Individual Engineering Project Final Report

Implementing a 3-Band Equaliser

Student Name: Luca Ricagni

Student Number: 16010178

Module code: UFMFH8-15-3

Submission Date: 23/04/20

Main Text Word Count: 2516

Department of Engineering, Design, and Mathematics
University of the West of England

Contents

Introduction

The original aim of this project was to design a 3-band equaliser and implement it using the Cypress FM4 board. However due to the Covid-19 outbreak in early 2020 and the subsequent lockdown the implementation of the equaliser was changed to use MATLAB instead.

An equaliser is made up of low pass, high pass and band pass filters and each of these filters must meet the requirements of having pass band ripples with a maximum value of 5%, as well as having stop band ripples with a minimum value of 40dB. The equaliser will be designed using either a finite impulse response (FIR) filter, or an infinite impulse response (IIR) filter.

The human audio spectrum has a range of about 20Hz to 20kHz of audible frequencies. However, for many adults the upper limit of the audible range of frequencies is about 17kHz (Purves, et al., 2011). The equaliser will need to be designed to work within this range.

Background Research

What is an equaliser

An equaliser is a system (which can be hardware or software) which adjusts the volume, or loudness, of certain frequencies. Some frequencies sound louder than others to the human ear, despite having equal or even greater energy. The closer these frequencies get to the bounds of the human audio spectrum the softer they sound (Trivedi, 2016).

An equaliser can have a range of numbers of bands, from the simplest ones having two to professional audio systems having upwards of 20. The more bands an equaliser has the greater the level of control over the sound. This is because each band is narrower thus providing more control over the overall sound (Trivedi, 2016).

7-Band equaliser

A 7-band equaliser is a common design for an equaliser in audio systems. Each band has a distinct frequency range and can be manipulated individually to ensure a good output. The common bands for a 7-band equaliser are as follows (Shepherd, 2010):

Sub-bass	Bass	Low Midrange	Midrange	Upper Midrange	Presence	Brilliance
20-60 Hz	60-250 Hz	250-500 Hz	500-2000 Hz	2-4 kHz	4-6 kHz	6-20 kHz

Sub-bass – Often “felt” more than heard and provides the thump or boom to a sound

Bass – The band which determines the main beat or rhythm

Low Midrange – Adds warmth and weight as well as clarity to the sound

Midrange – Gives body and tone to instruments in the sound

Upper Midrange – This gives the edge or bite to instruments such as guitars

Presence – Adds clarity to the sound, especially for snare drums

Brilliance – This band adds the air or sparkle to the sound

3-Band equaliser

3-band equalisers function in much the same way as 7-band equalisers, with the notable point of having fewer bands and thus having less control over the sound. In some systems these bands are then further split to allow for a higher resolution sound. A 3-band equaliser commonly has bands as follows:

Bass	Midrange	Treble
20-250 Hz	250-4000 Hz	4-20 kHz

Types of digital filters

Finite Impulse Response (FIR) Filters

An FIR filter computes the output at a given time based off the current and previous inputs. It does not consider previous outputs therefore making it not recursive (Barr, 2002). FIR filters do not have an equivalent in the analogue domain and are constructed using coefficients in order to define the response of the filter.

FIR filters are the most efficient filters used in digital signal processing and are often designed using a windowing technique such as a rectangular, von Hann or Hamming window. This will truncate (limit) the impulse response by creating a viewing window in order to calculate a finite number of coefficients which will subsequently repeat.

The advantages and disadvantages of an FIR filter are as follows (Anon, 2020):

Advantages	Disadvantages
• Always stable	• Require a large memory storage
• Simple and easy to design	• Long processing times
• Have and ideal linear phase delay	• Necessitate a higher order

Infinite Impulse Response (IIR) Filters

An IIR filter works by computing the output at a given time based off both the current and previous inputs, as well as the previous outputs (Grout, 2008). Due to this feedback these filters are recursive. These are usually built using bilinear transformation to map an analogue filter into a digital one such as a Butterworth or Chebyshev filter.

The advantages and disadvantages of an IIR filter are as follows (Anon, 2020):

Advantages	Disadvantages
• Computationally efficient	• Non-linear phase response
• Lower order can be used	• Can be unstable
• Less memory storage required	

Windowing techniques

Truncating the infinite impulse response which was calculated using the Fourier Transform requires the use of a window. Using such a window will create a finite viewing window for the impulse response.

In order to meet the requirements set out in the design specification several windows were investigated including the Rectangular, Triangular, von Hann and Hamming windows. A Hamming window was chosen because it has a main spectral lobe similar to the triangular window but with sidelobes much smaller than the triangular window and thus has much improved sidelobe performance. In addition, using the Hamming window meant that the stop band ripples were successfully attenuated to 40dB due to its longer transition band.

Design

Requirements

	Suggested	Proposed	Reason
Equaliser bandwidth	22 kHz	24 kHz	Sampling frequency is 48 kHz
Low pass band	0 → 7.33 kHz	0 → 375 Hz	Smaller band for more important frequencies
Band pass band	7.33 → 14.66 kHz	0 → 3.5kHz	Medium band for moderately important frequencies
High pass band	14.66 → 22kHz	3.5 → 24 kHz	Large band for less important frequencies
Pass band ripple	Max. 5%	Max. 5%	Must remain the same as per requirements
Stop band rejection	Min. 40dB	Min. 40dB	Must remain the same as per requirements

Lowpass Filter

This section will detail the calculation of the lowpass filter coefficients and discuss its frequency response plot.

Coefficients calculations

a) Calculate the cut-off frequency in radians Ω_c and the order of the filter N

$$Cutoff\ Frequency\ f_c = 375\ Hz$$

$$Sampling\ Frequency\ f_{sample} = 48\ kHz$$

$$Cutoff\ Frequency\ (rads)\ \Omega_c = \frac{f_c}{f_{sample}} * 2\pi = \frac{375}{48000} * 2\pi = \frac{\pi}{64} = 0.04908738521$$

$$M = 25$$

$$Number\ of\ coefficients\ N = 2M + 1 = (2 * 25) + 1 = 51$$

b) Formula to calculate filter coefficients with window truncation from impulse response values and the window formula (Lynn & Fuerst, 2000).

$$h[n] = \frac{\Omega_c}{\pi} sinc(n\Omega_c) = \frac{\sin(n\Omega_c)}{n\pi}$$

$$Hamming\ Window\ function\ w[n] = \left[0.54 + 0.46 \cos\left(\frac{n\pi}{M}\right)\right]$$

$$h_w[n] = h[n] * w[n] = \frac{\sin(n\Omega_c)}{n\pi} * \left[0.54 + 0.46 \cos\left(\frac{n\pi}{M}\right)\right]$$

c) Calculate first three coefficients

In order to calculate filter coefficient when $n = 0$ $h_w[0]$ L'Hôpital's rule must be used to avoid dividing by 0. L'Hôpital's rule states that to do this simply use the derivatives of both the numerator and the denominator:

$$h[0] = \frac{\frac{d\{\sin(n\Omega_c)\}}{dn}}{\frac{d\{n\pi\}}{dn}} = \frac{\Omega_c \cos(n\Omega_c)}{\pi} = \frac{\Omega_c}{\pi}$$

The $\cos(n\Omega_c)$ can be ignored as $n = 0$ and $\cos(0 * \Omega_c) = 1$. The window function $w[n]$ can also be ignored for a similar reason.

$$h_w[0] = \frac{\Omega_c}{\pi} = \frac{\frac{\pi}{64}}{\pi} = \frac{1}{64} = 0.015625$$

$$h_w[1] = \frac{\sin\left(1 * \frac{\pi}{64}\right)}{1 * \pi} * \left[0.54 + 0.46\cos\left(\frac{1 * \pi}{25}\right)\right] = 0.015562073$$

$$h_w[2] = \frac{\sin\left(2 * \frac{\pi}{64}\right)}{2 * \pi} * \left[0.54 + 0.46\cos\left(\frac{2 * \pi}{25}\right)\right] = 0.01537446642$$

Frequency Response Diagrams

Figure A.1 shows the Normalized Frequency response graphs for both magnitude (dB) and phase (degrees) for the lowpass filter. Figure A.2 shows a zoomed in version of the magnitude response graph. These graphs show that the filter meets the requirements as the stop band attenuation happens well before -40dB and the pass band ripple is significantly less than 5%. These figures can be obtained by selecting the ‘magnitude and phase response’ diagram option on MATLAB’s filterDesigner. Alternatively, these can be obtained by exporting the coefficients from filterDesigner to MATLAB and using the function freqz() to plot the frequency response plots.

