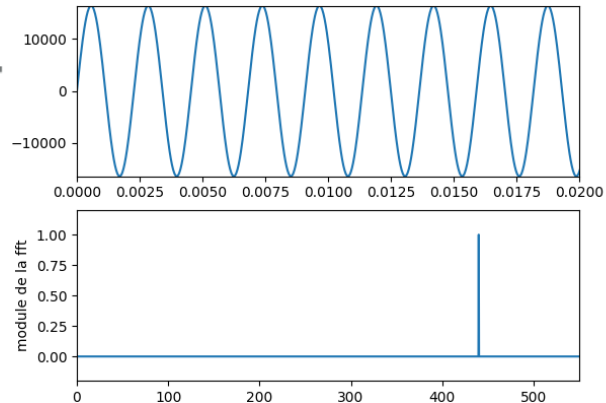


## Exemples d'utilisations du module son en ligne de commande sous Linux

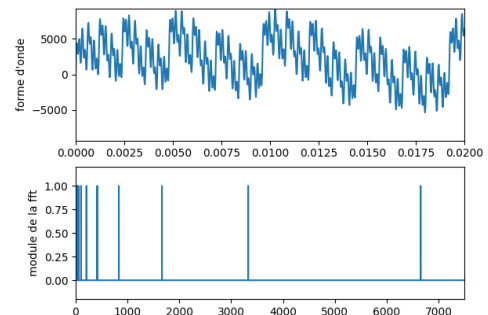
Pour obtenir un fichier sonore constitué d'une sinusoïde pure et afficher la représentation en forme d'onde et dans le domaine de Fourier :

```
popsi@AspireOneJean:~/cassiope$ ls
cassiope.wav Demo13_03 NormalEch.wav old __pycache__ shepard.py son.py ssEch.wav test.wav
popsi@AspireOneJean:~/cassiope$ python3
Python 3.5.3 (default, Sep 27 2018, 17:25:39)
[GCC 6.3.0 20170516] on linux
Type "help", "copyright", "credits" or "license"
>>> import son
>>> import shepard
>>> import matplotlib.pyplot as plt
>>> donnee = shepard.note(440)
>>> donnee.graphFFT(fmax=550)
>>> plt.show()
>>> donnee.ecrire("la")
creation du fichier en cours...
le fichier la.wav de duree 3.0s a ete cree
```



Pour obtenir le fichier sonore d'une note de Shepard de fréquence fondamentale au choix et afficher la représentation en forme d'onde et dans le domaine de Fourier :

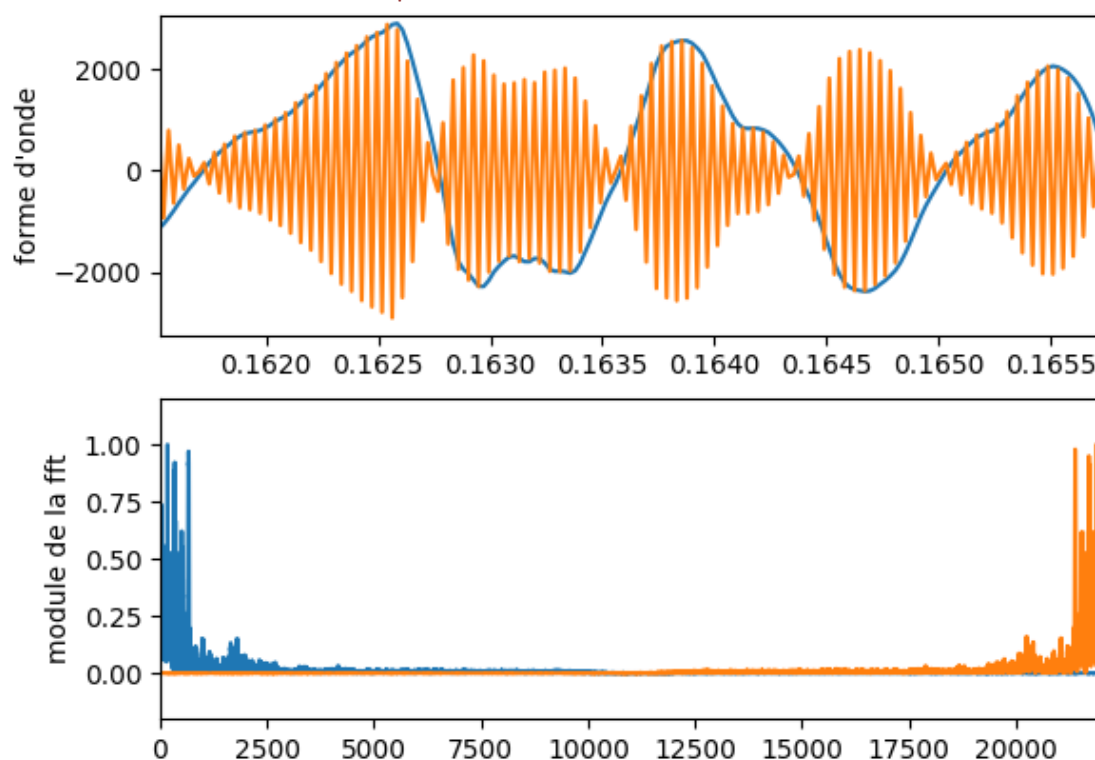
```
popsi@AspireOneJean:~/cassiope$ ls
cassiope.wav Demo13_03 la.wav NormalEch.wav old __pycache__ shepard.py son.py ssEch.wav test.wav
popsi@AspireOneJean:~/cassiope$ python3
Python 3.5.3 (default, Sep 27 2018, 17:25:39)
[GCC 6.3.0 20170516] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import son
>>> import shepard
>>> import matplotlib.pyplot as plt
>>> donnee = shepard.noteShepard(13)
>>> donnee.graphFFT(fmax = 7500)
>>> plt.show()
>>> donnee.ecrire("noteShepard")
creation du fichier en cours...
le fichier noteShepard.wav de duree 3.0s a ete cree
>>> exit()
popsi@AspireOneJean:~/cassiope$ ls
cassiope.wav Demo13_03 la.wav NormalEch.wav noteShepard.wav old __pycache__ shepard.py son.py ssEch.wav test.wav
```



