Exercise 2

# Aliasing,Sampling and Reconstruction

Univerza *v Ljubljani*

Sergio Gasquez Arcos

# Index

# 1 Objective

The main target of this laboratory exercise is to better understand and see,in a graphic way,how this three effects affects signals.Also is to reinforce the three concepts since they are all three basics.Each exercise will cover one of those three topics.

# 2 Exercise 1:Anti-Alising filter

**Determine corner frequency and lowest possible order of anti-aliasing filter with slope of $-N \cdot 20dB/dec$ SNR in the band from 5kHz to 10kHz must be 100dB.**

We are interested in the frequencies between 5kHz and 10kHz,so as corner frequency we will take 15kHz,we are leaving a bit more because is not a theoretical filter. Now we need to determine the order of the filter,first order will only give us additional 40dBs,which is not enough to get to the threshold of 100dBs due to that reason we will need a second order filter

After creating the simulink model where we can see that we defining 4 signals and then moving it to 10kHz.Finally are defining two variables that will have this output one with filter and another without.

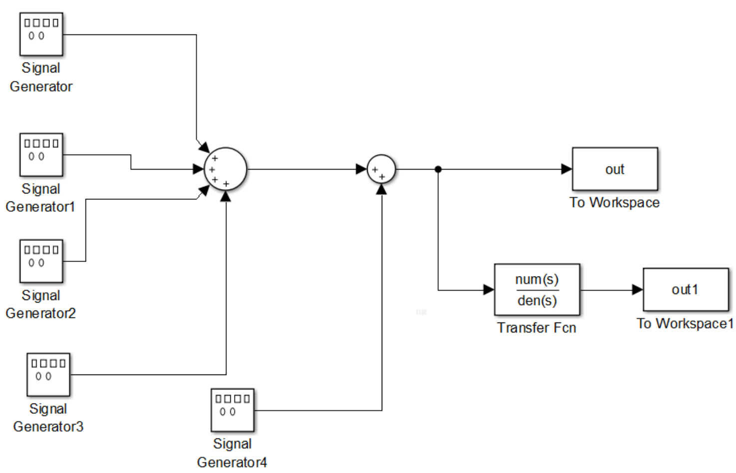


Figure 1: Simulink model for Exercise 1

As for the code,we will only define the constants and signals given in the statement of the exercise,then we will use the function butter to create the Butterworth filter that we use in the simulink model.Finally we plot both outputs of the simulink simulation.

