

Casagrande - Boite de cisaillement

Informations :

- Tenue du wiki : Pierre Thery & Johane Dufour
- Contact : johane.dufour@etu.sorbonne-universite.fr
- Etudes : Licence 3 - Sciences de la Terre
- Date du projet : Juin 2024
- Encadrant : Pierre Thery

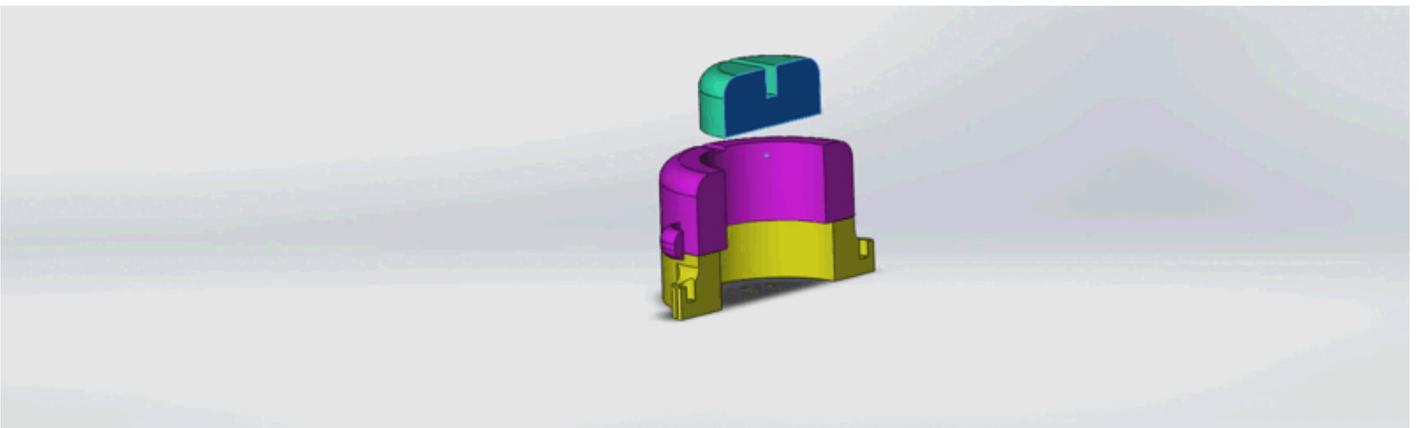
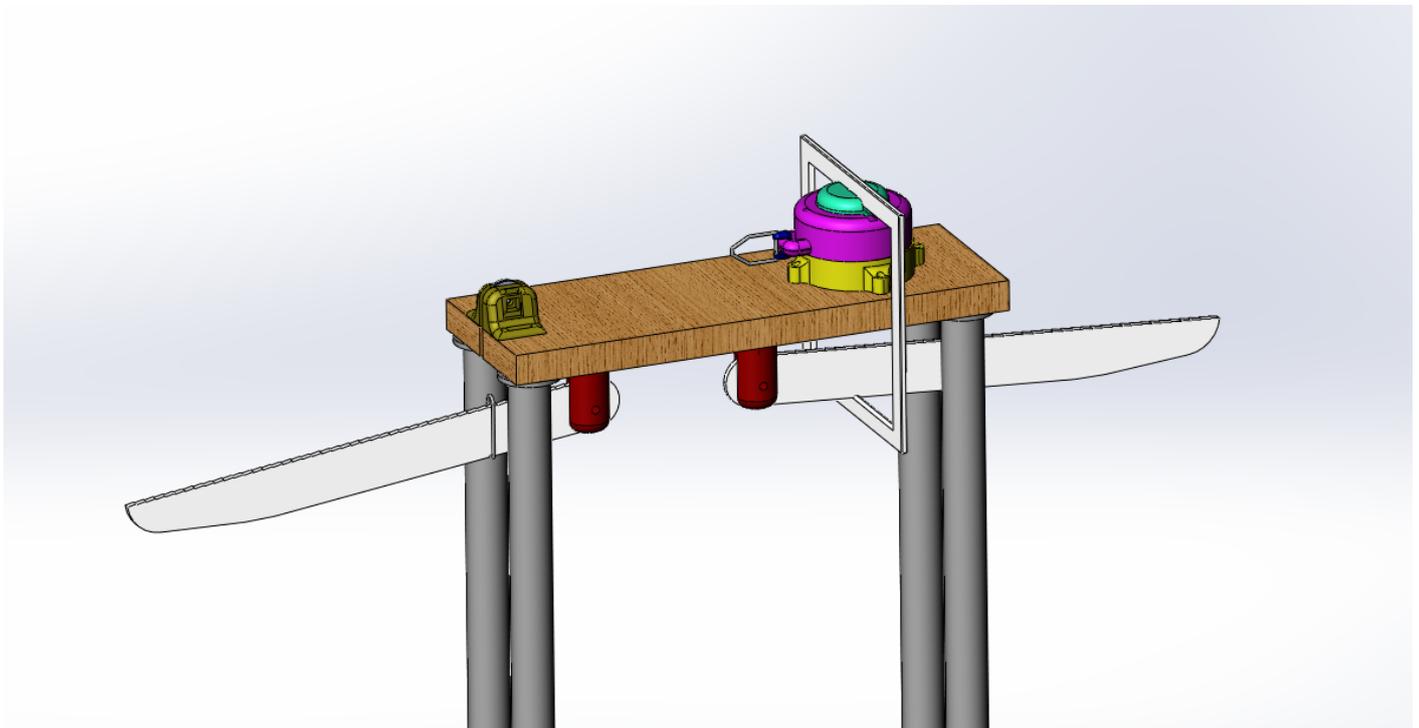
Sommaire :

1. Introduction et objectif du projet
2. Matériel, machines et logiciels utilisés
3. Conception de la boîte de Casagrande + Codes OpenSCAD
4. Journal de Bord

1. Introduction et objectif du projet

L'objectif du projet est de reprendre la boîte de Casagrande utilisée les travaux pratiques de géotechnique/géophysique niveau L3. Elle permet de mesurer la résistance au cisaillement (cohésion C et angle de frottement interne Φ) et notamment pour un sable humide.

On reprend le projet précédent : [wiki:projet:casagrande](https://wiki.fablab-sorbonne.fr/projet:casagrande) [FablabSU]



Suite aux multiples travaux pratiqués avec la précédente version, des points d'améliorations sont à prévoir:

- Création de goupilles pour le remplissage de la boîte, en effet le bras de levier entraînait le cisaillement avant même la manipulation
- Prévoir l'installation sur pied car il est plus pratique de manipuler à hauteur
- Prévoir un butoir car une fois la rupture atteinte, le bras de levier de la force tangentielle descend jusqu'au sol
- Solidifier la structure car elle n'était pas prévue pour une utilisation intense par des étudiants (plusieurs pièces cassés)
- A voir: inclure un dynamomètre pour le calcul de la force tangentielle ou d'un montage Arduino + écran + capteurs FSR400 + $R=10k\Omega$

--><https://www.instructables.com/Arduino-pressure-sensor-FSR-with-LCD-display/>

2. Matériel, machines et logiciels utilisés

> Matériel nécessaire :

Capteurs FSR400	Carte Arduino Uno	Câbles Arduino	Module encodeur rotatif GT055	Planche de plexiglas de 6mm d'épaisseur	Écran LED
				fablab	

