

Interactions Homme-Machine

A series of horizontal lines in teal and light blue colors, with varying lengths and offsets, creating a layered effect across the middle of the slide.

7 – Toolkits Graphiques: C++ - Arduino + Processing

Plan du cours

1. Les Toolkits graphiques.
2. C++ / Processing + Arduino
3. Exemple d'IHM 1 – Projet 1
4. Exemple d'IHM 2 – Projet 2

Toolkits Graphiques

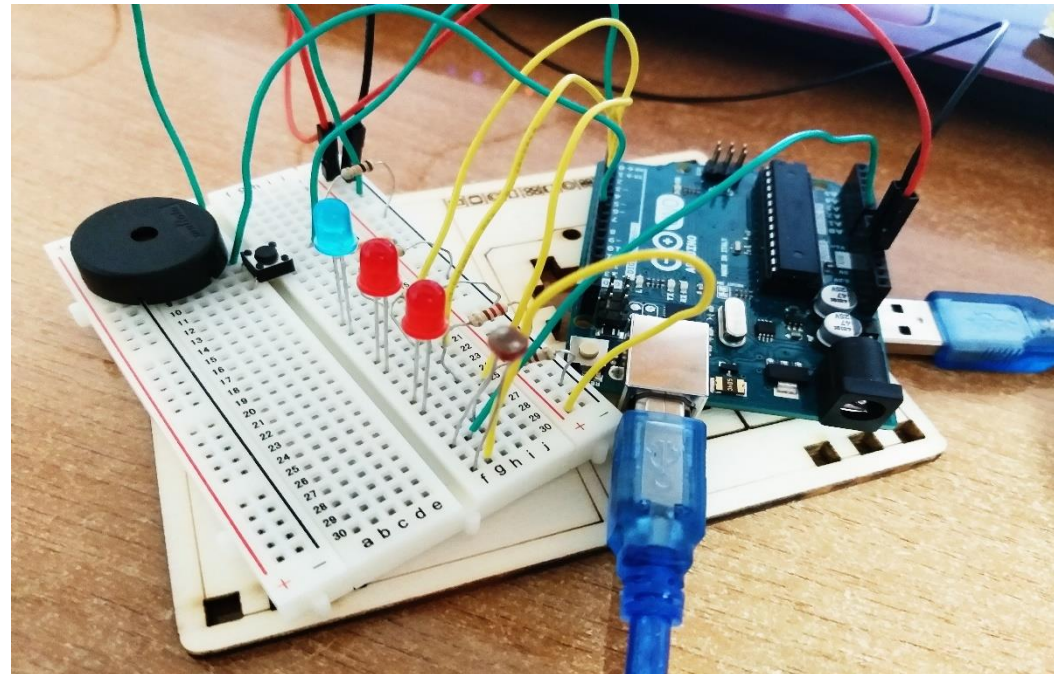
- Programmation d'IHM – Interactions et interfaces.
- = Bibliothèques permettant le développement et la réalisation des interfaces graphiques.
- Différents toolkits pour différents usages et langages de programmation :
 - ✓ Java – Swing
 - ✓ Python – Tkinter
 - ✓ GTK The Gimp Toolkit
 - ✓ Web – HTML CSS Bootstrap
 - ✓ Javascript - JQuery
 - ✓ **C++ - Processing / Arduino / Qt**
 - ✓ Etc.

Toolkits Graphiques

C++ - Processing / Arduino

Projet 1:

- Détection et Alarme via des LED (visuel), boutons poussoirs, et buzzer (sonore).
- Capteur photorésistance.

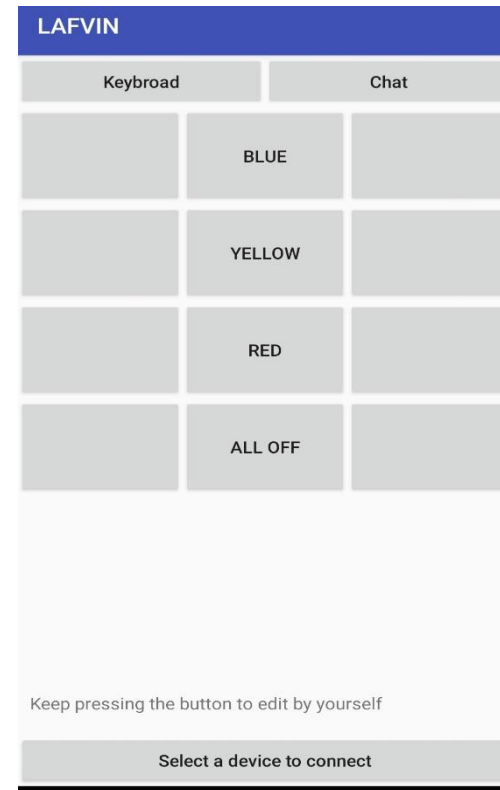
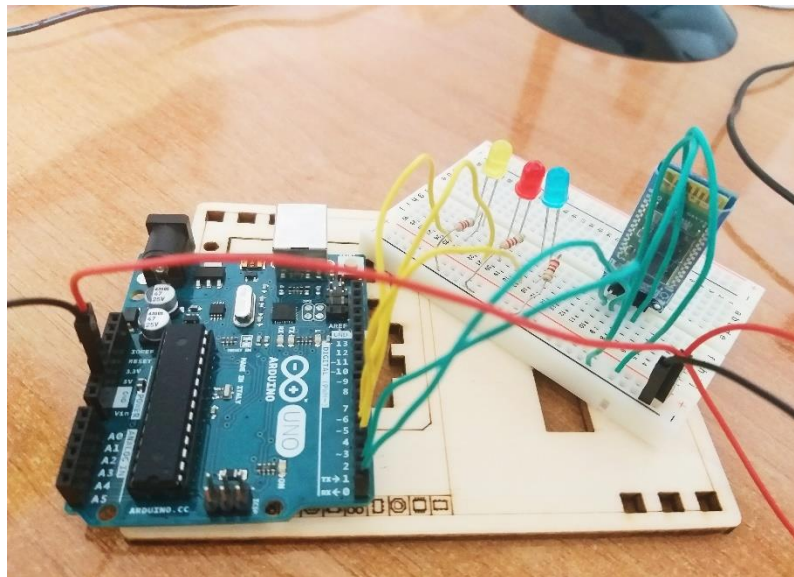
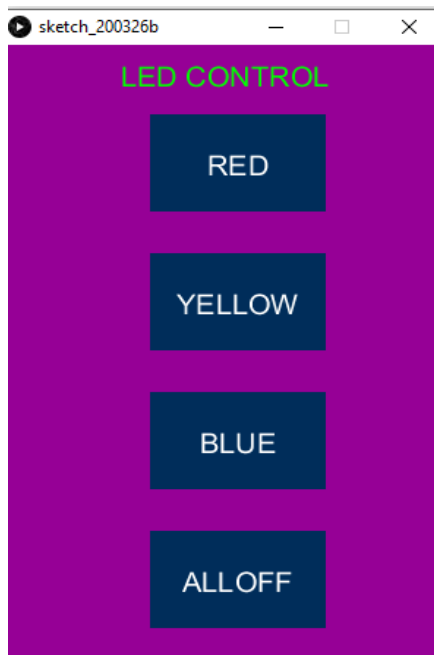


Toolkits Graphiques

C++ - Processing / Arduino

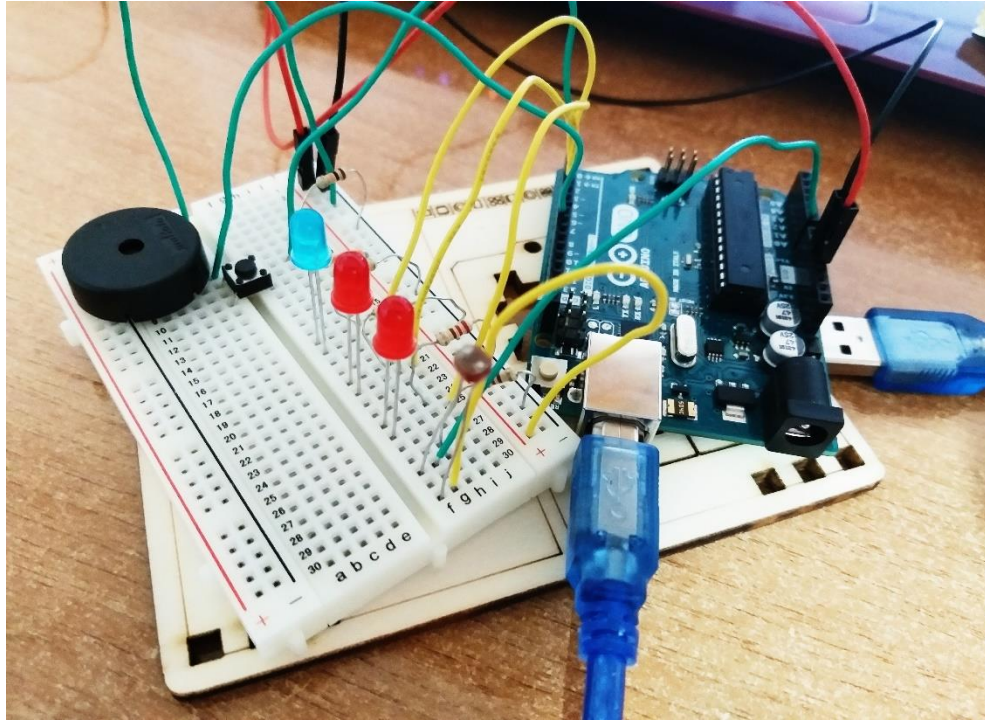
Projet 2:

- Contrôler des LEDs via des **GUI Desktop et Mobile** et communication **Bluetooth**.

