

Université des Sciences et de la Technologie
Houari Boumédiène

Faculté de Génie Electrique - Département de Télécommunication

Licence de Télécommunications

Matière

Communications Numériques

TP #1

Initiation à GNU Radio Companion

Année universitaire 2024/2025

1. Objectif du TP

L'objectif de ce premier TP est de se familiariser avec le logiciel GNU Radio Companion (GRC) pour schématiser une chaîne de communication numérique depuis la source jusqu'au destinataire. Pour cela, ce TP vise l'exploration de quelques outils de traitement du signal notamment la visualisation du signal dans les domaines temporel et fréquentiel ainsi que le filtrage des fréquences.

2. Fonctionnalités de GNU Radio Companion (GRC)

GRC est un outil graphique de conception de flux (flowgraph) de traitement de signal englobant les fonctionnalités suivantes (Fig. 1.1) :

- Interface graphique : GRC permet de créer des flux de traitement de signal en connectant des blocs fonctionnels (appelés "blocks") via une interface glisser-déposer.
- Blocs prédéfinis : Il inclut une bibliothèque étendue de blocs pour des opérations courantes comme la modulation, le filtrage, la FFT, la démodulation, etc.
- Personnalisation : Les utilisateurs peuvent créer leurs propres blocs en Python ou en C++.
- Intégration matérielle : GRC fonctionne avec des périphériques électroniques externes.
- Simulation et test : Il permet de simuler des systèmes de communication complets sans avoir besoin de matériel physique.

Le téléchargement de GRC peut se faire à partir du lien :

<https://github.com/ryanvolz/radioconda>

Plusieurs versions sont disponibles pour Windows, Linux ou macOS.

Dans le cadre de cette matière, les principales utilisations sont les suivantes :

- Analyse d'une chaîne de communication numérique dans les domaines temporel et fréquentiel.
- Etude de la qualité de la transmission numérique d'une séquence binaire en l'absence et en présence du bruit.
- Etude des techniques de modulation pour la transmission numérique d'une séquence binaire.

Dans l'interface graphique, plusieurs boutons sont disponibles pour exécuter les commandes notamment (Fig.1.2) :

- Groupe 1 : Nouveau, Ouverture et Sauvegarde d'un schéma.
- Groupe 2 : Copier, Coller, Supprimer, Annuler.
- Groupe 3 : Exécuter et Arrêter la simulation.

- Groupe 4 : Connecter ou Déconnecter les liens entre les blocs.
- Groupe 5 : Recherche de blocs.

