Nina Guérin, Lara Narbona Sabaté and Adrien Vergès

17-01-2017, First protocol:

Do lentils sense a wider light-wavelength spectrum than an electronic sensor?

The aim of this project is to compare the spectrum of light-wavelength that lentils sense in comparison on which spectrum does an electronic sensor perceive.

Materials:

For the biosensor:
- Green lentils
- Cotton
- Water

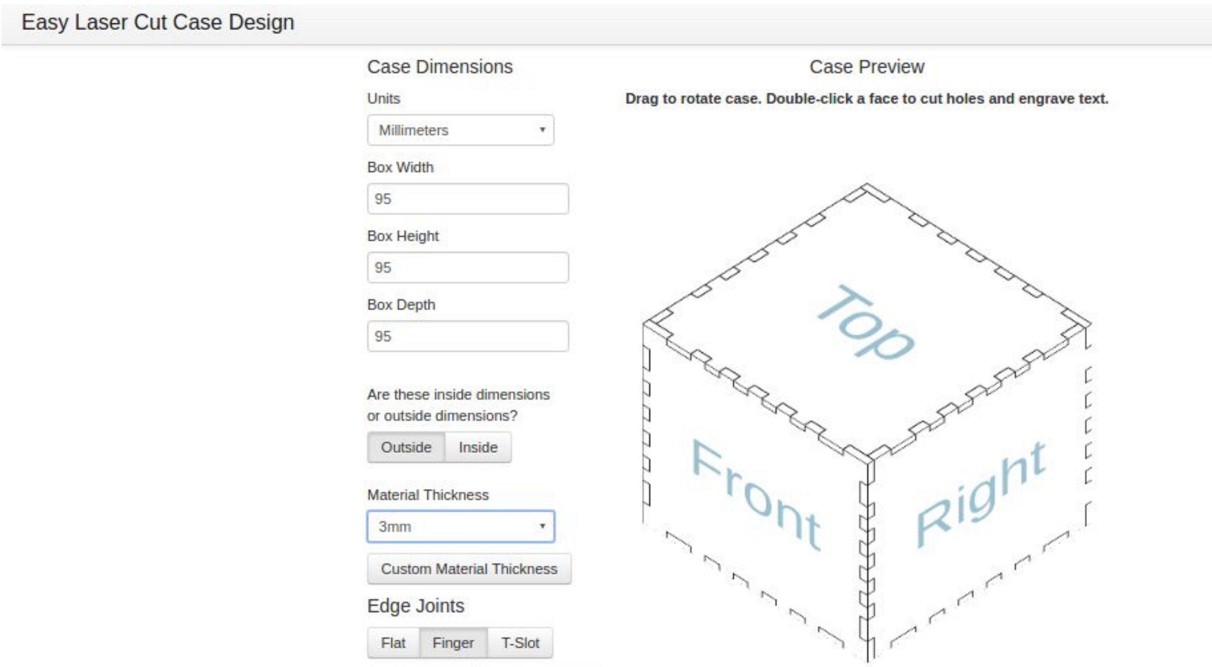
For the electronic sensor:
- Adafruit light sensor TSL2561
- Arduino Leonardo
- Cables

For the light exposure:
- Different LEDs of different wavelengths:
    - UV
    - Blue
    - Green
    - Red
    - IR
- Crosslinker ?
- 19 boxes (to prevent external light to reach the lentils)
    - 15 boxes for non-control experiences
    - 4 boxes for positive and negative controls
- Black scotch

<u>Protocole:</u>

<u>To prepare the boxes:</u>

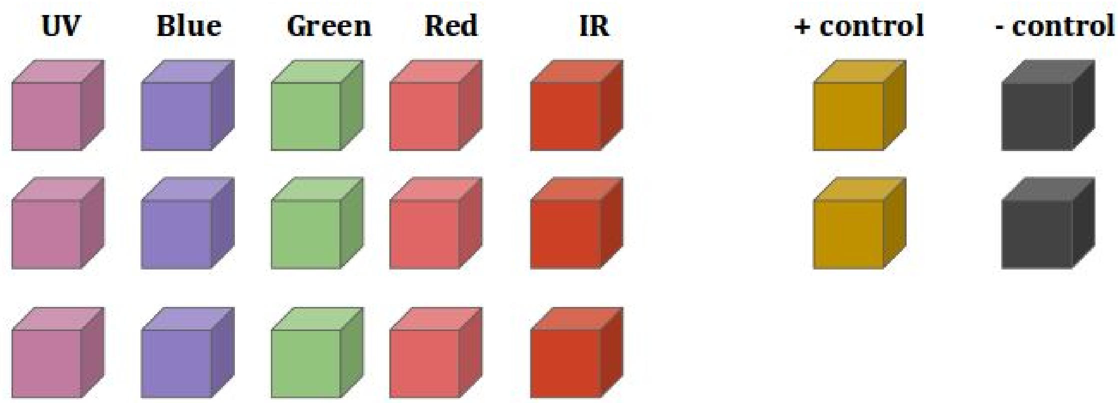
- To make the plans, "makercase" was used, with the following settings:



