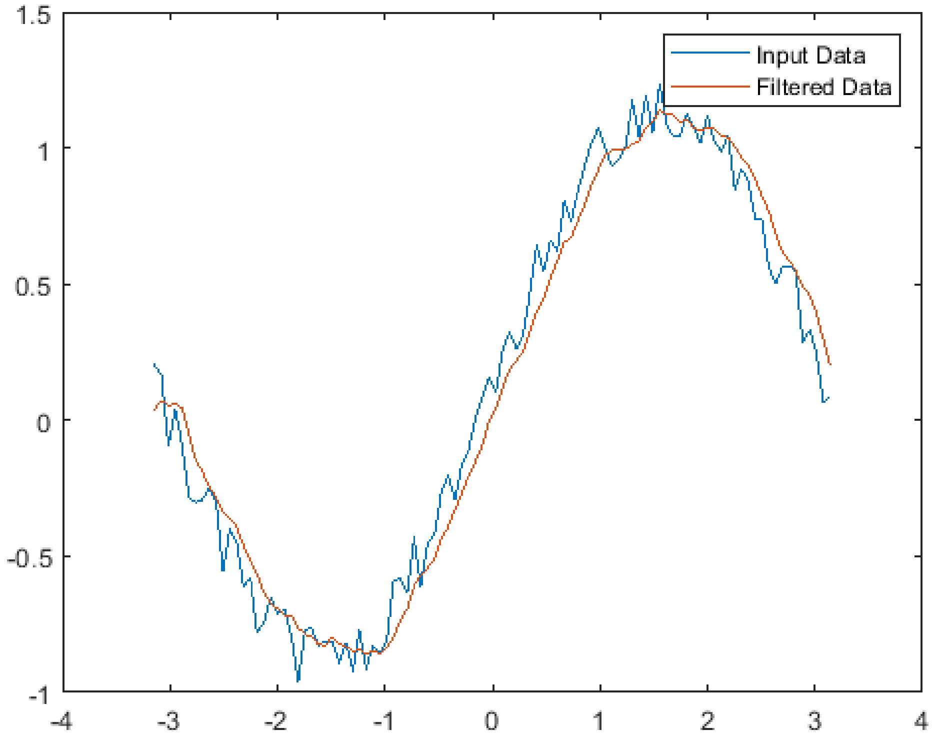


Lab Report

DIGITAL FILTERING AND APPLICATIONS

Dmytro Savchuk | Digital Signal Processing | 20/12/2021

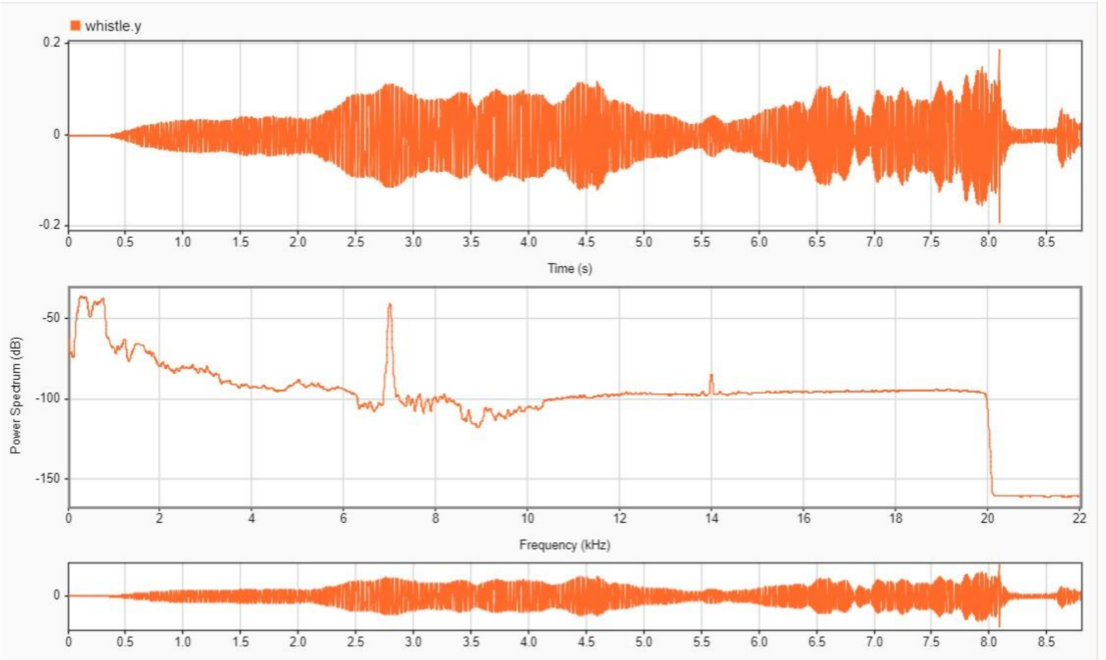
Introduction

In this lab, we were supposed to do the tutorial which gives us a brief introduction to signal filtering. Then, several assignments were done where we had to analyze our recorded data and other real-life data with the methods and theory that we learnt in Lectures 1 to 7.

