TP5

- VIN Charles
- RECHAK LAMBERT Dorian

Question 1

On doit utiliser la fenêtre rectagulaire car son attenuation minimal vaut $A_smin = 21db$ et on cherche une attenuation de 16db.

Question 2

$\Omega_s - \Omega_p = \frac{1.8\pi}{M}$

$\Leftrightarrow M = \frac{1.8\pi}{\Omega_s - \Omega_p}$

$\Leftrightarrow M = \frac{1.8\pi}{0.3\pi - 0.2\pi}$

$\Leftrightarrow M = \frac{1.8\pi}{0.1\pi}$

$\Leftrightarrow M = \frac{1.8\pi}{0.1\pi}$

$\Leftrightarrow M = 18$

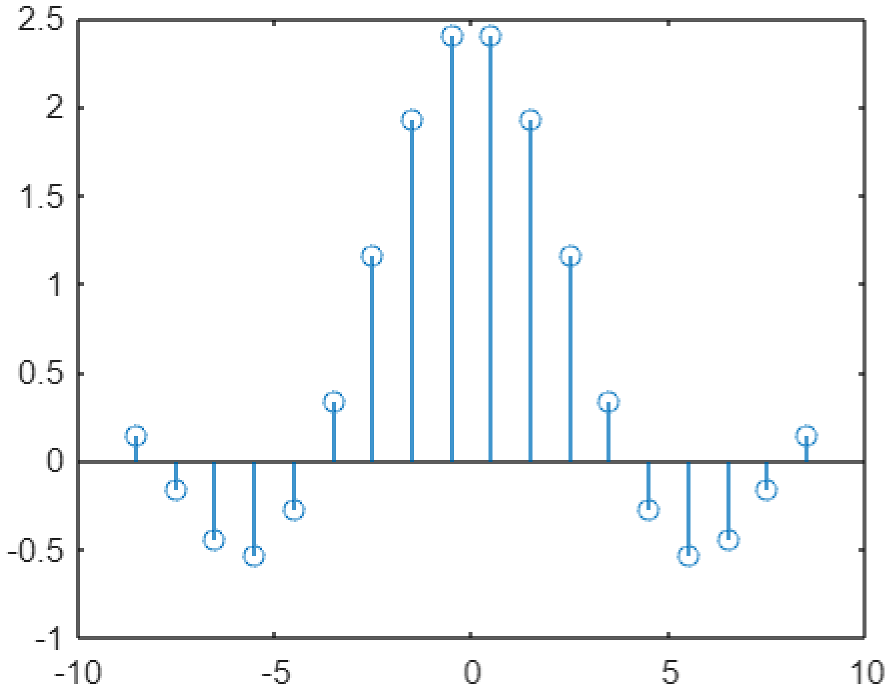
L'ordre du filtre est $M - 1 = 17$. On a M coeficients, c'est à dire 18 zéros

```
M = 18;
```

Question 3

On n'a pas besoin de multiplier par rect en définisant n avec uniquement les points de la fenêtre

```
Omega_p = 0.2*pi;
Omega_s = 0.3*pi;
Omega_c = (Omega_p + Omega_s)/2;
n = -(M-1)/2 : 1 : (M-1)/2;
h = sin(n.*Omega_c)./n.*pi;
stem(n, h);
```



Question 4

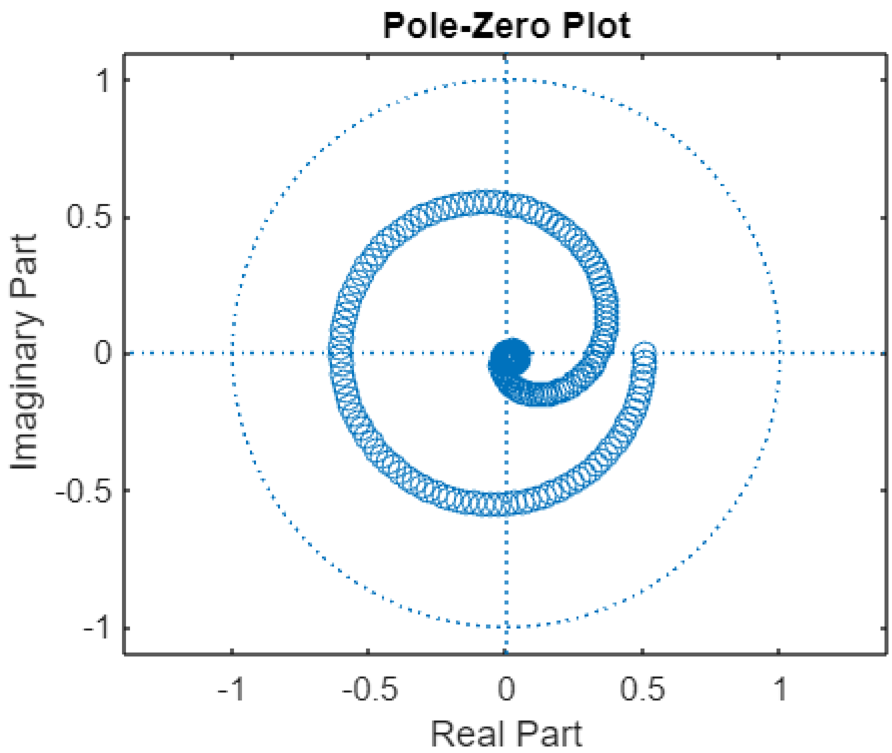
```
% f = 0 : 0.01 : 3;
% w = 2*pi*f;
% Z = exp(1i*w);
% H = h(1) + h(2).*Z.^(-1) + h(3).*Z.^(-2) + h(4).*Z.^(-3) + h(5).*Z.^(-4) + h(6).*Z.^(-5) + h(
```

Autre méthode :

```
[H,w] = freqz(h, M);
```

Question Bonus

```
zplane(H);
```



Assez joli, tous à l'intérieur du cercle donc stable

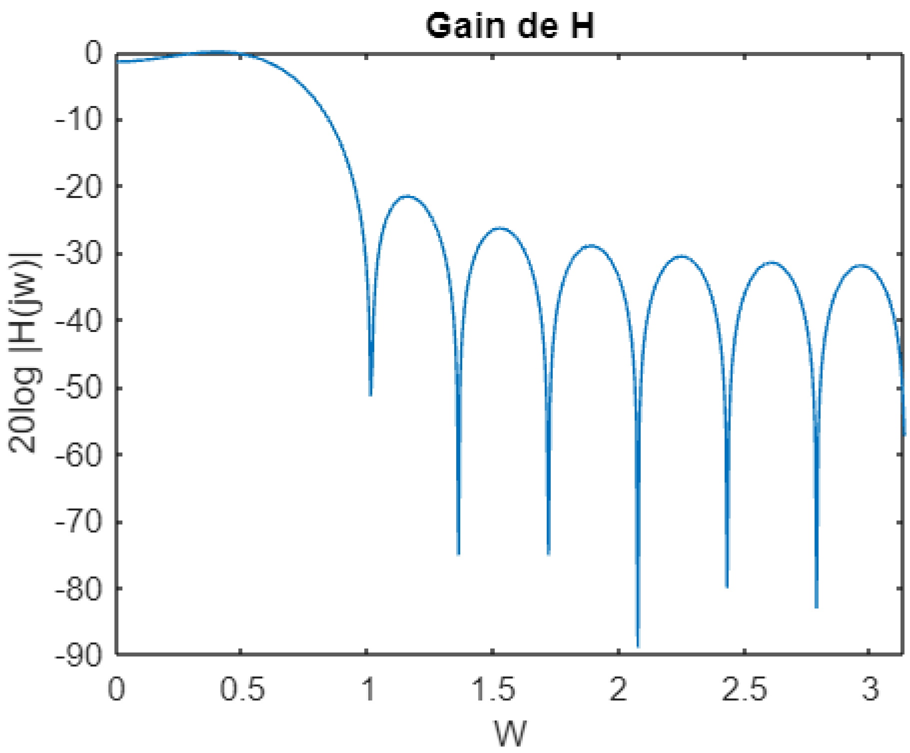
Plot du module et de la phase

```
H = H/max(H); % Normalisation de H
```