Highpass Filter

This section will detail the calculation of the highpass filter coefficients and discuss its frequency response plot.

Coefficients calculations

a) Calculate the cut-off frequency in radians Ω_c and the order of the filter N

$$Cutoff\ Frequency\ f_c = 3500\ Hz$$

However, this $Cutoff\ Frequency\ f_c$ must be shifted, as this is a highpass filter, to essentially be the bandwidth of the filter, therefore:

$$Cutoff\ Frequency\ f_c = 24000 - 3500 = 20500\ Hz$$

$$Sampling\ Frequency\ f_{sample} = 48\ kHz$$

$$Cutoff\ Frequency\ (rads)\ \Omega_c = \frac{f_c}{f_{sample}} * 2\pi = \frac{20500}{48000} * 2\pi = \frac{41\pi}{48} = 2.683443725$$

$$M = 25$$

$$Number\ of\ coefficients\ N = 2M + 1 = (2 * 25) + 1 = 51$$

b) Formula to calculate filter coefficients with window truncation from impulse response values and the window formula (Lynn & Fuerst, 2000).

$$h[n] = \frac{\Omega_c}{\pi} sinc(n\Omega_c) = \frac{\sin(n\Omega_c)}{n\pi}$$

$$Hamming\ Window\ function\ w[n] = \left[0.54 + 0.46\cos\left(\frac{n\pi}{M}\right)\right]$$

Due to this being highpass filter coefficients $h[n]$ will also have to be multiplied by: $\cos(n\pi)$

$$h_w[n] = h[n] * w[n] = \frac{\sin(n\Omega_c)}{n\pi} * \left[0.54 + 0.46\cos\left(\frac{n\pi}{M}\right)\right] * \cos(n\pi)$$

c) Calculate first three coefficients

In order to calculate filter coefficient when $n = 0$ $h_w[0]$ L'Hôpital's rule must be used to avoid dividing by 0. L'Hôpital's rule states that to do this simply use the derivatives of both the numerator and the denominator:

$$h[0] = \frac{\frac{d\{\sin(n\Omega_c)\}}{dn}}{\frac{d\{n\pi\}}{dn}} = \frac{\Omega_c\cos(n\Omega_c)}{\pi} = \frac{\Omega_c}{\pi}$$

The $\cos(n\Omega_c)$ can be ignored as $n = 0$ and $\cos(0 * \Omega_c) = 1$. The window function $w[n]$ can also be ignored for a similar reason.

$$h_w[0] = \frac{\Omega_c}{\pi} = \frac{\frac{41\pi}{48}}{\pi} = \frac{41}{48} = 0.8541666667$$

$$h_w[1] = \frac{\sin\left(1 * \frac{41\pi}{48}\right)}{1 * \pi} * \left[0.54 + 0.46\cos\left(\frac{1 * \pi}{25}\right)\right] * \cos(1 * \pi) = -0.1402742025$$

$$h_w[2] = \frac{\sin\left(2 * \frac{41\pi}{48}\right)}{2 * \pi} * \left[0.54 + 0.46\cos\left(\frac{2 * \pi}{25}\right)\right] * \cos(2 * \pi) = -0.1244413401$$

Frequency Response Diagrams

Figure A.3 shows the Normalized Frequency response graphs for both magnitude (dB) and phase (degrees) for the highpass filter. Figure A.4 shows a zoomed in version of the magnitude response graph. These graphs show that the filter meets the requirements as the stop band attenuation happens well before -40dB and the pass band ripple is significantly less than 5%. These figures can be obtained by selecting the 'magnitude and phase response' diagram option on MATLAB's filterDesigner. Alternatively these can be obtained by exporting the coefficients from filterDesigner to MATLAB and using the function freqz() to plot the frequency response plots.

Bandpass Filter

This section will detail the calculation of the bandpass filter coefficients and discuss its frequency response plot.

Coefficients calculations

a) Calculate the cut-off frequency in radians Ω_c and the order of the filter N

$$Lower\ Cutoff\ Frequency\ f_{c1} = 375\ Hz$$

$$Upper\ Cutoff\ Frequency\ f_{c2} = 3500\ Hz$$

$$Sampling\ Frequency\ f_{sample} = 48\ kHz$$

$$Cutoff\ Frequency\ (rads)\ \Omega_{c1} = \frac{f_{c1}}{f_{sample}} * 2\pi = \frac{375}{48000} * 2\pi = \frac{\pi}{64} = 0.04908738521$$

$$Cutoff\ Frequency\ (rads)\ \Omega_{c2} = \frac{f_{c2}}{f_{sample}} * 2\pi = \frac{3500}{48000} * 2\pi = \frac{41\pi}{48} = 2.683443725$$

$$M = 25$$

$$Number\ of\ coefficients\ N = 2M + 1 = (2 * 25) + 1 = 51$$

b) Formula to calculate filter coefficients with window truncation from impulse response values and the window formula (Lynn & Fuerst, 2000). Due to this being a bandpass filter the formula to calculate $h[n]$ will need to be adjusted similarly to include both f_{c1} and f_{c2}.

$$h[n] = \frac{\Omega_{c2}}{\pi} sinc(n\Omega_{c2}) - \frac{\Omega_{c1}}{\pi} sinc(n\Omega_{c1}) = \frac{\sin(n\Omega_{c2})}{n\pi} - \frac{\sin(n\Omega_{c1})}{n\pi}$$

$$Hamming\ Window\ function\ w[n] = \left[0.54 + 0.46\cos\left(\frac{n\pi}{M}\right)\right]$$

$$h_w[n] = h[n] * w[n] = \frac{\sin(n\Omega_c)}{n\pi} * \left[0.54 + 0.46\cos\left(\frac{n\pi}{M}\right)\right]$$

c) Calculate first three coefficients

In order to calculate filter coefficient when $n = 0$ $h_w[0]$ L'Hôpital's rule must be used to avoid dividing by 0. L'Hôpital's rule states that to do this simply use the derivatives of both the numerator and the denominator:

$$h[0] = \frac{\frac{d\{\sin(n\Omega_c)\}}{dn}}{\frac{d\{n\pi\}}{dn}} = \frac{\Omega_c \cos(n\Omega_c)}{\pi} = \frac{\Omega_c}{\pi}$$

Due to this being a bandpass filter the Ω_c value must be calculated slightly differently:

$$\Omega_c = \left[\frac{3500}{48000} * 2\pi\right] - \left[\frac{375}{48000} * 2\pi\right] = \frac{25\pi}{192} = 0.4090615434$$

The $\cos(n\Omega_c)$ can be ignored as $n = 0$ and $\cos(0 * \Omega_c) = 1$. The window function $w[n]$ can also be ignored for a similar reason.

$$h_w[0] = \frac{\Omega_c}{\pi} = \frac{\frac{25\pi}{192}}{\pi} = \frac{25}{192} = 0.1302083333$$

$$h_w[1] = \left[\frac{\sin\left(1 * \frac{41\pi}{48}\right)}{1 * \pi} - \frac{\sin\left(1 * \frac{\pi}{64}\right)}{1 * \pi}\right] * \left[0.54 + 0.46\cos\left(\frac{1 * \pi}{25}\right)\right] = 0.1247121295$$

$$h_w[2] = \left[\frac{\sin\left(2 * \frac{41\pi}{48}\right)}{2 * \pi} - \frac{\sin\left(2 * \frac{\pi}{64}\right)}{2 * \pi}\right] * \left[0.54 + 0.46\cos\left(\frac{2 * \pi}{25}\right)\right] = 0.1090668736$$

Frequency Response Diagrams

Figure A.5 shows the Normalized Frequency response graphs for both magnitude (dB) and phase (degrees) for the bandpass filter. Figure A.6 shows a zoomed in version of the magnitude response graph. These graphs show that the filter meets the requirements as the stop band attenuation happens well before -40dB and the pass band ripple is significantly less than 5%. These figures can be obtained by selecting the 'magnitude and phase response' diagram option on MATLAB's filterDesigner. Alternatively, these can be obtained by exporting the coefficients from filterDesigner to MATLAB and using the function freqz() to plot the frequency response plots.

