

## Rapport de séance :

Séance du 22/12 :

Objectif de la séance : Ecrire le code pour tester notre projet avec une capsule.

Cette séance nous nous sommes intéressées au « capacity touch ». Dans notre cas, c'est notre corps et la capsule qui joue le condensateur. L'arduino va mesurer en combien de temps le condensateur se charge, ce qui va lui permettre de calculer la capacité de celui-ci et donc permettre de déterminer la valeur de la résistance dont nous avons besoin (Amy s'est occupée de cette partie).

Notre but est que lorsque qu'une capsule est touchée un son sorte.

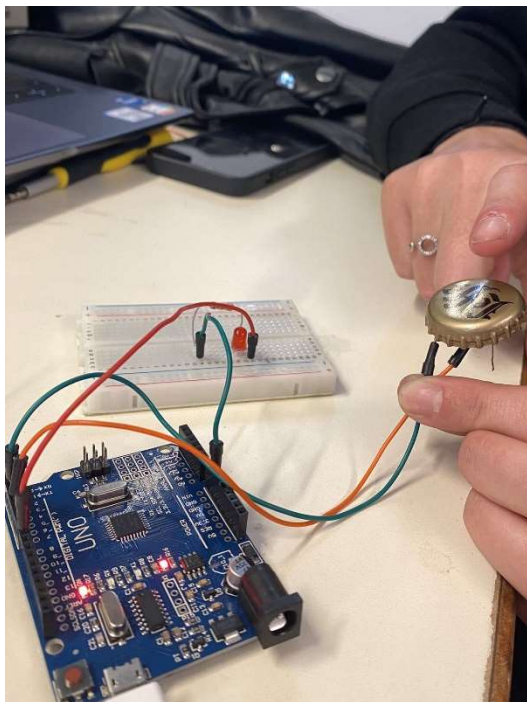
Dans un premier temps nous avons fait le test avec une capsule et lorsque celle-ci était touchée une led devait s'allumer.

Je me suis d'abord penchée sur la partie code. J'ai donc commencé par chercher sur internet et j'ai trouvé un site( [Capacitive-Touch Arduino Keyboard Piano : 10 Steps \(with Pictures\) - Instructables](#)) où il donnait un exemple de code. Je me suis rendue compte qu'il fallait télécharger la bibliothèque CapSense, ce que j'ai fait. Le code par contre ne fonctionnait pas donc j'ai écrit un nouveau code :

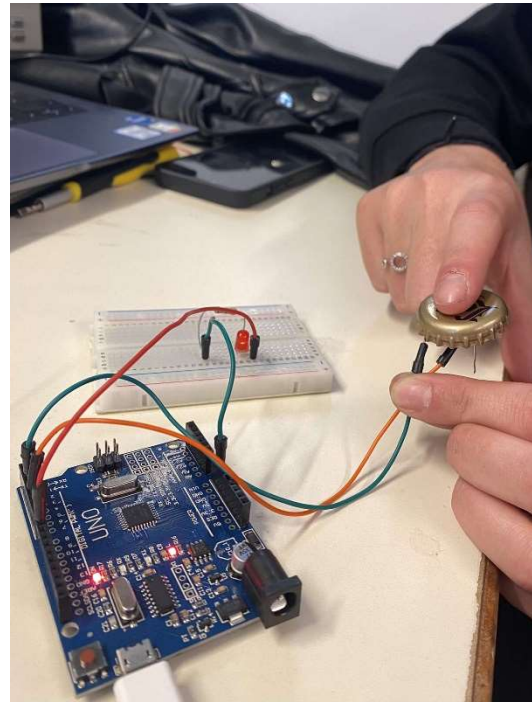
```
CapacitiveSensorSketch.pde
1  #include <CapacitiveSensor.h>
2
3  /*
4   * CapacitiveSense Library Demo Sketch
5   * Paul Badger 2008
6   * Uses a high value resistor e.g. 10M between send pin and receive pin
7   * Resistor effects sensitivity, experiment with values, 50K - 50M. Larger resistor values yield larger sensor values.
8   * Receive pin is the sensor pin - try different amounts of foil/metal on this pin
9   */
10
11
12 CapacitiveSensor cs_4_2 = CapacitiveSensor(4,2); // 10M resistor between pins 4 & 2, pin 2 is sensor pin, add a wire and or foil if desired
13
14 void setup()
15 {
16   cs_4_2.set_CS_Autocal_Millis(0xFFFFFFFF); // turn off autocalibrate on channel 1 - just as an example
17   Serial.begin(9600);
18 }
19
20 void loop()
21 {
22   long start = millis();
23   long total1 = cs_4_2.capacitiveSensor(30);
24
25   if(total1 > 1000) {
26     digitalWrite(8, HIGH);
27   } else {
28     digitalWrite(8, LOW);
29   }
30
31   delay(10); // arbitrary delay to limit data to serial port
32 }
33
```

Explication du code : J'utilise la librairie CapSense qui donne la capacitance. Une fois la capacitance récupérée, je fais allumer la led lorsque la capacitance atteint un certain niveau (1000). Lorsque la capacitance est au-dessous de ce niveau la led s'éteint.

J'ai ensuite fait les branchements de la led. Puis, nous avons mis en commun, avec Amy, nos montages ainsi que le code afin de tester si tout marchait. Au moment de l'importation du code dans l'arduino, je me suis rendue compte que je n'avais pas téléchargé le driver. Une fois le driver téléchargé, nous avons obtenu le résultat suivant :



Led éteinte



Led allumée