> Machines : Trotec Speedy 360 (découpeuse laser), Imprimante 3D, Perceuse

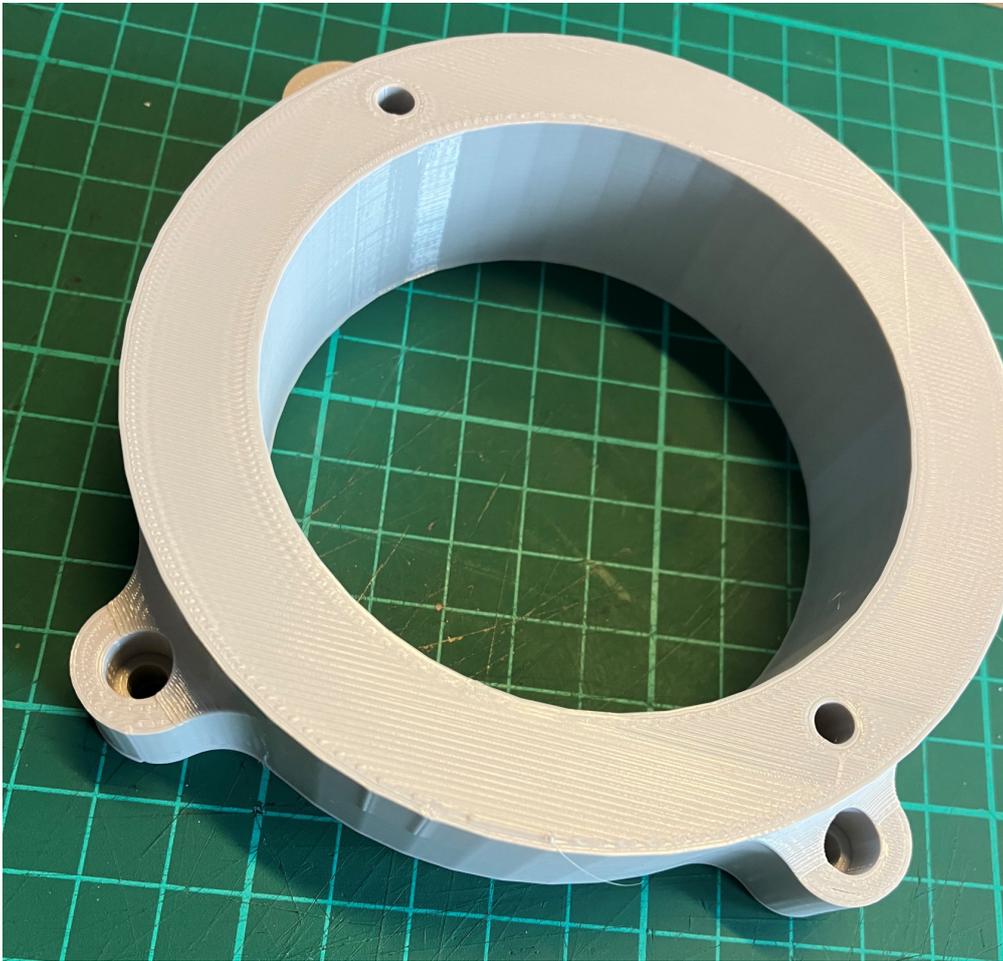
> Logiciels : OpenSCAD, Inkscape, Arduino IDE

3. Conception de la boîte de Casagrande + Codes OpenSCAD

> Impressions 3D :

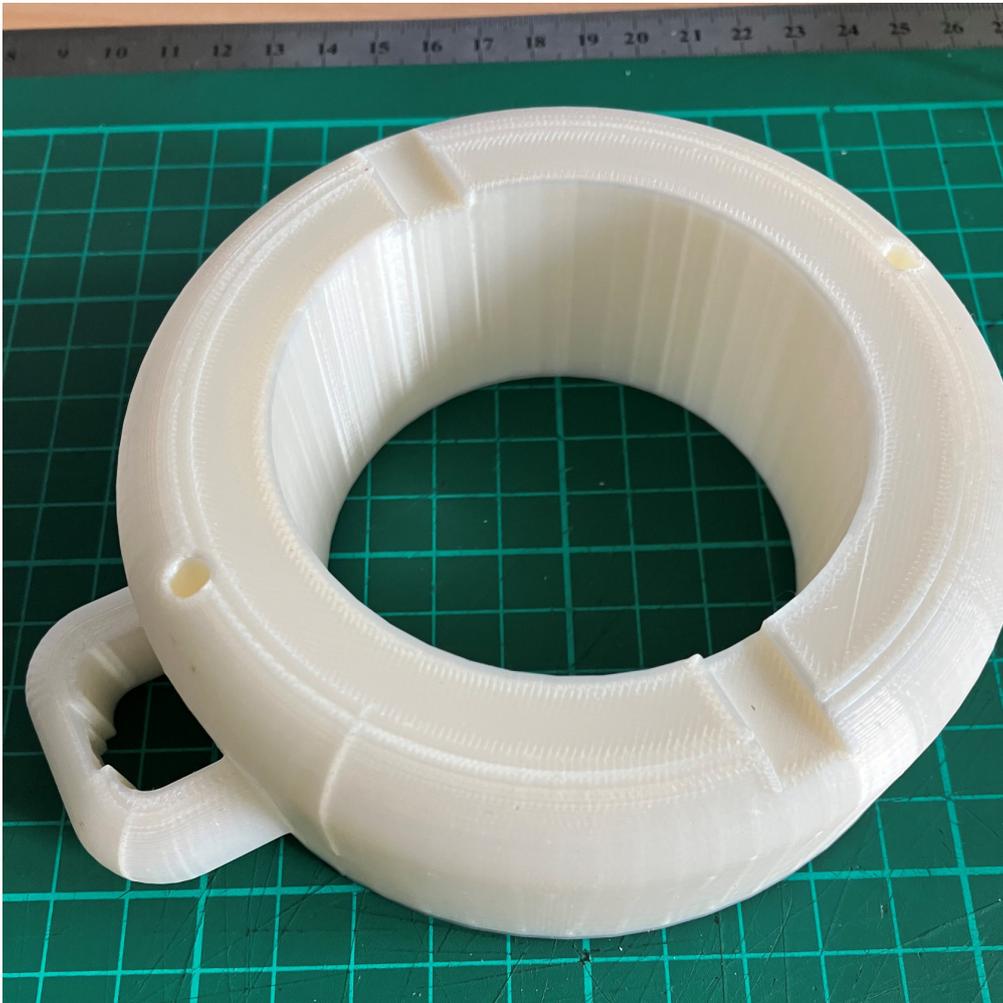
Partie Basse Boîte

Fichier STL : [AssemblageCasagrande - Casagrandeboitebasse-1-1.STL](#)



