

Sommaire

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Séance 1 : Jupyter Notebook**
   1. Etude préliminaire
   2. Travail en Séance
   3. Conclusion
2. **Séance 2 : Mesure de signaux périodiques à l’oscilloscope numérique**
   1. Etude préliminaire
   2. Travail Encadré

***Séance 1 : Jupyter Notebook :***

1. Travail Préliminaire :

Nous devions réaliser des scripts permettant de :

* Lire un fichier audio .flac
* D’afficher ses principales caractéristiques (Fe, Nombre de Bits, durée totale, nombre de piste)
* D’afficher la représentation temporelle du signal ainsi que son spectre d’amplitude

Une image contenant texte

Description générée automatiquementCette fonction nécessite plusieurs librairies, soundfile, \_soundfile, IPythonDisplay, os et le fichier python Util\_TP206.

Cette fonction nous permet de réaliser toutes les exigences de la phase préliminaire,

Nous lisons le fichier et stockons les valeurs dans un tableau et la Fréquence d’échantillonnage dans une variable. On détermine le nombre de piste en regardant la 2e colonne du tableau x.

Puis il nous reste plus qu’à afficher les informations relevé par la fonction info ou nos calcul

Pour l’affiche, nous avons utilisé une boucle prenant en paramètre le nombre de piste pour pouvoir afficher à la suite les figures correspondantes à ce nombre ici avec 2 pistes nous avons un affiche de la forme :



le script de la phase préliminaire étant fini nous pouvons nous attaquer la suite

1. Travail encadré

Nous devions maintenant réaliser des scripts pouvant :

* Extraire X secondes d’un fichier audio
* Echantillonner cet extrait à une fréquence inférieure

Quantifier les échantillons à un nombre inférieur de bits

Extraction :

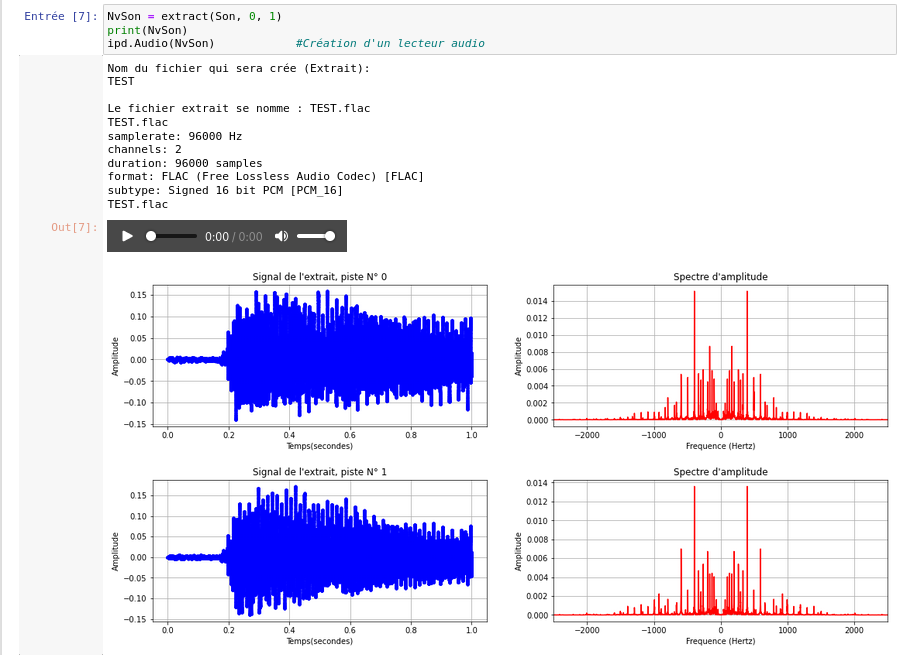
Une image contenant texte

Description générée automatiquement

Nous avons appelé ce premier script « extract » il prends en paramètre, le fichier audio, le début de l’extrait et sa fin.