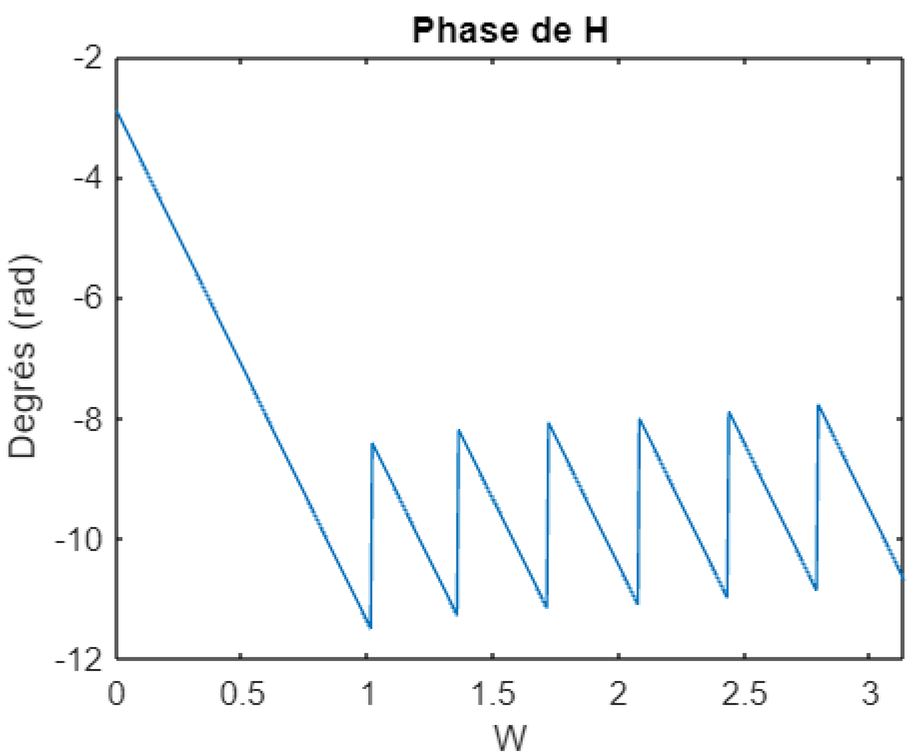
```
H = 512×1 complex
  -0.8205 - 0.2082i
  -0.8303 - 0.1652i
  -0.8380 - 0.1217i
  -0.8436 - 0.0779i
  -0.8470 - 0.0338i
  -0.8484 + 0.0104i
  -0.8475 + 0.0547i
  -0.8445 + 0.0989i
  -0.8394 + 0.1430i
  -0.8320 + 0.1868i
      ⋮
```

```
module = 20*log10(abs(H));
phase = unwrap(angle(H));

figure('name', 'Gain de H');
plot(w, module);
title('Gain de H');
xlabel('W');
ylabel('20log |H(jw)|');
```




```
figure('name', 'Phase de H');
plot(w, phase);
title('Phase de H');
xlabel('W');
ylabel('Degrés (rad)');
```



Vérifions les caractéristiques du filtre :

Pour cela je vais regarder le du module à 0.2π et 0.3π. Ils doivent être borné par $R_p = 7\text{db}$ et $A_s = 16\text{db}$

```
disp("=======0.2pi======");
```

```
=======0.2pi======
```

```
w(103)
```

```
ans = 0.6259
```

```
disp("L'indice 103 de W represente environ 0.2pi = 0.6283185307");
```

```
L'indice 103 de W represente environ 0.2pi = 0.6283185307
```

```
module(103)
```

```
ans = -1.9072
```

```
disp("On a le module de 0.2pi qui vaut environ -1.9072");
```

```
On a le module de 0.2pi qui vaut environ -1.9072
```

```

disp("");
disp("=======0.3pi======");
```

```
=======0.3pi======
```

```
w(155)
```

```
ans = 0.9449
```

```
disp("L'indice 155 de W represente environ 0.3pi = 0.9424777961");
```

```
L'indice 155 de W represente environ 0.3pi = 0.9424777961
```

```
module(155)
```

```
ans = -19.6104
```

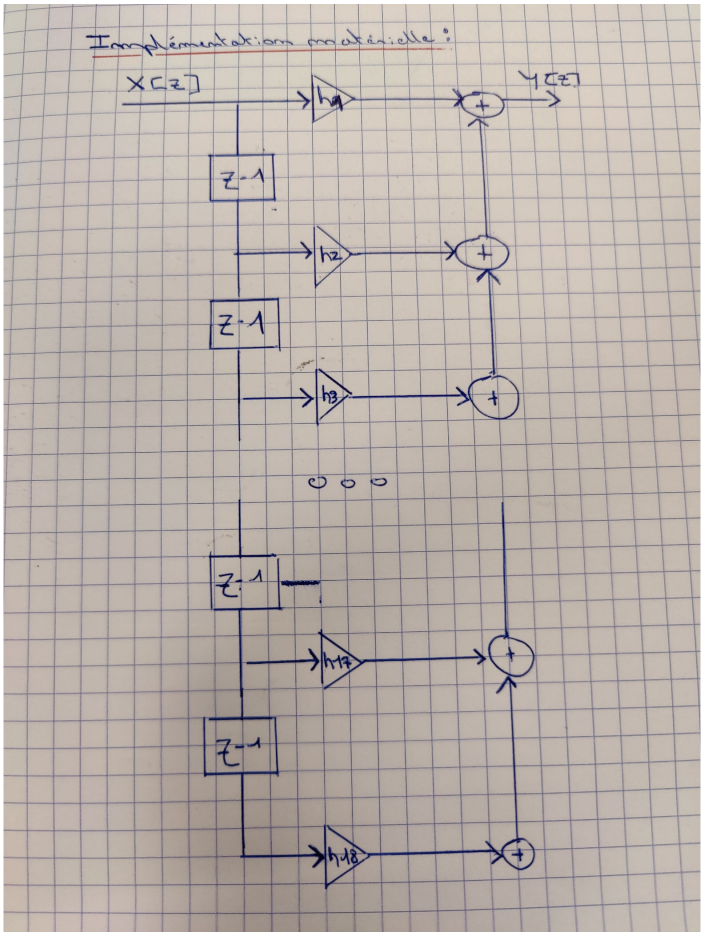
```
disp("On a le module de 0.3pi qui vaut environ -19.6104");
```

```
On a le module de 0.3pi qui vaut environ -19.6104
```

On peut voir que les caractéristiques du filtre sont respecté. Le module de 0.3π vaut $-1.9072 < R_p$. Et le module de 0.3π vaut $-19.6104 < A_s$, ainsi on filtre mieux.

