

RZ75 : Le haut-parleur lumineux

Les étapes de réalisations du RZ75

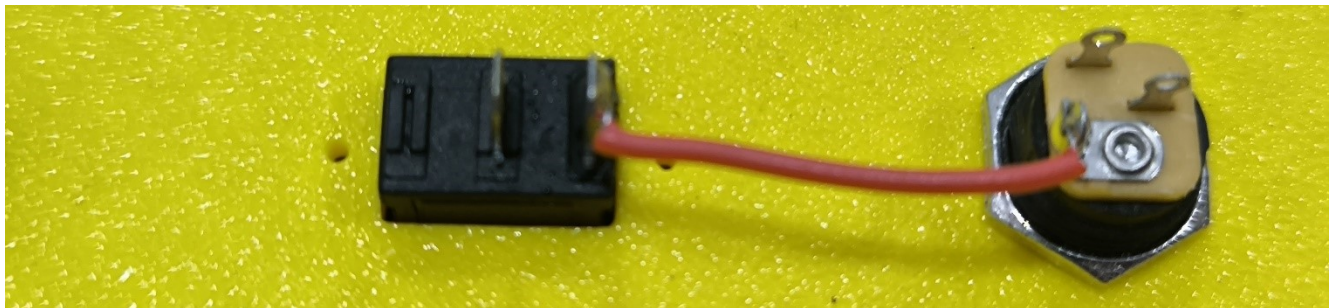
Les RZ75 que vous allez concevoir ne sont pas de simples prototypes : ce sont des objets que vous emporterez ensuite avec vous et qui pourront être utilisés comme haut-parleurs lumineux. Pour cette raison, chaque étape de fabrication doit être réalisée avec soin afin d'obtenir un objet fiable, solide, propre et sécurisé. Un montage approximatif, avec un câblage désordonné, des soudures fragiles ou des finitions négligées, peut entraîner des pannes et rendre l'appareil inutilisable. Une réalisation rigoureuse et de qualité est donc attendue de chaque groupe.

Demandez au professeur si vous avez un doute sur chaque étape

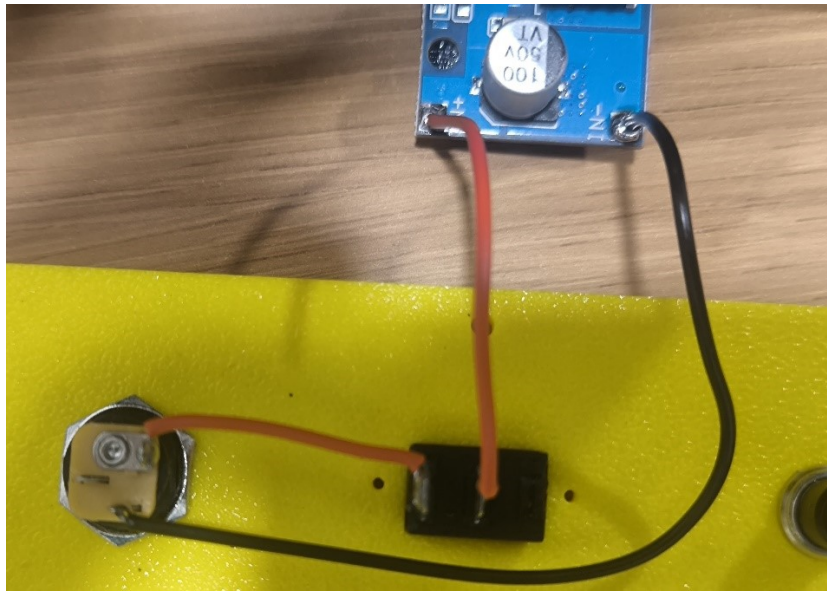
Stage 1 (Durée: 1H30min)

La prise d'alimentation, l'interrupteur et l'abaisseur de tension

- Relier la **borne +** de la prise d'alimentation avec la borne **de gauche** de l'interrupteur
 - Longueur du fil rouge: **5cm au minimum**

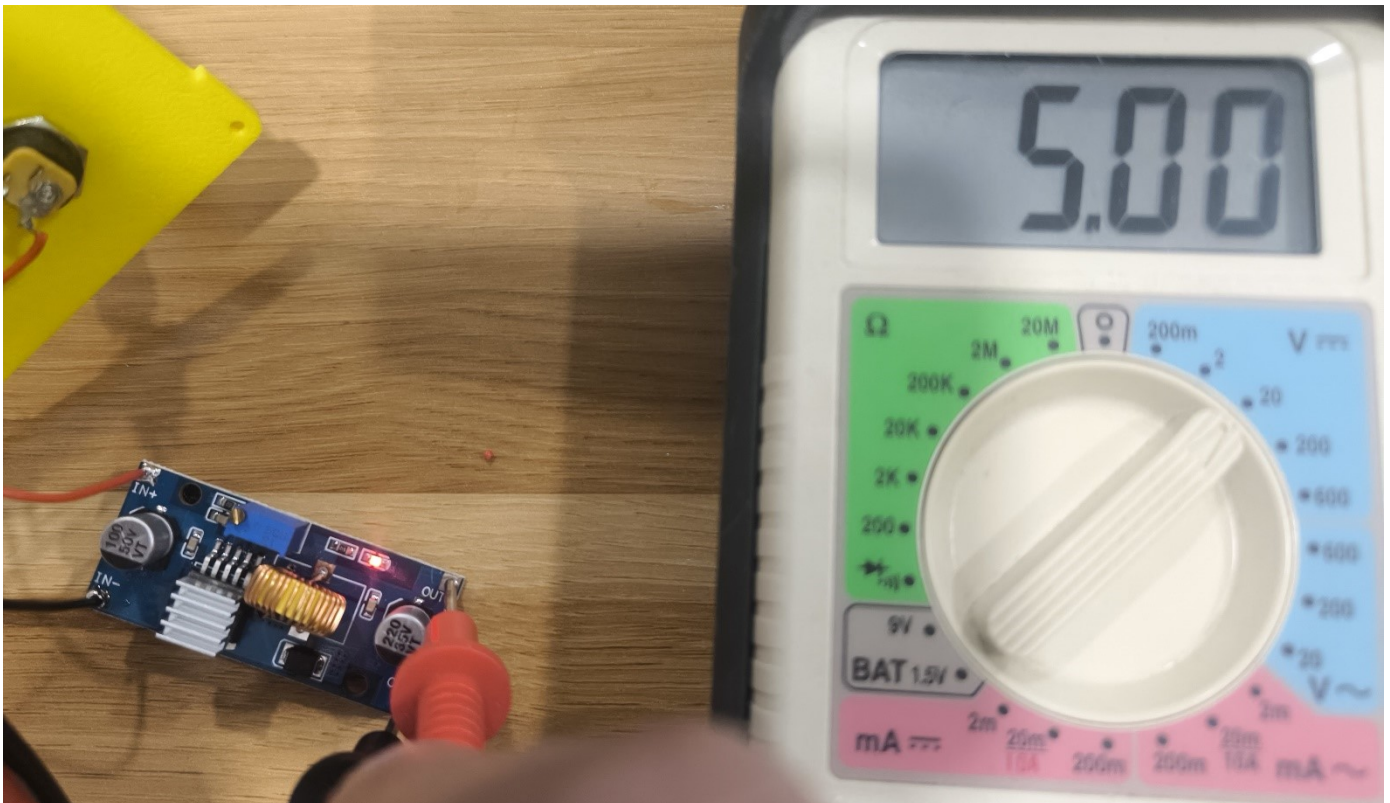


- Relier la borne **de droite** de l'interrupteur au **IN+** de l'abaisseur de tension
 - Longueur du fil rouge : **6cm au minimum**
- Relier la **borne -** de la prise d'alimentation au **IN-** de l'abaisseur de tension
 - Longueur du fil noir : **12cm au minimum**



- Brancher l'alimentation de **12Volts** et à l'aide du voltmètre (Tension continue | Sélecteur sur 20) et d'un tournevis régler la tension de sortie de l'abaisseur à exactement **5Volts**

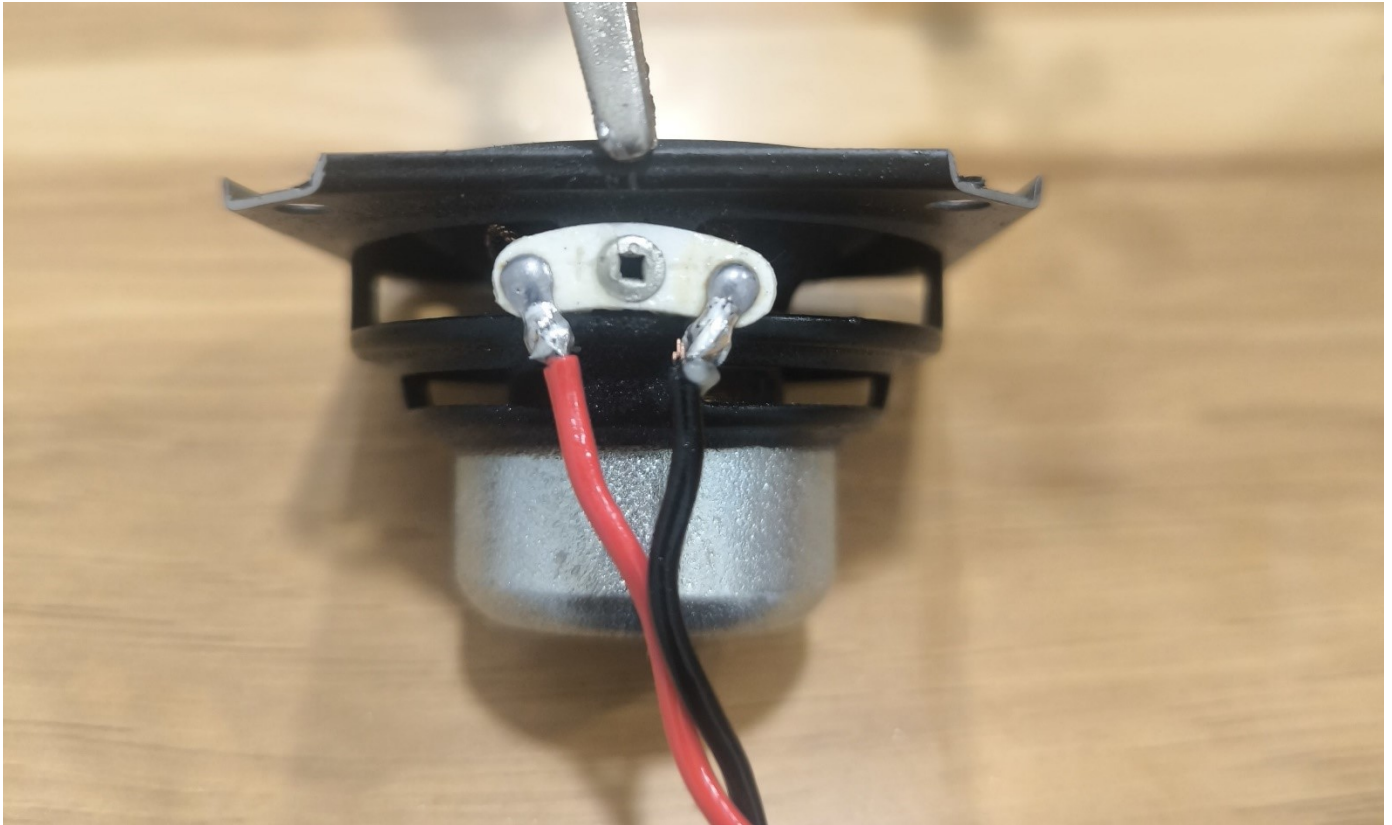
Important : La sortie de l'abaisseur (sur 5Volts) va alimenter l'ESP32 et les matrices LEDs qui ont besoin de **5Volts** exactement sinon risque de détérioration.



