Laboratory 6a:

**System Output   
via**

**Convolution with Impulse Response**

In this laboratory you calculate the output of several linear time invariant systems in the time domain for any arbitrary input signal using the convolution with the impulse response of the system. This method is often used in audio applications to cause “convolution reverb” effect.

**Exercise 1** *System Comparison*

Please read the complete exercise before you start to solve part (a).

The impulse response of 3 different analog systems is given below (in Matlab syntax):

% 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\*3\*t);

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

Hint-1: you can approximate an integral calculation in Matlab using the function *cumsum()*.

1. Calculate via convolution with the impulse response the output of systems 1, 2 and 3 for a periodic square function.

Hint-2: you can declare in Matlab a periodic square function as: *square(2\*pi\*fsig\*t)*

Hint-3: use the function *conv()* in Matlab to calculate the convolution (*Faltung*). For example:

y1\_t = tstep\*conv(u\_t,yd1\_t);

Hint-4: It is proposed to use the following constants and parameters:

% Parameters

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

tstep = RC/50;

t = 0:tstep:10\*RC;

L = length(t);

% Input Signal

Tsig = 3;

u\_t = 0.5\*(1 + square(2\*pi\*t/Tsig));

Hint-5: 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?

Hint-6:

LPF – low pass filter, let low frequencies through and attenuates high frequencies

HPF – high pass filter, let high frequencies through and attenuates low frequencies

BPF – band pass filter, let frequencies inside band through (a range of frequencies around a center frequency) and attenuates other frequencies (below and above selected range).

**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*

**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).
* 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>