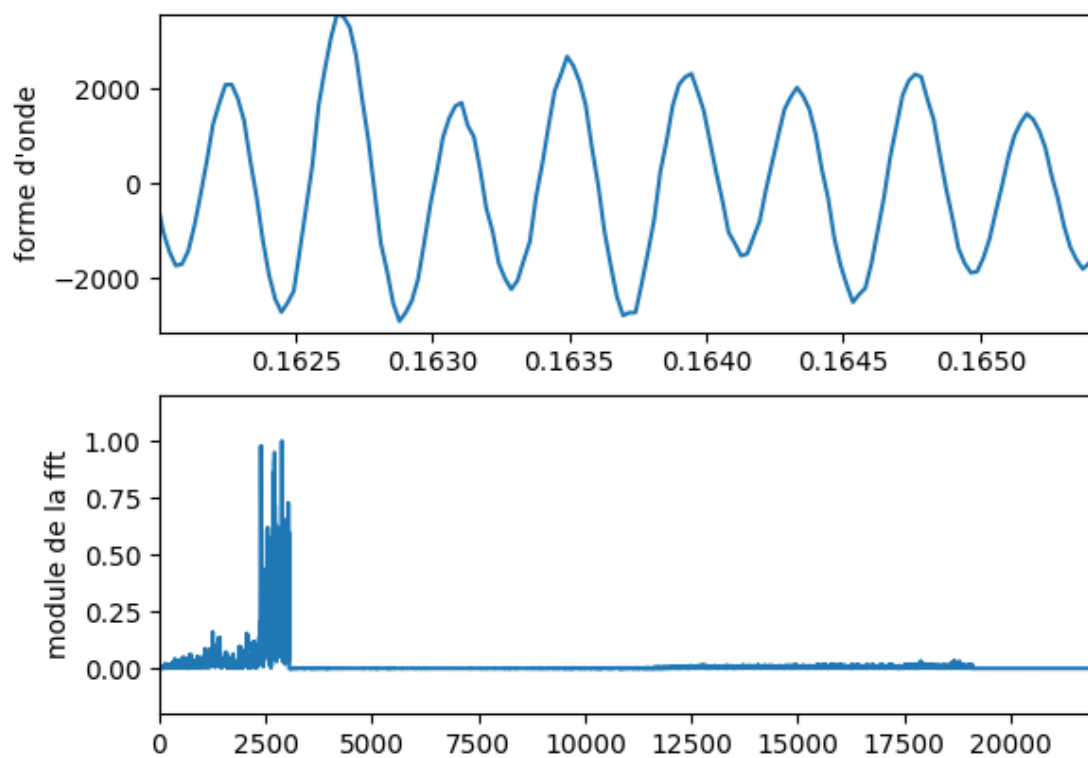
Pour inverser et décaler le spectre d'un fichier sonore (ici, de parole), visualiser le résultat de l'opération et exporter le fichier sonore en résultant :

```
popsi@AspireOneJean:~/cassiope$ ls
cassiope.wav Demo13_03 la.wav NormalEch.wav noteShepard.wav old __pycache__ shepard.py son.py ssEch.wav test.wav
popsi@AspireOneJean:~/cassiope$ python3
Python 3.5.3 (default, Sep 27 2018, 17:25:39)
[GCC 6.3.0 20170516] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import son
>>> import shepard
>>> import matplotlib.pyplot as plt
>>> donnee = son.ouvrir("cassiope")
>>> donnee.graphFFT()
>>> donnee.inverseFFT()
>>> donnee.graphFFT()
>>> plt.show()
>>> donnee.decalFFT(-19000,[19100,22050])
>>> donnee.graphFFT()
>>> plt.show()
>>> donnee.ecrire("cassiopeReverse")
creation du fichier en cours...
le fichier cassiopeReverse.wav de duree 1.0s a ete cree
>>> exit()
popsi@AspireOneJean:~/cassiope$ ls
cassiopeReverse.wav cassiope.wav Demo13_03 la.wav NormalEch.wav noteShepard.wav old __pycache__ shepard.py son.py ssEch.wav test.wav
```

