# Kombucha Workshop

*Lélian N.*

Kombucha is drink of Mongolian origin composed of tea and sugar fermented by a symbiotic culture of yeast and bacteria.

**Biohybrid Devices: Prototyping Interactive Devices with Growable Materials**

Madalina Nicolae
Léonard de Vinci Pôle Universitaire, Research Center
Paris La Défense, France
Saarland University, Saarland Informatics Campus
Saarbrücken, Germany
Télécom Paris, CNRS i3 (UMR 9217)
Institut Polytechnique de Paris
Palaiseau, France
madalina.nicolae@devinci.fr

Vivien Roussel
Léonard de Vinci Pôle Universitaire, Research Center
Paris La Défense, France
Nîmes University, Laboratories Projekt and UFR Chrome
Nîmes, France
vivien.roussel@devinci.fr

Marion Koelle
OFFIS – Institute for Information Technology
Oldenburg, Germany
marion.koelle@offis.de

Samuel Huron
Télécom Paris, CNRS i3 (UMR 9217)
Institut Polytechnique de Paris
Palaiseau, France
samuel.huron@telecom-paris.fr

Jürgen Steimle
Saarland University, Saarland Informatics Campus,
Saarbrücken, Germany
steimle@cs.uni-saarland.de

Marc Teyssier
Léonard de Vinci Pôle Universitaire, Research Center,
Paris La Défense, France
marc.teyssier@devinci.fr

Base Materials | Electronics embedding during material's Life Cycle Phases | Biohybrid Devices
Bacterial Cellulose | Traditional electronics | Growth | Stabilization | Inanimate

**Figure 1: We combine Bacterial Cellulose with traditional electronics to create *Biohybrid Devices*. We identify three phases to embed electronics, based on the material's life cycle: Growth, Stabilization, and Inanimate. The embedding techniques allow to grow functional input and output devices.**

**ABSTRACT**

Living bio-materials are increasingly used in HCI for fabricating objects by growing. However, how to integrate electronics to make these objects interactive still needs to be clarified. This paper presents an exploration of the fabrication design space of Biohybrid Interactive Devices, a class of interactive devices fabricated by merging electronic components and living organisms. From the exploration of this space using bacterial cellulose, we outline a fabrication framework centered on the biomaterials' life cycle phases. We introduce a set of novel fabrication techniques for embedding conductive elements, sensors, and output components through biological (e.g. bio-fabrication and bio-assembling) and digital processes. We demonstrate the combinatory aspect of the framework by realizing three tangible, wearable, and shape-changing interfaces. Finally, we discuss the sustainability of our approach, its limitations, and the implications for bio-hybrid systems in HCI.

**ACM Reference Format:**
Madalina Nicolae, Vivien Roussel, Marion Koelle, Samuel Huron, Jürgen Steimle, and Marc Teyssier. 2023. Biohybrid Devices: Prototyping Interactive Devices with Growable Materials. In *The 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23), October 29–November 01, 2023, San Francisco, CA, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3586183.3606774

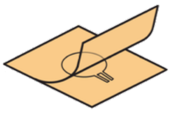
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
*UIST '23, October 29–November 01, 2023, San Francisco, CA, USA*
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0132-0/23/10...$15.00
https://doi.org/10.1145/3586183.3606774

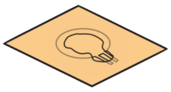
**1 INTRODUCTION**

The HCI community has pivoted towards making use of the natural growth or reproduction processes of living organisms for fabrication. As an emergent and promising material practice, Growing Design [15] employs fermentation, biomineralization or cellular division as a form of bio-fabrication. Growing materials is a compelling, novel addition to the fabrication of DIY Materials [7], allowing designers to engage with the material at different steps

We used it here to create flexible material in which we can implement electronics components, based on : *Biohybrid Devices: Prototyping Interactive Devices with Growable Materials,* Madalina Nicolae et al. 2023.

On the surface of the fermentation, a film made of glucose is created by the bateries in order to protect them and explore the world around them. We then use the film to create our kombucha electronic component.
There is different methods to create this component, the two we've explored in this workshop is layering and growing.





*Please find with the QR code the git-hub repositery links to this project :*

## Table of Contents

***Stretch Sensor*** ______ ***2***

***Matrix sensor*** ______ ***4***

***Capacitive Sensor*** ______ ***6***

# Stretch Sensor

## Explaination

I made a Stretch sensor with the layering method. I used 2 slice of scoby in betwwen which I put a piece a resistive textile.  In order to visualize the sensitivity of the sensor we use a voltage divider bridge to visualise the voltage delta between the flat state and the bend state. Then with a microcontroler with and analog read capability we can graphically visualize the changing state. Check out the QR code to watch a video of the sensor.

