Cochlear Implant Project:
Phase 2: Filter Design

October 9, 2024
Josh Morcombe
Sean Aitken
Henrique Rodrigues
Andor Siegers
MTE 252

Introduction

In this phase of the project, the goal is to develop a filter function that creates a passband for certain frequencies, and then split the input audio files into multiple frequency bands, which can stimulate the electrodes of a cochlear implant. Several passbands are created to span from 100 Hz to 8 kHz. These filters help analyze different ranges of the sound spectrum with each frequency band corresponding to a distinct electrode along the implant. Tasks include configuring the number of bands, selecting types of filters, and implementing the filter bank in MATLAB to simulate the full signal processing algorithm. After creating the filters, the audio files need to be enveloped so they can be played back on phase 3.

Filter Decisions

To create the system of filters for the cochlear implant, the specific design of the lowpass and bandpass filters needs to be decided. The third signal processing step needed, signal rectification is done through the form of an absolute value function and doesn't need a specific filter type. MATLAB's filter designer offers two different filter classes that can be selected, Finite Impulse Response (FIR), and Infinite Impulse Response (IIR). The main difference between these filtering methods is that IIR is recursive and uses some of the output as input while FIR is not recursive and does not have feedback [1]. This difference in behaviour causes differences in performance, some of these notable characteristics are summarized in Figure 1 below.

	IIR	FIR
Computational Speed	Fast – Low Order	Slow – High Order
Phase / Delay	Not constant	Constant
Stability	Sometimes	Always

Figure 1: IIR and FIR Filters Differences [1]

Since the cochlear implant is to be designed for use in everyday conversation, it is important that the computation speed of the implant is fast to allow immediate conveying of information to the user. For this reason, an IIR filter design was favoured. The other major reason for use of an IIR filter is that they offer a sharper rolloff, as seen below in Figure 2.

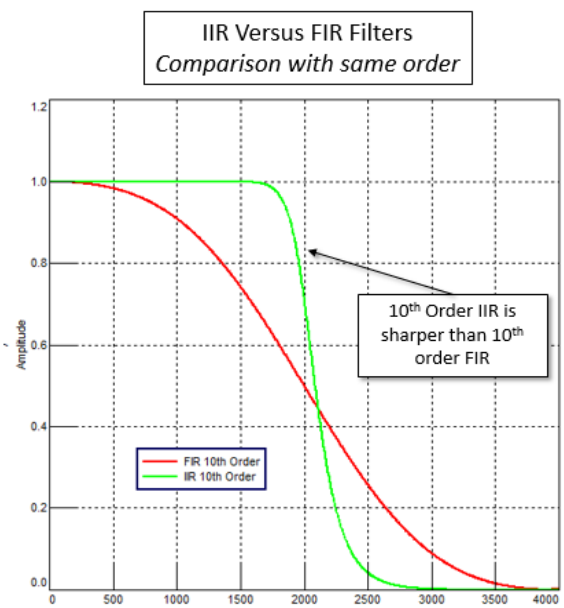

Figure 2: IIR vs FIR Rolloff [1]

At the same order, FIR and IIR filters have different rolloffs. Since higher order filters require more computation time, the desired rolloff of a high order FIR filter can be achieved with a lowerer order IIR filter. Not only does this allow for less computation time and faster results, but it also allows for the frequency ranges to be placed closer together with less interference. Due to the improved accuracy of bandpass filtering and higher speeds, an IIR filter design was selected.

When considering MATLAB’s IIR filter design options, Butterworth, Chebyshev type 1 and 2, and elliptic filters were considered. For simplicity of the first design, the same type of filter will be used in both the bandpass and lowpass filter steps of the full signal processing. This may be changed in later designs if different types of filters fit the lowpass and bandpass requirements more appropriately. Some bandpass filters using each of MATLAB’s IIR filter options can be seen below in figure 3.

