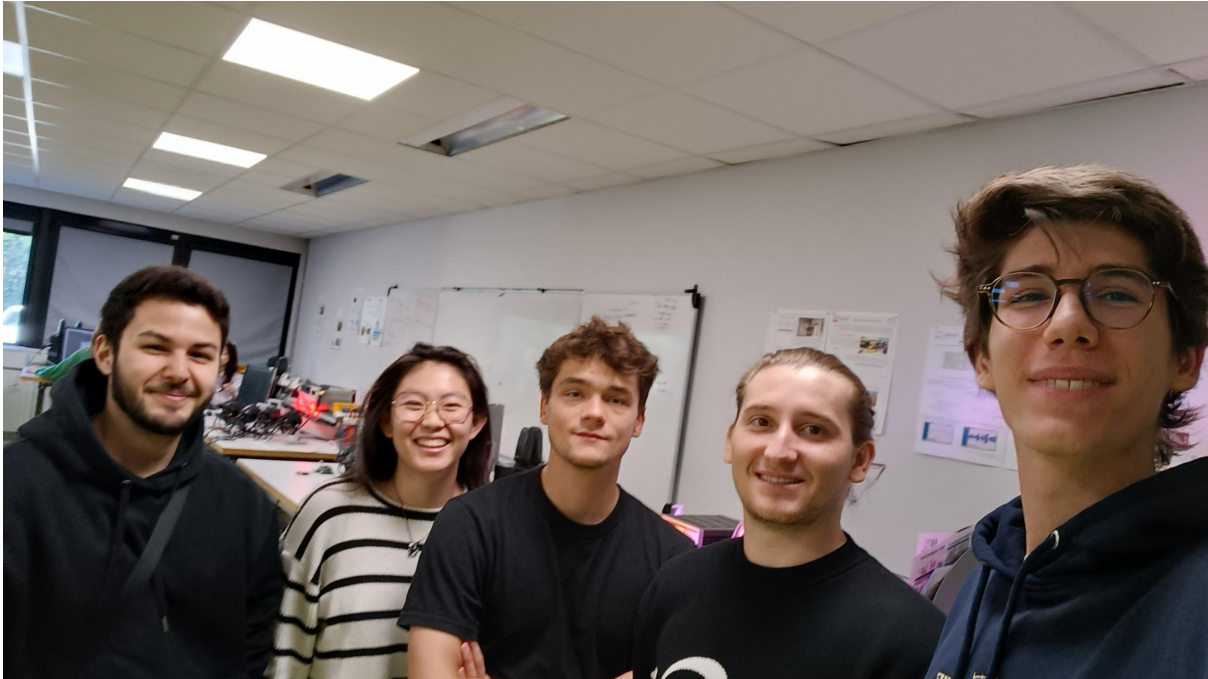


**Rabault Baptiste  
Hanoteau Qiusi  
Panabières Oscar  
Kessi Asyan  
Renouil Eliott**

## **Rapport Groupe 4 Equipe 3 BeatBox and Lights**



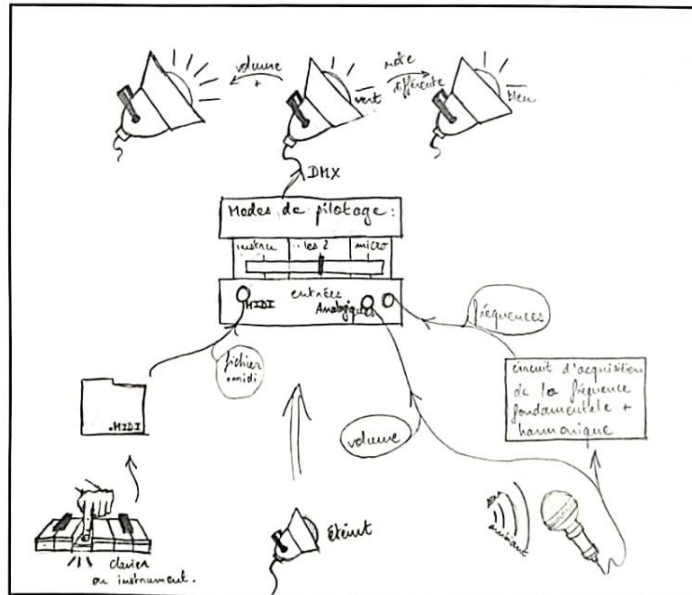
### **1. INTRODUCTION**

**OBJECTIF** : Piloter des projecteurs à l'aide d'un clavier (sortie MIDI) et/ou d'un micro captant le son ambiant (sortie analogique) L'intensité et la couleur de la lumière variant selon le volume et la note, respectivement.

#### **CAHIER DES CHARGES :**

- > Pour pilotage avec instrument :
  - commander différentes sources de lumières : DMX512, Lyre, Strobo
  - être utiliser de 7 à 77 ans
  - déterminer la luminosité en fonction de la force appliquée sur la touche
- > Pour un pilotage avec micro :
  - avoir une plage de détection de 0 à 110 dB
  - indicateurs lumineux si le volume ambiant est supérieur à 90dB pendant 30s