Les haut-parleurs, l'abaisseur de tension et la carte son (XY-P15W)

- Souder les câbles torsadés des hauts parleurs (Rouge dans + et Noir dans -)

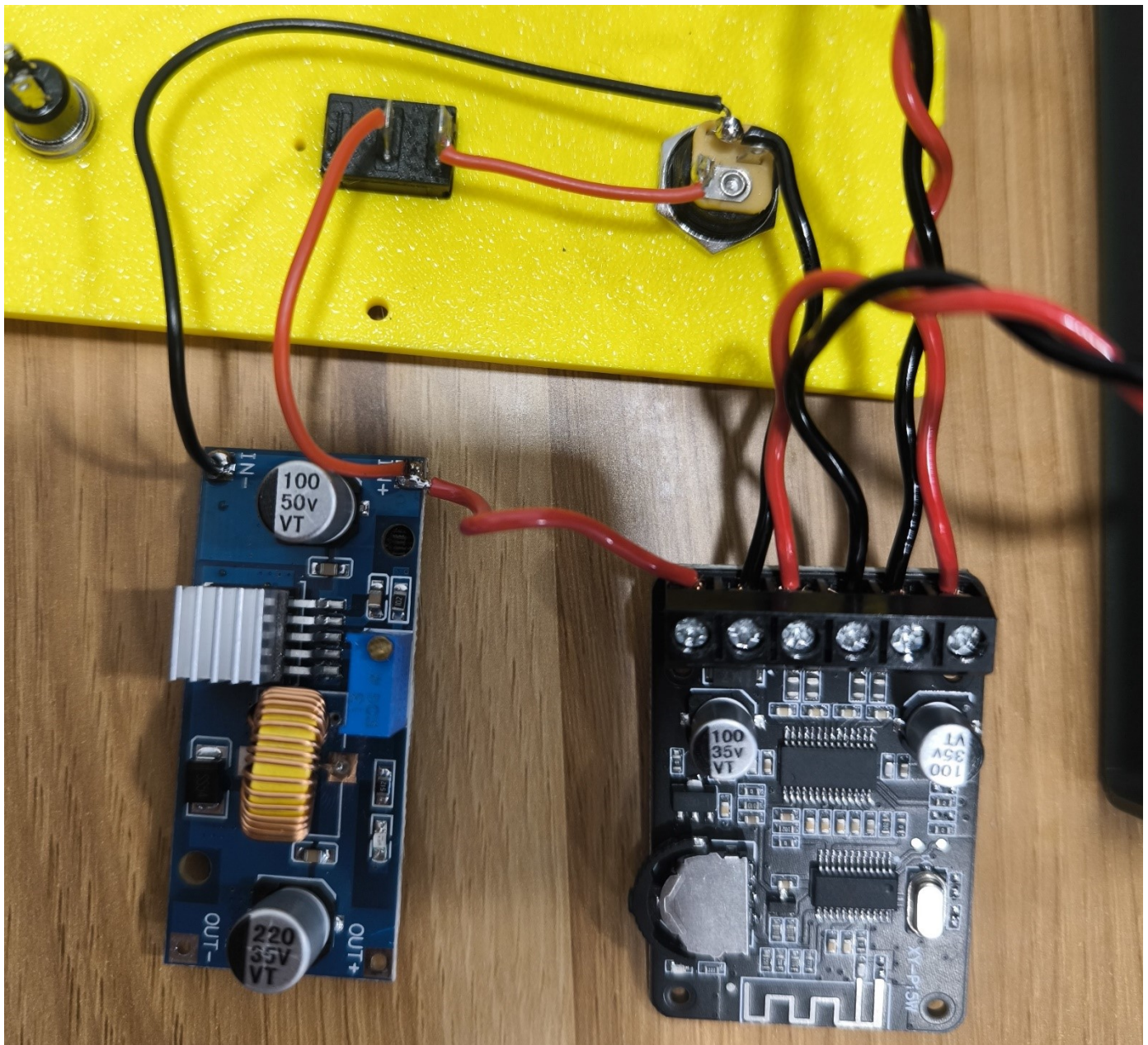
- Longueur câble torsadé (rouge et noir pour haut-parleur) : **20 cm**



- Fixer les 2 hauts parleurs sur le boîtier principal
 - Référence des vis : **M2x6**



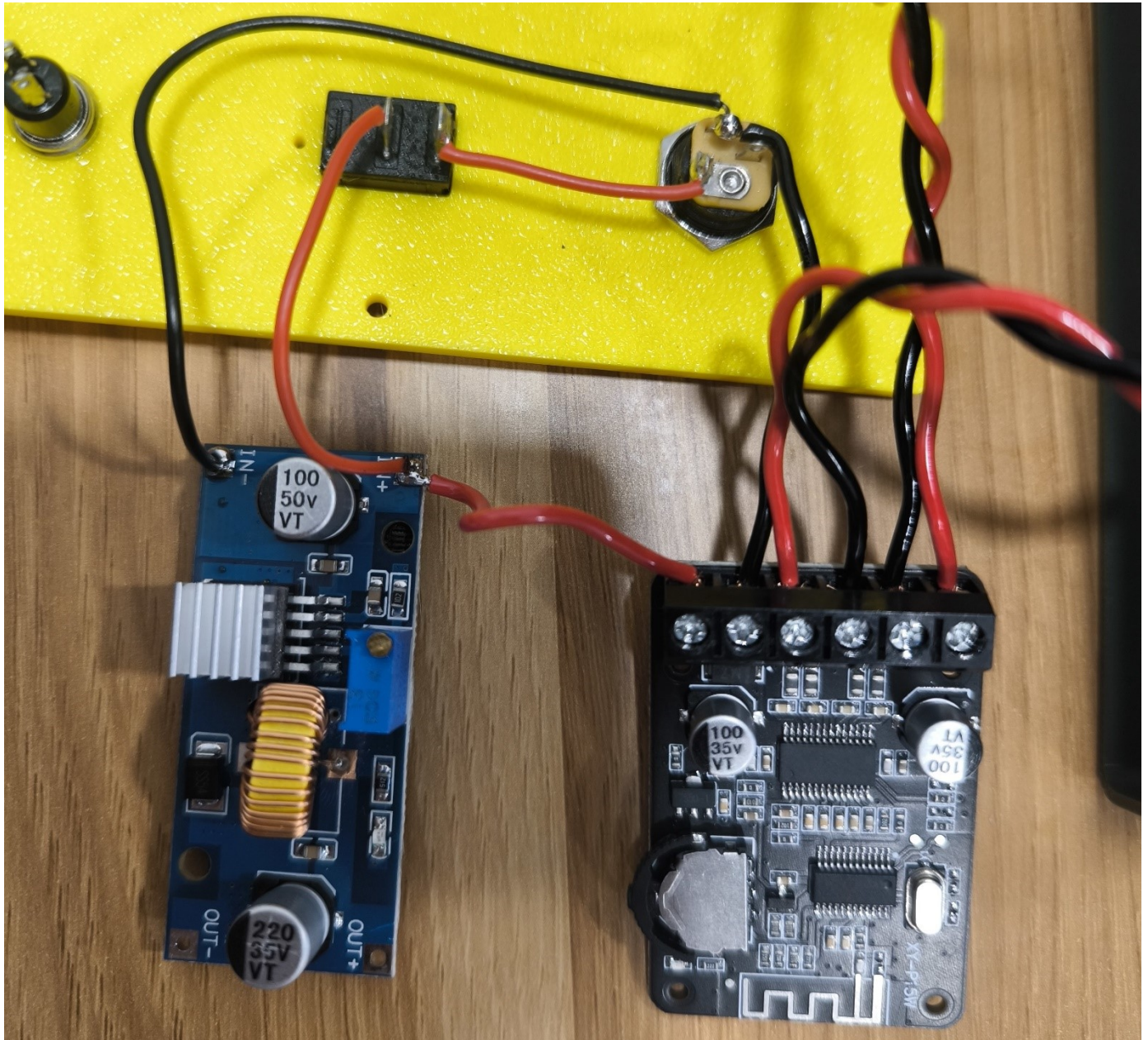
- Relier la borne **de droite** de l'interrupteur (ou IN+ de l'abaisseur) avec **Power +** de la carte son. x
 - Longueur du fil torsadé rouge: **7cm**
- Relier la **borne -** de la prise d'alimentation (ou IN+ de l'abaisseur) avec **Power -** de la carte son.
 - Longueur du fil torsadé noir : **7cm**



