

CHATIRON Thibault

DENG Fangzhou

SRT3

Automne 2013

TP n°1 :
Modulation et démodulation

SY06

Sommaire

Introduction :.....	3
Modulation d'amplitude	4
Transformée de Fourier	6
Modulation.....	8
Transformée de Fourier du signal avec porteuse	10
Signal total.....	12
Démodulation.....	14
Conclusion	20

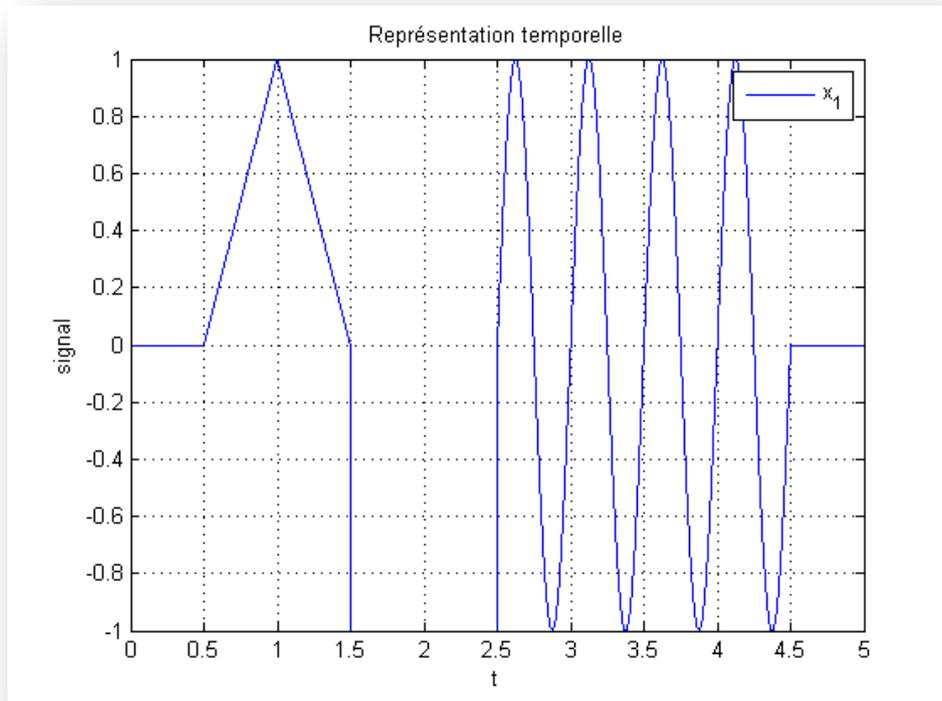
Introduction :

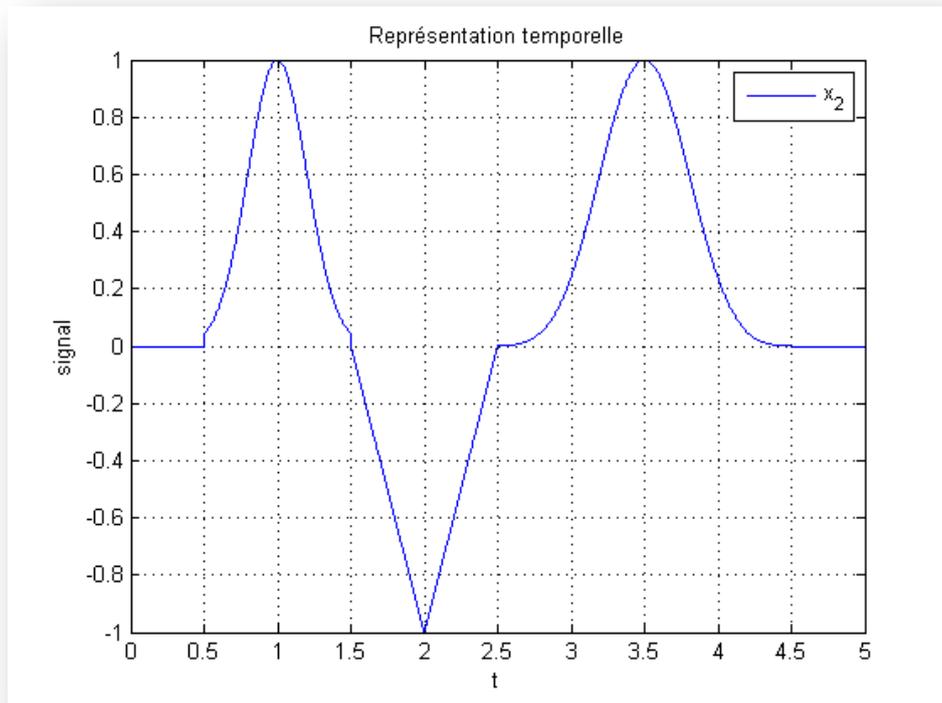
L'objectif de ce TP est de modéliser la modulation d'un signal composé de deux signaux, puis la démodulation de ses deux signaux distincts envoyés en même temps pour enfin essayer d'isoler et de récupérer uniquement l'un des deux par une technique de filtrage.

Nous utiliserons l'outil Matlab afin de simuler les différents signaux et filtres, tout en modifiant les différents paramètres pour pouvoir en étudier l'influence.

Modulation d'amplitude

Sous Matlab, nous proposons d'étudier la modulation d'amplitude sur deux signaux que l'on ajoute l'un à l'autre. Le premier signal est constitué de plusieurs fonctions (signal nul, triangulaire, rectangulaire, sinusoïdale et nul) et un deuxième signal (signal nul, Gaussien, triangulaire, Chebychev et nul).