  - varier la luminosité en fonction du volume ambiant
  - varier la couleur en fonction des fréquences ambiantes
- > Modularité :
  - pouvoir être piloté selon 3 modes : instrument, micro, ou les 2

### **SCHÉMA DE PRINCIPE**

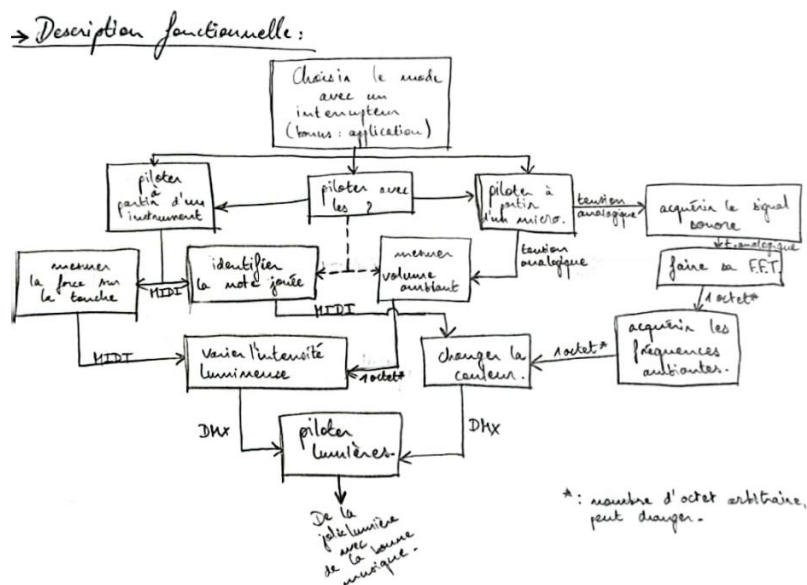


## NOTICE D'UTILISATION

L'utilisateur/ice (lumières non pointées sur le public)

- 1) sélectionne le mode : clavier, strobo, micro par l'intermédiaire du bouton se trouvant sur la carte nucléo (mode strobo quand mode clavier)
- 2) - Mode 1 CLAVIER : lorsque l'utilisateur joue du clavier, chaque touche permet aux lights d'émettre une couleur et une intensité
  - Mode 2 STROBO : régler la vitesse du strobo à l'aide du potentiomètre 1 du PAD
  - Mode 3 MICRO : lorsque l'utilisateur siffle près du micro les lights émettent une couleur selon la fréquence moyenne du son et une intensité lumineuse proportionnelle à l'intensité sonore