- Then, once the plans exported to the computer, CorelDRAW Essentials X6 was used to send them to the laser cutter (Epilog Laser mini).
- The used wood was 5mm thick (boxes one to five) or 3mm thick (boxes six to nineteen).
- The boxes were assembled and then scotched with gray scotch (all the parts except one, that will serve to open/close).
- For the non-control boxes (boxes one to fifteen), the light system was prepared in the following way:
    - LED were put on the inside of the box with both of its terminals *protruding* outside.
    - They were scotched: terminals won't move and external light won't enter the box.
    - VOIR ALIMENTATION AVEC KEVIN. COMMENT FAIRE CELLE DU CROSSLINKER?
    - All LEDs will be powered on the same way, so the conditions in every box will be the same.
- For the control boxes (box sixteen to nineteen):
    - Positive control:
        - 2 boxes without a LED nor scotch to cover the hole: to let exterior light enter.
    - Negative control:

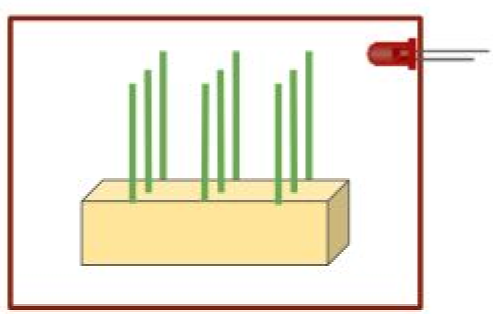
- 2 boxes without a LED but with scotch in the hole: without light exposure.

- We put cotton on the inferior of each box to create a place where to grow lentilles.



<u>To test the biosensor:</u>

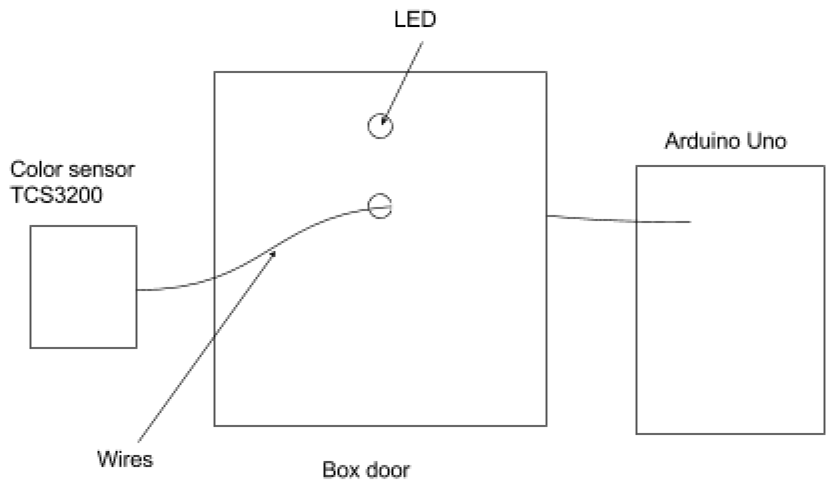
- Germinated-in-the-dark lentils were taken: they were ready to grow but hadn't a sprout yet.
- 9 lentils were distributed on the different boxes in the following way:



- The lentils must be on the cotton and separated by one inch of each other.
- Lentils of each box were photographed before the box were closed.
- All boxes were closed in order to start the experience.
- After 18h of the day after the boxes were opened.
- Lentils of each box were photographed in order to be able to analyze its growth.

To test electronic sensor:

- After the lentils experiment was done, the boxes were reused to test the Adafruit light sensor TSL2561 and know which spectrum does it sense.
- The sensor was put in the middle of the boxes as in the schema:



Data analysis:

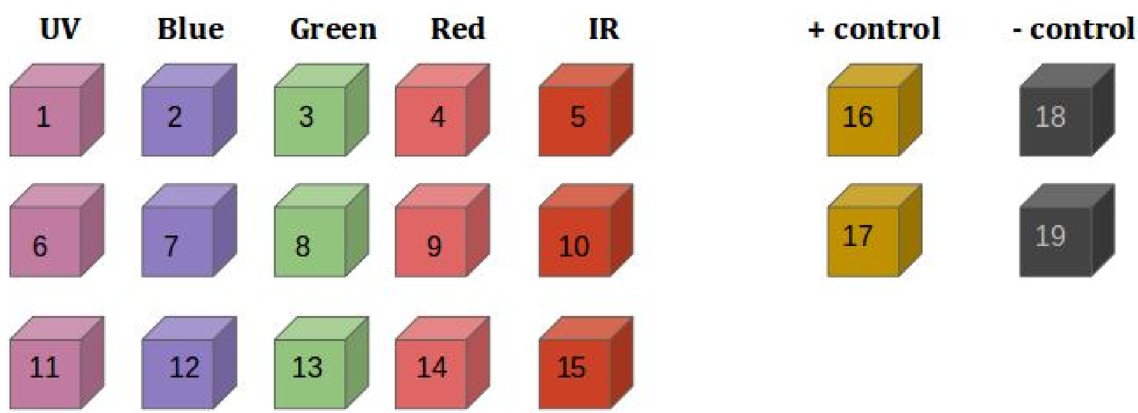
- Biosensor:
  - We measure the size of the plants and direction of lentils sprouts.
  - From this image we do the data analysis:
    - We count how many sprouts have grown towards the light
    - We do a % of sprouts that have moved (out of the total) and put it in a graph in relation with the wavelength.
- Electronic sensor:
  - The caption of those wavelengths will be also added in a graph in relation to the intensity.

18-01-2017

We went to buy the wood today.
We started cutting the wood for the boxes using 'plywood 4mm' setting (for the 5mm thick wood).
We used a plan, as said before, made with makecage (PJ)
After five boxes we realized that the rest of the slabs were curved, so we couldn't use them to do the other boxes.
We changed the wood: we took the 3mm thick. So to continue we changed the print-settings to 'plywood 3mm' and lowering the speed of 10%.