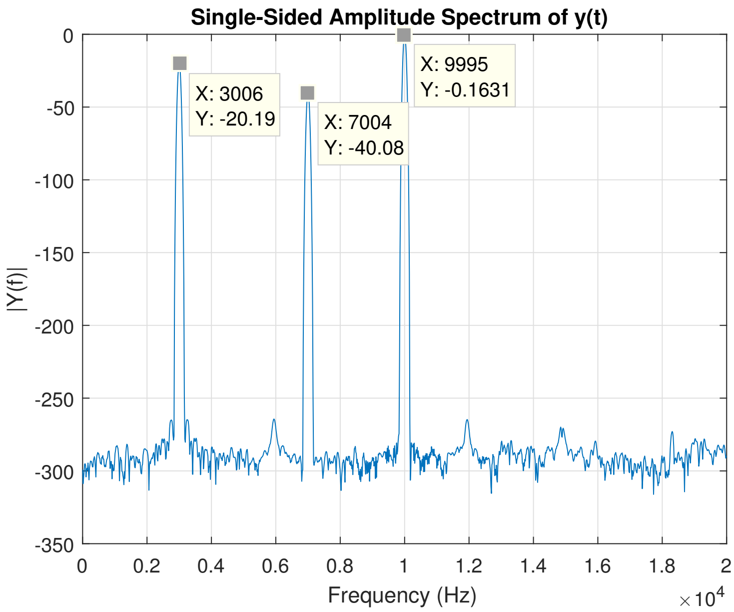


Figure 2: Sampling without anti-aliasing filter

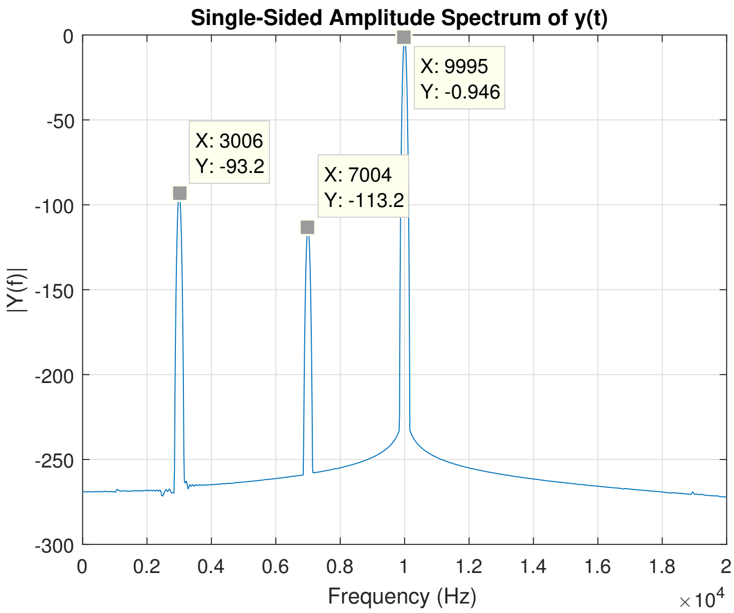


Figure 3: Sampling with anti-aliasing filter

We are getting a cleaner version of the signal than in the pdf given to us because we are taking less number of steps(N).

If we compare both graphics we can see that before the filter the two first components wont pass through the filtering window.That is why in the first image those components are higher than in the second, while the third component remains the same. We also see less noise after the filter,that is because the filter will decrease all except the passing frequencies.

# 3 Exercise 2:Reconstruction filter

**12 bit D/A converter converts digital sine signal with f0=100 kHz to analog signal with sampling frequency fs=1MHz and amplitude A=1V**

- Plot the signal spectrum of the D/A output in range from 0 to 3fs
- Calculate order (N) and corner frequency fp of smoothing filter

  ($S = -N * 20dB/dec$) SNR from 0 to 2fs must be higher than 40dB (sinx/x effect )

As filter we will use a first order Butterworth,that will fulfil our necessities and his corner frequency will be f0.

In the simulink model we will use the "Zero-Order Hold" which is a sampler. The first output(named out2) will contain the signal without any treatment,the second one(out) will be the signal after sampling with Zero-Order Hold and the last one(out1) consist of a Zero-Order Hold and the Butterworth filter.

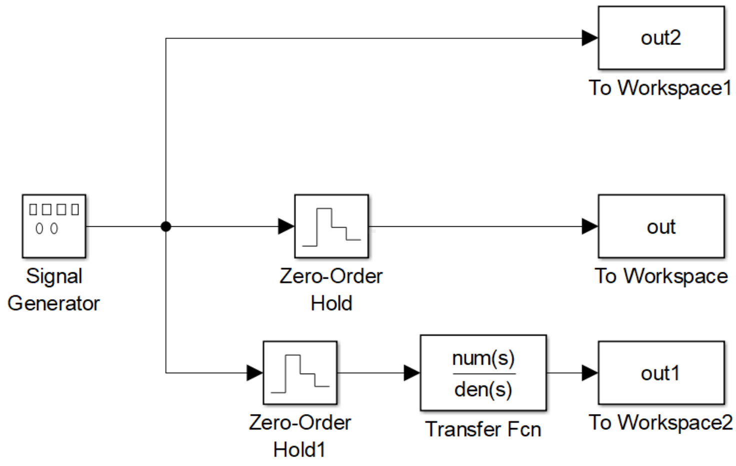


Figure 4: Simulink model for Exercise 2

Later,in the code, we will inserting 9 zeros between input samples in the first output.