Question 5

Version simple :



Version optimisé :

$$
\begin{aligned}
H(z) = {} & h(1) + h(2)\times z^{(-1)} + h(3)\times z^{(-2)} + h(4)\times z^{(-3)} + h(5)\times z^{(-4)} \\
& + h(6)\times z^{(-5)} + h(7)\times z^{(-6)} + h(8)\times z^{(-7)} + h(9)\times z^{(-8)} + h(10)\times z^{(-9)} \\
& + h(11)\times z^{(-10)} + h(12)\times z^{(-11)} + h(13)\times z^{(-12)} + h(14)\times z^{(-13)} + h(15)\times z^{(-14)} \\
& + h(16)\times z^{(-15)} + h(17)\times z^{(-16)} + h(18)\times z^{(-17)}
\end{aligned}
$$

Le filtre est symétrique :

donc $h_1 = h_{18}$

$h_2 = h_{17}$

$h_3 = h_{16}$

$h_4 = h_{15}$

$h_5 = h_{14}$

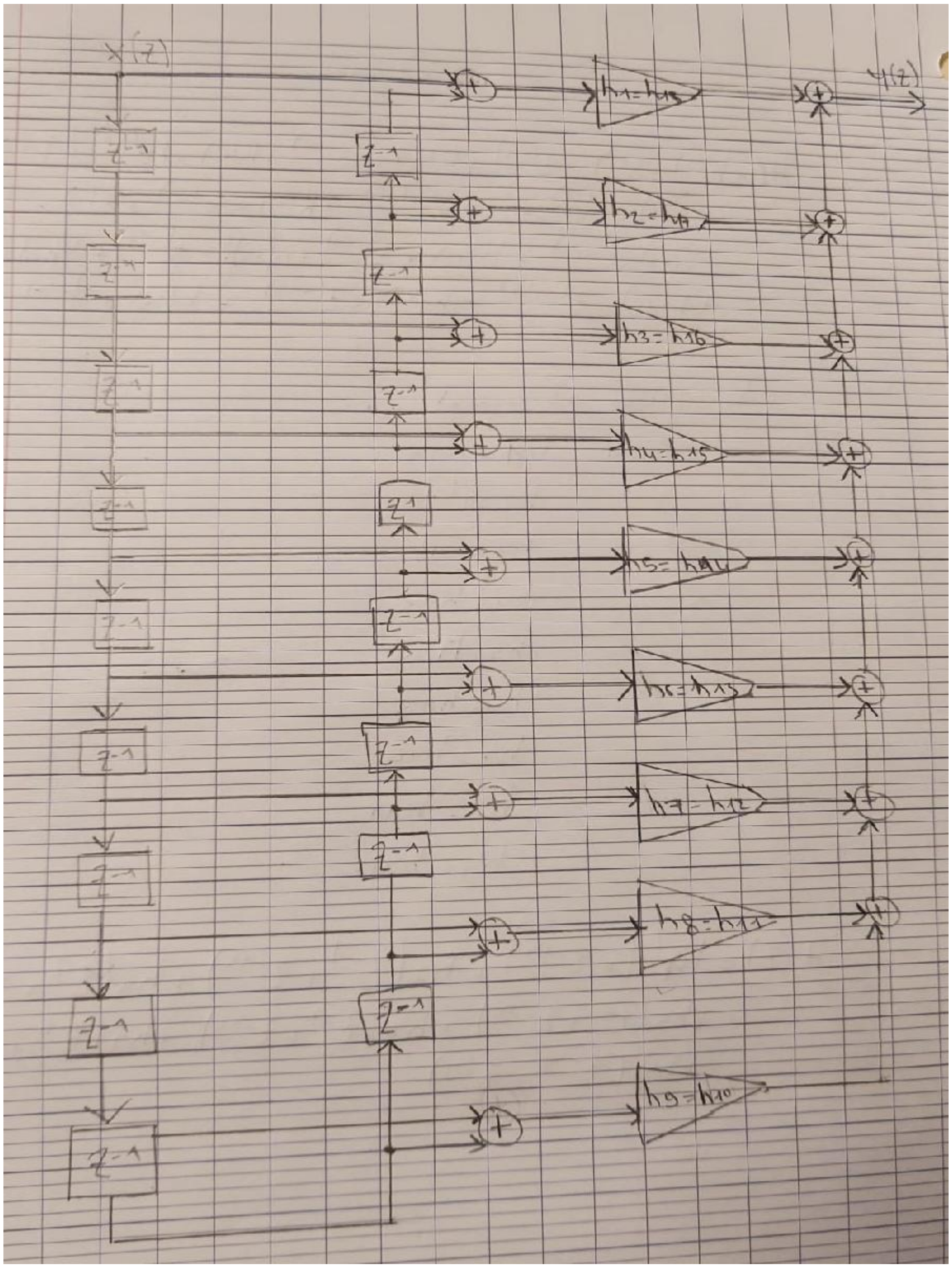
$h_6 = h_{13}$

$h_7 = h_{12}$

$h_8 = h_{11}$

$h_9 = h_{10}$

$$
\begin{aligned}
H(z) = {} & h_1(1 + z^{-17}) + h_2(z^{-1} + z^{-16}) + h_3(z^{-2} + z^{-15}) + h_4(z^{-3} + z^{-14}) \\
& + h_5(z^{-4} + z^{-13}) + h_6(z^{-5} + z^{-12}) + h_7(z^{-6} + z^{-11}) + h_8(z^{-7} + z^{-10}) \\
& + h_9(z^{-8} + z^{-9})
\end{aligned}
$$



Question 6

Ce filtre est plus couteux à implémenter en compaison avec celui du TP précédent. Il néanmoins plus performant et nous évite les problèmes de stabilité des pôles.