## SCHÉMA FONCTIONNEL



## 2. DETAILS DU CODE ET DES FONCTIONS

Le développement des systèmes d'acquisition ou de commande embarqué sur une carte STMicroelectronics Nucléo F767ZI. L'environnement de développement Keil Studio en ligne.

```
dmx_data[32]=1; //choix du mode/fonction ; ici couleurs par les biaux du canal 4-7
dmx_data[35]=255; //canal 4 correspondant controle luminosite
dmx_data[36]=100; //rouge
dmx_data[37]=100; //vert
dmx_data[38]=100; //bleu

//spot 2
dmx_data[40]=1;
dmx_data[43]=255;
dmx_data[44]=100;
dmx_data[45]=0;
dmx_data[46]=0;

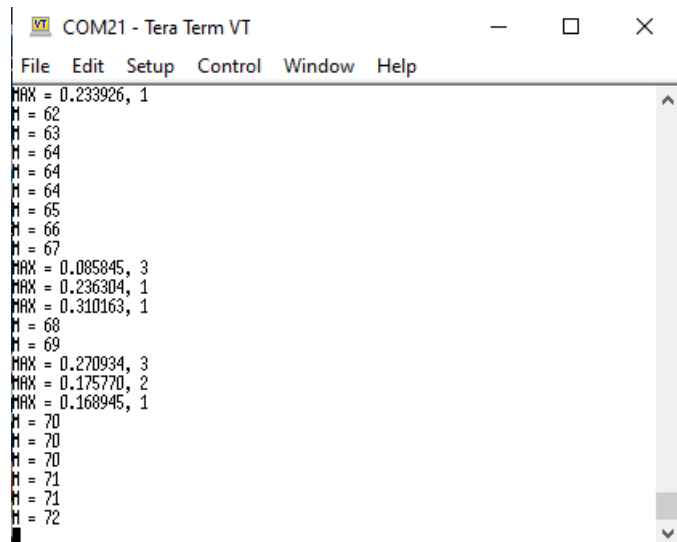
//spot 3
dmx_data[48]=1;
dmx_data[51]=255;
dmx_data[52]=0;
dmx_data[53]=100;
dmx_data[54]=0;

//spot 4
dmx_data[56]=1;
dmx_data[59]=255;
dmx_data[60]=0;
dmx_data[61]=0;
dmx_data[62]=100;
```

On commence par initialiser nos 4 spots. 1 spot correspond à 5 lignes de code. En effet, la première permet sélectionner le mode du spot. Ici, on veut le mode 1 qui permet de choisir la couleur du spot. La deuxième ligne permet de régler l'intensité. Pour finir, les trois dernières règlent la couleur.

Puis, on rentre dans le while (1)

On vient ensuite choisir le mode que l'on veut utiliser (le micro ou les instruments). Pour cela, on utilise un bouton poussoir qui ajoute +1 à la variable mode. Lorsque cette dernière est paire, seuls les instruments sont activés. A l'inverse, si elle est impaire, c'est le micro qui est activé. La variable du bouton poussoir est



```
COM21 - Tera Term VT
File Edit Setup Control Window Help
MAX = 0.233926, 1
M = 62
M = 63
M = 64
M = 64
M = 64
M = 65
M = 66
M = 67
MAX = 0.085845, 3
MAX = 0.236304, 1
MAX = 0.310163, 1
M = 68
M = 69
MAX = 0.270934, 3
MAX = 0.175770, 2
MAX = 0.168945, 1
M = 70
M = 70
M = 70
M = 71
M = 71
M = 72
```

Lorsque on est dans le cas où le piano fonctionne, on vient détecter si une touche du piano est enfoncée grâce à la fonction : isNoteMIDIdetected.

