

Instituto
Tecnológico y de
Estudios Superiores
de Occidente –
ITESO



ITESO
Universidad Jesuita
de Guadalajara

Materia: Sistemas de comunicaciones digitales

Profesor: Dr. Omar Longoria

Fecha: 28/09/2022

Autor(es): Alexis Luna Delgado

TAREA 5

EJERCICIO 1.

- Con el código de la Tarea 4 genere la sección de la Lena que va a utilizar como se indicó arriba y verifique la cantidad de bits a transmitir.

```
%% 1,2,3,4
%load the image
load lena512.mat
%assign image to variable
a = lena512;
%show image
imshow(uint8(lena512))
%show the eye only
lenarec=lena512(251:297,321:367);
imshow(uint8(lenarec))
%Put the bits of the image in a vector
b=de2bi(lenarec,8);
b=b';
bits=b(:);
```



lenaRec

47x47 double

- Aplique el código correspondiente para verificar que puede reconstruir la imagen a partir del vector de bits.

```
%Recover the bits as a matrix
b2 = reshape(bits, 8, 47*47);
b2 = b2';
%get the decimal values of the bit matrix
dec = bi2de(b2);
%convert the decimal vector into a matrix
lenaReconst = vec2mat(dec,47,47);
%rotate image 90° to fix error
lenaReconst = lenaReconst';
imshow(uint8(lenaReconst))
```



lenaReconst

47x47 double

- Con la Tarea 4, genere las señales (tren de pulsos) Unipolar NRZ, Polar NRZ, Manchester y Bipolar RZ.

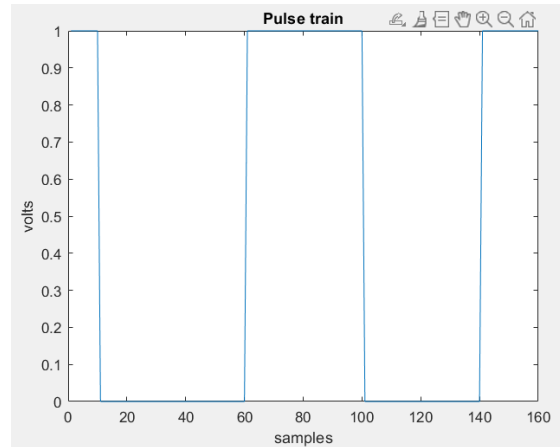
Unipolar NRZ:

```
mp = 10; % samples per pulse
Fs = 96000; % It can be different
Ts = 1/Fs;
Rs = Fs/mp;
Rb = Rs;
%Base pulse
```

```

pnrz = 1*ones(1,mp);
wvtool(pnrz)
%Impulse train
s = zeros(1,numel(bits)*mp);
s(1:mp:end) = bits;
figure
stem(s(1:mp*16))
xlabel('samples')
ylabel('volts')
title('Impulse train')
%Pulse Train
xUNRZ = conv(pnrz,s);
figure
plot(xUNRZ(1:mp*16))
xlabel('samples')
ylabel('volts')
title('Pulse train')

```



Polar NRZ:

```

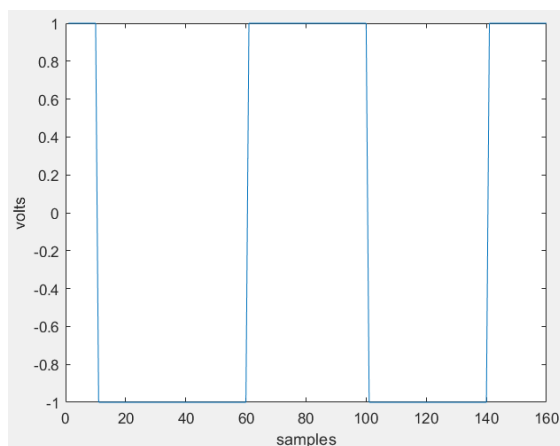
%% Polar NRZ
%load the image
load lena512.mat
%assign image to variable
a = lena512;
%show image
imshow(uint8(lena512))
%show the eye only
lenarec=lena512(251:297,321:367);
imshow(uint8(lenarec))