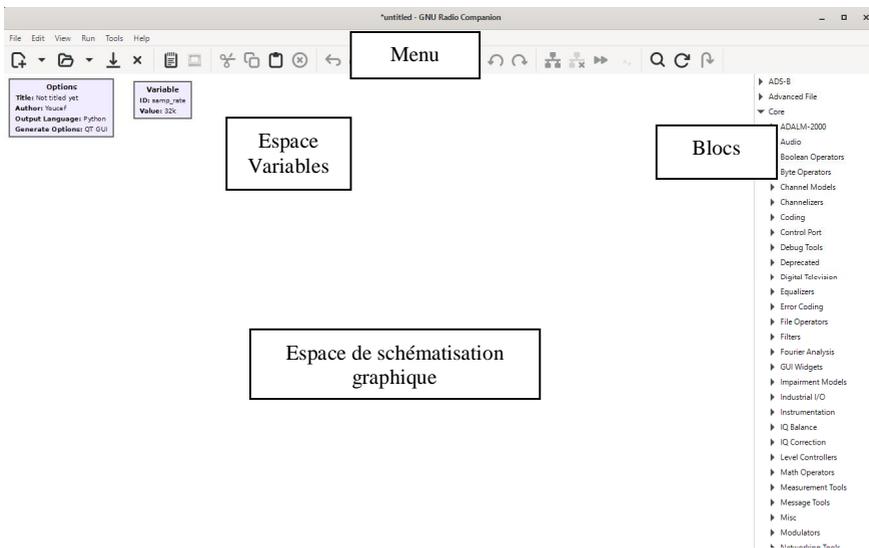


Fig.1.1. Interface de GNU Radio Companion

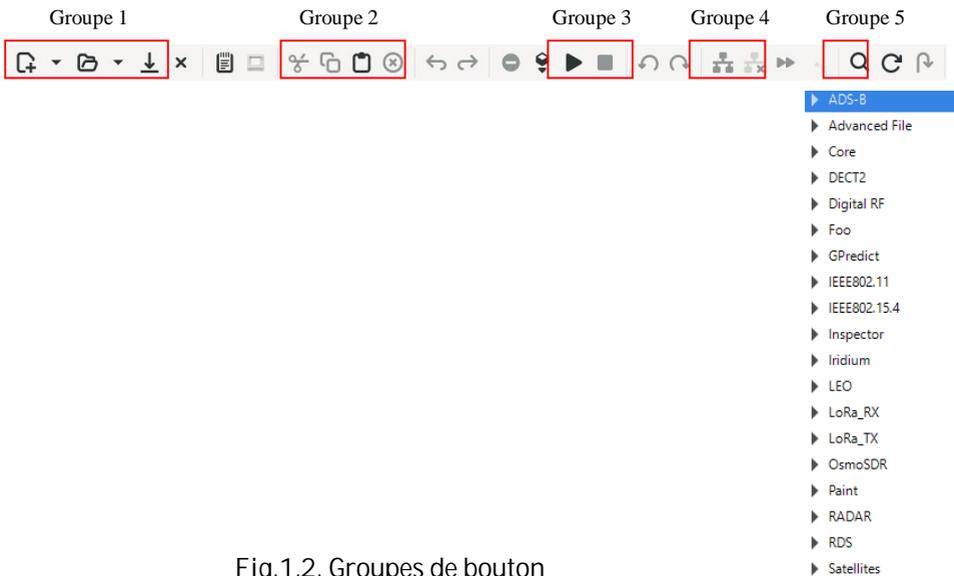


Fig.1.2. Groupes de bouton

Sous-catégorie	Blocs
Waveforms generators : Pour la génération des signaux	<ul style="list-style-type: none"> ▼ Waveform Generators <ul style="list-style-type: none"> Constant Source Fast Noise Source GLFSR Source Noise Source Random Source Random Uniform Source Signal Source
Instrumentation : Pour la visualisation des signaux (Temporel, fréquentiel, etc)	<ul style="list-style-type: none"> ▼ QT <ul style="list-style-type: none"> fosphor sink (Qt) QT GUI Bercurve Sink QT GUI Constellation Sink QT GUI Eye Sink QT GUI Frequency Sink QT GUI Histogram Sink QT GUI Number Sink QT GUI Sink QT GUI Time Raster Sink QT GUI Time Sink QT GUI Vector Sink QT GUI Waterfall Sink
Filtrage : Le filtrage des signaux (Passe-bas, passe-haut, FIR, etc)	<ul style="list-style-type: none"> ▼ Filters <ul style="list-style-type: none"> Band-pass Filter Taps Band-reject Filter Taps Band Pass Filter Band Reject Filter DC Blocker Decimating FIR Filter FFT Filter FFT Low Pass Filter FFT Root Raised Cosine Filter Filter Delay Filter Taps Loader Generic Filterbank High-pass Filter Taps High Pass Filter Hilbert IIR Filter Interpolating FIR Filter Low-pass Filter Taps Low Pass Filter Root Raised Cosine Filter RRC Filter Taps

Tableau 1.1. Liste des Blocs de la catégorie Core.

Le groupe 5 regroupe les différents blocs prédéfinis pour réaliser un schéma de traitement du signal. Les principaux blocs utilisables dans cette matière sont regroupés dans la catégorie Core. En cliquant sur Core, on peut découvrir les sous-catégories puis les blocs nécessaires à la réalisation des schémas. Les principales sous-catégories utilisables dans cette matière sont reportées dans le tableau 1.1.

2. Expérience 1

Dans cette première expérience, on se propose de générer un signal harmonique additionné à un bruit gaussien pour le visualiser dans les domaines temporel et fréquentiel. Le signal bruité est décrit par l'équation suivante :

$$y(t) = x(t) + b(t)$$

où $x(t) = A \cos(2\pi f_c t)$. A et f_c représentent respectivement l'amplitude et la fréquence du signal. La représentation du signal dans le domaine numérique nécessite son échantillonnage en respectant le théorème de Shannon :

$$F_e \geq 2f_{max}$$

Dans le cas de cette expérience, $f_{max} = f_c$.