On a ensuite codé chaque touche du clavier. Pour la présentation en salle de TP, on a choisi de jouer le main theme d'Harry Potter. C'est pourquoi on a codé ces touches avec des couleurs bien spécifiques pour avoir un résultat harmonieux. Quant aux autres touches, on a décidé de laisser l'aléatoire choisir les couleurs des différents spots comme on peut le voir avec le code de la touche 31.

```
if(note_data == 0x31){
    dmx_data[32]=1;
    dmx_data[35]=255;
    dmx_data[36]=rand() % 256;
    dmx_data[37]=rand() % 256;
    dmx_data[38]=rand() % 256;

    dmx_data[40]=1;
    dmx_data[43]=255;
    dmx_data[44]=rand() % 256;
    dmx_data[45]=rand() % 256;
    dmx_data[46]=rand() % 256;

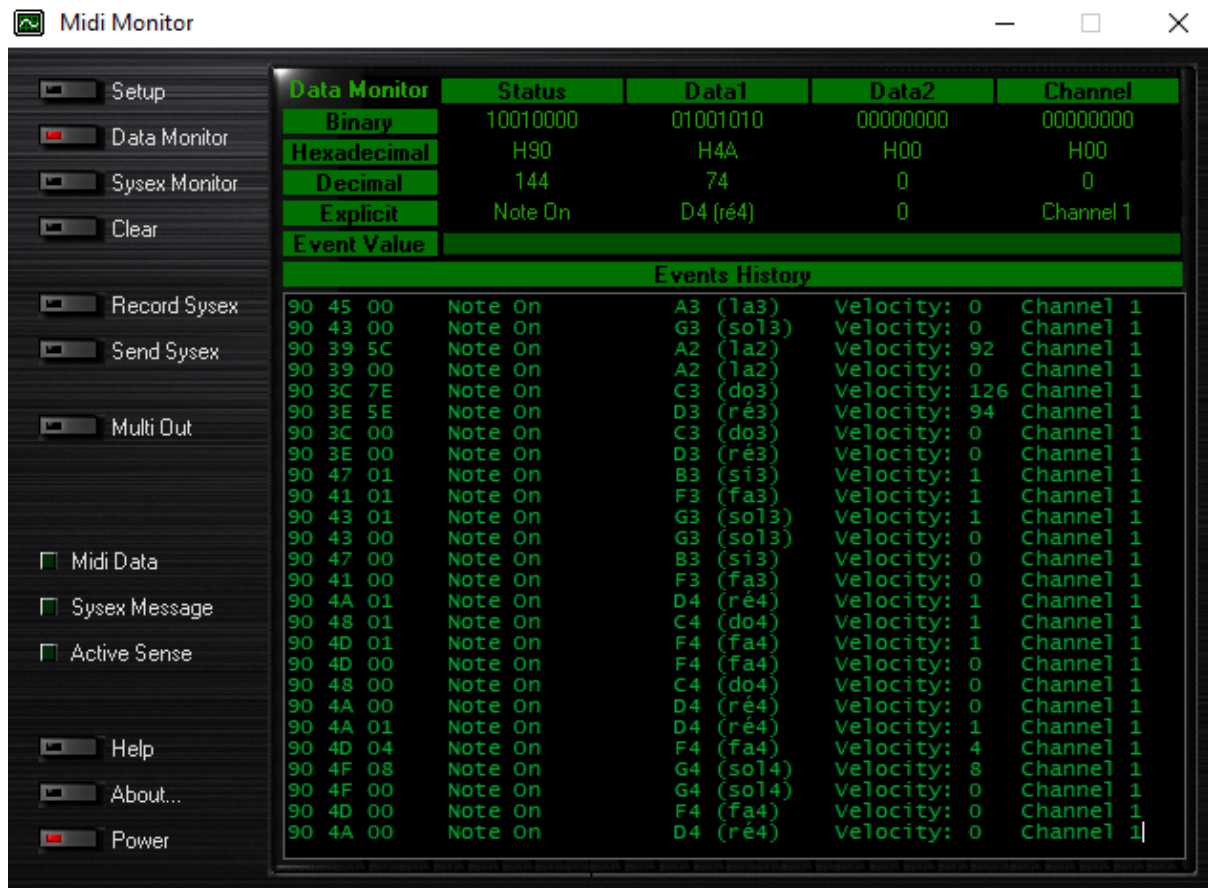
    dmx_data[48]=1;
    dmx_data[51]=255;
    dmx_data[52]=rand() % 256;
    dmx_data[53]=rand() % 256;
    dmx_data[54]=rand() % 256;

    dmx_data[56]=1;
    dmx_data[59]=255;
    dmx_data[60]=rand() % 256;
    dmx_data[61]=rand() % 256;
    dmx_data[62]=rand() % 256;
}
```

