

Laboration 2 - Implementation av digitala filter

Simon Berthilsson

January 28, 2015

You just heard of very interesting conference on digital filtering that you really want to go to. The problem is that it is a 17 hour flight away and your schedule will only permit you to take the flight that lands a few hours before the conference starts. This all means that you need to walk off that flight well rested. Unfortunately, air plane noise makes it hard for you to sleep so you need to find a way of taking the noise out of the flight. Using regular hearing protection would work but you are a bit apprehensive about missing out on important information delivered by the flight attendants. Luckily, you have come up with a solution; to modify a pair of regular hearing protection with a set of microphones and some filtering hardware that filters out the noise but relays any information to a pair of speakers that you mounted inside the ear cups. In order to design this filter, you found a recording of a flight attendant at a site called freesound.org uploaded by a user called 303creative. By analysing a piece of the recording where the flight attendant was not speaking you find that the aircraft noise has the characteristics shown in Figure 1

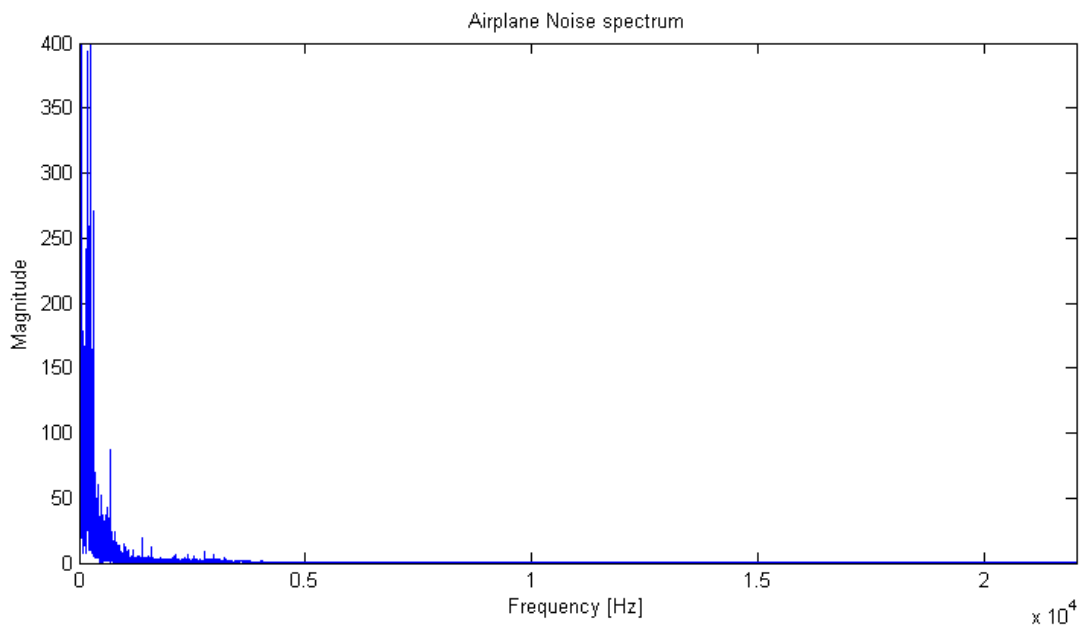


Figure 1: Your typical aircraft noise spectrum

Your task is to design and implement a high pass IIR filter on a realization of your choice that blocks the aircraft noise and lets through any information the aircraft personnel may give you.

A friendly advice; a filter with a less extreme cut off frequency (say, in the middle of the allowed frequency region) is often easier to implement due to it having lower gains and needing less extreme

scaling. It may be a good idea to get a "easy" filter running first so that you know that your code works.

Your tasks:

1. Design a desired filter characteristic
2. Design a filter of order four matching that characteristic, using 'ellip' in MATLAB. protip: use the command 'format long' in order to make the MATLAB prompt round less
3. Verify that the filter you designed matches your intentions by using the script 'labBode.m' found on studentportalen
4. Recast the filter into a realization form of your choice
5. Implement the filter and verify that it matches your design goals
6. Write a report detailing all steps in the procedure. Take extra care to include the filter design steps.

This report should:

- (a) Be written in correct, understandable Swedish or English.
- (b) Describe what you have done and why you have chosen to do so (from an implementation point of view, *not* 'I chose to do this because I was told to.').
- (c) Show how well the filter matches the characteristics you chose in step 1.
- (d) Contain a discussion about any deviations from the goals you set up in step 1.
- (e) Contain the vital parts of the c-code.
- (f) Support any claims that you make by figures or references. I will post Figure 1 on studentportalen and you may use this figure in the report without attribution.
- (g) Be written in the form of a report, not in the form of answers to a set of questions.
- (h) Be in my inbox no later than Wednesday Feb.11, 23:59.

I will, upon failure to comply with items a–g, comment your report and return it to you for corrections.

If you do not hand the report in before the deadline expires, I *may* still process your report, but no guarantees are given as to when I may do so. In other words, if you need your hsp, **hand the report in on time.**