Combining the three filters

The plots shown in Figure A.7 were generated by exporting the coefficients for the three filters from filterDesigner and calculating the sum of these. Then using the freqz() function to plot the frequency response graphs. These graphs prove that the combination of the three filters meets the requirements as the magnitude response is within $\pm 5\%$ of 0dB.

Proof the filters and the equaliser work

Figure A.8 shows the combined sine wave form of sine waves with frequencies of: 200Hz, 1000Hz, 3000Hz, 4500Hz and 10000Hz. Figure A.9 shows the sine wave after it has been put through the lowpass filter, as can be seen most of the frequencies have been attenuated as expected. Figure A.10 shows the sine wave after it has been put through the bandpass filter, two of the frequencies passed through as anticipated. Figure A.11 shows the sine wave after it has been put through the highpass filter two of the frequencies passed through as predicted. Figure A.12 shows the sine wave after it was put through the combination of the three filters (the equaliser), as can be seen this looks identical to Figure A.8. This proves that the equaliser functions correctly.

Conclusion

The coefficients manually calculated exactly matched those calculated by MATLABs FilterDesigner, this can be seen in the Excel sheet attached to this report. The frequency response plots show that each individual filter and the combination of the three filters meet the requirements and attenuate the relevant frequencies successfully.

All MATLAB code used can be found in the zipped folder attached.

While the filters could potentially be changed to contain more coefficients in order to improve the transition band the filters used were sufficient. As such the project was overall a success.

Design Exercise

Task 1.i

$$h_1[n] = \left\{\frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}\right\}$$

$$h[n] = 0.2(\delta[n+2] + \delta[n+1] + \delta[n] + \delta[n-1] + \delta[n-2])$$

$$H(z) = \sum_{k=-2}^{k=2} h[k]z^{-k} = 0.2(z^2 + z^1 + 1 + z^{-1} + z^{-2})$$

$$H(\Omega) = \sum_{k=-2}^{k=2} h[k]e^{-jk\Omega} = 0.2(e^{j2\Omega} + e^{jk\Omega} + 1 + e^{-j\Omega} + e^{-j2\Omega})$$

Shift to make it causal

$$H(z) = 0.2(z^2 + z^1 + 1 + z^{-1} + z^{-2})z^{-2} = 0.2(1 + z^{-1} + z^{-2} + z^{-3} + z^{-4})$$

$$H(z) = 0.2(1 + z^{-1} + z^{-2} + z^{-3} + z^{-4}) * \frac{z^4}{z^4} = \frac{0.2(z^4 + z^3 + z^2 + z^1 + 1)}{z^4}$$

Poles are located at $z^4 = 0$ so there are therefore 4 poles located at the origin.

Zeros are located where $0.2(z^4 + z^3 + z^2 + z^1 + 1) = 0$

$$z^5 - 1 = (z-1)(z^4 + z^3 + z^2 + z^1 + 1)$$

$$z^5 - 1 = 0 \rightarrow z^5 = 1 \rightarrow e^{j5\Omega} = 1 \rightarrow e^{j5\Omega} = \cos(5\Omega) + jsin(5\Omega) = 1$$

$$\therefore \cos(5\Omega) = 1 \;\&\; sin(5\Omega) = 0$$

$$\cos(5\Omega) = 1 \rightarrow 5\Omega = 0 + 2k\pi \rightarrow \Omega = \frac{2k\pi}{5}$$

$$k = 1 \rightarrow \Omega = \frac{2\pi}{5} \rightarrow z = e^{j\frac{2\pi}{5}}$$

$$k = 2 \rightarrow \Omega = \frac{4\pi}{5} \rightarrow z = e^{j\frac{4\pi}{5}}$$

Only interested in the values of $\Omega < \pi$ so only $k = 1$ & $k = 2$. Take the coefficients of π, assuming $f_{sample} = 8kHz$ then $f = 4kHz$.

$$\therefore \frac{2}{5} * 4000 = 1600Hz \;\&\; \frac{4}{5} * 4000 = 3200Hz$$

These frequencies match the notch frequencies specified in the question. Figure A.13, generated using MATLAB, show this as well.

Task 1.ii

A 5-point moving average filter with a cut-off frequency of 200Hz will have two notches. Much like the previous task these notches will be located at $\frac{2}{5}$ & $\frac{4}{5}$ of the bandwidth of the filter, in this case 200Hz. Therefore, these notches will be located at:

$$\frac{2}{5} * 200 = 80Hz$$

$$\frac{4}{5} * 200 = 160Hz$$

The normalised magnitude response plot for this can be found in Figure A.14

Task 2.i

$$h_2[n] = \left\{\frac{1}{12}, \frac{1}{4}, \frac{1}{3}, \frac{1}{4}, \frac{1}{12}\right\}$$

$$h[n] = \frac{1}{12}\delta[n+2] + \frac{1}{4}\delta[n+1] + \frac{1}{3}\delta[n] + \frac{1}{4}\delta[n-1] + \frac{1}{12}\delta[n-2]$$

$$H(z) = \frac{1}{12}z^2 + \frac{1}{4}z + \frac{1}{3} + \frac{1}{4}z^{-1} + \frac{1}{12}z^{-2}$$

$$H(z) = \frac{1}{12} + \frac{1}{4}z^{-1} + \frac{1}{3}z^{-2} + \frac{1}{4}z^{-3} + \frac{1}{12}z^{-4}$$

$$H(z) = \frac{\frac{1}{12}z^4 + \frac{1}{4}z^3 + \frac{1}{3}z^2 + \frac{1}{4}z^1 + \frac{1}{12}}{z^4}$$

Poles are located at $z^4 = 0$ so there are therefore 4 poles located at the origin.

The zeros for this equation are $-\frac{1}{2} - \frac{\sqrt{3}}{2}j$ & $-\frac{1}{2} + \frac{\sqrt{3}}{2}j$

Figure A.15 shows the normalized magnitude response and was generated using MATLABs freqz() function with the impulse response coefficients $h_2[n]$ as the argument. When Figure A.15 is compared to Figure A.13 it can be seen that while both have two notches the notches in Figure A.15 have a much lower magnitude than those in Figure A.13. Figure A.16 shows the unit circle diagram with the poles and zeros plotted on it.

Task 2.ii

Filters designed using the moving average method and the Fourier transform method both have advantages and disadvantages.

	Moving average method	Fourier Transform method
Advantages	• Most common filter type • Optimal at reducing random noise • Very simple conceptually	• Very little information is lost during transformation • Good stopband attenuation
Disadvantages	• Slow roll off • Poor stopband attenuation characteristics	• More complicated to implement • More computationally intense

Which method to use would depend entirely on what the application is, for simple applications that don't need to be as accurate a moving average filter would be appropriate. For more complex applications which require more accuracy using the Fourier transform method would be more suitable.

References

Anon, 2018. *Audio Spectrum Explained.* [Online]
Available at: https://www.teachmeaudio.com/mixing/techniques/audio-spectrum
[Accessed 12 April 2020].

Anon, 2020. *Introduction to Filters: FIR versus IIR.* [Online]
Available at: https://community.plm.automation.siemens.com/t5/Testing-Knowledge-Base/Introduction-to-Filters-FIR-versus-IIR/ta-p/520959
[Accessed 12 April 2020].

Arar, S., 2016. *Design Examples of FIR Filters Using the Window Method.* [Online]
Available at: https://www.allaboutcircuits.com/technical-articles/design-examples-of-fir-filters-using-window-method
[Accessed 12 April 2020].

Arar, S., 2016. *FIR Filter Design by Windowing: Concepts and the Rectangular Window.* [Online]
Available at: https://www.allaboutcircuits.com/technical-articles/finite-impulse-response-filter-design-by-windowing-part-i-concepts-and-rect
[Accessed 12 April 2020].

Arar, S., 2016. *From Filter Specs to Window Parameters in FIR Filter Design.* [Online]
Available at: https://www.allaboutcircuits.com/technical-articles/filter-specs-window-parameters-in-finite-impulse-response-filter-design
[Accessed 12 April 2020].

Barr, M., 2002. *Introduction to Finite Impulse Response Filters for DSP.* [Online]
Available at: https://barrgroup.com/embedded-systems/how-to/digital-filters-fir-iir
[Accessed 13 April 2020].

Boucher, T., 2017. *Waterfall view in gqrx.* [Online]
Available at: https://medium.com/@timboucher/waterfall-view-in-gqrx-15a8b87ca52c
[Accessed 12 April 2020].