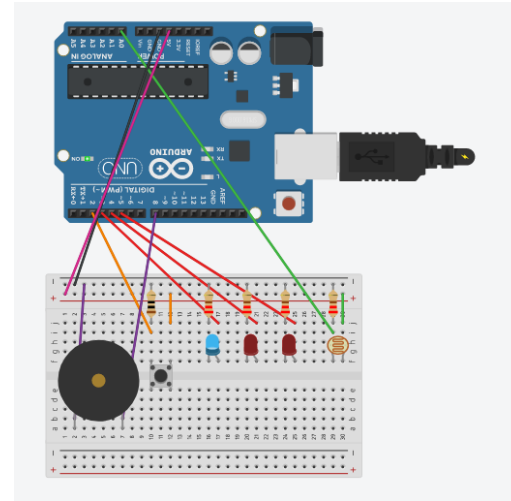


Exemple d'IHM 1 - Bouton, Son, et Visuel

- Circuit Diagram : <https://www.tinkercad.com/things/cORXW73sL5o>
- Code Source Project :
<https://github.com/GitTeaching/Exemple1ArduinoCoursIHM>
- Réalisation et Démo: Voir vidéo : <https://vimeo.com/412009232>
Mot de passe vidéo sur Vimeo : exemple1



Exemple d'IHM 1

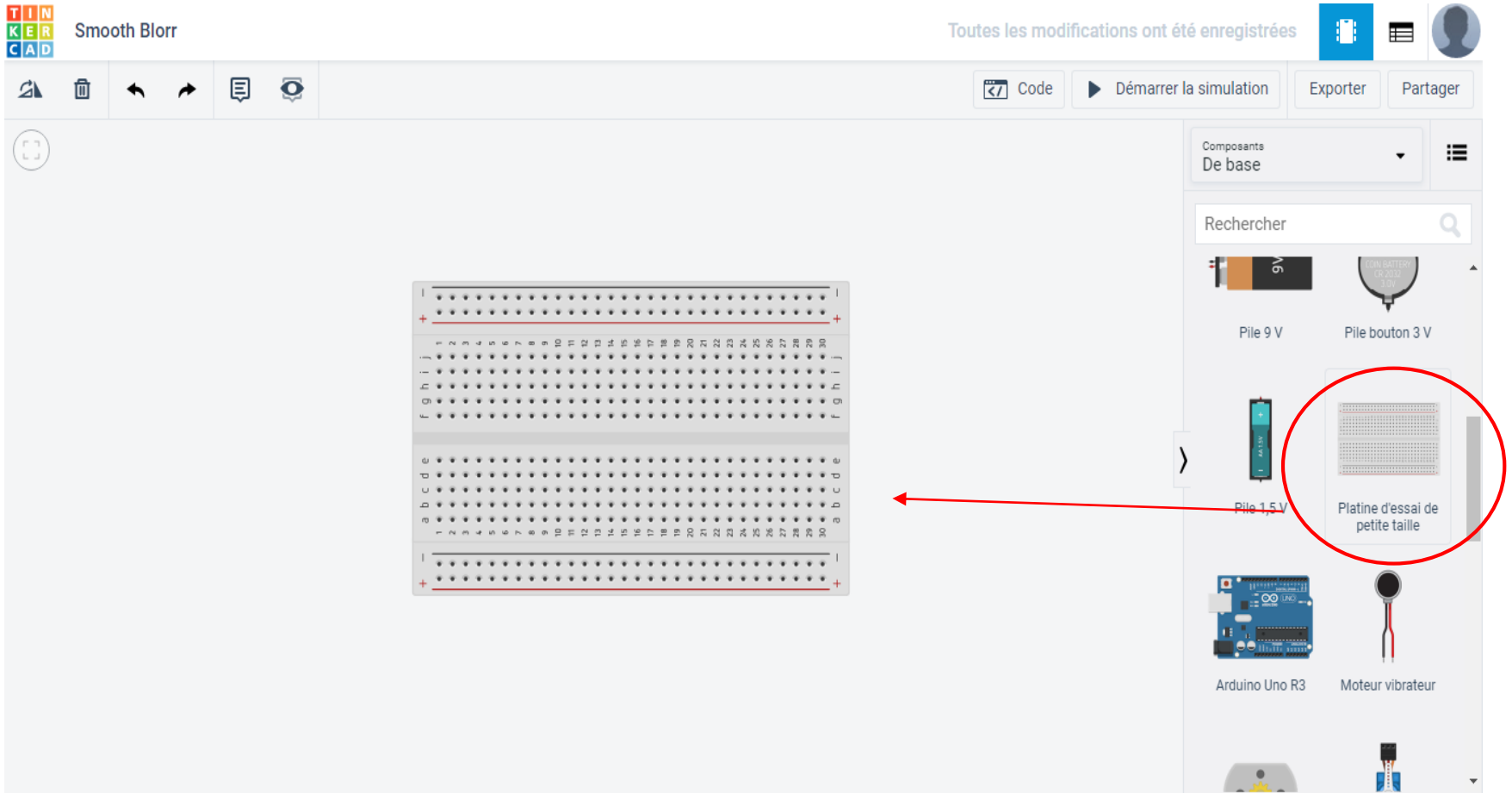


- Aller sur Tinkercad Circuits : <https://www.tinkercad.com/circuits>

Création et simulation gratuites des circuits électroniques. **Arduino Virtuel.**

- Cliquez sur S'inscrire (ou Connexion, si vous êtes déjà inscrit). Suivez le processus d'inscription jusqu'au bout.
- Puis, connectez-vous et sélectionnez la rubrique **Circuits**. Cliquez ensuite sur **Créer un Circuit**.
- Ajouter une breadboard (platine d'essai de petite taille) comme suit :

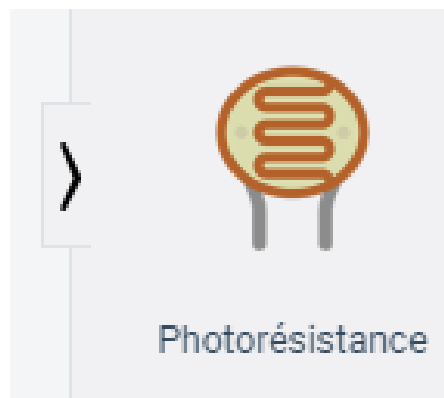
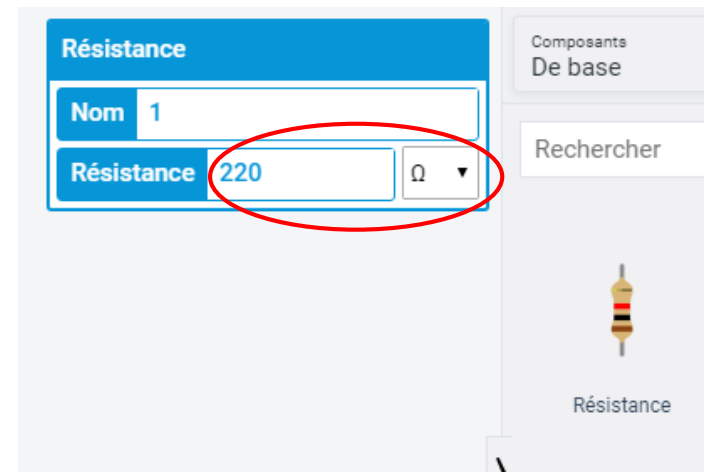
Exemple 1