Code Matlab

```

clc
clear
close all

Fe = 1000;
f1 = 250;
f2 = 350;
t = (0:1/Fe:5)';
ttmp = (0:1/Fe:2-1/Fe)';
ytmp = sin(2*pi*2*ttmp);
x1 = [zeros(500,1); triang(1000); -rectwin(1000); ytmp ;zeros(501,1)];
x2 = [zeros(500,1); gausswin(1000); -triang(1000); chebwin(2000)
;zeros(501,1)];

figure(1);
plot(t,x1);
xlabel('t');
ylabel('signal');
grid on;
legend('x_1');
title('Représentation temporelle');

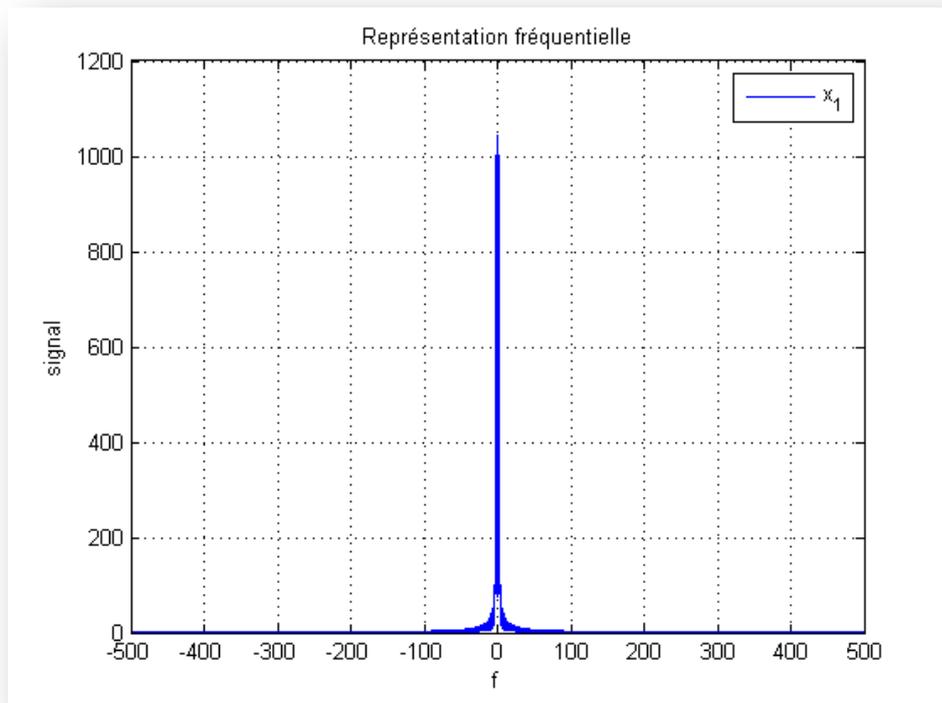
figure(2);
plot(t,x2);
xlabel('t');
ylabel('signal');
grid on;
legend('x_2');
title('Représentation temporelle');

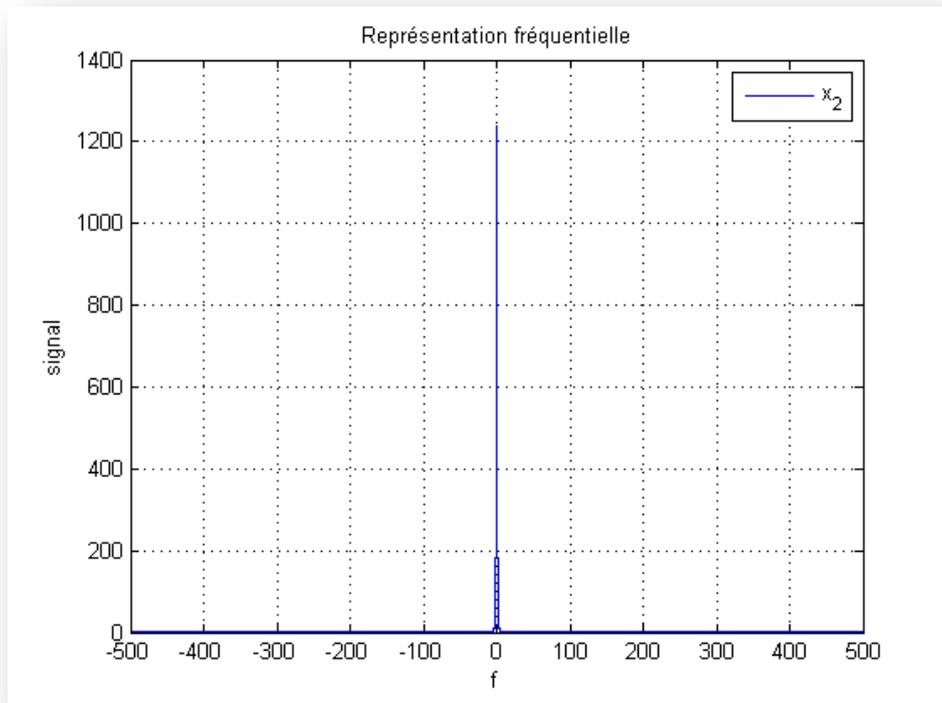
```

Transformée de Fourier

Pour la suite du TP, nous allons calculer la transformée de Fourier de chaque signal afin de pouvoir les sommer pour obtenir le signal total de sortie.

