December 13, 2020

5 Band Equalizer using linear phase FIR filter

Muhammad Bilal Babar	muhammad.babar@mail.mcgill.ca	260963564
Venkata Naga Sai Praneeth Atmuri	venkata.atmuri@mail.mcgill.ca	260952012

Abstract

This report presents a systematic approach to design and implement a 5-band equalizer using linear-phase FIR filters based on DRRS architectures. A detailed explanation of DRRS is presented along with evaluation of different filter lengths to design a particular filter (lowpass, high pass, bandpass). Detailed testing of design was done in MATLAB and various performance parameters are reported. The design algorithm was verified by plotting individual frequency response of each band, and by application of multiple sinusoids along with white noise.

Table of Contents

I. Introduction

A type of equalization which is used for adjusting the tone of an audio system is referred to as tone control. Tone control can be implemented using analog electronics. This method was used in compact disc playback system. However, with advancement in digital field, audio tome control can now be easily implemented using digital signal processing. Berkhout and Eggermont in [1] gave a detailed explanation about the booming digital signaling which reduces the complexity and storage issues that occur in case of analog processing. They have mentioned three different algorithms for digital signal processing in their paper namely echo, (digital) tone control and reverberations. In case of tone control, they had proposed a basic first order filter of the direct form 1. This system uses two loops to reduce the complexity and to increase independence between the lower (bass) and higher (treble) frequencies. However, such recursive systems provide non-linear phase, which is a significant drawback for that system. The independence between bass and treble provides a good tone compensation and provides great flexibility to tone control [2].

FIR filters preserve the phase and provide linear phase, which is essential in many playback audio systems, whereas IIR filters do not provide linear phase. However, in [3] authors used first order IIR filter to control the bass and treble with focus on the speed of the audio to be heard in real time. Lim in [4] proposed use of linear phase second order FIR filter along with frequency masking techniques and demonstrated a tone control unit without any distortions in frequency response. Transfer function for this filter is given as:

$$H(z) = 0.25 + 0.5z^{-1} + 0.25z^{-2} \tag{1}$$

A technique is suggested in [5] to synthesize a sharp linear phase filter with very sparse coefficients which leads to a filter with very low arithmetic complexity. Over the years, Frequency Response Masking (FRM) gained popularity and was used in various applications such as array beamforming [6], hearing aids [7] and software radio [8].

To further reduce the arithmetic complexity, Yang in [9] proposed use of Dual Recursive Running Sum (DRRS) structure which leads to reduction of summations and delay elements. A fourth order multiplication free low pass FIR filter for tone control applications was proposed in [10]. Transfer function of this system is given as:

$$H(z) = 0.125 + 0.25z^{-1} + 0.25z^{-2} + 0.25z^{-3} + 0.125z^{-4} \tag{2}$$

Digital tone control units are used in CD-players and multimedia systems. Moreover, with advancement in the field of digital processing, new approach was proposed in [11] which implements adaptive filtering methods to design a digital equalizer.

The idea of this work is to design linear phase FIR filters using DRRS structure and to apply FRM techniques to build a 5-band equalizer. A brief explanation of DRRS structure is documented in section II. Description of the proposed algorithm using DRRS and FRM techniques along with a brief discussion of audio effects for the designed system is reported in section III. Section IV provides results and discussions while section V concludes the report.

II. Dual Recursive Running Sum (DRRS) Filters

Finite Impulse Response (FIR) filters are mostly required in audio processing applications because of their linear phase property. On contrary to Infinite Impulse Response (IRR) filters, FIR filters provide causality, stability, and linear phase. IIR filters are relatively easy to implement while FIR filters have high computational complexity which presents challenges in their high throughput and low-cost VLSI implementations. Consequently, computationally efficient structures are required for VLSI implementations of FIR filters.

Recursive Running Sum (RRS) filters are among such computationally efficient implementation structures for FIR filters which cause significant reduction in required arithmetic operations. DRRS structure is a cascade of two RRS structures to achieve better attenuation [12]. High pass DRRS filter can be obtained by inversion of low pass DRRS filter. Moreover, bandpass filters can be implemented by subtracting low pass filter from high pass connected in appropriate configurations and by applying frequency masking technique. To explain RRS structure, consider a simple FIR filter with following transfer function:

$$H(z) = \sum_{i=0}^{N+1} z^{-i} \tag{3}$$

Conventional implementation of (3) requires many arithmetic additions, which have a cost in VLSI. Application of geometric series expansion gives:

