DU SYSTEME A LA FONCTION Projet : Synthèse d’un banc de filtres numériques

Sommaire

Introduction

Dans ce projet on avait pour but de réaliser un banc de filtre numérique. Pour ce faire il nous a fallu tout d'abord chercher les équations des différents filtres utilisés sur notre banc. On a donc commencé par les filtres « simples » passe haut et passe bas de premier ordre, puis on s'est occupé du deuxième ordre en n'oubliant pas les filtres passes bandes.

Ensuite il a fallu bien sur expérimenter toute la partie théorique recherchée précédemment. On va tout d'abord observer différents signaux afin d'observer le domaine auditif de l'Homme ainsi que d'autres signaux comme le bruit blanc. Tout ceci afin de programmer le logiciel Matlab afin d'obtenir un banc de filtre numérique.

I. Programmation sous Matlab

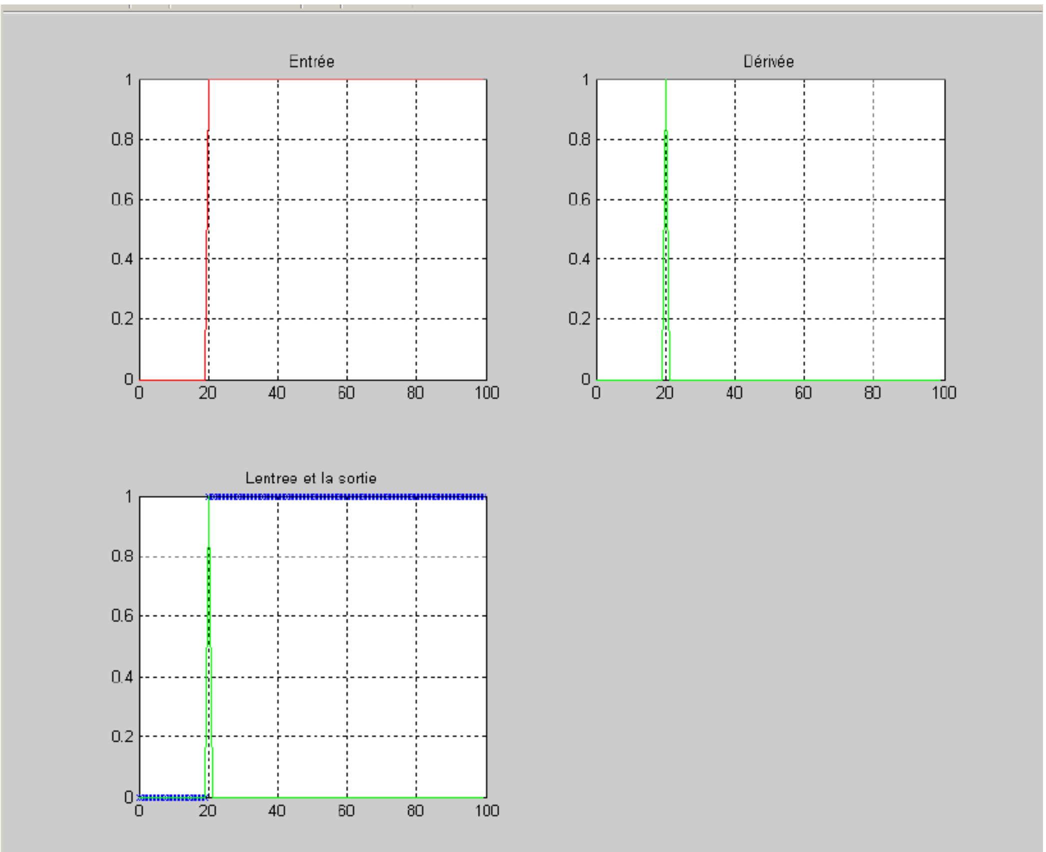
I. Modélisation et calculs sous Matlab

Q1) On obtient le chronogramme suivant pour les instructions saisies :

Choronogramme de la dérivée :

Sous Matlab :

```
%---------------Program° d'une dérivée--
for t=2:N     d(t)=k*(e(t)-e(t-1)); end
```



Q2)

Voici le message d'erreur :

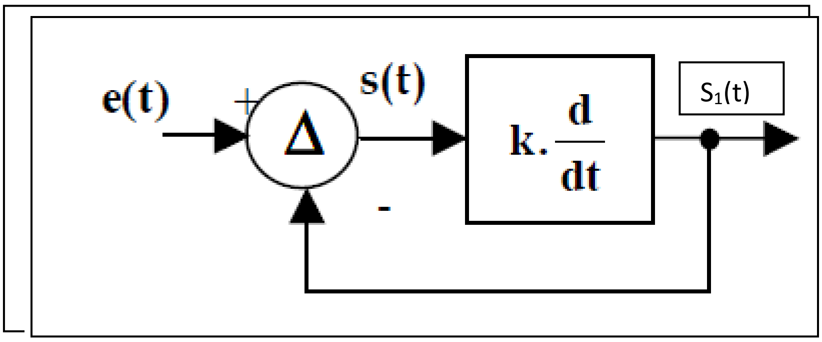
??? Error: File: L1derivativ.m Line: 13 Column: 35
Illegal use of reserved keyword "end".

II. Programmation des filtres.

On commence bien évidemment par déterminer les équations des filtres que nous utiliserons sur Matlab.

Un filtre passe bas et un filtre qui laisse passer les basses fréquences et qui atténue les hautes fréquences, c'est-à-dire les fréquences au dessus de la fréquence de coupure.

I. Filtrage passe-bas



On prend ce schéma comme base pour traiter les formules qui vont suivre.

Q3)

On part des formules connues du TP 3 : $S_2(t)=e(t)-S_1(t)$

On cherche S2 pour le filtre passe bas.

