

Vumètre fréquentiel

N° projet 2225 - Version B

Mandataire

Entreprise/Client:	ETML-ES	Département:	SLO
Demandé par (Prénom, Nom):	Ph. Bovey	Date:	05.08.2024

1 Objectif - Cahier des charges

Le but de ce travail de diplôme est de réaliser un vumètre fréquentiel permettant de visualiser des fréquences allant de 20 à 20kHz, représentant ainsi le spectre auditif moyen. Le système doit permettre un « échantillonnage » des fréquences (gamme voulue : 20 – 50 – 100 – 200 – 500 – 1k – 2k - ... 20kHz) avec une visualisation de l'amplitude.

Le système sera vu comme une boîte noire, dans laquelle on « injecte » un signal (entrée) et celui-ci ressort sans être modifié (*true bypass*¹ (utilisé pour les pédales à effet - guitare)), et une 2^{ème} sortie sera prévue pour mesurer les effets des filtres numériques implémenter sur le microcontrôleur (uC).

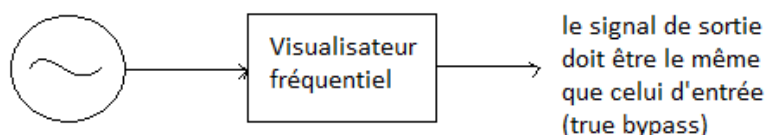


Figure 1 : Boîte Noire - mode True ByPass

La visualisation des gammes de fréquence se fera à l'aide de leds RGB (une Led par gamme de fréquence), avec une représentation du niveau d'amplitude (échelle en dB) en couleur (voir représentation ci-contre). Il faut prévoir aussi une connexion avec le projet de Matrice de Led (n°2126), uniquement la matrice, sans son module commande.

L'entrée du système sera équipée d'un connecteur de type jack 3.5 – femelle, permettant de brancher un appareil « grand public » (baladeur MP3, chaîne Hifi, autres dispositifs audio), et d'un autre type de connecteur pour brancher un générateur de fonction.

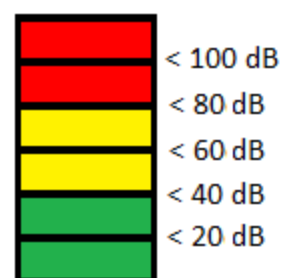


Figure 2 : représentation visuelle pour une gamme de fréquence

¹ Explication *true bypass* : https://cnzaudio.com/blogs/cnz-audio/true-bypass-explained?srltid=AfmBOopIIF9mRmQoZ_JxePvBUIE_AveI99TC7gUcxyAUjaWTMr1yOMdV

Le système comportera plusieurs sorties :

- Une sortie avec un DAC et une sortie audio (pour visualiser l'effet des filtres numériques mis en place)
- Une sortie sans traitement de signal (mode *true bypass*)

Les sorties seront aussi équipées de connecteur jack, voir d'autres connecteurs pour brancher facilement des appareils de mesure (ex : oscilloscope)

Le diplômé devra prévoir une communication USB-UART à l'aide d'un chip **FTDI**² permettant de modifier par la suite le type de filtrage (ordre des filtres choisis – constantes pour ceux-ci) ou de sélectionner le système d'affichage voulu (leds RGB ou matrice).

Le système sera alimenté par USB (prévoir un connecteur USB-C).

Le diplômé peut s'inspirer de la version A, tout en sachant que cette version n'est pas fonctionnelle et que le microcontrôleur choisi dans la version précédente n'est pas adéquat.

Partie Hardware

Le diplômé devra concevoir une carte électronique basé sur un microcontrôleur de chez Microchip ; cette carte doit contenir au minimum :

- Microcontrôleur – Fabricant Microchip – famille PIC32³
- Alimentation
- Leds RGB – low current
- ADC – étage d'entrée (résolution possible)
- DAC – avec étage de sortie audio (amplification)
- Sortie sans traitement de signal (*true bypass*)
- Communication UART via USB
- Communication avec matrice à leds (projet 2126)
- ...

Il devra soit choisir un boîtier du commerce et l'adapter (design des ouvertures pour leds – connecteur entrée/sortie – alimentation, etc...) ou le designer lui-même.

Partie Firmware

L'étudiant réalisera un Firmware (partie microcontrôleur) permettant de faire au minimum les tâches suivantes :

- Algorithme permettant de filtrer un signal à une fréquence voulue (Filtrage ou **FFT**⁴)
- Commande LEDs RGB
- Communication entre uC et PC (extérieur au système)
- Communication entre uC et DAC externe
- Sauvegarde paramètres reçues dans la mémoire du uC
- Commande des matrices de LEDs (projet 2126) – **Point Optionnel**
- ...

