteur pas à pas, il est également possible de télécharger une librairie mais on économise de la mémoire en restant simple.

---

# Step 3 - Câblage et réglage du driver TB6600

On insère une résistance de 10 MOhm entre la pin 4 et 2 de l'Arduino.

La pin 2 nous sert de réception à laquelle est connectée également la structure métallique et la pin 4 de sortie du signal.

Selon le seuil atteint par la mesure, on va activer notre moteur avec le pin 9 (ENA+) et lui donner une direction sur le pin 7 (DIR+).

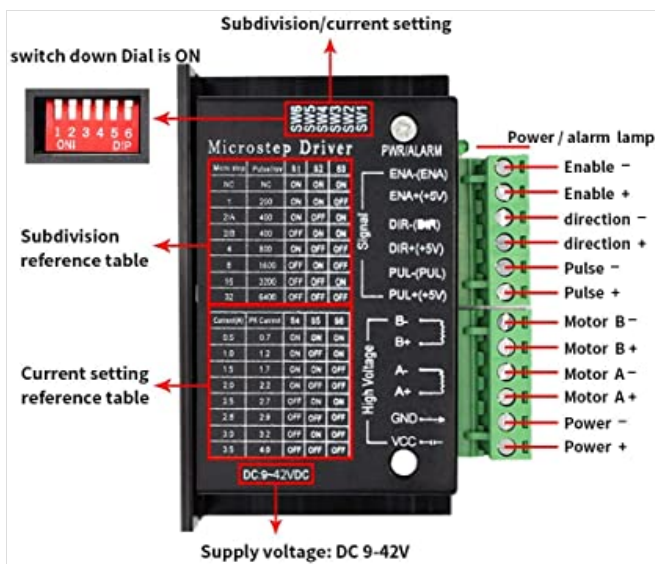
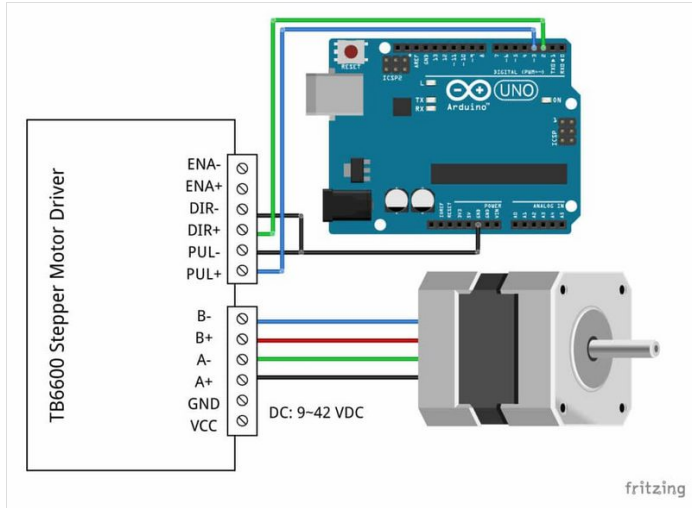
Le pin 8 (PUL+) nous sert à lui indiquer le nombre de pas.

Sur le TB6600, il y a 6 micro switches notés de SW1 à SW6. Les 3 premiers SW1 à SW3 ont été laissés à OFF, ce qui permet d'avoir un réglage à 32 micro steps, il faudra jouer avec la vitesse pour obtenir un mouvement fluide.

SW4 et SW5 ont été mis à ON mais pas SW6 pour limiter le courant à 1.5A (0.4A/phase soit  $4 \times 0.4 = 1.6A$ ).

On relie également les 4 fils de notre moteur 17HS15-0404S à notre driver. Comme il y a 2 bobinages, les fils vont par paire (faire le test des fils croisés): noir(A+)/vert(A-) et rouge(B+)/bleu(B-).

Et on relie le (DIR-)/(PUL-)/(ENA-) au GND de l'Arduino afin de leur définir un état bas (il est généralement déconseillé de laisser des pattes en l'air).



## Step 4 - Quelques explications du code Arduino

Comment fonctionne le programme, on définit les pins entre lesquels on récupère l'information avec « `CapacitiveSensor cs_4_2 = CapacitiveSensor(4,2)` » : la pin 2 étant notre capteur relié à la structure.

Dans la partie `Setup()`, on définit juste nos sorties digitales avec output pour le DIR, le ENA et le STEP du moteur. Et on initialise la liaison série avec `Serial.begin()`.

Dans la boucle `Loop()` du programme, on active l'ouverture ou la fermeture du cerveau selon la valeur de trigger récupérée après l'exécution de `CSread()`.

---