```

```

mp = 10; % samples per pulse
Fs = 96000; % It can be different
Ts = 1/Fs;
Rs = Fs/mp;
Rb = Rs;
%Base pulse
pnrz = ones(1,mp);
wvtool(pnrz)
s1 = bits;
s1(s1==0) = -1;
s = zeros(1,numel(s1)*mp);
%Impulse train
s(1:mp:end)=s1;
stem(s(1:mp*16))
xlabel('samples')
ylabel('volts')
%Pulse Train
xUNRZ = conv(pnrz,s);
figure
plot(xUNRZ(1:mp*16))
xlabel('samples')
ylabel('volts')

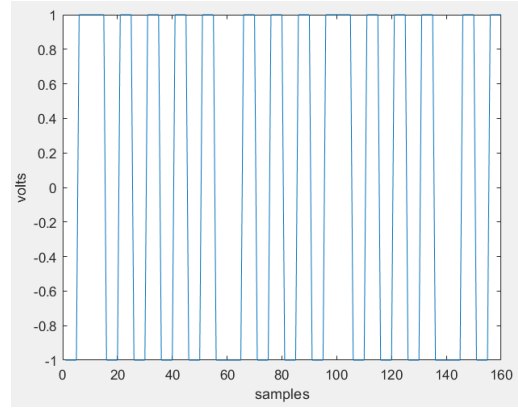
```



Manchester:

```
%% Manchester
load lena512.mat
a = lena512;
imshow(uint8(lena512))
lenarec=lena512(251:297,321:367);
imshow(uint8(lenarec))
b = de2bi(lenarec,8);
b = b';
bits = b(:);

mp = 10; % samples per pulse
Fs = 96000; % It can be different
Ts = 1/Fs;
Rs = Fs/mp;
Rb = Rs;
%Base pulse for 1
manch = [-1*ones(1,mp/2) ,
ones(1,mp/2)];
wvtool(manch)
s1 = bits;
s1(s1==0) = -1;
%Impulse train
s = zeros(1,numel(s1)*mp);
s(1:mp:end) = s1;
stem(s(1:mp*16))
xlabel('samples')
ylabel('volts')
%Pulse train
xMANCH = conv(manch,s);
figure
plot(xMANCH(1:mp*16))
xlabel('samples')
ylabel('volts')
```



Bipolar NRZ:

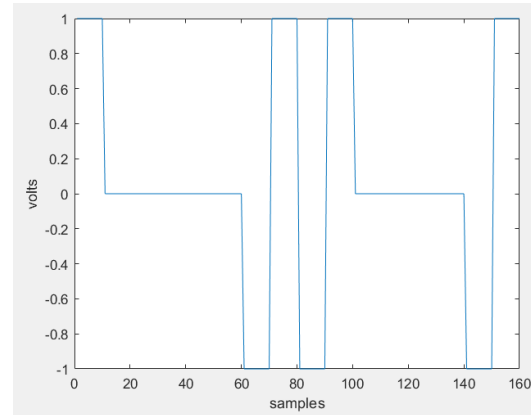
```
%% Bipolar NRZ (AMI)
load lena512.mat
```

```

a = lena512;
imshow(uint8(lena512))
lenarec=lena512(251:297,321:367);
imshow(uint8(lenarec))
b = de2bi(lenarec,8);
b = b';
bits = b(:);

mp = 10; % samples per pulse
Fs = 96000; % It can be different
Ts = 1/Fs;
Rs = Fs/mp;
Rb = Rs;
% The bit rate
%Base pulse
bprz = ones(1,mp);
wvtool(bprz)
s1 = bits;
flag = -1;
for count = 1:numel(s1)
    if s1(count) == 1
        flag = -flag;
    end
    s1(count) = (s1(count))*flag;
end
s = zeros(1,numel(s1)*mp);
%Impulse train
s(1:mp:end) = s1;
stem(s(1:mp*16))
xlabel('samples')
ylabel('volts')
%Pulse Train
xBPRZ = conv(bprz,s);
figure
plot(xBPRZ(1:mp*16))
xlabel('samples')
ylabel('volts')

```



- Diseñe filtros pasabajas de orden 100 y frecuencias de corte NORMALIZADAS en 0.05, 0.2 y 0.4