On a donc : $S_2(t)=e(t)-\frac{k*(d\,S_2(t))}{dt}$

On peut dès lors obtenir la formule suivante (en suivant la même démarche que pour le début du projet)

$$S_2(t)=e(t)-\frac{k*(S_2(t)-S_2(t-1))}{1}$$

Il devient donc évident que l'on a : $S_2(t)=e(t)-k*(S_2(t)-S_2(t-1))$

$$(k+1)S_2(t)=e(t)-k*S_2(t-1)$$

La suite devient donc :

On trouve donc α = 1/(k+1) et β = k/(k+1)

Nous obtenons donc :

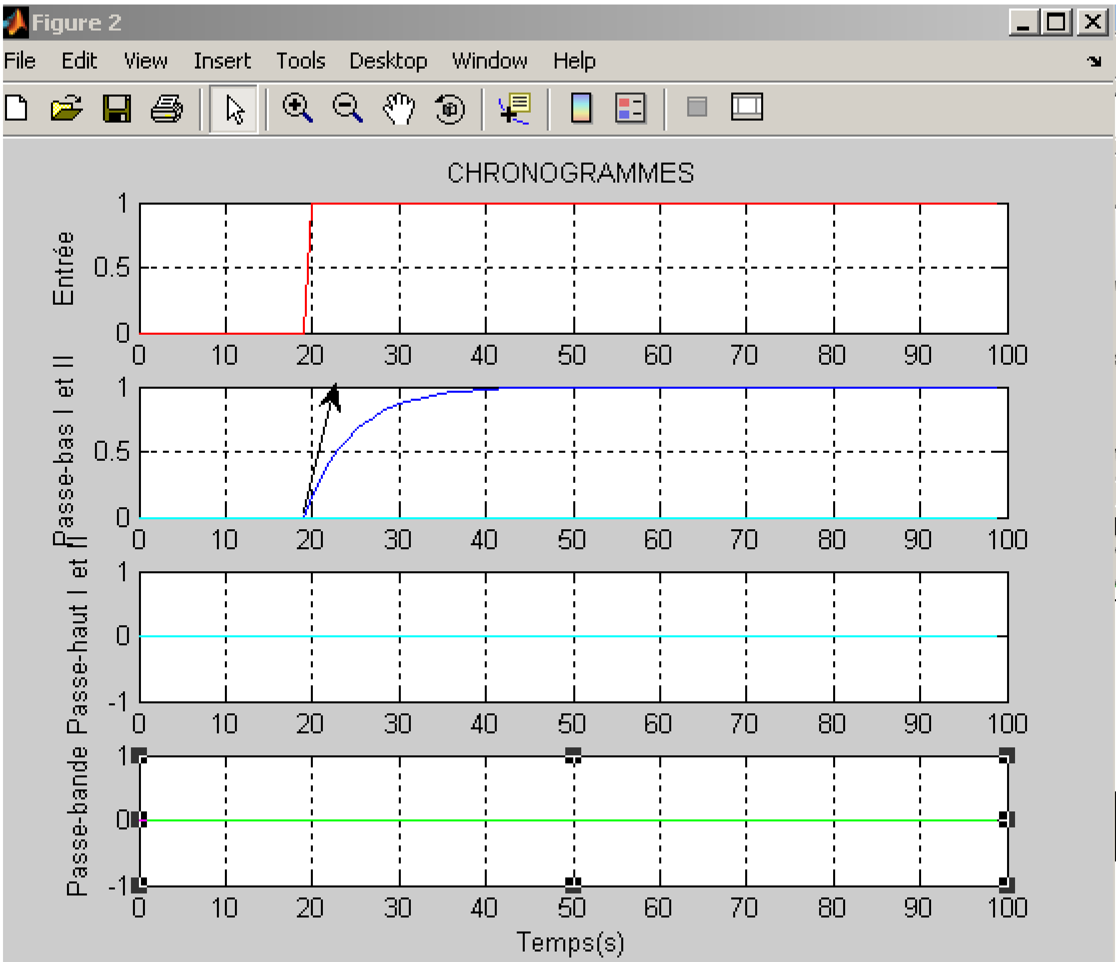
$$\alpha + \beta = \frac{1}{1+k} + \frac{k}{k+1} = \frac{1+k}{1+k} = 1$$

Cette formule est la vérification demandée dans le sujet pour la question 3.

Q4)

Nous avons programmer un filtre passe-bas et avons relever sa réponse indicielle pour différentes valeurs de k. Nous verrons à la fin de cette question que représente k.