## Pictures and Videos

Demo Video

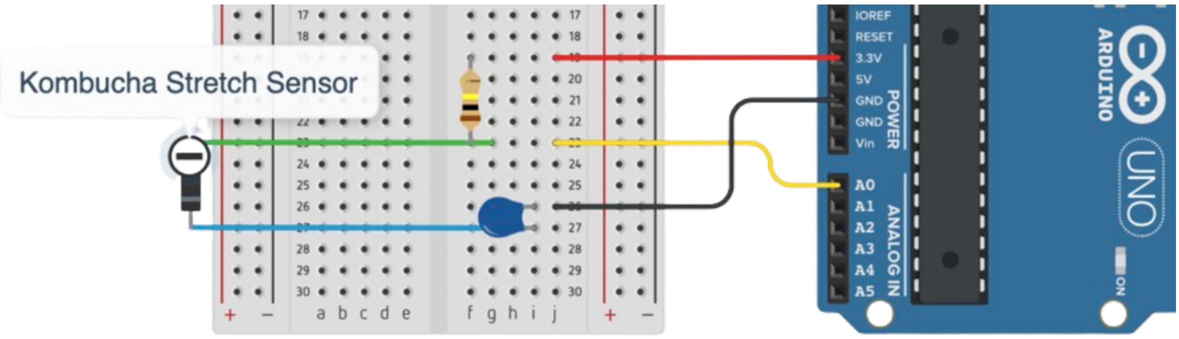
Ohmeter video

## Realisation

Components :

- Stretch Sensor
- 100kΩ resistor
- Arduino Uno
- 1pF capacitor to reduce noise
- Wires

Circuit :

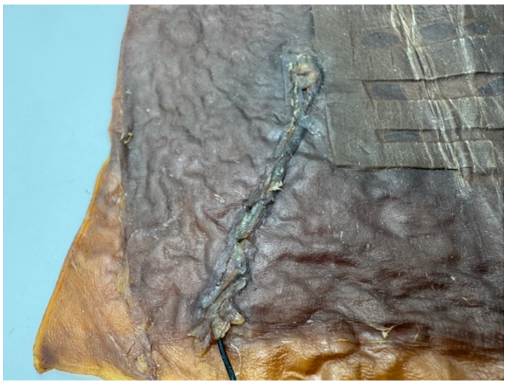


Code :

```
//C++ code
//Implement in the Arduino for Kombucha Stretch Sensor

int stretchSensor  = A0;   //analog entry
int precision = 10;        //number of read by value displayed

void setup() {
  Serial.begin(115200);
  Serial.println("KombuchaWorkShop - Stretch Sensor");
  Serial.println("Lélian Nahon @IFT");
}

void loop() {
    Serial.println(state());
    delay(200);
}

long state() {
  long read = 0;
  for(int i = 0;i<precision;i++){
    read += analogRead(stretchSensor);
  }
  read = read/precision;
  return read;
}
```

## Interresting point

During my tests, one of the wire of the sensor detachs itself. But with the help of ly supervisor and the kombucha properties I could have fix it quite easily. With some water with sugar kombucha get more flexible, it allows to cut it in order to put back the wire. And then, I put some kombucha fall on to the back in place wire to maintains it in the layering !

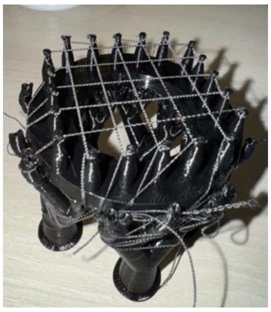


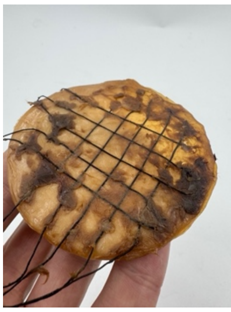
# Matrix sensor

## Explaination

For matrice sensor I wanted to use the growing around method. However, my matrice wasn't perfectly into the kombucha so I fix it but using the layering method and I cover it with a layer of kombucha.
I printed a 3d model to hold conductive string which would be the string of the matrice. Then I take it out and find my matrice not completely inside the kombucha… I used a Muca board, in order to treat signal. I connect it to an arduino, and then with a serial communication between arduino and processing (software) I could visualize the matrice touch point.
At the moment, I cannot visualize perfectly where I click on the matric, nevertheless it acts like a capasitive sensor which have the ability to know to pressure my finger gives.