$$H_{RRS}(z) = \frac{1 - z^{-N}}{1 - z^{-1}} \tag{4}$$

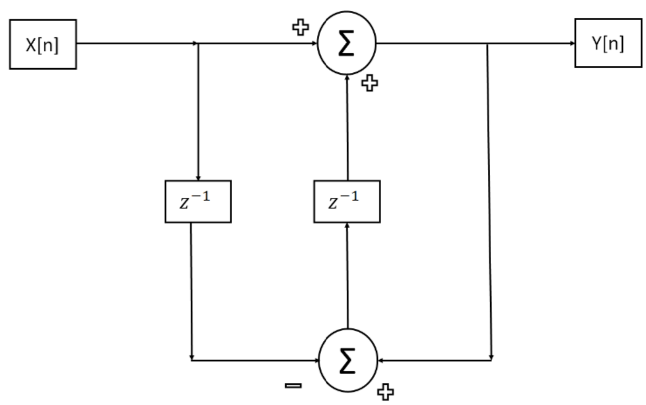


Fig .1. Direct form realization of RRS structure given by (4).

Direct realization of (4) is given in Fig. 1 and is called RRS structure. Regardless of the length 'N' of filter, RRS implementation requires only two arithmetic additions and one multiplication (for unity gain) to generate each output. Its frequency response is given as:

$$H_{RRS}\left(e^{j\omega}\right) = \frac{e^{-j\omega N/2}}{e^{-j\omega/2}} \frac{\sin\,(\omega N/2)}{\sin\,(\omega/2)} \tag{5}$$

For unity gain, a multiplication with $\frac{1}{N}$ is required in (4) and (5). A cascade of two RRS of lengths L_1 and L_2 gives more attenuation in stop band and is called DRRS with following transfer function:

$$H_{DRRS}(z) = \frac{(1 - z^{-L_1})(1 - z^{-L_2}\,)}{(1 - z^{-1})^2} \tag{6}$$

Fig. 2 gives frequency response of low pass RRS and DRRS which shows that DRRS provides more attenuation in stopband as compared to RRS. In addition, DRRS structure gives relatively small transition width although it is still large in comparison with other window-based methods.

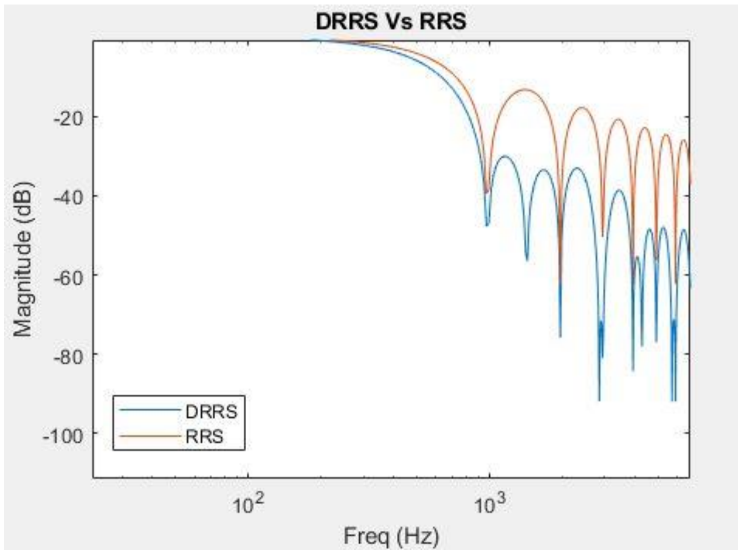


Fig. 2. Frequency response of low pass RRS and DRRS filters.

III. Design Methodology

This section presents detailed design methodology for implementation of a five-band equalizer using DRRS filters. As a first step, frequency bands need to be selected. For this project, following bands were used:

Band 1: 0 Hz – 500 Hz, *Band 2:* 500 Hz – 1.5 kHz, *Band 3:* 1.5 kHz – 4.5 kHz, *Band 4:* 4.5 kHz – 7.5 kHz and *Band 5:* 7.5 kHz – 20 kHz

Since it is audio tone control system, maximum frequency of interest is 20 kHz. Therefore, based on Nyquist rate, a sampling frequency $\boldsymbol{f_s = 44.1\ kHz}$ is used. Realization of frequency bands require 1 high pass, 1 low pass and 3 band pass filters. Respective lengths of DRRS filters are determined based on required cutoff frequencies. For low pass filter with cutoff frequency of 500 Hz, L_1 can be determined from (6) as:

$$|H_{RRS}(e^{j\omega})| = \left|\frac{1 - e^{-j\omega L_1}}{1 - e^{-j\omega}}\right| \tag{7}$$

At $\omega = 2\pi\frac{f}{f_s} = 2\pi\frac{500}{44100}$, magnitude should be zero, which gives:

$$\left|\frac{1 - e^{-j0.071 L_1}}{1 - e^{-j0.071}}\right| = 0 \quad => 0.071 \times L_1 = \pi \; => L_1 \approx 45 \tag{8}$$