Separation of two whistles:

In this assignment, we used our mobile phones to record 2 people whistling simultaneously for about 10s. One of them was whistling in the medium band and the other in the high-pitched band. The aim of this assignment is to separate the two whistles.

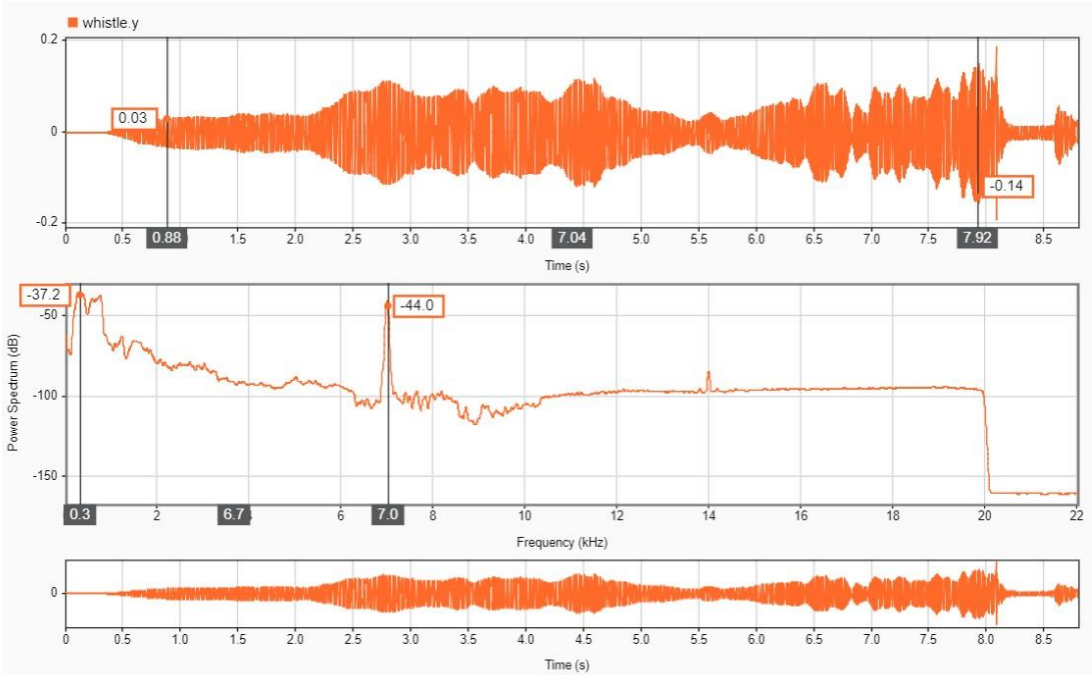
1. Spectrum of the recorded signal:



The script:

```
[y,fs] = audioread('Voice 023.m4a');
whistle = timetable(seconds((0:length(y)-1)'/fs),y);
sound(y,fs);
```

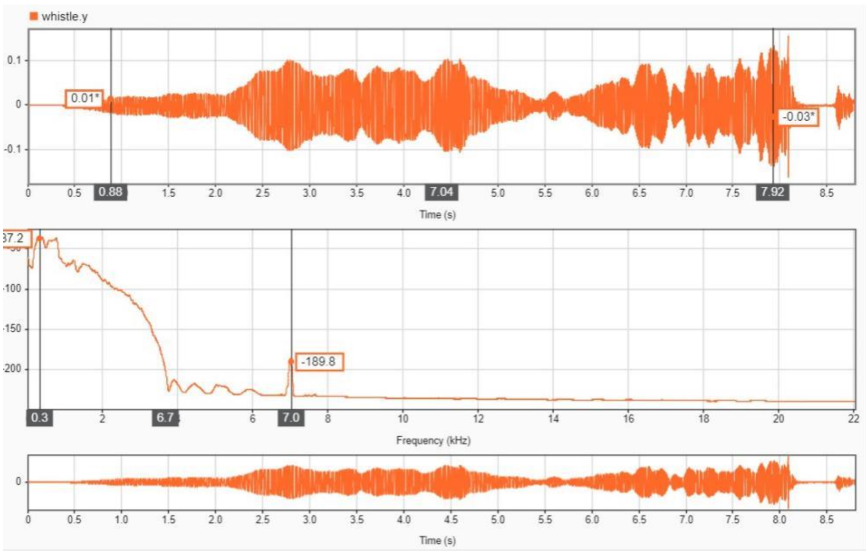
2. The dominating frequency of each of the two whistles:
As the following shot shows, we found that the dominating frequency of the medium band whistle was 300HZ while the high-pitched one was 7000 HZ.



