Lab 1 - Introduction to Digital Signals using MATLAB

Section	L01	L02	L03	L04
Lab Date	Sep. 25	Sep. 18	Sep. 26	Sep. 19
Due Date for Report	Oct. 1	Oct. 1	Oct. 1	Oct. 1

Assessment: 5% of the total course mark.

Objectives:

- To gain experience in creating, sampling, quantizing and analyzing digital signals within `MATLAB`
- To become more familiar with `MATLAB` programming

Assessment:

- Your grade for this lab will be based on your ability to create and work with digital signals within `MATLAB`, and on your reporting of the results.
- Clearly label all plots and their axes (points for style will be deducted otherwise)
- Please attend the lab section to which you have been assigned
- This is an individual lab
- By the end of the lab session you must demonstrate to your TA the MATLAB code
- The electronic submission on Avenue to Learn of all the `MATLAB` source code and the report have to be submitted by 11:59 pm on Oct. 1, 2023.

Pre-lab:

- Carefully read through this lab description, so that you know what is required.
- Read through the lecture notes so that you know how to answer the questions.
- Familiarize yourself with the MATLAB commands that may be required for this lab – see the list at the end of this lab description for some hints.

Experiments:

1. **Point sampling of a sinusoid:**
   The general form of the continuous time sinusoid is:

$$x(t) = A\cos(2\pi ft + \phi)$$

   (a) Create a `MATLAB` script that point-samples $x(t)$ at a sampling frequency F_s of 100 Hz, for $t = 0$ s to 0.5 s. Using the `MATLAB` function `stem()`, plot the discrete-time sequence $x[n]$ versus sample number n obtained for each of the following sets of parameters:

	Amplitude: A	Frequency: f (Hz)	Phase: ϕ (rad)
A	5	10	0
B	5	25	0
C	5	40	0
D	5	60	0
E	5	40	$\frac{\pi}{2}$
F	5	60	$\frac{\pi}{2}$

(b) Visually determine the period of each of the discrete-time signals. Compare the discrete time signals' period and waveform with those of the continuous time signals. Explain the differences in the periods using your knowledge of the Nyquist sampling theorem.

(c) Explain the effect of the phase change in cases C to F.

2. **Working with unit impulses and unit steps**

   (a) Create the unit impulse signal $\delta[n-16]$ and the unit step signal $u[n-12]$ for samples $n = 1, 2, \ldots, 30$ and plot both using the function `stem()`.

   (b) Plot the difference signal $x_1[n] = u[n-14] - u[n-15]$. Interpret this signal as $\delta[n-k]$ and find the value of k from the plot.

   (c) Plot another difference signal $x_2[n] = u[n-9] - u[n-16]$. Determine the width of the resulting rectangular "pulse". Express this signal as a sum of unit impulses.

3. **Complex signals**

   Generate the following complex-valued discrete-time signal:

$$x[n] = Ae^{j\omega n}, \text{ for } n = 1, 2, \ldots, 40 \quad A = 1 \quad \omega = \frac{\pi}{10}$$

   (a) Plot $x[n]$ in the complex plane (i.e., $\Im(x[n])$ imaginary part versus $\Re(x[n])$ real part), using the `MATLAB` command `plot()`.

   (b) Extract the real and imaginary parts of signal using the functions `real()` and `imag()` and plot them versus the sample number n using `stem()`. Use `subplot(2,1,1)` and `subplot(2,1,2)` commands prior to each `stem()` command to create plots of real and imaginary parts placed on the same figure but different axes.

   (c) Generate another figure with separate stem plots of the magnitude and phase of $x[n]$ versus sample number n. Does the plot of the magnitude make sense? What is the slope of the phase versus sample number curve, and how does this relate to the parameters used to generate $x[n]$?

4. **Quantization of a speech signal**

   (a) Load the supplied speech signal into `MATLAB` using the command:
   `[y,fs] = audioread('defineit.wav');`

   (b) Plot the speech waveform `y` using the `plot()` command, and plot a histogram of amplitude values using the command `histogram(y,50)`. Describe the waveform and the shape of the histogram in terms of:

   - The number of bits at which the .wav file was quantized.
     The following command can be used to get the number of bits used.
     `info = audioinfo('defineit.wav');`
   - The typical probability density function of speech.

   (c) Listen to `y` using the `soundsc(y, fs)` command. Describe the sound quality.

(d) Write a `MATLAB` script for a 3-bit rounding uniform quantizer for the range [-1 1].

(e) Scale the speech signal `y` so that no saturation ("peak clipping") will occur when it is passed through the quantizer and full quantization range is used – call this new variable `y_scaled`. Hint: divide by the maximum amplitude.

(f) Quantize the scaled signal `y_scaled` (call the quantized signal `y3bit`) and repeat parts b and c above, but for the newly quantized signal, comparing the new results to the results from parts b and c. In addition, plot the waveform and histogram for the error `e = y_scaled - y3bits`, and describe this in terms of the standard statistical model of quantization noise.

(g) Rescale `y` to greatly exceed the maximum scale range of your 3-bit quantizer, such that substantial peak clipping will occur – call this new variable `y_pclip`. Quantize this rescaled signal (call it `y3bit_pclip`) and repeat part f above, but for the peak clipped quantized signal. Again, compare the new results to the previous results.

REPORT: The report should contain

- Any mathematical calculations or derivations carried out
- `MATLAB` plots of results with brief descriptions
- Answers to questions

You **do not** need to include the `MATLAB` code in the report. However, you have to submit the `MATLAB` code separately.

POTENTIALLY USEFUL `MATLAB` COMMANDS:

Note that this is not an exhaustive list! You are not required to incorporate all of these in your scripts.

help topic	helpwin	figure	plot	stem
histogram	subplot	hold on	xlabel	ylabel
legend	title	function	clear	close
clc	zeros	ones	cos	exp
abs	round	max	min	find
if	for	end	real	imag
angle	unwrap	phase	audioinfo	audioread
audiowrite	soundsc			