

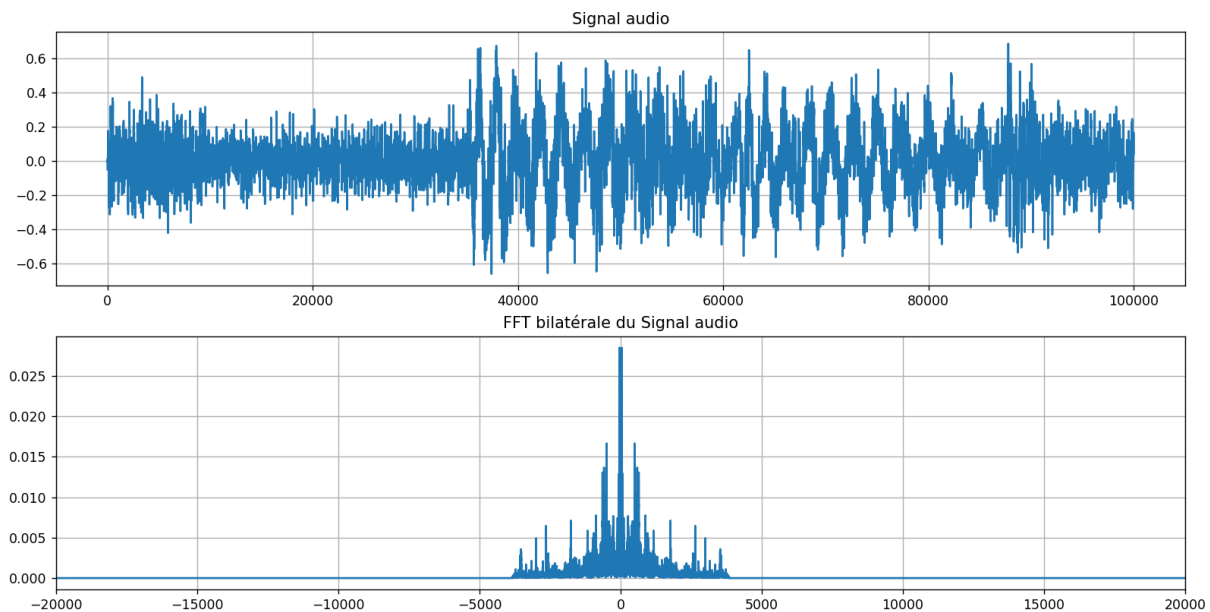
SAE 301 – Partie 1 : modulation et démodulation AM d'un signal audio

Caractéristique du signal audio

Pour la SAE, nous vous fournissons le signal audio nommé **NR4.wav** (New Radical : you get what you give) qui sera utilisé pour effectuer la modulation/démodulation AM synchrone du signal.

On vous donne le fichier **Emission_AM_Etu.py** écrit en **Python** qui

- charge le signal audio avec la librairie soundfile, en utilisant **soundfile.read()**
- cette fonction retourne la fréquence d'échantillonnage qui est de **$f_e = 100\text{kHz}$** , et le tableau de points du signal : **tab_signal** qui comporte **4000000 échantillons**, soit **40 secondes** de signal.
- visualise une tranche **[1000000 ; 1100000]** de ce signal dans le domaine temporel et sa FFT en bande de base. On peut remarquer que le signal occupe une **bande spectrale de 4 kHz**.



Remarques :

- Pour plus d'informations n'hésitez pas à analyser les librairies Python des fonctions utilisées.
- On ne traitera qu'une tranche du signal
- Toutes les FFT seront représentées en bilatérale

Travail à réaliser :

Vous complétez le programme pour qu'il puisse réaliser les 3 étapes décrites ci-dessous :

Etape 1 : Modulation AM :

- Compléter le programme pour qu'il puisse calculer les échantillons du **signal modulé AM**. On prendra comme fréquence porteuse un signal sinusoïdal de fréquence **$f_p = 10\text{ kHz}$** , et une amplitude **$A_p = 1\text{ Volt}$** .
- Représenter sur la **figure 1 (Fig.1)** le signal de la porteuse et le signal modulé AM.
- Faites le calcul de la Densité Spectrale de Puissance (DSP) de ce signal modulé en prenant comme axe des ordonnées, la **puissance en dBm**. On prendra une résistance de **$R = 50\ \Omega$** .

- Afficher la [figure 2 \(Fig.2\)](#) de la **DSP bilatérale du signal audio** et la **DSP bilatérale du signal modulé AM**.

Etape 2 : Démodulation synchrone :

- On se rappelle que pour la démodulation synchrone, il faut multiplier le signal modulé AM par la porteuse. Réalisez le programme qui fait ce calcul.
- Représenter sur la [figure 3 \(Fig.3\)](#), le signal modulé AM et ce même signal multiplié par la porteuse, signal que l'on nommera **signal_dem1**.
- Faites le calcul de la DSP du **signal_dem1** en prenant comme axe des ordonnées, la puissance en dBm.
- Afficher sur la [figure 4 \(Fig.4\)](#) la DSP bilatérale du signal audio et la DSP bilatérale du signal_dem1.
- Faire un **filtrage passe-bas** : on pourra prendre un **filtre de butterworth**, on choisira correctement l'**ordre** et la **fréquence de coupure**. Soit **signal_dem2** le signal en sortie du filtrage passe-bas.
- Afficher sur la [figure 5 \(Fig.5\)](#) la DSP du **signal_dem2** et expliquer le résultat.
- Afficher sur la [figure 6 \(Fig.6\)](#) le signal audio et le signal démodulé (signal_dem2) sur une même figure. Comparer ces 2 signaux et estimer le retard de transmission dû au calcul.

Etape 3 : Effet d'une erreur de synchronisation de 0.1 Hz en réception :

En réception (pour la démodulation) on a un **défaut de synchronisation de 0.1 hz**, donc la fréquence porteuse vaut maintenant **fp1 = 10,0001 kHz**.

- Refaites les simulations précédentes et afficher sur la [figure 7 \(Fig.7\)](#) le signal démodulé sans erreur avec le signal démodulé ayant subi une erreur de **0.1 Hz**.
- Conclure

Déposer votre fichier Python sur le dépôt Moodle prévu à cet effet.