Voici la représentation graphique de la transformée de Fourier de chaque signal :





Code Matlab

```

N = length(t);

k=0;

for f = -Fe/2 : Fe/N : Fe/2
k = k+1;
X1(k) = x1' * exp(2*j*pi*f*t);
end

figure(3);
plot((-Fe/2 : Fe/N : Fe/2),abs(X1));
xlabel('f');
ylabel('signal');
grid on;
legend('x_1');
title('Représentation fréquentielle');

k = 0;

for f = -Fe/2 : Fe/N : Fe/2
k = k+1;
X2(k) = x2' * exp(2*j*pi*f*t);
end

figure(4);
plot((-Fe/2 : Fe/N : Fe/2),abs(X2));
xlabel('f');
ylabel('signal');
grid on;
legend('x_2');
title('Représentation fréquentielle');

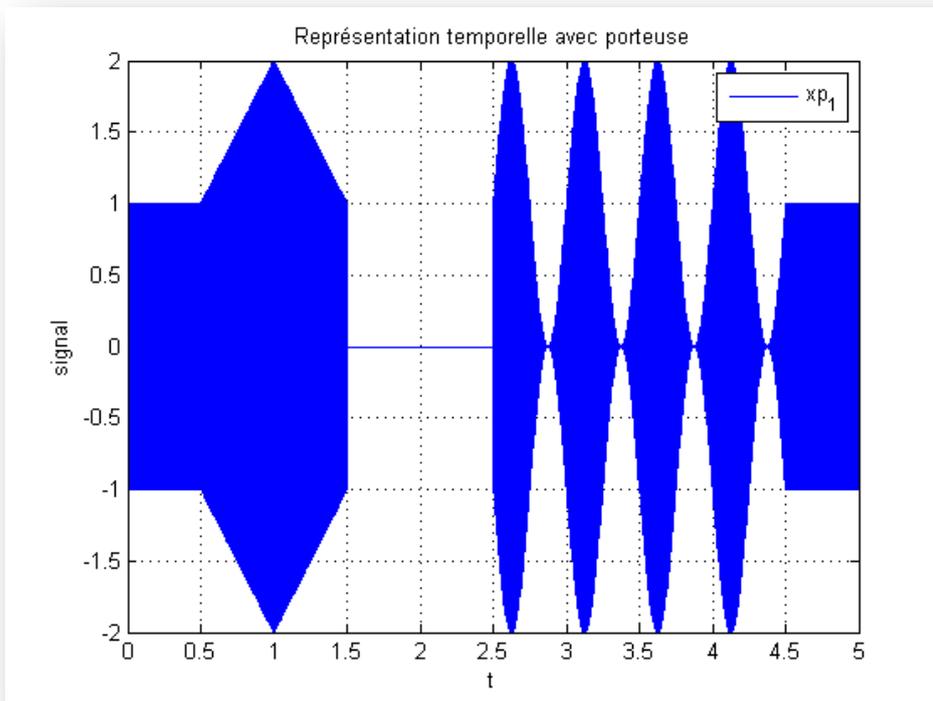
```

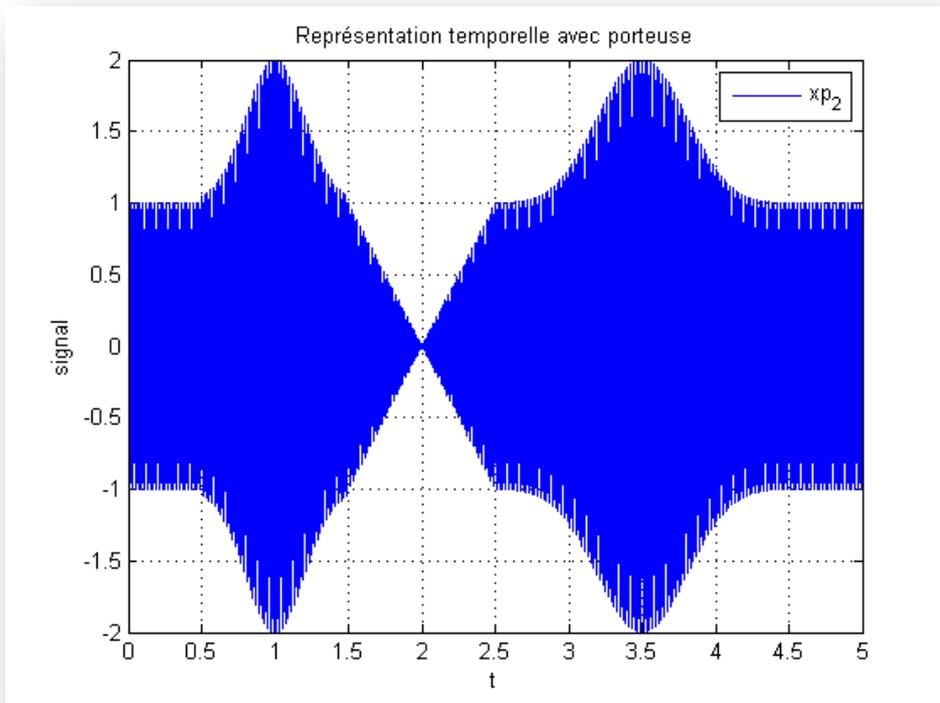
Modulation

Pour analyser et traiter les signaux, nous allons les moduler à l'aide d'une porteuse sous la forme $x_p = (1+a*x) * \cos(w*t)$

On prendra $a=1$ par défaut.