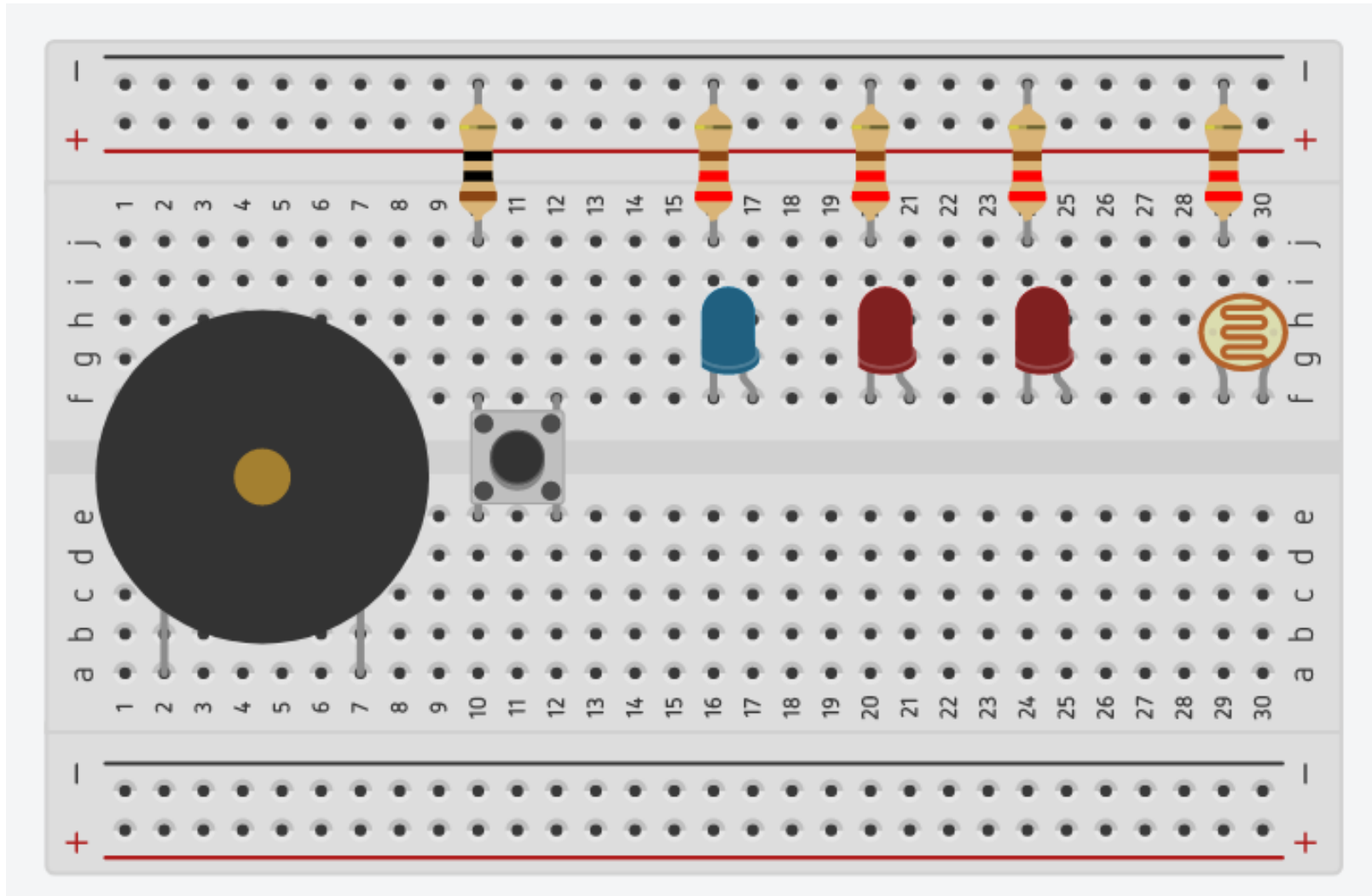
Exemple 1

Ajouter et positionner les composants suivants comme suit :

- 3 résistances 220 ohm
- 1 résistance 10 ohm
- 3 LED : 2 rouge et 1 bleu
- 1 switch (bouton poussoir)
- 1 piezo électrique
- 1 photorésistance