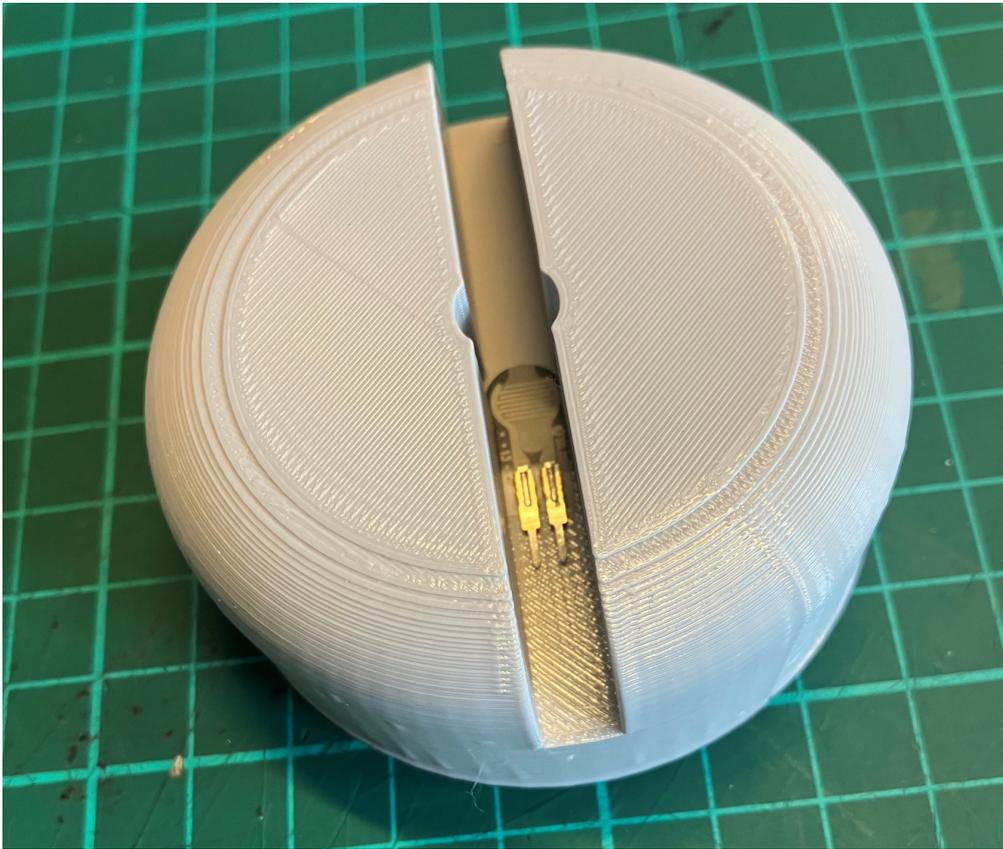
Partie Haute Boîte

Fichier STL : [AssemblageHaut2 - Casagrandeboitehaute-1.STL](#)



Bouchon Boîte

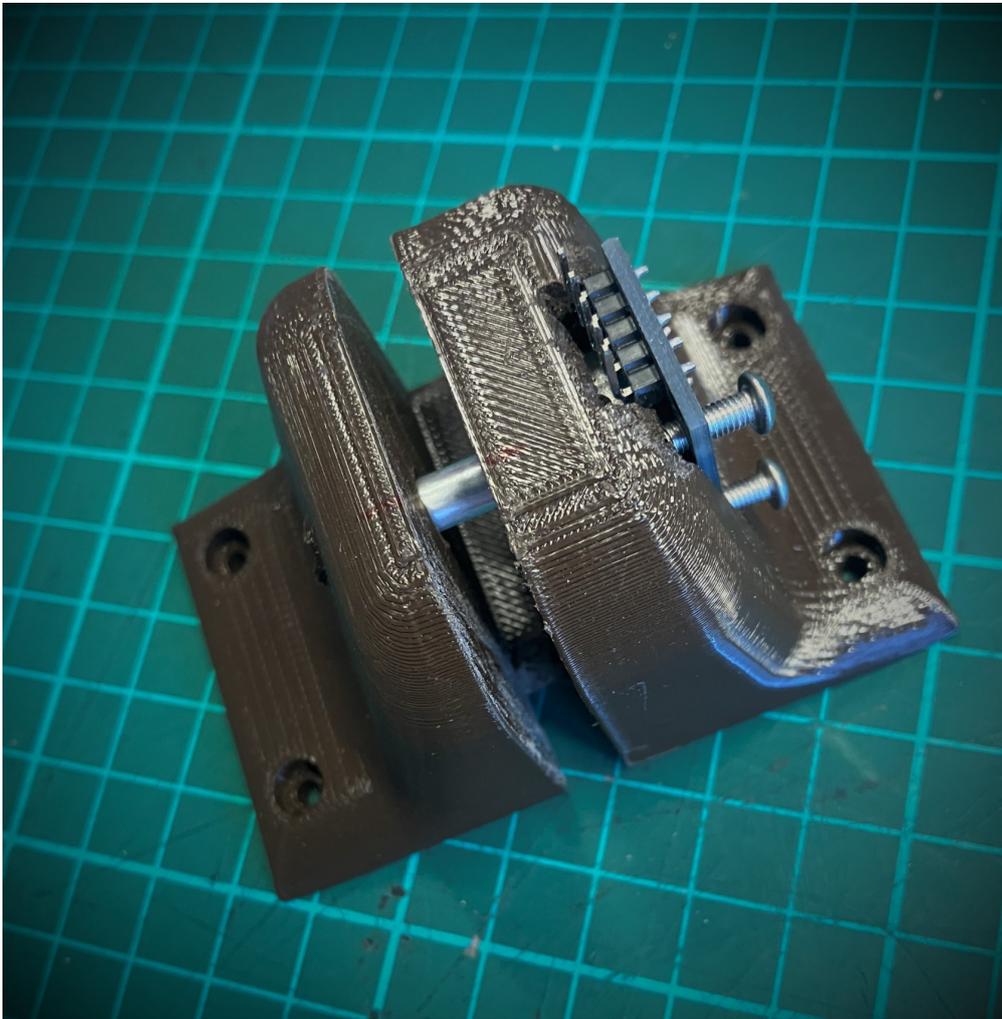
Fichier STL : [AssemblageCasagrande - Casagrandeboitebouchon-1-1.STL](#)



Porte Poulie et Poulie

Fichier STL : [Portepoulie.STL](#)

[Poulie.STL](#)



Porte poulie et potentiomètre

Attaches Levier (x2)

Fichier STL : [Attachelevierx2.STL](#)



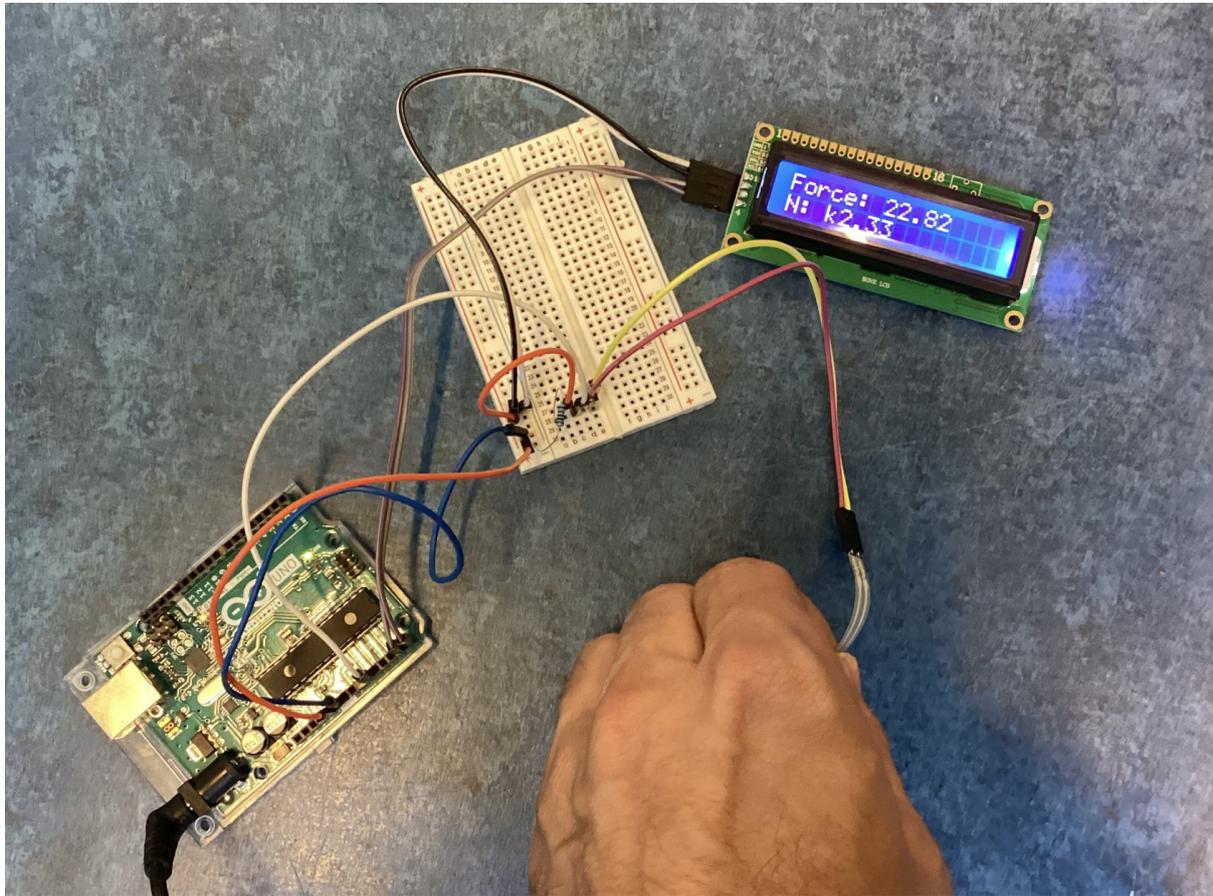
Photo de l'assemblage général de la boîte :



