Laboratory 6a:

**System Output   
via**

**Convolution with Impulse Response**

In this laboratory you experiment calculating the output of linear time invariant systems in the time domain for arbitrary input signals using the convolution with the impulse response of the system. This method is often used in audio applications to implement the “reverb” effect.

**Exercise 1** *System Comparison*

Consider the following Matlab code template, describing the impulse response of three LTI systems: Sys1, Sys2 and Sys3

% SiSy \_ Exercise Sys Comparison

% Convolution with Impulse Response

% ===================

clear all, close all, clc

% PARAMETERS

R = 2.2e3; C = 220e-6; RC = R\*C;

tstep = RC/50;

t = 0:tstep:10\*RC;

L = length(t);

% SYSTEM DEFINITION

% Sys1

yd1\_t = (1/RC) \* exp(-t/RC);

% Sys2

del\_tstep = [1/tstep zeros(1,L-1)];

yd2\_t = ((-1/RC) \* exp(-t/RC)) + (del\_tstep);

% Sys3

yd3\_t = yd1\_t .\* cos(2\*pi\*7/3\*t);

1. Plot in Matlab the impulse responses and the step responses of systems 1, 2 and 3.

*Hint:* you can approximate an integral calculation in Matlab using: *cumsum(…)\*tstep*.

1. Calculate via convolution with the impulse response the output of systems 1, 2 and 3 for a periodic square function u(t) with period Tsig = 3, and amplitude varying between ±1.

*Hints:*

* you can declare in Matlab a periodic square function as: *square(2\*pi\*fsig\*t)*
* use *conv(…)\*tstep* in Matlab to calculate the convolution (*Faltung*)
* check how long is the output of the *conv()* operation, and declare a t\_long vector that you can use to plot the three system responses

1. Which frequency behaviour (LPF, HPF, BPF) do you observe by these 3 systems?
2. Add the following lines to your Matlab code, and explain what these lines are calculating and plotting:

% ??? Responses

y1\_f = (1/L)\*fft(yd1\_t);

y2\_f = (1/L)\*fft(yd2\_t);

y3\_f = (1/L)\*fft(yd3\_t);

u\_f = (1/L)\*fft(u\_t);

Fs = 1/tstep;

fstep = Fs/L;

f = fstep\*[0:1:L-1];

% PLOTS

figure(3) % ???? Responses: log scales

subplot(131),semilogx(f,db(y1\_f),'b',f,db(u\_f),'k:'),grid on,

xlim([0 Fs/4]),ylim([-50 -10]),xlabel('f')

subplot(132),semilogx(f,db(y2\_f),'r',f,db(u\_f),'k:'),grid on,

xlim([0 Fs/4]),ylim([-50 -10]),xlabel('f')

subplot(133),semilogx(f,db(y3\_f),'g',f,db(u\_f),'k:'),grid on,

xlim([0 Fs/4]),ylim([-50 -10]),xlabel('f')

figure(4) % ???? Responses: linear scales

subplot(131),plot(f,abs(y1\_f),'b',f,abs(u\_f),'k:'),grid on,

xlim([0 Fs/4]),ylim([0 0.25]),xlabel('f')

subplot(132),plot(f,abs(y2\_f),'r',f,abs(u\_f),'k:'),grid on,

xlim([0 Fs/4]),ylim([0 0.25]),xlabel('f')

subplot(133),plot(f,abs(y3\_f),'g',f,abs(u\_f),'k:'),grid on,

xlim([0 Fs/4]),ylim([0 0.25]),xlabel('f')

1. What is the center frequency of the BPF? How was the impulse response of the BPF obtained? Compare this approach with a property of the Fourier Transformation.
2. Which harmonic of the periodic square input is coming through the BPF? How could you change the impulse response of the BPF to have the 5th harmonic of the periodic square visible at the output of the filter?

**Exercise 2** *Convolution Reverb*

The audio effect “convolution reverb” consists of adding reverberation or echo to a sound sample by convolving it with a previously recorded (or simulated) impulse response. This impulse response describes or models a physical or virtual space.

Let us test and hear the results of this effect in Matlab. Use the recorded impulse response of a church provided in the file:  *SMALL\_CHURCH\_E001\_downsampled.wav*

Calculate the convolution with an input audio sample (for example *audio\_input1.wav* ) in order to generate an output with the reverberation effect.

Input

Audio Data

*Discrete input signal*

LTD System

*Space with known impulse response*

Output

Audio Data

*Discrete output signal*

**Observation:**

The abbreviation LTD stands for “linear time-invariant and discrete” system. In this semester we treat the impulse response of an LTD as an LTI impulse response which was sampled with a known Fs.