Davis, N., 2017. *An Introduction to Audio Electronics: Sound, Microphones, Speakers, and Amplifiers.* [Online]
Available at: https://www.allaboutcircuits.com/technical-articles/introduction-audio-electronics-sound-microphones-speakers-amplifiers
[Accessed 12 April 2020].

Grout, I., 2008. *Digital Systems Design with FPGAs and CPLDs.* 1st ed. Limerick: Elsevier Ltd.

Lynn, P. A. & Fuerst, W., 2000. *Introductory Digital Signal Processing with Computer Applications.* 2nd ed. Chichester: John Wiley & Sons Ltd.

Mathworks, 2020. *FIR Filter Design.* [Online]
Available at: https://www.mathworks.com/help/signal/ug/fir-filter-design.html
[Accessed 12 April 2020].

Mathworks, n.d. *Filter Design Using MATLAB.* [Online]
Available at: https://uk.mathworks.com/discovery/filter-design.html
[Accessed 12 April 2020].

Oppenheim, A. V. & Schafer, W. R., 1989. DT Filter Design: IIR Filters. In: D. o. E. E. a. C. Science, ed. *Discrete-Time Signal Processing.* s.l.:Massachusetts Institute of Technology.

Punskaya, E., n.d. *Design of FIR Filters.* [Online]
Available at:
https://www.vyssotski.ch/BasicsOfInstrumentation/SpikeSorting/Design_of_FIR_Filters.pdf
[Accessed 12 April 2020].

Purves, D. et al., 2011. *Neuroscience.* 2nd ed. Sunderland: Sinauer Associates.

S., 2018. *What are the Different Sounds of Audio? "Warm", "Bright", "Neutral".* [Online]
Available at: https://www.wirerealm.com/info/sound-of-audio-devices-explained
[Accessed 12 April 2020].

Shepherd, I., 2010. *Production Advice: make your music sound great.* [Online]
Available at: https://productionadvice.co.uk/using-eq/
[Accessed 12 April 2020].

Trivedi, T., 2016. *What Is an Equalizer, and How Does It Work?.* [Online]
Available at: https://www.howtogeek.com/59467/htg-explains-what-is-an-equalizer-and-how-does-it-work/
[Accessed 12 April 2020].