Question 7 : Bartlett

```
M = 6.1 / 0.1 + 1;
M
```

```
M = 62.0000
```

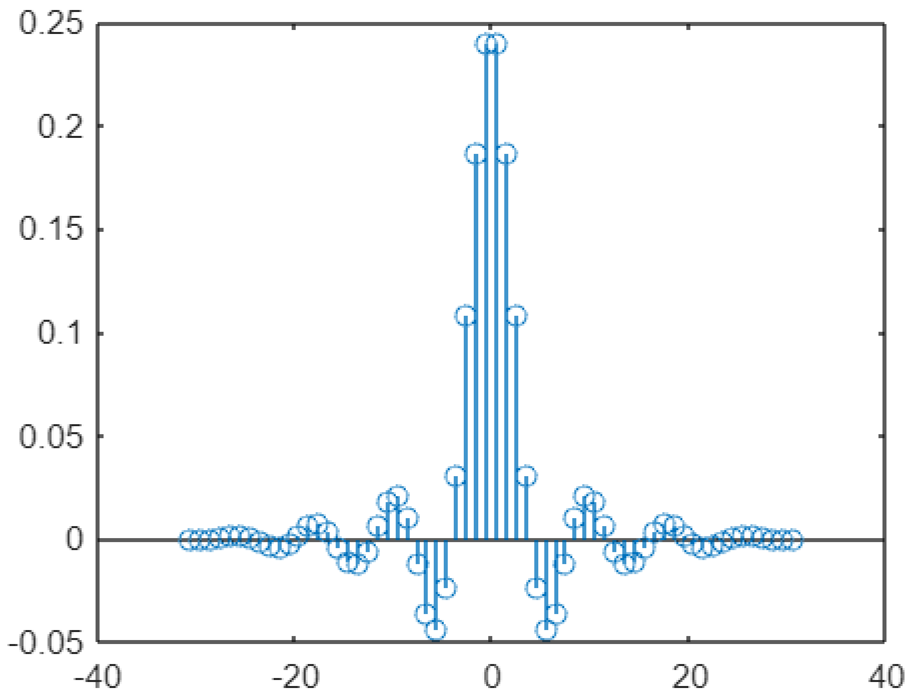
Comme M est impaire, on le rend paire pour éviter les problèmes de division par zéros dans h_d

```
n = -(M-1)/2 : 1 : (M-1)/2;
```

```
h_d = sin(n.*Omega_c)./(n.*pi+eps);
bw = bartlett(M);
```

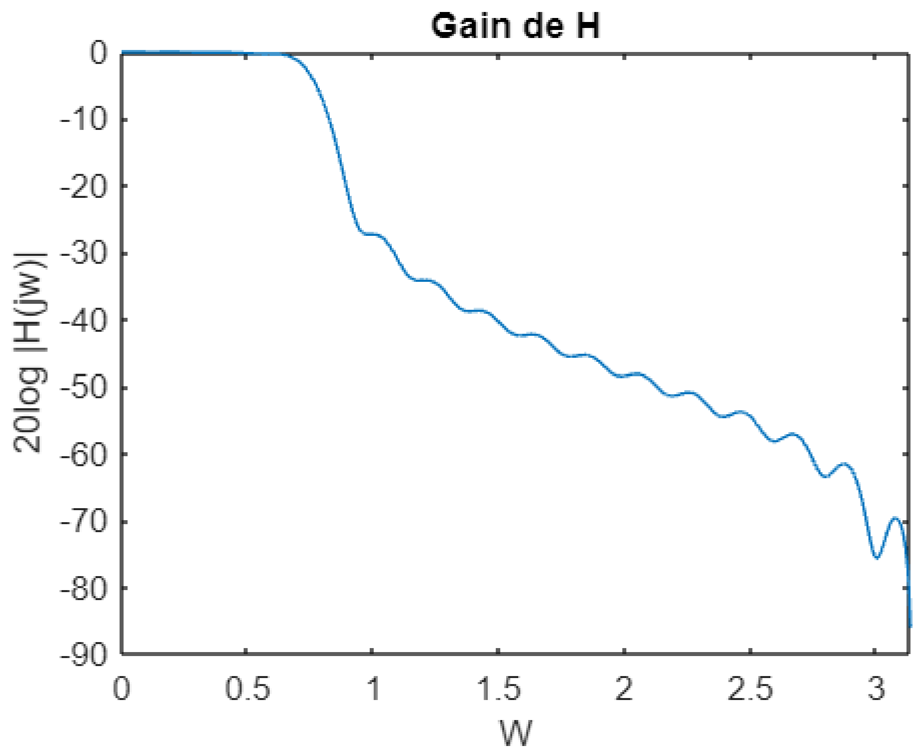
```
Warning: Rounding order to nearest integer.
```

```
h = h_d .* transpose(bw);
stem(n, h);
```

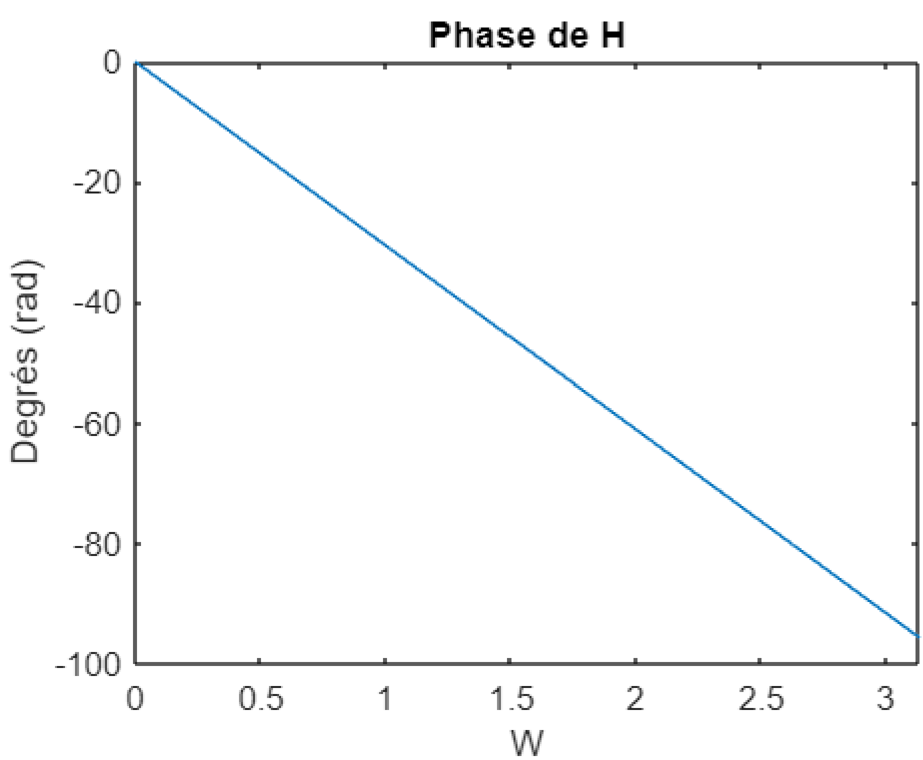



```
[H,w] = freqz(h, 1024);
H = H/max(H);
module = 20*log10(abs(H));
phase = unwrap(angle(H));

figure('name', 'Gain de H');
plot(w, module);
title('Gain de H');
xlabel('W');
ylabel('20log |H(jw)|');
```




```
figure('name', 'Phase de H');
plot(w, phase);
title('Phase de H');
xlabel('W');
ylabel('Degrés (rad)');
```



Vérifions les caractéristiques du filtre :

