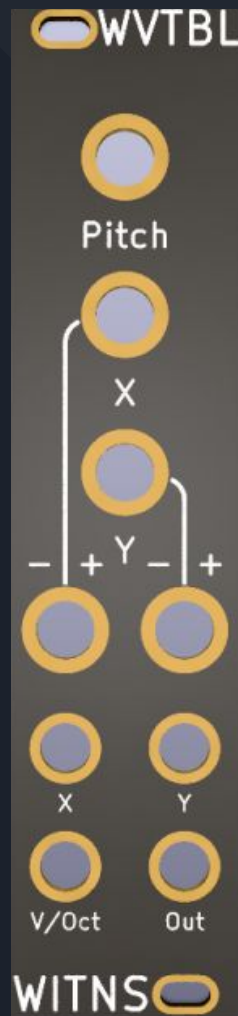


Synthétiseur table d'onde

Vincent EVRARD - Maxime PICHERY - Baptiste RODIER

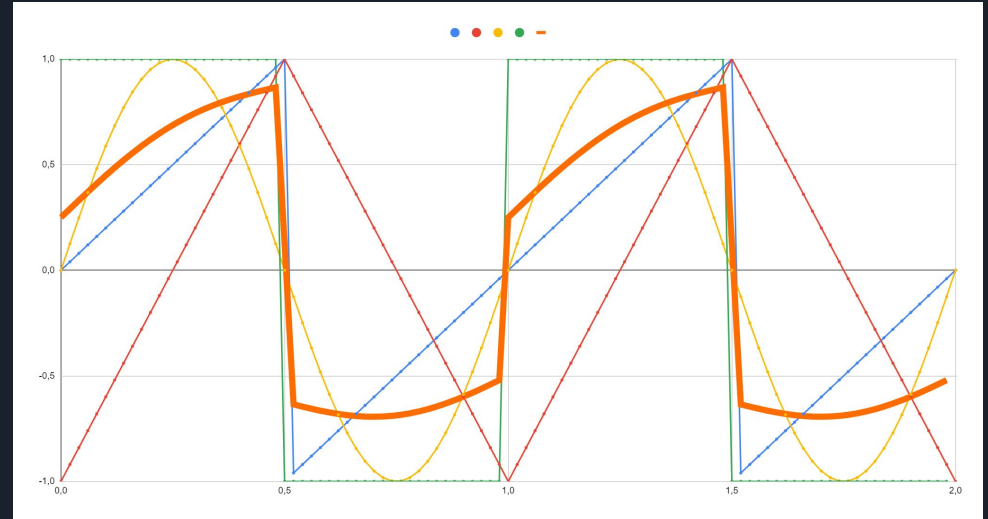




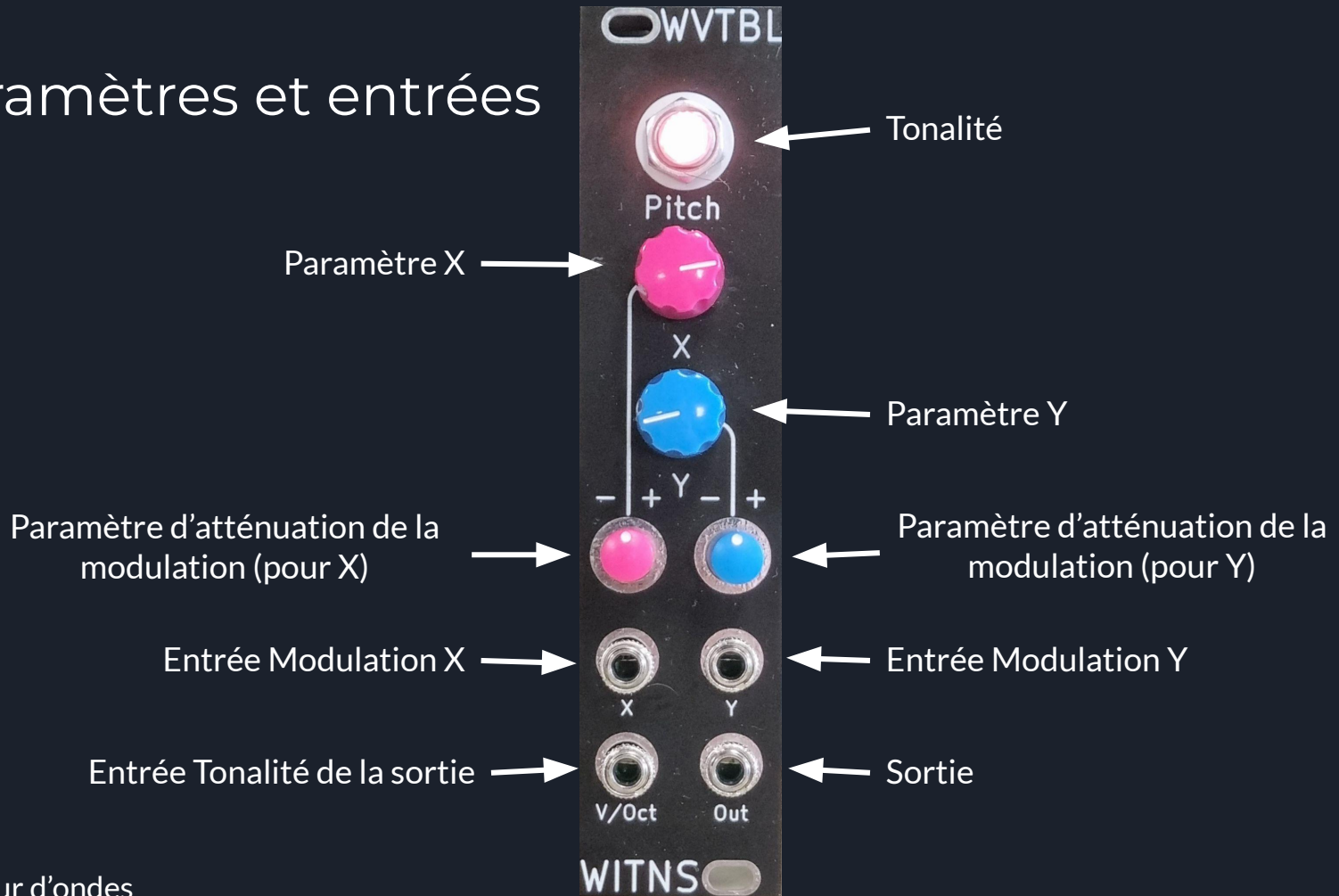
Sommaire

1. Synthétiseur mixeur d'ondes
2. Synthétiseur table d'onde

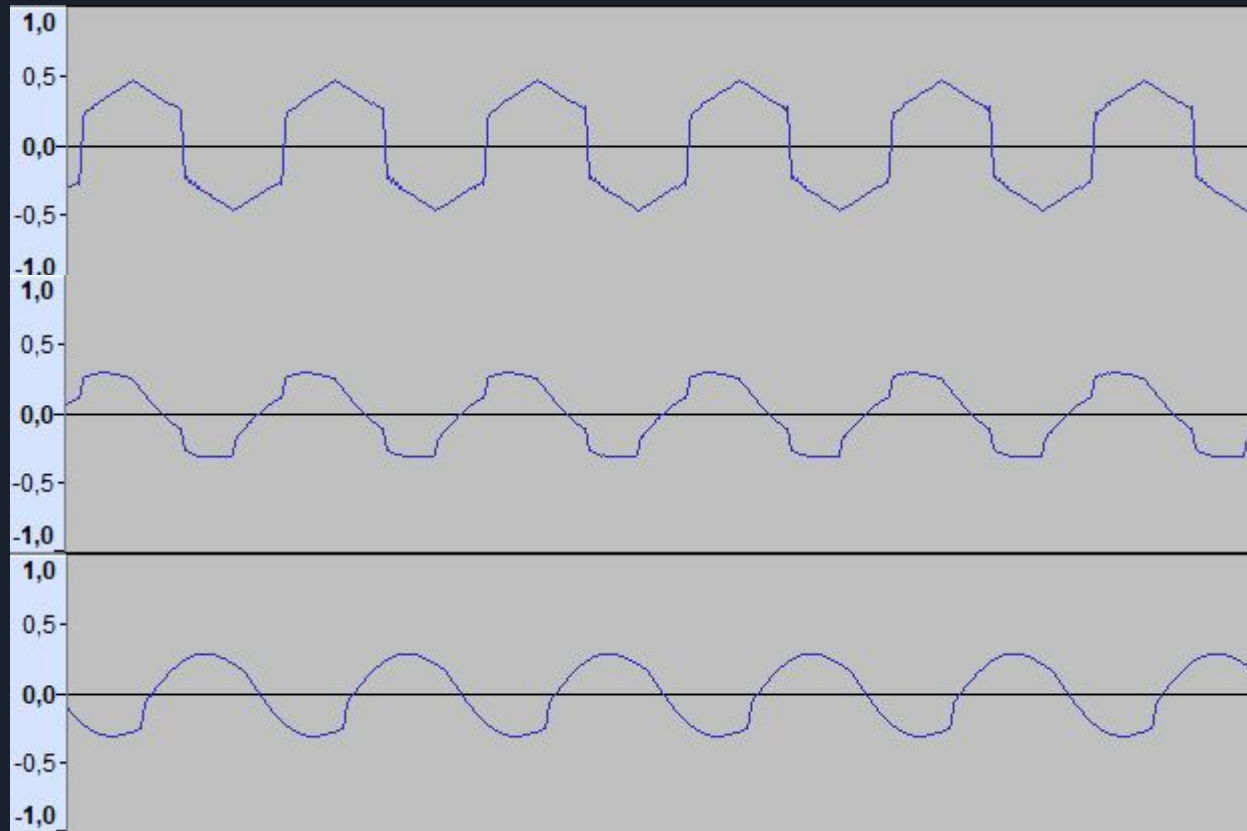
Principe de fonctionnement



Paramètres et entrées



Variation des paramètres



Synthétiseur Table d'onde

Start

```
1. importation des librairies numpy, pydub, matplotlib.pyplot et  
   mpl_toolkits.mplot3d, ainsi que la librairie os  
2. création des fonctions: get_waveform, normalize_array,  
   autocorr_fft, get_pseudo_period, get_waveform_array  
3. init input folder_path, limit_period,  
   data_path
```

```
waveform_array = get_waveform_array(n):
```

For arrêt de la boucle si le nombre de **fichiers** atteint **n**

21. **Condition** traitement du fichier uniquement s'il a l'extension .wav
