**Laurent Tremblay**

*Gr.001*

**Projet final**

*Pédale numérique pour effets de guitare*

Travaille présenté à

**M. Jasmin St-Laurent, M. Claude Barbaud et Mme. Chantal Milette**

dans le cadre du cours

*Projet de fin d’études, 243-631-MA*

**Collège Maisonneuve**

2018-05-11

Table des matières

**Aucune entrée de table des matières n'a été trouvée.**

**Introduction**

Dans le cadre du cours de projet de fin d’études, j’ai réalisé un projet électronique ayant comme fonction de modifié un signal audio provenant d’une guitare à l’aide d’un FPGA. Celui-ci possèderait une interface graphique pour afficher à l’utilisateur l’effet actif ainsi que les paramètres modifiables des effets par des potentiomètres externes. L’utilisateur aurait à sa disposition deux boutons poussoirs pour avancer et reculer entre les multiples effets ainsi qu’un bouton pour activer les effets audios ou non. Parmi ces effets, on peut retrouver un effet de distorsion et un effet de trémolo. L’effet de distorsion consiste à limiter le signal audio pour ajouter des harmoniques à celui-ci, et l’effet de trémolo consiste à moduler le signal audio à l’aide d’une onde de fréquence très basse, ce qui donne l’impression qu’on augmente et réduit automatiquement le volume du son.

J’ai choisi ce projet car je joue de la guitare dans mon temps libre et qu’il est important d’avoir des pédales pour modifier le son de la guitare. Il y a aussi le fait que les pédales à effets pour guitare sont très couteuses, car une pédale peut coûter entre 75 et 500 dollars, et celle-ci ne produit qu’un seul effet. On peut donc s’imaginé qu’on peut se retrouver avec une grosse facture seulement en possédant quelques effets.

L’objectif du projet était donc de pouvoir créer tous les effets voulus pour une fraction du prix que l’ensemble des pédales standard pourraient coutés. De plus, ce projet me donnait raison d’utilisé un FPGA, car ceux-ci sont extrêmement puissant pour le traitement de signaux numériques, et par ce fait approfondir mes connaissances au niveau de la composante et du langage VHDL.

**Bilan des travaux**

\*\* bilan a la fin

**Description générale**

Schéma synoptique

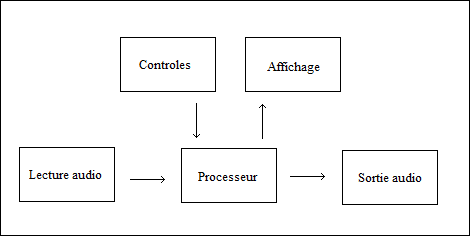
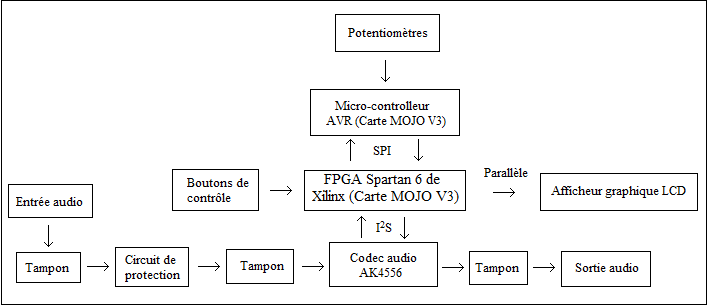


Schéma de principe



Il y a plusieurs principes utilisés dans ce projet. Ceux-ci peuvent être séparés en deux : soit une partie analogique et une partie numérique. Pour la partie analogique, il y a l’utilisation de l’amplificateur TL04 comme tampon et comme circuit de protection pour transmettre le signal audio au codec audio AK4556. Il est important d’utilisé un circuit tampon pour isoler les impédances, soit celle de l’entrée audio, celle de l’entrée du circuit d’acquisition du codec audio et celle à la sortie du codec audio. De plus, il faut aussi utiliser un circuit de protection pour limiter le niveau de tension de l’entrée audio, car elle pourrait endommager l’entrée du circuit d’acquisition du codec. Dans mon cas, j’ai choisi d’utilisé un circuit de compression audio : ce circuit ajuste automatiquement la tension du signal audio pour qu’elle soit à une tension fixe, prévenant les surtensions. En plus, ce choix permet d’avoir une meilleure résolution de notre signal audio, cependant il vient couper la dynamique du signal audio. Le deuxième aspect de la partie analogique est l’utilisation d’un codec Audio, celui-ci permettant le sur-échantillonnage et le filtrage du signal audio. La communication entre le codec Audio et le FPGA se fait à l’aide du protocole I2S.

Pour la partie numérique, on peut retrouver une interface pour l’utilisateur ainsi que plusieurs protocoles de communications. L’utilisateur possède trois boutons de contrôles, soit deux pour changer de pages (précédent et prochain) ainsi qu’un bouton activant la pédale. L’utilisateur peut aussi modifier les paramètres des effets à l’aide de trois potentiomètres. Les tensions de ces potentiomètres sont lues à l’aide des convertisseurs analogique à numérique du microcontrôleur AVR de la carte MOJO V3 et sont ensuite transmit via le protocole SPI au FPGA Spartan 6. Le dernier item de l’interface utilisateur est un afficheur graphique LCD. Le FPGA envoi des données en parallèle à l’afficheur étant donné qu’on ne manque pas de broches, que c’est plus simple et que c’est rapide.

**Description précise**

**Procédure d’assemblage**

**Procédure de calibration**

**Manuel de l’utilisateur**

**Conclusion**

**Annexes**