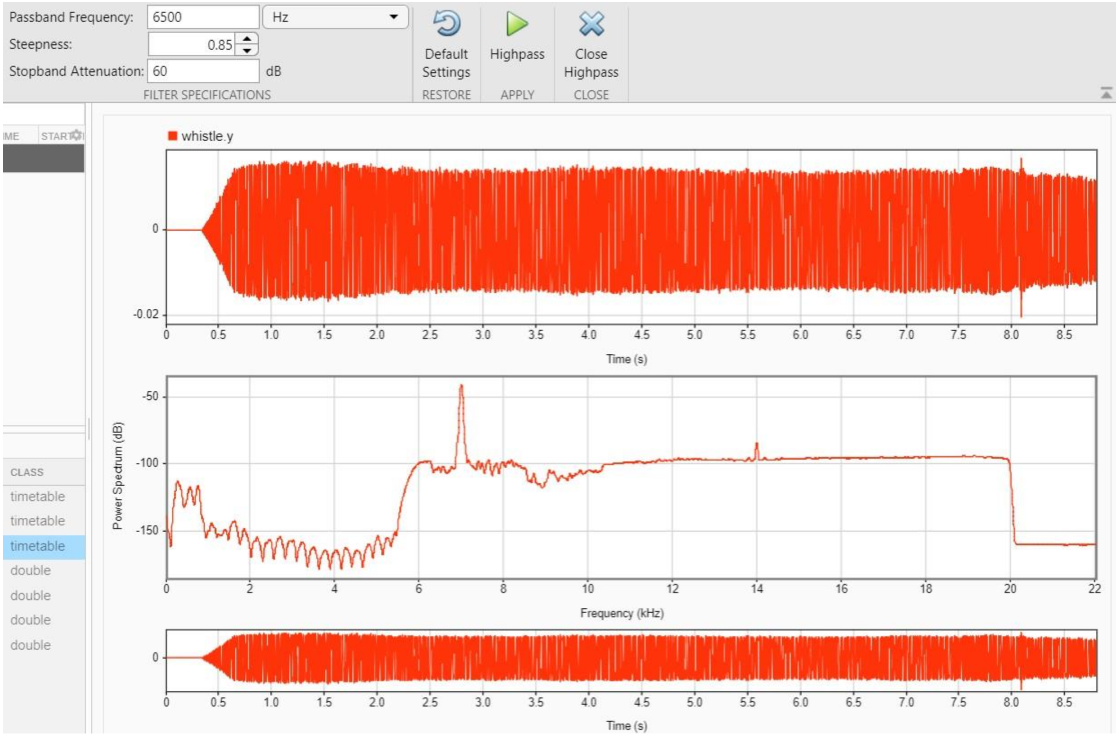
3. After analyzing the spectrum of the recorded whistle, we notice that the highest frequency recorded on one of the whistles was at 300HZ, so all of the below frequencies belong to this whistle. On the other hand, the high-pitched whistle had the fundamental frequency equal to 7000 HZ, so all below (and bigger than 300HZ) belong to this signal.
   We use the following script:

```
yfiltb = lowpass(y,400,fs);
lowPitched = timetable(seconds((0:length(yfiltb)-1)'/fs),yfiltb);
yfilth = highpass(y,6500,fs);
highPitched = timetable(seconds((0:length(yfilth)-1)'/fs),yfilth);
sound(yfilth,fs);
```

4. After implementing the result of the question 3, we get the following:
   For Law-pass filter (Medium band whistle):



For High-pass filter (High pitched whistle):