Nous devons donc lire le fichier puis extraire le nombre d’échantillon à utiliser pour obtenir l’extrait de durée X.

Après avoir fait cela nous pouvons afficher et lire le nouveau fichier crée



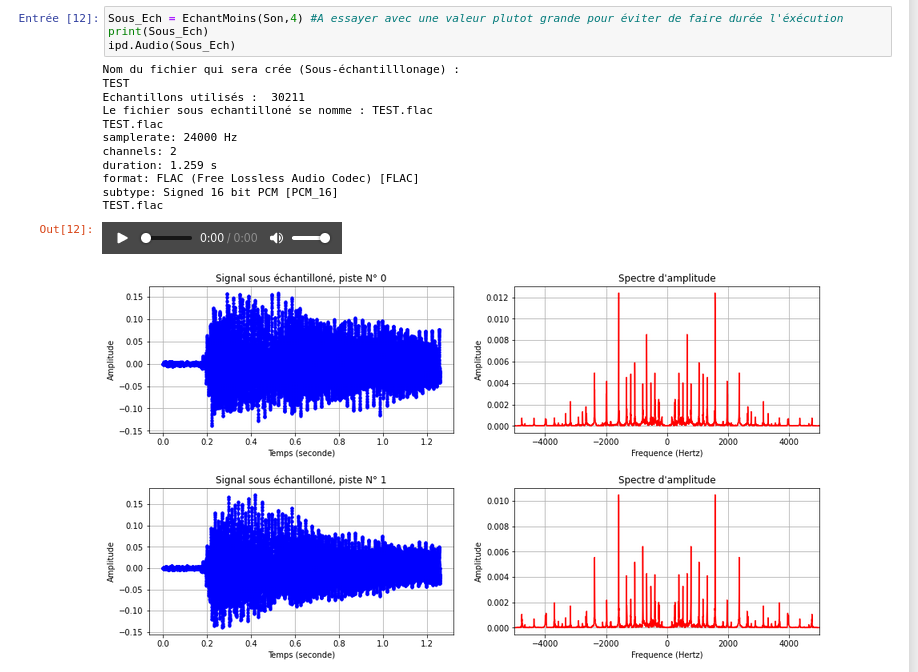
Echantillonnage :

Cette fonction nous permet de réduire le nombre d’échantillon utilisé dans la vidéo. Elle prend en paramètre le son et le nombre de fois que l’on souhaite diviser le nombre d’échantillon.

Une image contenant texte

Description générée automatiquement

Et elle produit l’affichage suivant



Quantification :

Dans celle-ci, nous devons réduire le nombre d’échantillon pour cela nous devons donc diminuer le nombre de bits/échantillon. Pour cela nous allons quantifier avec la loi uniforme et non uniforme (crée 2 fichiers que l’on peut ensuite comparé).