K = 5



K = 25

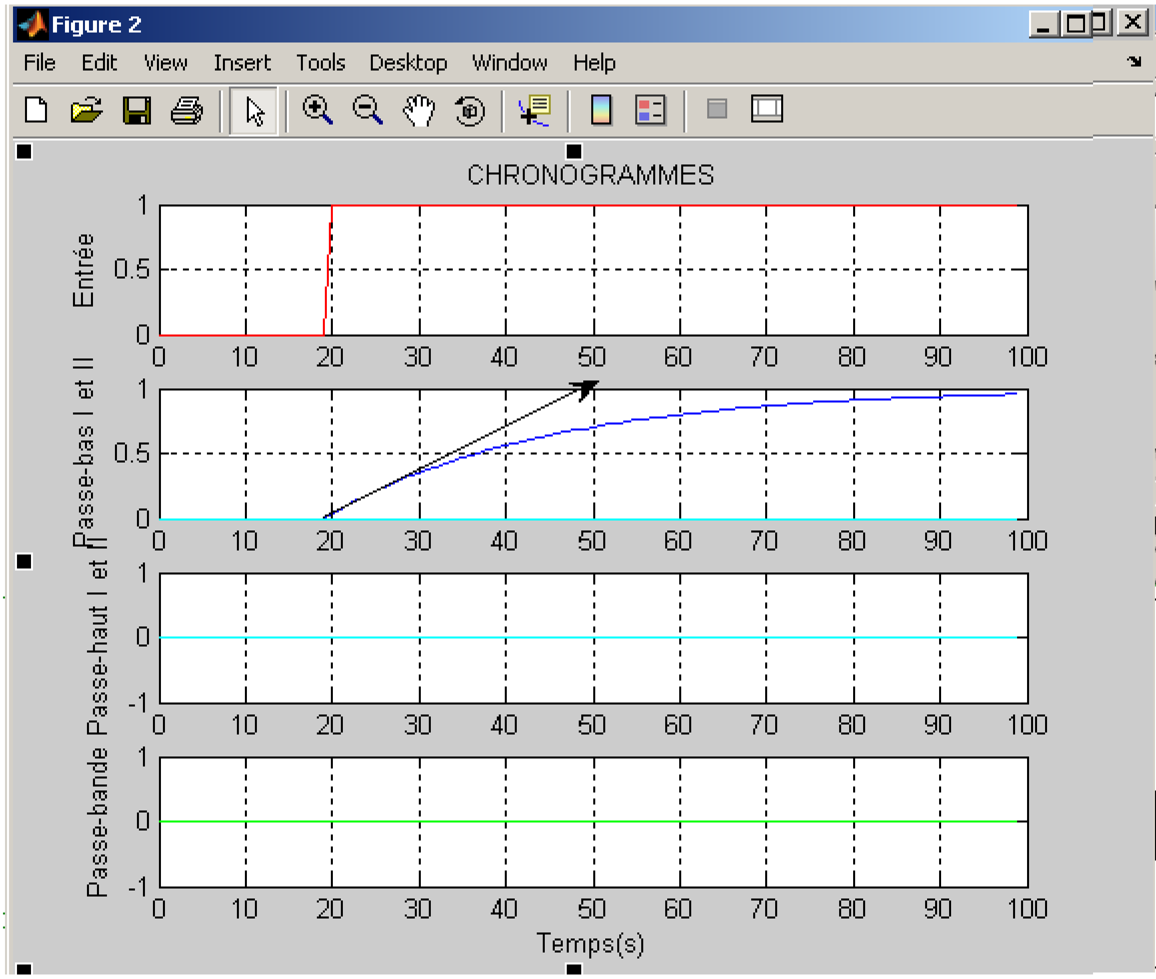
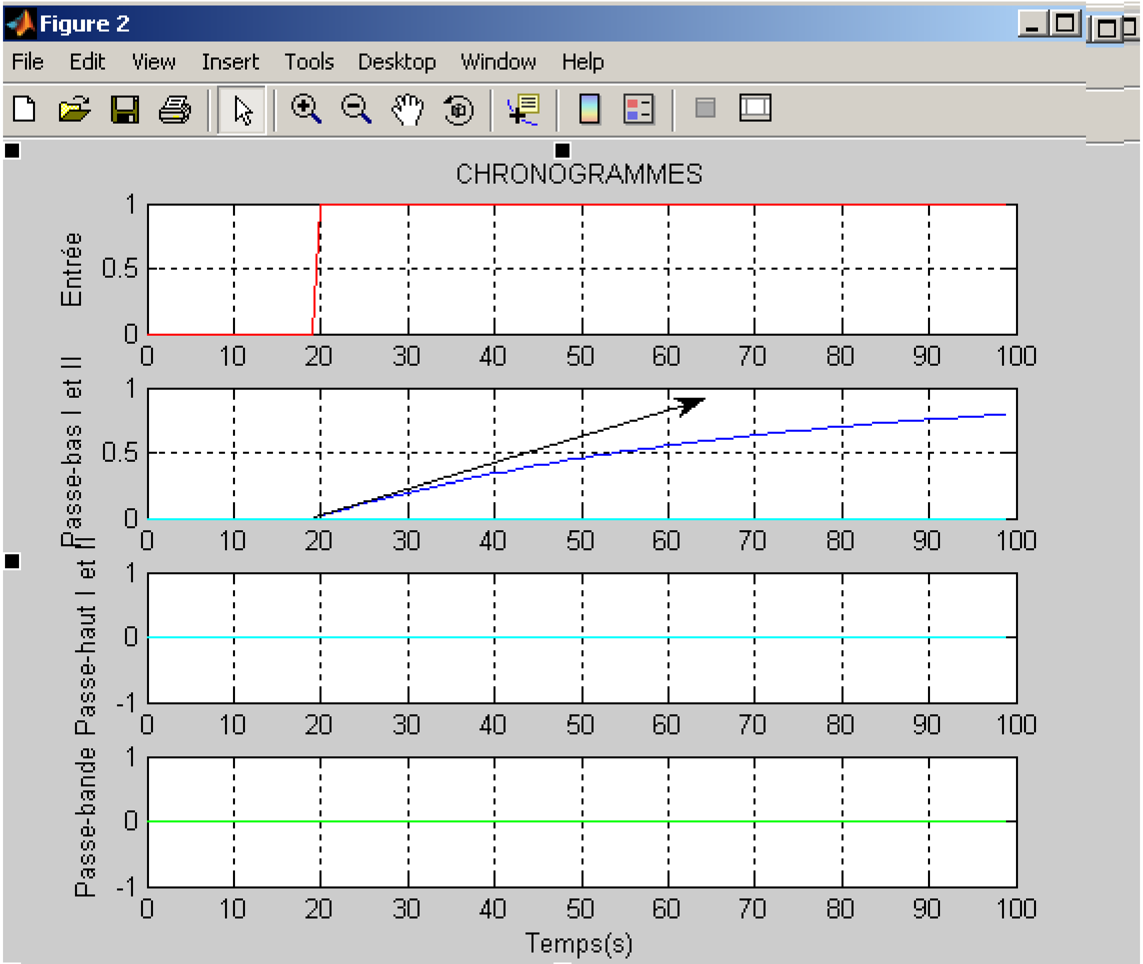


Figure 2
File Edit View Insert Tools Desktop Window Help
CHRONOGRAMMES
Entrée
Passe-bas I et II
Passe-haut I et II
Passe-bande
0 10 20 30 40 50 60 70 80 90 100
Temps(s)

K = 50



K est la constante de temps.

Plus k augmente, plus la fréquence de coupure augmente.
Plus k augmente, plus la constante de temps va croitre ce qui va ralentir la réponse indicielle, la courbe augmentera plus lentement du au temps de calcul dans l'intégrateur.

Cela est du à la formule : $f=(2\pi k)^{-1}$

II. Filtrage passe-haut

Q5)

Pour un filtre passe-haut, nous avons l'équation suivante :

