ECSE 436 **McGill University**

Prof. Bajcsy **Winter 2018**

Laboratory Assignment 2 – part 2

Due: with part 3

Reading: Lecture and lab/tutorial notes, handouts on Altera FPGA signal processing boards.

3. Speech Processing and Acoustics: Convolution and the Impulse Response

An overall effect of a room on any given sound signal *x[n]* can be modeled as a linear time-invariant system and can be completely described by the room's impulse response *h[n].* As mentioned in class, you can measure impulse response of any specific room by producing a very short sound-burst (an impulse such as a gun-shot or a clap) at the source and then measuring the resulting signal at the microphone which picks up not only the direct line-of-sight contribution from the sound source, but also echoes produced by reflections off walls and room objects. File "h.wav" contains the impulse responses h[n] for a specific room and source/microphone setup.

(a) Using Matlab, plot the impulse response for the channel and explain how the delay of the channel (i.e., the time the sound to travels from source to microphone) can be computed by looking at the impulse response plot. What is the distance of the microphone from the source of impulse ?

(b) The file "speech.wav" is a voice recording from an *anechoic* chamber (a studio with little or no echo), with the microphone very close to the voice. Convolve this recording with the impulse response h[n] and listen to the result before and after convolution. Explain what you hear.

(c) Repeat part (b) for an impulse response h'[n] = h[n] + h[n-3000] where the delay is in number samples. Again, explain what you hear.

HINTS: In Matlab, x=audioread('speech.wav') will read audio file named "speech.wav" into the vector "x". Because a computer can only store a discrete sequence of numbers, each file is a vector $v[n]$ sampling the original response $h(t)$ so that $v[n] \triangleq h(nT)$ for indices $n = 0,1,2$ etc. In this experiment the sampling rate for all recordings is $T \triangleq 1/16000$ seconds. Convolution can be implemented with the "conv" command and you can listen to any vector of data by using the command sound(v,16000)which plays back the vector $v[n]$ at a rate of 16000 samples per second.

(d) Describe how you can play the signal backwards in time and discuss what you hear.

(e) Play the signal at different playback speeds (13000, 14500, 17000, 18500, 20000) and describe how the signal quality/intelligibility changes.

(f) Observe the effect of aliasing - sub-sample the signal (2:1, 3:1, 4:1, 5:1 and 10:1), then play it back appropriately. Describe what you hear and sketch an appropriate signal spectrum plot, illustrating what is happening.

(g) Quantization is another signal processing operation that is performed (e.g., to save on data storage needs) but it can affect significantly quality of the play-back signal. Quantize your signal using appropriate uniform quantizer that has 16, 8, 4, 2 and 1 bits. Play the quantized signals back and discuss the impact of quantization on the quality of the sound.

4. Perception of Audio Signals

(a) Using Matlab, determine (approximately) the maximum and minimum positive frequency each member of your group members can hear. Describe how you did this and submit the actual values of cut-off frequencies. What happens as you approach these cut-off frequencies?

(b) Generate and play back additive *Gaussian* noise signal using 'randn' function in Matlab. Play it back using different amplitude scaling (power) and describe what you hear.

(c) Generate and play back additive random *impulsive* noise signal using Matlab. Describe your implementation and what you hear when you play it back using different amplitude scaling (power) and probability of the impulses.

(d) Corrupt the audio waveform from question 3 with additive Gaussian and Impulsive noise, respectively. Describe what you hear at SNR = -10 dB, 0 dB, 10 dB, 20 dB and 40 dB.

5. Real-time Audio Processing on FPGA Signal Processing Boards

Redo the following cases from the Question 3 in real time. In each of the following cases, submit all implemented scripts, describe what you observed and demonstrate your implementation.

(a) Implement a real-time system to play back audio quantized to 8, 4, 3, 2 1 bits per sample.

(b) To demonstrate the effect of aliasing, replay the incoming signal sub-sampled by a factor 2:1, 4:1 and 8:1. (Hint: Use zero-order hold approach OR try to change the output sampling rate.)

(d) Implement a module generating additive Gaussian noise samples using addition of 6 (or 12) independent, uniformly generated samples from interval [-1,1]. (Hint: Use provided uniform generator module.)

(e) Using histogram method in Matlab, verify that the noise generated by the approach in part (d) is in fact *close* to being Gaussian distributed.

(f) What is the exact mean and variance of noise from part **(d)**? What is the reason that the generated noise is near-Gaussian distributed, even though it comes from uniform noise samples? When can the used the Gaussian approximation become a serious problem in practice? Fully justify your individual answers.

(g) Replay a real-time audio signal corrupted by Gaussian noise at SNR = 40 dB, 30 dB, 20 dB, 10 dB, 0 dB and -10 dB. Describe how you adjusted the noise power and what you hear in each case.