Les touches du clavier ont été codées en récupérant les valeurs de la deuxième colonne s'affichant du Midi Monitor. La commande associée à un code en hexadécimale:

- colonne 1 : statu byte de la Note on: 90 toujours pour le contrôleur
- colonne 2 valeur de la note
- colonne 3 : niveau d'intensité sur la touche : 00 touche relâchée et max 7E

Sur un accord, la touche de plus bas niveau est prioritaire, c'est la première détectée.



```

if(!ISCCMIDIDetected2()){
  //playNoteMIDI2(contrô1_ch2,contrô1_value2);

  if(contrô1_ch2 == 12){

    dmx_data[56] = 210;
    dmx_data[57] = 150;
    dmx_data[58] = 2 * contrô1_value2;

    dmx_data[48] = 210;
    dmx_data[49] = 150;
    dmx_data[50] = 2 * contrô1_value2;

    dmx_data[40] = 210;
    dmx_data[41] = 150;
    dmx_data[42] = 2 * contrô1_value2;

    dmx_data[32] = 210;
    dmx_data[33] = 150;
    dmx_data[34] = 2 * contrô1_value2;
  }
}

```

Après avoir fini le code pour le piano, on s'est dit que cela pourrait être sympathique d'utiliser les boutons du pad pour l'effet stroboscopique. Grâce à la fonction : ISCCMIDIDETECTED2, on détecte la présence ou non l'arrivée d'informations provenant du pad.

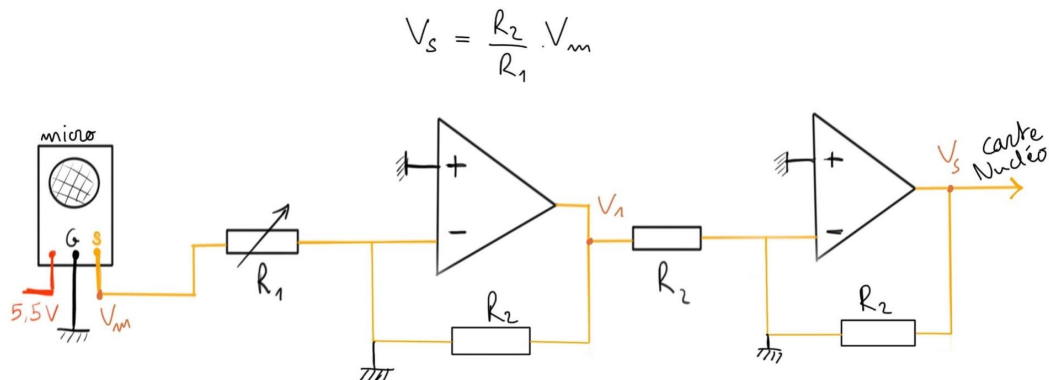
De la même manière que pour le piano, il faut 3 lignes de code pour les 4 spots. La première permet de choisir le

mode, ici 210 sélectionne le mode stroboscopique. La deuxième l'intensité et pour finir la dernière permet de contrôler la vitesse de l'effet en fonction de la position du bouton.

Après avoir codé les deux parties séparément, il a fallu passer sur une carte Nucléo a 2 entrées pour pouvoir utiliser le piano et le pad en même temps. Cette fusion a été à l'origine de nombreux problèmes informatiques que l'on a résolu au fur et à mesure des séances.