Une image contenant texte

Description générée automatiquement

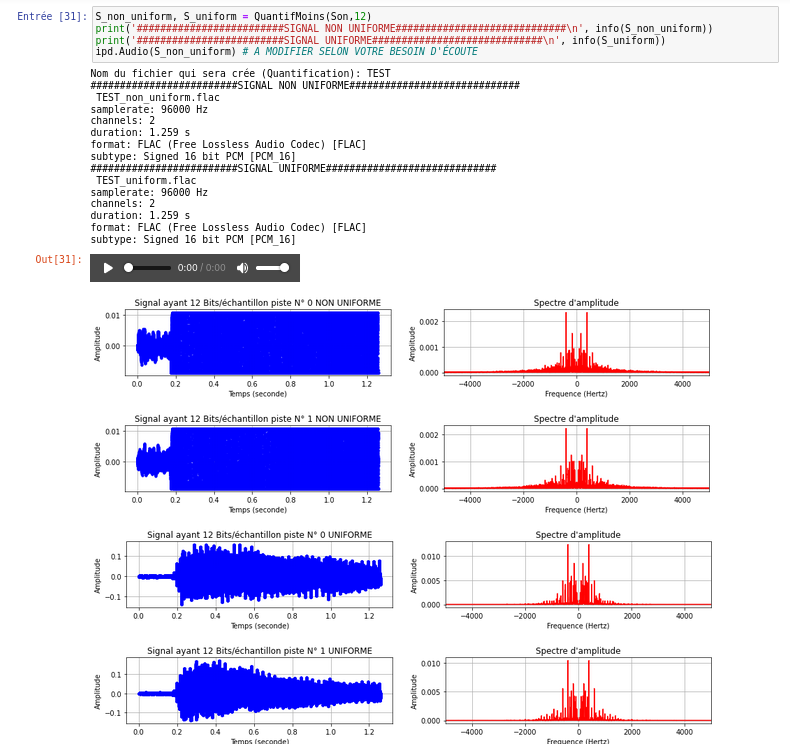
Une image contenant texte

Description générée automatiquement

Une image contenant texte

Description générée automatiquement

Cela produit l’affichage suivant :



Maintenant que tout les scripts pythons sont fonctionnel nous devons effectuer extraire d’un fichier HD un fichier SD.

Pour cela nous devons donc utiliser les scripts précédents :

Une image contenant texte

Description générée automatiquement

L’enchainement de toutes les fonctions mise en place nous permettent donc de réaliser la mission finale.

Nous avons donc en sortie, un fichier audio SD provenant d’un fichier HD.

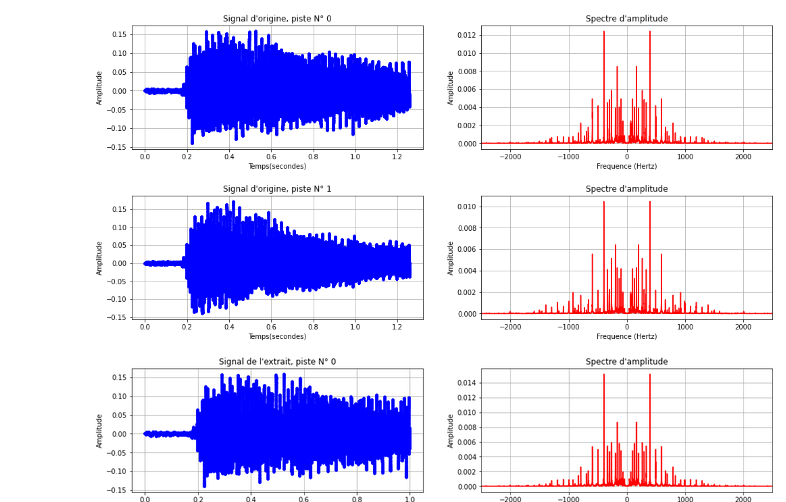
Affichage :