Pour cela je vais regarder le du module à 0.2π et 0.3π. Ils doivent être borné par $R_p = 7\text{db}$ et $A_s = 16\text{db}$

```
disp("=======0.2pi======");
```

```
=======0.2pi======
```

```
w(103)
```

```
ans = 0.6259
```

```
disp("L'indice 103 de W represente environ 0.2pi = 0.6283185307");
```

```
L'indice 103 de W represente environ 0.2pi = 0.6283185307
```

```
module(103)
```

```
ans = -0.3213
```

```
disp("On a le module de 0.2pi qui vaut environ -0.3213");
```

```
On a le module de 0.2pi qui vaut environ -0.3213
```

```

disp("");
disp("=======0.3pi======");
```

```
=======0.3pi======
```

```
w(155)
```

```
ans = 0.9449
```

```
disp("L'indice 155 de W represente environ 0.3pi = 0.9424777961");
```

```
L'indice 155 de W represente environ 0.3pi = 0.9424777961
```

```
module(155)
```

```
ans = -26.3808
```

```
disp("On a le module de 0.3pi qui vaut environ -26.3808");
```

```
On a le module de 0.3pi qui vaut environ -26.3808
```

On peut voir que les caractéristiques du filtre sont respecté. Le module de 0.3π vaut $-0.3213 < R_p$. Et le module de 0.3π vaut $-26.3808 < A_s$, ainsi on filtre mieux.

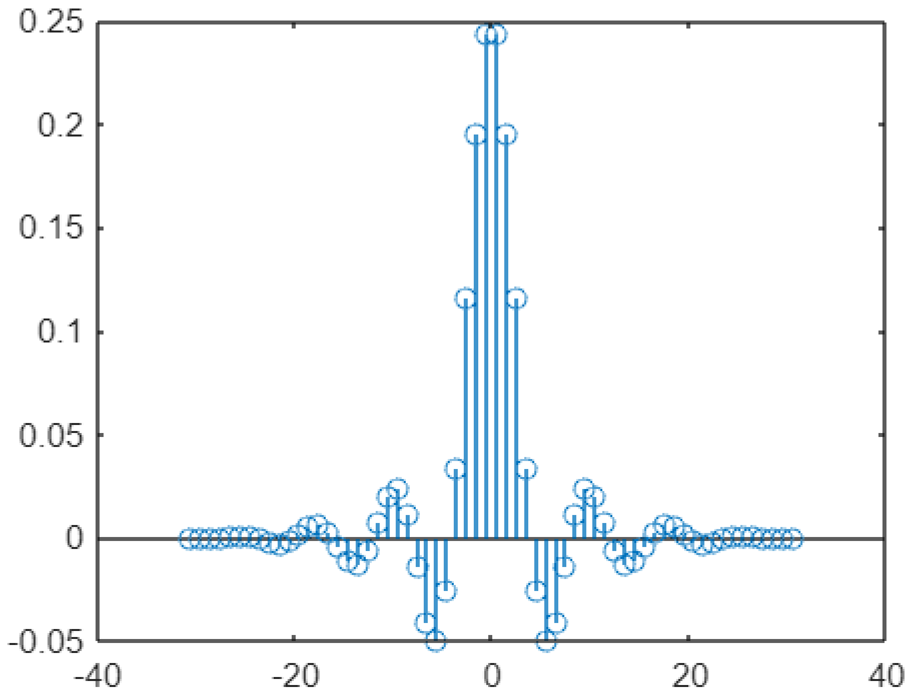
Question 8 : Hanning

```
M = 6.2 / 0.1 ;
M
```

```
M = 62
```

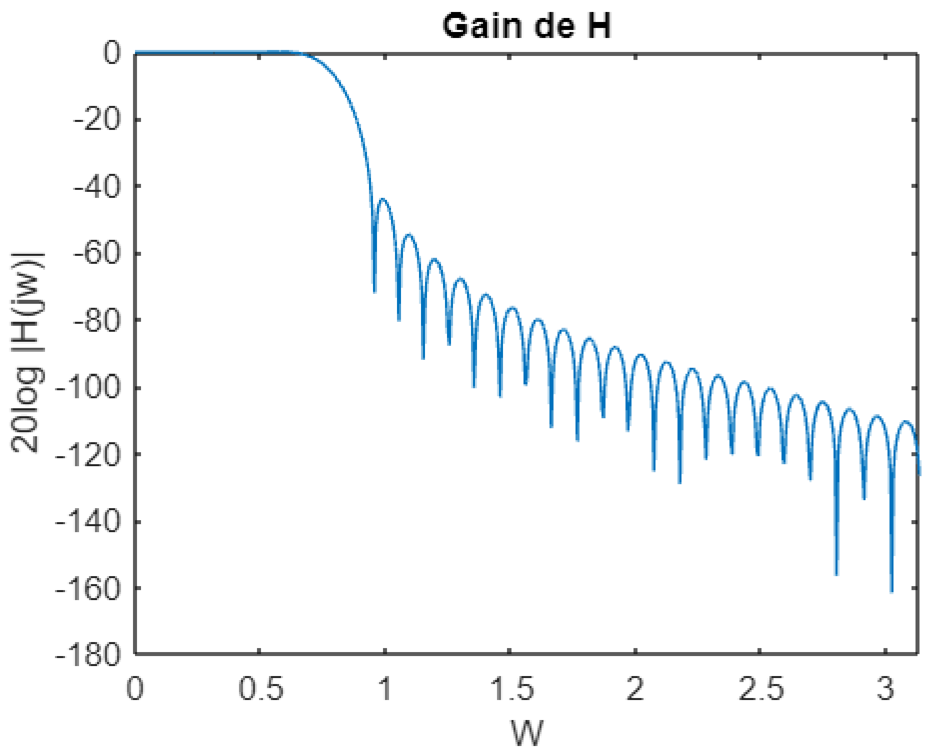
M est paire, pas de problème avec la division par 0 dans h_d.

```
n = -(M-1)/2 : 1 : (M-1)/2;
h_d = sin(n.*Omega_c)./(n.*pi+eps);
bw = hann(M);
h = h_d .* transpose(bw);
stem(n, h);
```