$$s_2(t) = e(t) - y(t)$$

$$s_2(t) = k\frac{d}{dt} * e(t) - k\frac{d}{dt} * s_2(t)$$

$$s_2(t) = k * e(t) - k * e(t-1) - k * [s_2(t) - s_2(t-1)]$$

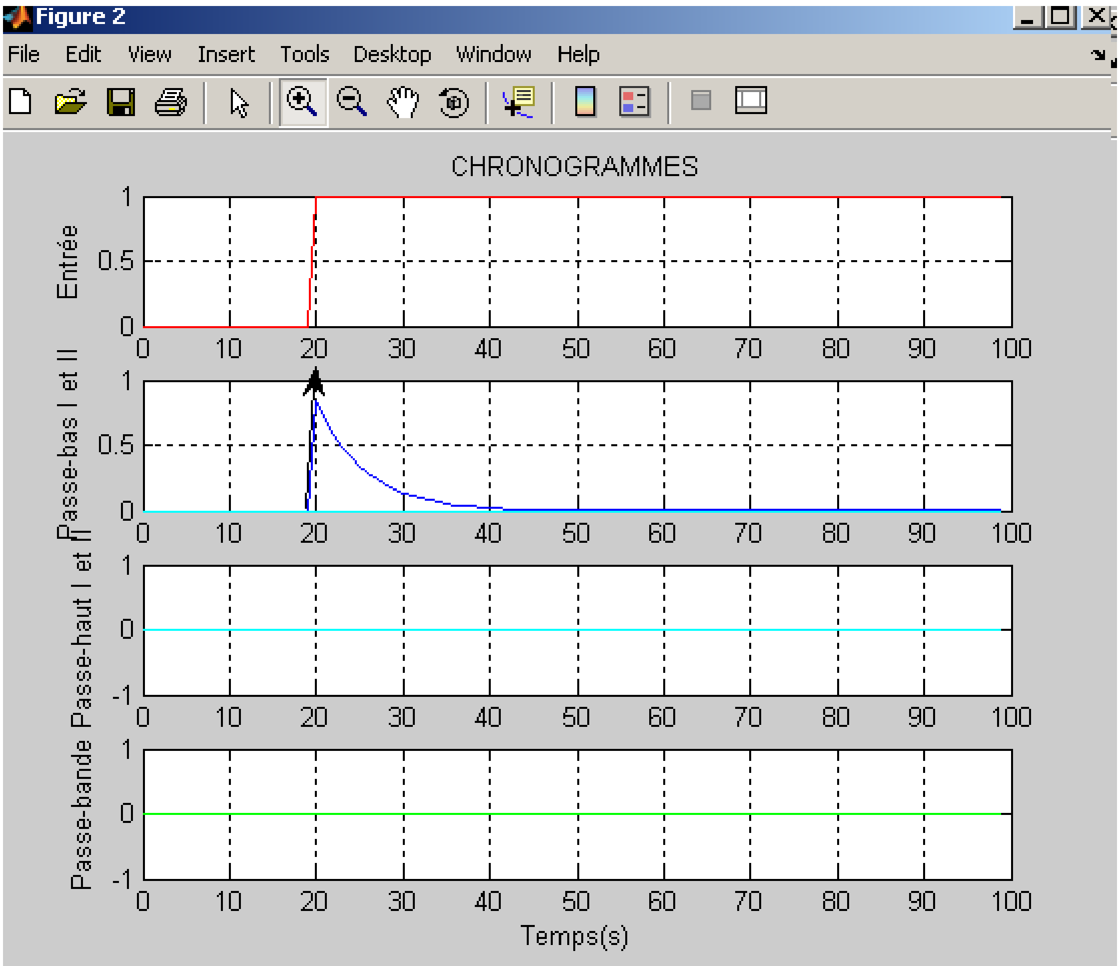
$$s_2(t) = k * e(t) - k * e(t-1) - k * s_2(t) + k * s_2(t-1)$$

$$(1+k) * s_2(t) = k * [e(t) - e(t-1) + s(t-1)]$$

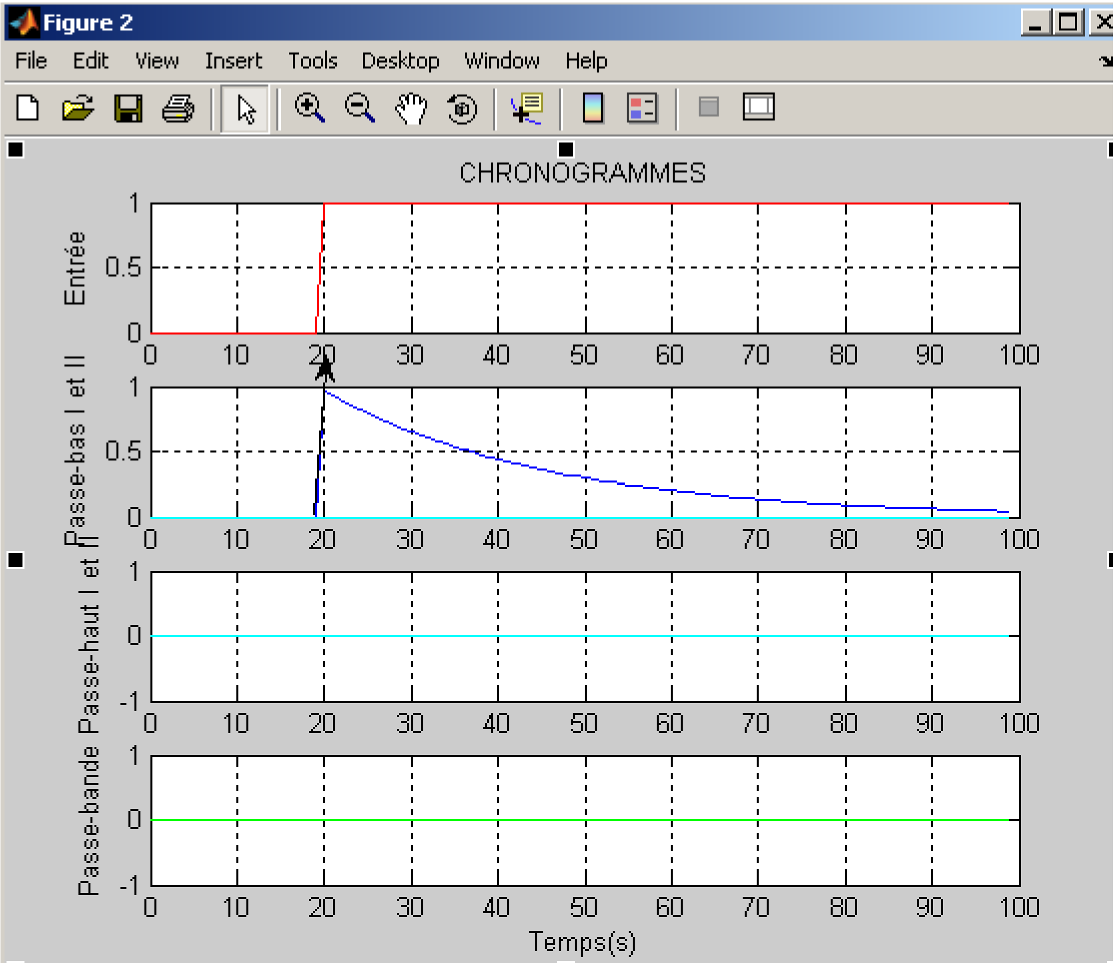
$$s_2(t) = \frac{k}{k+1} * [e(t) - e(t-1) + s_2(t-1)]$$

Pour programmer ce filtre sous le logiciel ***MATLAB***, il nous suffit donc d'utiliser β, précédemment égal à $\frac{k}{k+1}$ et l'inserons dans le code.