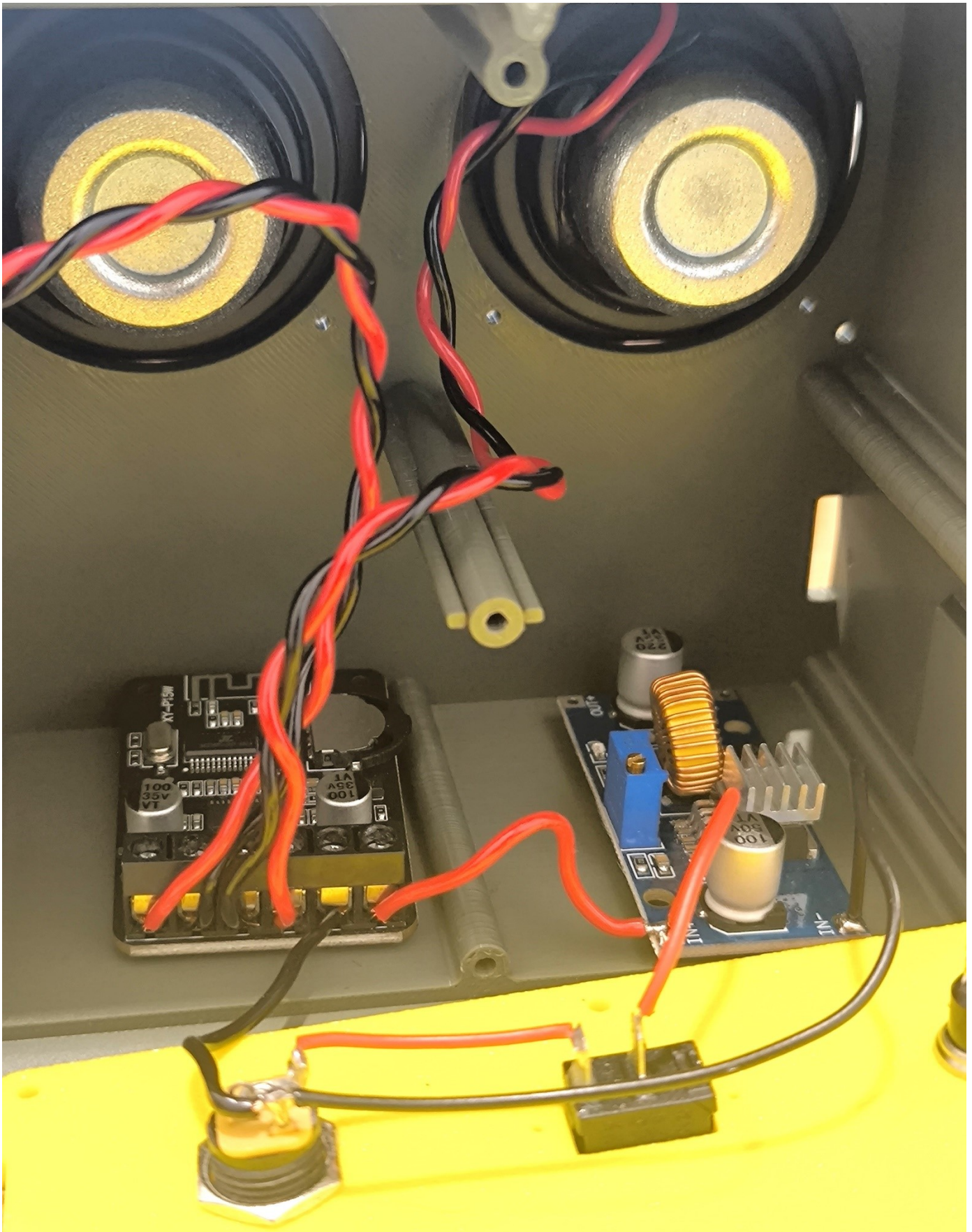
- Relier la **borne +** et la **borne –** de chaque haut-parleur à la borne + et la borne – de SPK1 et SPK2 de la carte son XY-P15W.

Important : Brancher votre alimentation de **12V** sur le RZ75, la LED de la carte son et la LED de l'abaisseur doivent s'allumer. Ensuite, à l'aide d'un ordinateur portable,

connectez-vous à la carte son par bluetooth. Testez le son !



Lorsque le son sera OK, vous pouvez utiliser les scratchs et fixer la carte son.



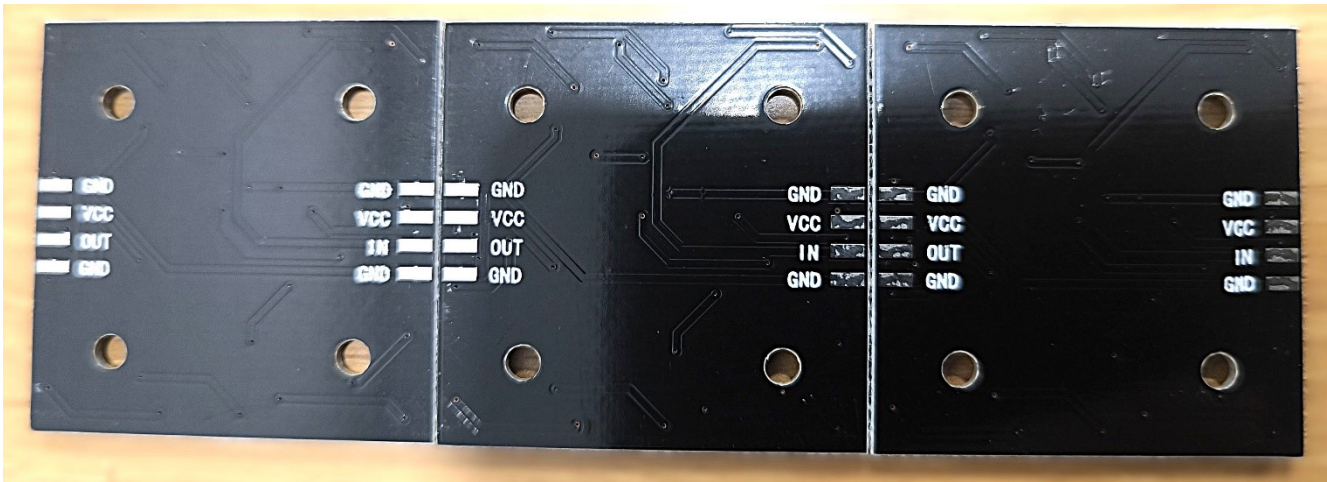
Important : Afin de passer à l'étape suivante et obtenir les composants de l'étape 2 vous devez absolument valider cette étape avec votre professeur.

Stage 2 (Durée: 30 min)

Le stage 2 demande une concentration et une précision maximale

Les 3 matrices LED

1. Placer les trois matrices comme sur la photo ci-dessous. Vérifier les positions des **IN** et **OUT** comme sur la photo. Les matrices doivent rester parfaitement alignées.



2. Créer 4 contacts métalliques à partir des barrettes de pins mâles type Dupont

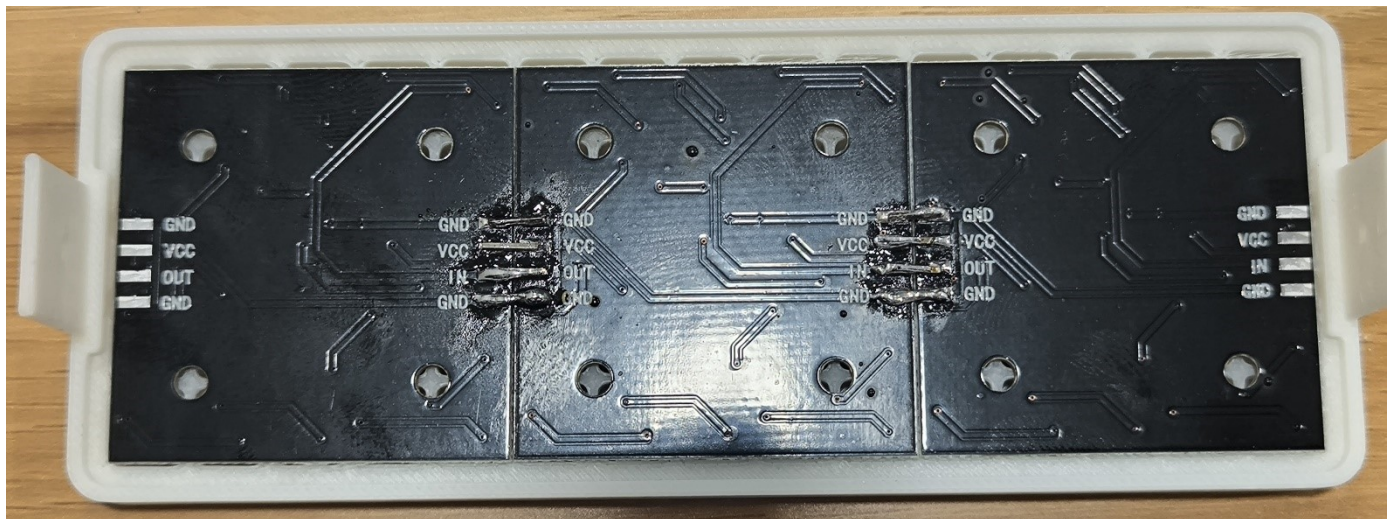


3. **Appeler le professeur pour vous conseiller sur cette étape :** Créer 8 contacts métalliques en coupant en 2 les 4 contacts des barrettes de pins mâles type Dupont. Utiliser la pince coupante et faire très attention aux éclats en sécurisant avec votre main



4. Souder les contacts entre les 3 matrices comme sur la photo ci-dessous

C'est l'une des parties les plus difficile du projet



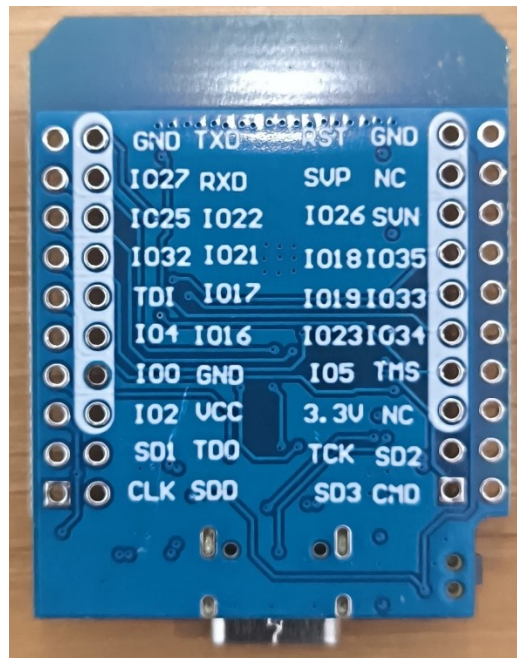
Une fois soudé, vérifiez qu'il n'y a pas de court-circuit. Corrigez-s'il le faut.