Voici la représentation graphique temporelle du signal modulé :



**Code Matlab**

```
w1=2*pi*f1;
w2=2*pi*f2;

xp1=(1+x1).*cos(w1*t);
xp2=(1+x2).*cos(w2*t);

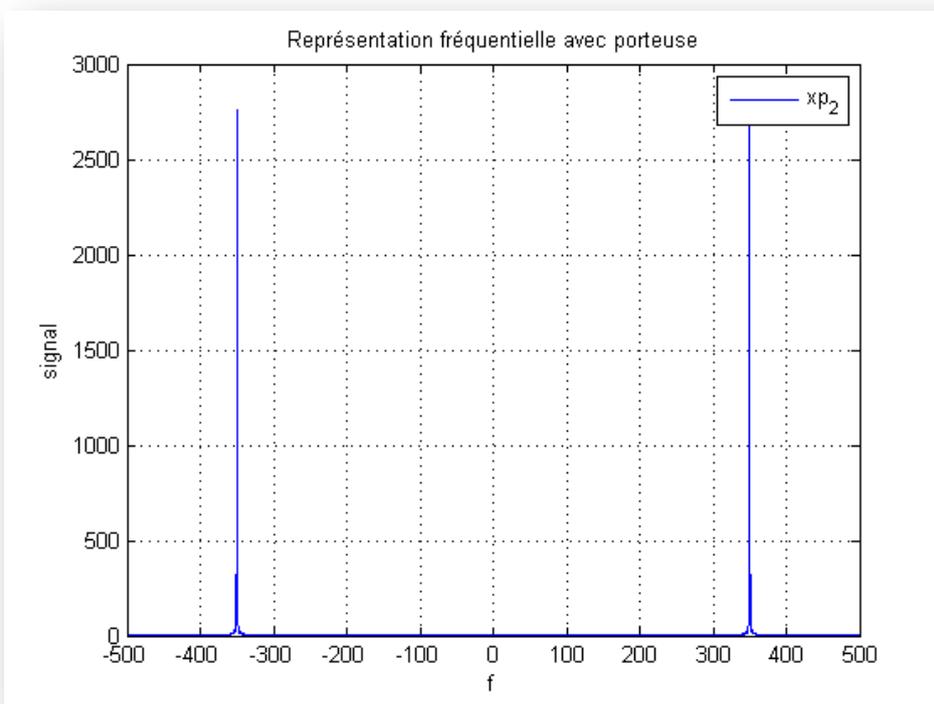
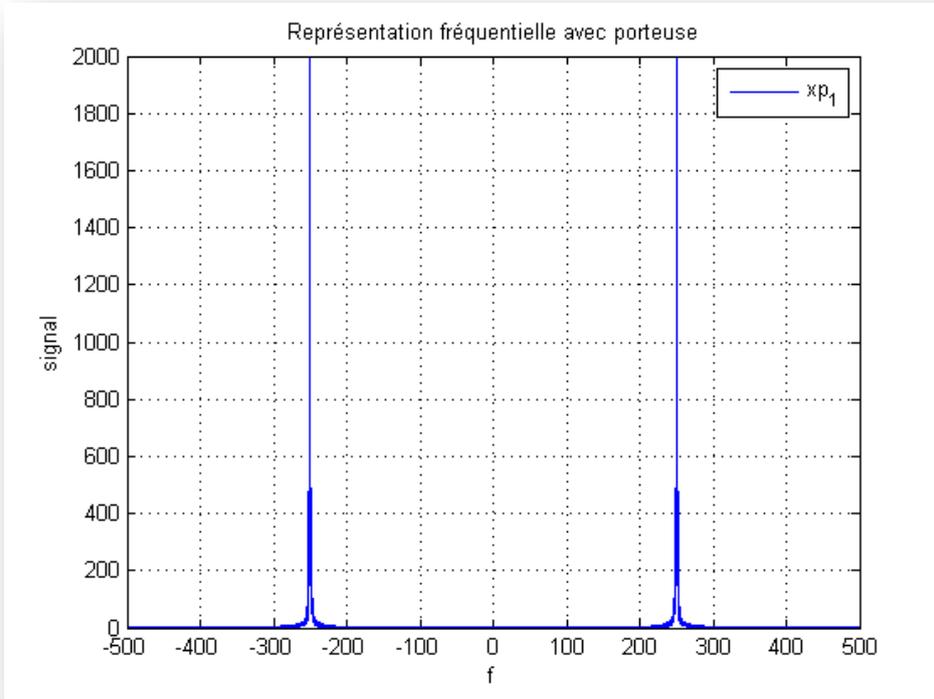
figure(5);
plot(t,xp1);
xlabel('t');
ylabel('signal');
grid on;
legend('xp_1');
title('Représentation temporelle avec porteuse');

figure(6);
plot(t,xp2);
xlabel('t');
ylabel('signal');
grid on;
legend('xp_2');
title('Représentation temporelle avec porteuse');
```

Transformée de Fourier du signal avec porteuse

Après avoir représenté le signal modulé dans le domaine temporel, on calcule la transformée de Fourier afin d'obtenir les composantes fréquentielles.

Voici la représentation fréquentielle des signaux modulés :