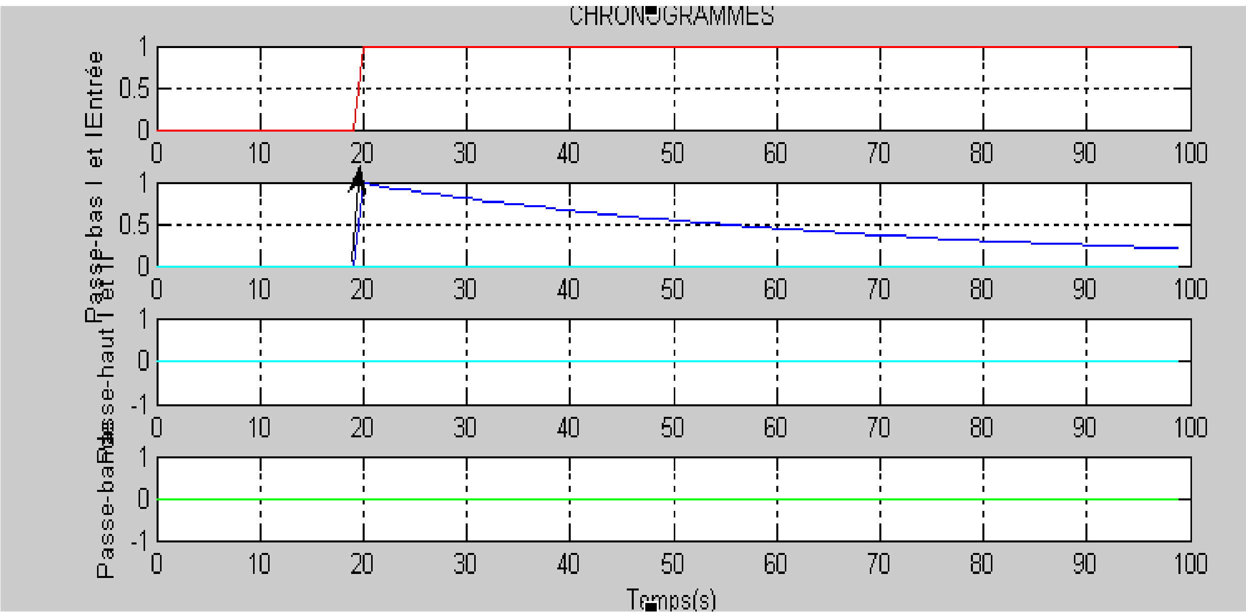
K = 5



K = 25



K = 50



On constate encore que plus k augmente, plus la réponse indicielle est lente.

III. Filtrage du second ordre

Q6)

Pour réaliser un filtrage du seconde ordre, on va tenter de calculer la dérivée seconde d'un signal sur un calculateur numérique. On remarque les filtres doivent être placés en cascades pour obtenir un filtre de second ordre.

$$\frac{d''s(t)}{dt} = \frac{d\frac{d\,s(t)}{dt}}{dt} = \frac{d\,k\,e(t) - e(t-1)}{dt}$$

$$\frac{d''s(t)}{dt} = k^2[e(t) - 2e(t-1) + e(t-2)$$

D'où

$$ss(t) = \frac{1}{k+1}s(t) + \frac{k}{k+1}ss(t-1)$$

Q7)

Synthèse filtre passe-bas

S(t) = alpha*e(t) + beta * s(t – 1)

Ss(t) = alpha * s(t) + beta*ss(t – 1)

Ss(t) = alpha (alpha * e (t) + beta * s (t -1) + beta * ss(t – 1)

Ss(t – 1) = alpha * s(t – 1) + beta * ss(t – 2)

Ss(t) = 1/(k+1) * s(t) + k / (k+1) * ss(t – 1)

Synthèse filtre passe-haut

$$Z(t) = k * \frac{dS(t)}{dt} = k * \frac{d(e(t) - z(t))}{dx} = k[e(t) - z(t) - (e(t-1) - z(t-1)]$$

$$= k * e(t) - k * z(t) - k * e(t-1) + k * z(t-1)$$

$$(k+1) * Z(t) = k * e(t) - k * e(t-1) + k * z(t-1)$$

$$Z(t) = \frac{k}{1+k} * (e(t) - e(t-1) + z(t-1))$$

IV. Filtrage passe-bande

Q8)

On peut réaliser un filtrage passe-bande en partant de ces deux formules :

$$s(t) = k_1\big(y(t) - y(t-1)\big)$$

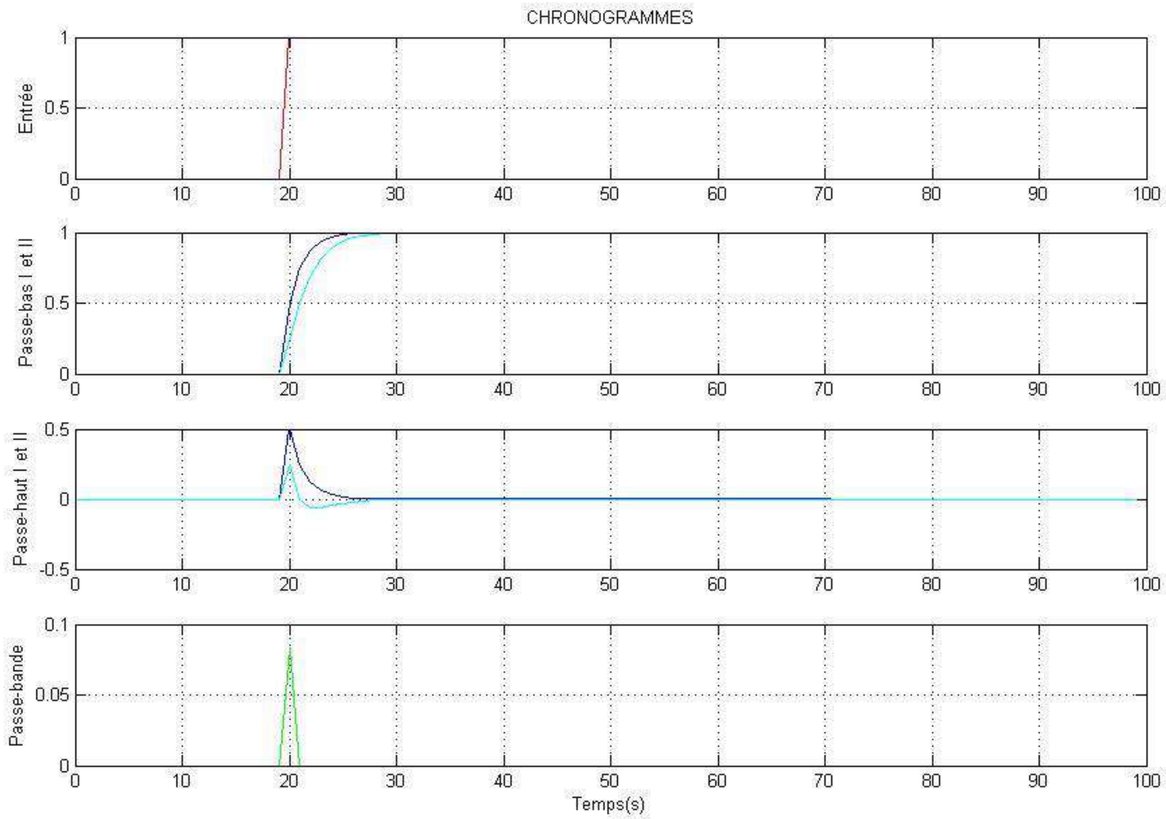
$$y(t) = x(t) - s(t) - k_2\frac{ds(t)}{dt}$$