Important : Afin de passer à l'étape suivante et obtenir les composants du stage 3 vous devez absolument valider cette étape avec votre professeur.

Stage 3 (Durée : 30 min)

Le stage 3 demande une concentration et une précision maximale et aussi dans le choix des types de fils.

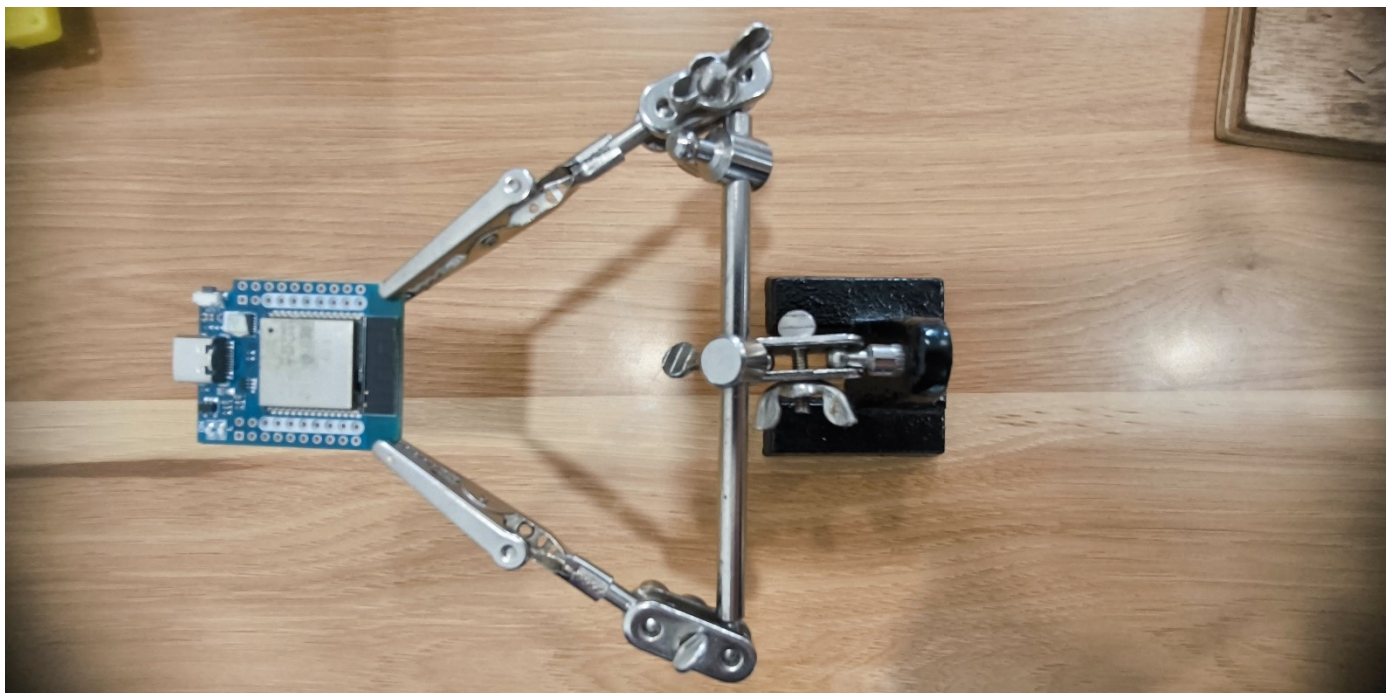
Le microcontrôleur ESP32 D1 MINI



1. L'ESP32 sera connecté **au microphone** grâce à **3** fils.
2. L'ESP 32 sera connecté **au bouton poussoir** grâce à **2** fils.
3. L'ESP32 sera connecté **à l'abaisseur de tension** (OUT + et OUT -) grâce à **2** fils.
4. L'ESP32 sera connecté **aux matrices LED** grâce à **1** fil.

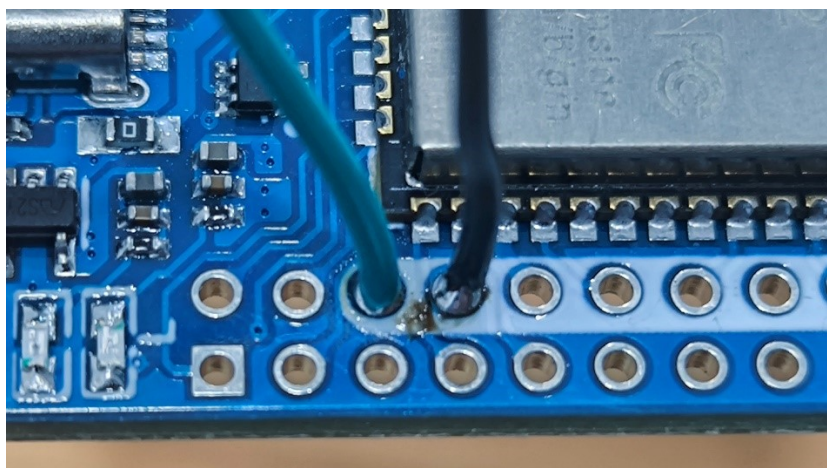
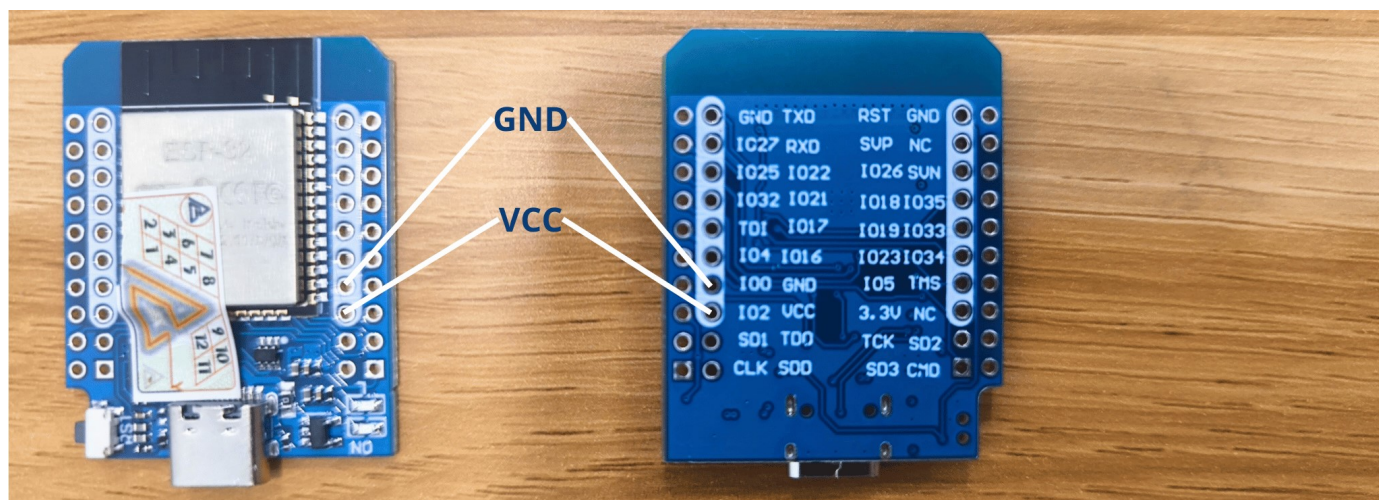
Dans un premier temps, nous allons souder sur l'ESP32 l'ensemble des fils que nous utiliserons, **c'est-à-dire 8 fils**.