### 3. Description de la partie micro

#### a) Montage électrique



Voici le schéma électrique du micro. Le micro est alimenté par une tension continue de 5,5V. Ainsi, à saturation, le micro enverra une tension d'environ 5V, ce qui est trop important pour la carte nucléo. C'est pourquoi on ajoute un ALI, qui jouera le rôle d'amplificateur à coefficient variable grâce à une résistance à valeur variable (on utilisera un potentiomètre). Ainsi, la tension de sortie est adaptée au 3.3V que la carte nucléo est capable de prendre en entrée. Le deuxième ALI est un inverseur qui permet d'envoyer une tension positive dans la carte nucléo (car la tension est devenue négative à cause du deuxième ALI).

#### b) Acquisition de la fréquence et amplitude moyennes:

Afin d'éviter toute redondance avec le sujet .zip contenant le code entier, nous ne détaillons dans ce qui suit seulement les points principaux du code (nous omettons donc les définitions et les parties liées au débogage).

```
void sample(){
    myled = 1;
    if(indice < SAMPLES){
        Input[indice] = myADC.read() - 0.5f;
        Input[indice + 1] = 0;
        indice += 2;
    }
    else{ trig = 0; }
    myled = 0;
}
```

Commençons par la fonction Sample, celle-ci a pour but de remplir le tableau Input contenant les échantillons du signal de sortie du montage électrique décrit précédemment. Le tableau Input s'organise de la sorte: 2 cases sont allouées au même échantillon, la première pour la partie réelle (à laquelle on enlève l'offset) et la deuxième pour la partie imaginaire

(égale à 0 forcément, car le signal est réel). La variable myled ne sert qu'au débogage afin

de vérifier que la prise d'échantillon se fait bien. A noter que le remplissage de ce tableau ne se fait que si la variable trig = 1.

```
arm_cmplx_mag_f32(Input, Output, FFT_SIZE);
```

On utilise ensuite l'instruction ci-dessus pour créer le tableau Output, représentant la FFT du tableau Input, dans lequel est stockée l'amplitude de chaque composante fréquentielle du signal.

```
for(int i = 0; i < SAMPLES; i = i+2){  
    amp_moy = amp_moy + 2*Input[i]/SAMPLES;  
}
```

On acquiert alors l'amplitude moyenne à partir du tableau Input et d'une simple boucle for.

Ensuite, on réalise une moyenne pondérée par l'amplitude de chaque composante fréquentielle pour récupérer la fréquence moyenne.

```
//Acq Somme  
for(int i = 0; i < FFT_SIZE; i++){  
    somme = somme + Output[i];  
}  
//Acq fmoy  
for(int i=0; i < FFT_SIZE; i++){  
    fmoy = Output[i] * i*pas /somme;  
}
```

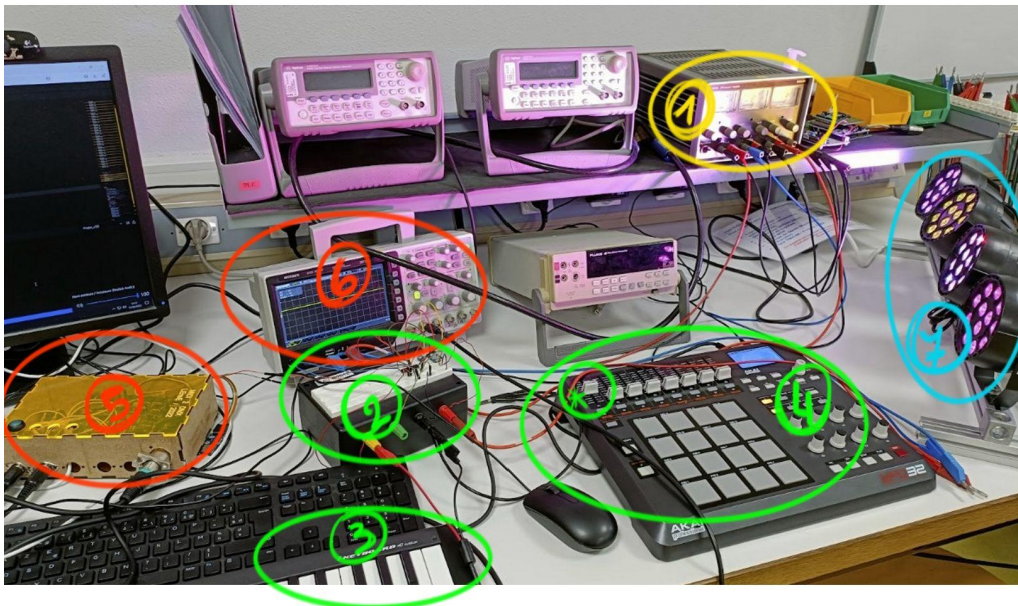
```
trig = 1;  
indice = 0;  
timer.attach_us(&sample, 40);
```