Code Matlab

```
N = length(t);

k=0;

for f = -Fe/2 : Fe/N : Fe/2
k = k+1;
Xp1(k) = xp1' * exp(2*j*pi*f*t);
End

figure(7);
plot((-Fe/2 : Fe/N : Fe/2),abs(Xp1));
xlabel('f');
ylabel('signal');
grid on;
legend('xp_1');
title('Représentation fréquentielle avec porteuse');

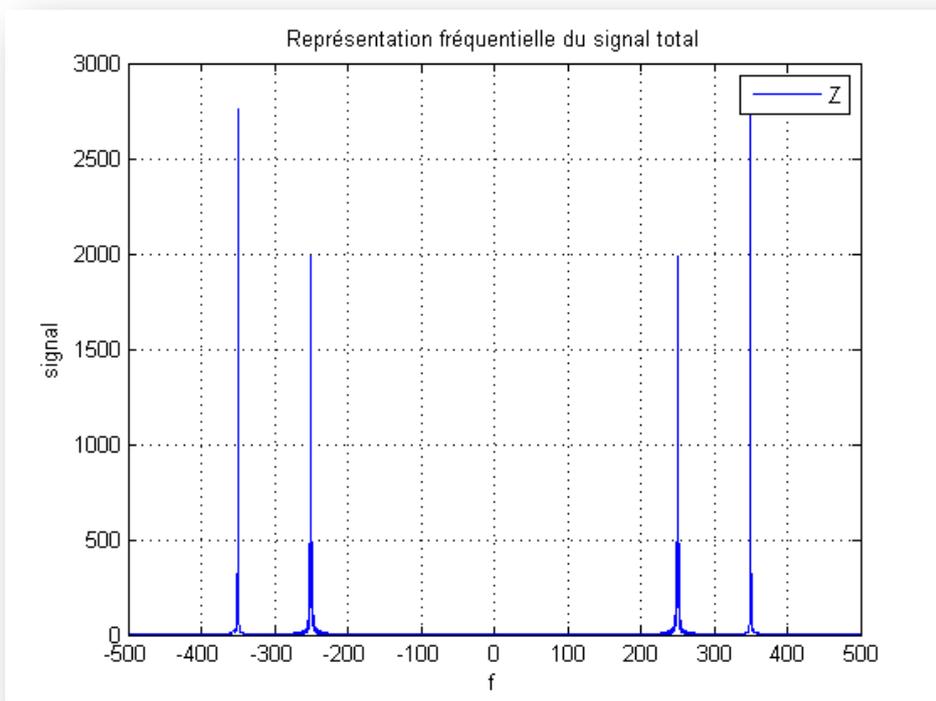
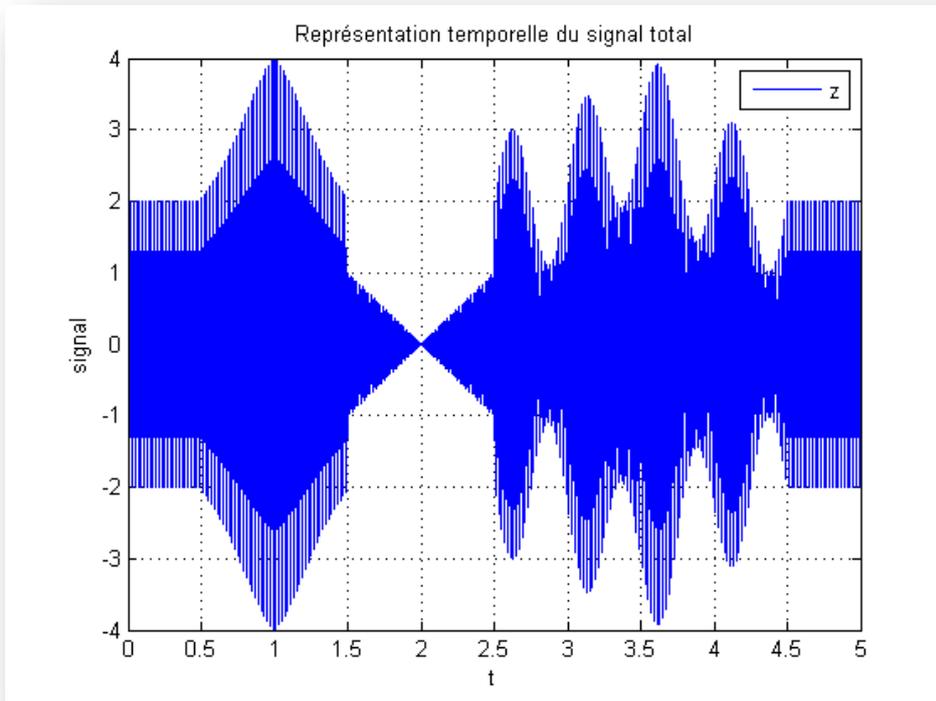
k = 0;

for f = -Fe/2 : Fe/N : Fe/2
k = k+1;
Xp2(k) = xp2' * exp(2*j*pi*f*t);
End

figure(8);
plot((-Fe/2 : Fe/N : Fe/2),abs(Xp2));
xlabel('f');
ylabel('signal');
grid on;
legend('xp_2');
title('Représentation fréquentielle avec porteuse');
```

Signal total

Le signal total est obtenu à partir de la somme des deux signaux modulés dont en voici la représentation graphique :



