COMP.SGN.100 Introduction to Signal Processing,
Exercise 11, 11.-12.10.2021

Pen & paper task solutions should be submitted to Moodle at least one hour before your exercise session. Matlab tasks are done during the exercise session.

Task 1. *(Pen & paper)* The signal $x(n)$ with the sampling rate 12 kHz should be converted to a signal with the sampling rate 15 kHz. Determine the steps of the conversion as a block diagram using resampling ($\boxed{\uparrow L}$ and $\boxed{\downarrow M}$) and low-pass filtering ($\boxed{H(z)}$). Specify the passband and stopband intervals of the required low-pass filters, when the frequencies on the interval $0 - 5.5$ kHz are to be preserved.

Task 2. *(Pen & paper)* The goal is to reduce the sampling rate of a 12 kHz signal to one kilohertz in several stages. The most important information in the signal is on the frequency interval $0 - 450$ Hz, which will therefore be strictly preserved. Find out the total amount of the coefficients of the required filters in each multistage implementation (4 cases are enough to investigate, i.e. only the cases where the decimation coefficients are in descending order) when the low-pass filters are designed with the Hamming window ($N = 3.3/\Delta f$).

Task 3. *(Pen & paper)* How many multiplications per second (MPS) are needed in each of the implementations in Task 2? This is calculated using the formula

$$\text{MPS} = \sum_{i=1}^{I} N_i F_i,$$

where I is the number of decimation blocks, N_i is number of coefficients in the i^{th} block, and F_i is the sampling rate of the input signal of the i^{th} block.

Task 4. *(Matlab)* We will implement in this and the next task the sampling rate conversion in Matlab with factor $\frac{2}{3}$. First load Matlab's test signal `laughter` (with `load laughter` to the variable `y`). The sampling frequency of the signal is 8192 Hz.

In this task we double the sampling rate of the signal. This requires the following steps:

- Create a zero signal `z` with length twice the one of the original signal.
- Use the command `z(1:2:end) = y;` to assing values of the original signal to every second element of the zero signal. Listen to the result. Note that the command `soundsc` needs also the sampling rate as an input. Otherwise, the result sounds incorrect.
- Design e.g. an FIR filter of order 100 (command `fir1`) and filter the generated interference.
- Listen to the filtering result and check that it sounds the same as the original signal.
- Plot the spectrograms of all three signals with the command `spectrogram`.

Task 5. *(Matlab)* Now we reduce the sampling rate of the output signal of Task 4 to one third as follows:

- Design an antialiasing filter of order 100.
- Filter the signal `z` by the filter.
- Take every third sample of the result.
- Listen to the result at the sampling rate of 5461 Hz, which is about 2/3 of the original sampling rate.
- Plot the spectrograms of all three signals with the command `spectrogram`.