Le nombre d'échantillons que l'on peut générer est lié à la durée d'observation du signal notée T_d . Ce nombre d'échantillons nommé N est calculé en utilisant l'équation suivante :

$$N = T_d F_e$$

Pour cette expérience, cinq (5) blocs sont nécessaires pour générer le signal bruité $y(t)$ comme décrit dans le tableau 1.1.

Bloc	Fonctionnalité
Signal Source	Générateur signaux : cosinus, sinus, porte et triangulaire.
Noise Source	Générateur de bruit : Gaussien, uniforme, ...
Add	Additionneur de signaux
QT GUI Time Sink	Visualiseur de signaux dans le domaine temporel.
QT GUI Frequency Sink	Visualiseur de signaux dans le domaine fréquentiel.
Variable	Fixateur de valeurs utilisables dans la simulation.

Tableau 1.2. Blocs de simulation et leurs fonctionnalités

La simulation d'un schéma de traitement du signal nécessite quatre (4) étapes :

- Déclaration des variables.
- Réalisation du schéma de simulation.
- Réglage des paramètres de simulation.
- Exécution de la simulation.

2.1. Déclaration des variables

Cette étape permet de fixer les variables à utiliser pour exécuter la simulation. Au lancement de GRC, la fréquence d'échantillonnage F_e nommée `samp_rate` est fixée comme une variable de valeur par défaut 32000 Hz. Cependant, son nom et sa valeur peuvent être changés selon les besoins de l'expérimentation. Dans le cas de cette expérience, F_e est nommée `sample_rate` (Taux d'échantillonnage).

Trois autres variables sont déclarées pour la simulation de cette première expérience :

- `f_c` : fréquence f_c du signal cosinus
- `n_period` : nombre de périodes à visualiser lié par la relation $T_d = n_p \times T_c$ où $T_c = 1/f_c$ et n_p est le nombre de périodes du signal à visualiser.
- `numpoints` : nombre de points à visualiser défini par la relation $N = T_d F_e$.

2.2. Réalisation du schéma de simulation

Cette étape permet de réaliser le schéma de simulation en ramenant les blocs par simple glisser-déplacer. Pour faciliter la recherche des blocs, il suffit de taper son nom en cliquant sur le bouton recherche  pour le sélectionner et le déplacer dans l'espace de schématisation graphique. La liaison des blocs est intuitive. Il suffit de cliquer sur la sortie d'un bloc puis de cliquer sur l'entrée d'un autre bloc pour les connecter.

2.3. Réglage des paramètres de simulation

Le réglage des paramètres est une étape importante pour la bonne simulation d'un système de traitement du signal. Chaque bloc a ses propres paramètres de simulation. Le tableau suivant décrit les paramètres importants pour chaque bloc :

Bloc	Paramètres
Signal Source	<ul style="list-style-type: none"> • Sample Rate : Fréquence d'échantillonnage • Waveform : Forme du signal • Frequency : Fréquence du signal • Amplitude : Amplitude du signal
Noise Source	<ul style="list-style-type: none"> • Noise Type : Type de bruit • Amplitude : Amplitude du bruit
QT GUI Time Sink	<ul style="list-style-type: none"> • Number of Points : Nombre de points à visualiser • Sample Rate : Fréquence d'échantillonnage • Autoscale : Echelle de visualisation manuelle (No)

	ou automatique (Yes).
QT GUI Frequency Sink	<ul style="list-style-type: none"> • FFT Size : Nombre de points de calcul de la Transformée de Fourier Rapide • Center Frequency (Hz) : Fréquence centrale du signal • Bandwidth (Hz) : Bande passante du signal fixée à la fréquence d'échantillonnage.

Tableau 1.3. Paramètres de simulation des blocs.

2.4. Exécution de la simulation

L'exécution de la simulation se fait en cliquant sur le bouton Run ►. Une fenêtre s'affiche montrant la simulation.

→ **Manipulation 1** : Réaliser le schéma de la simulation présenté dans la Fig.1.3. Pour cette manipulation, les paramètres du signal bruité sont fixés comme suit : $F_e = 40000\text{Hz}$, $f_c = 500\text{Hz}$, Amplitude du signal $A = 2$, Amplitude du bruit = 0.1, $n_p = 1$.