> Branchement du capteur FSR400 pour la force normale σ_N

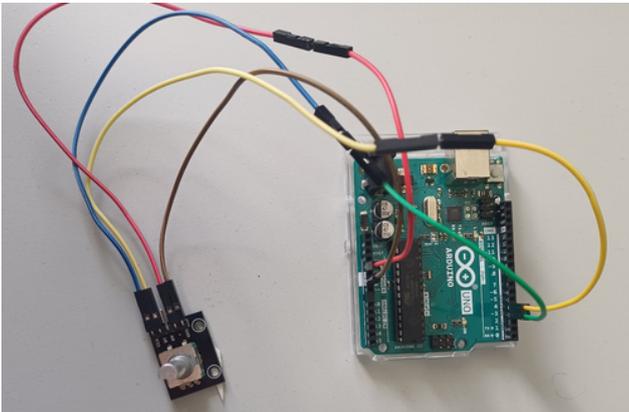
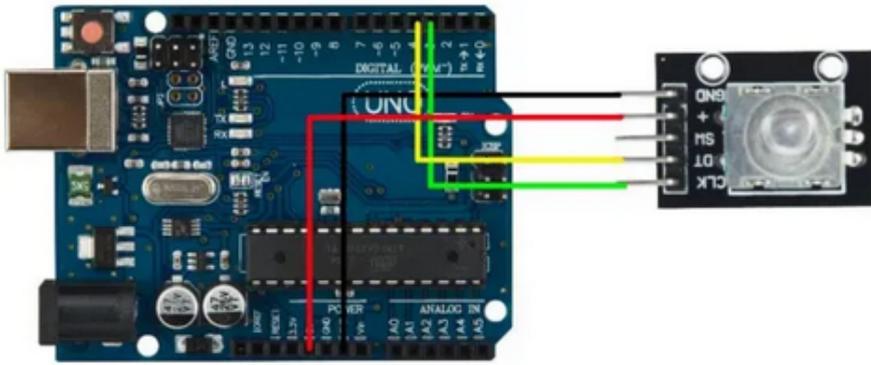
On relie le capteur et l'écran à la carte Arduino en suivant le montage suivant :

[Instructables.com/Arduino-pressure-sensor-FSR-with-LCD-display/](https://www.instructables.com/Arduino-pressure-sensor-FSR-with-LCD-display/) et on choisit une résistance de 4K7 ohm lors du montage, pour avoir une valeur de poids affichée à l'écran la plus fiable au poids réel possible.



> Branchement de l'encodeur rotatif GT055

Afin de déterminer précisément le point de rupture, on utilise un encodeur rotatif qui va donner la valeur de déplacement en degrés. On le branche à une carte Arduino en suivant le montage suivant :



Source : LEXTRONIC, 2022 (<https://www.lextronic.fr/module-encodeur-rotatif-51706.html>)

Encodeur rotatif - Code Arduino

```
```.cpp
// Déclaration des variables
int CLK = 3; // Broche CLK de l'encodeur sur la pin 3 de l'Arduino
int DT = 4; // Broche DT de l'encodeur sur la pin 4 de l'Arduino
int encoderPosCount = 0; // valeur de départ de l'encodeur à zéro
int CLKLast;
int aVal;
boolean bcW;
void setup() {
 pinMode (CLK,INPUT); // Pin CLK déclarée en entrée
 pinMode (DT,INPUT); // Pin DT déclarée en entrée

 CLKLast = digitalRead(CLK); // Renvoi la dernière valeur de l'encodeur
 Serial.begin (9600);
}
```

```

void loop() {
aVal = digitalRead(CLK);

if (aVal != CLKLast) { // Indique que l'encodeur tourne
// Si le bouton tourne, on déterminera son sens
// avec la broche DT
if (digitalRead(DT) != aVal) { // si CLK change en premier - rotation dans le sens des aiguilles
d'une montre
encoderPosCount ++;
bCW = true;
}
else { // sinon c'est DT qui change en premier et la rotation va donc dans le sens inverse
bCW = false;
encoderPosCount--;
}
Serial.print ("Sens de rotation: "); // Affichera le sens de rotation
if (bCW)
{
Serial.println ("clockwise"); // sens des aiguilles d'une montre
}else
{
Serial.println("counterclockwise"); // sens inverse
}
Serial.print("Degré: ");
Serial.println(encoderPosCount*9); // Affiche la position de l'encodeur - comptage des
impulsions
}
CLKLast = aVal ;
}
}

```

## > Découpe laser

### Découpe du Cadre

Fichier svg : [Découpe\\_Cadre\\_Bras\\_Casagrande.svg](#)



## Découpe des bras de levier

Fichier svg : [Découpe\\_Cadre\\_Bras\\_Gasagrande.svg](#)



Code OpenSCAD :

```
//dimensions du bras de levier
largeur = 40; //mm
diametreTrou = 4; //mm
nbUnites = 48; //nombre d'encoches
longueur = 500; //mm longueur de la barre à partir du trou