%% Filtros

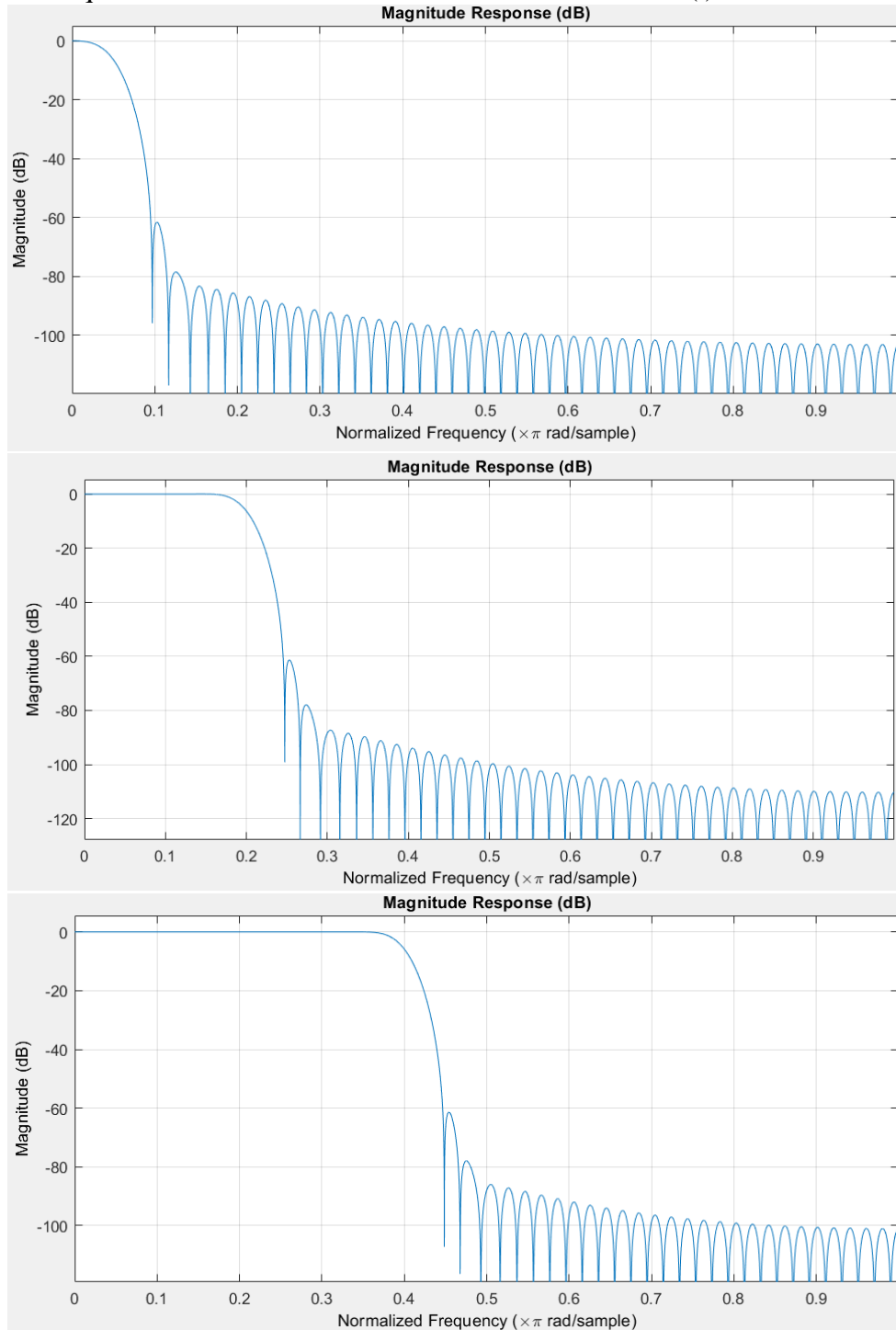
```

f= [0 0.2 0.2 1]; % Vector de Frecuencias
m= [1 1 0 0]; % Vector de Magnitudes
ford=100; % Orden del Filtro
f1 = fir2(ford,f,m); % Coeficientes del Filtro usando FIR2( )
fvtool(f1) % Herramienta para analizar el Filtro
f= [0 0.05 0.05 1]; % Vector de Frecuencias
f2 = fir2(ford,f,m); % Coeficientes del Filtro usando FIR2( )
fvtool(f2) % Herramienta para analizar el Filtro
f= [0 0.4 0.4 1]; % Vector de Frecuencias

```

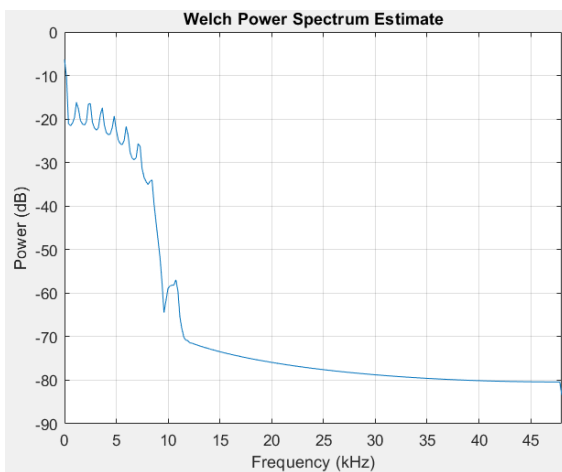
```
f3 = fir2(ford,f,m); % Coeficientes del Filtro usando FIR2( )
fvtool(f3)           % Herramienta para analizar el Filtro
```

- Verifique los LPF diseñados usando la herramienta fvtool().