We assembled the boxes with hot glue and then recovered them with gray opac scotch (in order not to let the light pass). The face with the hole was not attached to the rest of the box (to have an entrance to put the lentils in).

The boxes are numerated from one to nineteen (being one to five the ones that are made out of the 5mm wood):



For what comes to LEDs, we try to get them since yesterday, but it is being complicated for us to find the ones we need in the stores: either they are out of stock either we have to buy a lot of them. We decided to order on Radiospare the next LEDs:

- LED infrarouge, Kodenshi, KEL3894S, Traversant, 940 nm, 20mW/sr
- LED, Traversant 3 x 4.6mm, Kingbright, Rouge, 660 nm, 3 mcd, 60°, 3 mm (T-1)
- LED, Traversant 5 x 8.6mm, Kingbright, Vert, 568 nm, 20 mcd, 60°, 5 mm (T-1 3/4)
- LED, Traversant 5 x 8.6mm, Kingbright, Bleu, 455 nm, 60 mcd, 60°, 5 mm (T-1 3/4)
- LED UV, UV3TZ-390-30 Traversant, 390 nm, 30 °

19-01-2017

We finished to scotch the boxes, tried the alimentation for the LEDs and try to understand the RGB sensor TCS3200.

Alimentation of the LEDs:

We need three leds of the same color and with the same intensity five times (for our replicas), so we need to think of a way to do it.

We decided to put them in series so they will have the same intensity.

Since one LED needs about 2,5 V; for three LEDs we need 7,5 V.

First we thought to try with a power transformer that would get the electricity from the current.

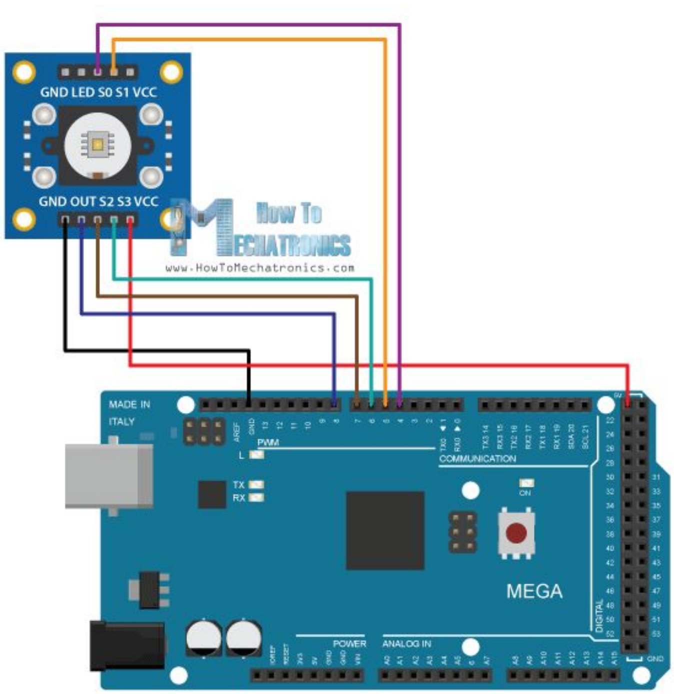
We cut the end of the cable, separate positive and negative cables and tried to power three LEDs.

We realized that the transformer gave only 5V to the LEDs and we needed more. We have to think in a new way to light them.

RGB Sensor:

At the beginning we wanted to use the Adafruit light sensor TSL2561, but finally we changed into a **TCS3200.** We decided to do so because we understood that TSL2561 didn't suit our purpose.

TCS3200 is a color light-to-frequency converter, that is programmable with Arduino. We've decided to work on this sensor because of its capacity to detect the color of the light.



Source of the figure

We used pin 4, 5, 6, 7, 8 instead of 2, 3, 4, 5, . Except this, we used the same connections.

<u>Material:</u>
Arduino UNO + USB cable
TCS3200 Color sensor
7 wires

<u>Connections:</u>

TCS3200 Color sensor	Arduino UNO
GND	Ground
VCC	5V
S0	4
S1	5
S2	6
S3	7
OUT	8

<u>Code</u>
Thanks to “BeyondLights” Team to let us their code and to Franck and Kevin to help us make it work.

```
/*    Arduino Color Sensing Tutorial
      *
      *  by Dejan Nedelkovski, www.HowToMechatronics.com
      *
      */

      #define S0 4
      #define S1 5
      #define S2 6
      #define S3 7
      #define sensorOut 8
      int frequency = 0;
      void setup() {
      pinMode(S0, OUTPUT);
      pinMode(S1, OUTPUT);
      pinMode(S2, OUTPUT);
      pinMode(S3, OUTPUT);
      pinMode(sensorOut, INPUT);

      // Setting frequency-scaling to 20%
```

```
digitalWrite(S0,HIGH);
digitalWrite(S1,LOW);

Serial.begin(9600);
}
void loop() {
// Setting red filtered photodiodes to be read
digitalWrite(S2,LOW);
digitalWrite(S3,LOW);

// Reading the output frequency
frequency = pulseIn(sensorOut, LOW);

// Printing the value on the serial monitor
Serial.print("R= ");//printing name
Serial.print(frequency);//printing RED color frequency
Serial.print("  ");
delay(100);

// Setting Green filtered photodiodes to be read
digitalWrite(S2,HIGH);
digitalWrite(S3,HIGH);

// Reading the output frequency
frequency = pulseIn(sensorOut, LOW);

// Printing the value on the serial monitor
Serial.print("G= ");//printing name
Serial.print(frequency);//printing RED color frequency
Serial.print("  ");
delay(100);

// Setting Blue filtered photodiodes to be read
digitalWrite(S2,LOW);
digitalWrite(S3,HIGH);

// Reading the output frequency
frequency = pulseIn(sensorOut, LOW);

// Printing the value on the serial monitor
Serial.print("B= ");//printing name
Serial.print(frequency);//printing RED color frequency
Serial.print("  ");
delay(1000);
}
```

20-01-2017

Today we fished the Arduino code. We found out that what yesterday's code showed were intensities of red, green and blue instead of RGB values.