Code Matlab

```
z=xp1+xp2;

figure(9);
plot(t,z);
xlabel('t');
ylabel('signal');
grid on;
legend('z');
title('Représentation temporelle du signal total');

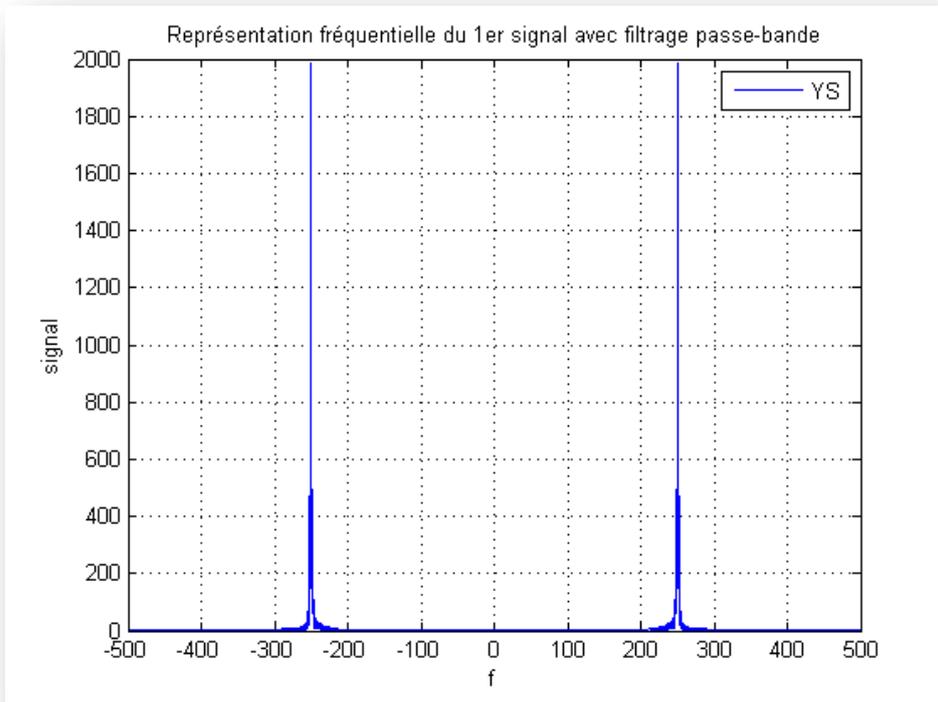
k = 0;
for f = -Fe/2 : Fe/N : Fe/2
k = k+1;
Z(k) = z' * exp(2*j*pi*f*t);
End

figure(10);
plot((-Fe/2 : Fe/N : Fe/2),abs(Z));
xlabel('f');
ylabel('signal');
grid on;
legend('Z');
title('Représentation fréquentielle du signal total');
```

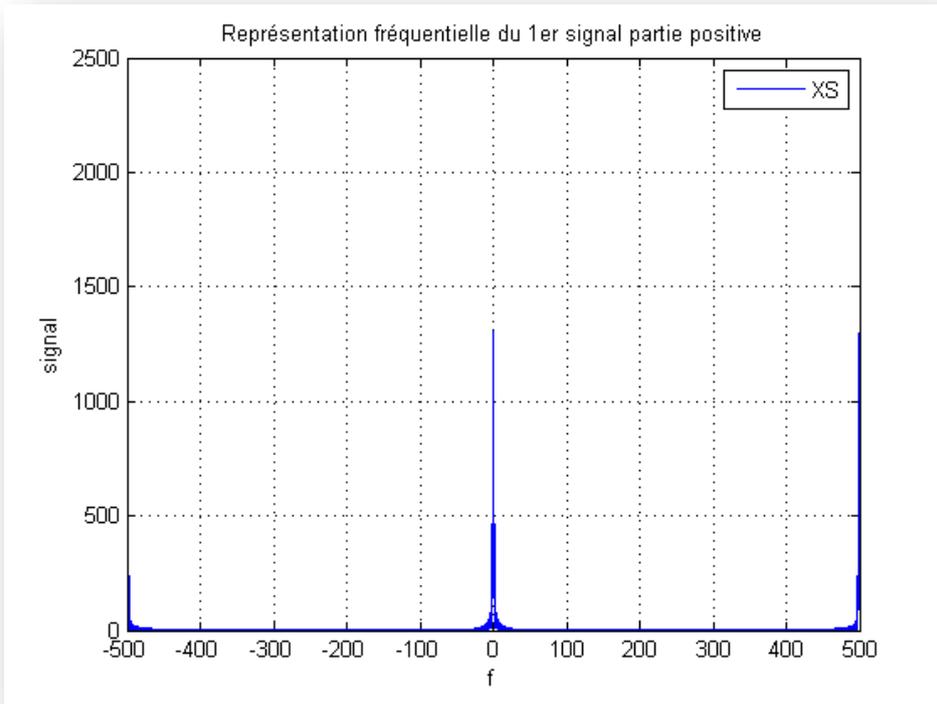
Démodulation

Cette partie du TP va permettre de voir si après filtrage, la démodulation permet de retrouver le signal d'origine.

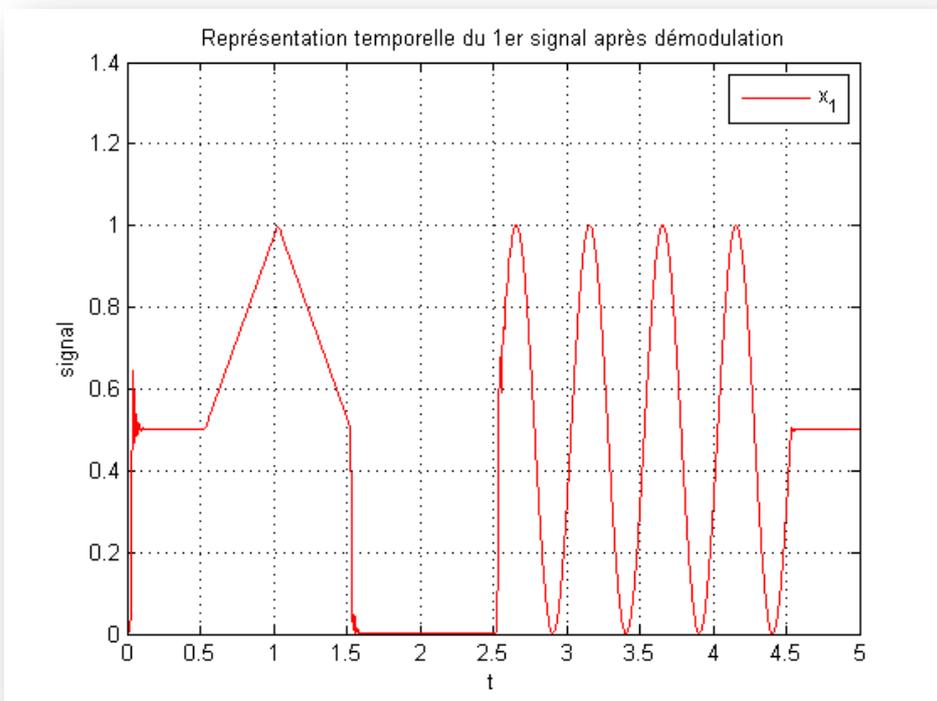
On commence donc par appliquer un filtre passe-bande au premier signal. En voici la représentation graphique dans le domaine fréquentiel :