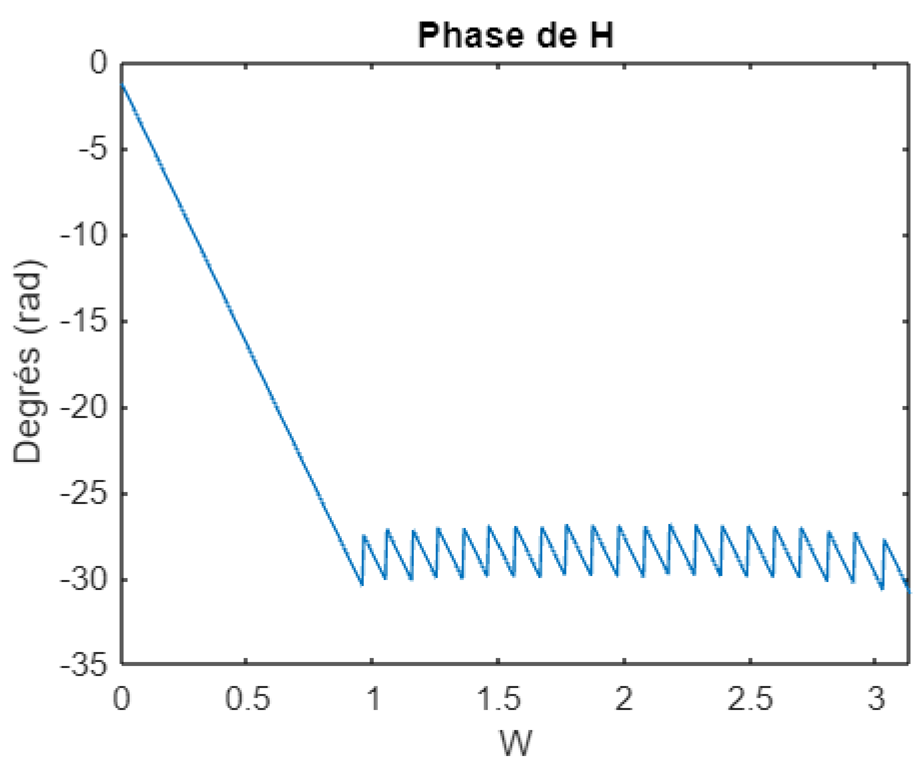



```
[H,w] = freqz(h, 1024);
H = H/max(H);
module = 20*log10(abs(H));
phase = unwrap(angle(H));

figure('name', 'Gain de H');
plot(w, module);
title('Gain de H');
xlabel('W');
ylabel('20log |H(jw)|');
```




```
figure('name', 'Phase de H');
plot(w, phase);
title('Phase de H');
xlabel('W');
ylabel('Degrés (rad)');
```



Vérifions les caractéristiques du filtre :

Pour cela je vais regarder le du module à 0.2π et 0.3π. Ils doivent être borné par $R_p = 7\text{db}$ et $A_s = 16\text{db}$

```
disp("=======0.2pi======");
```

```
=======0.2pi======
```

```
w(103)
```

```
ans = 0.6259
```

```
disp("L'indice 103 de W represente environ 0.2pi = 0.6283185307");
```

```
L'indice 103 de W represente environ 0.2pi = 0.6283185307
```

```
module(103)
```

```
ans = -0.1181
```

```
disp("On a le module de 0.2pi qui vaut environ -0.1131");
```

```
On a le module de 0.2pi qui vaut environ -0.1131
```

```

disp("");
disp("=======0.3pi======");
```

```
=======0.3pi======
```

```
w(155)
```

```
ans = 0.9449
```

```
disp("L'indice 155 de W represente environ 0.3pi = 0.9424777961");
```

```
L'indice 155 de W represente environ 0.3pi = 0.9424777961
```

```
module(155)
```

```
ans = -42.8125
```

```
disp("On a le module de 0.3pi qui vaut environ -42.8125");
```

```
On a le module de 0.3pi qui vaut environ -42.8125
```

On peut voir que les caractéristiques du filtre sont respecté. Le module de 0.3π vaut $-0.1131 < R_p$. Et le module de 0.3π vaut $-42.81 < A_s$, ainsi on filtre mieux.

Question 9 : Hamming

```
M = 6.6 / 0.1 ;
M
```

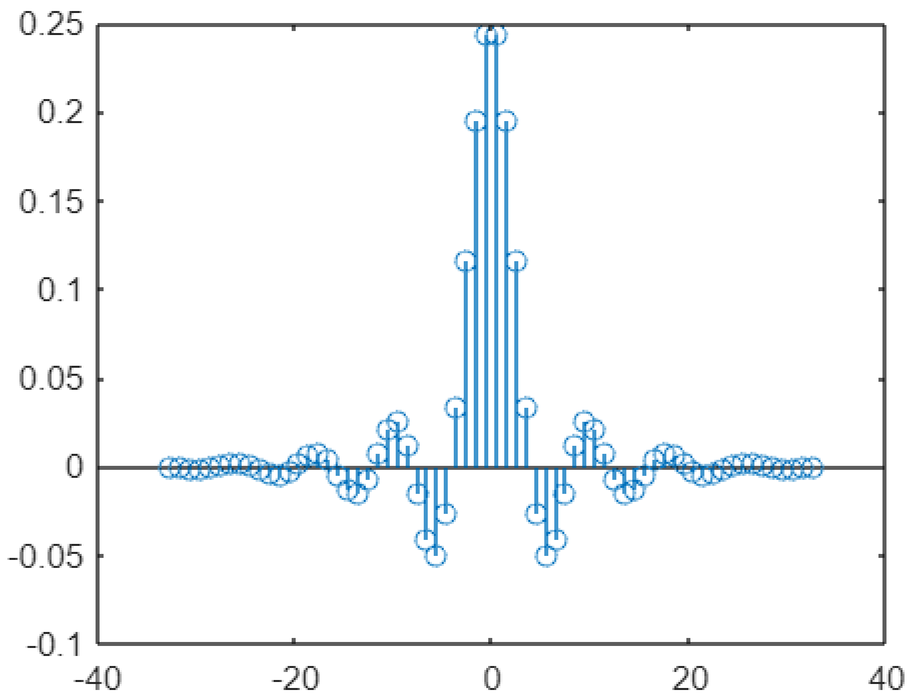
```
M = 66.0000
```

M est paire, pas de problème avec la division par 0 dans h_d.

```
n = -(M-1)/2 : 1 : (M-1)/2;
h_d = sin(n.*Omega_c)./(n.*pi+eps);
bw = hamming(M);
```

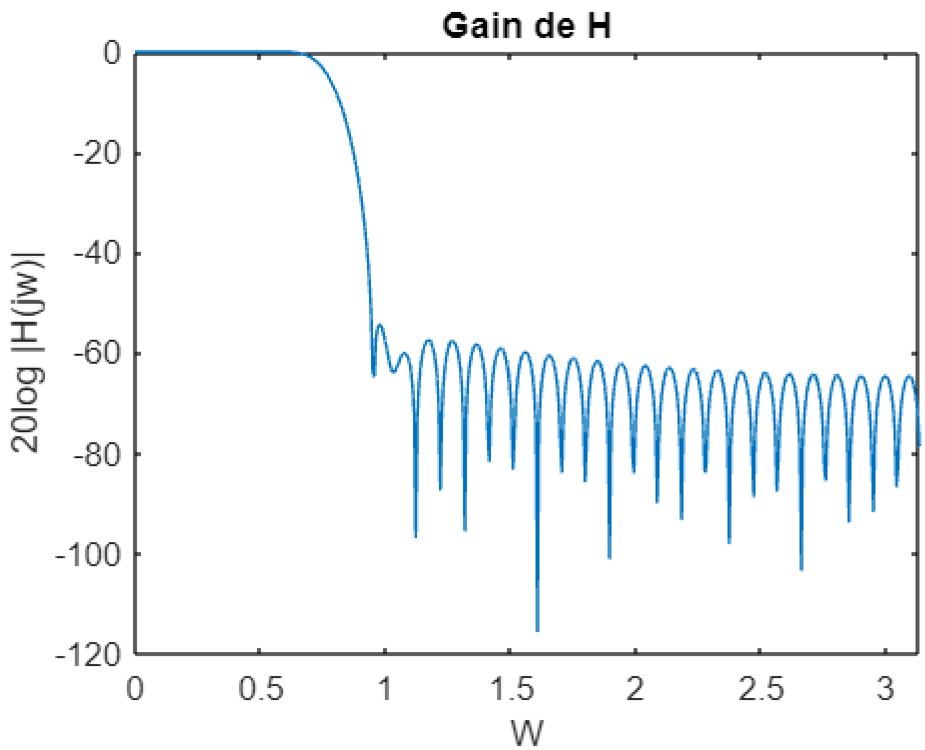
```
Warning: Rounding order to nearest integer.
```

```
h = h_d .* transpose(bw);
stem(n, h);
```