As explained in [13], for a -30 dB attenuation in stopband, $L_2 \approx \frac{L_1}{1.5} \approx 31$. Odd values have been selected despite closer even values to simplify the process of making overall system linear phase by adding appropriate delay blocks. Based on (7) and (8), selection of lengths for DRRS structures can be generalized as:

$$K_1 = \frac{44100}{2 \times f} \quad and \; K_2 = \frac{K_1}{1.5} \tag{9}$$

where $\boldsymbol{f}$ is the required corner frequency while $\boldsymbol{K_1}$ and $\boldsymbol{K_2}$ are the lengths of DRRS. Using (9) for remaining frequency band gives following structures:

(0 Hz – 500 Hz): $\mathbf{H}_{\boldsymbol{lp}}(\mathbf{z}) = \frac{\mathbf{1}}{\mathbf{1395}} \times \boldsymbol{DRRS}(\mathbf{45}, \mathbf{31})$

(500 Hz – 1500 Hz): $\mathbf{H}_{\boldsymbol{BP1}}(\mathbf{z}) = \frac{\mathbf{1}}{\mathbf{165}} \times \boldsymbol{DRRS}(\mathbf{15}, \mathbf{11}) - \frac{\mathbf{1}}{\mathbf{1395}} \times \boldsymbol{DRRS}(\mathbf{45}, \mathbf{31})$

(1500 Hz – 4500 Hz): $\mathbf{H}_{\boldsymbol{BP2}}(\mathbf{z}) = \frac{\mathbf{1}}{\mathbf{15}} \times \boldsymbol{DRRS}(\mathbf{5}, \mathbf{3}) - \frac{\mathbf{1}}{\mathbf{165}} \times \boldsymbol{DRRS}(\mathbf{15}, \mathbf{11})$

(4500 Hz – 7500 Hz): $\mathbf{H}_{\boldsymbol{BP3}}(\mathbf{z}) = \frac{\mathbf{1}}{\mathbf{3}} \times \boldsymbol{DRRS}(\mathbf{3}, \mathbf{1}) - \frac{\mathbf{1}}{\mathbf{15}} \times \boldsymbol{DRRS}(\mathbf{5}, \mathbf{3})$

(7500 Hz – 20000 Hz): $\mathbf{H}_{\boldsymbol{hp}}(\mathbf{z}) = \mathbf{1} - \frac{\mathbf{1}}{\mathbf{3}} \times \boldsymbol{DRRS}(\mathbf{3}, \mathbf{1})$

Finally, delay blocks are introduced to make the system linear phase. Overall block diagram of linear phase 5 band equalizer, with gain control, is shown in Fig. 3. Difference equation (time domain representation) is obtained by taking inverse z-transform of the system and MATLAB is used to implement it.

As an extension, an audio effect named ‘pitch’ can be applied to the output of the designed system. It helps an individual to judge sounds as higher or lower in association with musical notes. Decreasing a pitch refers to as decreasing the musical note of the audio. So, change in sampling frequency would result in higher or lower pitch values. For instance, if the sampling frequency is halved then pitch is decreased but the signal lasts twice the original signal. Therefore, in the user interface designed in this project, the user can choose to change the sampling frequency using the slider. Equation (10) is used to compute the new sampling frequency.

$$\boldsymbol{F_s = \alpha * F_s} \quad (\mathbf{10})$$

where $\boldsymbol{\alpha}$ is normalized input taken from user.