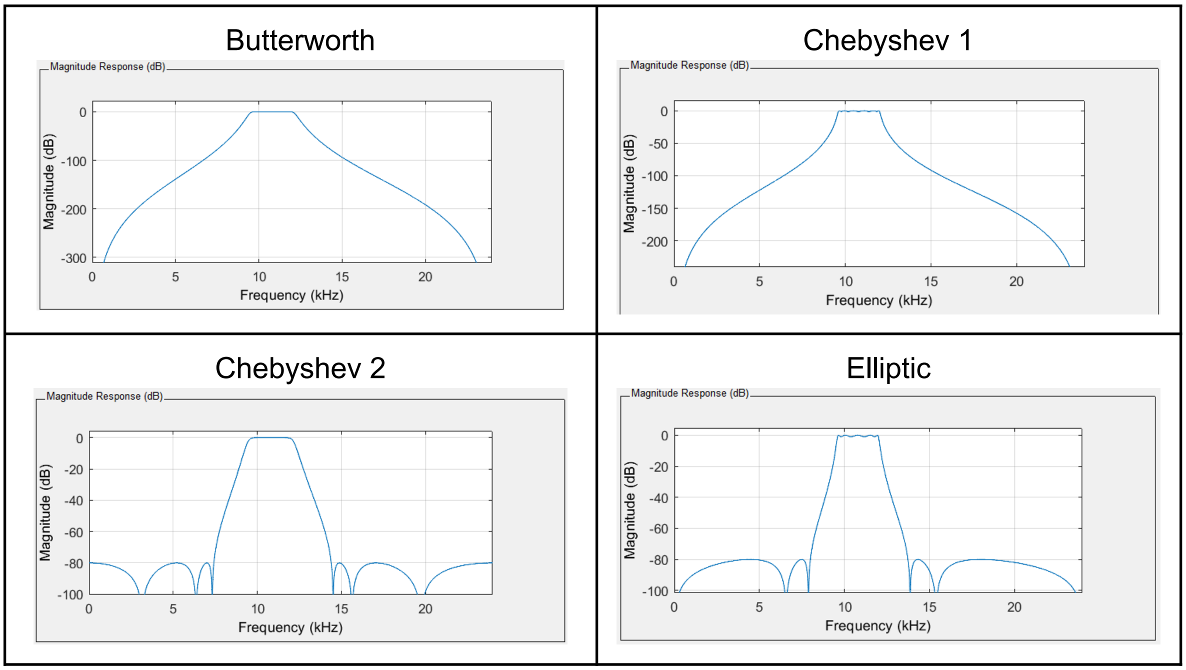

Figure 3: IIR Bandpass Filter Options

Each filter provides its own benefits; Butterworth has a flat passband and stopband, Chebyshev has a sharper rolloff, and elliptic has the sharpest rolloff of the options. However, each filter besides the Butterworth filter contains ripples in both the passband and stopband [3]. This means that either frequencies that should pass unmodified will experience minor amounts of interference, or frequencies that should be filtered out will not be removed entirely.

Since noise will already be created in the system due to the adjacent positioning of the passbands, it was decided any additional noise should be minimized. For this reason, the Butterworth filter design was selected as the passband and stopband will remain perfectly flat allowing for the frequencies being filtered to remain either unchanged, or be completely eliminated. The main drawback of the Butterworth filter came from its wide transition zone, but this could be accounted for by increasing the order of the filter causing a sharper rolloff. This higher order and sharper rolloff will increase computation time, but IIR filters inherently have a sharper rolloff. For these reasons, an IIR Butterworth filter was selected to be used in the design with the order of the filters being adjustable, allowing for improvement through iteration.

Bandpass Considerations

Another important decision that needed to be made was how to divide the large frequency range into a series of different frequency bands. Having more frequency bands would improve the overall clarity of the sound, however there are many limiting factors on how many bands can be used. Firstly, it would increase the amount of overhead in the system, increasing the computation time and therefore the delay. Beyond this, too many bands spaced too closely together would result in overlap of the frequency bands because the rolloff is not perfectly sharp, and there is also a limited amount of space in the inner ear to place the electrodes. Ultimately, based on these considerations, observations of Butterworth roll off rate and reference to other external research [2], it was determined that approximately 20 bands was an appropriate starting point for the initial number of bands for this first design.

It was also important to distribute the frequency bands to properly reflect patterns in human hearing. The human ear naturally perceives frequencies following a logarithmic scale, meaning people are better able to distinguish small frequency changes at lower frequencies rather than higher ones [4], and most human speech falls within roughly 1 to 2 kHz, near the bottom of the range of perceptible frequencies [2]. As a result, the bands of the cochlear implant were distributed logarithmically, so that the output will have a higher resolution at lower frequencies where that is more important. Overall, this will improve the clarity of the sound for the user by aligning with how people naturally perceive audio.