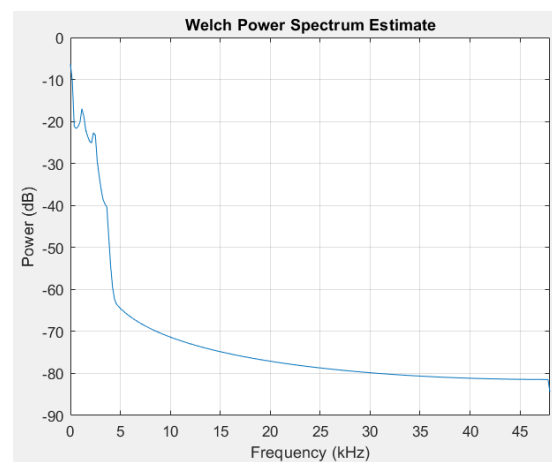


- Con cada LPF, filtre los códigos de línea generados.
- Grafique los espectros de cada una de las señales filtradas usando estimador espectral de potencia.
PSD de unipolar NRZ filtrado:

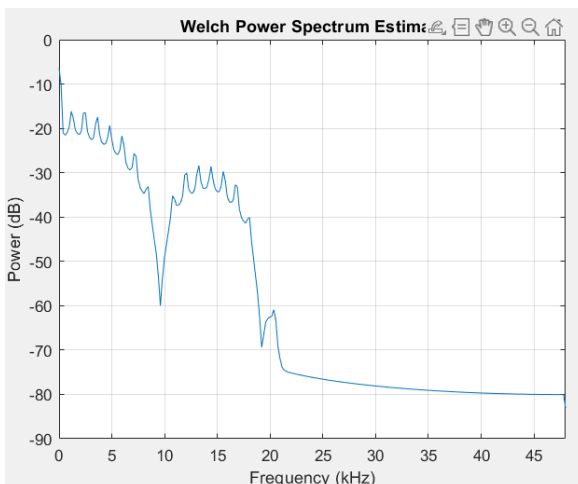
```
%Filtrado de unipolar NRZ
xfil1= conv(xUNRZ, f1);
xfil2= conv(xUNRZ, f1);
xfil3= conv(xUNRZ, f1);
pwelch(xfil1,500,300,500,Fs,'power');
figure
pwelch(xfil2,500,300,500,Fs,'power');
figure
pwelch(xfil3,500,300,500,Fs,'power');
```



$F_c=0.2$

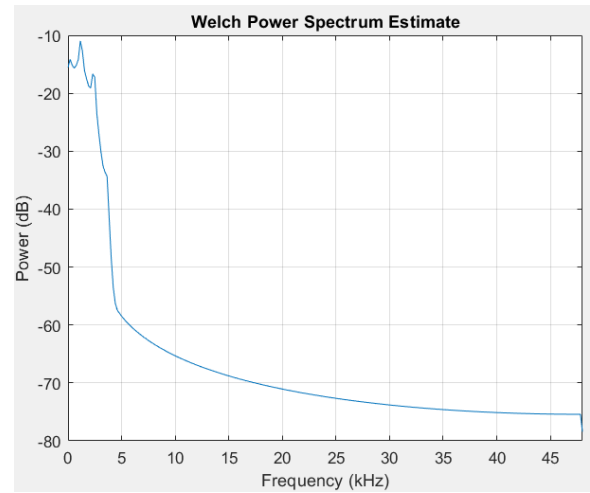
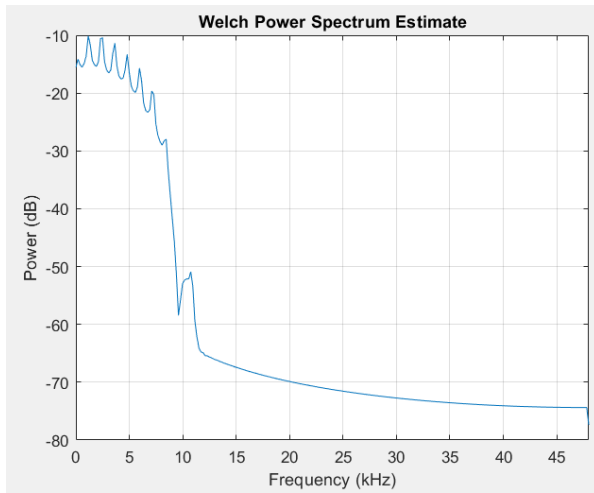