$$y(t) = x(t) - s(t) - k_2[s(t) - s(t-1)]$$

On obtient finalement :

$$s(t) = k1[(x(t) - s(t)) - k_2(s(t) - s(t-1))] - [x(t-1) - s(t-1) - k_2\big(s(t-1) - s(t-2)\big)]$$

Voici les résultats graphiques



III. Réponse impulsionelle et réponse harmonique

I. Observation des signaux et calculs des spectres

Q0

De nombreuses opérations sont possibles sous Matlab :
- La syntaxe suivante effectue le produit scalaire :

*x.*y*

- De même celle-ci effectue la multiplication terme à terme :

x*y'

Cependant nous ne pouvons utiliser l'opération *x*y,* en effet ceci effectuerait le produit de deux vecteurs horizontaux ce qui est impossible.

Nous pouvons également effectuer des calculs complexes sous Matlab :

$$z = r * e^{(j*\theta)}$$

ou encore le conjugué de z, *conj(z)* :

$$\bar{z} = r * e^{(-j\theta)}$$

ainsi que la multiplication de deux complexes :

$$z * \bar{z} = r^2 * e^{(j\theta)} * e^{(-j\theta)} = r^2$$

et le calcul du module de ceux-ci :

$$\sqrt{z * \bar{z}} = r$$

Q1

On échantillonne à une vitesse de 44.1 kHz. Il nous faut donc 44100 échantillons pour représenter le signal sur une fenêtre d'une seconde. La fréquence maximale des signaux que nous pourrons produire ou traiter sera donc, d'après le théorème de Shannon :

$f_{max=} = f_e/2 = 44100/2 = 22050$ Hz.

Q2

Le tableau ci-dessous comportent les différentes valeurs de f(allant de 20Hz à 50kHz), f_B, f_H

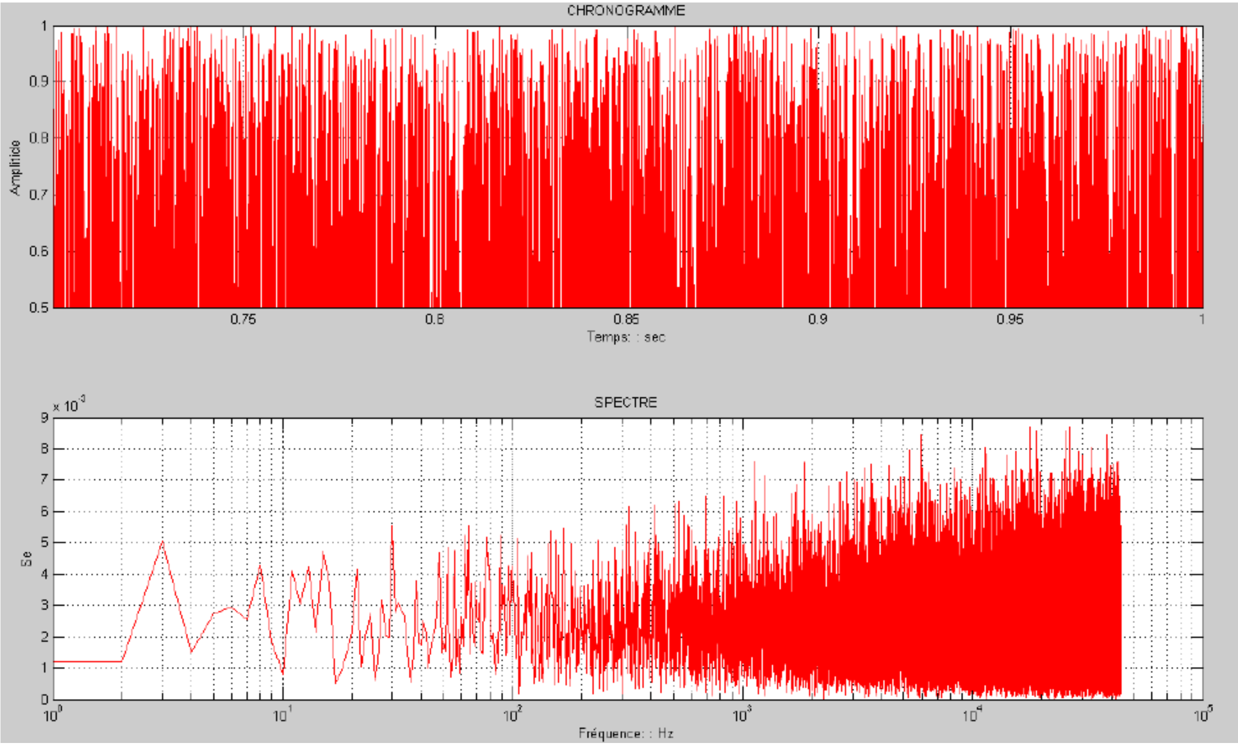
f (Hz)	**20**	**50**	**100**	**200**	**500**	**1.000**	**2.000**	**5.000**	**10.000**	**20.000**	**50.000**
f_B (Hz)	20	50	100	200	500	1.000	2 .000	5.000	10.000	20.000	59.000
f_H (kHz)	44,1	44,02	44.1	43,7	43,5	43,2	41,8	39,3	33,9	23,9	38,3
Audible	Y N	Y N	Y N	Y N	Y N	Y N	Y N	Y N	Y N	Y N	Y N

Nous pouvons remarquer qu'au dessus de 50kHz nous avons un repliement de spectre. Nous pouvons en déduire que les ultrasons ne peuvent être filtrés par l'odinateur. La bande passante de l'ouï humain étant comprise entre 20Hz et 20kHz cela n'est pas gênant.

