

Projet JukeBeatBox Music

Pluche Cyril
Thevenon Romain
IG3 - Groupe 1

I) Introduction

Qui n'a jamais eu envie d'éveiller en lui sa créativité, de devenir un pro de la musique ?

C'est l'idée que nous avons eu envie de créer, d'inventer une toute nouvelle plateforme de musique.

En effet, durant ce semestre, notre objectif était d'inventer un objet à partir d'un microcontrôleur. Nous avons donc décidé de partir sur l'invention d'une plateforme de BeatBox & LaserRoll Music. Cette dernière devait permettre à l'utilisateur de créer des sons en appuyant sur des fruits ou en passant sa main devant un faisceau lumineux. Elle avait pour objectif de divertir l'utilisateur et qui sait lui éveiller une nouvelle passion : la musique. Cependant, après l'annonce du prix donné par l'école pour ce projet et des trois seuls sites de commandes de capteurs autorisés, nous avons été contraints de revoir l'objectif que nous nous étions fixés.

Nous avons donc décidés par la suite, de continuer sur le thème de la musique mais en créant notre propre JukeBeatBox Music, en effet, elle permettra à l'utilisateur de lancer différentes musique stockés dans notre JukeBeatBox (créé par nous-mêmes) et également lui permettre de créer lui-même des sons.

Vous trouverez par la suite, tout le manuel de construction de notre projet.

II) Description du projet

A) Mode d'emploi

Pour pouvoir faire fonctionner notre projet, il faudra juste brancher l'Arduino à l'ordinateur et de vous laisser guider. En effet, tous les branchements seront déjà établis et cachés dans le boîtier de notre JukeBeatBox. Cependant, si vous voulez modifier le programme ou que ce dernier ne lance pas automatiquement il vous faudra télécharger et installer le logiciel Arduino ainsi que la librairie LiquidCrystal (expliqué dans la suite) et de lancer le programme principal.

B) Les moyens matériels

Pour effectuer notre projet, nous avons eu besoin :

- Un arduino avec son shield, en effet nous avons décidés de partir sur ce microcontrôleur, puisqu'il est adapté à notre projet.

- Un écran lcd, ce qui va permettre d'interagir avec l'utilisateur.
- des boutons (nous voulions partir sur des capteurs de mouvements cependant nous avons eu un énorme problème avec ceux-ci que l'on détaillera par la suite), ceux-ci vont permettre à l'utilisateur de gérer les différentes musiques.
- Des leds qui vont permettre d'avoir un meilleur rendu esthétique.
- Un buzzer qui lui va permettre de diffuser le son émis par l'arduino.

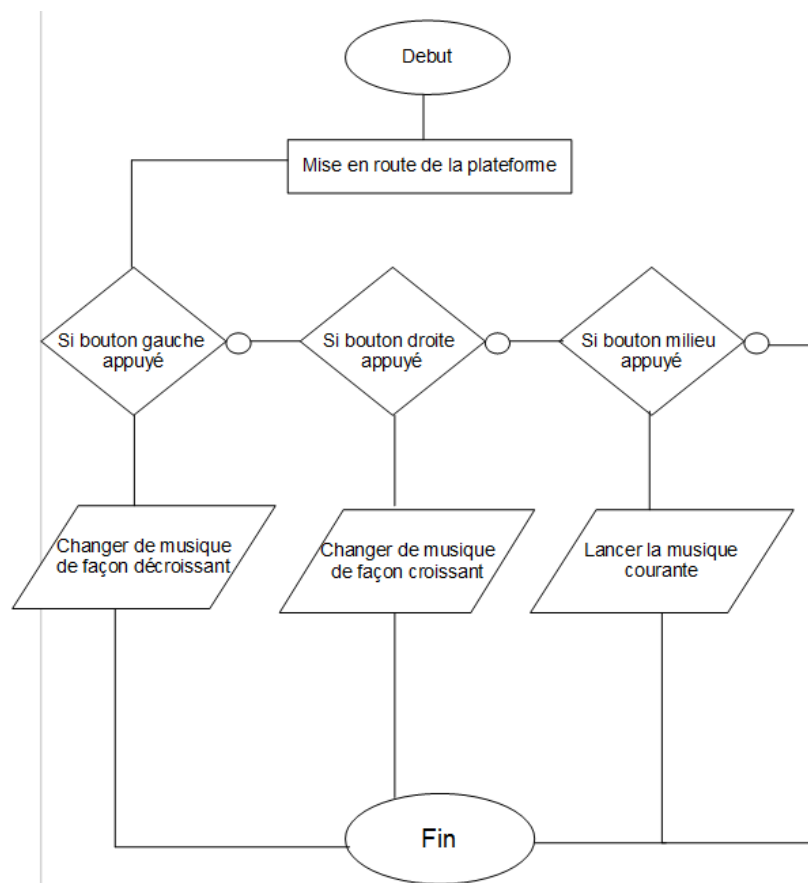
C) Les moyens humains

Concernant l'organisation du travail, nous avons effectué le travail ensemble pour la plus grande partie. Nous avons effectué la majorité du projet à polytech, vu qu'il y avait tous les outils nécessaires à la conception.

