DSP Project

DTMF decoding

This document provides an outline for a DSP project to be completed by the end of the course. You are expected to investigate, in detail, methods related to solving the problem. There is a substantial design element to the project, and a quantitative evaluation of the performance of the proposed methods must be performed and presented. You are to write up a comprehensive report (approximately 10 pages) describing your method and results. You may work in pairs if you so wish.

The Problem

Touch-tone telephones use a **dual tone multi frequency** (DTMF) scheme to encode key-presses as audio tones. Thus for every key pressed a unique combination of two distinct audio tones is created, with frequencies specified in the following table:

Freqs	1209 Hz	1336 Hz	1477 Hz
697 Hz	1	2	3
770 Hz	4	5	6
852 Hz	7	8	9
941 Hz	*	0	#

For example, when the digit 4 is pressed, tones at 770 Hz and 1209 Hz are generated and summed together.

The aim of this project is to investigate and implement methods for decoding a sequence of key presses, such a may occur when a complete telephone number is dialled.

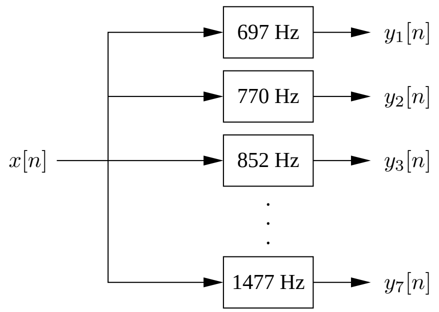
The task

There are many ways of solving this problem. Your task is to choose two possibilities, and investigate the relative merits of each. Some suggestions are provided later in this section.

Alternatively, you may choose one method and implement it in hardware. Your Motorola microprocessor boards are an ideal platform for the task — although not particularly high-performance, they should manage at the sampling rates required. If you choose to do an implementation, there will naturally be a decrease in the extent of the algorithm analysis expected from you.

The main component of a solution involves recognising the tones present in a portion of signal corresponding to a single keypress. Possible building blocks towards a solution to this problem are:

- **Bank of bandpass filters:** A bank of bandpass filters of the form



  may be applied to the input signal. When a key is being pressed, the magnitude of the output of two of these filters should increase. It should therefore be possible to decode a sequence of keypresses by looking at the energy of the output sequences over short time intervals.

- **Modulation and filtering:** An approach related to the bank of lowpass filters involves modulating the signal by varying amounts (related to the tone frequencies) and using a single bandpass (or lowpass) filter to produce the resulting energy signal.

- **Segmentation and spectrum analysis:** A simple way to use spectrum analysis to approach the problem is to start by segmenting the signal into periods corresponding to a key being pressed and no key being pressed. Each portion corresponding to a keypress can then be extracted and subjected to spectrum analysis. Since two frequencies are present each spectrum should contain two peaks, which can be detected and used to decode the sequence.

  The windowed DFT provides what is called a *periodogram* estimate of the data spectrum. The periodogram and its variations are the most commonly-used form of spectral estimates, but from a statistical perspective are quite poor. Model-based spectral estimates, such as the autoregressive (AR) spectral estimate, are better and more powerful. If you would like to explore one of these methods, which are actually quite simple, then let me know and we can work something out.

- **Spectrogram analysis:** A spectrogram is a short-time Fourier transform (STFT), useful for general time-frequency analysis. In the continuous world, a typical STFT of a signal $x(t)$ is given by
$$X(\Omega, \tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} x(t) w(t-\tau) e^{-j\Omega t} dt.$$
Here w defines a window which is usually symmetric about the origin, and has a limited (nonzero) support. A Gaussian function is often used. This window is moved across the data (according to the value of τ) and sets all but a small region of the signal to zero. The Fourier transform of the result is then calculated, which contains the frequency information in the vicinity of the time instant of interest.

Analysis

You are expected to provide insight into the problem and your method of solution. Some suggestions for questions that you might try to answer are the following:

- Why are the frequencies of the tones chosen as they are?
- How often does your algorithm correctly decode a digit, and correctly decode a sequence? Are there any digits that are regularly confused with one another?
- What is the tradeoff between the duration of the keypress and the accuracy of the recognition for your algorithm?
- What is the effect of adding noise of different power to the signal to be decoded?
- What is the effect of an interference (such as background speech) superimposed on the signal?

Good answers to these (and other) questions should provide both theoretical justifications for the behaviour as well as quantitiative evidence for the conclusion. One means of providing this evidence is to simulate the behaviour of the algorithm for ranges of different operating conditions and inputs, and obtain results for the corresponding performance. To do this will require a Matlab function to generate example tone-dial signals with different parameters, such as number sequence, tone duration, intertone spacing, and noise power.

A better (and more compelling, if time and resources permit) analysis involves recording actual real-world signals with a microphone and evaluating the performance

on real data. It is exceptionally important in any project to “enter” the real-world at some stage — a solution built, trained, and tested entirely in a computer is no solution at all (if you’re an engineer).

The intention here is to provide an objective evaluation of the strengths and weaknesses of your algorithm to a disinterested third party, in this case the reader of your report. If the methods you choose to evaluate your solution are not sensible in a practical situation, nobody will care about your results. Also, if your results are not believable, then nobody will believe them. The emphasis is therefore on an honest and critical assessment of the method.

Resources

- http://www.systolix.co.uk has a free filter design package called FilterExpress. You need to register to be able to see the filter outputs.