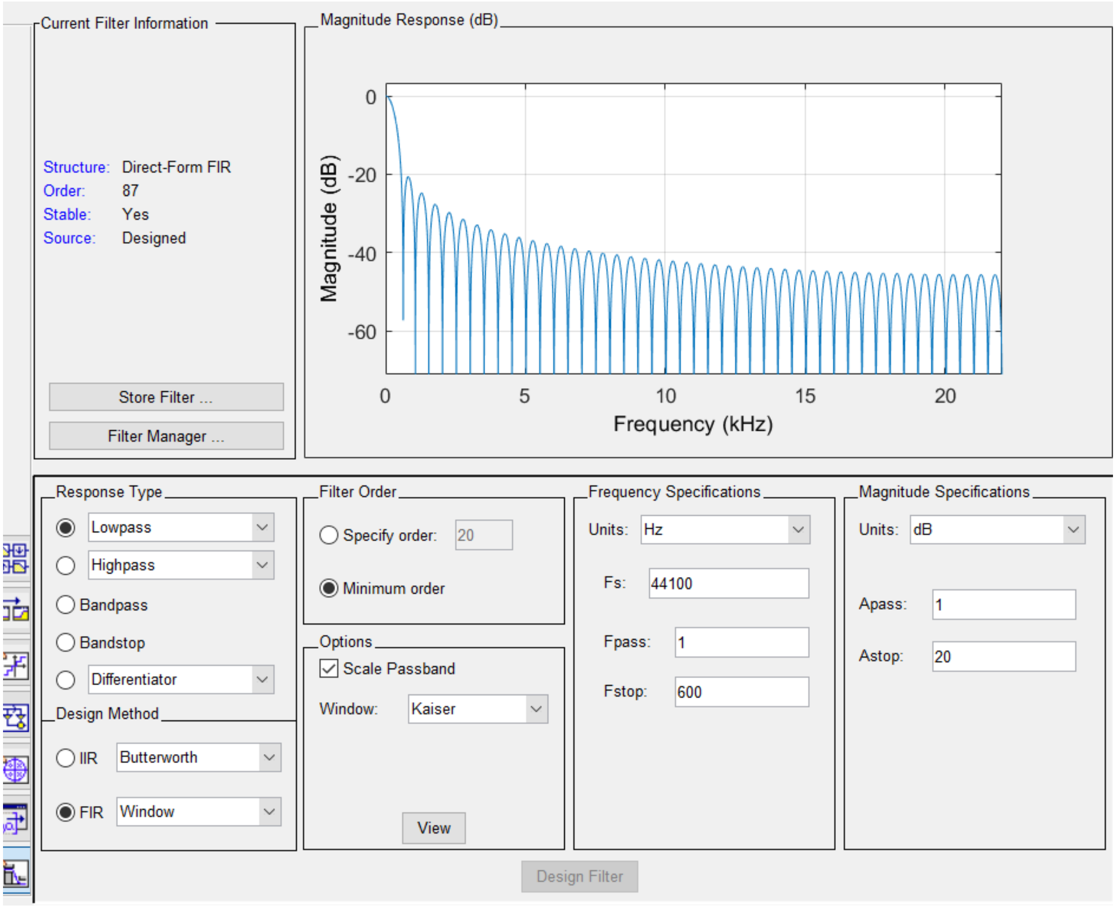


To get these representations, we go to signal analyzer on matlab, we choose analyzer, we choose high-pass or low pass filter and the filtering frequency.

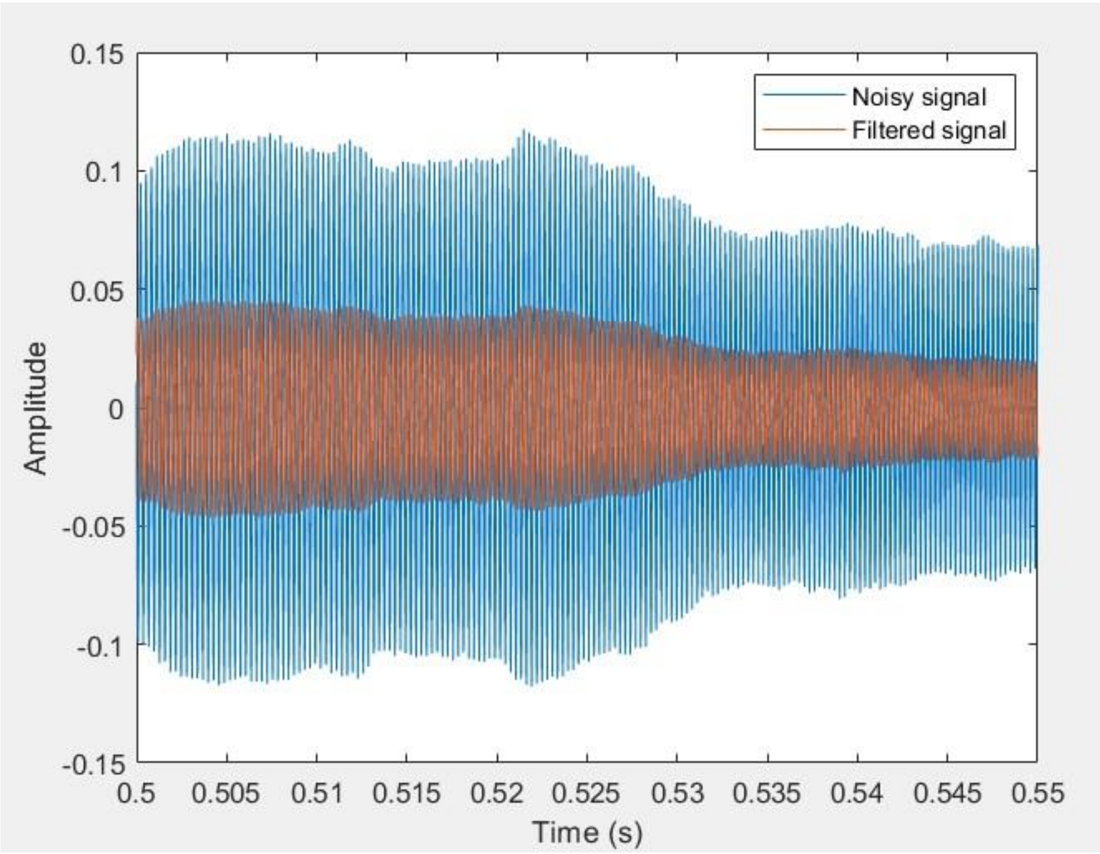
After listening to the resulting sounds after the filter, we find that we achieve the goal of the assignment where we can listen to each one of the whistles separately.