## Pictures and Videos





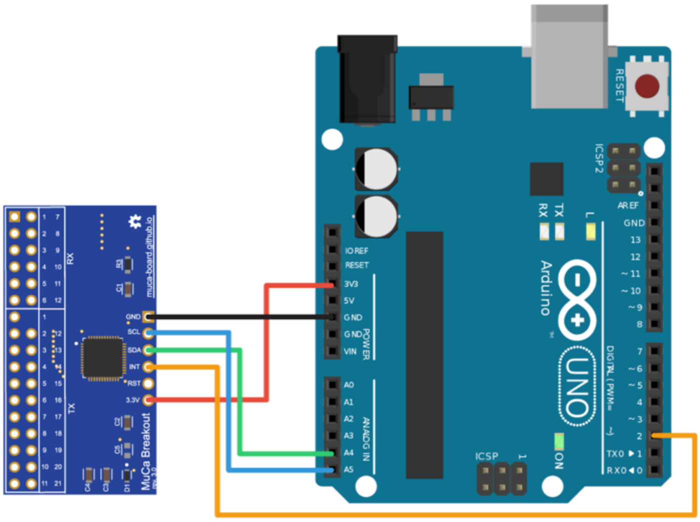
Demo Video

## Realisation

Components :

- Matrix
- Muca Board
- Arduino Uno
- Wires

Circuit :



Then connect the vertical lines on the rx entry and the horizontal on the tx entry. (from the Muca git-hub)

Code :

```
//C++ code
//Implement in the Arduino for Matrix Sensor
//This code only provide a Serial output to visualize nuerically each touch
//Please use the version in git hub to visualize it on processing

#include <Muca.h>
//librairy which you will need the one that you can find
//on muca web site is not complete please use mine that I
//have modify on the git of this project

Muca muca;

void setup() {
  Serial.begin(115200);
  muca.init(true);
  muca.setResolution(90, 90);
  muca.setReportRate(6);
  Serial.println("KombuchaWorkShop - Matrix Sensor");
  Serial.println("Lélian Nahon @IFT");
}
```

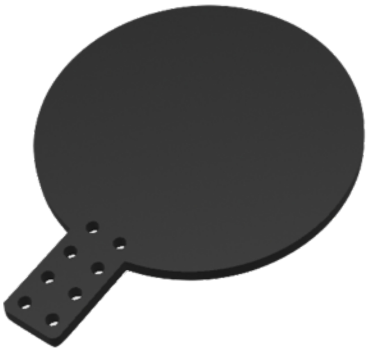
```
void loop() {
  GetTouchSimple();
  delay(5);
}

void GetTouchSimple() {
  if (muca.updated()) {
    for (int i = 0; i < muca.getNumberOfTouches(); i++) {
      if (i != 0)Serial.print("|");
      Serial.print(muca.getTouch(i).id); Serial.print(":");
      Serial.print(muca.getTouch(i).flag); Serial.print(":");
      Serial.print(muca.getTouch(i).x); Serial.print(":");
      Serial.print(muca.getTouch(i).y); Serial.print(":");
      Serial.print(muca.getTouch(i).weight);
    }
    Serial.println("|");
  }
}
```

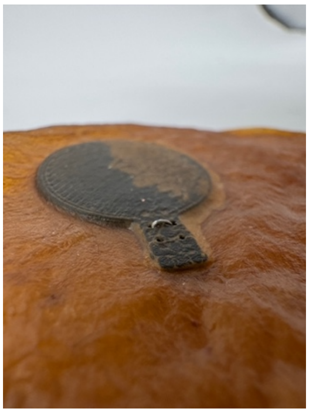
# Capacitive Sensor

## Explaination

For this last sensor, the sensitive one I choose the growth around method. I consist of a small conndductive PLA piece which has been placed on the kombucha after it reach a width of about 3-4 mm. This sensor consist to be a point of connection of the circuit to the ground. The ground is the human hand, when someone touch the sensor the signal which is 1 at the beginning turn to 0 (as the ground is in the circuit). Like this with just 2 digital pin of an arduino we can detect when the sensor is touch.

## Pictures and Videos





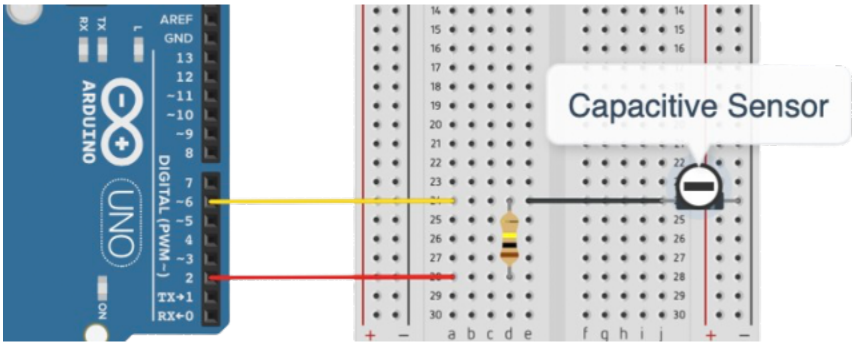
Demo Video

## Realisation

Components :