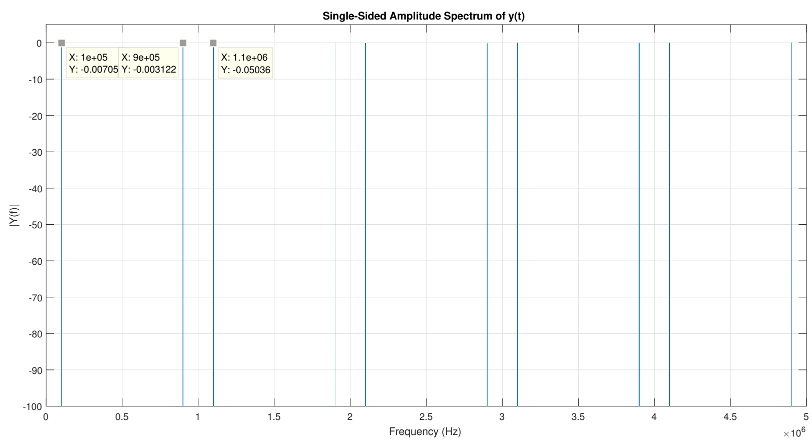


Figure 5: Original signal

As we can see,the original signal is repeated every fs and has the positive component and the negative one.In the f=0 we only see the positive one.

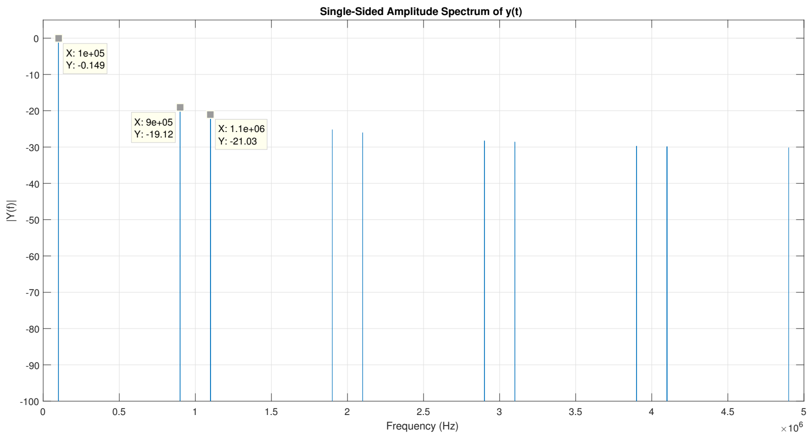


Figure 6: Signal after S/H

As we can see, only the first component remains the same.The other ones are decreased that is because the S/H behaves like a sinc on frequency domain.So each component will be more attenuated than his previous one.

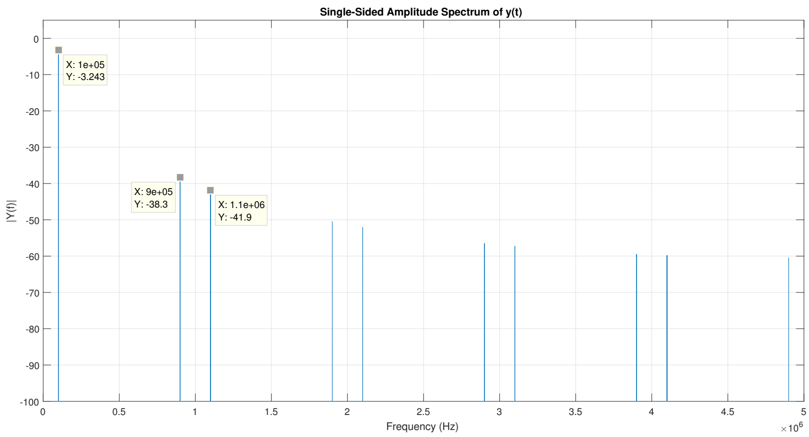


Figure 7: Signal after S/H and filter

After using Sample/Hold and the designed filter we can see that the first component remains the same,but the other ones a decreasing sample after sample.Now it decreases faster since it decreases from two sources(the filter and the sinc from S/H).