Utilisez la pince crocodile comme ci-dessous :



5. Sur l'ESP32 : Pour alimenter l'ESP32 qui a besoin de 5Volts, il faut souder 2 fils sur VCC et GND

- VCC, longueur du fil rouge: **15cm en AWG22 ou AWG24 ou AWG26**
- GND, Longueur de fil noir : **15cm en AWG22 ou AWG24 ou AWG26**

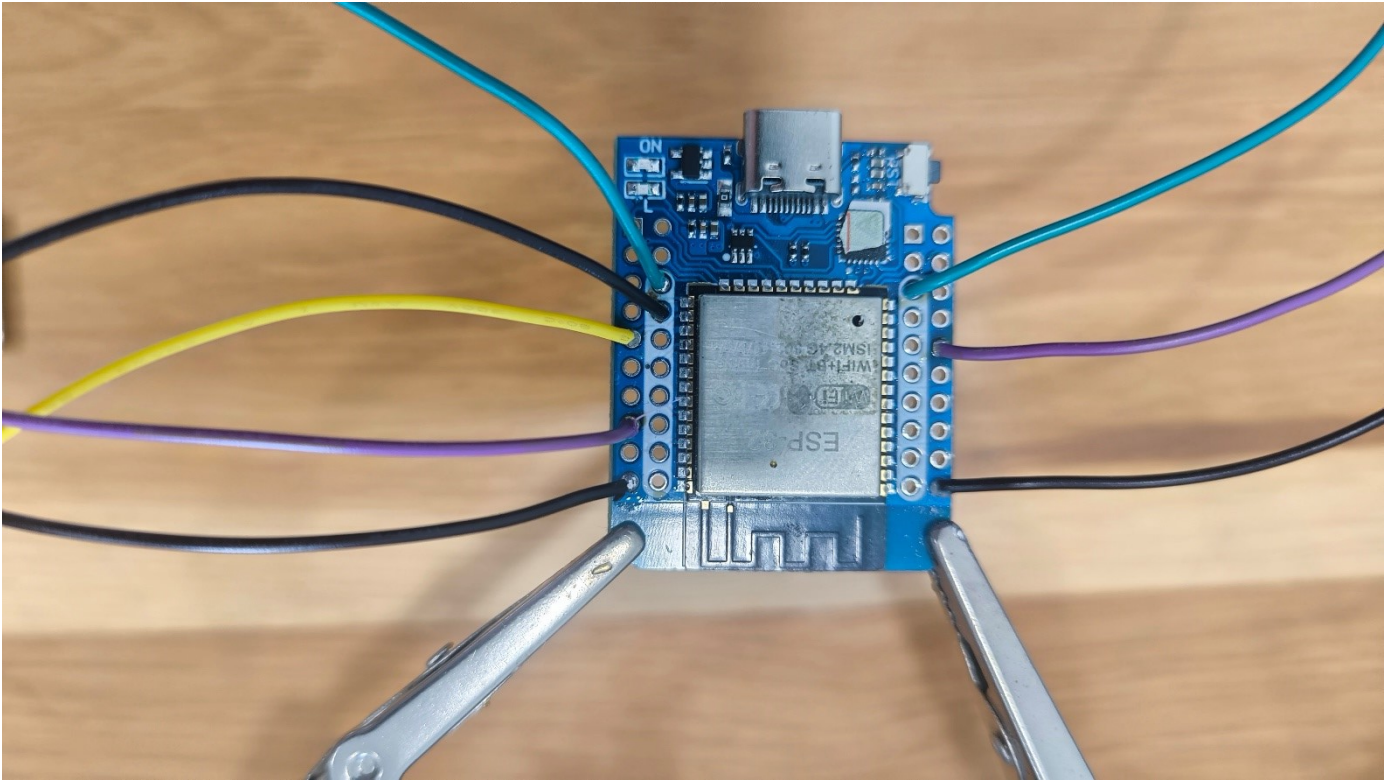


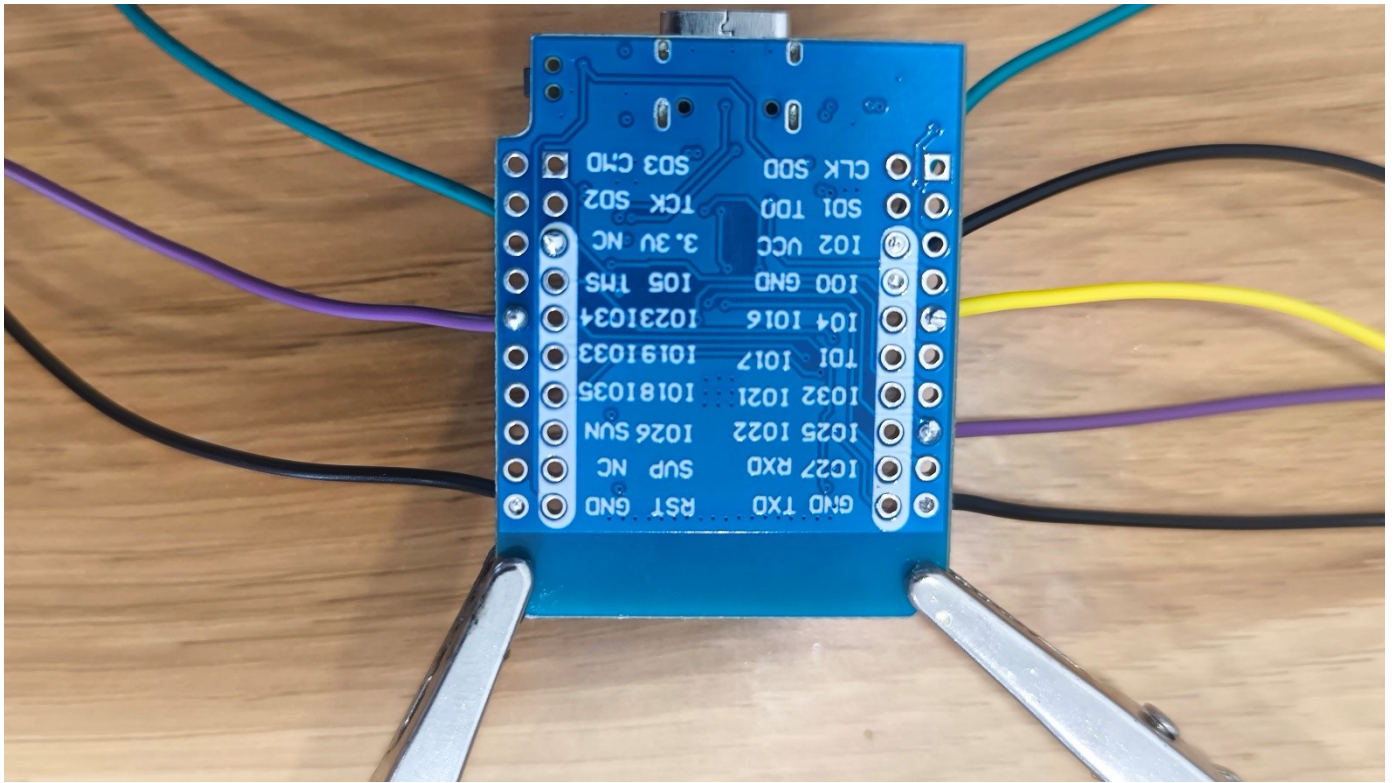
6. Pour le reste des soudures, suivre le tableau ci-dessous :

Les 8 fils doivent avoir une longueur de 15cm chacun

Élément	Couleur du fil (conseillée)	Sur le module	Sur l'ESP32
Alimentation carte	Rouge	OUT + (abaisseur 5V)	VCC (5V)
Alimentation carte	Noir	OUT – (abaisseur)	GND
Matrice LED	Rouge 22AWG	VCC	VCC (5V)
Matrice LED	Noir 22AWG	GND	GND

Matrice LED (signal)	Mauve/Jaune / Blanc	IN	IO4
Micro (GY-MAX4466)	Rouge	VCC	3.3V
Micro (GY-MAX4466)	Noir	GND	GND
Micro (signal)	Jaune / Blanc	OUT	IO34
Bouton poussoir	—	Patte 1	IO25
Bouton poussoir	—	Patte 2	GND



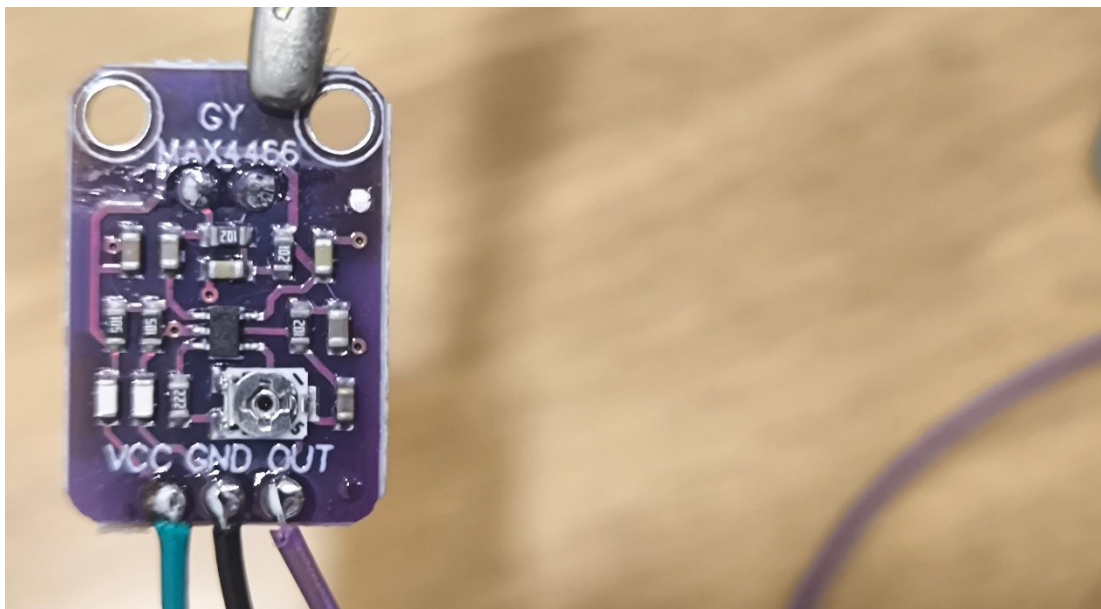


Important : Afin de passer à l'étape suivante et obtenir les composants du stage 4 vous devez absolument valider cette étape avec votre professeur.