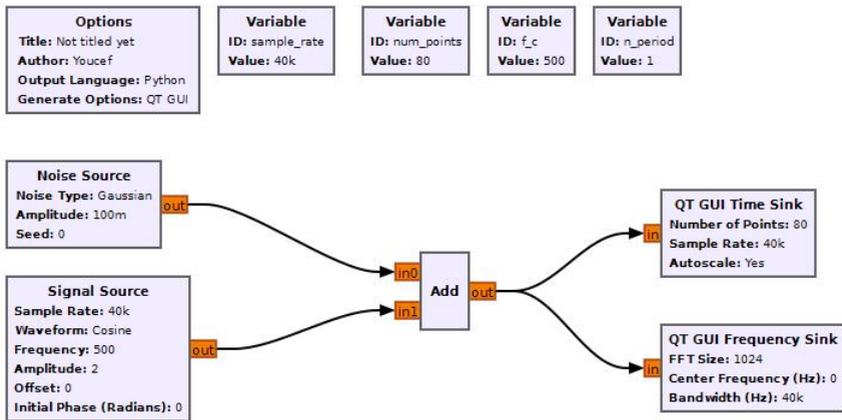


Fig.1.3. Schéma de simulation d'un signal harmonique bruité.

Qu'observe-t-on dans les domaines temporel et fréquentiel ?

→ **Manipulation 2** : Déterminer l'amplitude maximale du bruit pour avoir un rapport signal à bruit (SNR : Signal to Noise Ratio) supérieur à 20 dB. Dans le cas de cette manipulation, le SNR est défini par l'équation suivante :

$$SNR = 10 \log \left(\frac{P_S}{P_B} \right)$$

où P_S et P_B sont respectivement les puissances du signal et du bruit liées à leurs amplitudes par : $P_S = A_S^2$ et $P_B = A_B^2$.

Pour quelle valeur de l'amplitude du bruit peut-on considérer que le signal utile est noyé dans le bruit ?

3. Expérience 2

Dans cette deuxième expérience, on se propose de générer un signal harmonique avec une amplitude aléatoire pouvant prendre ± 2 . Cette amplitude aléatoire est similaire à une séquence binaire où le bit "0" est représenté par une amplitude -2 et le bit "1" par une amplitude $+2$. Le signal bruité prend alors la forme suivante :

$$y(t) = x(t) + b(t)$$

avec $x(t) = A \cos(2\pi f_c t)$ et A est une amplitude aléatoire de loi uniforme qui peut prendre ± 2 . La simulation d'un tel signal est montrée dans Fig. 1.4.

Dans cette simulation, on distingue trois parties : Déclarations des variables, la chaîne de simulation du signal bruité et la visualisation des signaux dans les domaines temporel et fréquentiel.

Pour générer une amplitude aléatoire d'amplitude ± 2 avec une loi uniforme, les blocs suivants sont utilisés :

- Random Uniform Source : Génère des valeurs entières entre ± 1 .
- Int to Float : Convertit les valeurs entières en valeurs réelles.
- Multiply Const : Multiplie les valeurs entières par 2 pour générer des amplitudes ± 2 .
- Repeat : Répète la valeur de l'amplitude du signal sur une période du signal.
- Multiply : Multiplie l'amplitude A par $\cos(2\pi f_c t)$.

Pour la visualisation du signal, deux blocs sont utilisés :

- Virtual Sink : permet d'affecter la sortie d'un bloc à une source virtuelle. Pour cela, il faut lui donner un nom pour pouvoir l'utiliser dans le bloc de visualisation.
- Virtual Source : permet de visualiser une source virtuelle définie dans le bloc Virtual Sink.