Appendices

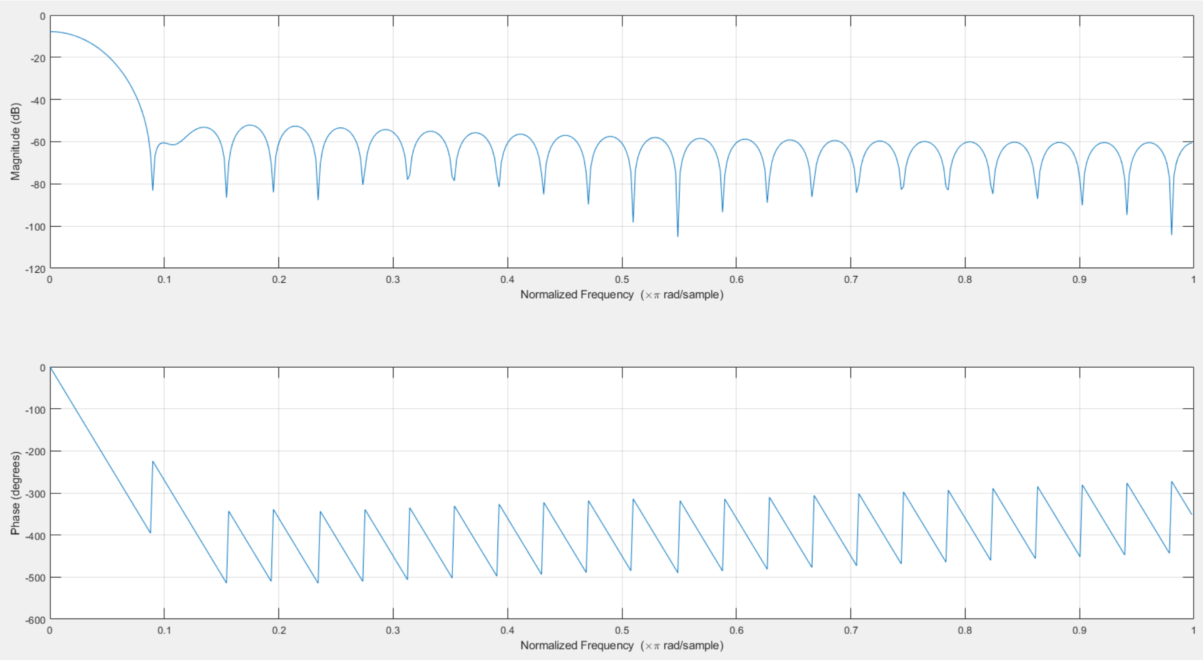

Figure A.1

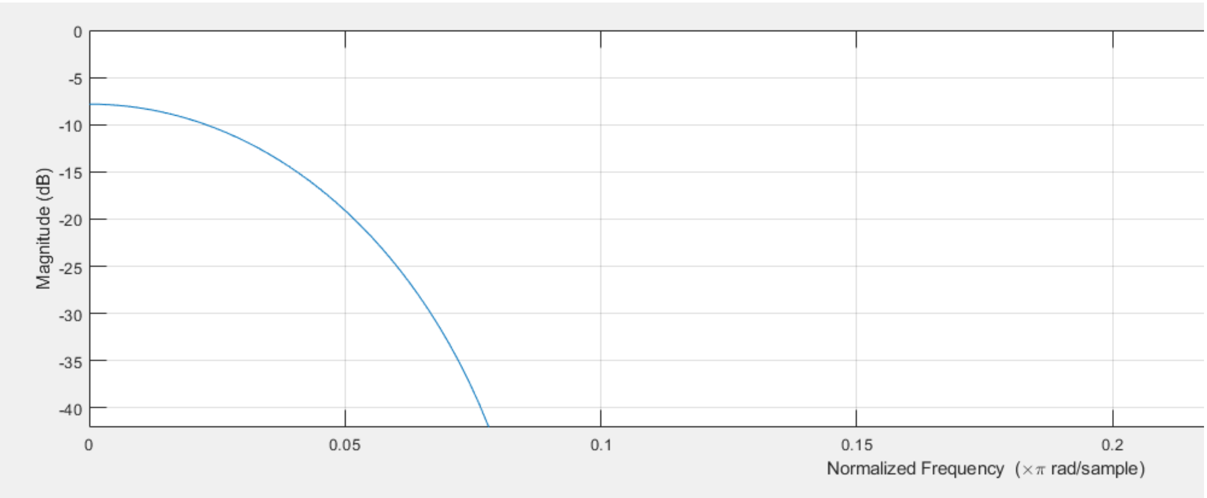

Figure A.2

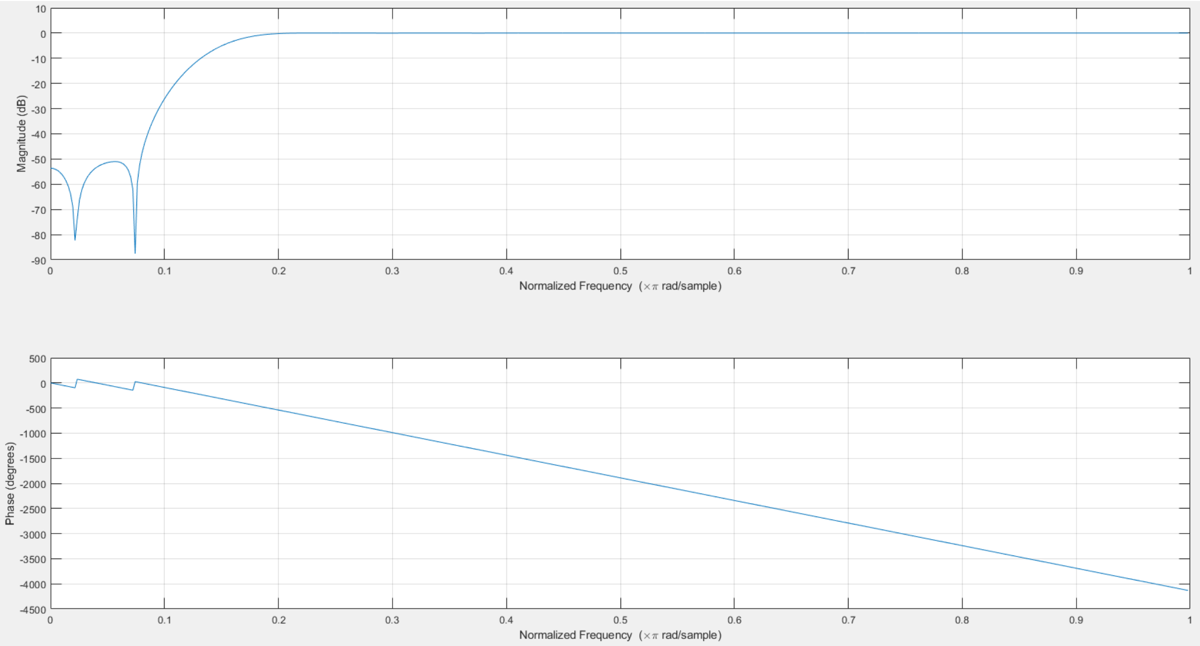

Figure A.3

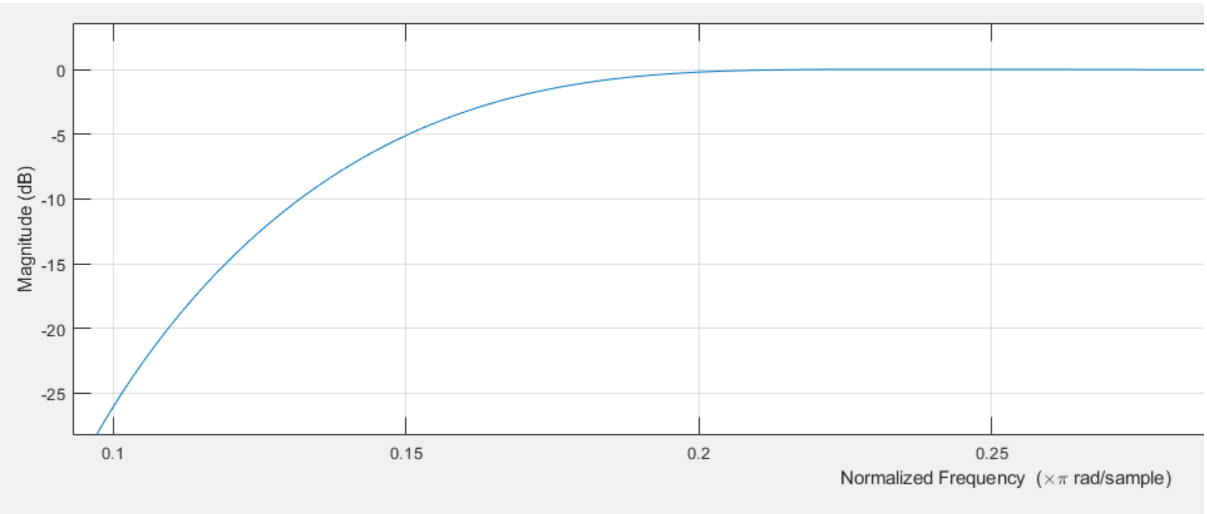