So we looked for a right code that gave us those RGB values. We also thought that because what sent the light to our sensor would be a LED, we should have to turn off the four LEDs of the sensor.

The final code (for the moment) is:

```
/*     Arduino Color Sensing Tutorial
       *
       *  by Dejan Nedelkovski, www.HowToMechatronics.com
       *
       */

       #define S0 4
       #define S1 5
       #define S2 6
       #define S3 7
       #define LED 13
       #define sensorOut 8
       int frequency = 0;
       void setup() {
       pinMode(S0, OUTPUT);
       pinMode(S1, OUTPUT);
       pinMode(S2, OUTPUT);
       pinMode(S3, OUTPUT);
       pinMode(LED, OUTPUT);
       pinMode(sensorOut, INPUT);

       // Setting frequency-scaling to 20%
       digitalWrite(S0,HIGH);
       digitalWrite(S1,LOW);


       Serial.begin(9600);
       }
       void loop() {
       // Setting red filtered photodiodes to be read
       digitalWrite(S2,LOW);
       digitalWrite(S3,LOW);

       // Reading the output frequency
       frequency = pulseIn(sensorOut, LOW);

       // Printing the value on the serial monitor
       Serial.print("R= ");//printing name
       Serial.print(frequency);//printing RED color frequency
```

```
Serial.print("  ");
delay(100);

// Setting Green filtered photodiodes to be read
digitalWrite(S2,HIGH);
digitalWrite(S3,HIGH);

// Reading the output frequency
frequency = pulseIn(sensorOut, LOW);

// Printing the value on the serial monitor
Serial.print("G= ");//printing name
Serial.print(frequency);//printing RED color frequency
Serial.print("  ");
delay(100);

// Setting Blue filtered photodiodes to be read
digitalWrite(S2,LOW);
digitalWrite(S3,HIGH);
digitalWrite(LED,LOW);

// Reading the output frequency
frequency = pulseIn(sensorOut, LOW);

// Printing the value on the serial monitor
Serial.print("B= ");//printing name
Serial.print(frequency);//printing RED color frequency
Serial.print("  ");
delay(1000);
}
```

Today, and because we still don't have the leds, we put the control lentils on their cages and run the experiments to have data this afternoon for the Data Analysis class. We've decided to let the experiment run for 18 hours.

Finally, we've also decided not to count how many sprouts turned to the light (since they've already germinated and grown) but we decided to measure their size at the beginning and the end to be able to calculate how much did they grow in relation with their exposure to light.

We put 9 lentils in every box (taking in account that the hole should be at the top of one of the sides) and measured them before putting them on their place. The boxes had humid cotton on the bottom to allow lentils grow better. We've done a table where we can see **lentils' size** on **each box**, their **position**, **condition** as well as at which **hour** the boxes were closed:

Box	**Light**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Time**
16 ctrl +	Sunlight	7.1	3.5	3	4.5	2.3	1.7	2.6	1.6	2	13:59
17 ctrl +	Sunlight	3.8	1.9	3.4	3.7	3.5	2.2	4	3.2	2.2	14:04
18 ctrl -	None	4.2	2.4	3.7	3.3	5	1.9	2.2	2.7	2	14:15

19 ctrl -	None	4.6	1.2	2.6	2.2	3.1	2.7	3.5	2.6	4.9	14:18

We've also welded the cables on the RGB color sensor now that we know that our code works. We've labeled each cable with were it will be connected on the arduino and then assembled all of them with a plastic link. So they will be all of them in a limited amount of space and will be able to pass through a hole in the boxes (that we will do in order to be able to put the sensor inside and close the box).

Once all this setup was prepared, the sensor stopped working (it showed all the time the same values, even when we lighted the LEDs -one of them didn't work-). We thought it would be the arduino, so we changed it. The sensor continued without working. Tomorrow we will try with the one of "BeyondLights" Team.

Today we've also received our LEDs! We've got an UV LED too, we won't use the Crosslinker. So we did its installation:
For each LED we watched the datasheet to find the forward current and the forward voltage. We gathered the data and used a Led calculator (link here) to determine the resistance to add and to draw the electrical circuit

LED	Voltage	Intensity	Resistance
infrared	1.5 V	50 mA	10 Ohm
red	2.3 V	25 mA	2 : 18 than 1 : 120
green	2.2 - 2.5V	20 mA	2 : 22 than 1 : 150
blue	4 V	20 mA	1 : 56 than 1 : 56 than 1 : 56
uv	3.8 V	20 mA	1 : 68 than 1: 68 than 1 : 68