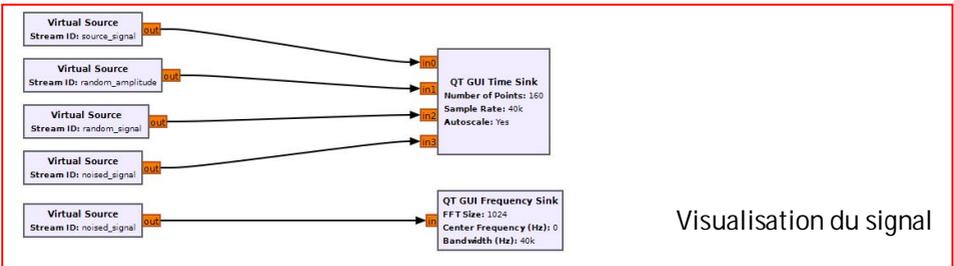
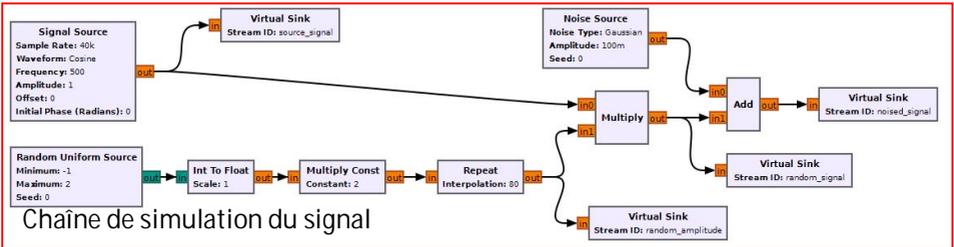
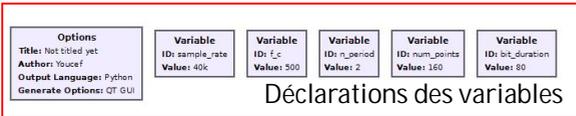


Fig.1.4. Schéma de simulation d'un signal harmonique d'amplitude aléatoire bruité.

Dans le schéma de simulation, quatre sources virtuelles sont utilisées pour la visualisation des signaux dans le domaine temporel. Par contre, dans le domaine fréquentiel, nous exploitons uniquement une source virtuelle.

→ **Manipulation 3** : Réaliser le schéma de simulation de Fig.1.4. Interpréter les visualisations temporelles et fréquentielles. Que peut-on dire ?

4. Expérience 3

Dans cette expérience, on se propose de générer la somme de trois signaux harmoniques additionnés avec un bruit gaussien. Cela se traduit par l'équation suivante :

$$y(t) = \sum_{k=1}^3 A_k \cos(2\pi f_k t) + b(t)$$

tels que $A_k = 2$ pour $k = 1, 2, 3$, $f_1 = 500\text{Hz}$, $f_2 = 1500\text{Hz}$ et $f_3 = 3500\text{Hz}$.

→ **Manipulation 4** : Réaliser le schéma de simulation de Fig.1.5. Interpréter les visualisations temporelles et fréquentielles. Que peut-on dire ?

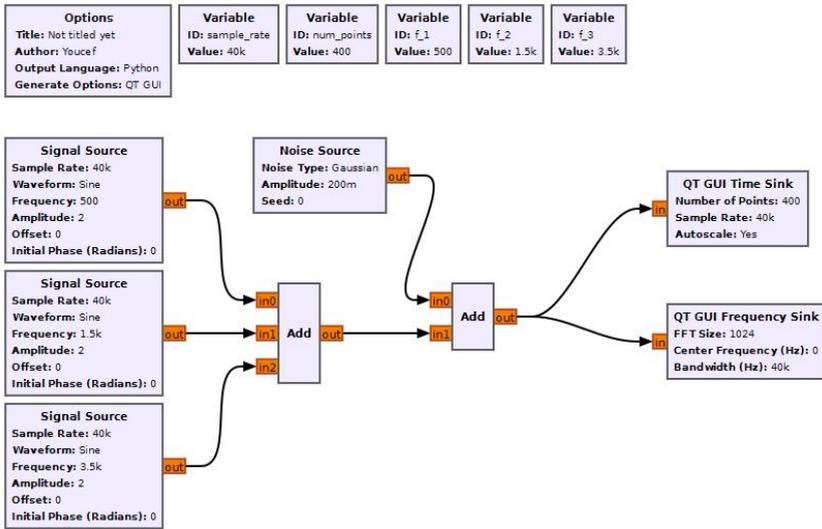


Fig.1.5. Schéma de simulation d'un signal avec trois harmoniques additionné à un bruit gaussien.

Pour une bonne visualisation du signal, sa durée est fixée à 10 ms. Déterminer le nombre nécessaire de points pour une bonne visualisation.

5. Expérience 4

Dans cette expérience, on se propose de filtrer le signal avec les trois harmoniques pour supprimer les fréquences f_2 et f_3 . Pour cela, on introduit un filtre passe-bas réglé à la fréquence de coupure 1000Hz. Le schéma de simulation est représenté dans Fig.1.6.