MATLAB Implementation

Once each signal is loaded into MATLAB and resampled, it needs to be processed through the passband filters. To implement the 20, evenly spaced (on a logarithmic scale) passbands, first each boundary value was calculated. This was done by taking the starting and ending frequencies of the entire band being analyzed, in this case 100 Hz and 8 KHz, and calculating their logarithms. Once the ending bounds have been converted to logarithmic space, a list of 21 (n+1) evenly spaced values can be calculated. These values are then converted into frequencies by raising 10 to the power of each value, creating a list of boundary values which define each passband.

To split the signal into its corresponding passbands, a for loop is used. For any given loop *i*, the original signal is passed through a passband filter with cutoff frequencies *i* and *i+1* in the boundary list, effectively filtering out any frequencies not within those bounds. The filtered signal is saved to an array containing each passband, which is later used for signal plotting. Next, the signal is rectified, and passed through a lowpass filter with a 400 Hz cutoff frequency, effectively enveloping it, outputting a function that traces the signal amplitude. This enveloped signal is then saved to another array. Once these arrays have been populated, the signals can finally be plotted. Only the first and last elements in each array are displayed showing the highest and lowest frequency bands of the design, as per the assignment instructions. The resulting plots can be seen in the Results section below.

Results

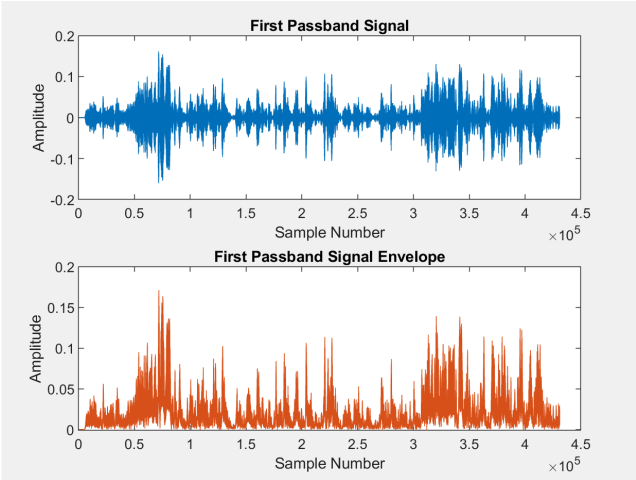

Figure 4: Filtered Signal in Lowest Band

The above plot shows the passband filtered signal in the first passband, along with its corresponding envelope. This passband has a range of 100 Hz to 124.5 Hz.

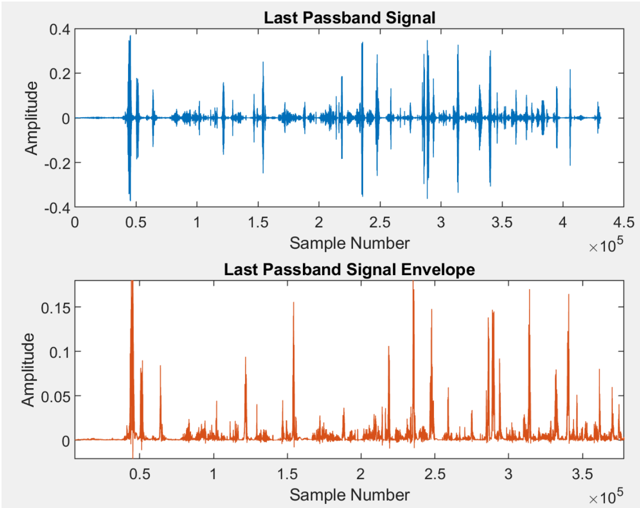

Figure 5: Filtered Signal in Highest Band

The above plot shows the passband filtered signal in the final passband, along with its corresponding envelope. This passband has a range of 6.425 kHz to 7.999 kHz.

References

[1] “Siemens Digital Industries Software Community.” Accessed: Nov. 18, 2024. [Online]. Available: https://community.sw.siemens.com/s/article/introduction-to-filters-fir-versus-iir

[2] A. K. Bosen and M. Chatterjee, “Band importance functions of listeners with cochlear implants using clinical maps,” *The Journal of the Acoustical Society of America*, vol. 140, no. 5, p. 3718, Nov. 2016, doi: 10.1121/1.4967298.

[3] “IIR Filters and FIR Filters,” NI. Accessed: Nov. 11, 2024. [Online]. Available: https://www.ni.com/docs/en-US/bundle/diadem/page/genmaths/genmaths/calc_filterfir_iir.htm

[4] “Understanding Sound Levels,” Kiss Your Ears. Accessed: Nov. 20, 2024. [Online]. Available: https://www.kissyourears.com/pages/understanding-sound-levels