Résultat visualisation inversion de spectre :



Résultat visualisation décalage fréquentiel après inversion de spectre :



Help on module son:

## NAME

son - Ce module permet de definir les objets donnees\_son.

## CLASSES

builtins.object  
DonneeSon

```
class DonneeSon(builtins.object)
```

objet DonneeSon

- data le tableau des valeurs des echantillons
- nbOctet = 1 ou 2 le nombre d'octets utilise par echantillon
- fech la frequence d'echantillonnage

Methods defined here:

`__init__(self, data, nbOctet, fech)`

Initialize self. See help(type(self)) for accurate signature.

`decalFFT(self, f, plage=[70, 350])`

Modifie l'objet de maniere a obtenir les echantillons correspondant à un decalage de la plage de frequence plage de f herz (f peut etre negatif).

`ecrire(self, nom)`

Cree, sous le nom nom, le fichier au format wave correspondant a l'objet.

`formGraphFFT(self, fmin=0, fmax=22100)`

Renvoie un tableau de 2 tableaux [f,Y].

- Le premier element du tableau est le tableau f representant la valeur des frequences en Hz.

Le second element est le tableau Y representant le module des valeurs des echantillons correspondantes dans le domaine de Fourier une fois normalise.

`formeGraphTemps(self)`

Renvoie un tableau de 2 tableaux [t,y].

- Le premier element du tableau est le tableau x representant la valeur du temps en s.

- Le second element est le tableau y representant les valeurs des echantillons correspondantes.

`graphFFT(self, tmin=0, tmax=0.02, fmin=0, fmax=0)`

Permet de visualiser le module de la transformee de fourier en regard de la forme d'onde.

Il reste a effectuer un matplotlib.pyplot.show() apres avoir ajoute toutes les courbes voulues.

- tmin et tmax en s

- fmin et fmax en Hz

`inverseFFT(self)`

Modifie l'objet de maniere a realiser l'operation d'inversion du spectre (on inverse le signe des echantillons une fois sur deux)

`ssEch(self, N=2)`

Si le nombre d'echantillons de l'objet est bien multiple de N, modifie l'objet de maniere a obtenir sa version sous-echantillonnee a fech/N.

-----  
Data descriptors defined here:

`__dict__`

dictionary for instance variables (if defined)

`__weakref__`

list of weak references to the object (if defined)

## FUNCTIONS

`ouvrir(nom)`

Si nom est une chaine de caractere correspondant a un fichier wave (sans l'extension), renvoie l'objet donnees son correspondant a ce fichier sonore.

Help on module shepard:

#### NAME

shepard - Ce module permet de générer des objets donnees\_son representant des sons de Shepard.

#### FUNCTIONS

gammeShepard(listFond=[16.3,..., 29.1, 30.8], nbOctet=2, fech=44100, dureeNote=0.35, dureeSilence=0.15, v

Renvoie l'objet donnees\_son correspondant a un enchainement de notes de Shepard. Permet de tester l'illusion auditive.

- duree en s positive
- dureeSilence en s positive
- volume float entre 0 et 1
- nbOctet un entier entre 1 et 2
- fech en Hz entier positif
- listFond un tableau contenant dans l'ordre chronologique la liste des valeurs en Herz des fondamentales des notes de chepard desirees|

note(fNote, nbOctet=2, fech=44100, duree=3, volume=0.5)

Renvoie l'objet donnees\_son correspondant a une sinusoide pure de frequence fNote avec les parametres de pas de quantification, de frequence d'echantillonnage, de duree et de volume entre.

- fNote float positif en Hz
- nbOctet un entier entre 1 et 2
- fech entiers positifs en Hz
- duree en s positive
- volume float entre 0 et 1

noteShepard(fFond, nbOctet=2, duree=3, volume=0.5)

Renvoie l'objet donnees\_son correspondant a une note de Shepard de fondamentale fFond en Hertz.

- fFond en Hz des entiers positifs
- duree en s positive
- volume float entre 0 et 1
- nbOctet un entier entre 1 et 2