Using the filter designer:

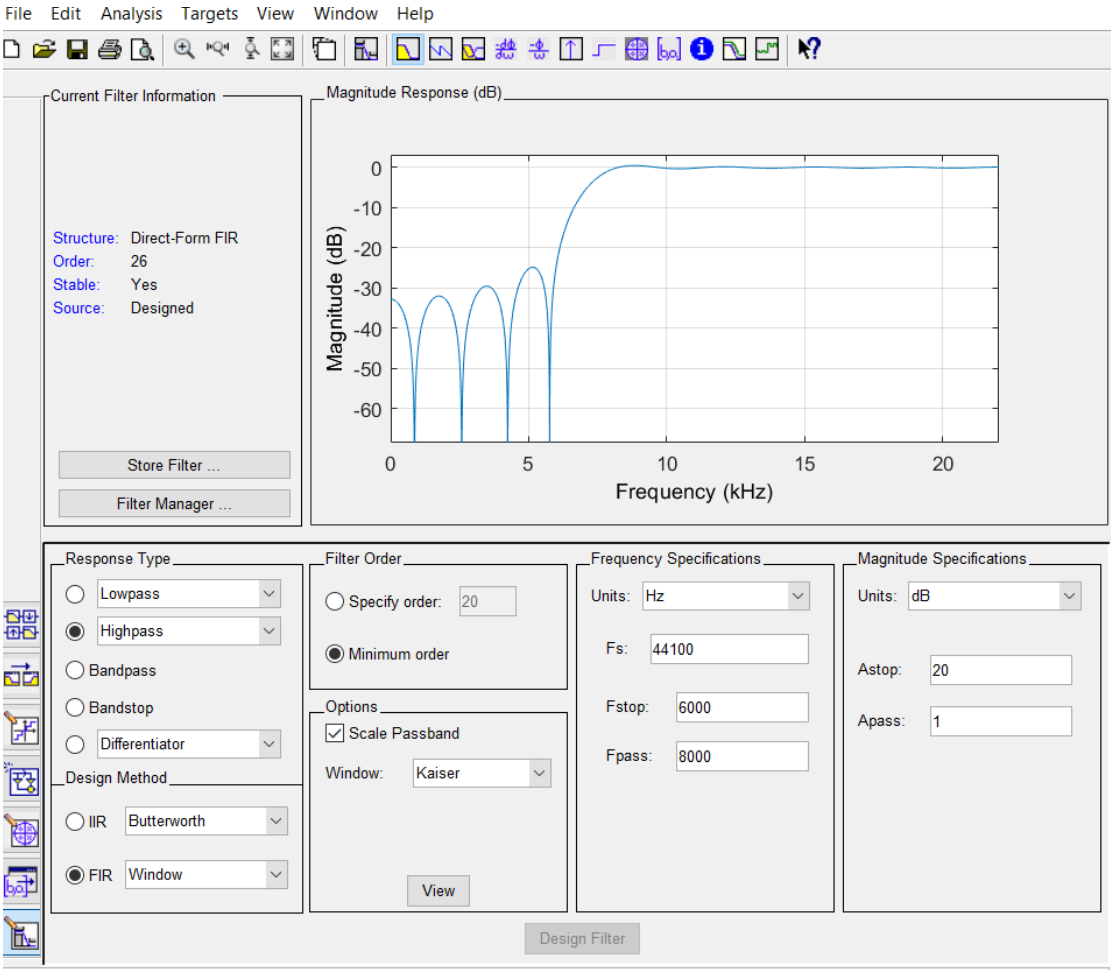
To get the low band whistle we have to design a low-pass filter, as it's mentioned in the picture below:

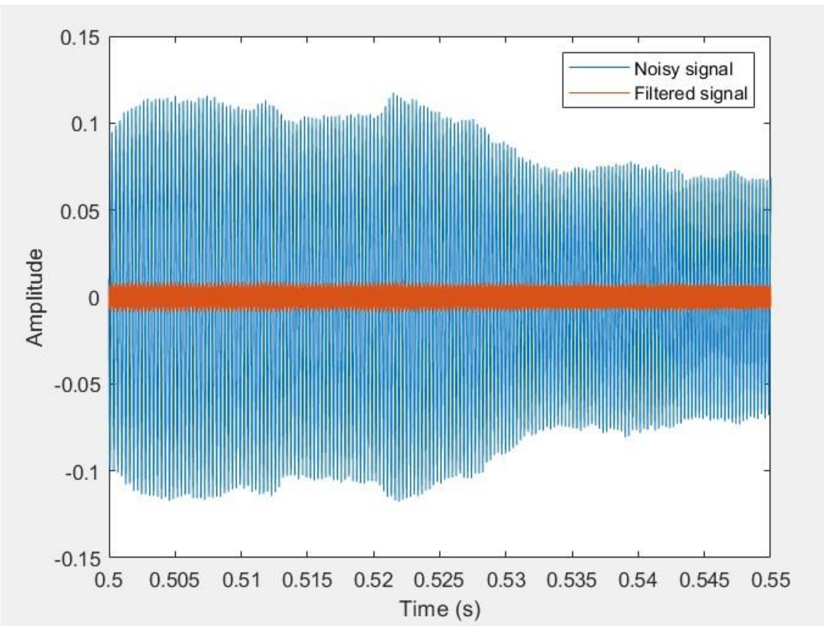


PS: we use 600 HZ here because using 400HZ gives us a really low sound.