$F_c=0.05$

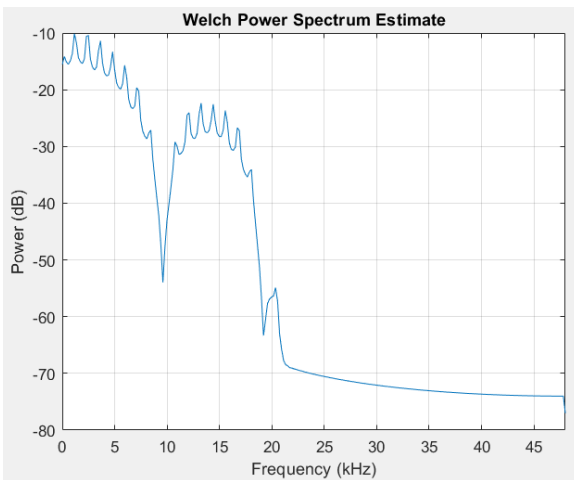


$F_c=0.4$

PSD de Polar NRZ filtrado:



$F_c=0.2$

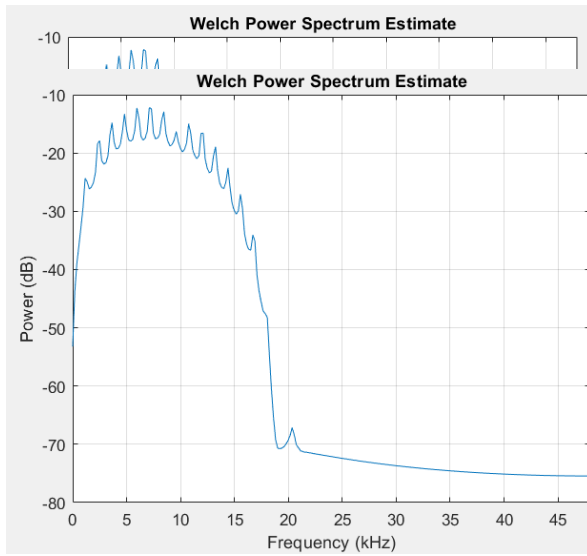


$F_c=0.05$

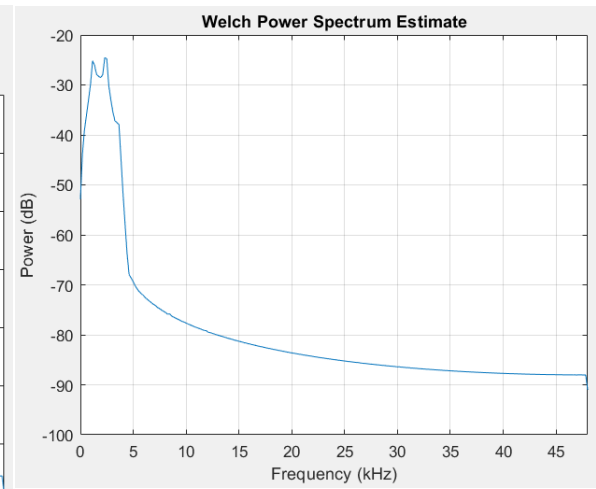
$F_c=0.4$

PSD de Manchester Filtrado

$F_c=0.2$