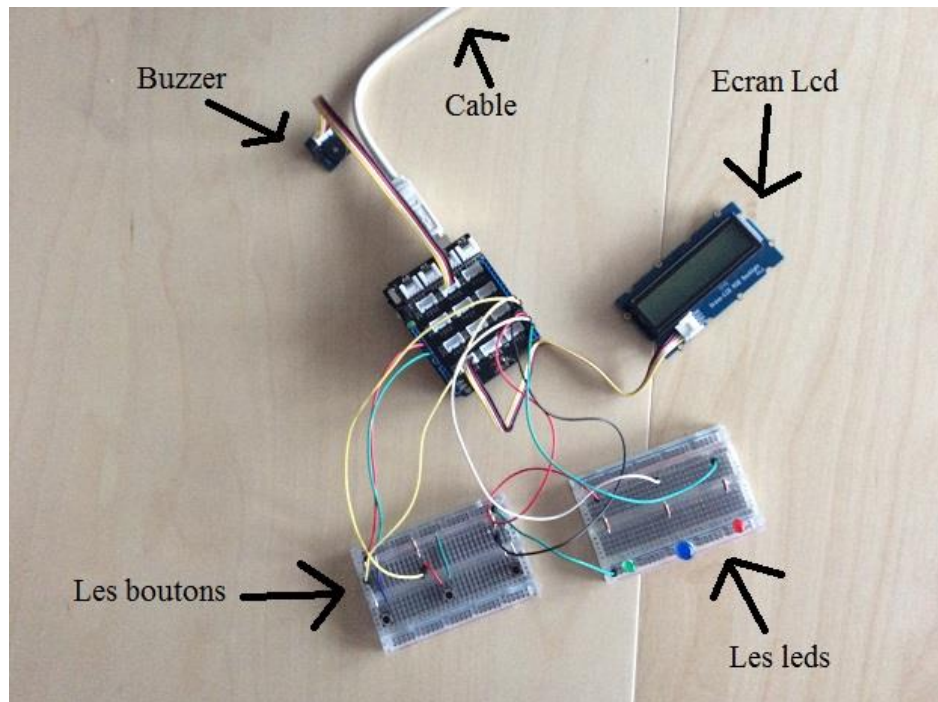
D) Architecture du projet

Au niveau de l'architecture des fichiers, nous avons deux fichiers contenus dans le même dossier, l'un contient tous le code de notre JukeBeatBox avec plusieurs fonctions (que nous détaillerons par la suite) et un fichier s'intitulant pitches.h et qui contient les différentes notes permettant de jouer une mélodie (que l'on détaillera également par la suite).

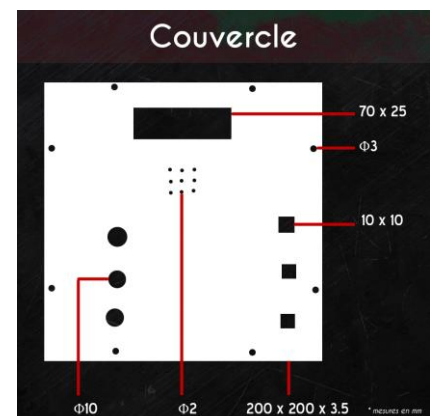
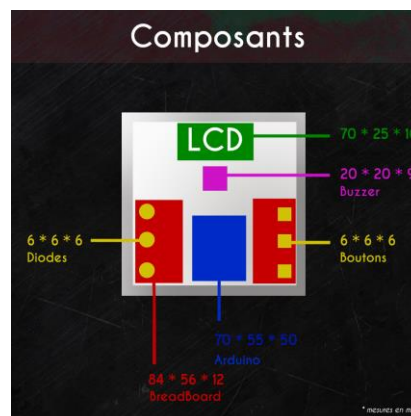
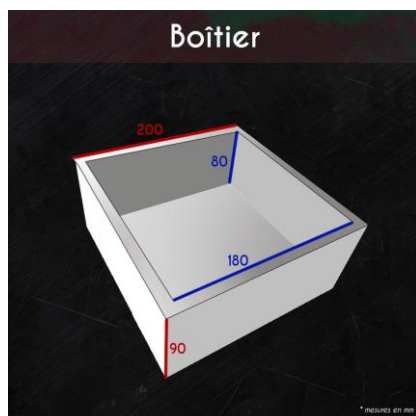
Voici un algorithme permettant de voir comment fonctionne notre projet :



Au niveau de l'architecture matérielle voici les différents composants et leurs branchements :



Par la suite, nous avons décidés de construire un petit boîtier pour rendre notre JukeBeatBox beaucoup plus esthétique, voici les différents schémas que nous avons conçus avant la fabrication de ce dernier :



E) Description du code et des différents branchements

Nous allons commencer ce paragraphe, en expliquant les différents branchements électriques en particulier pour les boutons et les leds. En effet, nous avons été dans l'obligation, d'utiliser des breadboards, et différentes résistances permettant d'ajuster le circuit de façon à ce que chaque composant ait ce dont il a besoin.

