Contents

1 Dual-tone multi-frequency (DTFM) signaling

DTMF signaling is the way analog phones send the number dialed by a user over to the central phone office. This was in the day before all-digital networks and cell phones were the norm, but the method is still used for in-call option selection ("press 4 to talk to customer service"...).

The mechanism is rather clever: the phone's keypad is arranged in a $4x3$ grid and each button is associated to two frequencies according to this table:

	1209 Hz	336 Hz	477 Hz
697 Hz	1	2	3
770 Hz	4	5	6
852 Hz	7	8	9
941 Hz	*	0	#

The frequencies in the table have been chosen so that they are "coprime"; in other words, no frequency is a multiple of any other, which reduces the probability of erroneously detecting the received signals due to interference. When a button is pressed, the two corresponding frequencies are generated simultaneously and sent over the line. For instance, if the digit '1' is pressed, the generated signal will be:

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j\frac{2\pi}{N}nk},\ k = 0, 1, ..N-1$$

FFT Module

In Python, we will use the fft module in Numpy to compute the DFT

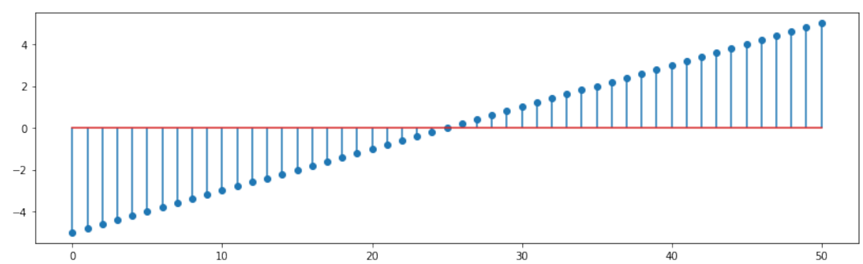
```
# First the usual bookkeeping
%matplotlib inline
import matplotlib
import matplotlib.pyplot as plt
```

```
import numpy as np
import IPython
from scipy.io.wavfile import write
```

```
# Sets the size of the output plots
plt.rcParams["figure.figsize"] = (14,4)
```

Typically, we will take a vector of data points, compute the DFT and plot the magnitude of the result. For instance, consider the DFT of a linear ramp:

```
x = np.arange(0, 10.2, 0.2) - 5
#draw figure
plt.stem(x);
```




```
X = np.fft.fft(x);
plt.stem(abs(X));
```