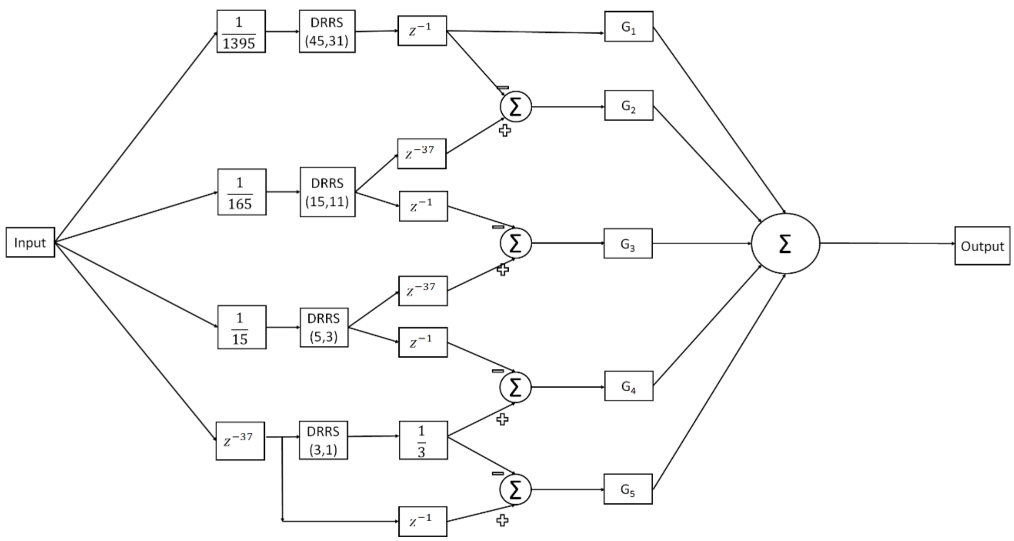
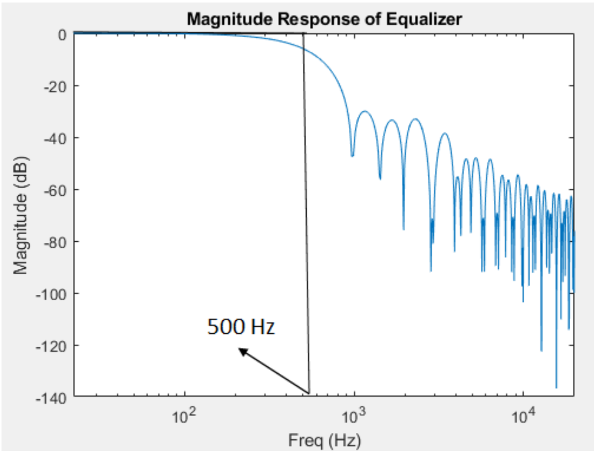


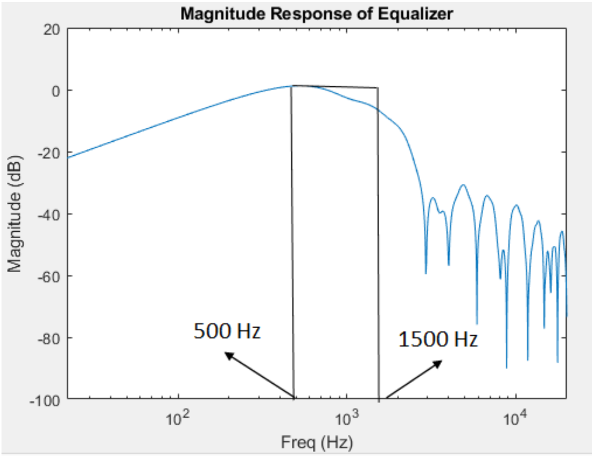
Fig. 3. Bock diagram of 5 band linear phase FIR audio tone control system implemented using DRRS filters.

IV. Results and Discussions

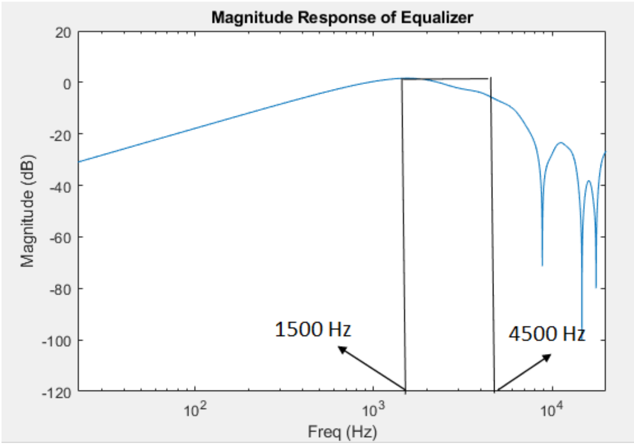
To test performance of the equalizer, it was implemented in MATLAB. As a first step, frequency response of the overall system is plotted, and each frequency band is turned on stepwise. The observed response is compared with ideal rectangular responses. These results are shown in Fig. 4. In Fig. 4 (a), low pass filter (0 Hz - 500 Hz) is turned on with a gain of 0 dB while other bands have a gain of -100 dB. The response shows that this equalizer successfully cuts off frequencies that are beyond 500 Hz and achieves a stopband attenuation of -30 dB. For Fig. 4 (b), only 2nd frequency band (500 Hz-1.5 kHz) is turned on with a gain of 0 dB. Comparison with ideal response indicates that corresponding DRRS based filters achieve required corner frequencies. However, this DRRS structure has large transition widths and passband ripple. Similarly, Fig. 4 (c) and Fig. 4 (d) shows implementations of 3rd and 4th frequency bands. Finally, Fig. 4 (e) shows magnitude response for 5th frequency band which is realized by a high pass filter. Although results indicate that the topology of Fig. 3 successfully implements respective frequency bands, but DRRS structures have long transition widths and high stopband as well as passband ripples, which might make them unsuitable for modern applications. Other sophisticated window-based approaches exist to implement FIR filters which offer small transition widths and ripple but at the cost of added computational complexity.