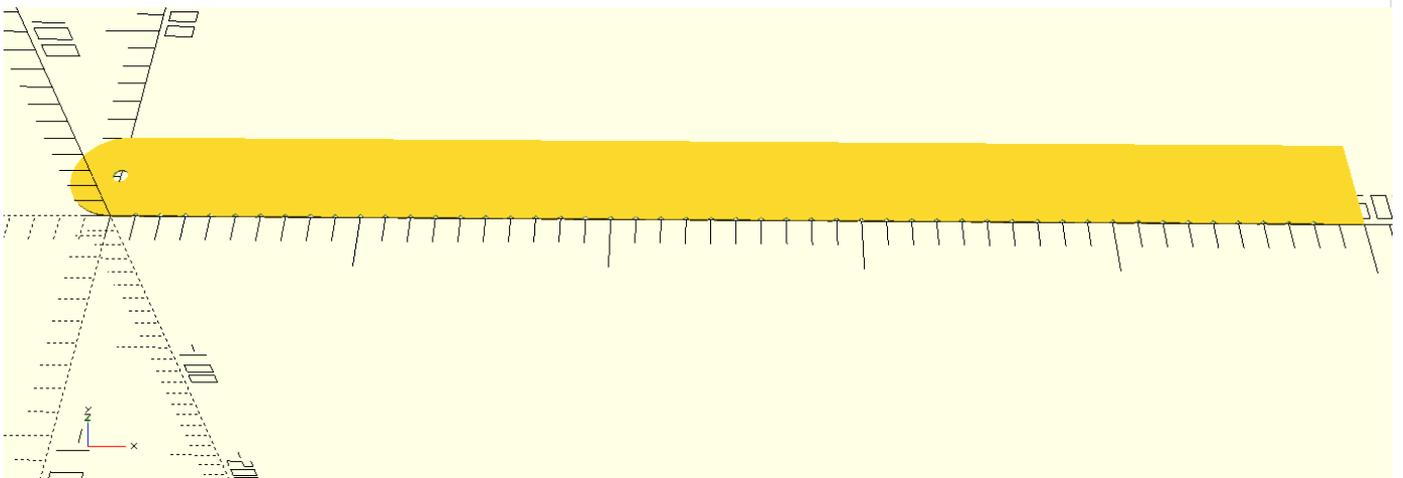
//dimensions des encoches
profondeurEncoche = 2; //mm
var = 0;
```

```

difference(){
 square([longueur, largeur]);
 //trou
 translate([0, largeur/2, 0]){
 circle(diametreTrou/2, $fn = 360);
 }
 //encoches
 for (i=[1:nbUnites]){
 var = i*10;
 translate([var, 0, 0]){
 rotate([0, 0, 45]){
 square([profondeurEncoche, profondeurEncoche], center = true);
 }
 }
 }

 //Pour la forme arrondie
 translate([0, largeur/2, 0]){
 difference(){
 circle(largeur/2);
 //trou
 circle(diametreTrou/2, $fn = 360);
 }
 }
}

```



> Assemblage

## 4. Journal de Bord

### Journal de Bord

#### 5/06 :

Branchement du capteur de pression Arduino

Tests de différentes résistances pour avoir un poids affiché à l'écran (en newton et en kg) fiable.

#### 6/06 :

#### > Création d'un socle permettant de tester le capteur de force (contrainte normale)

Afin de déterminer quelle résistance on utilise pour le capteur de force, on réalise en premier lieu un socle permettant de pouvoir tester différents poids sur le capteur. Le capteur étant très précis, il faut un système qui appuie parfaitement à l'endroit du capteur.

Code OpenSCAD pour le socle sous le capteur :

```
cotecube = 50 ; //mm

difference(){

cube([cotecube,cotecube,10]);

translate([0,20,0]){

cube([25,10,3]);

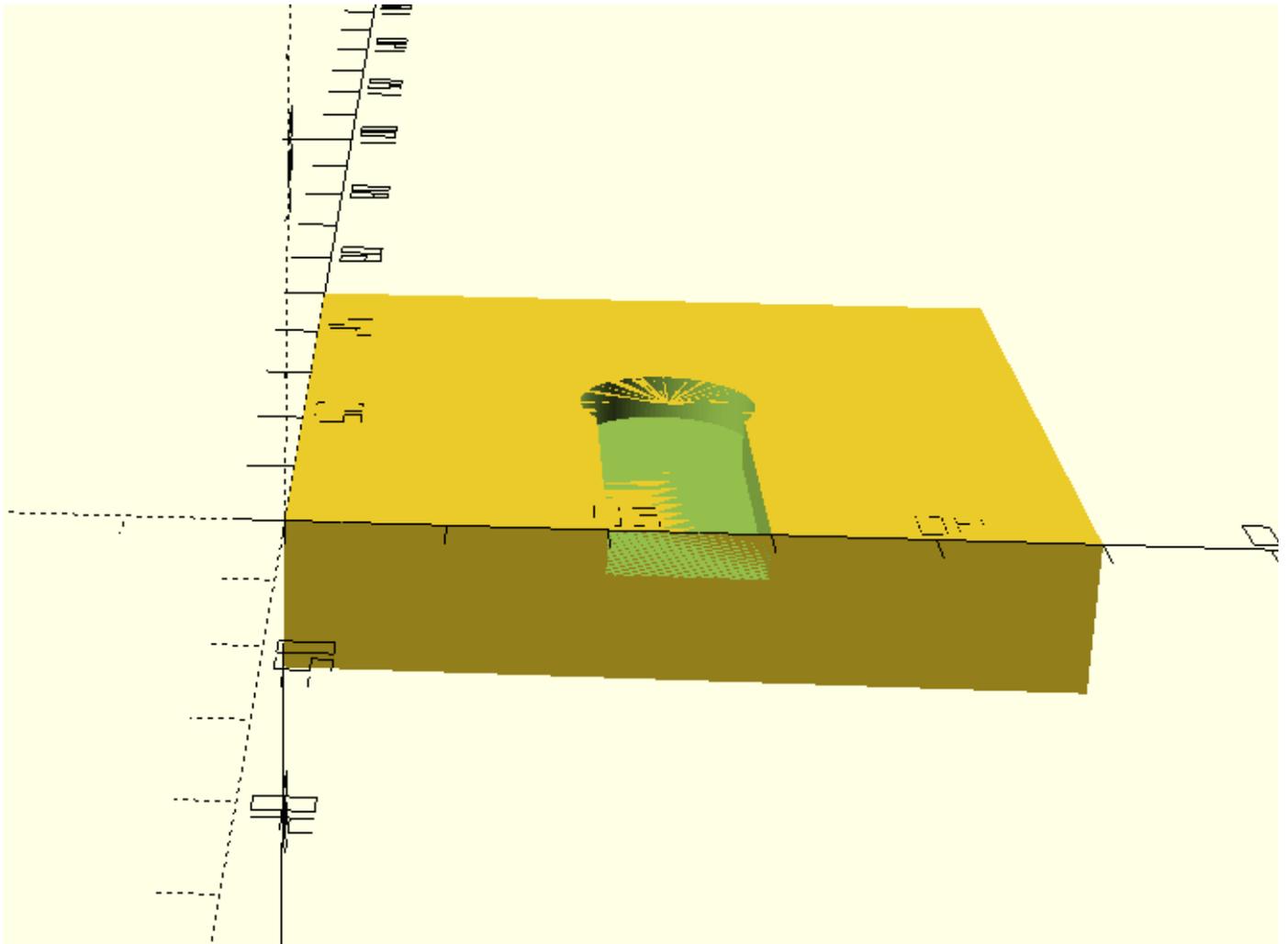
translate([25,5,0]){

cylinder(r=6,h=3,$fn=100);