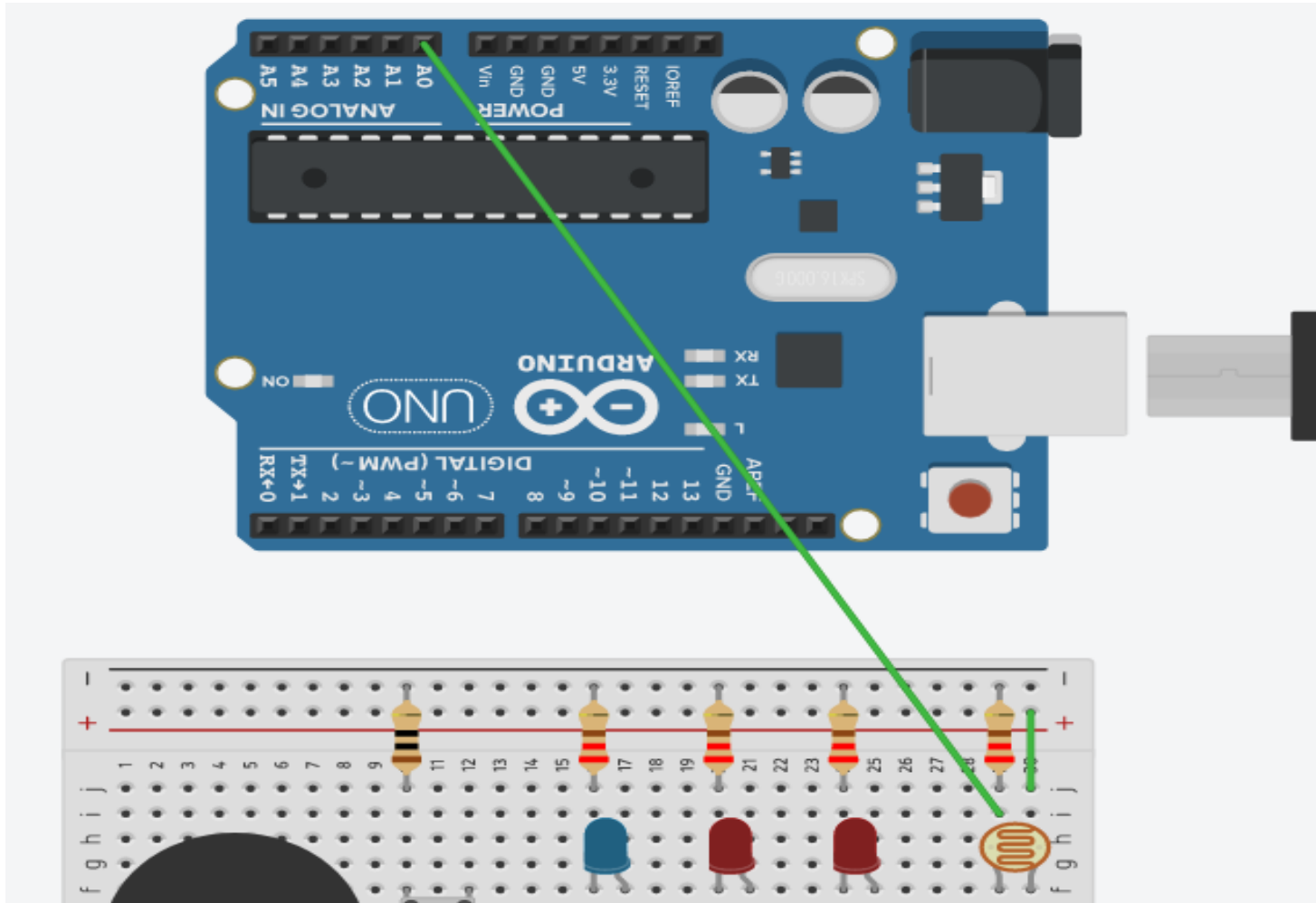


Exemple 1

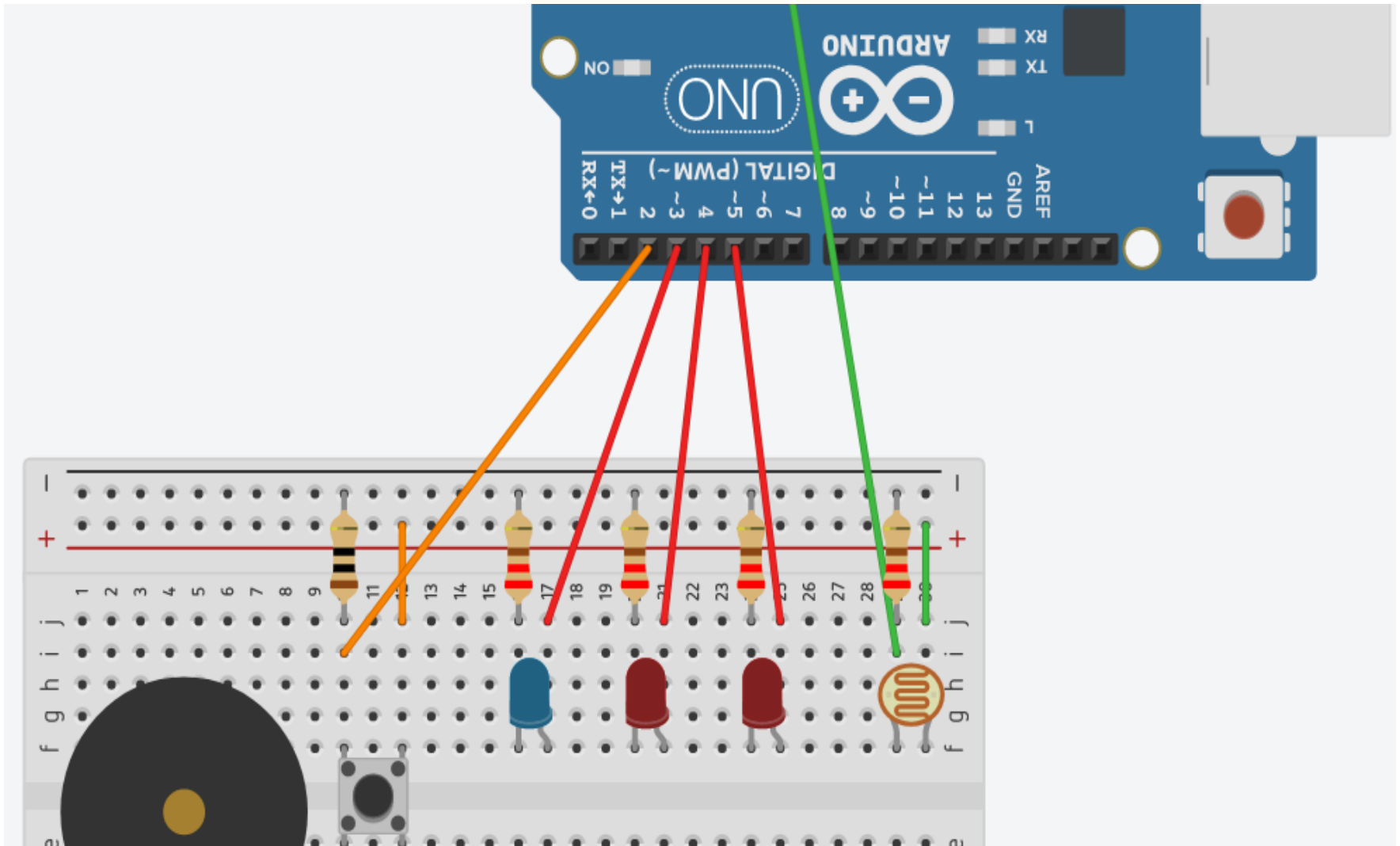


Exemple 1

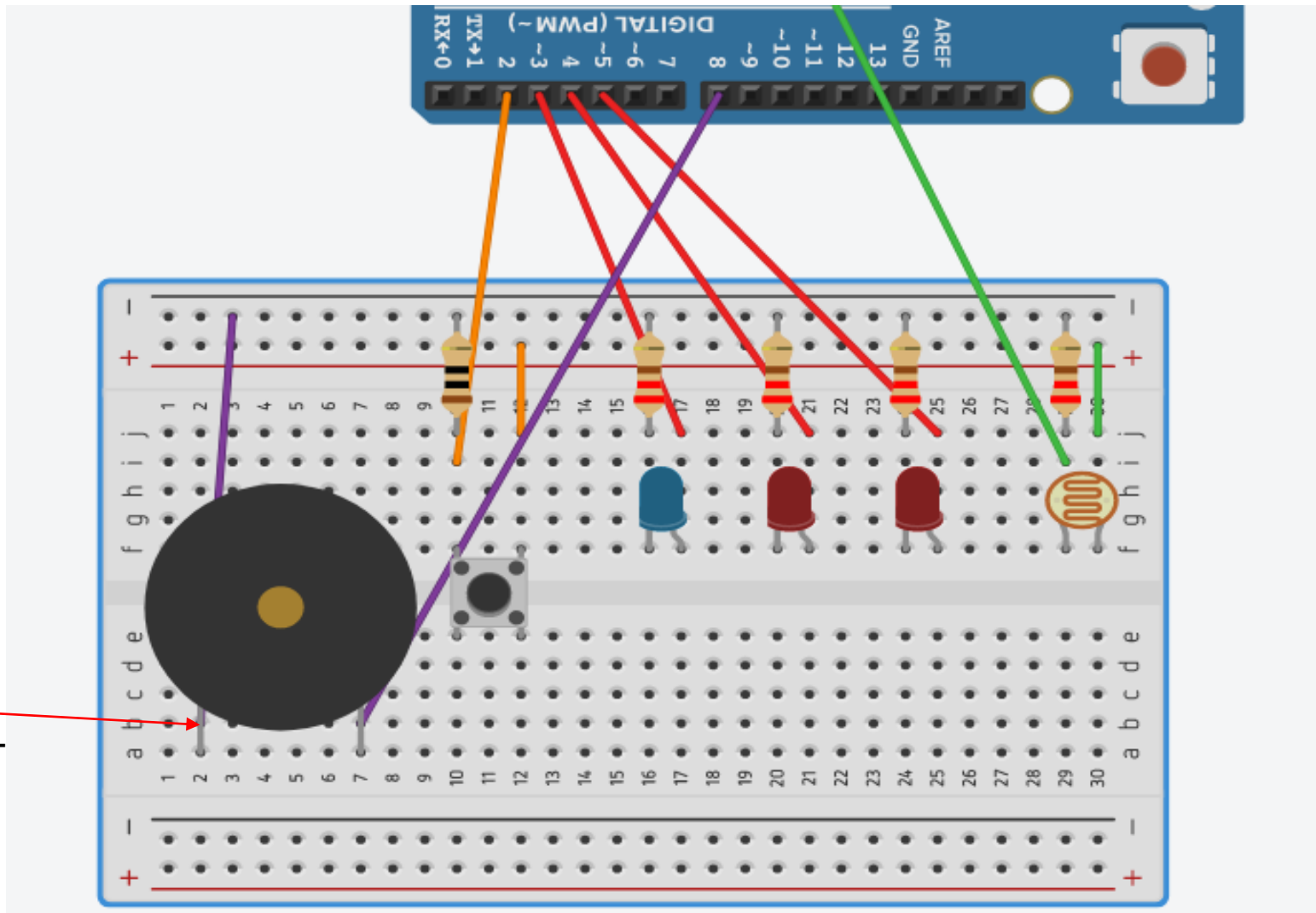
Câblage comme suit :



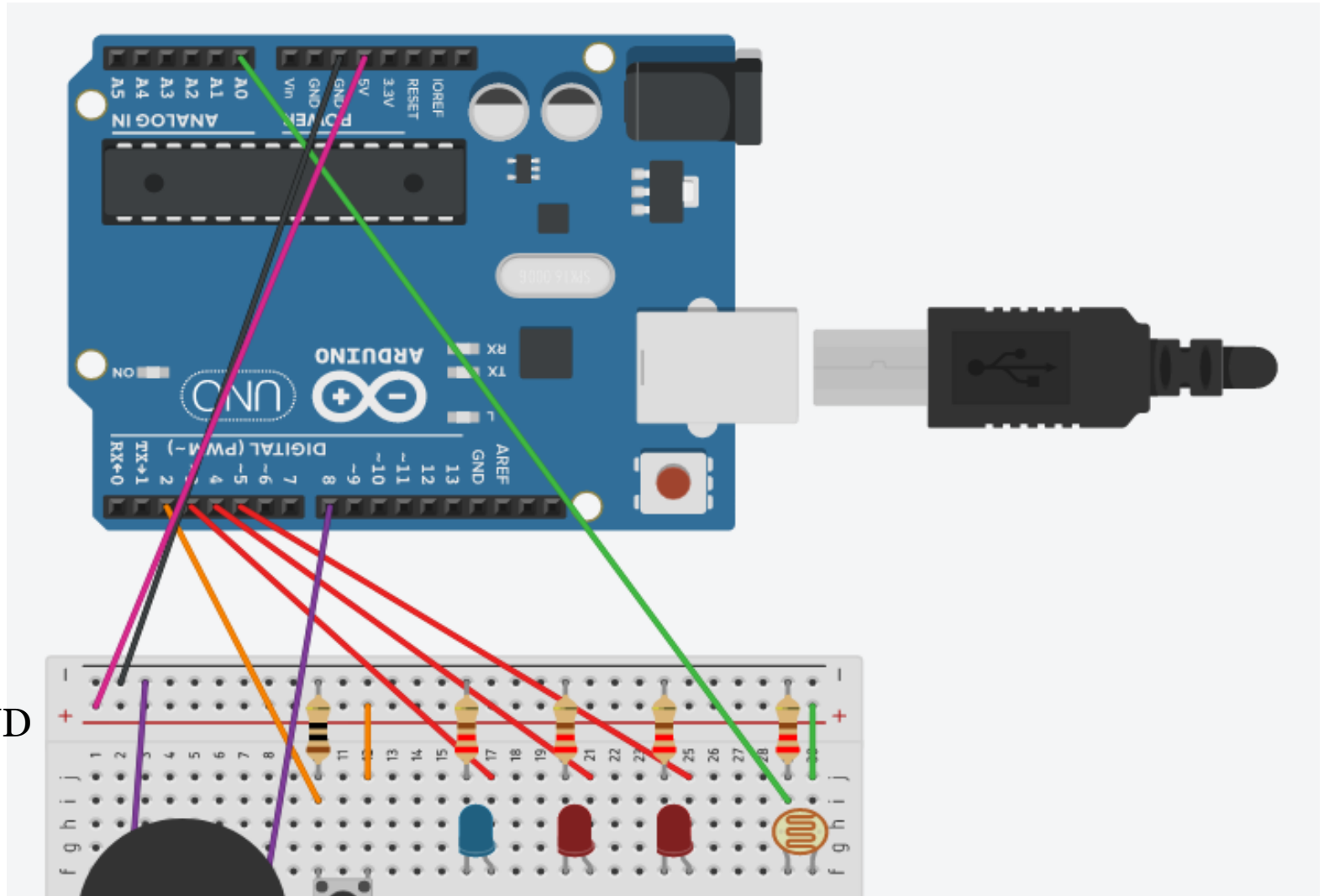
Example 1



Exemple 1



Exemple 1

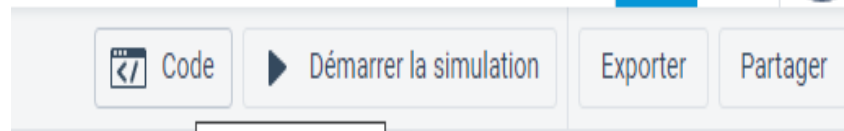


+ vers 5V
- Vers GND

Exemple 1

Programmer l'Arduino en chargeant le code comme suit :