Figure A.4

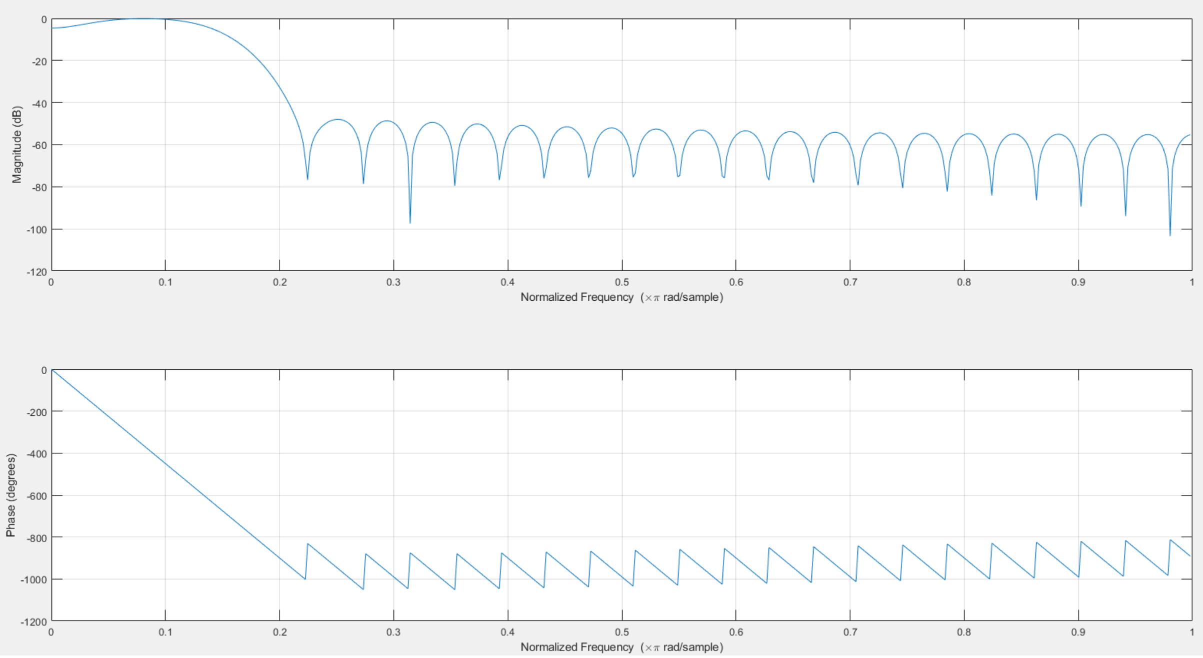

Figure A.5

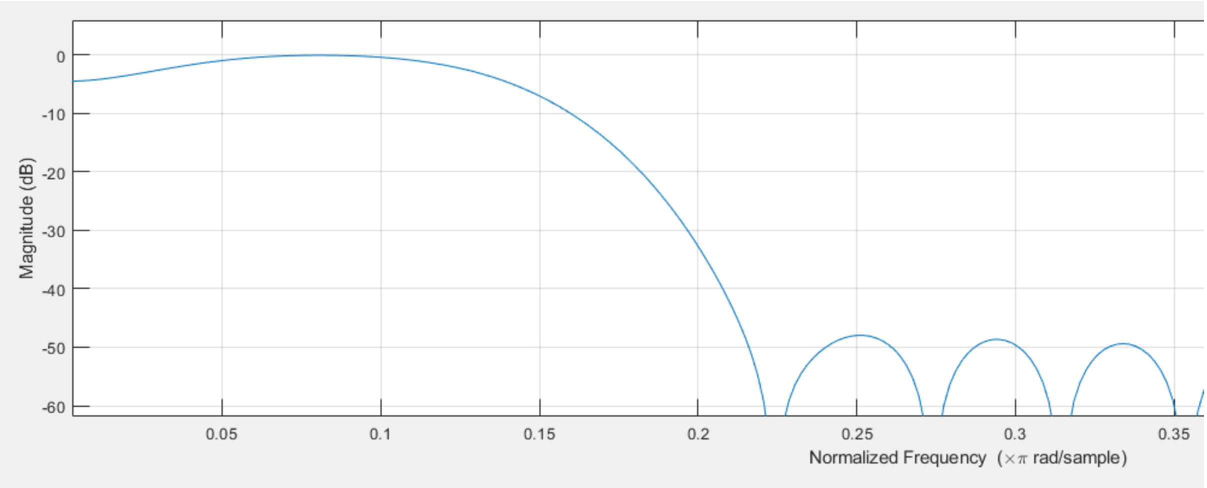

Figure A.6

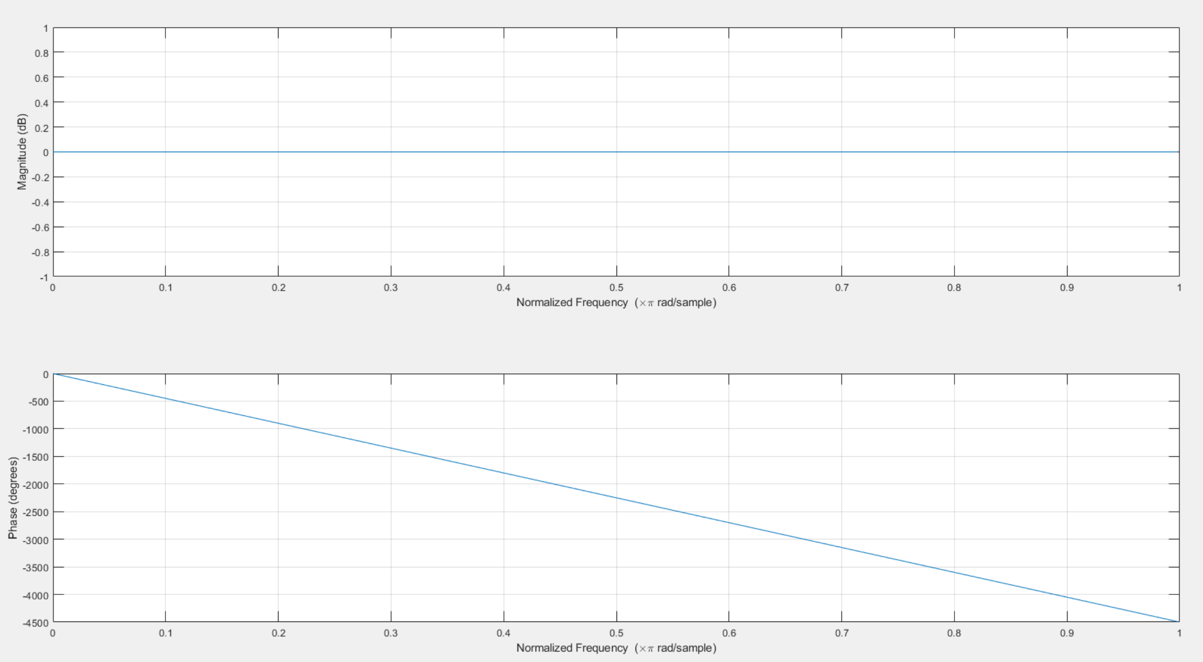

Figure A.7

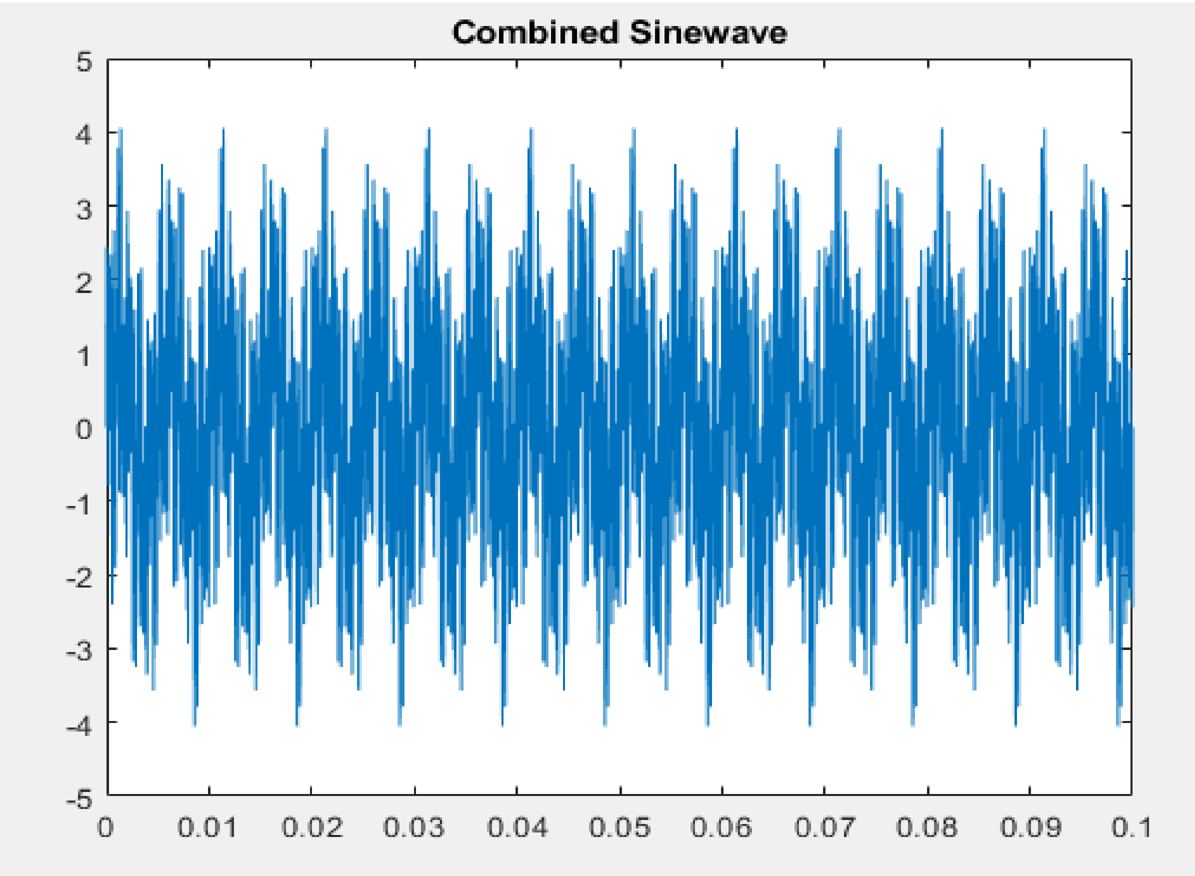

Figure A.8