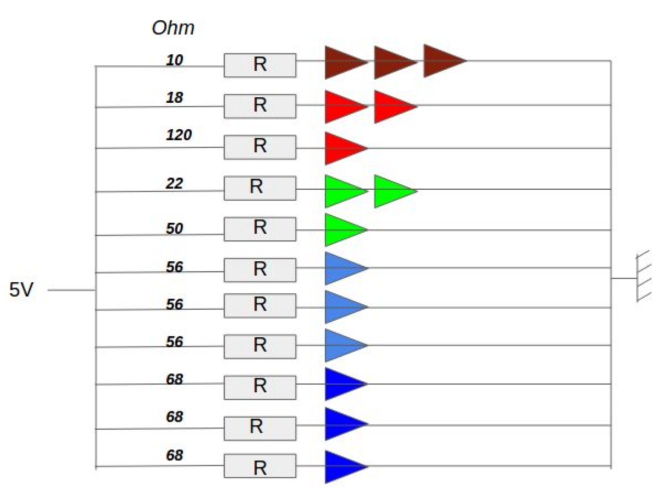


Figure of the electrical installation :
Depending of the voltage maximum of the LED, you can put successive LED on one branch.

We used 2 sources of electricity for practical reasons (too much cables for one PCB board) : 2 identical chargers producing 5V and 2.1 A. They were USB cables, we cut the end and weld it to the ends of the circuit (figure of electrical installation) thanks to the PCB boards.

Once the connections and welding made, we attached the LEDs to the boxes: we did it thanks to opac scotch, that we scotched on both sides of the cover in order to minimize the amount of light that could enter the box. We've also put a little scotch in one of the terminals of every LED to prevent contact failures.

Once all the covers were done, we started putting lentils on the boxes. As done with the controls, we measured and put 9 lentils par box (except box nine that has eight lentils) and looked that the LED would be on the top of one of the sides. Once the lentilles were on all the boxes, we lighted the LEDs and annotate this hour as the beginning of the experience. The resultant data table was:

Box	Light	1	2	3	4	5	6	7	8	9	Time
1	UV	4.8	2	4.5	1.9	3	4	5.6	3.7	2.3	23:13
2	Blue	3.5	5.6	5.3	2	5.7	3.4	4.6	2.5	2.1	23:13
3	Green	7.5	4.6	2.1	4.3	2.8	4	6	3.3	2.3	23:13
4	Red	5.9	4.6	3	3.5	5	4	2	4.6	3.6	23:13
5	IR	4.4	3.2	3.8	5.2	4.2	4.1	2.1	6.8	2.1	23:13
6	UV	2.7	4.8	3.8	2.7	5.2	5.3	1.5	2.6	4	23:13
7	Blue	3.4	2.7	4.1	3.6	2.1	2.5	4.8	5.2	2.7	23:13
8	Green	1.9	5.1	5.6	3.5	4.4	3.2	1.5	3.5	2.4	23:13
9	Red	2.1	4.1	5.6	6.1	3.9	2.7	5.3	3.5		23:13
10	IR	4.1	3.8	5.2	5.4	2	4.3	4.8	4.2	2.6	23:13
11	UV	3	1	5.7	3.9	3.6	4.8	3.1	5.7	3	23:13
12	Blue	5	5.1	3.2	3.5	3	2.2	5.7	4	3.6	23:13
13	Green	4.9	4.7	1.9	2.5	y5	4.4	4	4.7	3.5	23:13
14	Red	4.6	3.8	6.3	3.9	3.4	4.2	4.6	4.2	2.9	23:13
15	IR	4.1	2.7	1.7	2.4	4.3	2.3	5.7	3	1.9	23:13

We used a ruler to measure the size of each lentil.

21-01-2017

Today we've started at 8 a.m. with measurements: the length of lentils sprouts after 18h of controls running.

The resultant table was:

Box	Light	1	2	3	4	5	6	7	8	9	Time
16 ctrl +	Sunlight	7.1	4.9	3.9	6	3.6	1.8	2.6	2.2	2.2	8:10
17 ctrl +	Sunlight	5.1	3.8	4.5	3.9	4	2.3	4.2	4	2.5	8:12
18 ctrl -	None	4.6	2.7	4.1	4.3	2.5	2.4	2.2	3	2.1	8:17
19 ctrl -	None	4.6	1.6	3.5	2.5	3.4	3	3.1	2.8	4.8	8:21

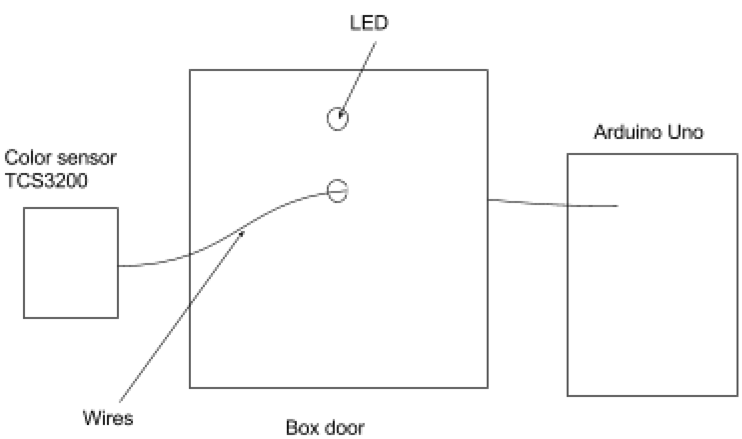
We measured others lentils at 5 p.m. after letting them grow during 19 hours.