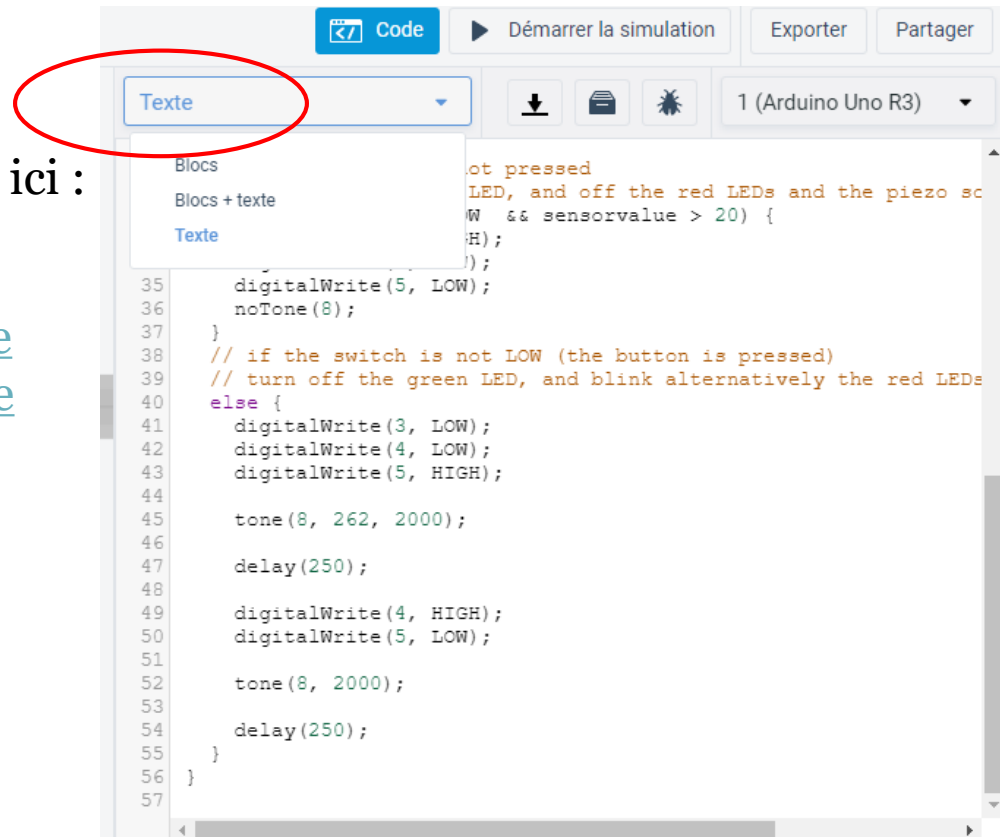
- Cliquer sur Code :



- Sélectionner **Texte** dans la liste

- Copier/Coller le code qui se trouve ici :

<https://github.com/GitTeaching/Exemple1ArduinoCoursIHM/blob/master/exemple1.ino>

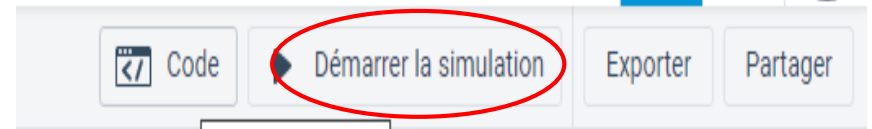


Exemple 1

Simulation de l'Arduino comme suit :

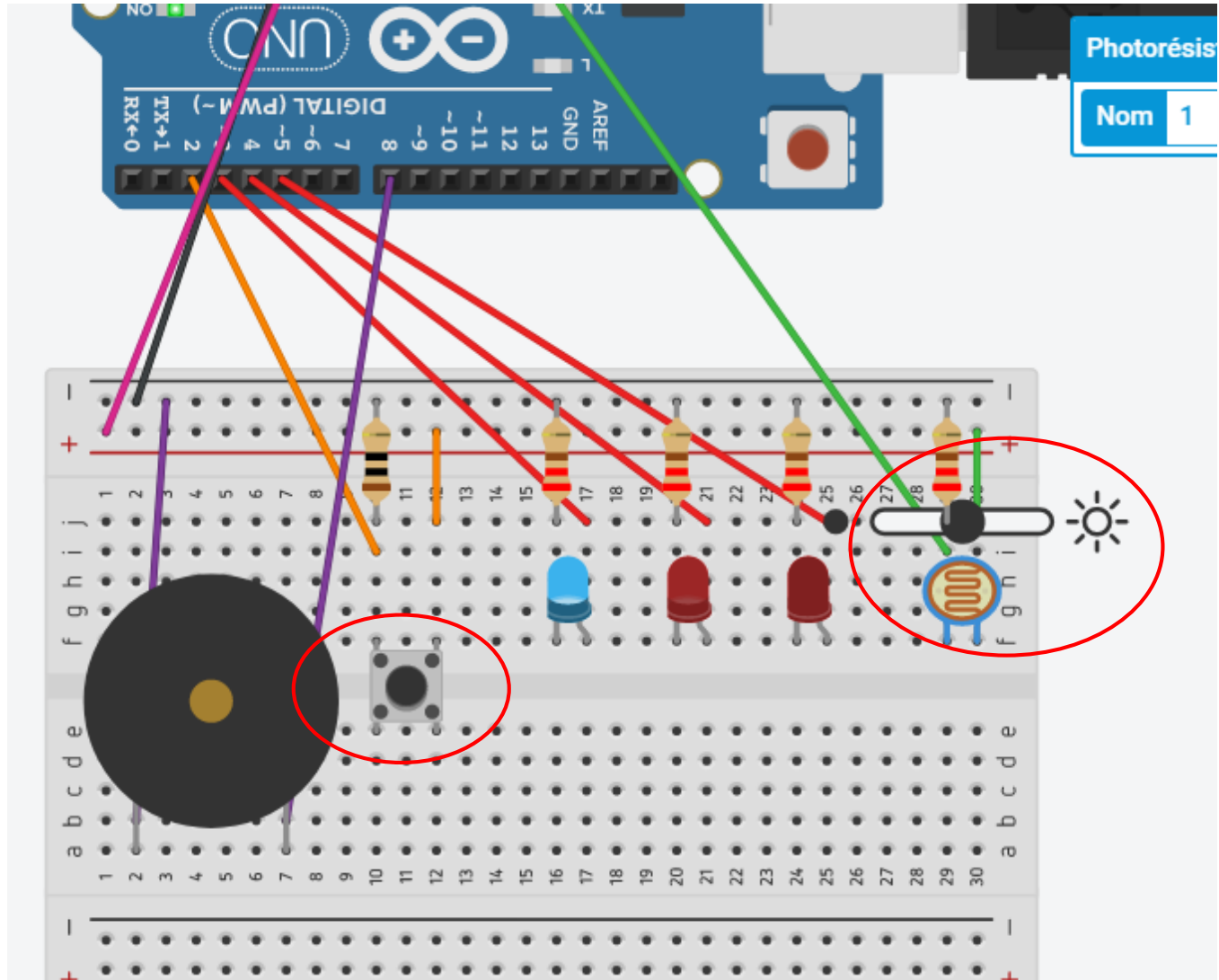
- Re Cliquer sur Code pour réduire son onglet

- Cliquer sur Démarrer la simulation :



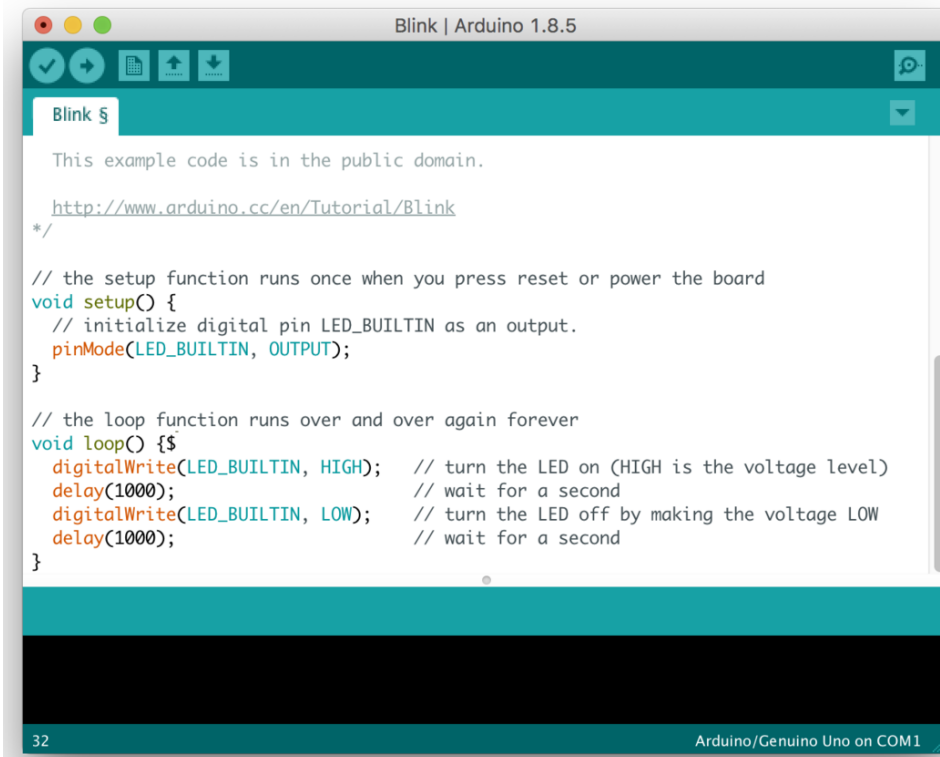
- Cliquer sur le photorésistance pour augmenter et diminuer la lumière ou sur le bouton switch pour actionner, comme suit :

Exemple 1



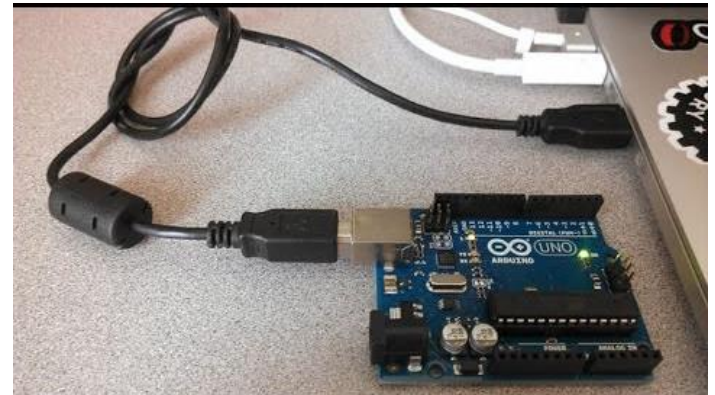
Exemple 1

- Dans le cas réel, la programmation de l'Arduino se fait en utilisant l'IDE Arduino.
- A télécharger d'ici : <https://www.arduino.cc/en/main/software>
- Le code est compilé puis téléversé vers l'Arduino déjà connecté via USB.