}

}

}
```



Code OpenSCAD pour la partie supérieure reposant sur le capteur :

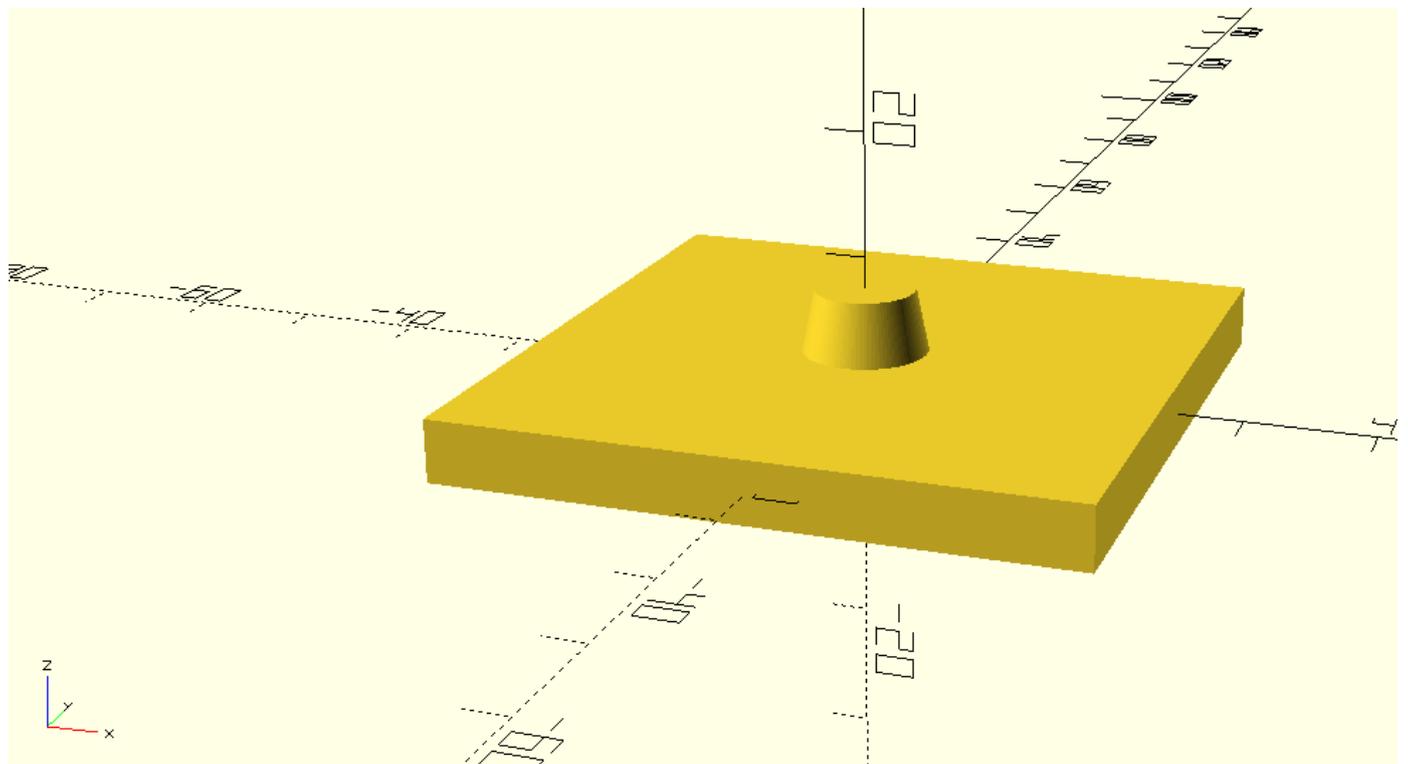
```
cotecube = 50 ; //mm

cube([cotecube,cotecube,5],true);

translate([0,0,2.5]){

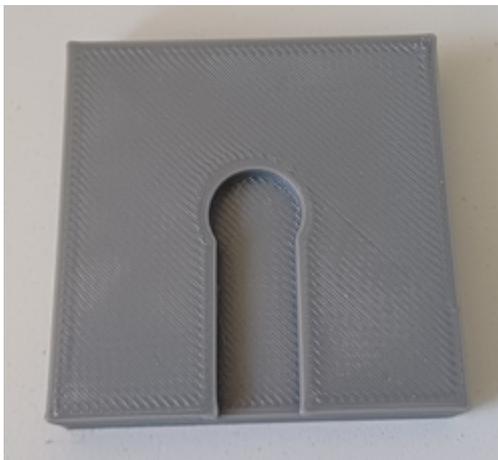
 cylinder(r1=5,r2=4,h=5,$fn=100);

}
```



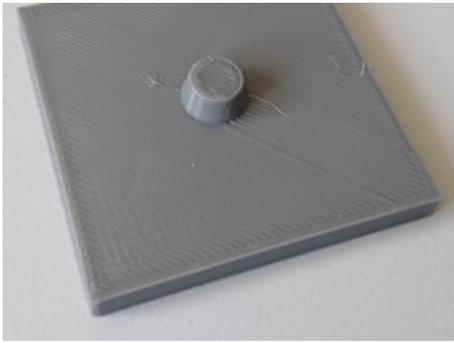
### **> Impression du Socle du testeur**

On crée le socle à l'aide de l'imprimante 3D :



**7/06 :**

### **> Impression de la Partie Supérieur du test**



Masse Partie Supérieur du test capteur : 7.0g

> Réalisation de différents tests avec différentes résistances électriques et différents poids

Poids réel (g)	Résistance 4K7 (g)	Résistance 10K (g)	Résistance 3K3 (g)
428	390	800 g	
425	330		
527	510		
833	700	1800 (?)	600

Les tests ne sont pas très cohérents mais la résistance qui mène au poids le plus fiable est celle de 4K7 ohm.

#### 10/06 :

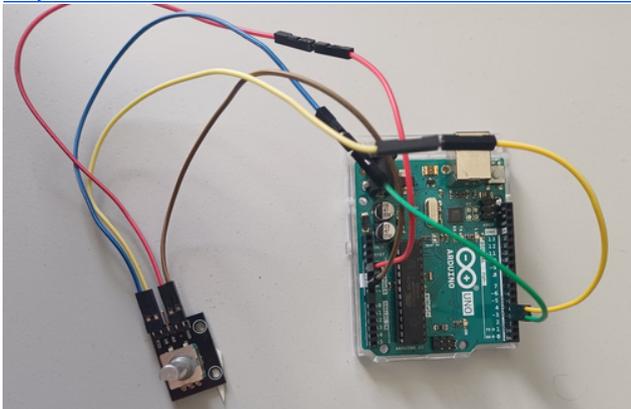
> Dessin d'une pièce sur OpenSCAD pour mesurer la contrainte tangentielle mais abandon parce que ce système ne sera pas assez précis (effets de frottements qui vont fausser la valeur donnée par le capteur).

> Recherche d'un capteur de pression en S à utiliser à la place.

#### 11/06 :

> Branchement de l'encodeur sur une carte Arduino, où on suit le montage suivant :

<https://www.lextronic.fr/module-encodeur-rotatif-51706.html>



> Code Arduino :

```

// Déclaration des variables
int CLK = 3; // Broche CLK de l'encodeur sur la pin 3 de l'Arduino
int DT = 4; // Broche DT de l'encodeur sur la pin 4 de l'Arduino
int encoderPosCount = 0; // valeur de départ de l'encodeur à zéro
int CLKLast;
int aVal;
boolean bCW;

void setup() {
pinMode (CLK,INPUT); // Pin CLK déclarée en entrée
pinMode (DT,INPUT); // Pin DT déclarée en entrée

CLKLast = digitalRead(CLK); // Renvoi la dernière valeur de l'encodeur
Serial.begin (9600);
}

void loop() {
aVal = digitalRead(CLK);

if (aVal != CLKLast) { // Indique que l'encodeur tourne
// Si le bouton tourne, on déterminera son sens
// avec la broche DT
if (digitalRead(DT) != aVal) { // si CLK change en premier - rotation dans le sens des aiguilles d'une montre
encoderPosCount ++;
bCW = true;
}
else { // sinon c'est DT qui change en premier et la rotation va donc dans le sens inverse
bCW = false;
encoderPosCount--;
}
Serial.print ("Sens de rotation: "); // Affichera le sens de rotation
if (bCW)
{
Serial.println ("clockwise"); // sens des aiguilles d'une montre
}
}
}

```

```

{
Serial.println("counterclockwise"); // sens inverse
}
Serial.print("Degré: ");
Serial.println(encoderPosCount*9); // Affiche la position de l'encodeur - comptage des impulsions
}
CLKLast = aVal ;
}