Stage 4 (Durée : 45 min)

Le microphone max 4466 et branchement final

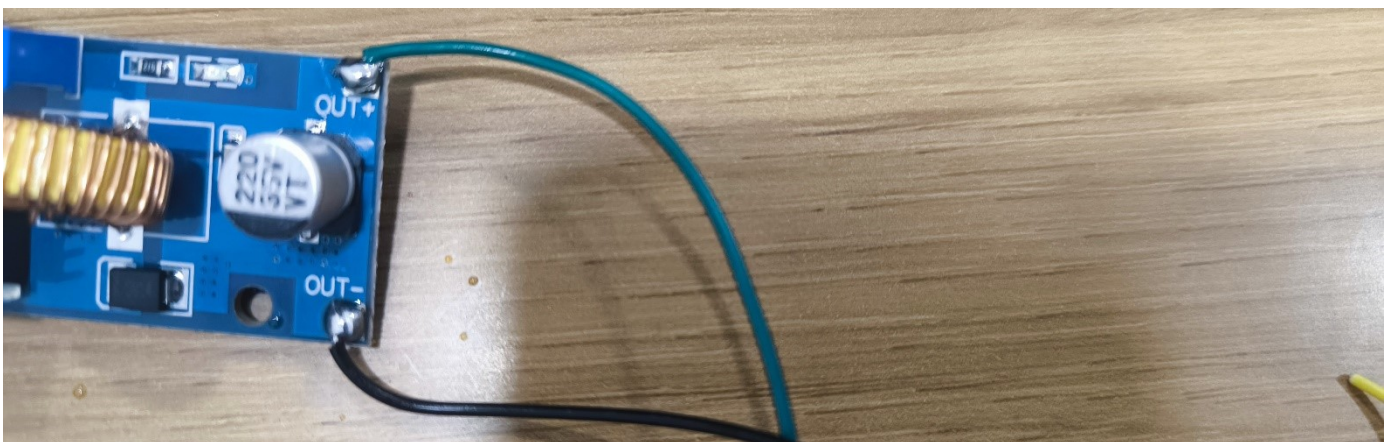
1. Nous allons connecter l'ESP32 au microphone comme indiqué sur le tableau ci-dessus (**3.3V | GND | IO34**)



2. Nous allons connecter l'ESP32 au bouton poussoir comme indiqué sur le tableau ci-dessus (**GND | IO25**)

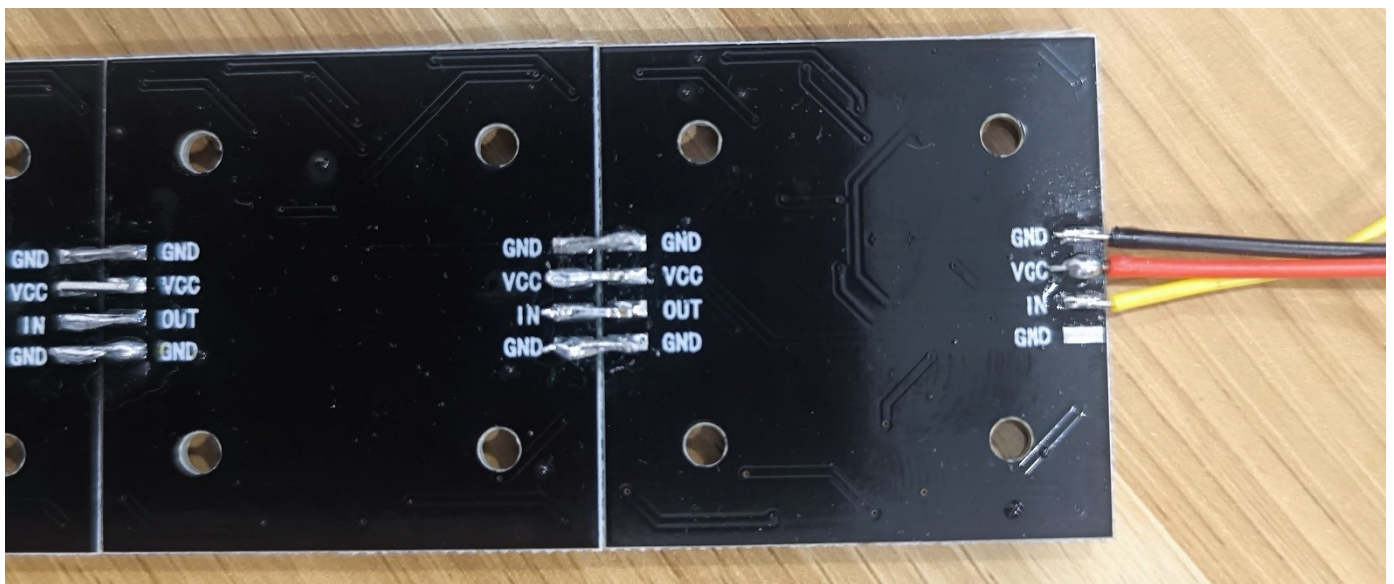


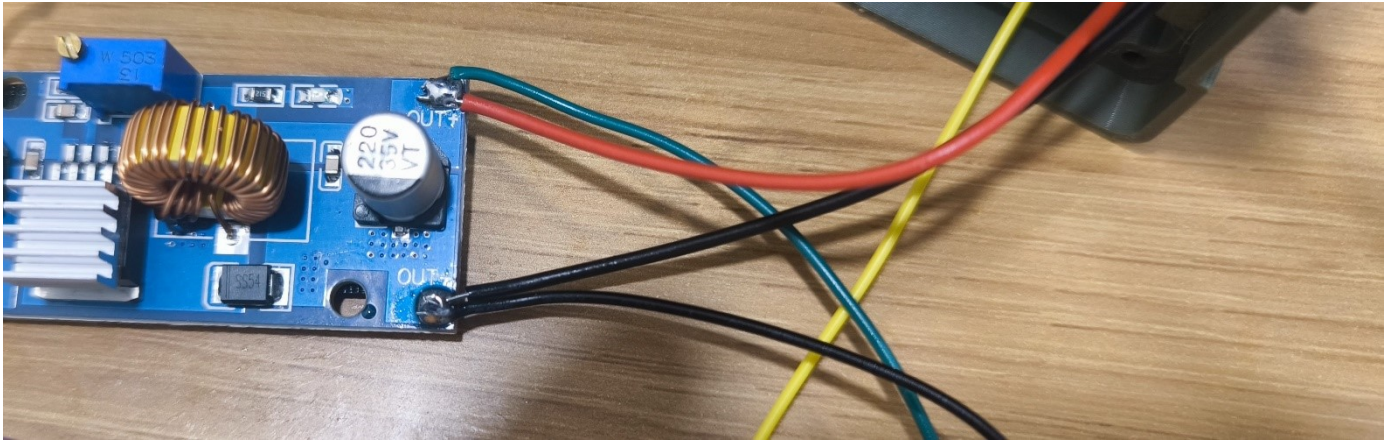
3. Nous allons connecter la matrice LED avec l'abaisseur de tension et avec l'ESP32 comme indiqué sur le tableau ci-dessus. (**VCC|GND**)



4. Nous allons connecter l'ESP32 et la matrice LED comme indiqué sur le tableau ci-dessus (**5V | GND | IO4**)

Important : Faire passer le fil entre IN et IO4 de l'intérieur du RZ75 (**VCC et GND : 10 cm 22AWG**)

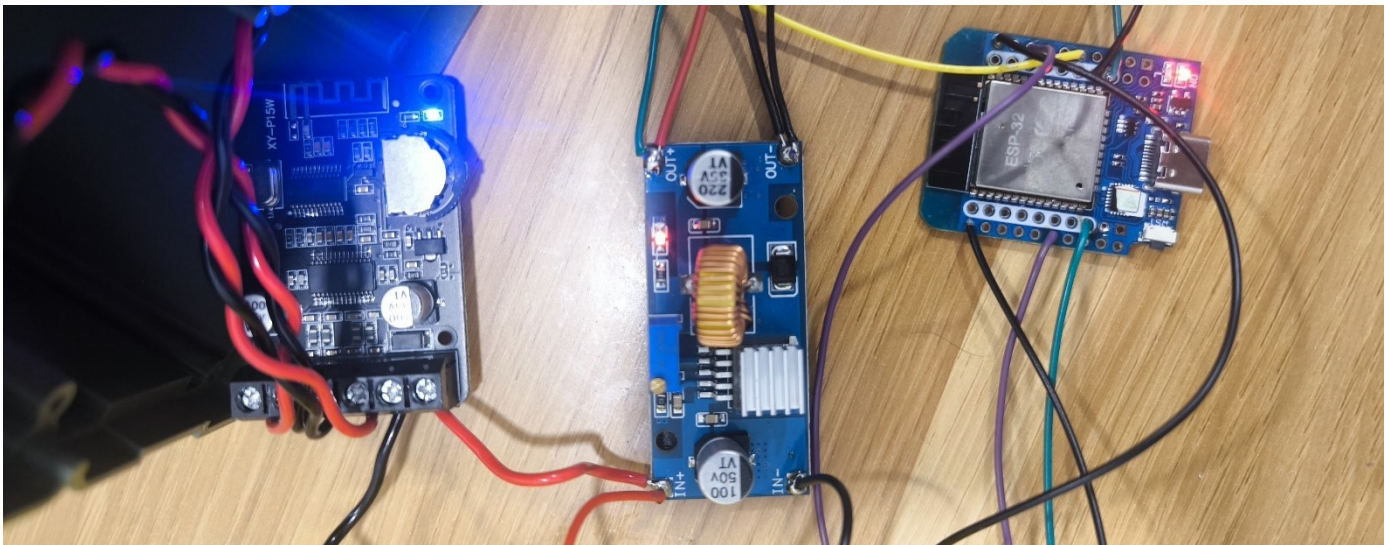




Important : Afin de passer à l'étape suivante et obtenir les composants du stage 5 vous devez absolument valider cette étape avec votre professeur.

Stage 5 (Durée : 30 min)

1. Alimenter en 12V le RZ75 : les LEDS de la carte son, de l'abaisseur de tension et de l'ESP32 vont s'allumer normalement.