The screenshot shows the Arduino IDE 1.8.5 interface. The title bar reads "Blink | Arduino 1.8.5". The menu bar includes "File", "Edit", "Tools", and "Help". The toolbar contains icons for opening, saving, compiling, and uploading. The main text area displays the Blink example code, which is in the public domain. The code includes comments and function definitions for setup and loop. The status bar at the bottom indicates "32" and "Arduino/Genuino Uno on COM1".

```
Blink §  
  
This example code is in the public domain.  
  
http://www.arduino.cc/en/Tutorial/Blink  
*/  
  
// the setup function runs once when you press reset or power the board  
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000); // wait for a second  
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW  
  delay(1000); // wait for a second  
}
```



Exemple d'IHM 2 - GUI, Mobile, et Visuel

- Circuit Diagram : <https://www.tinkercad.com/things/etKq6SZxGe3>
- Code Source Project:
<https://github.com/GitTeaching/Exemple2ArduinoCoursIHM>
- Deux versions du projet :
 - GUI Desktop
 - GUI Mobile + Communication Bluetooth
- Réalisation et Démo : Voir vidéos:

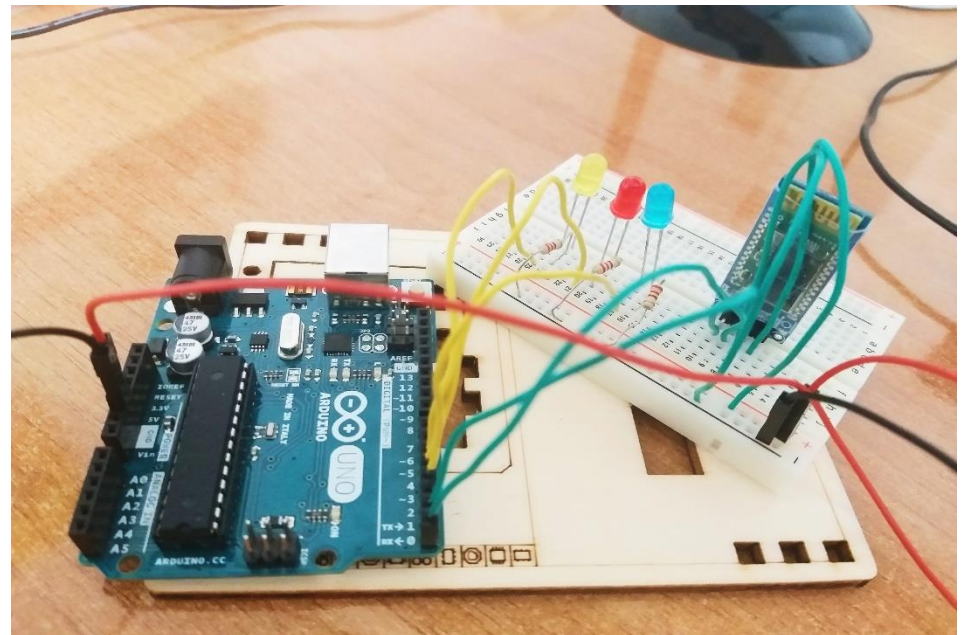
Desktop :

<https://vimeo.com/412016184>

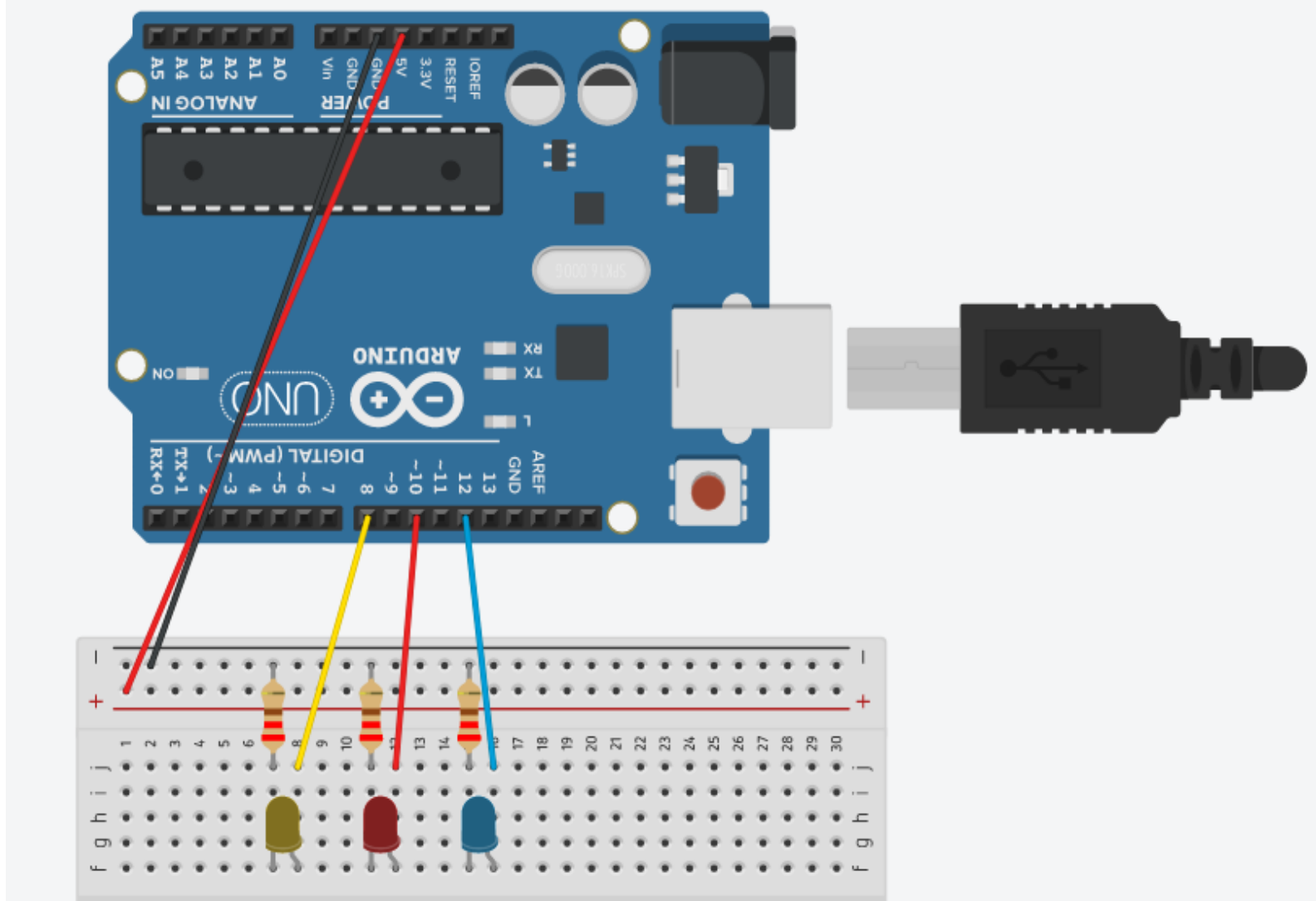
Mobile + Bluetooth

<https://vimeo.com/412020940>

PW : exemple2



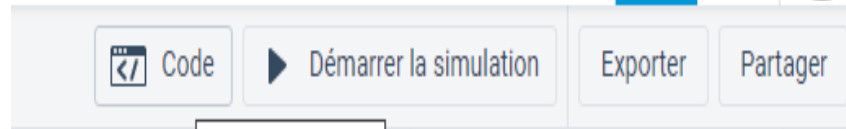
Exemple d'IHM 2



Exemple 2

Programmer l'Arduino en chargeant le code comme suit :