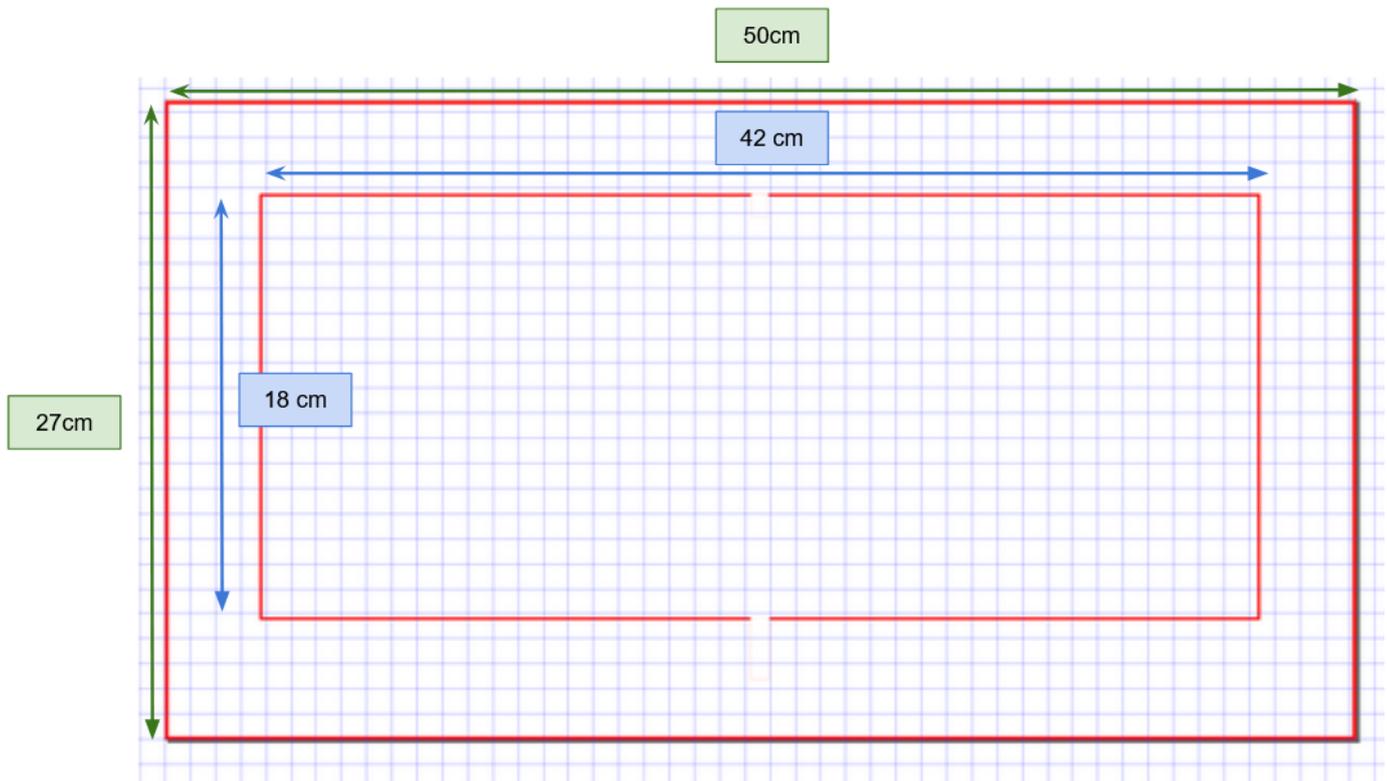
```

Pour chaque clic : indication d'une rotation de 18 degrés.

**12/06 :**

> **Dessin du cadre sur Inkscape :**

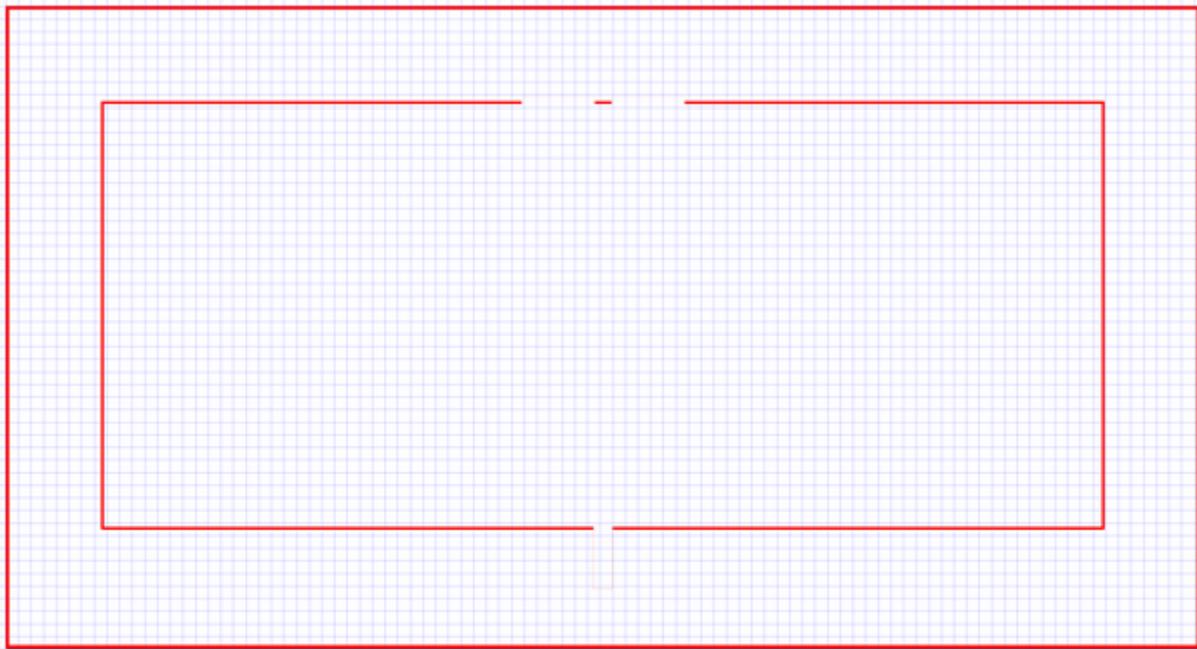
Le cadre présente une encoche pour faire passer le bras de levier et un petit rectangle qui permettra d'appuyer sur le capteur pour la force normale.



**14/06 :**

> **Arrangement du cadre sur Inkscape**

Pour que ce soit plus stable lors de l'utilisation.

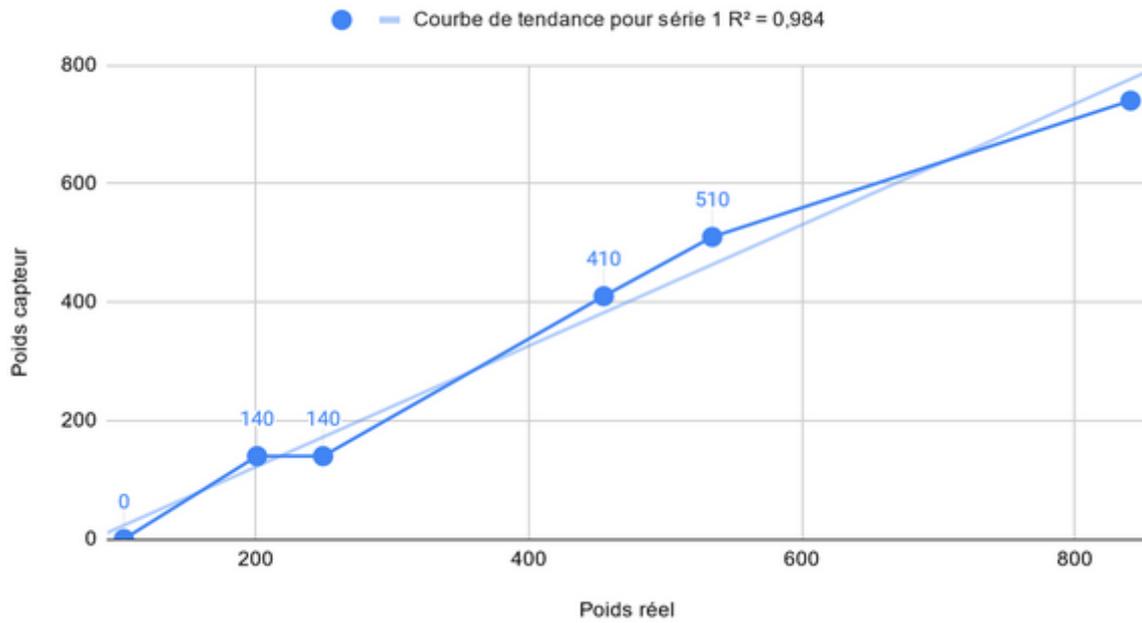


### > **Second test du capteur - force normale**

Lors des tests pour le capteur de la force normale, l'utilisation d'une résistance de 4K7 ohm menait à des poids proches de ceux réels. Bien que le capteur ne donne pas la valeur exacte, on réalise des tests avec différents poids pour voir si on peut trouver une relation de proportionnalité entre la valeur que le capteur nous donne et la valeur réelle :

Poids réel (g)	Poids avec Résistance 4K7 ohm (g)
103.3	0
200.8	140
249.2	140
454.8	410
534.3	510
840.8	740
836.1	

## Poids capteur en fonction du poids réel



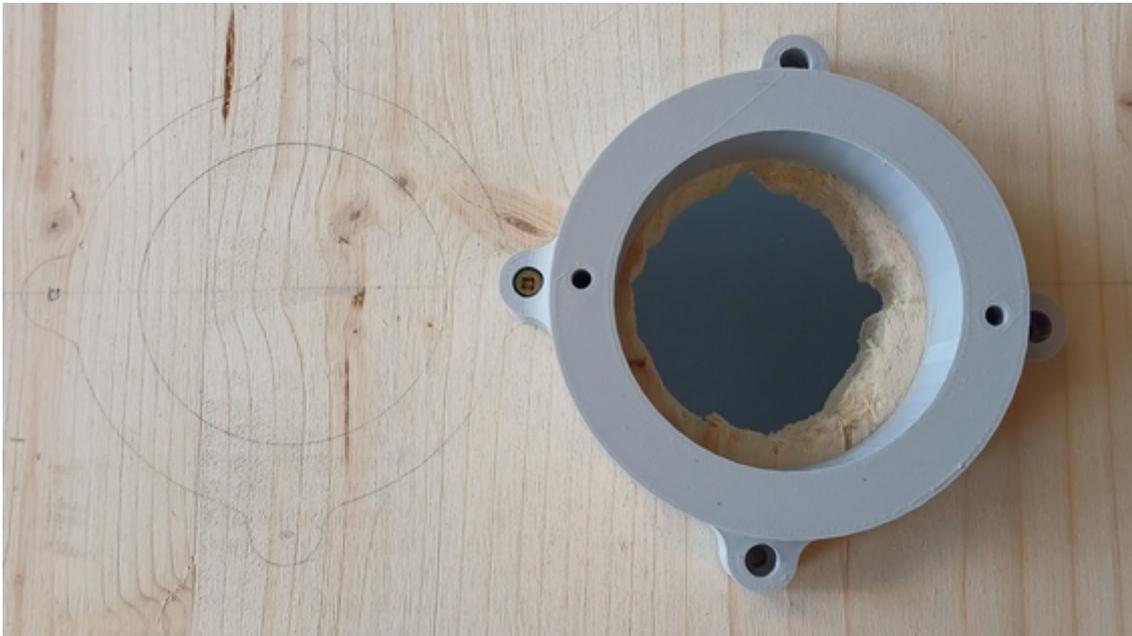
Il faut multiplier la valeur donnée par le capteur par environ 1.3 pour obtenir la valeur de poids réelle, mais le capteur reste plutôt imprécis.

**20/06 :**

> Visser pieds pour les bras de levier sous la table



> Visser socle boîte cisaillement et faire un trou pour vider la boîte du sable après utilisation :



> Découpe laser bras de levier :



> Découpe cadre :



**24/06 :**

On réalise des tests du capteur force normale avec le système entier :

Poids éléments ( $m_{\text{éléments}}$ ) :

> Bras : 134.2 g

- > Cadre : 396.4 g
  - > Total poids bras + cadre =  $m_{\text{éléments}}$  : 530.2 g
  - > Piston : 48.6 g
  - > Bouteille qu'on utilise pour la force normale  $m_{\text{eau}}$  : 857.8 g
- Sans ajouter la bouteille, le capteur affiche : 9.59 N et 0.89kg

- Calcul du facteur d'augmentation de la masse :  $f = L/$