Calcul de la résistance pour la led : on utilise pour cela la formule $U=I \cdot R$ avec $U = 5V - 3V = 2V$ et $I = 20mA = 0.02A$

On a donc $R = 2V / 0.02A = 100 \text{ Ohms}$.

Sachant que l'on a trois boutons, nous avons donc besoin de trois résistances de 100Ohms.

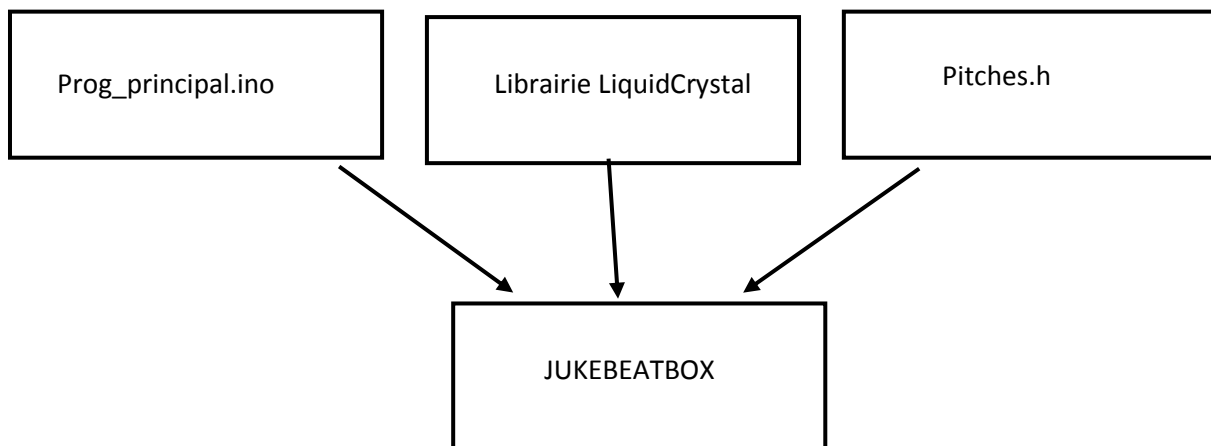
On utilise la même formule pour les boutons poussoirs, et on obtient qu'il nous faille trois résistances de 10kOhms.

Maintenant, au niveau du code, nous avons rassemblé nos deux fichiers dans un même dossier. L'un des deux fichiers contient tout le programme, avec 5 fonctions **void son1()** (pour le son1) qui va permettre de créer une des musiques. Il contient également la fonction **void son(int i)** qui va permettre de lancer la musique qui est donné en paramètre. Et enfin les fonctions **void setup**, **void loop** qui sont les deux fonctions principal (équivalent au main en C).

Tandis que l'autre, pitches.h, ne contient juste les notes de musiques qui sont utiles pour créer nous-même nos musiques.

De plus nous avons utilisé la librairie LiquidCrystal qui va permettre d'utiliser l'écran Lcd du pack GrovePi.

Au niveau des sources, à part l'utilisation du github de la librairie LiquidCrystal : <https://github.com/arduino/Arduino/tree/master/libraries/LiquidCrystal> et de quelques tuto sur le branchement électrique, nous n'avons utilisé aucune autre source, nous avons réussi à réaliser ce dernier par nous-mêmes.



F) Perspectives

Si nous avions eu le temps, et bien sûr le budget, nous aurions bien aimé rester sur notre idée première qui été de créer une plateforme de musique avec des fruits et des faisceaux lumineux.

Maintenant sur le projet réalisé, l'une des grandes perspectives serait de pouvoir rajouter une application connecté avec notre JueBeatBox permettant à l'utilisateur de lancer ces propres musiques depuis son smartphone pendant il faudrait changer le buzzer en une enceinte.

L'une des autres perspectives, seraient de pouvoir laisser l'opportunité à l'utilisateur de créer lui-même ses propres musiques avec d'autres boutons ou des capteurs de mouvements. Pour ses différentes perspectives, cela ne devrait pas prendre énormément de temps, mais il faudra acheter de nouveaux capteurs.