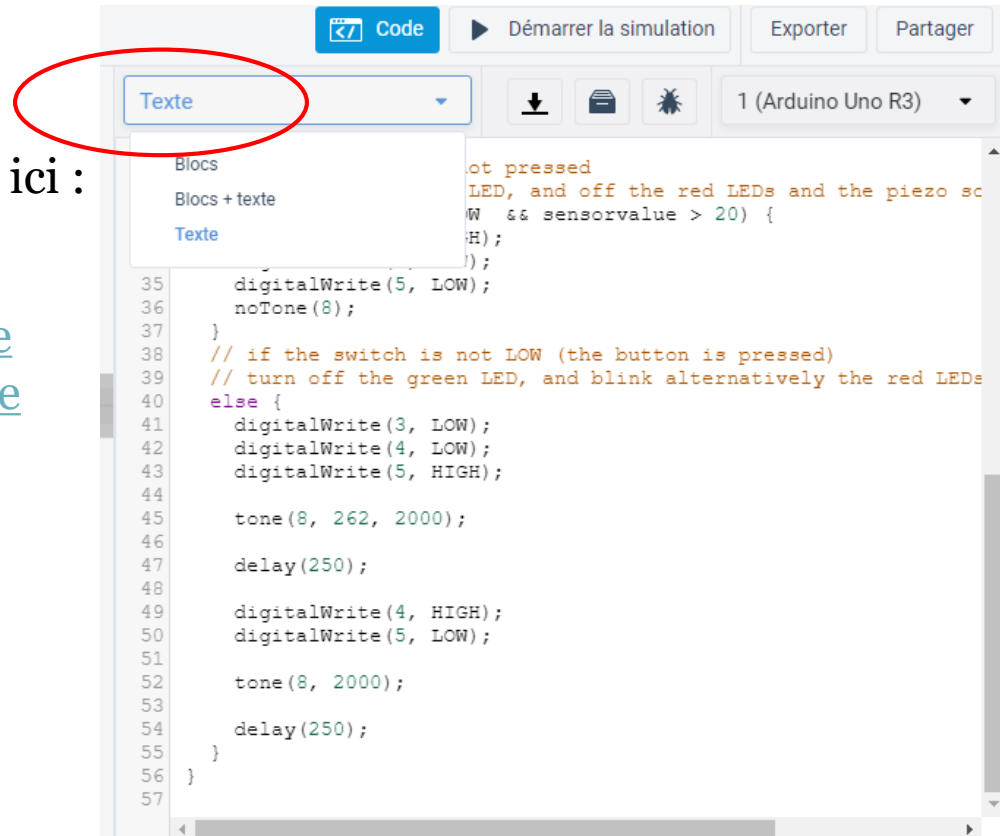
- Cliquer sur Code :



- Sélectionner **Texte** dans la liste

- Copier/Coller le code qui se trouve ici :

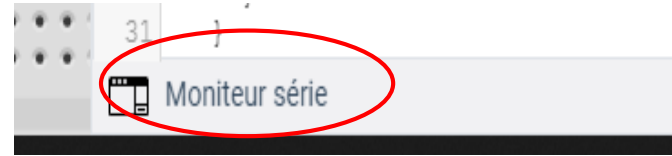
<https://github.com/GitTeaching/Exemple2ArduinoCoursIHM/blob/master/exemple2.ino>



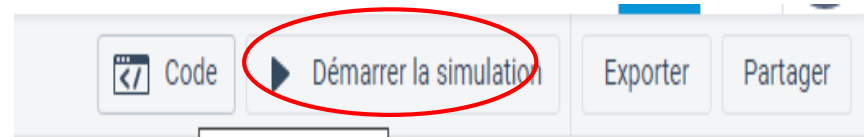
Exemple 2

Programmer l'Arduino en chargeant le code comme suit :

- Cliquer sur Moniteur Série en bas :

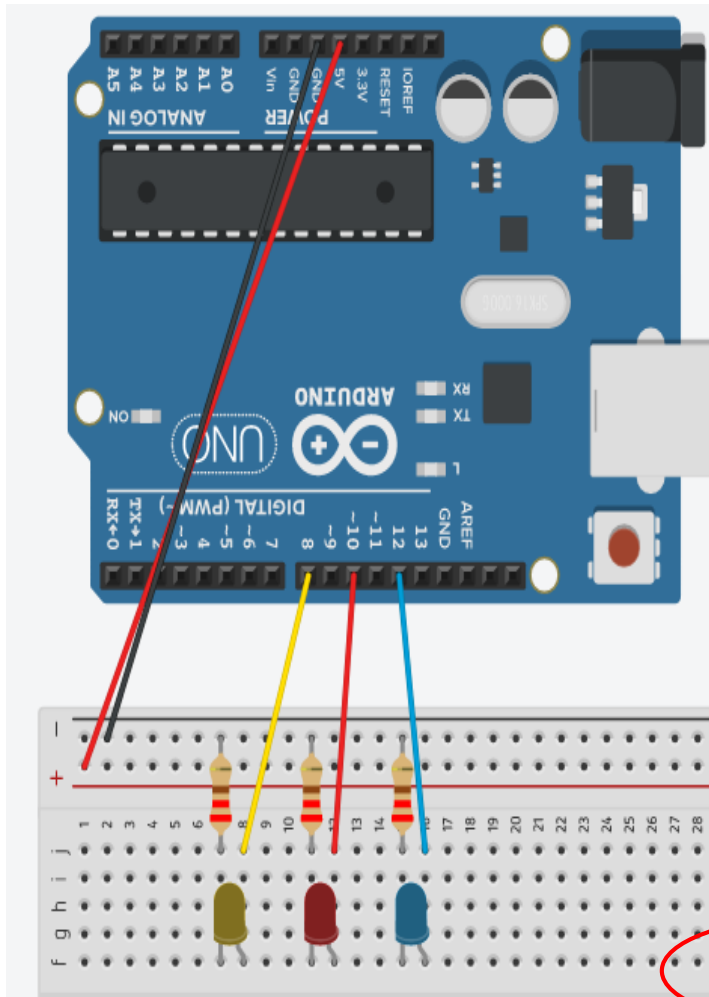


- Cliquer sur Démarrer la simulation :



- Entrer 'b' pour allumer le LED bleu, ou 'y' pour LED jaune, ou 'r' pour LED rouge, ou 'f' pour tout éteindre, puis cliquer sur Envoyer, comme suit:

Exemple 2



```
Texte 1 (Arduino Uno R3)
4 pinMode(10, OUTPUT); //set pin as output , red led
5 pinMode(8, OUTPUT); //set pin as output , yellow led
6
7 Serial.begin(9600); //start serial communication @9600 bps
8 }
9
10 void loop(){
11
12   if(Serial.available()){ //id data is available to read
13
14     char val = Serial.read();
15
16     if(val == 'r'){ //if r received
17       digitalWrite(10, HIGH); //turn on red led
18     }
19     if(val == 'b'){ //if b received
20       digitalWrite(12, HIGH); //turn on blue led
21     }
22     if(val == 'y'){ //if y received
23       digitalWrite(8, HIGH); //turn on yellow led
24     }
25   }
26 }
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Moniteur série

b

Envoyer Effacer

Exemple 2

Seulement en cas de réalisation réelle.



Version GUI Desktop

- Programmer l'interface graphique en utilisant l'IDE Processing:
- Téléchargement : <https://processing.org/download/>
- Code : https://github.com/GitTeaching/Exemple2ArduinoCoursIHM/blob/master/exemple2_GUI.pde

The screenshot shows the Processing IDE interface. The main window displays the code for sketch_200326b, which imports the ControlP5 library and sets up a GUI with four buttons: RED, YELLOW, BLUE, and ALLOFF. The code includes comments and variable declarations for the serial port and the ControlP5 object. A smaller window titled 'sketch_200326b' shows a preview of the GUI, which has a purple background and four dark blue buttons with white text.