L : longueur entre l'articulation du bras de levier et l'encoche sur laquelle est posée la masse/la bouteille

l : longueur entre l'articulation du bras de levier et l'encoche sous laquelle est glissée le cadre (fixe, 13 cm pour notre machine)

- Calcul de la pression :  $P = m \cdot g$   
 $m$  : masse en kg donc  $m = m_{\text{eau}} \cdot f + m_{\text{éléments}}$   
 $g$  : accélération :  $9.81 \text{ m} \cdot \text{s}^{-2}$

Ici,  $m_{\text{éléments}} = 530.2 \text{ g} = 0.5302 \text{ kg}$  et  $m_{\text{eau}} = 857.8 \text{ g} = 0.8578 \text{ kg}$

Calcul en N :

?<sub>N</sub> placé à la 26<sup>ième</sup> graduation donc  $P = (0.5302 + 26/13 \times 0.8578) \cdot 9.81 = 22.02 \text{ N} = 2.24 \text{ kg}$  --> capteur affiche moins de 1 kg

?<sub>N</sub> placé à la 28<sup>ième</sup> graduation donc  $P = (0.5302 + 28/13 \times 0.8578) \cdot 9.81 = 23.32 \text{ N} = 2.38 \text{ kg}$  --> mais capteur affiche moins de 1 kg

On réalise d'autre tests mais le capteur reste à 0.60 ou 0.80 kg quelque soit la graduation où on pose la bouteille --> imprécision du capteur, il ne doit pas bien capter le cadre posé dessus.

### 26/06 :

On refait les tests avec un piston différent qui permet au capteur de mieux fonctionner :



Numéro de graduation où est placée la bouteille	Pas de bouteille	26e	28e	35e	40e
Poids réel (N)	5.19	22.02	23.32	27.86	31.09
Poids par capteur (N)	5.37	10.36	17.07	32.03	45

Encore des incohérences dans le poids donné, on ne trouve pas relation de proportionnalité entre la réalité et les valeurs affichées par les capteur de force.

On pense alors utiliser différents capteurs pour capter le poids de la force normale: un capteur Arceli HX711 ADC

**28/06 :**

Branchement du nouveau capteur pour la force normale : le capteur Arceli HX711 ADC

---

Revision #20

Created 5 April 2023 12:20:41 by Pierre Thery

Updated 28 June 2024 09:46:49 by Dufour Johane