
Teoria de Senyal:

Pràctica 1 de laboratori:

Table of Contents

Damià Casas & Pau Manyer	1
Estudi Previ:	1
Activitats al laboratori:	2

Damià Casas & Pau Manyer

Estudi Previ:

Qüestió 1.3:

$$y[n+2] = x[n+2] - \frac{2}{3}x[n+1] + \frac{7}{8}y[n+1] - \frac{3}{32}y[n]$$

L'equació en diferències finites ens demana que el vector y contingui almenys $y[0]$ i $y[1]$ per poder inicialitzar la recurrència.

```
a=[1 ; -7/8 ; 3/32];  
b=[1; -2/3];  
N=length(x);
```

```
%  
% function y=iir_fd1(a,b,x,N)  
% y(1)=b(1)*x(1);  
% y(2)=b(1)*x(2)+b(2)*x(1)-a(2)*y(1);  
% for k=1:N-2  
%     compox=b(1)*x(k+2)+b(2)*x(k+1);  
%     compoy=-a(2)*y(k+1)-a(3)*y(k);  
%     y(k+2)=(1/(a(1)))*(compox+compoy);  
% end  
% end
```

```
% function y=iir_fd1(a,b,x,N)  
% y=[0 0];  
% for k=3:N  
%     compox=b(1)*x(k)+b(2)*x(k-1);  
%     compoy=-a(2)*y(k-1)-a(3)*y(k-2);  
%     y(k)=(1/(a(1)))*(compox+compoy);  
% end  
% end  
%  
% function y=iir_fd2(a,b,x,N)
```

```
% y(1)=b(1)*x(1);
% y(2)=b(1)*x(2)+b(2)*x(1)-a(2)*y(1);
% for k=1:N-2
%     y(k+2)=(1/(a(1)))*(b(1)*x(k+2)+b(2)*x(k+1)-a(2)*y(k+1)-
a(3)*y(k));
% end
% end

% function y=iir_fd2(a,b,x,N)
% y=[0 0];
% for k=3:N
%     y(k)=(1/(a(1)))*(b(1)*x(k)+b(2)*x(k-1)-a(2)*y(k-1)-a(3)*y(k-2));
% end
% end
```

Unrecognized function or variable 'x'.

Error in practica_1PAU (line 15)
N=length(x);

Activitats al laboratori:

Activitat 3.1:

Generem el senyal $x[n]$ tal que

$$x[n] = \delta[n] - 2\delta[n-1] + 4\delta[n-2]$$

```
N=40;
for k=0:N-1
    x(k+1)=delta(k)-2*delta(k-1)+4*delta(k-2);
end
```

Ara representem el senyal $x[n]$ després de passar-lo per les diferents funcions implementades *iir_df1* i *iir_fd2*.

El senyal resultant per *iir_df1* és el següent:

```
a=[1 -7/8 3/32];
b=[1 -2/3];
N=length(x);

y1=iir_fd1(a,b,x,N);

figure(1)
stem(y1, '-o')
grid on
title('iir_df1(x[n])');
```

De la mateixa manera, representem el senyal resultant per *iir_df1*:

```
y2=iir_fd2(a,b,x,N);
```

```
figure(2)
stem(y2, '-o')
grid on
title('iir\_df2(x[n])');
```

Activitat 3.2:

El objectiu ara és el d'obtenir la resposta impulsional del filtre que hem implementat en les funcions *iir_df1* i *iir_fd2*. Es a dir el filtre descrit per:

$$y[n+2] = x[n+2] - \frac{2}{3}x[n+1] + \frac{7}{8}y[n+1] - \frac{3}{32}y[n]$$

Per obtenir la resposta impulsional $h_{IIR}[n]$, només hem de fer passar per el filtre un senyal equivalent a una delta tal que $x[n] = \delta[n]$.

```
clear all
clc

x=[1 zeros(1,39)];

a=[1 -7/8 3/32];
b=[1 -2/3];
N=length(x);

h=iir_fd1(a,b,x,N);

figure(3)
stem(h, '-o')
grid on
title('Resposta impulsional h_{IIR}[n] del filtre (1)');
```

Mirem ara si per un senyal graó $x[n]$ obtenim la mateixa resposta quan el fem passar pel filtre (1) (calcular $iir_df1(x[n])$) o quan el fem passar per un sistema FIR amb resposta impulsional $h_{IIR}[n]$ (calcular $x[n]*h[n]$ on $*$ representa el producte de convolució).

```
clear x
N=40;
for k=0:N-1
    x(k+1)=step(k);
end
a=[1 ; -7/8 ; 3/32]; b=[1; -2/3];
h=h([1:10]);

y1=iir_fd1(a,b,x,N);
y2=conv(x,h);

figure(4)
stem(y1, '-b')
hold on
stem(y2, '-*')
grid on
title('');
```

```
legend('Resposta del filtre (1)', 'Resposta del sistema  
FIR', 'Location', 'southwest');
```

Activitat 3.3:

```
clear all  
clc
```

```
N=40;  
a=[1 -sqrt(2) 1];  
b=[1 1.1 0.5];
```

```
x1=cos((pi/4)*[0:N]);  
x2=cos((3*pi/4)*[0:N]);
```

```
y1=filter(b,a,x1);  
y2=filter(b,a,x2);
```

```
figure(5)  
stem(y1, '-r')  
hold on  
stem(y2, '-b')  
grid on
```

```
function y=iir_fd1(a,b,x,N)  
y(1)=b(1)*x(1);  
y(2)=b(1)*x(2)+b(2)*x(1)-a(2)*y(1);  
for k=1:N-2  
    compox=b(1)*x(k+2)+b(2)*x(k+1);  
    compoy=-a(2)*y(k+1)-a(3)*y(k);  
    y(k+2)=(1/(a(1)))*(compox+compoy);  
end  
end
```

```
function y=iir_fd2(a,b,x,N)  
y(1)=b(1)*x(1);  
y(2)=b(1)*x(2)+b(2)*x(1)-a(2)*y(1);  
for k=1:N-2  
    y(k+2)=(1/(a(1)))*(b(1)*x(k+2)+b(2)*x(k+1)-a(2)*y(k+1)-a(3)*y(k));  
end  
end
```

```
function out=delta(in)  
if in==0  
    out=1;  
else  
    out=0;  
end  
end
```

```
function out=step(in)  
if in>=0  
    out=1;  
else  
end
```

```
        out=0;  
end  
end
```

Published with MATLAB® R2020a