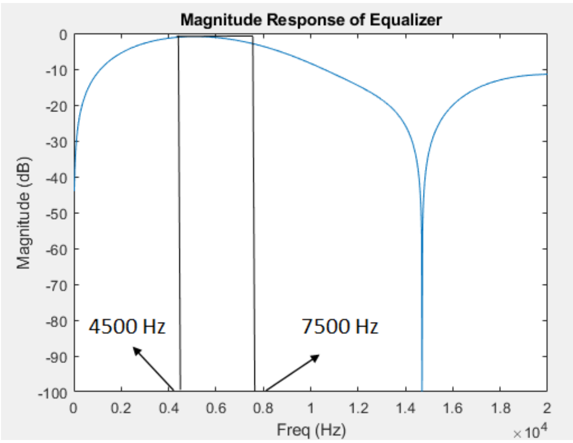
(a)



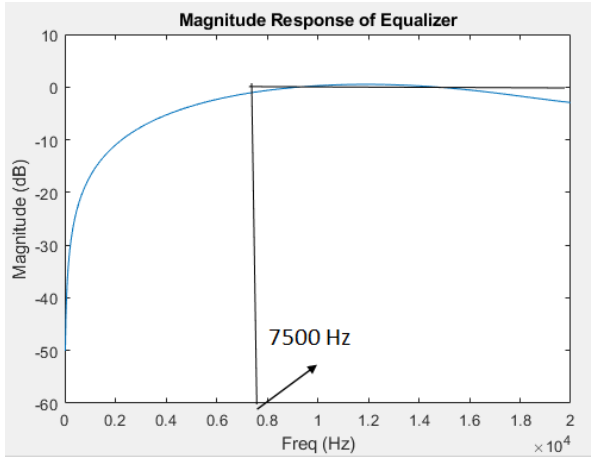
(b)



(c)



(d)

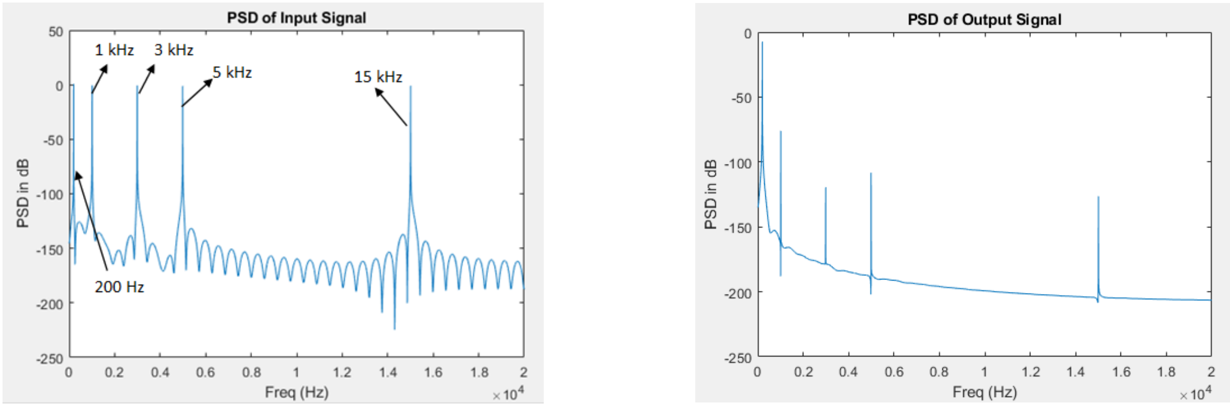


(e)

Fig. 4. Magnitude response of equalizer when only (a) band 1 is active, (b) band 2 is active, (c) band 3 is active, (d) band 4 is active and (e) band 5 is active.