To get the high-pitched whistle we have to design a high-pass filter, as it's mentioned in the picture below:

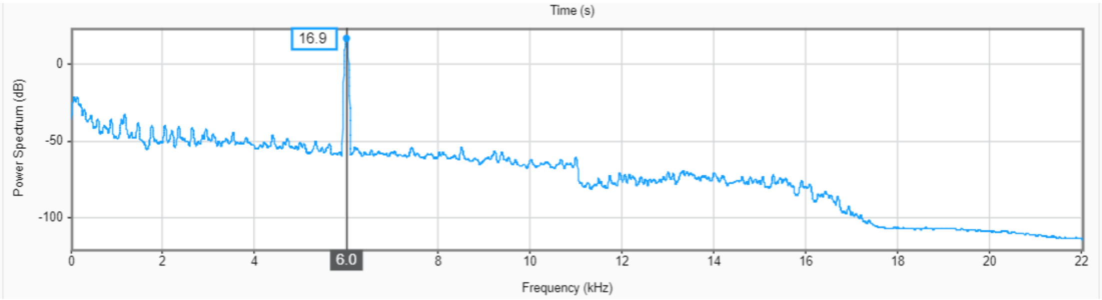




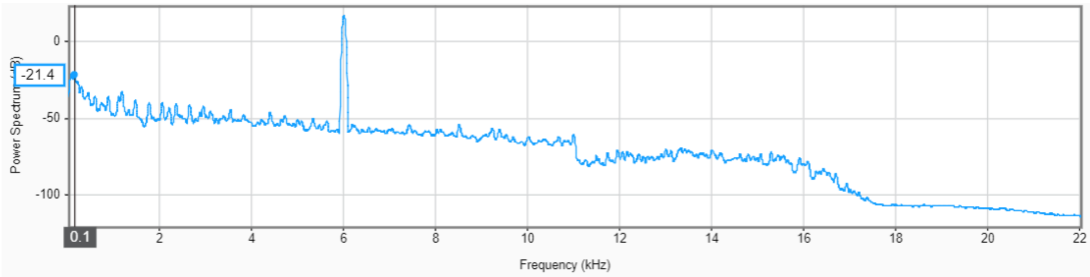
Noisy artefact in a piece of music:

In this exercise, we had a file music_track.mat contains a musical signal stored in y and sampled at fs that is contaminated by a noisy artefact. We were asked to get rid of the noise so we can have a clear music sound.

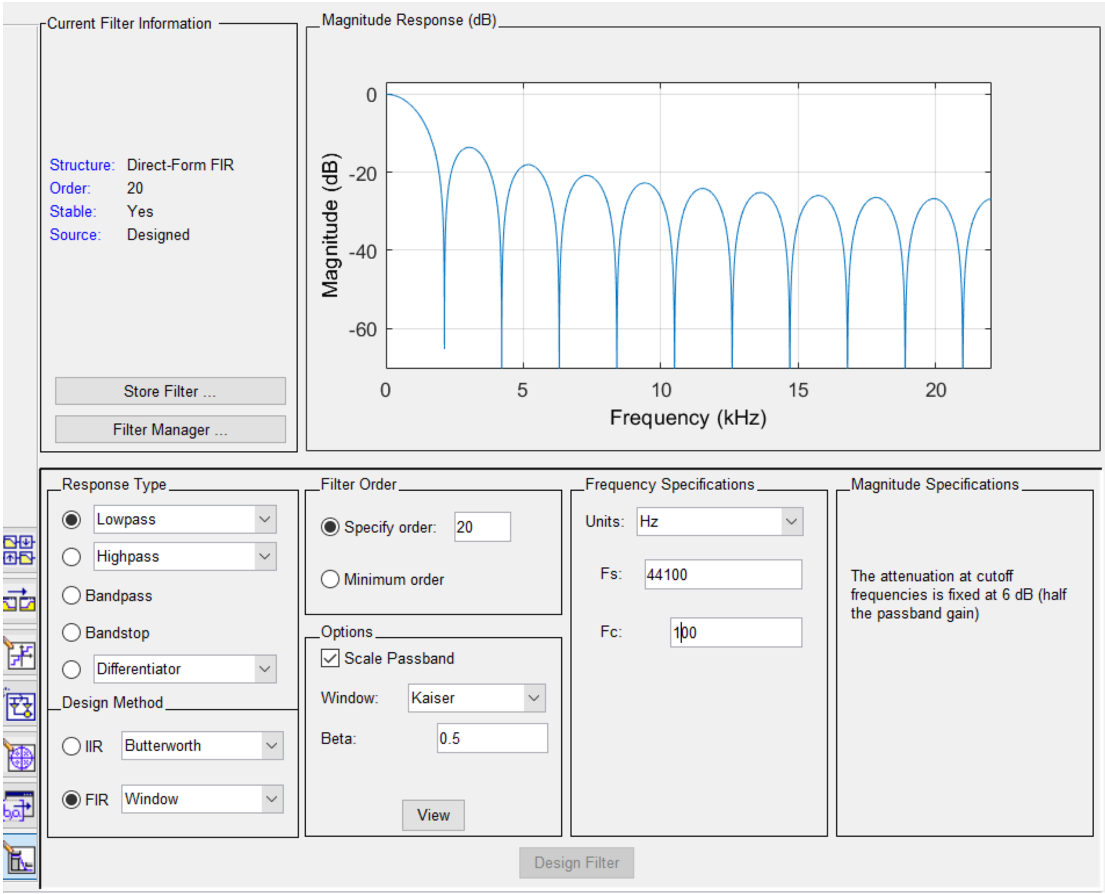
1. After listening to the noisy signal, we can't detect what it is exactly, or what instrument is played.
2. The frequency of the spurious harmonic: 6KHZ The noise is high-pitched