² Chip FTDI : <https://ftdichip.com/product-category/products/ic/>

³ Microchip – PIC32 (famille) : <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU32/ProductDocuments/Brochures/32-bit-Brochure-30009904.pdf>

⁴ FFT : Fast Fourier Transform

Partie Software

Le diplômé devra réaliser une petite application basique en C# permettant de transférer des données sur le système

- Envoi de constante(s) pour calcul et ordre de filtrage à appliquer
- Choix du type d'affichage que l'on veut sélectionner
- ...

1.1 Données en lien avec l'objectif – les grandes lignes

- Recherche et implémentation de solutions **Hardware** :
 - Microcontrôleur (fabriquant Microchip)
 - Définir un uC permettant de faire du traitement signal audio
 - Voir famille PIC32MZ
 - ...
 - Alimentation
 - Par câble USB-C (externe)
 - ...
 - Attention** : les tensions subsidiaires au montage (interne) devront être réalisées par des alimentations à découpage DC-DC
 - Leds RGB
 - Déterminer le nb de leds par gamme de fréquence
 - Led type - low current
 - Possibilité leds déportées
 - Prévoir connectique et câblage
 - ...
 - Convertisseur : ADC – A définir
 - Externe ou interne au uC
 - Plage de tension
 - Echantillonnage – résolution
 - ...
 - Convertisseur : DAC – A définir
 - Externe ou interne au uC
 - Plage de tension
 - Echantillonnage – résolution
 - ...
 - Entrée(s)
 - Filtrage
 - Niveau de tension admissible
 - ...
 - Sortie(s)
 - Le signal d'entrée non altéré en sortie
 - Niveau de tension en sortie
 - Connecteurs à définir
 - ...
 - Module FTDI (chip)
 - Communication UART – USB
 - Communication avec le uC
 - ...
 - Connexion avec matrice à Leds (projet 2126)
 - Communication avec le uC
 - Type de connecteur
 - ...
- Recherche de composants électroniques autour des composants décrits ci-dessus : AOP, résistances, condensateur, self, autres
- Réalisation de schématique(s) complet(s) et d'un PCB - à réaliser sous ALTIUM de préférence
- Recherche (achat) ou réalisation d'un boîtier sous SolidWorks

- Recherche et implémentation de solutions au niveau **Firmware**
 - Algorithmes
 - Filtrage numérique – minimum premier ordre
 - FFT
 - Entre ces deux traitements de signaux, voir ce qui est plus adéquat / facile à mettre en œuvre
 - Rapport amplitude (signal de sortie et signal d'entrée)
 - Référence de tension à définir
 - Représentation en dB
 - ...
 - Commande LED RGB
 - Pilotage leds RGB
 - Variation de couleur en fonction du rapport d'amplitude
 - < 40dB => vert
 - < 80 dB => jaune/orange
 - < 100 dB => rouge
 - > 100 dB => rouge avec clignotement
 - ...
 - Communication entre le uC et le chip FTDI
 - Protocole de communication (SPI, I2C, UART, ...)
 - Configuration trames
 - Longueur trame – début et fin trame – taux de transfert
 - Datas à transmettre ou recevoir
 - ...
 - Communication entre le uC et DAC externe (si pas implémenter dans le uC)
 - Protocole de communication (SPI, I2C, UART, ...)
 - Configuration trames
 - Longueur trame – début et fin trame – taux de transfert
 - Résolution & Echantillonnage signal de sortie
 - Datas à transmettre
 - ...
 - Sauvegarder Datas dans le uC
 - Utilisation de la Flash du uC pour sauvegarder des données
 - Constante(s) pour filtre
 - Choix affichage
 - ...
 - ...
 - **Partie Optionnelle** – Commande des matrices de LEDs (projet 2126)
 - Protocole de communication
 - Gestion des leds
 - Représentation gammes de fréquence & amplitude
 - ...
 - ...
- Recherche et implémentation de solutions au niveau **Software**
 - Application C#
 - Communication série entre PC et le système
 - Longueur trame – début et fin de trame – taux de transfert
 - Datas à transmettre
 - Constantes pour filtre
 - Choix affichage
 - ...

- Démontrer par différentes simulations et mesures que les parties Hardware, Firmware et Software ont été bien implémentées.