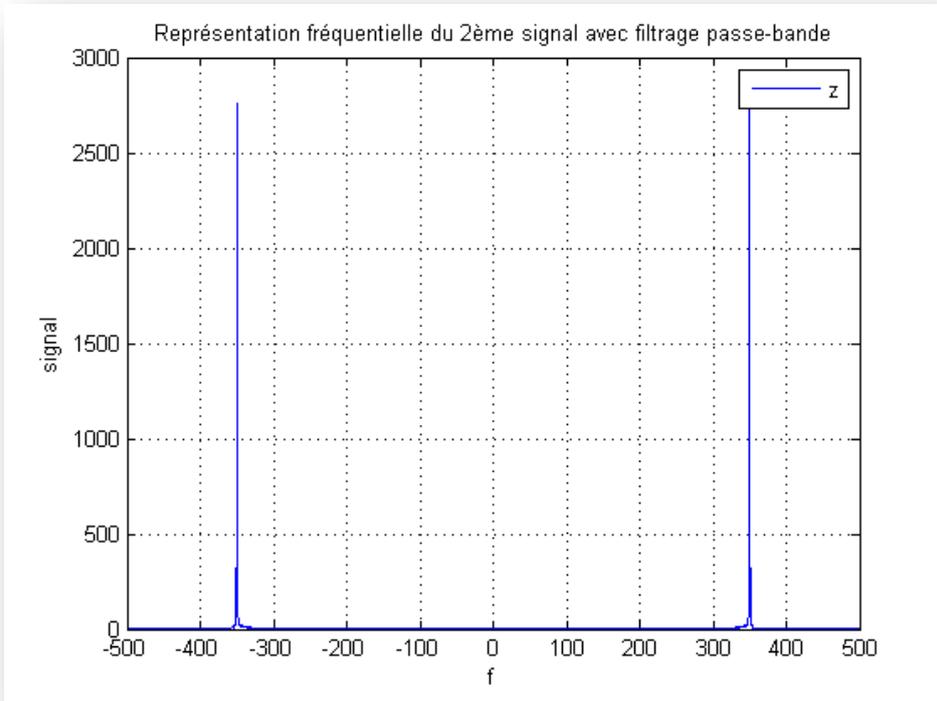
Après le filtrage passe-bande, on prend la partie positive du premier signal. Voici la représentation fréquentielle du premier signal :



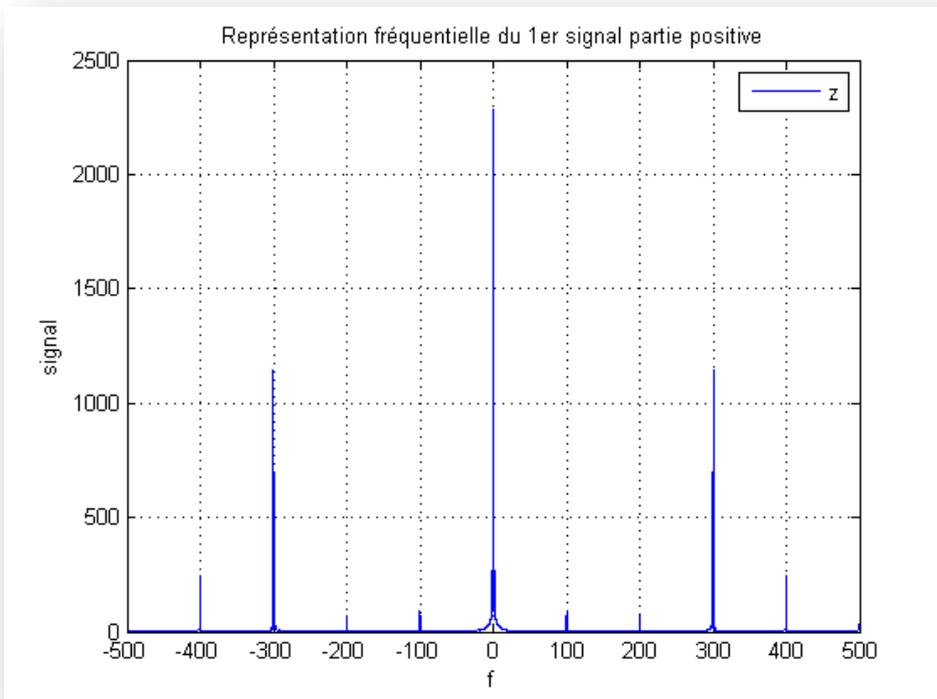
On repasse dans le domaine temporel pour essayer d'obtenir le premier signal d'origine. Voici le résultat :