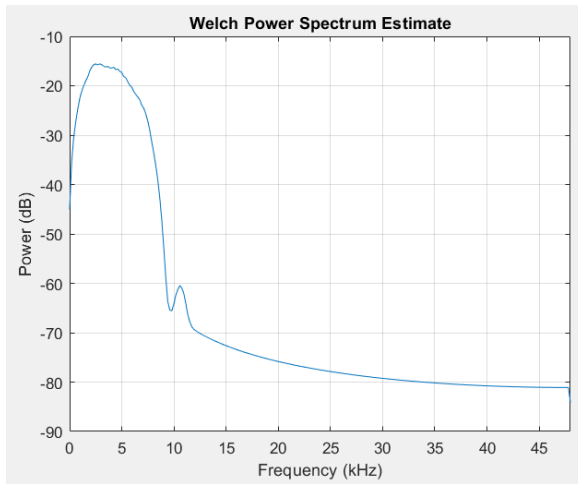


$F_c=0.05$

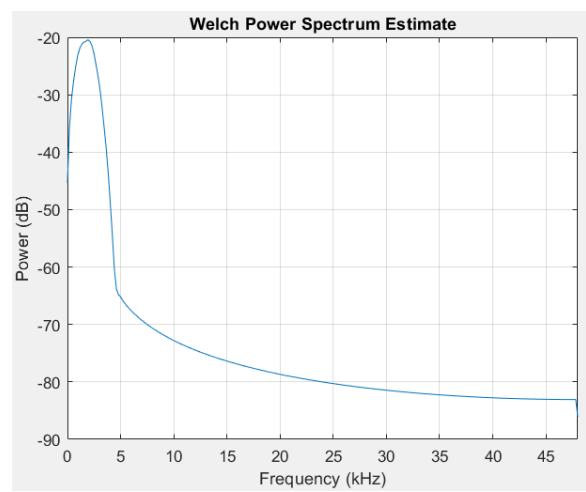


$F_c=0.4$

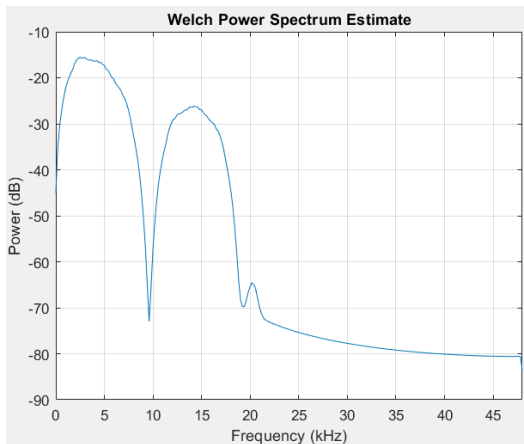
PSD de Bipolar NRZ



$F_c = 0.2$



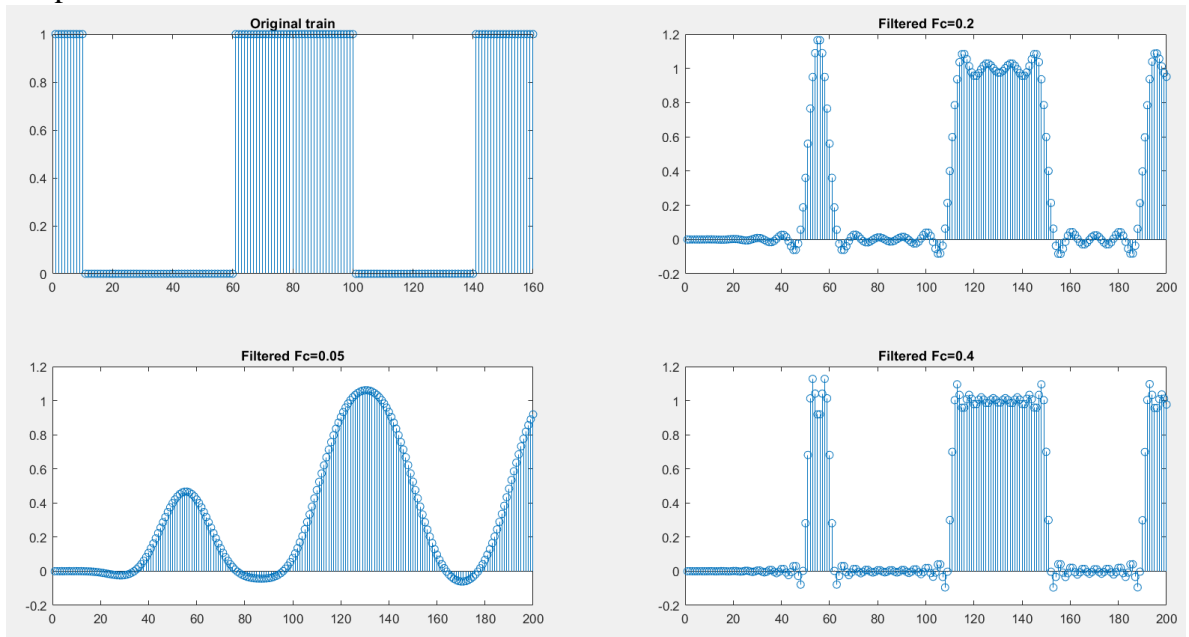
$F_c=0.05$



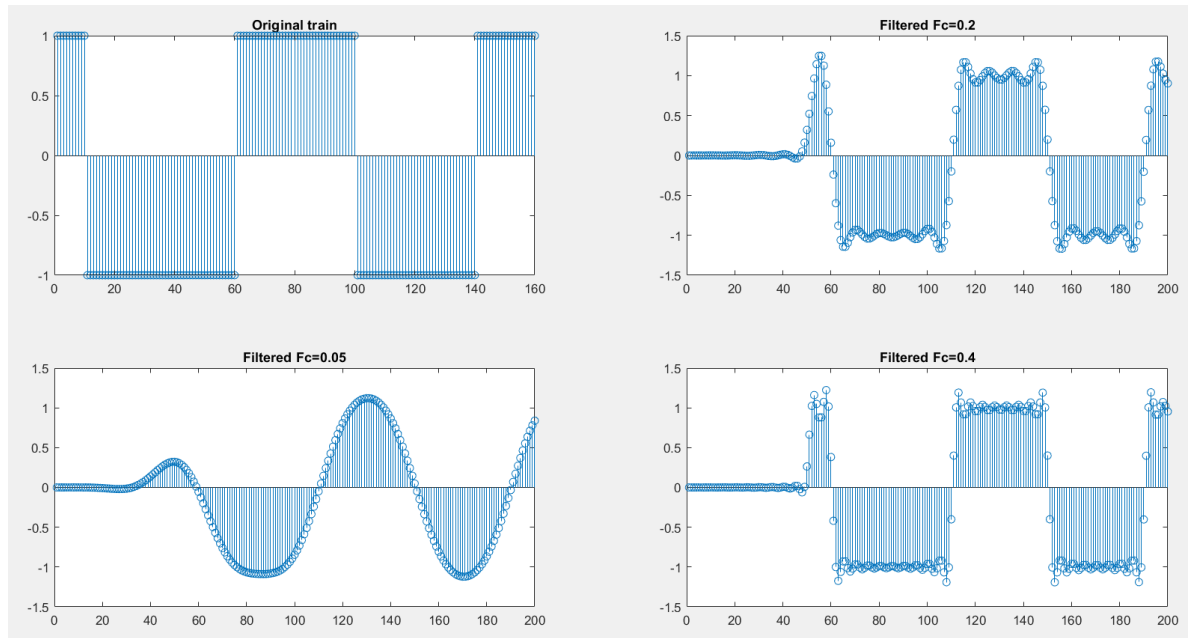
