ECSE 436 **Laboratory Assignment 3 part1** **McGill University**

Winter 2018 Due: with part 2 **Prof. Jan Bajcsy**

0. Random noise generation on an FPGA board (revisited)

Implementing a digital module generating additive Gaussian noise samples can be done using addition of 6 (or 12 or more) independent, uniformly generated samples from interval [-1,1].

(a) Using histogram and *rand* functions in Matlab, verify that noise generated by the above approach is in fact close to being Gaussian distributed.

(b) Determine the exact mean and variance of the noise above. What is the reason that the generated noise is near-Gaussian distributed, even though it comes from uniform noise samples? Justify your answer.

(c) When can the used Gaussian approximation become a serious problem in practice? Give an example when this is the case.

(d) Use the provided uniform generator module to generate Gaussian noise samples in your FPGA.

(e) Replay a real-time audio signal while corrupting it by Gaussian noise at SNR about 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. Demo your implementation to the TA.

1. **More Advanced Audio Signal Processing**

(a) Using FFT and IFFT routines, low-pass filter the provided speech file to it to 4000 Hz, 2000 Hz, 1000 Hz and 500 Hz. Play the filtered audio signal back then describe what you hear and why.

(b) Corrupt the original speech signal with additive Gaussian noise generated using Matlab's *randn* function so the signal to noise ratio is 40, 30, 20, 10, 0, -10 and -20 decibels. Redo part (a) with these noisy signals and describe what you hear. Discuss why filtering helps and quantify (using SNR) how much does the low pass filtering of the noisy signal improves its quality. When does the filtering become detrimental to the quality?

(c) Corrupt the original speech signal with a jamming signal discussed in class to prevent listening to radio broadcast. Generate and add a 400 Hz sinusoid with large amplitude to the original audio signal. Listen to the jammed signal and describe what you hear depending on the amplitude of the jamming signal. At what SIR is the signal difficult to understand?

(d) Design a system that processes the jammed speech signal from part (c) and makes it intelligible again. Describe your design and submit your implementation script.

2. FPGA board implementation of a Viterbi Decoder

Implement the Viterbi Decoding Algorithm for the Hamming (8,4,4) Code Hamming using Cyclone FPGA board.

(a) Start with implementing decoding for one noisy codeword using trellis diagram of this code and the ADD-COMPARE-STORE routine.

(b) Simplify the edge metric calculation so the overall decoding of a codeword takes less than 30 floating point operations (FLOPS include additions, substractions, comparisons or multiplies of 2 real numbers, etc.)

(c) Then extend your Verilog/VHDL implementation into real-time for a stream of packets, that are generated on the board and corrupted using the provided Gaussian noise module.