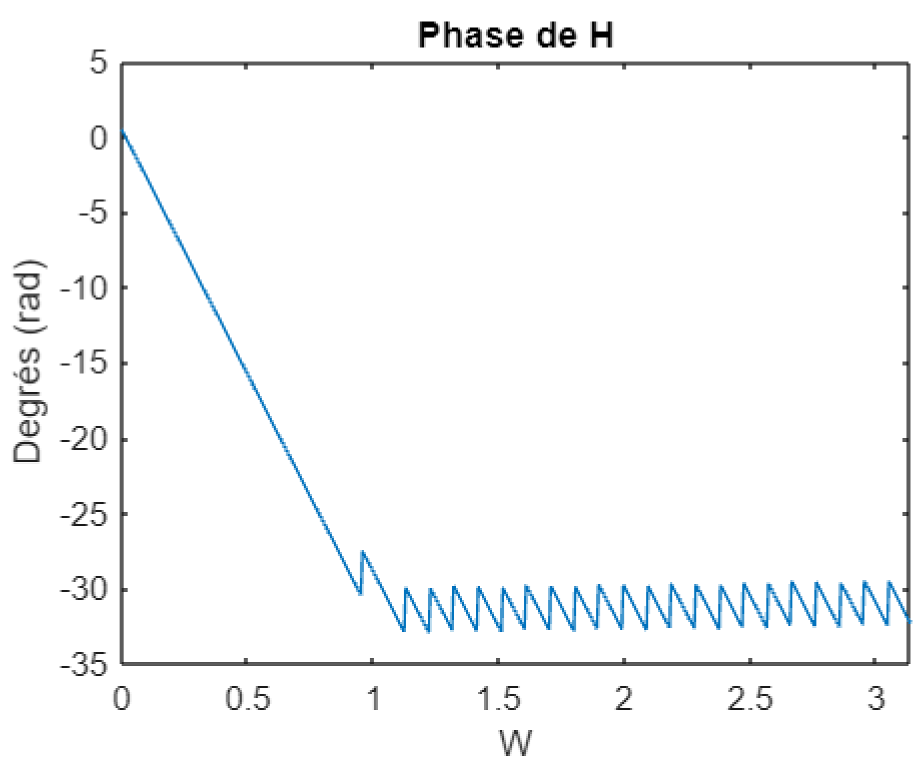



```
[H,w] = freqz(h, 1024);
H = H/max(H);
module = 20*log10(abs(H));
phase = unwrap(angle(H));

figure('name', 'Gain de H');
plot(w, module);
title('Gain de H');
xlabel('W');
ylabel('20log |H(jw)|');
```




```
figure('name', 'Phase de H');
plot(w, phase);
title('Phase de H');
xlabel('W');
ylabel('Degrés (rad)');
```



Vérifions les caractéristiques du filtre :

Pour cela je vais regarder le du module à 0.2π et 0.3π. Ils doivent être borné par$R_p = 7\text{db}$ et $A_s = 16\text{db}$

```
disp("=======0.2pi======");
```

```
=======0.2pi======
```

```
w(103)
```

```
ans = 0.6259
```

```
disp("L'indice 103 de W represente environ 0.2pi = 0.6283185307");
```

```
L'indice 103 de W represente environ 0.2pi = 0.6283185307
```

```
module(103)
```

```
ans = -0.0362
```

```
disp("On a le module de 0.2pi qui vaut environ -0.0362");
```

```
On a le module de 0.2pi qui vaut environ -0.0362
```

```

disp("");
disp("=======0.3pi======");
```

```
=======0.3pi======
```

```
w(155)
```

```
ans = 0.9449
```

```
disp("L'indice 155 de W represente environ 0.3pi = 0.9424777961");
```

```
L'indice 155 de W represente environ 0.3pi = 0.9424777961
```

```
module(155)
```

```
ans = -52.7115
```

```
disp("On a le module de 0.3pi qui vaut environ -52.7115");
```

```
On a le module de 0.3pi qui vaut environ -52.7115
```

On peut voir que les caractéristiques du filtre sont respecté. Le module de 0.3π vaut $-0.0362 < R_p$. Et le module de 0.3π vaut $-52.7115 < A_s$, ainsi on filtre mieux.

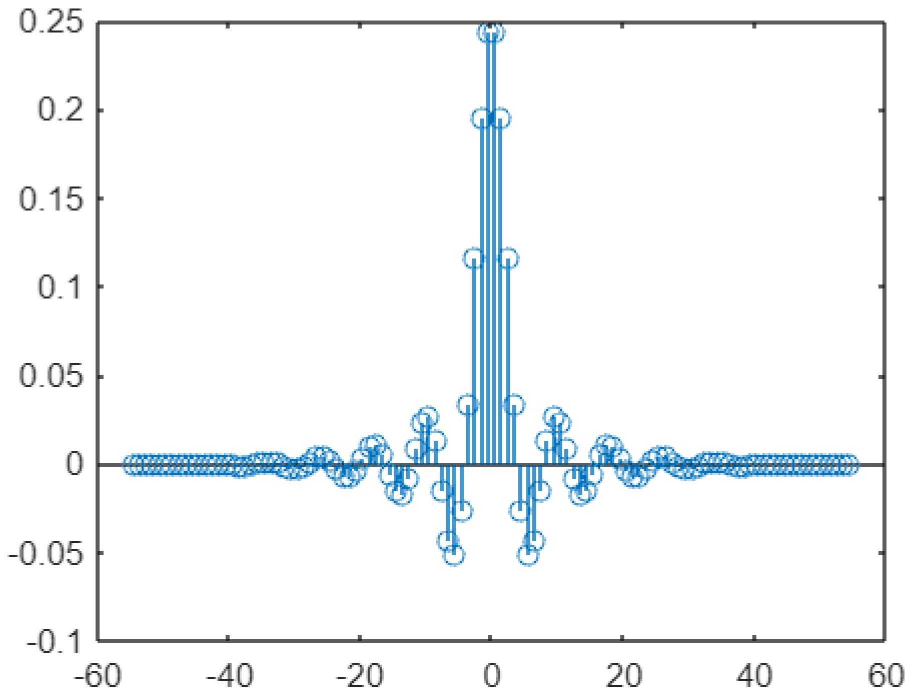
Question 9 : Blackman

```
M = 11 / 0.1 ;
M
```

```
M = 110
```