2. Calculer la tension de sortie de l'abaisseur de tension entre OUT+ et OUT-

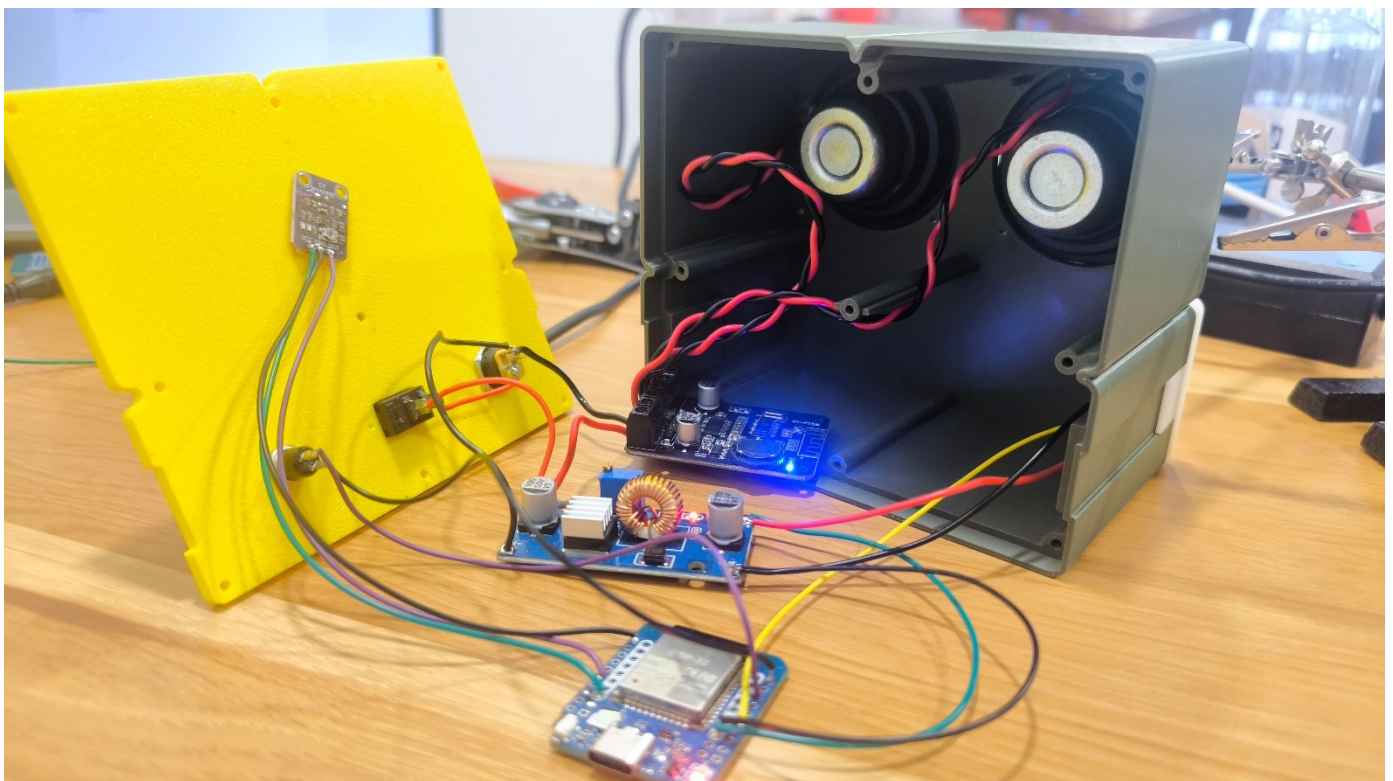
Tension entre OUT+ et OUT-	
----------------------------	--

3. Remplir le tableau ci-dessous :

Quelle couleur le fil et sur quel IO au niveau de l'ESP32 ?

Élément	Couleur du fil	Sur l'ESP32
Matrice LED (signal)		
Micro (signal)		
Bouton poussoir (signal)		

4. Le montage doit ressembler à ceci



Important : Afin de passer au stage 6 vous devez absolument valider cette étape avec votre professeur.

Stage 6 (Durée : 30 min)

On va Tester le microphone et les LEDs

1. Téléverser le fichier pour tester le microphone et régler la sensibilité du microphone en tournant la molette du microphone.

Test_micro_max4466_ESP32.ino

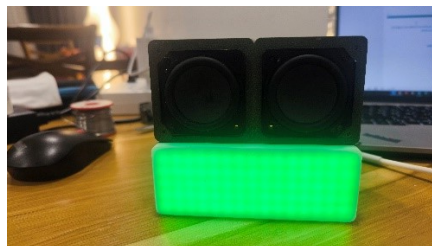
```
Test_micro_max4466_ESP32 | Arduino IDE 2.3.6
Fichier Edition Croquis Outils Aide
WEMOS D1 MINI ESP32
Test_micro_max4466_ESP32.ino
1
Sortie Moniteur série x
Message (Enter to send message to 'WEMOS D1 MINI ESP32' on 'COM43')
Retour chariot
min=1757 max=1826 mean=1809 amplitude=69 -> silence
min=1757 max=1843 mean=1808 amplitude=86 -> silence
min=1755 max=1823 mean=1809 amplitude=68 -> silence
min=1762 max=1822 mean=1809 amplitude=60 -> silence
min=1769 max=1827 mean=1810 amplitude=58 -> silence
min=1755 max=1829 mean=1809 amplitude=74 -> silence
min=1759 max=1826 mean=1808 amplitude=67 -> silence
min=1744 max=1834 mean=1808 amplitude=90 -> silence
min=1757 max=1827 mean=1809 amplitude=70 -> silence
min=1761 max=1827 mean=1810 amplitude=66 -> silence
min=1759 max=1834 mean=1810 amplitude=75 -> silence
min=1749 max=1837 mean=1810 amplitude=88 -> silence
min=1755 max=1835 mean=1810 amplitude=80 -> silence
min=1750 max=1840 mean=1809 amplitude=90 -> silence
min=1755 max=1835 mean=1808 amplitude=80 -> silence
min=1755 max=1831 mean=1808 amplitude=76 -> silence
min=1762 max=1857 mean=1809 amplitude=95 -> silence
min=1764 max=1827 mean=1807 amplitude=63 -> silence
min=1763 max=1840 mean=1808 amplitude=77 -> silence
min=1200 max=2683 mean=1821 amplitude=1483 -> SON DETECTE
min=1023 max=2503 mean=1807 amplitude=1480 -> SON DETECTE
min=1243 max=2352 mean=1804 amplitude=1109 -> SON DETECTE
min=1390 max=2381 mean=1810 amplitude=991 -> SON DETECTE
min=1397 max=2305 mean=1802 amplitude=908 -> SON DETECTE
min=1535 max=2007 mean=1809 amplitude=472 -> SON DETECTE
min=1744 max=1842 mean=1809 amplitude=98 -> silence
min=1750 max=1826 mean=1808 amplitude=76 -> silence
min=1761 max=1839 mean=1808 amplitude=78 -> silence
min=1763 max=1828 mean=1809 amplitude=65 -> silence
L 5, col 22 WEMOS D1 MINI ESP32 sur COM43 2
```

- Si le mot SON DÉTECTÉ s'affiche tout le temps → tourner la molette vers la gauche
- Si rien ne se déclenche → tourner la molette vers la droite

2. Téléverser le fichier pour tester les LEDS

Test_LEDS_ESP32.ino

- Si ça marche, c'est bon. Si les 3 matrices ne s'allument pas, revérifier les soudures.



3. Téléverser le fichier pour tester le bouton poussoir

Test_Bouton_Poussoir_ESP32.ino

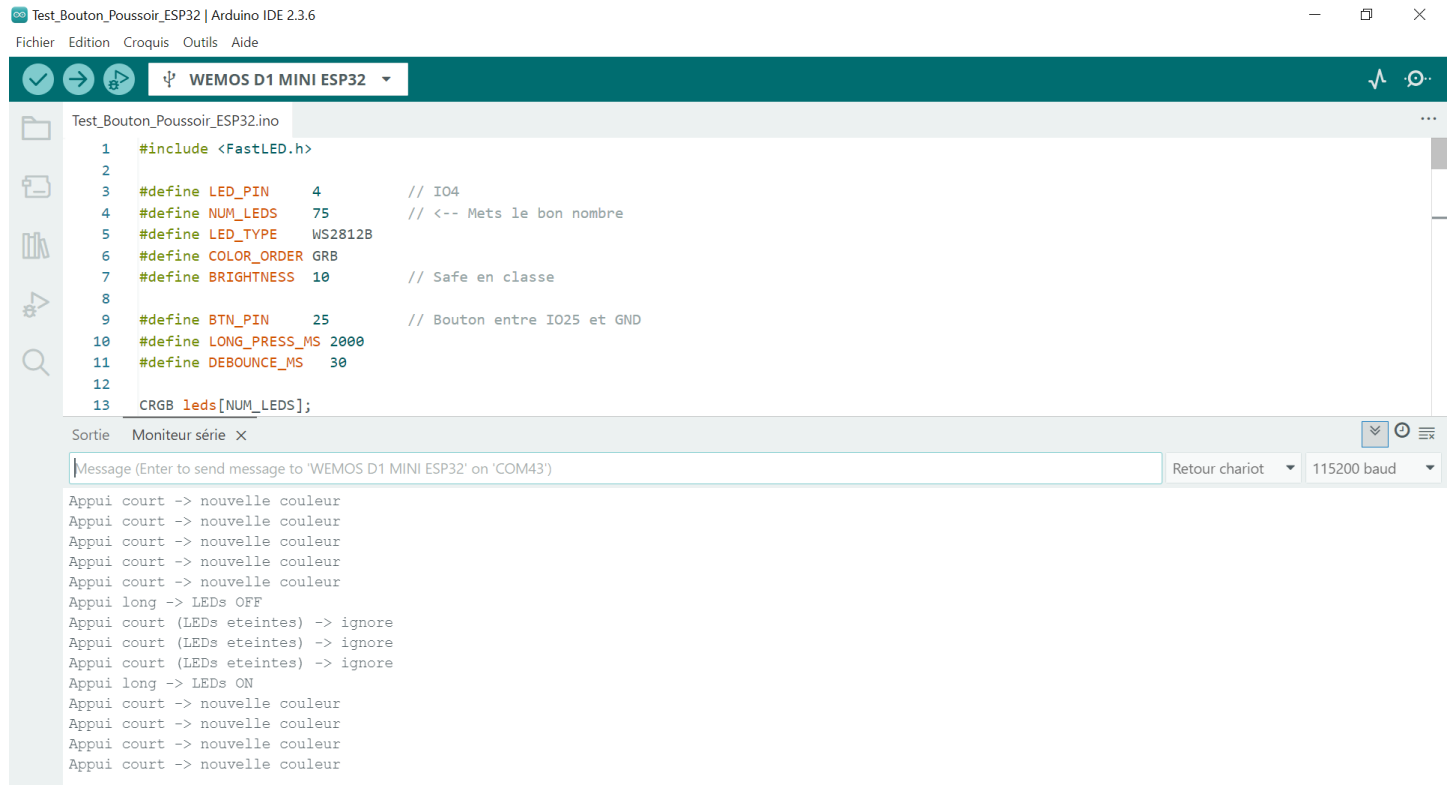
Explication du code test :