- Capacitive sensor
- Arduino Uno
- 100kΩ resistor
- Wires

Circuit :



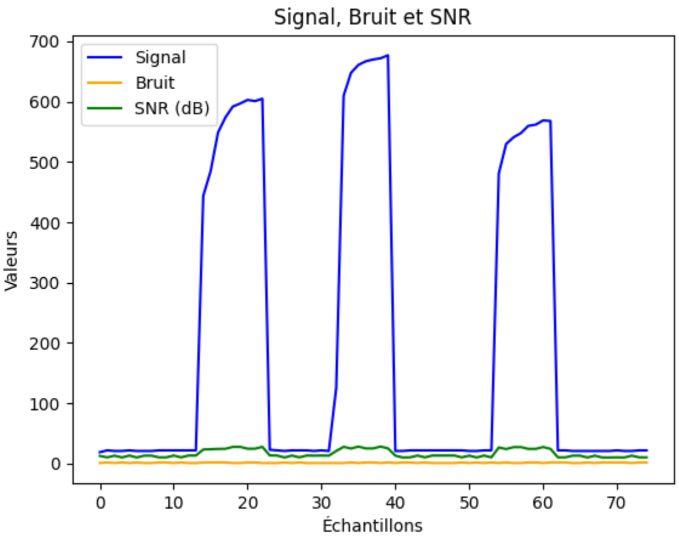
Code:

```
//C++ code
//Implement in the Arduino for Sensitive Sensor

#include <CapacitiveSensor.h>

int precision = 10;
CapacitiveSensor cs  = CapacitiveSensor(2, 6);

void setup() {
  Serial.begin(115200);
  Serial.println("KombuchaWorkShop - Capacitive Sensor");
  Serial.println("Lélian Nahon @IFT");
  pinMode(2,OUTPUT);
  pinMode(6,INPUT);
}

void loop() {
  digitalWrite(2,HIGH);
  Serial.println(state());
  delay(50);
}

long state(){
  long read = 0;
  for(int i = 0; i<precision;i++){
    read = read + cs.capacitiveSensor(100000);
  }
  read = read/precision;
  return read;
}
```

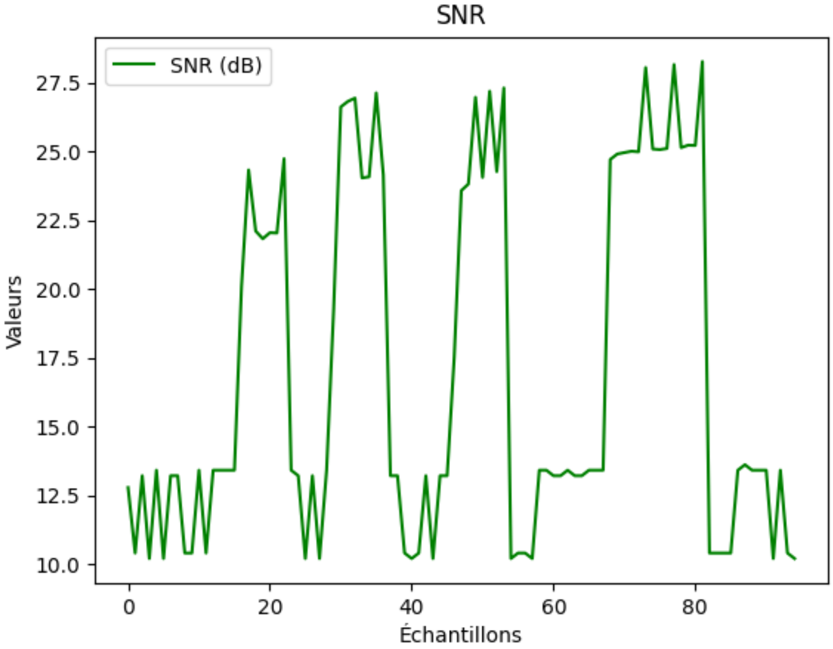
SNR :

The Signal-to-Noise Ratio (SNR) is a metric that expresses the power of a signal relative to the power of the noise. It is expressed in decibels (dB).

$$SNR = 20.\log\left(Signal/Noise\right)$$



In this graph we visualize the noise (orange) the signal (blue) when I touch the sensor it goes up and stay low otherwise. And in green we have the Signal to Noise Ratio. On the following one you have the same conditions but zoomed for the SNR values only.



As the SNR ratio is below 30dB, it is communly say that's a good value !

You can find in the git all the c++, python, graph, 3D files I use for this workshop.

*Image from* **:**
Biohybrid Devices: Prototyping Interactive Devices with Growable Materials, Madalina Nicolae **et al. 2023
Lélian Nahon – Kombucha Workshop
Tinkercad screenshots