Result table:

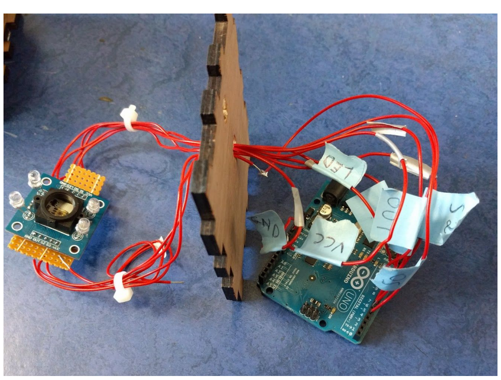
Box	Light	1	2	3	4	5	6	7	8	9	Time	Comments
16 ctrl +	Sunlight	7.1	4.9	3.9	6	3.6	1.8	2.6	2.2	2.2	8:10	
17 ctrl +	Sunlight	5.1	3.8	4.5	3.9	4	2.3	4.2	4	2.5	8:12	
18 ctrl -	None	4.6	2.47	4.1	4.3	2.5	2.4	2.2	3	2.1	8:17	
19 ctrl -	None	.6	1.6	3.5	2.5	3.4	3	3.1	2.8	4.8	8:21	
1	UV	5.5	2.4	5	2	3.8	6	6.8	3.4	2.3	17:00	
2	Blue	6	4.5	6.5	2.5	5.7	3.5	5.2	2.6	2.5	17:00	Error counting. Shift somewhere.
3	Green	5.8	2	4	4.8	8.8	3	5.1	6.3	3	17:00	It felt, the length doesn't correspond to the number.
4	Red	7.5	6.5	4	4.2	5.1	4.1	2.2	5	3.9	17:00	
5	IR	5.7	3.6	4.5	5.5	5.5	5.1	2.1	8.1	2.4	17:00	
6	UV	3.5	6.1	4	3.5	5	6	1.5	3	4	17:00	
7	Blue	4	1.6	4.2	3.6	2.4	2.6	5.2	5.5	2.6	17:00	

8	Green	1.4	5.4	5.4	3.5	4	3.5	1.8	4	2.1	17:00	They are all elongated except the 7th.
9	Red	2.3	4.5	3	7	6.5	4.3	3.8	6.3	4.7	17:00	
10	IR	4.3	4.6	5.7	5.2	2	4.2	5.5	5.5	3.1	17:00	
11	UV	1	4	4	3	3.2	3	6.7	6.2	4.5	17:00	It also felt, the length doesn't correspond to the lentil. Except the 9th.
12	Blue	6.3	5	3.5	4	3.9	2.7	5.9	4.3	3.8	17:00	
13	Green	5.8	5.4	2.5	2.5	5.5	5	5.1	4.7	4.7	17:00	
14	Red	5.3	4.3	6.6	4.4	3.6	5.1	6.6	4.7	3.5	17:00	
15	IR	4.4	2.9	2	2.3	6.4	2.3	7.3	2.7	2.4	17:00	

We measured sprouts by completely unfolding the stem (we didn't measured the roots length). Gray parts are lentils we didn't use for the data analysis (due to the reasons explicated on the extra column).



After that we gathered data for electronical sensor TCS3200. We put the sensor in the same box device than the lentils, sensor turned in direction of the hole in which we put the LED.



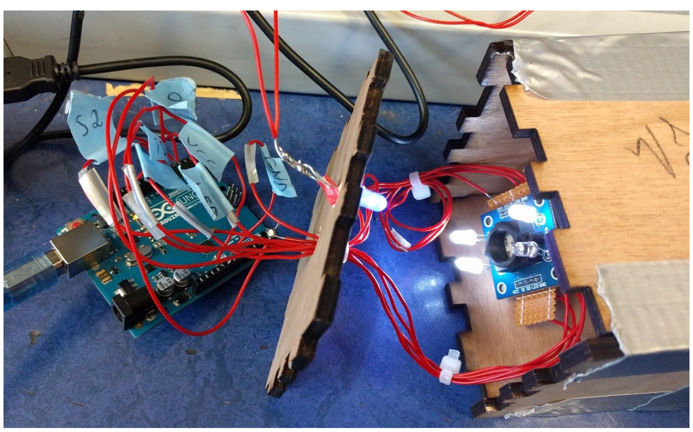
We've changed the code again. The following code was executed on the arduino UNO:

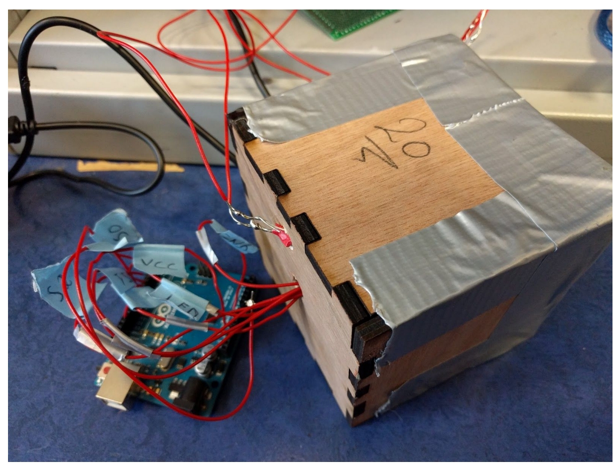
```
/*    Arduino Color Sensing Tutorial
      *
      *  by Dejan Nedelkovski, www.HowToMechatronics.com
      *
      */

      #define S0 4
      #define S1 5
      #define S2 6
      #define S3 7
      #define sensorOut 8
      int frequency = 0;
      void setup() {
      pinMode(S0, OUTPUT);
      pinMode(S1, OUTPUT);
      pinMode(S2, OUTPUT);
      pinMode(S3, OUTPUT);
      pinMode(sensorOut, INPUT);

      // Setting frequency-scaling to 20%
      digitalWrite(S0,HIGH);
      digitalWrite(S1,LOW);

      Serial.begin(9600);
      }
      void loop() {
      // Setting red filtered photodiodes to be read
      digitalWrite(S2,LOW);
      digitalWrite(S3,LOW);

      // Reading the output frequency
      frequency = pulseIn(sensorOut, LOW);

      // Printing the value on the serial monitor
      Serial.print("R= ");//printing name
      Serial.print(frequency);//printing RED color frequency
      Serial.print("  ");
      delay(100);

      // Setting Green filtered photodiodes to be read
      digitalWrite(S2,HIGH);
      digitalWrite(S3,HIGH);

      // Reading the output frequency
      frequency = pulseIn(sensorOut, LOW);

      // Printing the value on the serial monitor
      Serial.print("G= ");//printing name
      Serial.print(frequency);//printing RED color frequency
      Serial.print("  ");
```

```
        delay(100);

        // Setting Blue filtered photodiodes to be read
        digitalWrite(S2,LOW);
        digitalWrite(S3,HIGH);

        // Reading the output frequency
        frequency = pulseIn(sensorOut, LOW);

        // Printing the value on the serial monitor
        Serial.print("B= ");//printing name
        Serial.print(frequency);//printing RED color frequency
        Serial.print("  ");
        delay(1000);
        }
```

Data gathering:
We put the LED we wanted to test on the hole, and then closed the box (with the sensor inside). We protected the box from exterior light by hiding it under a blanket. The LED was switched off at the beginning. We opened the monitor for the arduino card and started the data gathering. After 3 seconds we switched on the LED. After 5 seconds we cut the power again, waited 3 seconds and stopped the data gathering by unplugging the USB arduino wire.

Then, to gather this data, we copied what the monitor showed and then pasted it in a document named 'condition'.txt (where condition could be either 'uv', 'blue', 'green', 'red', 'ir' or 'blanc' (the last one for white light that we did with a lamp).
For each LED, 3 replicates were made in the same conditions.





22-10-2016

Today we confectioned python code for the biosensor data analysis and tried to understand the data collected with the electrical sensor.
About the electrical sensor : we collected the RGB components under different wavelength light. These values were recorded with and without a "mapping" system of Arduino (map function allows to define an interval in which the experimental values stay).

```
nomapR= 1146  R= 194  nomapG= 2036  G= 346  nomapB= 1982  B= 336
nomapR= 1139  R= 193  nomapG= 2035  G= 345  nomapB= 1982  B= 336
nomapR= 1145  R= 194  nomapG= 2035  G= 345  nomapB= 1983  B= 337
nomapR= 1147  R= 194  nomapG= 2035  G= 345  nomapB= 1981  B= 336
nomapR= 1146  R= 194  nomapG= 2036  G= 346  nomapB= 1980  B= 336
nomapR= 1145  R= 194  nomapG= 2036  G= 346  nomapB= 1982  B= 336
nomapR= 1054  R= 179  nomapG= 1840  G= 312  nomapB= 1791  B= 304
nomapR= 1056  R= 179  nomapG= 1842  G= 313  nomapB= 1793  B= 304
nomapR= 1057  R= 179  nomapG= 1844  G= 313  nomapB= 1800  B= 306
nomapR= 1057  R= 179  nomapG= 1844  G= 313  nomapB= 1795  B= 305
nomapR= 1057  R= 179  nomapG= 1845  G= 313  nomapB= 1796  B= 305
nomapR= 1057  R= 179  nomapG= 1846  G= 313  nomapB= 1803  B= 306
nomapR= 1059  R= 180  nomapG= 1854  G= 315  nomapB= 1798  B= 305
nomapR= 1059  R= 180  nomapG= 1848  G= 314  nomapB= 1799  B= 305
nomapR= 1060  R= 180  nomapG= 1847  G= 313  nomapB= 1798  B= 305
nomapR= 1059  R= 180  nomapG= 1847  G= 313  nomapB= 1805  B= 306
nomapR= 1060  R= 180  nomapG= 1854  G= 315  nomapB= 1798  B= 305
nomapR= 1060  R= 180  nomapG= 1856  G= 315  nomapB= 1800  B= 306
nomapR= 1061  R= 180  nomapG= 1851  G= 314  nomapB= 1806  B= 307
nomapR= 1060  R= 180  nomapG= 1850  G= 314  nomapB= 1800  B= 306
nomapR= 1054  R= 179  nomapG= 1851  G= 314  nomapB= 1801  B= 306
nomapR= 1061  R= 180  nomapG= 1852  G= 314  nomapB= 1801  B= 306
nomapR= 1061  R= 180  nomapG= 1852  G= 314  nomapB= 1801  B= 306
```

Table of raw data obtained from the electrical sensor :
Data were recorded under the following form :

- nomapX (where X = R, B or B) : without mapping system. These are raw data.
- X is the value after defining the interval. We defined the interval between 0-255 but values exceeded 255. This problem has not been solved for now.

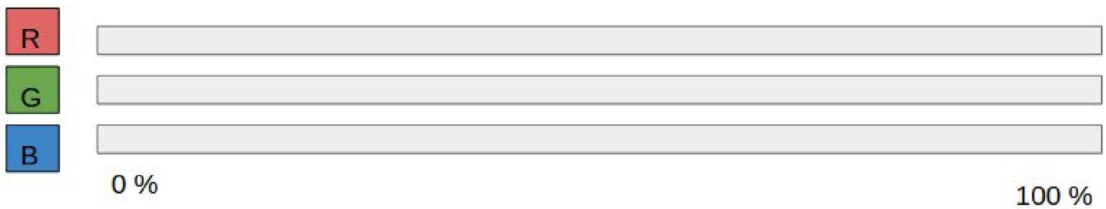
We calculated the mean of RGB and nomapRGB values. Each time there were 3 tries.