```
1 import controlP5.*; //import ControlP5 library
2 import processing.serial.*;
3
4 Serial port;
5
6 ControlP5 cp5; //create ControlP5 object
7 PFont font;
8
9 void setup(){ //same as arduino program
10
11   size(300, 450); //window size, (width, height)
12   printArray(Serial.list()); //prints all available serial ports
13
14   port = new Serial(this, "COM6", 9600); //i have connected arduino
15
16   //lets add buton to empty window
17
18   cp5 = new ControlP5(this);
19   font = createFont("calibri light bold", 20); // custom fonts for
20
21   cp5.addButton("red") // "red" is the name of button
22     .setPosition(100, 50) //x and y coordinates of upper left corner
23     .setSize(120, 70) // (width, height)
24     .setFont(font);
25
26   ;
27
28   cp5.addButton("yellow") // "yellow" is the name of button
```

LED CONTROL

RED

YELLOW

BLUE

ALLOFF

[0] "COM6"

ControlP5 2.2.6 infos, comments, questions at <http://www.sojamo.de/libraries/controlP5>

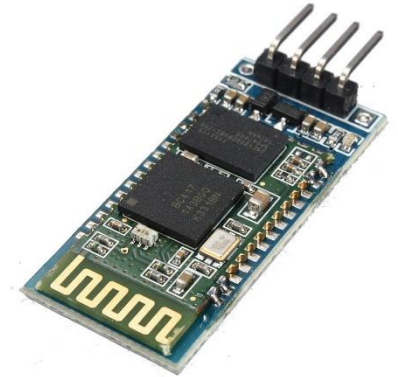
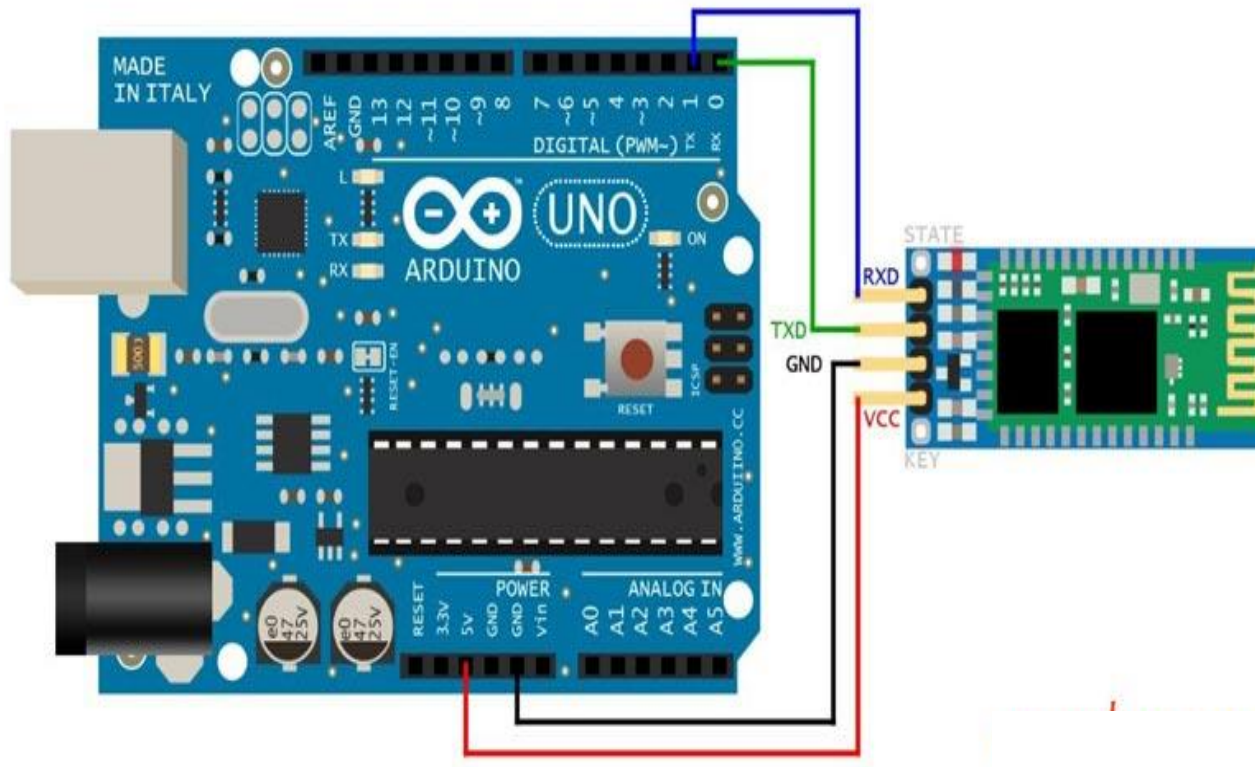
"calibri light bold" is not available, so another font will be used. Use PFont.list() to show available fonts.

Exemple 2

Seulement en cas de réalisation réelle.

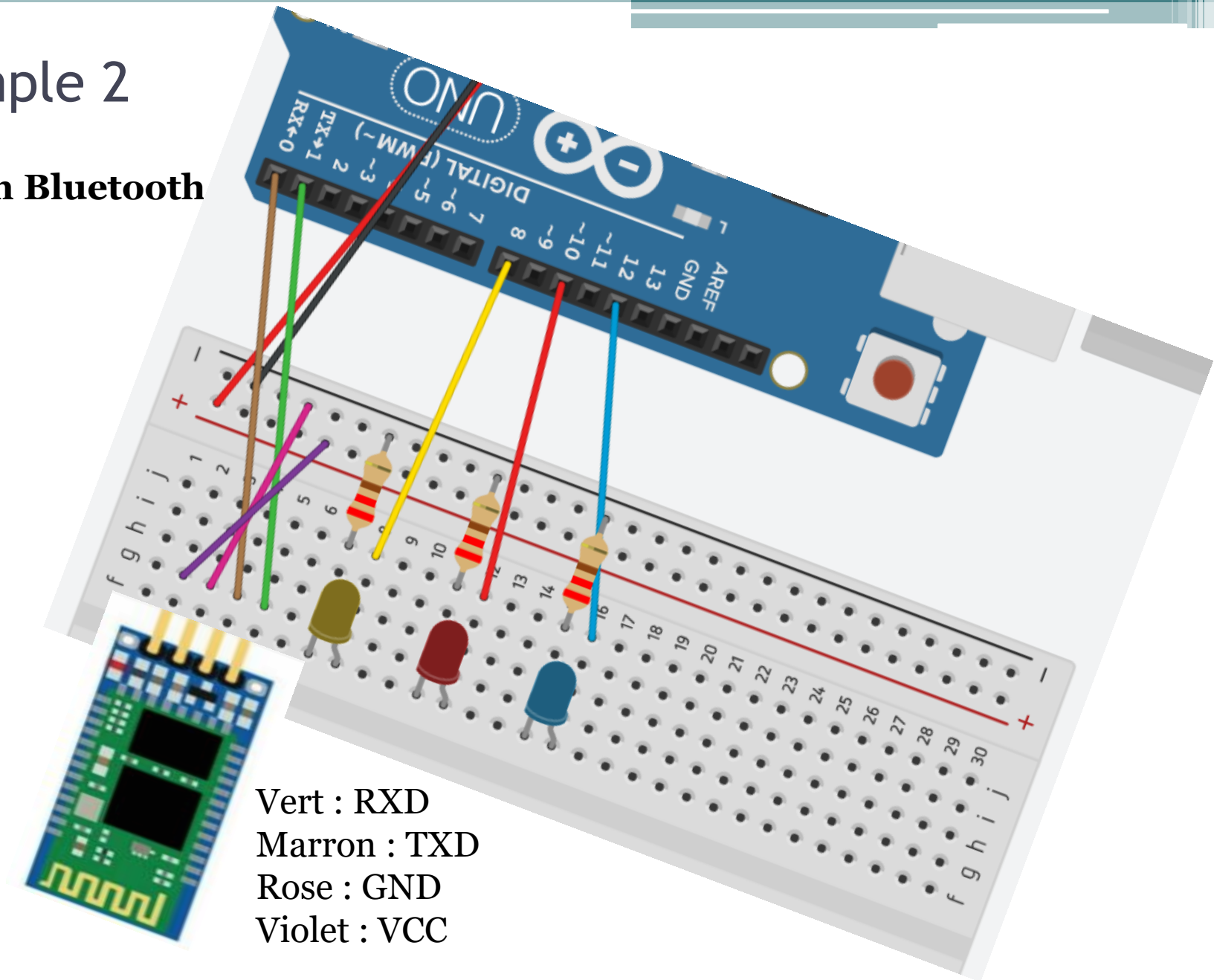
Version Bluetooth

- En utilisant le Module Arduino Bluetooth HC-06 :



Exemple 2

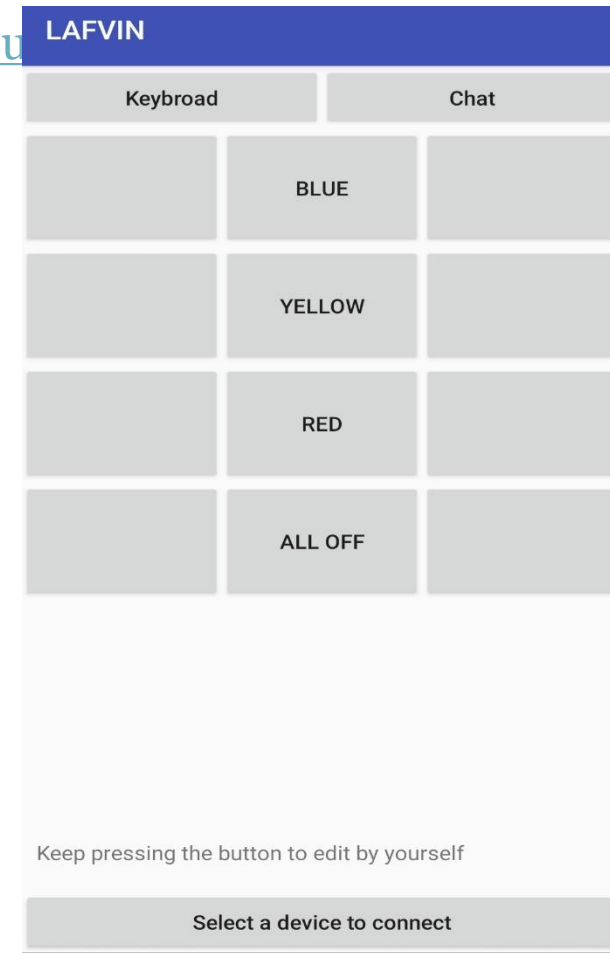
Version Bluetooth



Exemple 2

Version Bluetooth

- Programmer l'Arduino avec le code suivant :
<https://github.com/GitTeaching/Exemple2ArduinoCouplexemple2withbluetooth.ino>
- Interface graphique : Application Mobile Android à installer sur son smartphone.

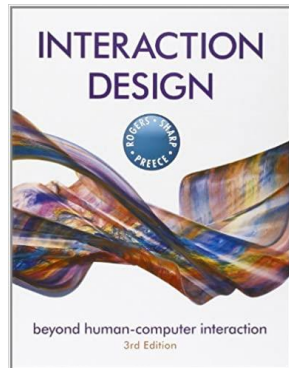


Références



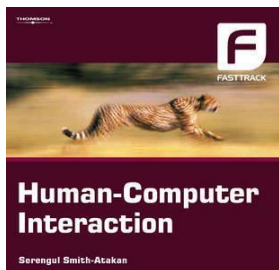
Designing Interactive Systems: A Comprehensive Guide to HCI, UX and Interaction Design, 3rd Edition

- ✓ Auteur : David Benyon
- ✓ Éditeur : Pearson
- ✓ Edition : 2013



Interaction Design: beyond human-computer interaction (3rd edition)

- ✓ Auteur : Yvonne Rogers, Helen Sharp & Jenny Preece
- ✓ Éditeur : Wiley
- ✓ Edition : 2011



The FastTrack to Human-Computer Interaction

- ✓ Auteur : Serengul Smith-Atakan
- ✓ Éditeur : Thomson Learning
- ✓ Edition : 2006