Finalement, on réinitialise la variable indice (sert à l'indentation de la fonction sample) et on incrémente la variable trig afin de relancer une acquisition d'échantillons du signal de sortie du micro.

### 3. BILAN

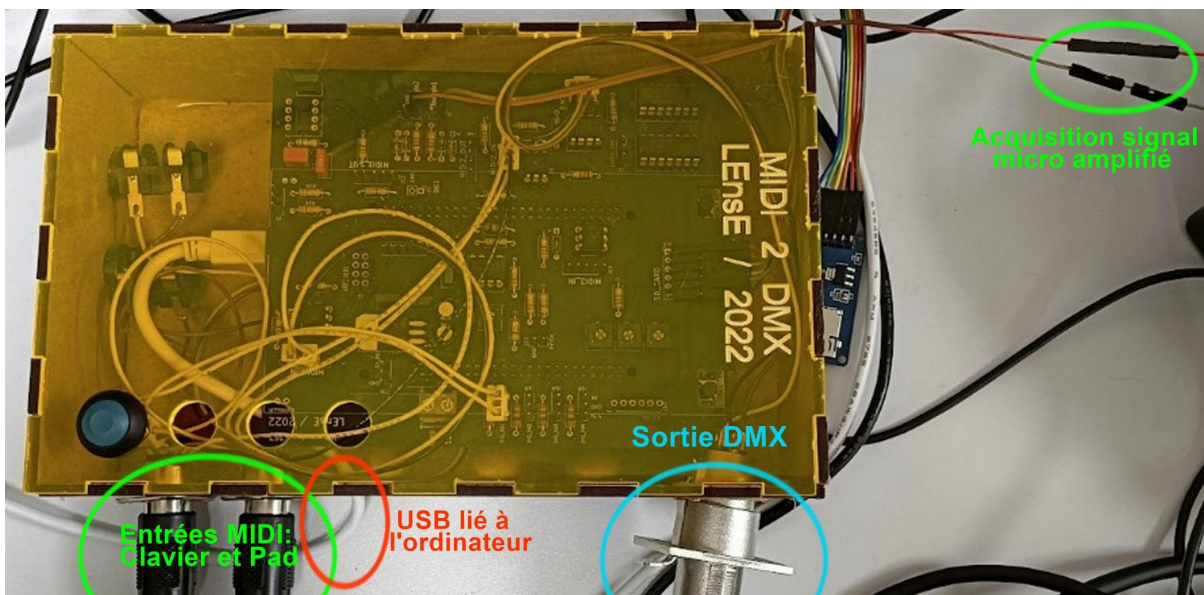
#### VUE D'ENSEMBLE

Voici une vision d'ensemble du projet avec les différents éléments

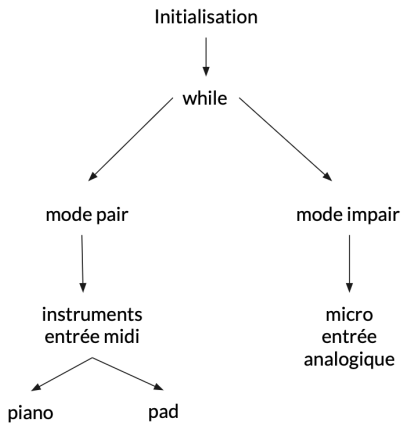


- 1- Alimentation stabilisée
- Entrées:
  - 2- Circuit micro amplifié
  - 3- Clavier maître Icon iKeyBoard
  - 4- Contrôleur MIDI Akai MPD26 (\*: contrôleur stroboscope)
- Contrôle et traitement:
  - 5- Carte Nucléo F767ZI
  - 6- Oscilloscope
- Sortie:
  - 7- DMX: Renkforce LVPT12/ RGBW-15W