	R	G	B	R	G	B
NOIR	195	348	340	1153	2046	1998
BLEU 1	192	327	269	1129	1928	1587
BLEU 2	184	319	266	1088	1880	1570
BLEU 3	187	328	297	1092	1935	1738
VERT 1	196	347	339	1156	2044	1997
VERT 2	188	333	338	1110	1964	1990
VERT 3	189	340	338	1115	1998	1990
IR 1	20	24	24	115	144	145
IR 2	19	20	20	112	124	124
IR 3	23	27	27	145	156	160
ROUGE	159	312	300	936	1832	1774
ROUGE2	149	300	294	876	1770	1725
ROUGE3	140	287	281	823	1700	1665
UV1	134	290	145	794	1705	860
UV2	85	210	70	500	1270	430

UV3	137	289	157	812	1705	930
BLANC	155	265	266			

UV2 values are obviously aberrant. They were not taking into account.

The biggest problem we faced was to give a meaning to these values so we focused on the values obtained after a mapping. Because the sensor is working absorbing the light, RGB components are high and will be used to define the upper bound of the intervale. Inversely, the white should be the lowest values so we defined them at the lower boundary. Every wavelength will be considered as a ratio :



Intensity according to relative gap between black and white

Calculations :

For each value : $\frac{Xi - Xwhite}{Xblack - Wwhite} * 100$ with : Xi = value of R,G,B obtained

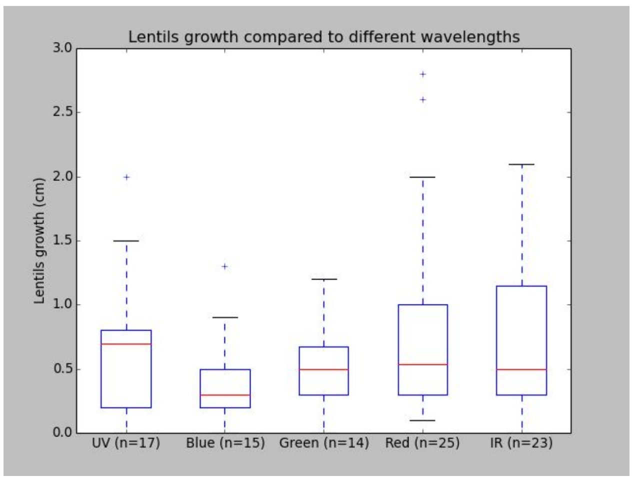
	Absorption in pourcentage			
Relative gap	R	V	B	
White	**100**	**100**	**100**	
Black	**0**	**0**	**0**	
Blue	**100**	**70**	**14**	
Green	**100**	**100**	**100**	Green is not recognised
Red	**0**	**47**	**48**	
IR				
UV	**<0**	**33**	**<0**	

We noticed that blue is not recognised, it appears like black. IR and UV displays values that can not be interpreted with this system of ratio (both have values below white).
Red and Green are recognised but in differents way that are hard to quantify.

For the biosensor data analysis, we've decided to use boxplots to represent our data. We've represented for how much every sprout have grown.

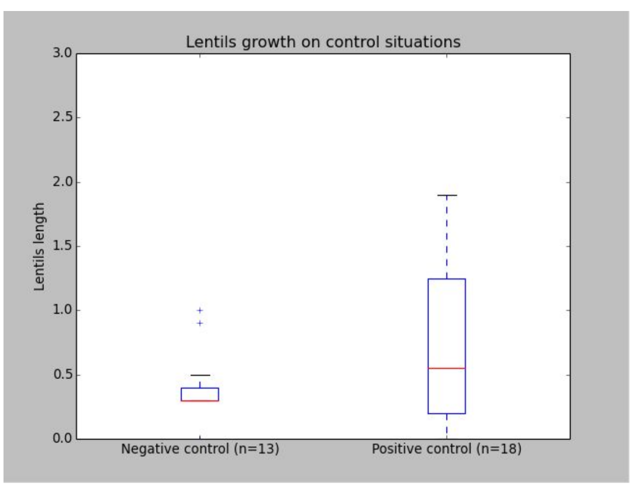
We've made a boxplot for each condition. We've found negative values that we decided to delete considering them aberrants (cf. python code).

23-10-2016



Today we've finished both graphs. The ones for the biosensor and the ones for the RGB sensor:

On those boxplots we can see that the means do a courbe with two maximums (situated on the domain of the UVs and of the Red - IR). Even though, the results on the Red, IR and Green are near (or almost equal) the ones that we got on the positive control (sunlight). We could then extrapolate that what the lentil get the most from sunlight is Red and IR.
But we know that sunlight has UVs, how does it come that lentils didn't got those UVs? Our other hypothesis would be that actually, as we put lentils on their cages at 14h and let them indoors, they got artificial light, that doesn't emit that much UVs.



What we also see is that the growth on the blue spectrum is close to the negative control, where lentils didn't have any light exposition. We can compare this with the data we got from the RGB sensor, that doesn't detect blue light either. This could be because maybe those blue LEDs don't light enough for the sensors to sense them.