$F_c=0.4$

- Grafique con la función stem () las señales originales y las filtradas, usando solamente las muestras correspondientes a los primeros 16 bits (es decir con 16 x mp muestras + el retardo del filtro).

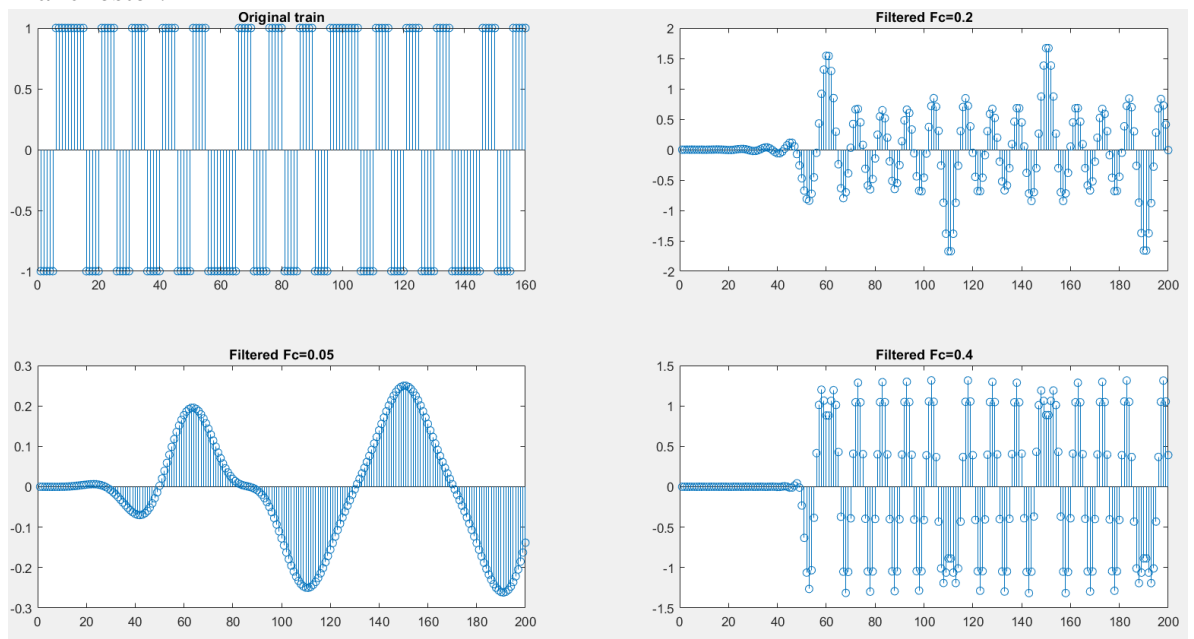
Unipolar NRZ:



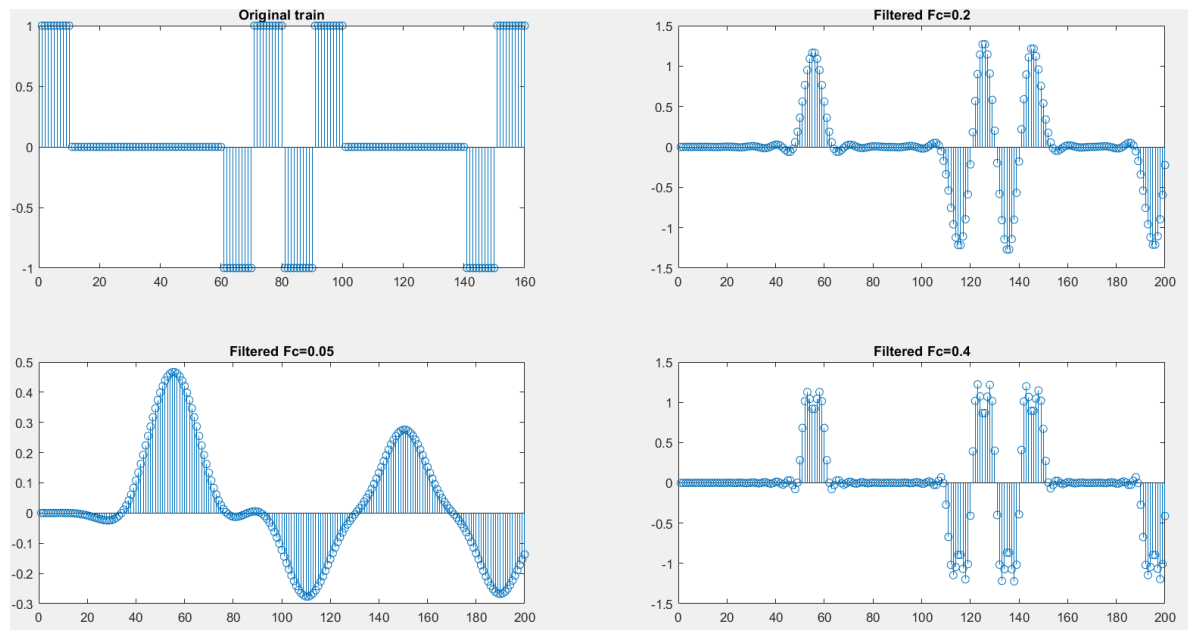
Polar NRZ:



Manchester:



Bipolar NRZ



- Suponiendo que tenemos canales de comunicación que se comportan como los LPF que diseñamos,
¿Cuál es la factibilidad de transmitir/recibir cada señal por esos canales? Explique sus argumentos.

No es muy factible transmitir las señales al menos por los canales con menor frecuencia de corte, principalmente cuando se usa código Manchester o Bipolar NRZ. El canal de menor frecuencia de corte no es apto para ninguna señal y los otros dos varían dependiendo del código utilizado, siendo el de $F_c=0.4$ el más apto.

¿Cuál fue el impacto del filtrado sobre la señal original transmitida? Explique con sus propias palabras

Las señales además de sufrir un retraso presentaron distorsión principalmente en variaciones de la señal cuadrada a altas frecuencias, de tal forma que cada vez se obtenía una señal menos cuadrada conforme la frecuencia de corte bajaba.

¿Cómo diseñaría la funcionalidad del receptor? Explique con sus propias palabras
Utilizaría un receptor con un ancho de banda grande, para que permita pasar de buena manera las señales que se le transmitan, aunque también dependiendo del código de línea utilizado para adaptarse lo más posible a los requerimientos.

- Un monitor LCD de resolución 1920x1200 cuantifica el valor de cada pixel con 24 bits, y refresca la imagen a 60 veces por segundo. ¿Cuál es la tasa de bit en el cable entre el monitor y la computadora?
Ya que cada pixel consume 24 bits, se tiene una imagen de 1920x1200x24 bits, que nos da un total de 55,296,000 bits en toda la pantalla. Ahora si la imagen se refresca a 60 veces por segundo, quiere decir que todo eso se multiplica x60, dando 3,317,760,000 bps.
- ¿Cuál es la tasa de bit si consideramos un monitor del tipo 4K?
Para 4K de 4096x2600, son 255,590,400 bits por imagen, por lo que son 1.534e10 bps.
- ¿Cuál es la tasa de bit que soporta USB3.1, HDMI y PCIe 4.0?
USB 3.1 acepta velocidades de hasta 5Gbps, HDMI de 48Gbps y PCIe 4.0 16GT/s.