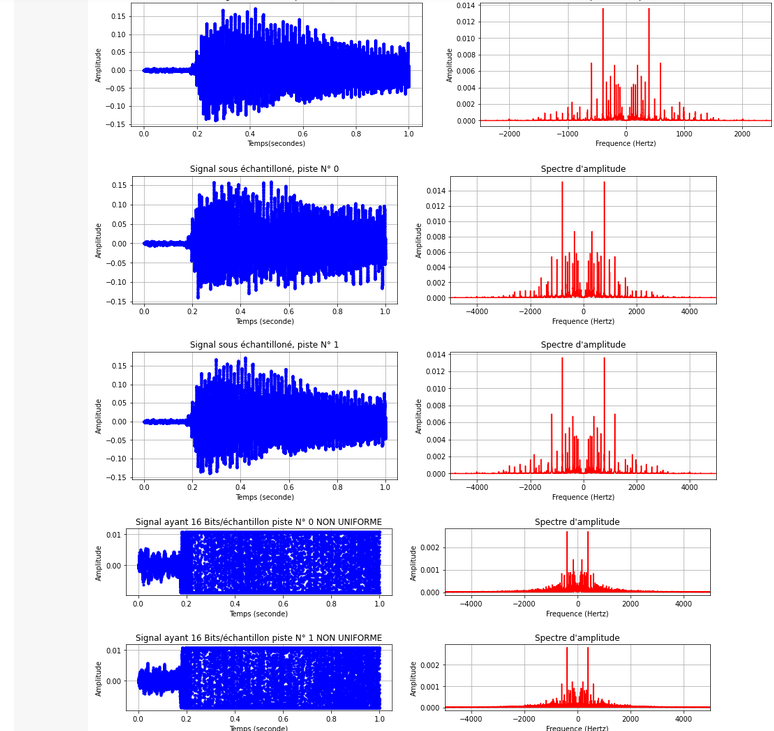
Une image contenant texte

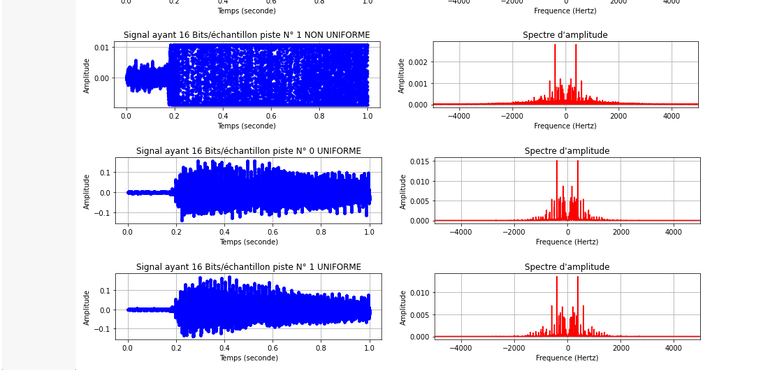
Description générée automatiquement

Une image contenant texte

Description générée automatiquement







* 1. Conclusion :

Grace à cette SAE, nous avons appris qu'un fichier audio est en HD lorsque sa quantification est de 24bits, un fichier audio SD quant à lui est quantifié à 16 bits ce qui démontre une différence de niveau élevé est donc de qualité en même temps dans la théorie.

Nous devions donc réaliser 3 principales fonctions nous permettant de comparer ces 2 qualités.

Ces 3 fonctions sont donc :

- L'extraction de donnée (Récupérer un échantillon entre 2 valeurs choisis A et B)

- Le sous-échantillonnage (Permet de recréer un échantillonnage de qualité SD car SD = 48000Hz et HD = 96000Hz)

- La sous-quantification (Diminue le nombre de niveau du son et par la même occasion sa précision)

Lorsque l'on quantifie, 2 options s'offre à nous :

- La quantification uniforme

- La quantification non-uniforme

Ces 2 solutions ont chacune leurs avantage mais dans notre cas, selon les spectres que l'on peut apercevoir lorsque l'on fait appel à ces quantifieurs, la quantification uniforme maintient un niveau de qualité plus intéressant que la quantification non-uniforme. On peut donc s'apercevoir que lorsque les valeurs en amplitude sont assez proches il faut prioriser une quantification uniforme et une non-uniforme si le signal possède des amplitudes très éloigné les unes des autres.

On peut également apercevoir que l'extension ".flac" ne permet pas d'identifier le nombre exact de bits utilisé dans un fichier audio. Il n'affiche que 16bits si nous sommes en dessous de cette valeur.

Pour finir, on ne peut pas facilement différencier à l'oreille un fichier de qualité SD et HD et lorsque l'on effectue cette manipulation, la qualité du son diminue quand même comme le démontre les spectres temporels et d'amplitudes.