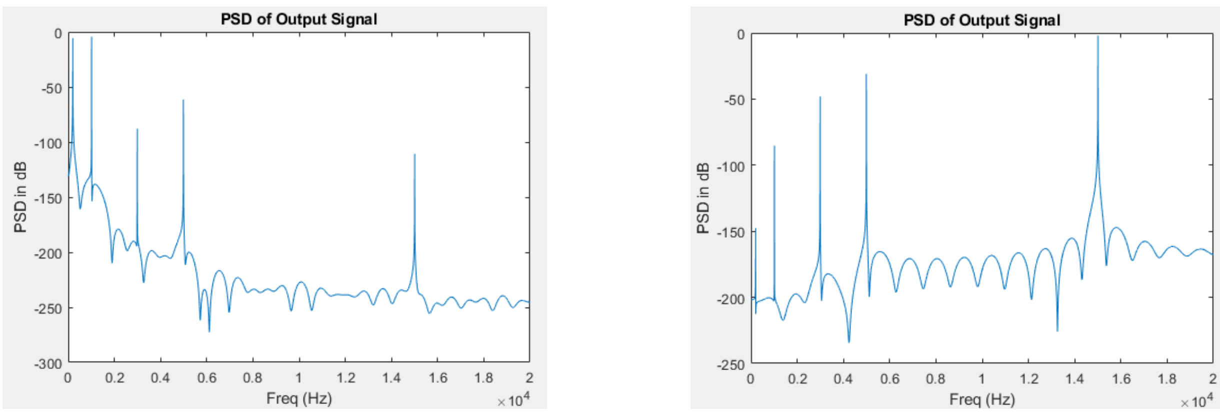
In order to further verify this behavior in time domain, five sinusoids having different frequencies are superimposed so that resulting signal has a frequency component in each band. Power spectral density (PSD) of this input signal is shown in Fig. 5 (a). Fig. 5 (b) shows PSD of output when only 1^{st} band is turned on. The equalizer successfully attenuates frequencies that are above 500 Hz, as a result, only 200 Hz peak is dominant in the output. Fig. 6 (a) plots PSD of output when both 1^{st} and 2^{nd} bands are active while others have a gain of -100 dB. As expected, only first two peaks are retained while other frequencies are attenuated. Finally, only band 5 is turned on and the resulting PSD is plotted in Fig. 6 (b). Once again, only frequency of interest has sufficient gain while others are attenuated.

Lastly, in order to test linearity of the system, a white noise signal is applied when gains of all bands are set to 0 dB. As expected, output is a delayed version of input and is shown in Fig. 7 (a). For better visibility, output is mapped on the input in Fig. 7 (b) which shows that this system can be largely categorized as linear phase. Phase response of this system is shown in Fig. 8 which further strengths this claim of linearity.



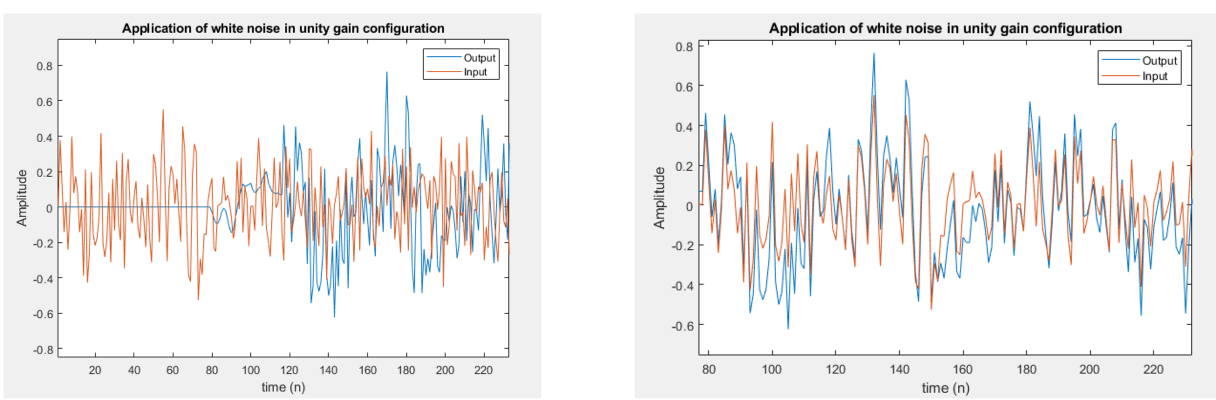
(a) (b)

Fig. 5. Power spectral density of (a) input signal and (b) output signal when $G_1 = 0$ dB and $G_2=G_3=G_4=G_5= -100$ dB.



(a) (b)

Fig. 6. Power spectral density of (a) output signal when $G_1 = G_2 = 0$ dB and $G_3=G_4=G_5= -100$ dB, and (b) output signal when $G_5 = 0$ dB and $G_1=G_2=G_3=G_4= -100$ dB.



(a) (b)

Fig. 7. (a) Application of white noise for 0 dB gain in each band with delayed output response and (b) with output mapped over input for enhanced visibility.

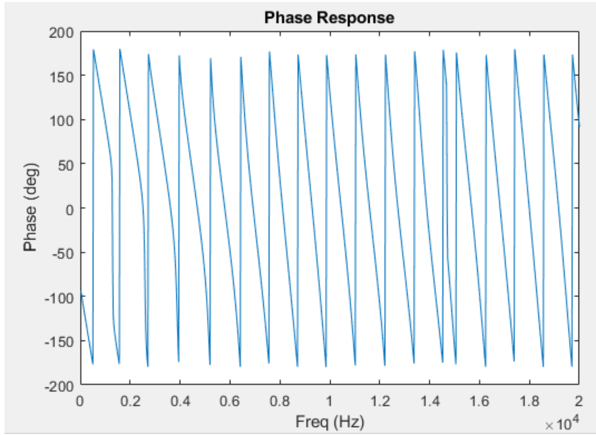


Fig. 8. Phase response of 5 band equalizer.