Since we have 2 mixes sounds the fundamental of the low-pitched sound (the music sound) :100HZ



We design a low-pass filter as the picture shows:



After implementing the filter, we finish with a smooth guitar sound.

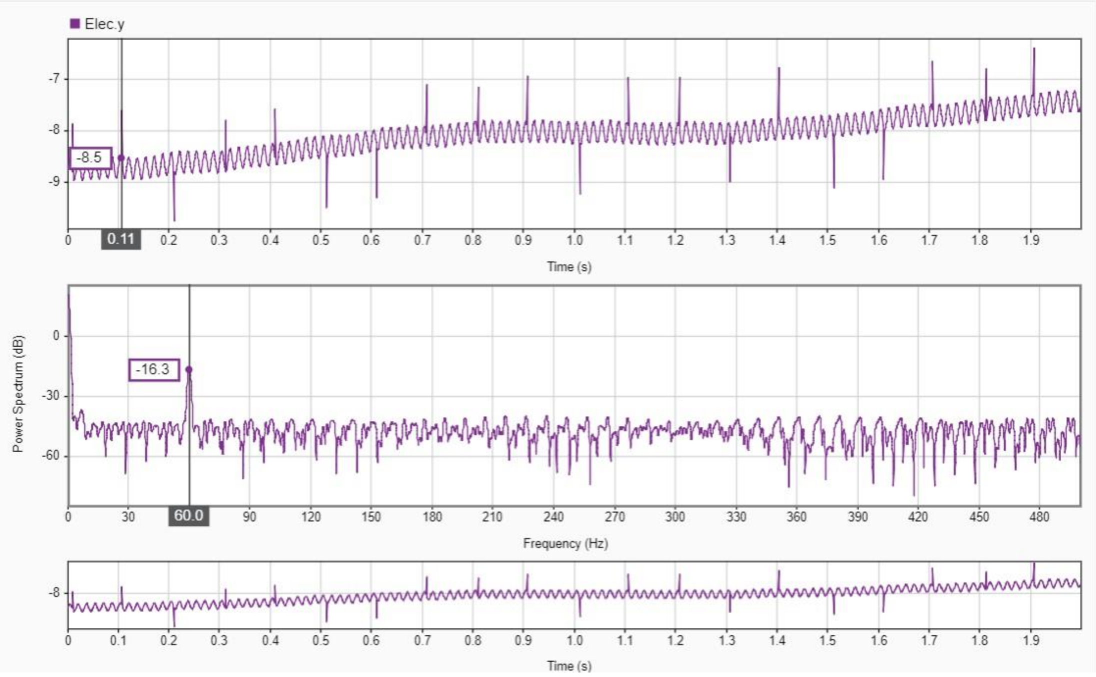
We can do the same thing with the following script:

```
sound(y,fs);
music = timetable(seconds((0:length(y)-1)'/fs),y);
yfiltb = lowpass(y,100,fs);
xf = filter(Num1,1,y);
```

Spiky electrical voltage:

In this exercise, we have a file called electrical_voltage.mat which contains a voltage signal stored in y and sampled at fs that is contaminated by unwanted spikes. The goal of this exercise is to get rid of the spikes using a median filter.