Le filtrage du signal est opéré par le bloc Low Pass Filter. Les paramètres de réglage pour supprimer les deux fréquences f_2 et f_3 sont :

- Sample Rate : Fréquence d'échantillonnage
- Cutoff Freq : Fréquence de coupure
- Transition Width : Largeur de la bande de transition
- Window : Fenêtre utilisée pour la conception du filtre numérique

→ **Manipulation 5** : Réaliser le schéma de simulation de Fig.1.6. Interpréter les visualisations temporelles et fréquentielles. Que peut-on dire ?

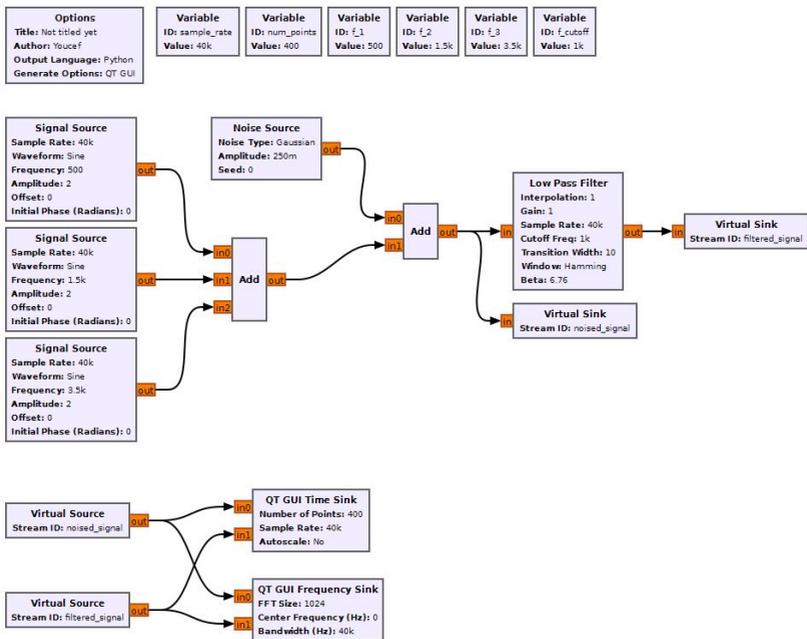


Fig.1.6. Schéma de simulation de filtrage d'un signal.

6. Expérience 5

Dans cette expérience, on se propose de lire un son réel pour l'écouter et l'afficher dans les domaines temporel et fréquentiel. Les signaux sont enregistrés numériquement dans l'un des formats wav ou mp3.

La lecture et l'écoute d'un son réel se font en utilisant les blocs : Wav File Source et Audio Sink.

Dans cette expérience, on utilise les signaux suivants :

- un chant d'oiseau d'une durée trois secondes échantillonné à une fréquence de 22050Hz.
- son "bip-bip" envoyé par la sonde spatiale sputnik en 1959 d'une durée de 4 secondes échantillonné à une fréquence 44100Hz.
- son d'une voix humaine d'une durée de deux secondes échantillonné à une fréquence de 22010Hz.

Les signaux seront fournis lors de la séance de TP.

→ **Manipulation 6** : Réaliser le schéma de simulation de Fig.1.7. Interpréter les visualisations temporelles et fréquentielles. Que peut-on dire ?

Le schéma de lecture et d'écoute du signal ainsi que les représentations temporel et fréquentiel est reporté dans Fig.1.7.

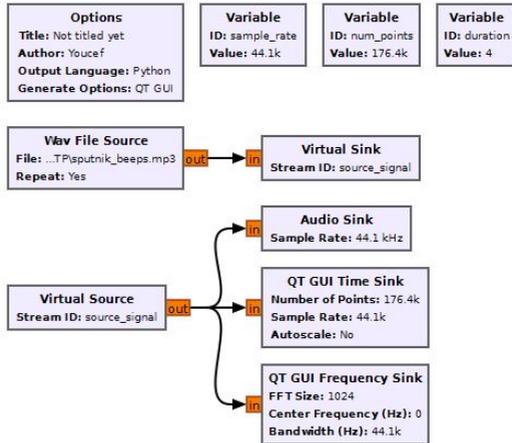


Fig.1.7. Schéma de lecture et d'écoute d'un signal audio et la visualisation dans les domaines temporel et fréquentiel.