22. **appel** de la fonction get_pseudo_period pour obtenir la valeur de la pseudo-période
23. **Condition** si la valeur de la pseudo-période est inférieure à limit_period, traitement du fichier suivant
24. **sinon**, appel de la fonction get_waveform pour extraire les sous-tableaux de taille $1/T$
25. ajout du tableau des sous-tableaux de taille $1/T$ à la liste waveform_list
26. incrémentation du compteur de fichiers traités
27. **retourne** la liste waveform_list

Générateur Table d'onde

-> création d'une fonction `figure_display` qui affiche chaque sous-tableau de la liste `waveform_array` sous forme de figure
-> création d'une fonction `write_waveform_c_array` qui prend la liste `waveform_array` et le chemin d'un fichier c en entrée

30. ouverture du fichier en écriture
31. écriture de la déclaration du tableau `waveform_array` en C
32. parcours de chaque fichier dans la liste `waveform_array`
33. écriture des sous-tableaux du fichier courant dans le fichier C
34. retourne le tableau des sous-tableaux de taille N

Générateur Table d'onde

get_waveform prend le chemin d'un fichier audio et un entier N en entrée

3. lecture du fichier audio et extraction des N premiers échantillons ou de la totalité du fichier audio si N est supérieur à la longueur du fichier

4. normalisation des échantillons extraits

5. division des échantillons en sous-tableaux de taille N

6. retourne le tableau des sous-tableaux de taille N

normalize_array normalise les valeurs d'un tableau en utilisant la valeur maximale

autocorr_fft calcule l'autocorrélation d'un signal audio en utilisant la FFT

get_pseudo_period prend un signal audio y et une fréquence d'échantillonnage fs en entrée

11. calcul de l'autocorrélation du signal audio en appelant la fonction `autocorr_fft`

12. suppression des valeurs négatives de lags

13. recherche du premier maximum après le pic principal

14. conversion de l'index en temps

15. retourne la valeur de la pseudo-période T