BIRZEIT UNIVERSITY

Faculty of Engineering and Technology

Electrical and Computer Engineering

ENCS4310 (DSP) Course Project Fall 2023/2024

Submission deadline: Sat. 20/1/2024

About the course project:

A team of two or three students should do this project. The best arrangement is to choose a division of the project so that each of you can work on separate but interlocking parts. Individual work is not accepted.

Learning teamwork is also one of the more general goals of this course, so team projects will pick up points for demonstrating a successful ability to work with others.

The projects will be graded based on a project report (of around 2-4 pages) as well as online short presentations and/or discussion (online).

Project submission must be done via Moodle only, but please use PDF format and **not** Word .DOCX files if at all possible, since I often have formatting problems with Word files.

Your report must have the following structure, using these section headings and using IEEE paper format [template can be found on Moodle]:

Introduction: A general description of the area of your project and why you're doing it.

Problem Specification: A clear technical description of the problem you're addressing. Formulating a general problem (e.g., transcribing music) into a well-defined technical goal (e.g., reporting a list of estimated fundamental periods at each time frame) is often the most important part of a project.

Data: What are the real-world and/or synthetic signals you are going to use to develop and evaluate your work?

Evaluation Criteria: How are you going to measure how well your project performs? The best criteria are objective, quantitative, and discriminatory. You want to be able to demonstrate and measure improvements in your system.

Approach: A description of how you went about trying to solve the problem. Sometimes you can make a nice project by contrasting two or more different approaches.

Results and Analysis: What happened when you evaluated your system using the data and criteria introduced above? What were the principal shortfalls? (This may require you to choose or synthesize data that will reveal these shortcomings.) Your analysis of what happened is one of the most important opportunities to display your command of signal processing concepts.

Development: If possible, you will come up with ideas about how to improve the shortcomings identified in the previous section, and then implement and evaluate them. Did they, in fact, help? Were there unexpected side-effects?

Conclusions: What did you learn from doing the project? What did you demonstrate about how to solve your problem?

References: Complete list of sources you used in completing your project, with explanations of what you got from each.

The reason for this somewhat arbitrary structure is simply to help you avoid some of the more problematic weaknesses I've seen in past years. If you're having trouble fitting your work into these sections, you should probably think more carefully about your project.

Project description:

This Project can be divided into two phases:

Phase One:

Design, Implement and test an English alphabet character voice-frequency encoder which represents every English character by a combination of three voice-band frequency components (low, middle and high).
For example, it encodes character 'a' by signal contains frequencies (100HZ, 1100HZ, 2500Hz). Therefore, given the frequency combination for each character, you should be able to encode any character to the corresponding signal contacting the corresponding frequencies (see table below).

Table 1.1 show you the frequencies (100HZ-3500Hz) for each character. Assume the duration of each character signal is around 40ms.

Table 1.1: Encoding frequencies for each English character

Character	Low frequency component	Middle frequency component	High frequency component
a	100	1100	2500
b	100	1100	3000
c	100	1100	3500
d	100	1300	2500
e	100	1300	3000
f	100	1300	3500
g	100	1500	2500
h	100	1500	3000
i	100	1500	3500
j	300	1100	2500
k	300	1100	3000
l	300	1100	3500
m	300	1300	2500
n	300	1300	3000
o	300	1300	3500
p	300	1500	2500
q	300	1500	3000
r	300	1500	3500
s	500	1100	2500
t	500	1100	3000
u	500	1100	3500
v	500	1300	2500
w	500	1300	3000
x	500	1300	3500
y	500	1500	2500
z	500	1500	3000
space	500	1500	3500

To sum up, in this phase you have to design the following:

1. **Implement the English character encoder** using the above description and specifications.
2. **Build a GUI (graphical user interface)** by which the user can encode any English string (sentence) and the system should generate the corresponding signal for the given sentence.
3. **Two choices are available for the generated signal**:
   - Play the generated signal so that the user can hear it.
   - Save the generated signal as a (.wav) audio file in the current working directory.

Phase Two:

In this phase you have to design, implement and test a decoder for the system in part one, which can recover the text string from the encoded multi-frequency signal. Simply, your system should take an audio file (.wav) as an input and recognizes the encoded string and display it on the GUI screen.
You must use the following two approaches to build your decoder:

- Use **frequency** analysis (e.g. Fourier transform) of the input signal to determine which frequencies that have the highest amplitudes in each **40ms** and decode the character from them.

- Use bandpass **Filters,** so that you design filters represent the given frequencies and pass the input signal through them to pick the frequencies that passes and ones that rejected by the filters, then determine the frequencies in each **40ms**.

To sum up, in this phase you have to design the following:

1. **Two systems**, each can decode any sequence of tones to the corresponding alphabetic English characters. One of the systems uses frequency analysis and the second built by using filters.
2. **Build a GUI** so that the user can upload any audio file and a **run** button to be able to convert the sequence of tones in the audio into corresponding characters and show the result on a text box.
3. **Sufficiently test your two systems** with different number of encoded strings with different length, and report the accuracy of your systems. Accuracy is the number of correctly recognized letters divided by the string length.

Evaluation:

For the second phase, We'll provide you with a set of encoded testing strings in the same way described in phase one. Each team has to use their decoder system to recognize the encoded text and submit it on the provided online form within a specified date.

Project Deliverables:

1- Mini-report as described above.

2- System implementation (source code) and a demonstration of each phase as described above.

You can use any programming language you prefer for implementing your project. However, I highly recommend MATLAB/Python because they have many useful functions.

Dr. Abualsoud Hanani
Dr. Ashraf Rimawi