V. Conclusions

DRRS structures were explored in this project and a multi-band equalizer for audio tone control was developed as a target application of this concept. DRRS based filter designs have less computational complexity. As a result, they can be implemented with fewer resources. However, such filters have wide transition window and high ripples in passband as well as stopband. This might limit their use in modern applications which require more ideal frequency response. These structures need to be compared and analyzed with relatively complex window-based FIR filter design methodologies.

References

1. P. Berkhout and L. Eggermont, "Digital audio systems," in IEEE ASSP Magazine, vol. 2, no. 4, pp. 45-67, October 1985, doi: 10.1109/MASSP.1985.1163755.
2. P. Soardo, "A Versatile Tone Control Circuit and Preamplifier," IEEE Trans. on Audio, vol. 11, pp. 195-201, Nov. 1963.
3. M. Gupta, B. K. Kaushik and L. Chand, "Performance Analysis of Postprocessing Algorithm and Implementation on ARM7TDMI," *2009 International Conference on Computer Engineering and Technology*, Singapore, 2009, pp. 560-564
4. YO. C.. Lim, "Linear-Phase Digital Audio Tone Control," *J. Audio Eng. Soc.*, vol. 35, no. 1/2, pp. 38-40, (1987 January/February.)
5. Yong Lim, "Frequency-response masking approach for the synthesis of sharp linear phase digital filters," in IEEE Transactions on Circuits and Systems, vol. 33, no. 4, pp. 357-364, April 1986, doi: 10.1109/TCS.1986.1085930.

6. Y. Liu and Z. Lin, “On the applications of the frequency-response masking technique in array beamforming,” Circuits, Syst. Signal Process., vol. 25, no. 2, pp. 201–224, Apr. 2006
7. Y. Lian and Y. Wei, “A computationally efficient nonuniform FIR digital filter bank for hearing aids,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 52, no. 12, pp. 2754–2762, Dec. 2005.
8. H. Saleh, E. Zimmermann, G. Brandenburg, and H. Halling, “Efficient FPGA-based multistage two-path decimation filter for noise thermometer,” in Proc. 13th Int. Conf. Microelectron., 2001, pp. 161–164.
9. Rong-Huan Yang, "Linear-phase digital audio tone control using dual RRS structure," in Electronics Letters, vol. 25, no. 5, pp. 360-362, 2 March 1989
10. Y. Lian, and YO. CH. Lim, "Linear-Phase Digital Audio Tone Control Using Multiplication-Free FIR Filter," J. Audio Eng. Soc., vol. 41, no. 10, pp. 791-794.
11. S. Cecchi, L. Palestini, E. Moretti and F. Piazza, "A New Approach to Digital Audio Equalization," 2007 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, New Paltz, NY, USA, 2007, pp. 62-65.
12. Yong Lian and Poh Choo Ho, "ECG noise reduction using multiplier-free FIR digital filters," Proceedings 7th International Conference on Signal Processing, 2004. Proceedings. ICSP '04. 2004., Beijing, 2004, pp. 2198-2201 vol.3, doi: 10.1109/ICOSP.2004.1442214.
13. R. H. Yang, S. B. Chiah and W. Y. Chan, "Design and implementation of a digital audio tone control unit using an efficient FIR filter structure," Proceedings of Digital Processing Applications (TENCON '96), Perth, WA, Australia, 1996, pp. 273-277 vol.1, doi: 10.1109/TENCON.1996.608815.

Appendix – I

Instructions to run user application

1. Download and extract the “DSP_Venkata_Muhammad.zip” file.
2. The songs (wav format) and ‘ToneControl_2017.mlapp’ files should be in same directory (instrumental music files with .wav format gives better results).
3. Open the ‘ToneControl_2017.mlapp’ file using MATLAB 2017a. If it doesn’t work, open ‘ToneControl_2018.mlapp’ file with MATLAB 2018a or higher. On opening, a new window pops up. Click the run button from top menu as shown in Fig. 1.
4. The above step results to an equalizer window shown in Fig. 2. On opening of the equalizer window, load the desired music file.
5. Please note that changing the gains and pressing the play button without loading the file can lead to error.

6. After loading the music file, the equalizer window might disappear, but it can be accessed from the taskbar (the window just minimizes but will not be closed).
7. Once the file is loaded, change the gains of different bands and press the play button to play the music. Please wait till it gets played. Pressing multiple times “play” without allowing it to get played, would cause problems.
8. On the left side of the equalizer, pitch effect is available. To access this effect for the designed system, click the checkbox and move the slider. After this, pressing the play button, plays the music (the slider for pitch here is not in dB).