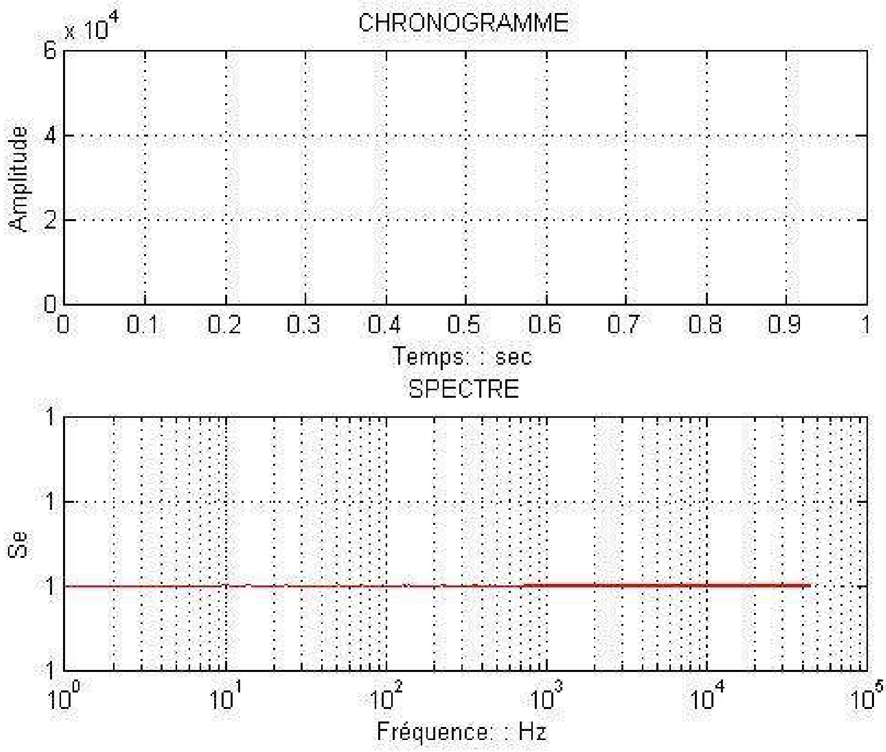
Q3

C'est un son très riche, car il comporte toutes les fréquences, de façon aléatoire. Par analogie, la lumière blanche contient toutes les longueurs d'ondes. On entend comme un signal radio…

Nous allons maintenant générer un bruit blanc (aléatoire et similaire à une bande FM) et observer son spectre :



Nous allons maintenant étudier le spectre d'une impulsion que nous n'avons pas entendu (silence) :



Nous constatons un spectre constant de niveau 1 .

II. Synthèse des filtres

Q4.

Pour une entrée impulsionnelle on n'entend rien. On peut seulement voir l'impulsion sur le chronogramme à t=0.

Chaque équation d'un filtre est composée de différentes constantes de temps

$k = \frac{1}{\omega c}$

$$k2 = \frac{q}{\omega 0}$$

$$q = \frac{\omega 0}{\Delta\omega}$$

En posant $\Delta\omega = 2\pi\Delta f$ ainsi qu'en prenant les fréquences réelles on obtient les résultats suivants :

$$q = \frac{\omega 0}{2\pi\Delta f}$$

$$k2 = \frac{1}{2\pi\Delta f} = \frac{fc}{\Delta\omega} * \frac{1}{\omega 0} = \frac{20}{20} * \frac{1}{2\pi fc} = 0{,}0079$$

$$k1 = \frac{2\pi\Delta f}{\omega 0^2} = \frac{1}{\frac{\omega 0^2}{\Delta\omega}} = \frac{1}{\frac{\omega 0^2}{2\pi\Delta f}} = 1$$

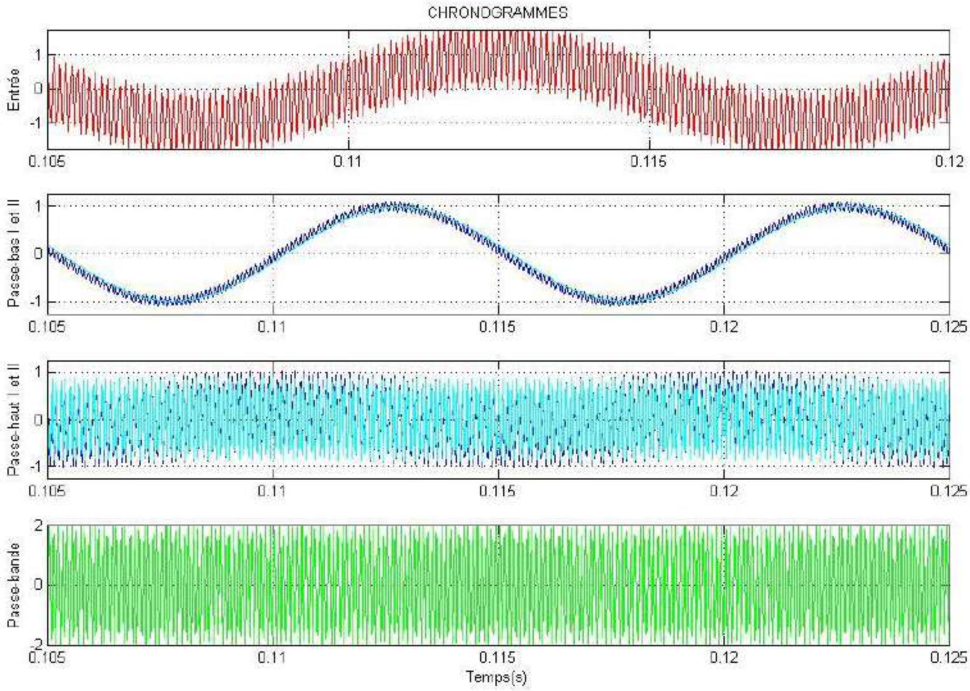
$$k = \frac{1}{2\pi fc} = 1{,}5915.\,10^{-4}$$