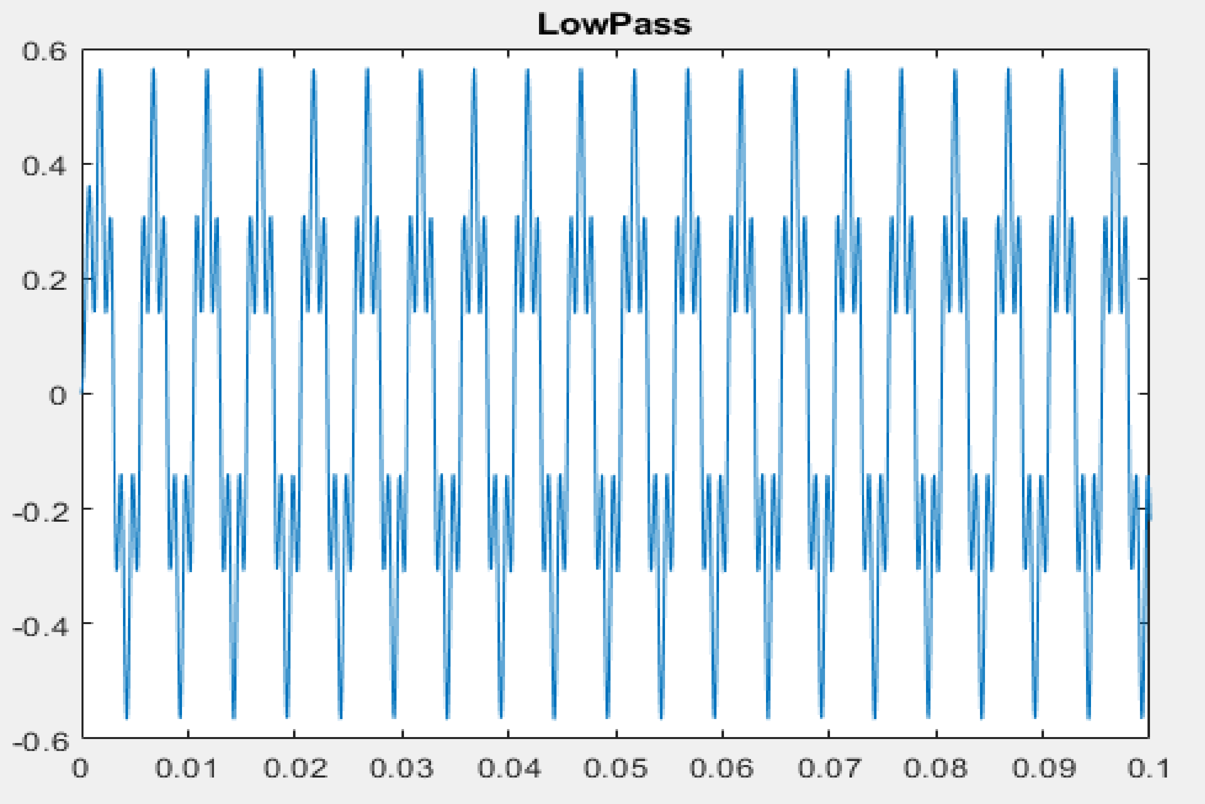

Figure A.9

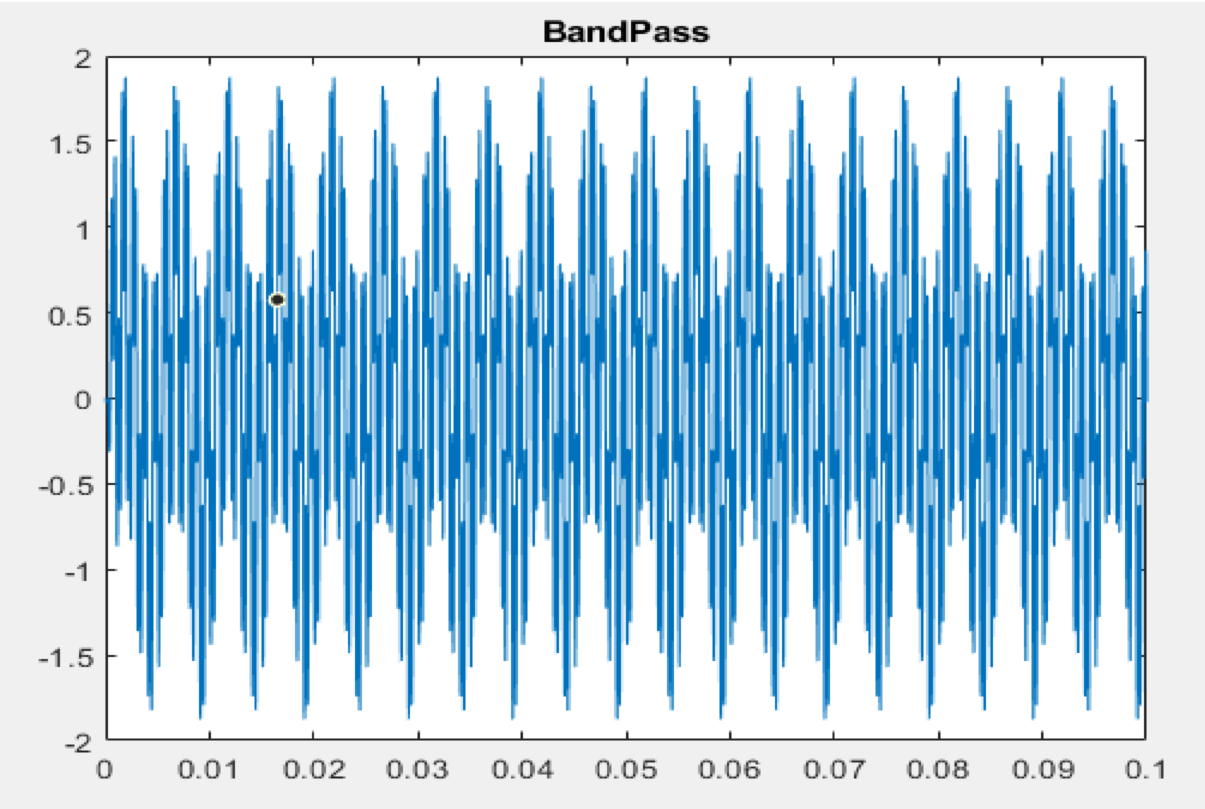

Figure A.10

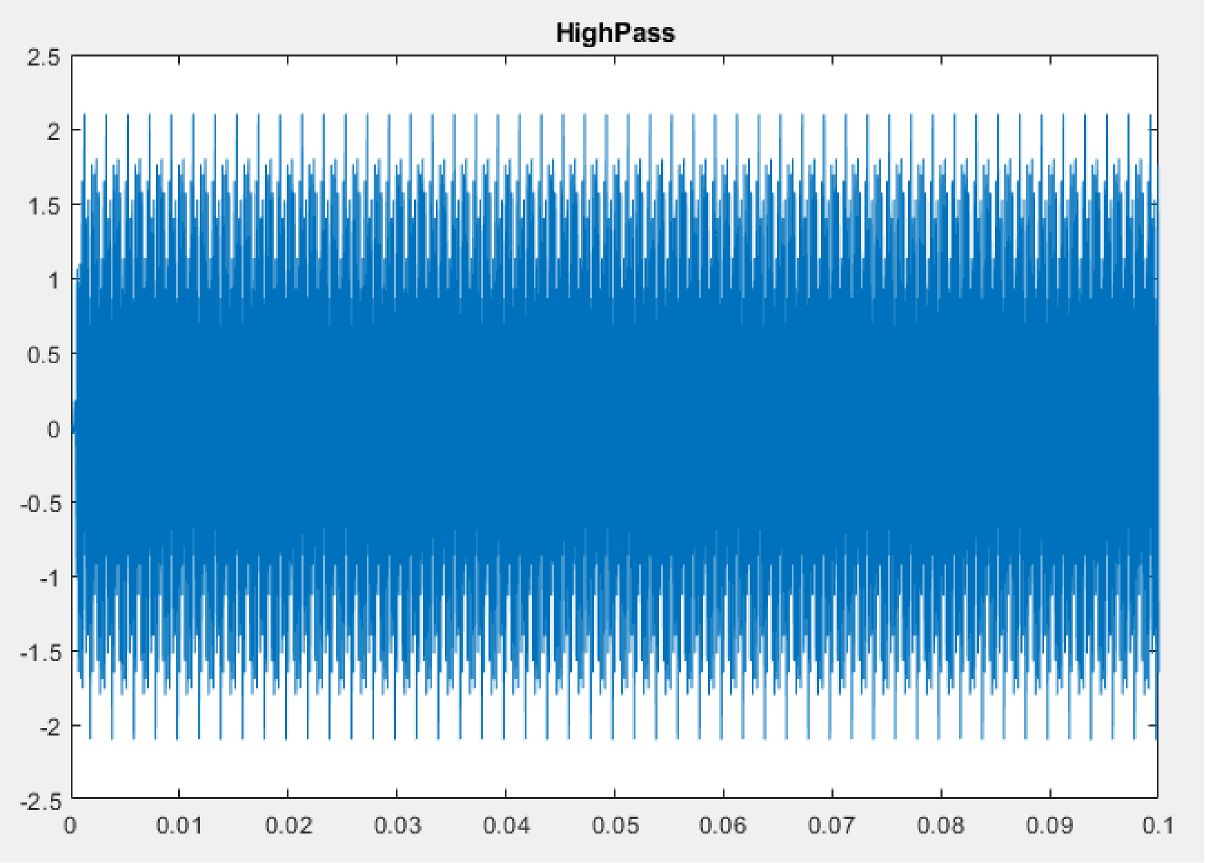

Figure A.11

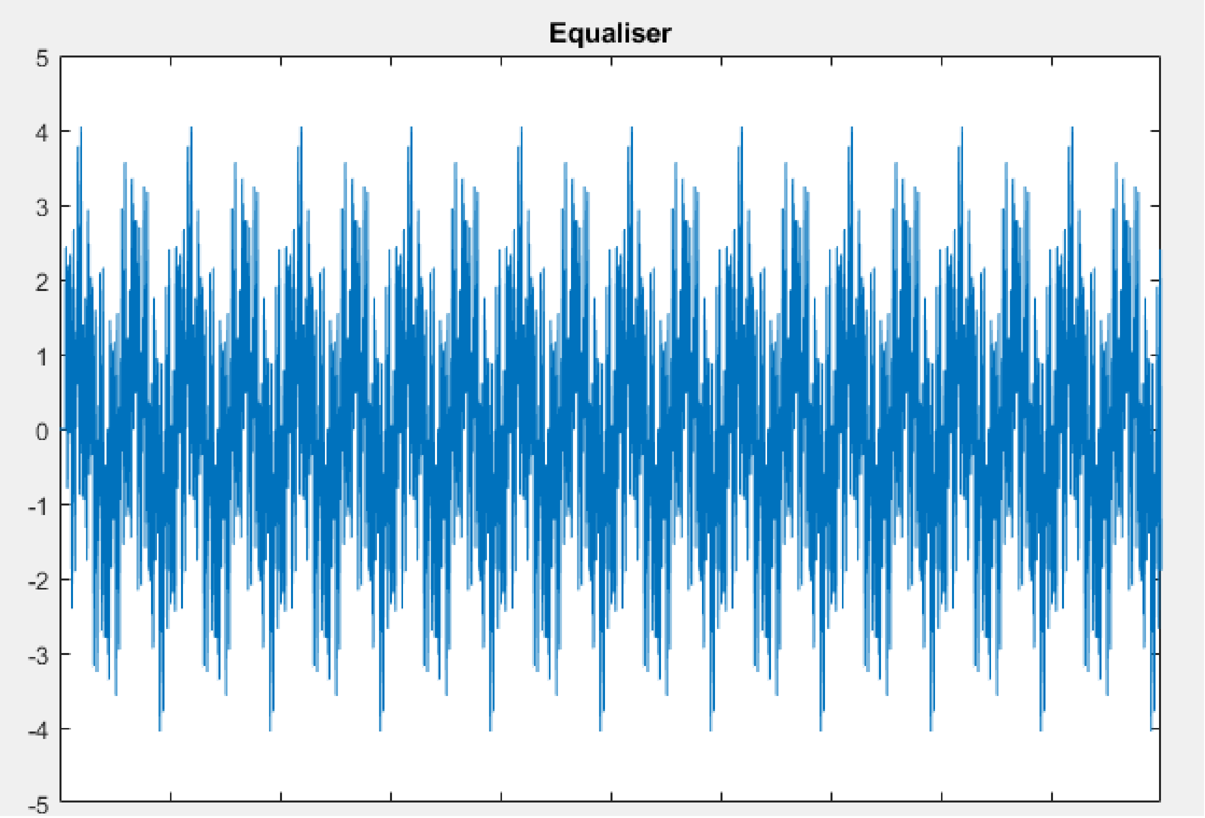

Figure A.12