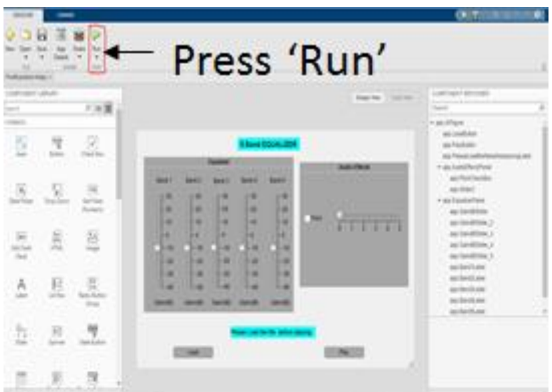


Fig. 1. Selecting ‘Run’ option from top menu.

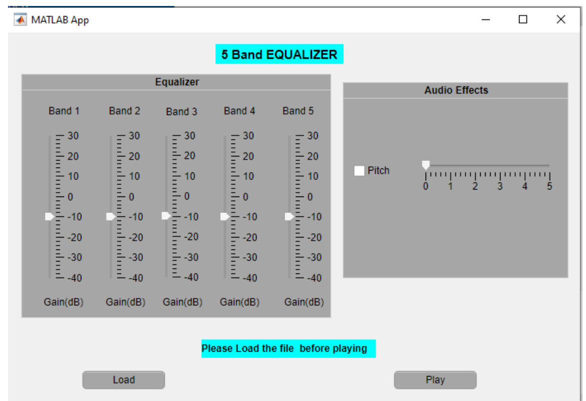


Fig. 2. Main window of GUI.

Appendix -2

Code for difference equations only (complete code is attached as .mlapp file)

```
function PlayButtonPushed(app, event)
%% Main code where the computation occurs %%

    [u,fs]=audioread(app.file_name); %read the audio file
     x=u;
     x=x';
     matrixSize = size(u); % To Match the matrix size of the input signal
     x=[zeros(matrixSize(2),77) x];   %for proper indexing in MATLAB
     N=length(x);
     y=zeros(1,N); %initializing output signal

            %Filter calculations
```

```
for n=78:1:N
  %DRRS to produce a lowpass filter
   DRRS_45_31_lp = (1/1395)*(x(1,n-1)-x(1,n-46)-x(1,n-32)+x(1,n-77));

  %DRRS to produce a bandpass-1 filter
   DRRS_15_11_bp1_2 = (1/165)*(x(1,n-37)-x(1,n-52)-x(1,n-48)+x(1,n-63));

  %DRRS to produce a  bandpass-2 filter
   DRRS_15_11_bp2_1 = (1/165)*(x(1,n-1)-x(1,n-16)-x(1,n-12)+x(1,n-27));
   DRRS_5_3_bp2_2 = (1/15)*(x(1,n-37)-x(1,n-42)-x(1,n-40)+x(1,n-45));

  %DRRS to produce a  bandpass-3 filter
   DRRS_5_3_bp3_1 = (1/15)*(x(1,n-1)-x(1,n-6)-x(1,n-4)+x(1,n-9));
   DRRS_3_1_bp3_2 = (1/3)*(x(1,n-37)-x(1,n-40)-x(1,n-38)+x(1,n-41));

  %DRRS to produce a highpass filter
   DRRS_3_1_hp = DRRS_3_1_bp3_2;

% Following equation gives final output in time domain, gains are obtained from
user.
% Here app refers to the object created default by MATLAB using which gains are
obtained.

y(1,n)=app.G1*(DRRS_45_31_lp)+ app.G2*(DRRS_15_11_bp1_2-
  DRRS_45_31_lp)+app.G3*(DRRS_5_3_bp2_2-DRRS_15_11_bp2_1)+app.G4*(DRRS_3_1_bp3_2-
  DRRS_5_3_bp3_1)+app.G5*(x(1,n-38)+x(1,n-40)-2*x(1,n-39)-DRRS_3_1_hp)-y(1,n-
  2)+2*y(1,n-1);

      end
      y=y'; % For proper orientation

            %%%%%%%%%%%%%%%%%%%%%% AUDIO EFFECTS %%%%%%%%%%%%%%%%%%%%%%%%%%%%%

            %% Pitch
            if(app.pitch) %if user checkboxes this effect
                a = sqrt(2)^(app.pitchValue);  % Normalizing the input obtained
             from user
                p = a*fs;
                fs = p;
            end
            clear sound % Clears the previous music.
            sound(y,fs); % Play the output
        end
```