1. The plot of the electrical signal:
   We can see that we have a spike every second.

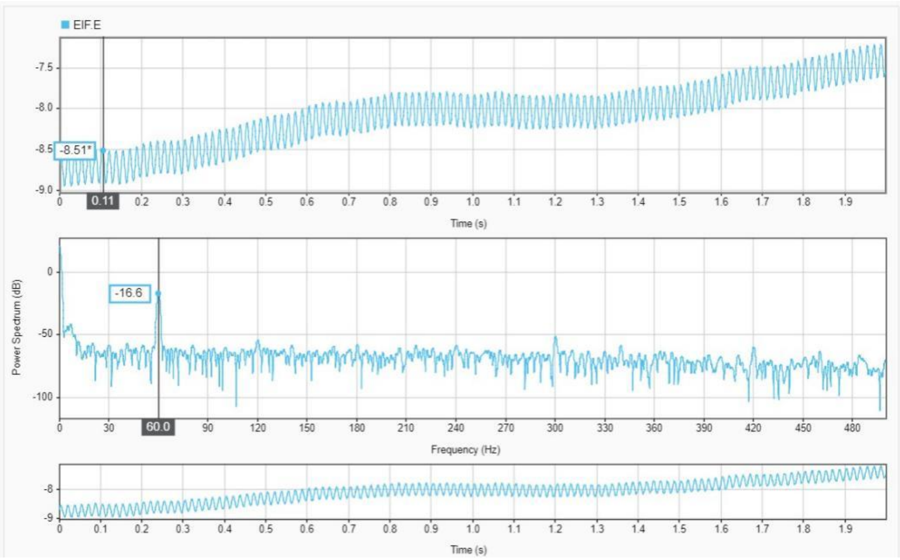


2. The implementation of a median filter to cancel out the unwanted spikes on the electrical signal:

Using a script:

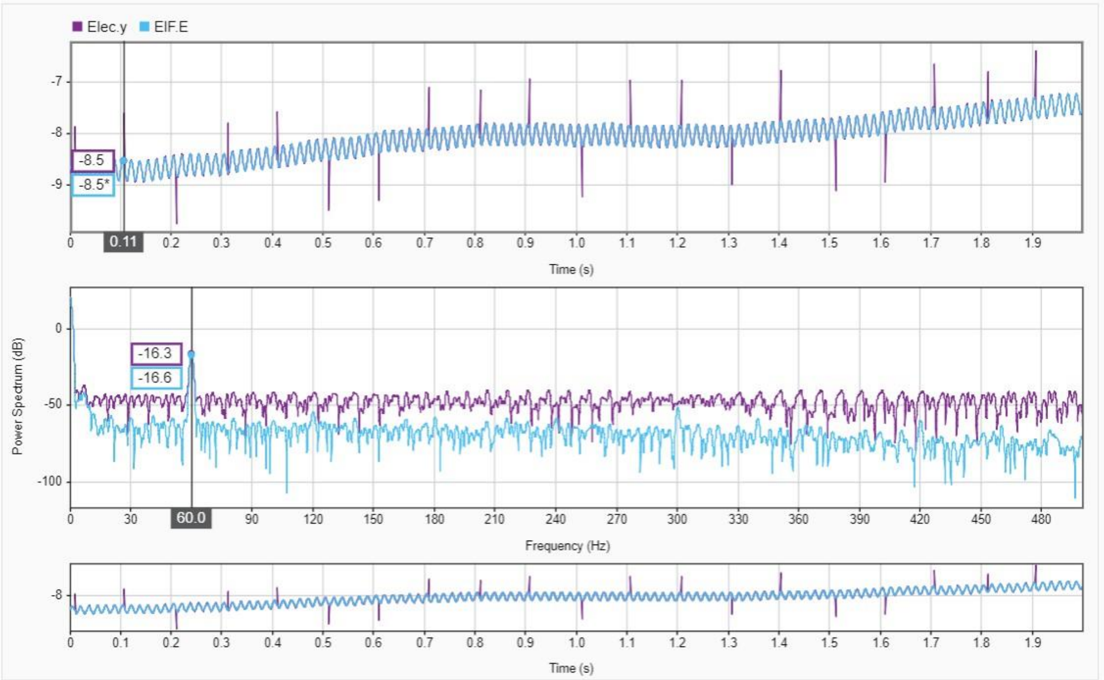
```
load electrical_voltage;
sound(y,fs);
Elec = timetable(seconds((0:length(y)-1)'/fs),y);
E = medfilt1(y);
ElF = timetable(seconds((0:length(y)-1)'/fs),E);
```

The result:



Using filter designer:

We didn't find how to design a median filter on the filter designer, so we only worked with the script, and as it's mentioned bellow, we got rid of all the spikes.



3. The fundamental frequency of the cleaned electrical signal is 60HZ.
4. The signal was recorded in the US but not in France, because the frequency is equal to 60 which is the electricity frequency of America while in France and the EU is 50 HZ (source: Wikipedia).

Conclusion:

Digital filter are used a lot in our everyday life, and their goal is to get rid of any unwanted noise that may damage the information that the signal hold.