

**T-61.3010 Digital Signal Processing and Filtering**

(v. 1.0, 13.3.2009), Matlab #6 (27.4.-5.5.2009)

**Registration** in WebOodi. Bring your own **headphones** if you have. The assistant will guide you through the exercises, but you may go on your own speed. Feel free to ask the assistant, if you have troubles. You can also consult [http://www.cis.hut.fi/Opinnot/T-61.3010/how\\_to\\_start\\_with\\_matlab.shtml](http://www.cis.hut.fi/Opinnot/T-61.3010/how_to_start_with_matlab.shtml) or [kuinka\\_aloitan\\_matlabin.shtml](http://www.cis.hut.fi/Opinnot/T-61.3010/kuinka_aloitan_matlabin.shtml).

Getting started: In **Windows** just click **Programs - Matlab**. Write down the code into separate files in your working directory (e.g. `Z:\DSP\`) for future use. Set the "Current Directory" in Matlab to point to the working directory (or type `cd <workdir>`).

The problems marked with [Pxx] are from the course exercise material (Spring 2009).

**In the end** of this session **you should know**: (a) that Matlab is a numerical software with finite accuracy of numbers (as all digital devices), (b) effects of truncating coefficients in a filter structure, (c) how upsampling and downsampling affect signal and its spectrum.

1. [M5000] **Task**: Calculate the following sums and roots first by hand and then with Matlab, and explain shortly the results.

- a)  $S = 1 - 0.3 - 0.1 - 0.1 - 0.2 - 0.2 - 0.1$
- b)  $S = 1 - 0.3 - 0.1 - 0.1 - 0.1 - 0.2 - 0.2$
- c)  $S = 1 - (0.3 + 0.1 + 0.1 + 0.2 + 0.2 + 0.1)$
- d) Find roots of  $H(z) = z - (-1) = z + 1 \Rightarrow z = -1$ , using Matlab `roots([1 1])`
- e) Find roots of  $H(z) = (z - (-1)) \cdot (z - (-1)) = z^2 + 2z + 1 \Rightarrow z_{1,2} = -1$ , using Matlab `roots([1 2 1])`
- f) Find roots of  $H(z) = (z - (-1))^4 = z^4 + 4z^3 + 6z^2 + 4z + 1 \Rightarrow z_{1,2,3,4} = -1$ , using Matlab `roots([1 4 6 4 1])`. Connection to lowpass Butterworth filters: `[B, A] = butter(4, 0.4); freqz(B,A); zplane(B,A); roots(B)`

2. [M5009] Consider a fourth order filter  $H(z)$

$$H(z) = K \cdot \frac{b_0 + b_1z^{-1} + b_2z^{-2} + b_3z^{-3} + b_4z^{-4}}{a_0 + a_1z^{-1} + a_2z^{-2} + a_3z^{-3} + a_4z^{-4}}$$

in Direct Form I as seen in Figure 1(a). If we are dealing with infinite wordlength, coefficients can have values such as  $b_0 = \pi$ . However, when implementing filters either in hardware or software (fixed-point), there will be a finite number of bits for representing each number, e.g.  $b'_0 = 3.125 = 3 \cdot 2^0 + 1 \cdot 2^{-3}$ . The change of coefficients ( $b_0 \rightarrow b'_0$ ) affects the computation of magnitude response as well as the locations of the roots in the pole-zero plot ( $ax^2 + bx + c = a \cdot (x - d_1) \cdot (x - d_2)$ ).

Filters can be realized as a cascade of second-order systems (SOS). Figure 1(b) shows a structure for a fourth order filter in this case,

$$H(z) = G \cdot H_1(z) \cdot H_2(z) = G \cdot \frac{b_1 + b_2z^{-1} + b_3z^{-2}}{1 + a_2z^{-1} + a_3z^{-2}} \cdot \frac{b_4 + b_5z^{-1} + b_6z^{-2}}{1 + a_5z^{-1} + a_6z^{-2}}$$

**Task**: Examine the filter

$$H(z) = \frac{0.5806 - 1.8325z^{-1} + 2.6023z^{-2} - 1.8325z^{-3} + 0.5806z^{-4}}{1 - 2.3933z^{-1} + 2.5562z^{-2} - 1.4889z^{-3} + 0.4302z^{-4}}$$

Compare behavior of the filter (i) with no quantization, (ii) quantizing coefficients of the direct form (DF), and (iii) quantizing coefficients of second-order systems (SOS). You can use m-files `D11quan.m` and `a2dR.m` from the course web site.

- a) Plot the pole-zero-diagram, amplitude response and start of the impulse response (`impz`) using the original fourth order direct form filter without any quantization (i). What kind of filter is it?
- b) Compute the coefficients of the second order implementation of Figure 1(b) (`tf2sos`). A direct form filter can be created from SOS using (`sos2tf`).
- c) Download the function `a2dR` from the course web site (function by Mitra). Using `a2dR` quantize coefficients both in direct form (Figure 1(a)) and in second order systems (Figure 1(b)). Decrease the number of bits from 16 step by step. One example is given in Figure 2.

Plot the location of poles and zeros for all cases. Plot the amplitude response for all cases. When do you find significant difference between (i), (ii) and (iii)? Any problems with stability?

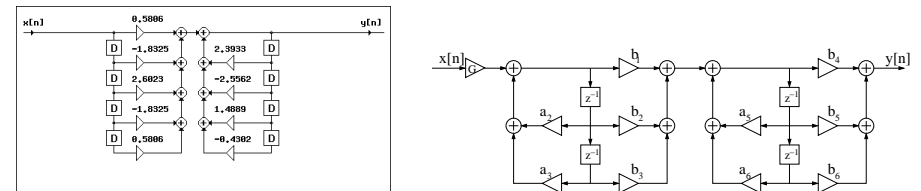


Figure 1: Problem 2: Filter  $H(z)$  in (a) Direct Form I, (b) Second order systems (SOS) in cascade.

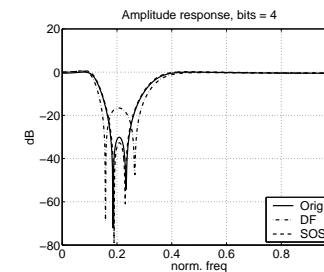


Figure 2: Problem 2: Example with 4 bits.

3. [M6002] Open the demo `M6002.m` about resampling an audio file. There are four steps and four windows in the demo. You can set the sampling factors  $L$  and  $M$  in the beginning of m-file.

**Task**: What is the sampling frequency in Figure 1? What is the frequency value in Hertz corresponding  $\omega = \pi/4$ ?

The signal is upsampled by factor  $L$ . Can you find  $L - 1$  zeros between each original sample in the waveform (top) in Figure 2? Is the spectrum just like given in the theory?

Next, the signal is lowpass filtered. See Figure 3, how does this affect in the signal waveform?

Finally, the signal is downsampled by factor  $M$ . You can also listen to the final signal (if the final sampling frequency is less than 44 kHz) directly from Matlab or writing it to a WAV file by `wavwrite(y, fsFinal, 8, 'yFinal.wav');`