**Séance 2 : Mesure de signaux périodiques à l’oscilloscope numérique**

1. Travail Préliminaire

1 – Recherches sur l’oscilloscope numérique

Contrairement aux modèles analogiques, le signal à visualiser est préalablement numérisé par un convertisseur analogique-numérique. La capacité de l'appareil à afficher un signal de fréquence élevée sans distorsion dépend de la qualité de cette interface.

Les oscilloscopes numériques possèdent tous à l’heure actuelle un module de calcul de FFT pour effectuer l’analyse fréquentielle des signaux. Contrairement aux véritables analyseurs FFT, ces oscilloscopes n’ont pas de filtre anti-repliement

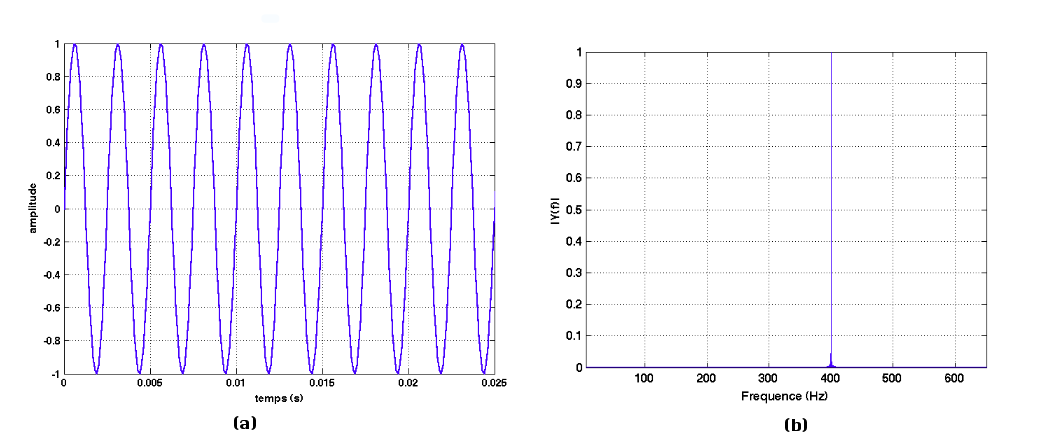
L'appareil est couplé à des mémoires permettant de stocker ces signaux et à un certain nombre d'organes d'analyse et de traitement qui permettent d'obtenir de nombreuses caractéristiques du signal observé :

* Mesure des caractéristiques du signal : valeur de crête, valeur efficace, période, fréquence
* Transformation rapide de Fourier qui permet d'obtenir le spectre du signal
* Filtres perfectionnés qui, appliqués à ce signal numérique, permettent d'accroître la visibilité de détails
* Décodage de signaux numériques : LIN, CAN, USB

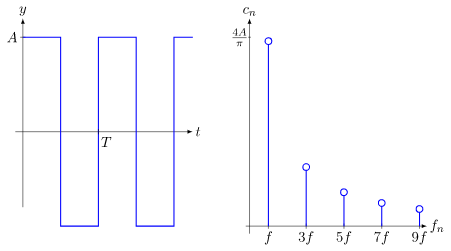
L'affichage du résultat s'effectue de plus en plus souvent sur un écran à cristaux liquides, ce qui rend ces appareils faciles à déplacer et beaucoup moins gourmands en énergie.

2 – Rappels des spectres théorique

Signal Sinusoïdal :



Signal Carré :



Signal Triangulaire :

Une image contenant texte

Description générée automatiquement

3- Documentation sur le phénomène d’aliasing

Le repliement de spectre (aliasing en anglais) est un phénomène qui introduit, dans un signal, des fréquences qui ne devraient pas s'y trouver, lorsque la fréquence porteuse ou la fréquence d'échantillonnage sont inférieures à deux fois la fréquence maximale contenue dans le signal.

Pour éviter cela, il suffit de respecter le théorème de Nyquist-Shannon disant que Fe > 2Fmax. 