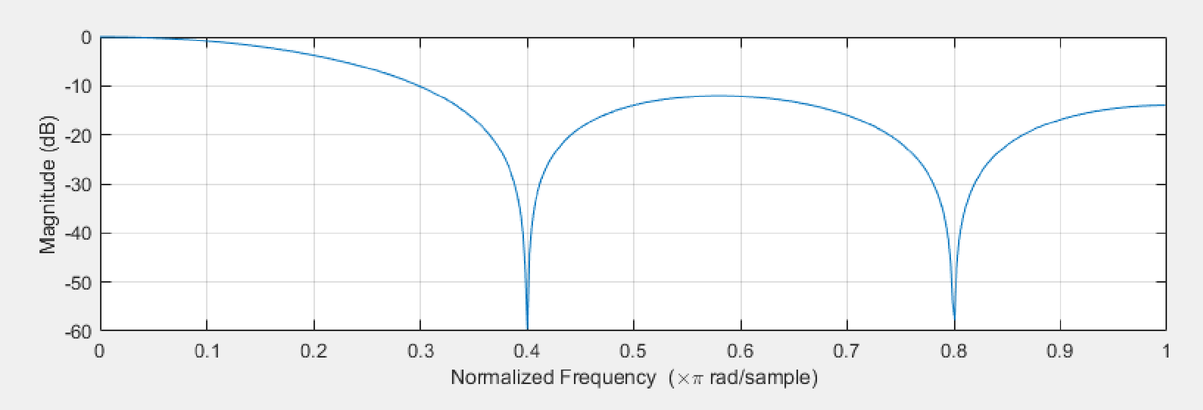

Figure A.13

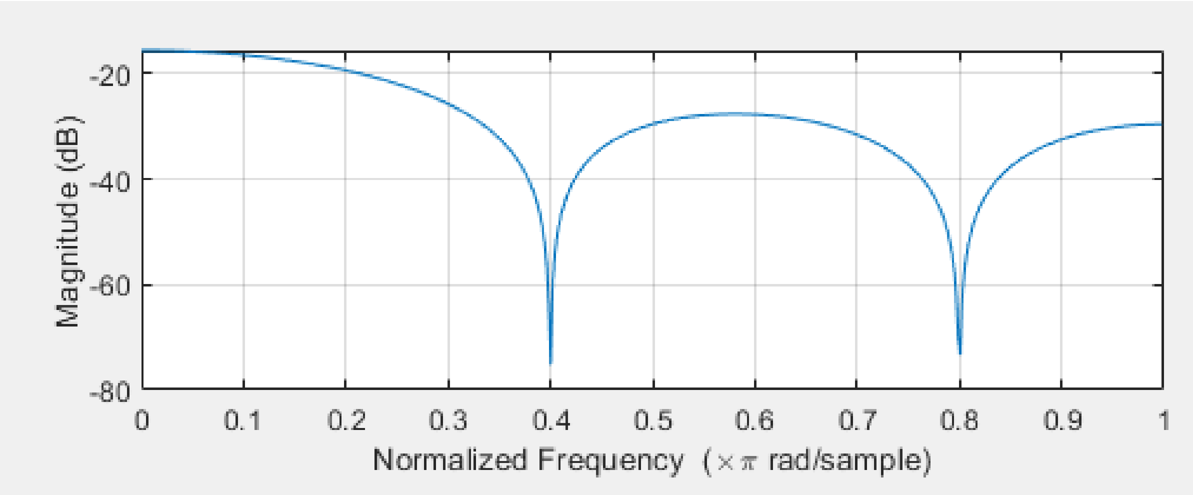

Figure A.14

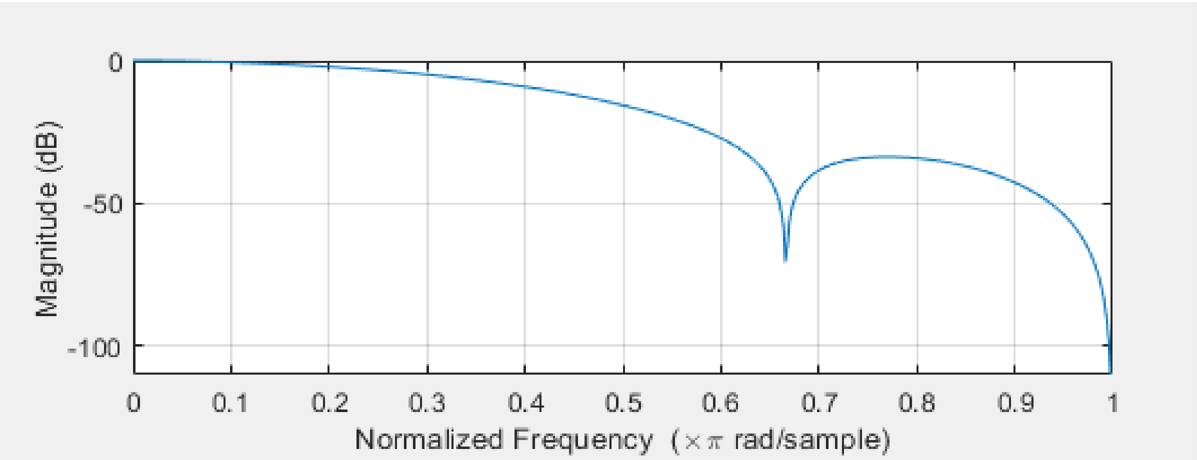

Figure A.15

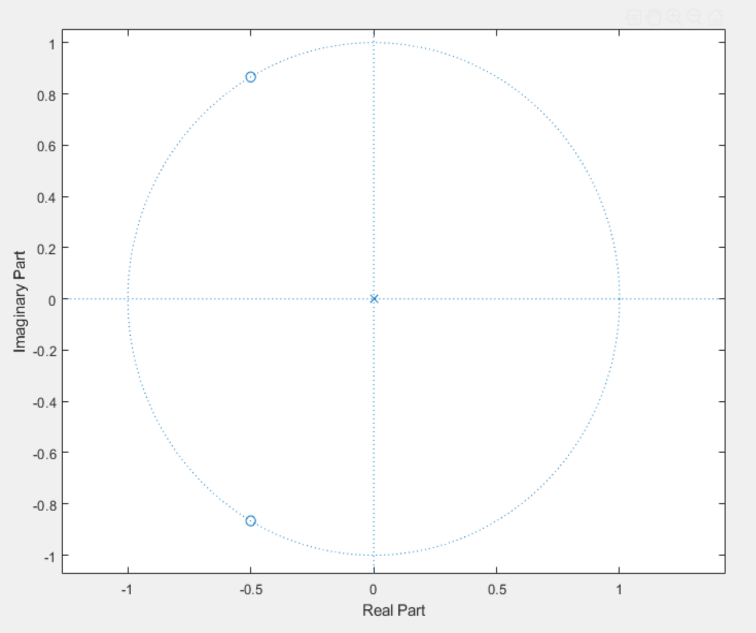

Figure A.16