Au démarrage : **couleur aléatoire**

Appui court : change de couleur aléatoire

Appui long (≥ 2 s) : éteint les LEDs

Appui long (≥ 2 s) quand c'est éteint : rallume et met une nouvelle couleur



The screenshot shows the Arduino IDE interface. The top bar indicates the file is 'Test_Bouton_Poussoir_ESP32' in 'Arduino IDE 2.3.6'. The menu bar includes 'Fichier', 'Edition', 'Croquis', 'Outils', and 'Aide'. The toolbar shows icons for saving, running, and uploading. The main editor displays the code for 'Test_Bouton_Poussoir_ESP32.ino' with the following content:

```
1 #include <FastLED.h>
2
3 #define LED_PIN 4 // IO4
4 #define NUM_LEDS 75 // <-- Mets le bon nombre
5 #define LED_TYPE WS2812B
6 #define COLOR_ORDER GRB
7 #define BRIGHTNESS 10 // Safe en classe
8
9 #define BTN_PIN 25 // Bouton entre IO25 et GND
10 #define LONG_PRESS_MS 2000
11 #define DEBOUNCE_MS 30
12
13 CRGB leds[NUM_LEDS];
```

Below the code editor is the 'Sortie' (Output) window, which is currently set to 'Moniteur série' (Serial Monitor). The serial monitor shows the following output:

```
Appui court -> nouvelle couleur
Appui court -> nouvelle couleur
Appui court -> nouvelle couleur
Appui court -> nouvelle couleur
Appui court -> nouvelle couleur
Appui court -> nouvelle couleur
Appui long -> LEDs OFF
Appui court (LEDs éteintes) -> ignore
Appui court (LEDs éteintes) -> ignore
Appui court (LEDs éteintes) -> ignore
Appui long -> LEDs ON
Appui court -> nouvelle couleur
Appui court -> nouvelle couleur
Appui court -> nouvelle couleur
Appui court -> nouvelle couleur
```

Si ça ne marche pas, informer votre professeur.

Stage 7

On va Tester une version d'un programme on peut choisir différents effets lumineux grâce au bouton poussoir qui réagissent au son.

RZ75_V8.ino

Voici un code final complet pour D1 mini ESP32 avec les nouvelles broches :

- LED DATA → **IO4**
- Bouton → **IO25** (bouton entre IO25 et GND)
- Micro GY-MAX4466 → **IO34** (VCC 3.3V, GND, OUT IO34)
- Luminosité = **10**

- Appui court : mode suivant
- Appui long ≥ 2 s : ON/OFF LEDs

Stage 8 (Optionnel) (Durée : 1H00)

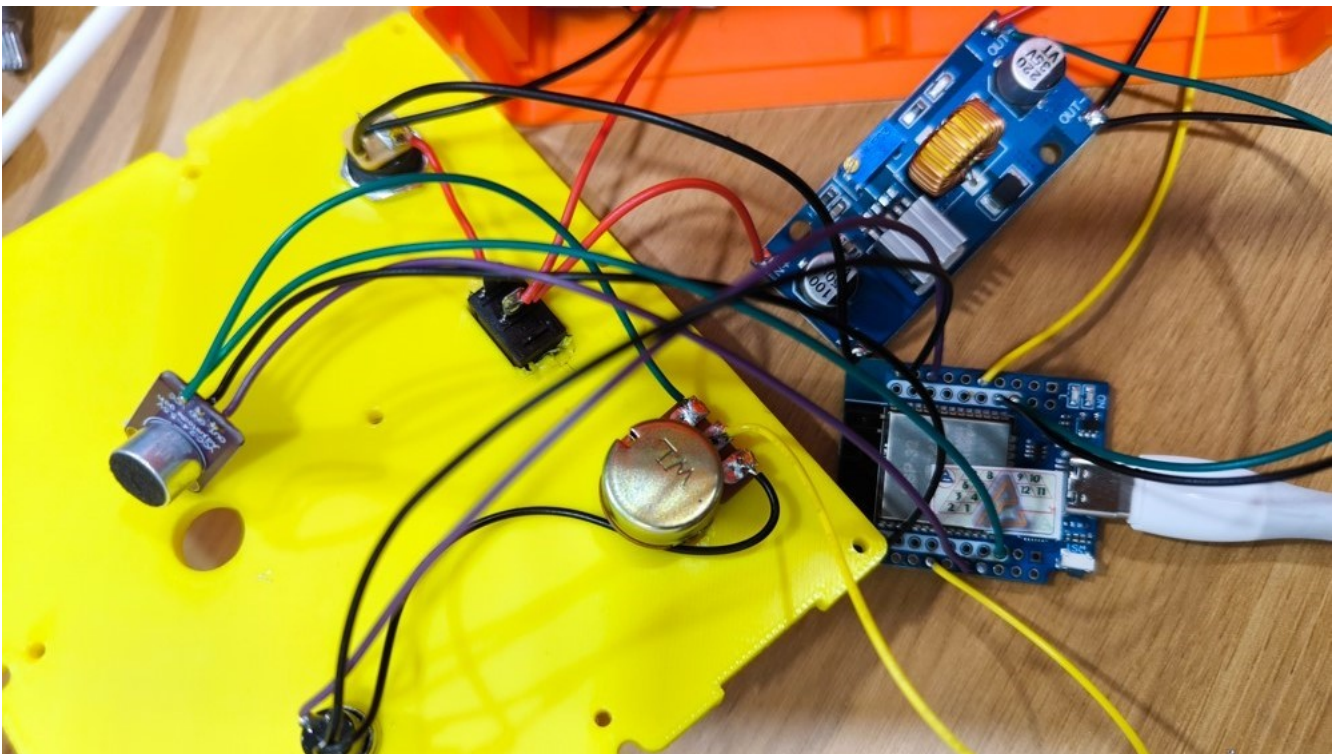
Installer le potentiomètre pour changer la luminosité

1. Nous allons alimenter le potentiomètre avec **3.3V** (une patte), **GND** (une patte) et **GPIO35** (patte du milieu) (**3.3V | GND | IO35**).

GND : 10 cm 28AWG, le connecter sur la patte GND du bouton poussoir

3.3V : 10 cm 28AWG, le connecter sur la patte VCC du microphone

IO35 : 15 cm 28AWG, le connecter en la patte du milieu du potentiomètre et IO35 de l'ESP32



2. Téléverser le fichier pour tester le potentiomètre

Test_Potentiometre.ino

Test_Potentiometre | Arduino IDE 2.3.6

Fichier Edition Croquis Outils Aide

WEMOS D1 MINI ESP32

```
Test_Potentiometre.ino
54 void loop() {
55   int brightness = readBrightness();
56   FastLED.setBrightness(brightness);
57   FastLED.show();
58 }
```

Sortie Moniteur série X

Message (Enter to send message to 'WEMOS D1 MINI ESP32' on 'COM46')

Retour chariot

```
ADC_raw=4095 | ADC_filt=4094 | Brightness=1 (0..120)
ADC_raw=4095 | ADC_filt=4094 | Brightness=1 (0..120)
ADC_raw=4095 | ADC_filt=4094 | Brightness=1 (0..120)
ADC_raw=4095 | ADC_filt=4094 | Brightness=1 (0..120)
ADC_raw=4095 | ADC_filt=4094 | Brightness=1 (0..120)
ADC_raw=4095 | ADC_filt=4094 | Brightness=1 (0..120)
ADC_raw=4095 | ADC_filt=4094 | Brightness=1 (0..120)
ADC_raw=4095 | ADC_filt=4094 | Brightness=1 (0..120)
ADC_raw=4095 | ADC_filt=4094 | Brightness=1 (0..120)
ADC_raw=4095 | ADC_filt=4094 | Brightness=1 (0..120)
ADC_raw=3965 | ADC_filt=4015 | Brightness=1 (0..120)
ADC_raw=3741 | ADC_filt=3649 | Brightness=13 (0..120)
ADC_raw=3312 | ADC_filt=3351 | Brightness=22 (0..120)
ADC_raw=3130 | ADC_filt=3155 | Brightness=28 (0..120)
ADC_raw=2897 | ADC_filt=2932 | Brightness=34 (0..120)
ADC_raw=2655 | ADC_filt=2687 | Brightness=40 (0..120)
ADC_raw=2480 | ADC_filt=2492 | Brightness=46 (0..120)
ADC_raw=2320 | ADC_filt=2337 | Brightness=52 (0..120)
ADC_raw=2191 | ADC_filt=2207 | Brightness=55 (0..120)
ADC_raw=2015 | ADC_filt=2041 | Brightness=61 (0..120)
ADC_raw=1856 | ADC_filt=1873 | Brightness=64 (0..120)
ADC_raw=1643 | ADC_filt=1664 | Brightness=70 (0..120)
ADC_raw=1514 | ADC_filt=1521 | Brightness=76 (0..120)
ADC_raw=1471 | ADC_filt=1475 | Brightness=76 (0..120)
ADC_raw=1462 | ADC_filt=1463 | Brightness=76 (0..120)
```

3. Téléverse le programme final de ce RZ75 avec Potentiomètre

RZ75_V8_Pot.ino

FIN