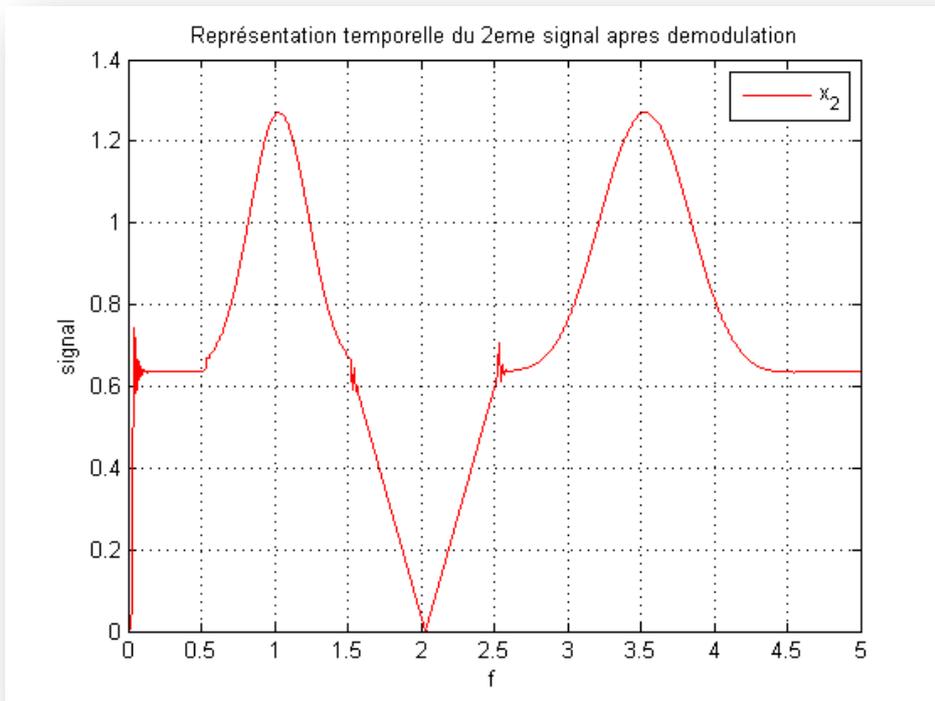
On fait de même avec le deuxième signal, on applique d'abord un filtre passe-bande dont en voici la représentation :



Puis on prend la partie positive :



Après filtrage et démodulation, on obtient quasiment le signal d'origine.



Code Matlab

```

zs=z;

% 1er signal

[vB,vA] = butter(8,[200 300]/(Fe/2)); %filtre passe-bande
ys = filter(vB,vA,zs);

k = 0;

for f = -Fe/2 : Fe/N : Fe/2
k = k+1;
YS(k) = ys' * exp(2*j*pi*f*t);
End

figure(10);
plot((-Fe/2 : Fe/N : Fe/2),abs(YS));
xlabel('f');
ylabel('signal');
grid on;
legend('YS');
title('Représentation fréquentielle du 1er signal avec filtrage passe-
bande');

xs=abs(ys); %partie positive

k = 0;

for f = -Fe/2 : Fe/N : Fe/2
k = k+1;

```

```

XS(k) = xs' * exp(2*j*pi*f*t);
End

figure(11);
plot((-Fe/2 : Fe/N : Fe/2),abs(XS));
xlabel('f');
ylabel('signal');
grid on;
legend('XS');
title('Représentation fréquentielle du 1er signal partie positive');

[vB,vA] = butter(8,50/(Fe/2)); %filtre passe-bas
ys = filter(vB,vA,xs);

figure(12);
plot(t,ys,'r');
xlabel('t');
ylabel('signal');
grid on;
legend('x_1');
title('Représentation temporelle du 1er signal après démodulation');

% On récupère le deuxième signal
[vB,vA] = butter(8,[300 400]/(Fe/2));
ys = filter(vB,vA,zs);

k = 0;

for f = -Fe/2 : Fe/N : Fe/2
k = k+1;
YS(k) = ys' * exp(2*j*pi*f*t);
End

figure(13);
plot((-Fe/2 : Fe/N : Fe/2),abs(YS));
xlabel('f');
ylabel('signal');
grid on;
legend('z');
title('Représentation fréquentielle du 2ème signal avec filtrage passe-
bande');

xs=abs(ys);

k = 0;

for f = -Fe/2 : Fe/N : Fe/2
k = k+1;
XS(k) = xs' * exp(2*j*pi*f*t);
End

figure(14);
plot((-Fe/2 : Fe/N : Fe/2),abs(XS));
xlabel('f');
ylabel('signal');
grid on;
legend('z');
title('Représentation fréquentielle du 1er signal partie positive');

[vB,vA] = butter(8,50/(Fe/2));

```

```
ys = filter(vB,vA,xs);  
  
figure(15);  
plot(t,ys,'r');  
xlabel('f');  
ylabel('signal');  
grid on;  
legend('x_2');  
title('Représentation temporelle du 2eme signal apres demodulation');
```

Conclusion

Influence du paramètre « a » :

On a pu remarquer au travers de nos différentes expériences que 'a', variant entre 0 et 1, influe beaucoup sur l'amplitude du signal que l'on obtient au final. Si 'a' se rapproche de 0, on constate que l'amplitude du signal est presque nul et on ne peut pas alors reconnaître la forme du signal. Au contraire, si 'a' tend vers 1, le signal obtenu est clair et son amplitude est maximale.

Récupération du signal :

Grace au filtrage passe-bande et à la démodulation, on parvient à récupérer le signal que nous avons créé au début de l'exercice, il n'est pas parfait mais le plus important pour la suite est qu'il respecte la forme d'origine.

Nous pouvons donc conclure que cette opération sur les signaux se révèle très utile. Et nous comprenons bien l'implication que ceci peut avoir à différents niveaux des systèmes de communication de la vie de tous les jours.