

# THEREMINE ARDUINO

## THEREMINE ARDUINO

Idée :

Au dernier atelier Arduino du Fablab avec Gauthier et Mathis , nous avons fait connaissance avec la carte Arduino et nous avons fait clignoter des leds .

Le **thérémine** est un des plus anciens [instruments de musique électronique](#), inventé en 1920 (connu sous le nom de « Léon Thérémine »).

<https://www.youtube-nocookie.com/embed/w5qf9O6c20o?si=cqRdPwdaFL0-zZCP>

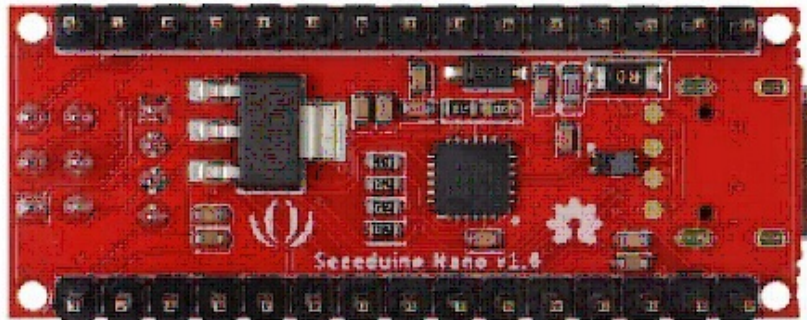
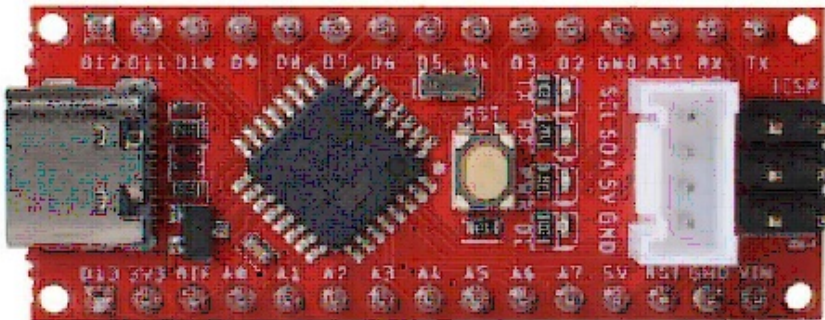
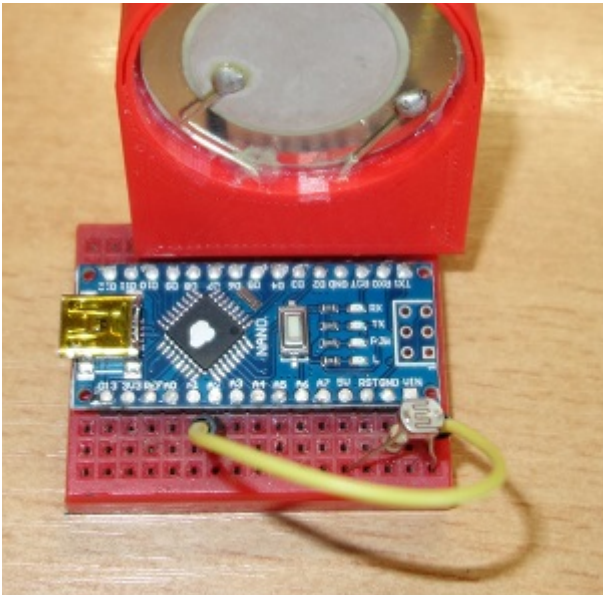
Le Mardi 26 Septembre à partir de 18 Heures , nous allons faire du son avec Arduino .

Matériel:

Chaque participant qui souhaite faire le THEREMINE ARDUINO devra venir si possible avec son PC Portable + souris et l'IDE ARDUINO installée .

Chaque participant disposera sur place d'un kit THEREMINE ARDUINO complet pour faire la manip :  
1 Arduino NANO , breadboard , 1 résistance 1k 1/4w , 1LDR , câbles 1 raccord USB C ou Mini , 1 buzzer .