2 A l'issue du projet de diplôme, l'étudiant fournira (liste non exhaustive) :

- Fichiers sources de CAO électronique du PCB réalisé (ALTIUM) + configuration logiciel (version utilisées)
- Tout le nécessaire pour fabriquer un exemplaire hardware :
fichiers de fabrication (GERBER) / liste de pièces avec références pour commande (BOM) / implantation (prototype) / modifications, etc
- Fichiers sources de programmation microcontrôleur (.c / .h) + configuration logiciel (version utilisées)
- Fichiers sources d'application externe au système électronique + configuration logiciel (version utilisées)
- Tout le nécessaire pour programmer le microcontrôleur (logiciel ou fichiers .hex), sous format numérique => utilisation de la structure de projet fourni par l'ES
- Tout le nécessaire pour programmer des applications (logiciel ou fichiers .hex) sous format numérique => utilisation de la structure de projet fourni par l'ES
- Tout le nécessaire à l'installation de programmes sur PC ou autres environnements utilisés durant le travail de diplômes
- Un rapport de diplôme contenant :
 - Les concepts du design **Hardware** :
 - Études des différents systèmes à implémenter (explications)
 - Choix des composants et dimensionnement de ceux-ci (calculs / simulation / autre)
 - Réalisation schématique / PCB / boîtier / montage de la carte
 - Les concepts du design **Firmware**
 - Structogramme / flowchart / Pseudocode
 - Explication des algorithmes mise en place
 - Démonstration par calculs / outils de debug / des résultats obtenus
 - Validation des concepts mis en place
 - Les concepts du design **Software**
 - Structogramme / flowchart / Pseudocode
 - Explication des algorithmes mise en place
 - Démonstration par calculs / outils de debug / des résultats obtenus
 - Validation des concepts mis en place
 - Tests & Validation :
 - Méthodologie de tests
 - Mesure(s)
 - Validation des résultats
 - Correction(s) apportée(s) au design (Hardware / Firmware / Software)
 - Estimation des coûts pour le design développé (un prototype)
 - Etat d'avancement & problème(s) rencontré(s)
 - Conclusion
 - Bibliographie / webographie / autre sources
- Les annexes :
 - Calculs détaillés des concepts
 - Listings complets des parties Firmware & Hardware que vous avez implémenté
 - Schématique + plan d'implémentation complète du PCB
 - Dessin / schématique du boîtier
 - Mesures
 - Pages utilisées des différents datasheets ou documentations exploités

- Mode d'emploi du système développé pendant le diplôme
- Journal de travail
- PV de séances hebdomadaires
- Un prototype

3 Autres demandes / contraintes / conseils

- **Planifier** dans le détail les travaux demandés.
- Se référer au planning régulièrement, **vérifier son avancement**, rédiger son **journal de projet** quotidiennement.
- Commencer à **rédiger le rapport de diplôme le plus tôt possible**, et régulièrement tout au long du travail de diplôme.
- Prendre du temps, préparer sa réflexion, rechercher des apports théoriques et des exemples pratiques, **envisager plusieurs possibilités** avant de finaliser une solution.
- **Numéroter et dater tous les documents**
- En cas de **problème** (retard, objectif à revoir, difficulté rencontrée, etc.), se référer à l'enseignant et au mandataire au plus vite.
- Toutes les **décisions importantes**, tant au niveau technique qu'organisationnel, doivent être posées **par écrit** dans le PV de séance, le rapport de diplôme et /ou figurer dans le journal de projet, après discussion avec l'enseignant / le mandataire.

4 Documents de références

Projet ES :

- **Prj 2126 – Affichage Matriciel**
K:\ES\PROJETS\SLO\2126_AffichageMatriciel
- **Prj 2107 – Filtre Numerique pour PIC32**
K:\ES\PROJETS\SLO\2107_FiltreNumeriquePIC32

Vidéo

- Transformée de fourrier – algorithme
<https://www.youtube.com/watch?v=htCj9exbGo0>

Pages web :

- **Articles**

FFT :

<https://www.f-legrand.fr/scidoc/docmml/numerique/tfd/fft3/fft3.html#:~:text=L'algorithme%20de%20transfor m%C3%A9e%20de,%2C%20soit%20N%3D2p.>

<https://nicolas.thiery.name/DESS/Notes/Cours4.pdf>

Filtrage numérique :

https://public.iutenligne.net/electronique/le-bars/num/fnum_ana.pdf

<https://www.jipehem.fr/cours/m-filnum.pdf>

<https://cpge.frama.io/fiches-cpge/Python/Filtres/80%20-%20Filtrage/>

➤ **Documentation FTDI**

Liste composants :

<https://ftdichip.com/product-category/products/ic/>

➤ **Documentation Microchip**

Liste famille uC :

<https://ww1.microchip.com/downloads/aemDocuments/documents/MCU32/ProductDocuments/Brochures/32-bit-Brochure-30009904.pdf>