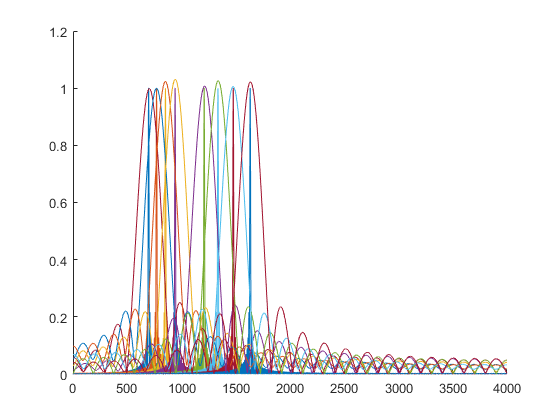
**Lab 8 and 9 Report**

**Encoding and Decoding Touch-Tone (DTMF) Signals**

**Kenneth Beartusk**

**4. Lab Exercise: DTMF Decoding**

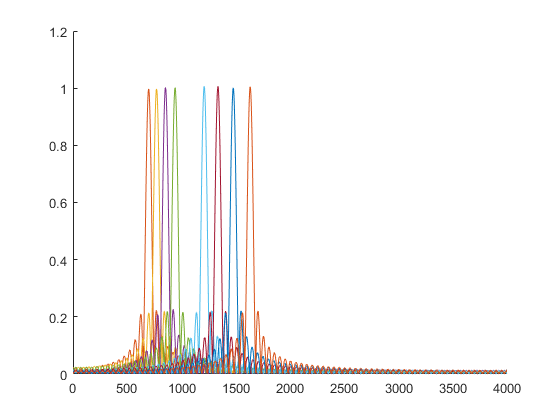
**4.1 Simple Bandpass Filter Design: dtmfdesign.m**

1. In order to get the frequency response to be at exactly 1, Beta was set to,
2. See MATLAB code.
3. See part f.
4. Below, are two graphs, the first one depicting the frequency responses of the 8 different band pass filters, and the second one depicting the peak magnitudes of each one of these filters.

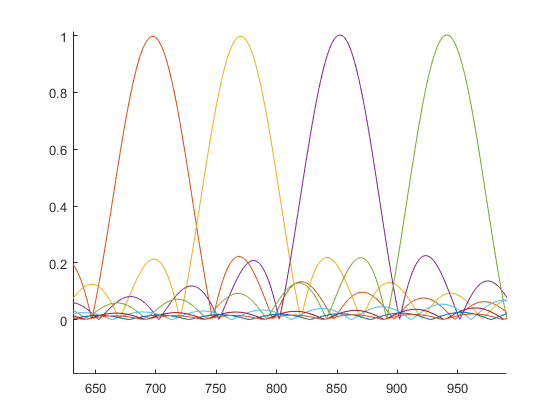
These measurements were obtained with the filter length being set to “L=40”. It is quite obvious that these filters have passband regions that intersect with eachother. This is apparent due to the fact that the passband region is where the magnitude is greater than or equal to or about 0.707. In order to separate the different dial tone frequencies, the passband widths cannot intersect. Below is zoomed up depiction of the frequency response shown above, and it shows these frequency responses intersecting at magnitudes of over 0.909.



1. For this portion of the lab, the 8 different band pass filters were again generated, but this time with a filter length of “L = 80”. Once again, The graphs below depict the frequency response and peak magnitudes.





From the plots shown above, it can be inferred that a filter length of 80 provides a much more precise filter with a narrower bandwidth. A zoomed in depiction of the frequency response shown above is shown below.

The lab handout states that the width of the passband is supposed to vary inversely with the filter length L. The filter length was doubled, therefore the bandwidth should be halved. Looking at the last filter response for both the filter lengths equaling 40 and 80, the bandwidth is about 175 and 44 respectively. This means that the bandwidth was reduced by a gain of ¼. This is more than what was expected.

1. In order to make sure that a frequency component lies within only one of the passband filters, the passbands of these filters must not intersect. According to the plots above, the filters passbands do not intersect. Although, the responses do intersect before the stopband is reached for some of the filters.

It seems that the hardest center frequency to meet these specifications for is the 770 Hz band pass filter. This filter is very close to the 697 Hz filter

**4.2 A Scoring Function: dtmfscore.m**

1. See MATLAB code.
2. See MATLAB code.
3. See MATLAB code.
4. I believe the reason that the magnitude of the filter response needs to be equal to one is to ensure that the magnitude of the output is the same as that of the input at the desired frequency.
5. The plot function was used in order to make sure the output signals were coming out correctly as a sinusoidal signal with magnitude of 1. It is commented out in the MATLAB script.

**4.3 DTMF Decode Function: dtmfrun.m**

See MATLAB code.

**4.4 Telephone Numbers**

Below is the code used to demo the *dtmfrun()* function. In the *dtmfrun()* function, the Key matrix from the *dtmfdial()* function was imported in order to identify which keys were pressed. Using, a for loop and some logic, the coordinates within the Key matrix were identified. The coordinates of course correlate with the different dial tone frequencies that each Key pressed will consist of.