**Hints:**

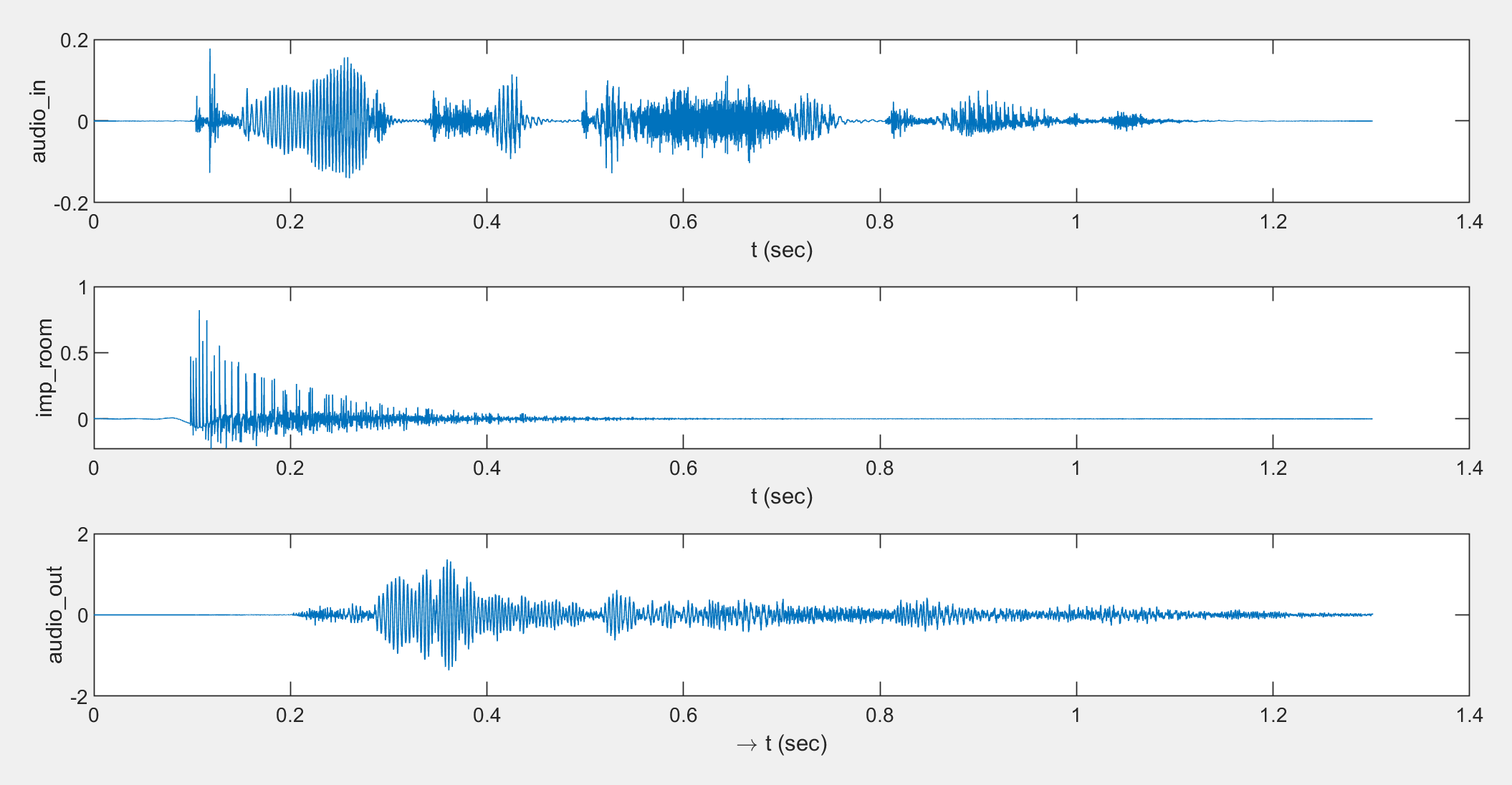
* Start a new Matlab script, with the usual first commands *clear all, close all, clc*;
* Read in both audio data files in Matlab using the command *audioread()* .

Check the syntax of this command with the help function.

* Calculate the output signal with the discrete convolution (command *conv()* in Matlab).

Obs.: No need to multiply with tstep ( as *conv(…)\*tstep* ) because we treat the convolution output here as a discrete signal

* Generate a plot of the three audio signals (audio input, impulse response and audio output signal) in the time domain.   
  Define the required time vectors, and use the axis command to set an adequate time interval (e.g. start with the same time interval for the three signals).



* Hear to the input and output signals via the PC sound card and headphones with the commands *audioplayer()* and *play()* in Matlab.   
  Check the syntax of these commands with the help function.   
  Hint: you can use the command *pause()* after the *play()* to have an interval between the 2 audio samples.
* If you still have time you can try further audio input signals and further impulse responses. The impulse response we used here, and further examples can be found under the link: <http://www.cksde.com/p_6_250.htm>