Carte Arduino NANO



Au programme:

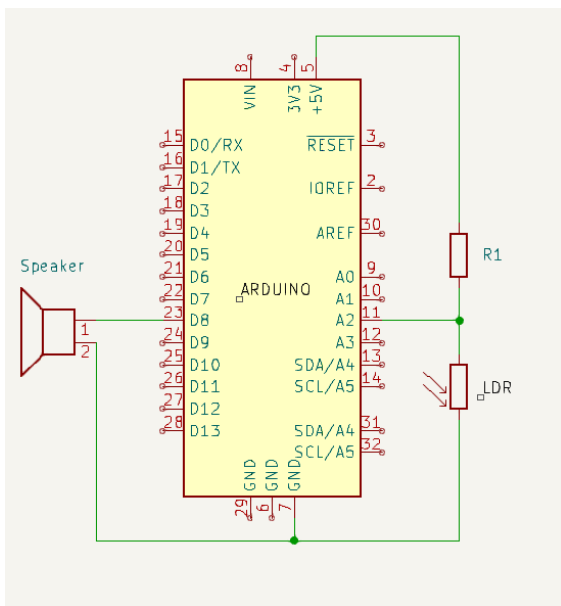
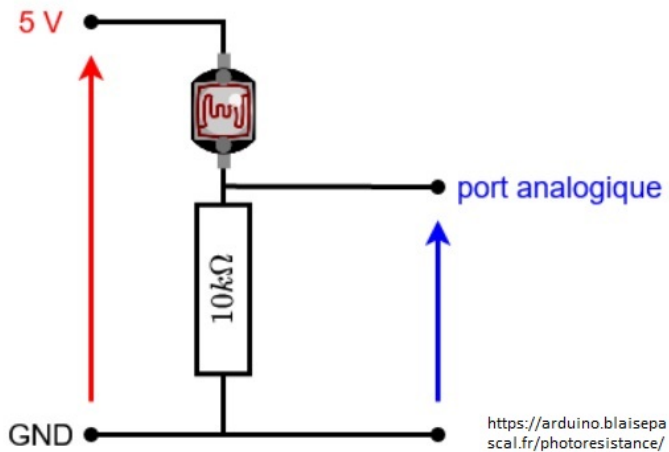
- 1 Téléverser le sketch sur la carte
- 2 Câbler le circuit
- 3 GENERER UN SON
- 4 Afficher le moniteur série

5 Améliorations du son en fonction de la distance des mains par rapport aux données dans le moniteur série.

## Shéma:

SCHEMA ELECTRIQUE:

Capteur LDR (LIGHT DEPENDANT RESISTOR)



Code "sketch":

## Données codées sur 8 bits :

- `char` ou `int8_t` : valeurs de -128 à +127
- `unsigned char` ou `uint8_t` : valeurs de 0 à 255

## Données codées sur 16 bits :

- `int` ou `int16_t` : valeurs de -32768 à 32767
- `unsigned int` ou `uint16_t` : valeurs de 0 à 65535

En observant les données affichées dans le moniteur série ,  
modifier les valeurs dans le sketch.

1 essayer d'autre valeurs : 80, 200, 80, 2800 pour réduire la  
fourchette par essais successifs

2 Commenter en plaçant des `//` au début  
des lignes concernant l'affichage en  
moniteur série.

```
//Montage d'un buzzer piezzo emettant un son dont la fréquence correspond
//à la luminosité perçue par une photorésistance

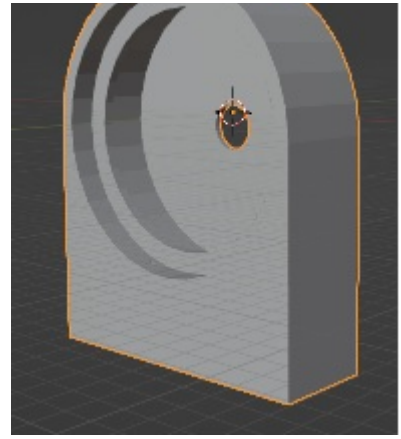
// déclarations
const uint8_t analogInPin = A0; // broche entrée
const uint8_t analogOutPin = 8; // valeur 8 (buzzer)
uint16_t luminosite = 0; //variable 0 à 1023
uint16_t frequence = 0; //variable 0 à 30000

void setup() {
    Serial.begin(9600); //vitesse de communication moniteur série
}
```

```
void loop() {  
  //lecture ADC  
  luminosite = analogRead(analogInPin);  
  //mise à l'échelle  
  frequence = map(luminosite, 0, 1023, 50, 30000); //reduire la fourchette par essais  
successifs  
  // Conversion tension en fréquence :  
  tone(analogOutPin, frequence);  
  
  // Suivi des valeurs sur le moniteur:  
  Serial.print("luminosite vue = " );  
  Serial.print(luminosite);  
  Serial.print("\t frequence = ");  
  Serial.print(frequence);  
  Serial.println(" Hz");  
  
  delay(2);  
}
```

## IMPRESSION 3D

Le support buzzer a été modélisé avec le logiciel BLENDER3D et imprimé sur la CREALITY ENDER 3V2 visible dans le Fablab.



LE SLICER S3D

Essai.

---

Revision #8

Created 19 September 2023 16:27:00 by Jean

Updated 31 October 2023 13:07:18 by admin\_idf