Faisons un petit zoom sur la **carte Nucléo F767ZI**. Elle présente, d'une part, deux entrées Midi permettant de la relier au piano et au pad, et d'autre part une entrée analogique pour connecter le micro. Enfin, la sortie DMX fait le lien avec les spots.



#### VUE D'ENSEMBLE DE L'ARCHITECTURE DU PROGRAMME



## PISTES D'AMÉLIORATION

- Application pour contrôler les couleurs etc.
- Bouton switch ou à gradation pour utiliser les deux modes ensembles ou séparément
- Pour le piano, associer une couleur au différents accords
- Le faire pour d'autres instruments (guitare électrique, harpe laser...)
- D'autres dispositifs d'éclairage

## RETROPLANNING et REPARTITION DES TACHES

Dates des séances:

17/05: Finalisation du projet

- Fin du code et vérification du bon fonctionnement
- Rédaction des livrables et conception de la vidéo de démonstration

10/05: Réunionification des 2 équipes

- On a fini de résoudre les problèmes sur le code lié aux entrées du piano
- On a travaillé sur la liaison entre la nouvelle carte et le micro
- difficulté technique rencontrée: problème avec l'acquisition de la fréquence moyenne
- Résolution du problème d'acquisition de la fréquence moyenne

17/04:

- équipe clavier:
  - Passage de deux cartes à une entrée à une carte à deux entrées
  - Difficulté technique rencontrée : Il fallait donc changer tous les paramètres d'entrées ce qui a causé des problèmes au niveau du code
  - On a donc commencé à régler ces problèmes durant la séance
- équipe micro:
  - rédaction et analyse du code permettant l'acquisition de la fréquence et amplitude moyennes du signal de sortie du micro amplifié.

03/04:

- équipe clavier:



- Finalisation de la partie avec un sous-projet fonctionnel.
- équipe micro:
  - Reprise des étapes précédentes avec un micro fonctionnel
  - > tout est bon pour la partie montage électrique

27/03:

- équipe clavier:
  - Collaboration avec Mr Villemejeane pour comprendre les codes laissés par les années précédentes.
  - Poursuite du code pour notre projet.
- équipe micro:
  - Test des différents montages pensés pour l'amplification de la sortie du micro, sans véritable succès
  - > En fait, le micro fourni ne fonctionnait juste pas.

08/03:

- Equipe clavier:
  - Découverte et prise en main de Keil Studio
  - Premier code fonctionnel pour la liaison MIDI DMX d'un spot
- Equipe micro:
  - Prise en main du micro
  - Recherche d'un montage amplificateur adapté à la carte nucléo

27/02: Séance d'organisation du projet

- Découpage du projet en 2 sections:
- Répartition des tâches avec la création de deux sous-équipes, l'une en charge de l'acquisition sonore par micro composée de Asyan et Eliott et l'autre en charge de la partie de production de lumière à partir d'un clavier
- Mise en place d'un plan d'action pour chacune des deux sous-équipes

13/02: Découverte du projet Beatbox and lights

- Rédaction du descriptif et scénario d'usage
- Prise en main du matériel à disposition

## **COMPÉTENCES ACQUISES**

1. Collaboration et travail d'équipe et communication
2. Gestion du temps et organisation: Planifier et organiser les différentes étapes du projet, établir des échéances,
3. Résolution de problèmes et adaptabilité, autonomie : Être capable d'identifier un problème de s'adapter aux ajustements nécessaires tout au long du projet,
4. Compétences techniques : langage C, électronique, établir et suivre un cahier des charges