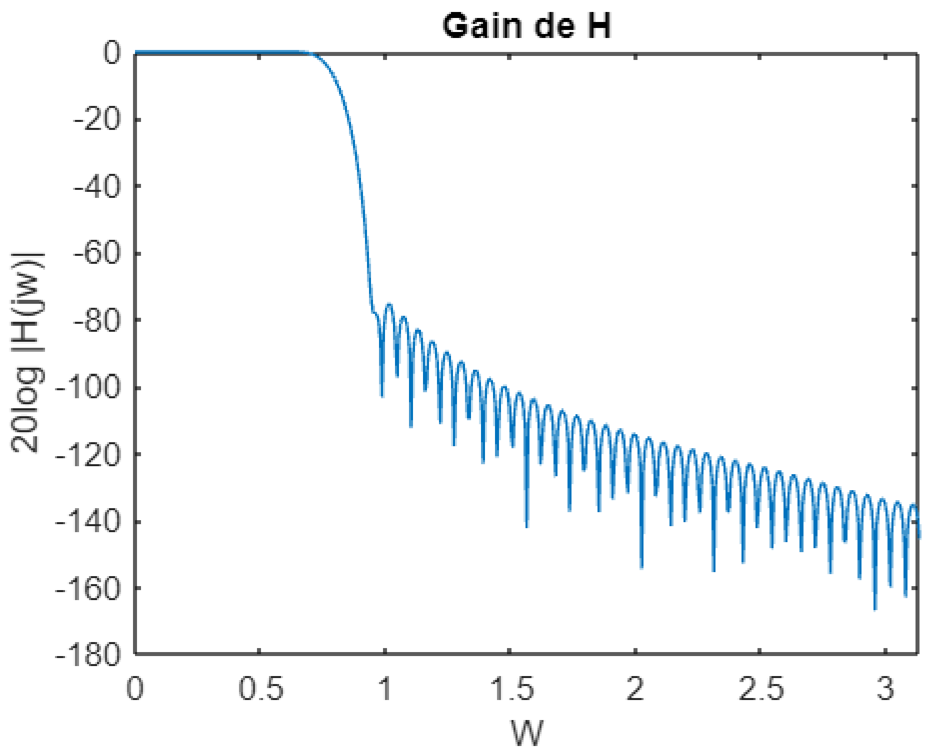
M est paire, pas de problème avec la division par 0 dans h_d.

```
n = -(M-1)/2 : 1 : (M-1)/2;
h_d = sin(n.*Omega_c)./(n.*pi+eps);
bw = blackman(M);
h = h_d .* transpose(bw);
stem(n, h);
```

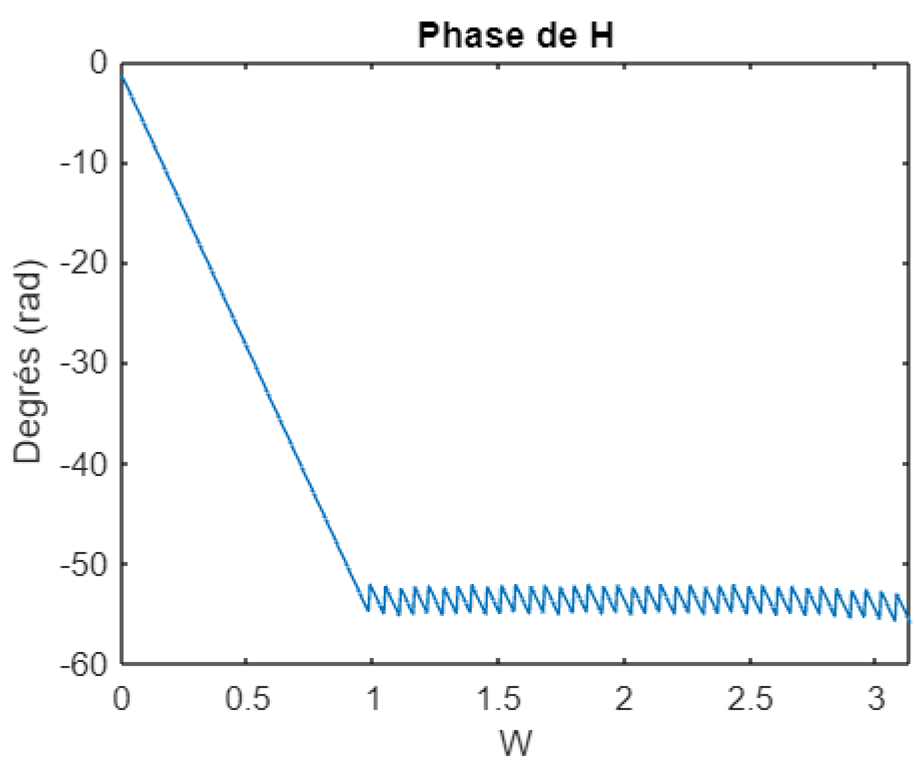



```
[H,w] = freqz(h, 1024);
H = H/max(H);
module = 20*log10(abs(H));
phase = unwrap(angle(H));

figure('name', 'Gain de H');
plot(w, module);
title('Gain de H');
xlabel('W');
ylabel('20log |H(jw)|');
```




```
figure('name', 'Phase de H');
plot(w, phase);
title('Phase de H');
xlabel('W');
ylabel('Degrés (rad)');
```



Vérifions les caractéristiques du filtre :

Pour cela je vais regarder le du module à 0.2π et 0.3π. Ils doivent être borné par $R_p = 7\text{db}$ et $A_s = 16\text{db}$

```
disp("=======0.2pi======");
```

```
=======0.2pi======
```

```
w(103)
```

```
ans = 0.6259
```

```
disp("L'indice 103 de W represente environ 0.2pi = 0.6283185307");
```

```
L'indice 103 de W represente environ 0.2pi = 0.6283185307
```

```
module(103)
```

```
ans = -0.0032
```

```
disp("On a le module de 0.2pi qui vaut environ -0.0032");
```

```
On a le module de 0.2pi qui vaut environ -0.0362
```

```

disp("");
disp("=======0.3pi======");
```

```
=======0.3pi======
```

```
w(155)
```

```
ans = 0.9449
```

```
disp("L'indice 155 de W represente environ 0.3pi = 0.9424777961");
```

```
L'indice 155 de W represente environ 0.3pi = 0.9424777961
```

```
module(155)
```

```
ans = -74.2304
```

```
disp("On a le module de 0.3pi qui vaut environ -74.2304");
```

```
On a le module de 0.3pi qui vaut environ -52.7115
```

On peut voir que les caractéristiques du filtre sont respecté. Le module de 0.3π vaut $-0.0032 < R_p$. Et le module de 0.3π vaut $-74.2304 < A_s$, ainsi on filtre mieux.

Conclusion sur ces dernières questions

On peut voir qu'en ajoutant des paramètres , en augmentant l'ordre du filtre, les performances deviennent de mieux en mieux. Mais celà à néanmoins un cout matériel certains. Que cela soit pour le nombre de composant du circuit, ou numériquement en temps de calcul où on le sens déjà sur nos ordinateurs avec matlab.