# 4 Exercise 3:Sampling rule

**From fl=10.21 MHz to fh=10.39 is the spectrum before the sampling. Prepare .m file where you calculate the minimal possible sampling frequency and base Nyquist zone before sampling. Plot the spectrum before and after sampling**

$$\frac{2f_L}{n} > \frac{2f_H}{(n+1)}$$

First we will do the maths and calculate what is the sampling frequency: $BW = f_H - f_L = 180KHz$ ,so thats means that the sampling frequency should be at least: $Fs = 2 * BW = 360KHz$,with this Fs , $f_H and f_L$ we can obtain n from the formula given in the statement $N = 56.7$,so we have to truncate it,N will be now 56 and recalculate de new Fs. $Fs = 364.6KHz$

As for the simulink model we will use this one:

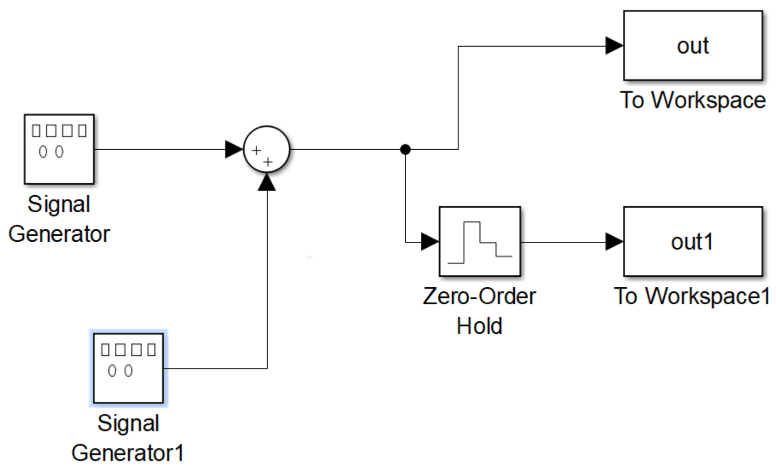


Figure 8: Simulink model for Exercise 3

We are adding the two components given $f_H$ and $f_L$,and having two outputs,one with any kind of processing and the other one with S/H with a Zero-Order Hold

After writing the code,we use a Fs of 2 times the BW we obtain this:

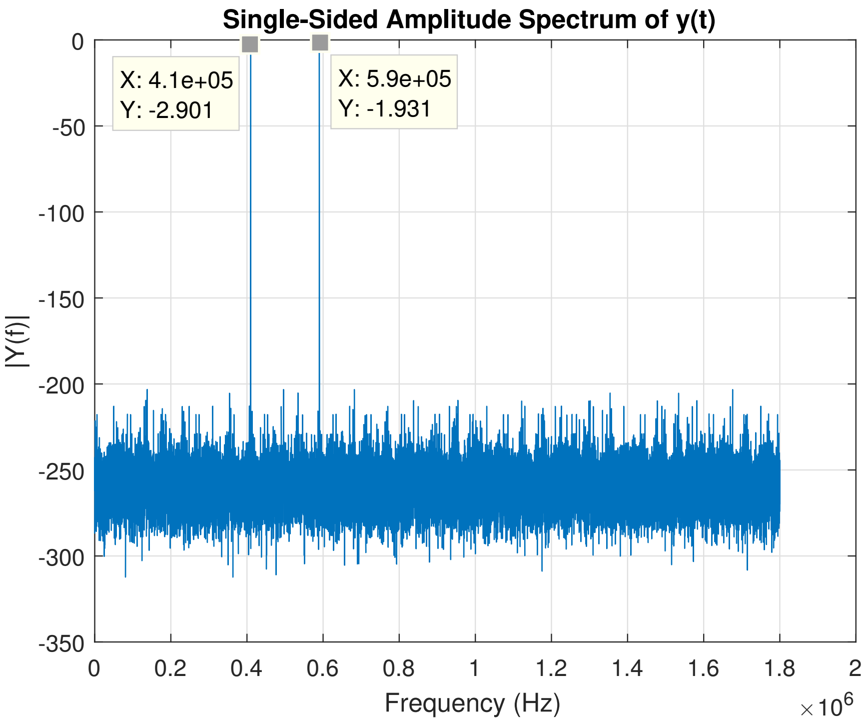


Figure 9: Signal

We can see the two components in the frequency domain,now we will print the sampled version of this two components:

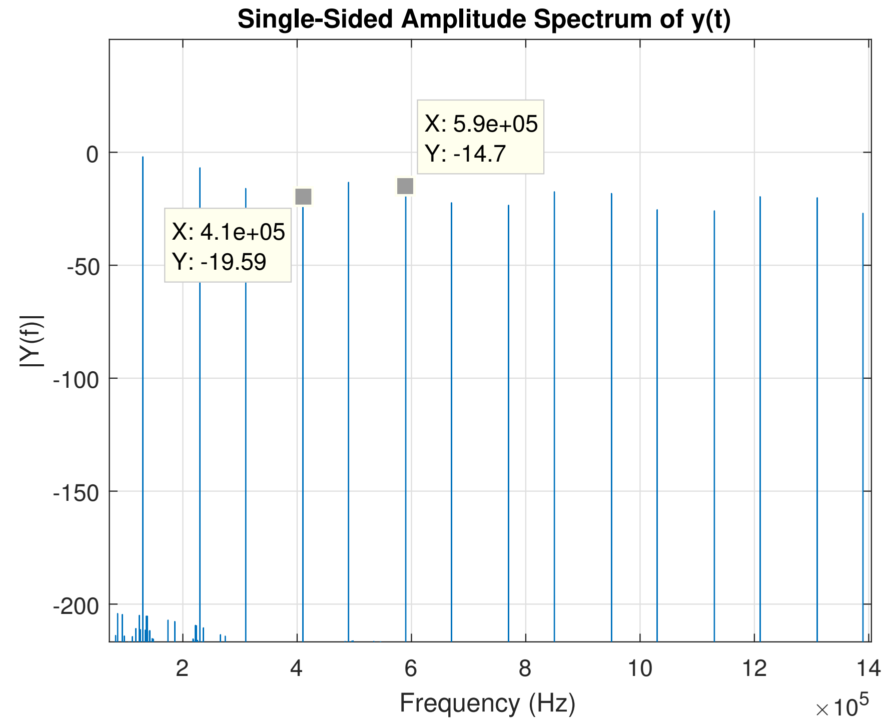


Figure 10: Signal after S/H

We can see that between the 2 samples there is another sampled of the next iteration.

Now we will see the different Nyquist zones,in the even part,the signal wil be replicated as for the odds part the signal will be mirrored.We are using the sample frequency calculated at the first part of the exercise(Fs=364.6KHz).Each Nyquist zone is Fs/2.This is the result:

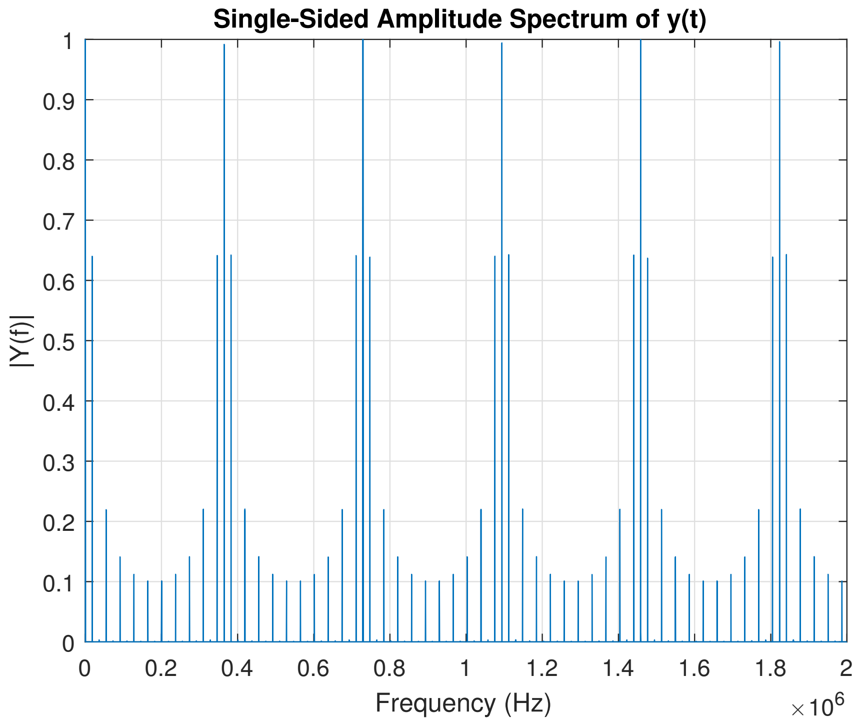


Figure 11: Nyquist zones

As we can see ,the image correspond with what we expected.

# 5 Appendix

## 5.1 Exercise 1

```
clear all;
close all;
% We define time of sampling,freq of sampling and number of steps
fs=1*1e6;
Ts=1/fs;
N=2^16;

% We define frequencies of all signals
f1=fs/2-(1*1e3);
```

```
f2=fs/2 + (2*1e3);
f3=fs+3*1e3;
f4=fs+7*1e3;

% We define the chosen corner frequency and order of the filter
wp=2*pi*15000; % We give the corner frequency in angular
    frequency
% since we are using the function butter and it's how it requires
ord=2;
[num,den]=butter(ord,wp,'s');

% Once created the simulink model we simule
options=simset('RelTol', 1e-3,'MaxStep', 1/(10*fs),'FixedStep',1/
    fs);
sim('Exercise12',(N-1)*Ts,options); %startup of Simulink model

% Finally we plot the results
fft_plot(out,fs,'lin','dB');
fft_plot(out1,fs,'lin','dB');
```

## 5.2 Exercise 2

```
clear all;
close all;

% We define time of sampling,freq of sampling and number of steps
Fs=1e6;
Ts=1/Fs;
N=2^15;
% We define Amplitude and f0
A=1;
f0=100*1e3;

% We declare the inputs of the Butterworth filter
ord=1; % We will use a first order filter
```

```
wp=2*pi*f0; % Angular freq
[num,den]=butter(ord,wp,'s');


% We simulate the simulink model
options=simset('RelTol', 1e-3,'MaxStep', 1/(Fs*10),'FixedStep',1/
    Fs);
sim('Exercise21',(N-1)*Ts,options);

% For out2 we will use the funciont usample because it needs to
    be N-times
% higher
out2_upsample=upsample(out2,10);
% Finally we plot the outputs
fft_plot(out2_upsample*10,Fs*10,'lin','dB');
fft_plot(out,10*Fs,'lin','dB');
fft_plot(out1,10*Fs,'lin','dB');
```

## 5.3 Exercise 3

```
clear all;
close all;

% We define the two components
Fl=10.21*1e6;
Fh=10.39*1e6;

% We calculate band width,number of points,fs and Ts
BW=Fh-Fl;
Fs=2*BW; % Fs>2BW to be able to recover the signal
% Fs=364.6e3; %For the second part of the exercese
Ts=1/Fs;
N=2^15;

% We simulate the simulink model
```

```
options=simset('RelTol', 1e-3,'MaxStep', 1/(Fs),'FixedStep',1/Fs);
sim('Exercise31',(N-1)*Ts,options);

% out_upsampled=upsample(out1,60); %For the second part of the exercese

% We print the results
fft_plot(out,10*Fs,'lin','dB');
fft_plot(out1,10*Fs,'lin','dB');
% fft_plot(out_upsampled*60,Fs*60,'lin','lin');  %For the second part of the exercese
```