III. Réponse harmonique

Q6

Nous allons à présent attaquer les filtre en utilisant un signal composite de la forme e(t) = sin $(2\pi f_1 t)+\sin(2\pi f_2 t)$ où^

F_1 = 100Hz F_2 = 10kHz

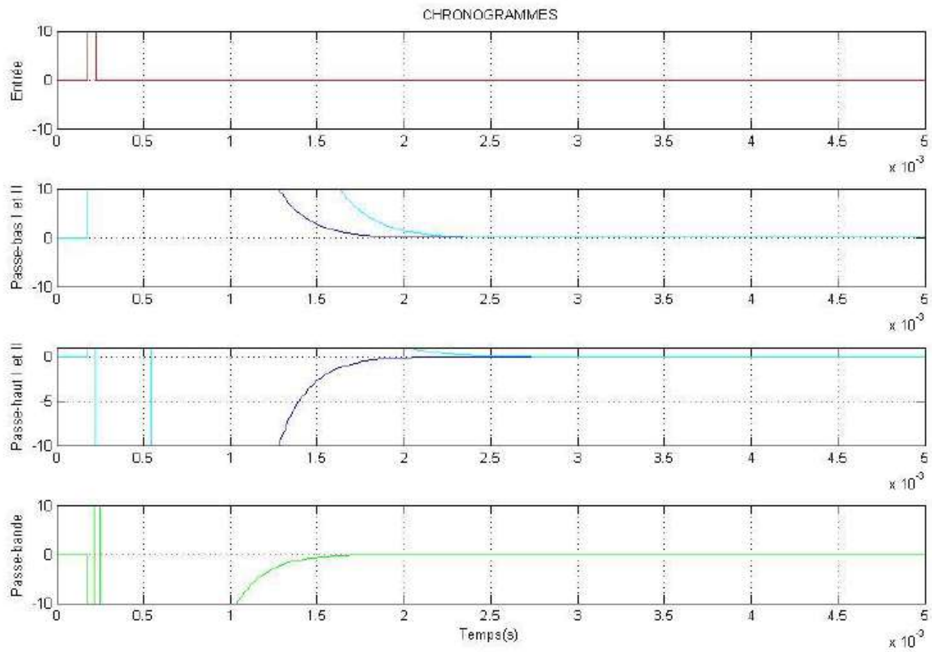


Nous pouvons observer que les filtres sont fonctionnels, en effet nous pouvons bien distinguer la différence entre les hautes fréquences (son aigus) et les basses fréquences (son graves).

On remarque que les filtres opèrent bien, puisque lorsque l'on écoute les signaux successivement, on entend bien la différence entre les sons aigus et les sons graves.

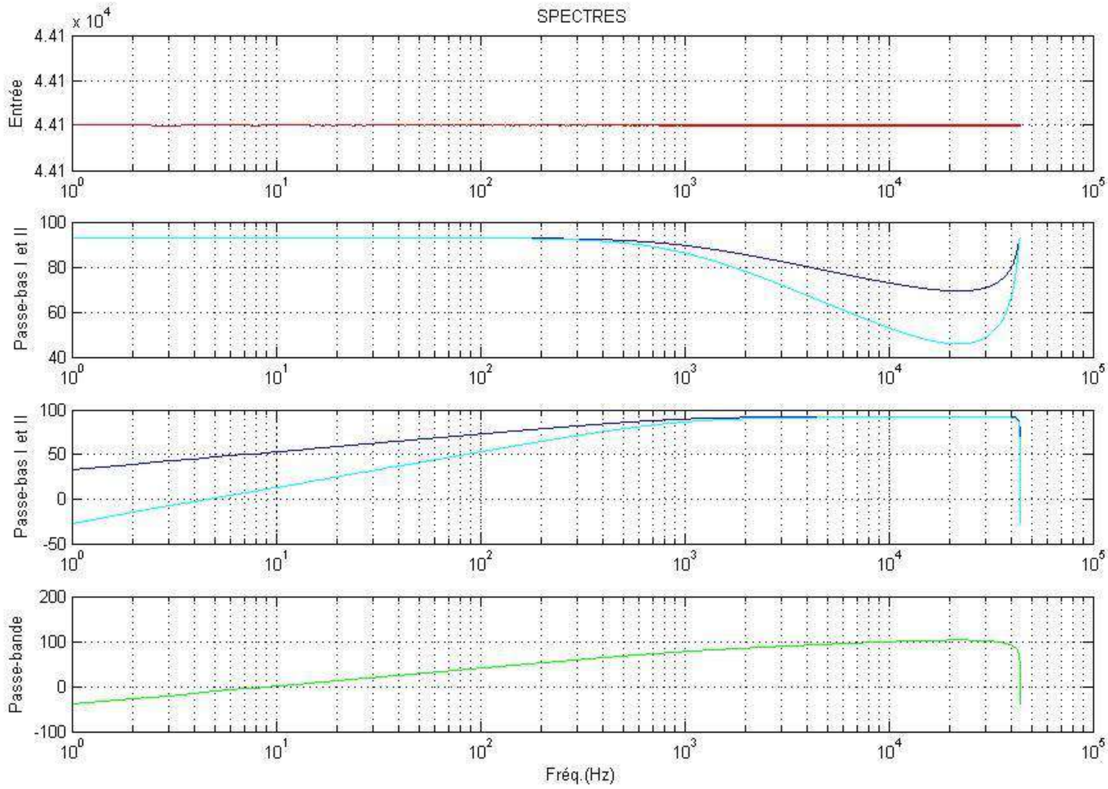
IV. Réponse impulsionnelle

Après génération d'une impulsion nous avons obtenu les chronogrammes et spectres suivants :



Nous pouvons remarquer que les chronogrammes correspondent aux diagrammes de Bode de chaque filtre, sauf pour le passe-bande. Les filtres passe-bas et passe-haut d'ordre 1 et 2 ont quant à eux une fréquence de coupure de 1kHz ainsi que des affaiblissements respectifs de -20dB, -40dB, +20dB et enfin +40dB.

Voici les spectres obtenus :



Nous remarquons que ces mesures sont certainement fausses, en effet au dessus de f_{max} = 22 050hz cela ne correspond plus au diagramme de Bode.

Conclusion

En conclusion, nous pouvons dire que nous avons appris de nombreuses choses concernant les ondes et les filtres qui ont la capacité de séparé des sons en élevant soient les ondes aigus soient les graves. De plus, ce projet nous a permis d'apprendre a nous servir de Matlab un logiciel assez puissant quand on arrive a connaitre toutes ses fonctionalités.