Subwoofer Frequency Response Optimization by Means of Active Control

Luis Dominguez[1]
California Polytechnic State University, San Luis Obispo, CA, 93407

Most subwoofer systems have difficulties producing frequencies in the low end of the hearing spectrum due to the added power requirements and instabilities. Active controls can transform the audio signal without changing physical characteristics and ultimately generating a more impressive audio system. A Linkwitz transform crossover was implemented to extend the low end frequency response of a sealed enclosure. A graphical user interface in MATLAB was written to aid in selection of components, driver and enclosure volume. The circuit board was built and integrated with a home theater system inside of a couch and tested with a Real Time Analyzer. The Linkwitz crossover was shown to extend the frequency response, transient response and improve the subwoofer system while reducing the required enclosure volume.

Nomenclature

C	=	Capacitor
dB	=	Decibel
F	=	Frequency
$H(s)$	=	Transfer Function
k	=	Gain
Q	=	Relative Damping Factor
R	=	Resistor
V_{as}	=	Compliance
β	=	Damping Factor
φ	=	Phase Angle
ω	=	Angular Frequency

Subscripts:

1-3	=	Number value For R and C
3	=	At – 3dB (Cutoff Frequency)
b	=	Enclosure Volume
c	=	Driver Enclosure
s	=	Speaker
t	=	Total Value
z	=	Transformed Value

I. Introduction

The word acoustics is derived from the Latin word *akoustikos* meaning "of or for hearing." From an early era the interest and study of electromagnetic wave propagation across the human hearing spectrum has been of great interest. The human hearing system has the tremendous ability to decipher the smallest detail in the characteristics of sound reproduction. Musicians and thespians benefited the most from the preliminary science of acoustics. Nowadays, almost everyone benefits from advances in acoustics due to the ability to record and playback audio signals.

[1]Undergraduate Student, Aerospace Engineering Department, 1 Grand Avenue, San Luis Obispo CA, 93407.

A basic audio system starts with an audio signal source, typically from a recording or microphone. Next, the signal will travel to a pre-amplifier (if used) which uses active means to alter the signal, also known as an active crossover. The signal then moves towards the amplifier where the signal amplitude is increased. Passive crossovers then take their role by separating the audio signal appropriately in order to pass frequencies that are within the loudspeaker range. The signal travels to the speaker system, which is composed of three fundamental speakers: tweeters, midrange drivers and subwoofers. Tweeters reproduce high frequencies typically 2000 Hz to 20 kHz, the midrange driver reproduces sound waves from 300 Hz to 5000 Hz and the subwoofer from 20 Hz to 400 Hz. Once the signal is reproduced properly it can be tested using a microphone in conjunction with a Real Time Analyzer (RTA) or the ultimate test, the human ear.

An audio system that lacks adequate audio representation in specific frequencies can easily be detected by ear which is heard by a decrease in volume or Sound Pressure Level (SPL) as different notes are played. This is most apparent in low frequencies when a lot of cone movement exists but not much is heard or felt. However, a poorly designed speaker will generate a third harmonic of a low frequency that cannot be reproduced properly. For example, if a 20 Hz tone is being generated a 60 Hz tone might be heard due to the inadequate mechanical system in the loudspeaker. For most audiophiles the ultimate test of a subwoofer is the lowest tone that can be produced properly at -3 dB, also known as the cutoff frequency (F_3). The reason this test dictates subwoofer performance is due to the increase in power and cone excursion required to produce low frequency tones with minimal losses in SPL. Even though most music does not go below 30 Hz, a system with these capabilities will enhance the realism of music with pipe organs and large percussion instruments. Lower frequency reproduction will also be enjoyed in movies during crashes and explosions further engaging the viewer.

Various methods exist to increase low end frequency response in a subwoofer system, which can be grouped into two categories: passive and active systems. Passive systems do not require a power supply to be integrated with the subwoofer enclosure and active systems need an appropriate power supply. The most basic type of enclosure is a sealed enclosure shown in Fig. 1 and as the cone traverses along the pole piece the compression and rarefactions of air molecules is interpreted as sound. The most common passive system to extend low end frequency response is known a bass reflex or vented enclosure shown in Fig. 2. This method uses a vent, typically cylindrical or rectangular duct, where the vent is tuned to a specific fundamental frequency towards the low end of the subwoofer frequency response. As a pressure differential is generated inside the enclosure and some of the air is moved towards the vent but does not become beneficial until within range of its fundamental frequency. At the air vent's fundamental frequency a piston of air is formed and oscillates inside the vent producing the tone. The loudspeaker/vent system work in conjunction to increase low frequency reproduction, having a roll-off slope of 12 dB/dec. The trade off of a vented enclosure is a delayed transient response which is apparent in music with rapid low frequency changes such double bass drums and fast bass guitar melodies. Various configurations exist using ported system, some involving multiple ports and chambers but require a larger enclosure volume to function properly and careful craftsmanship to acquire desired results. Another method is known as a passive radiator. It can be classified as a ported system but instead of using a piston of air to generate the sound a more dense material is substituted, changing the fundamental frequency of the opening. The passive radiator is composed of an opening occupied by a material usually similar to the subwoofer which is shown in Fig. 3. The most common uses a drone loudspeaker without electrical or magnetic components oscillating out of phase with the loudspeaker. The drone cone is tuned by changing its mass and can be modeled very similar to vented enclosure. Passive radiator systems have the ability to produce a lower

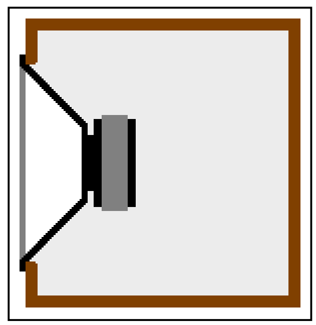

Figure 1. Sealed Enclosure

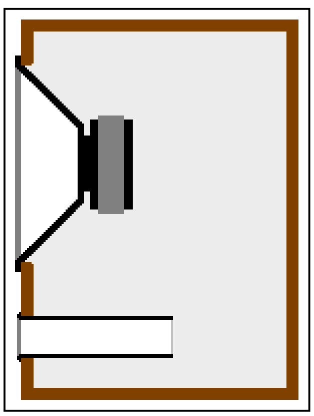

Figure 2.Bass Reflex Enclosure

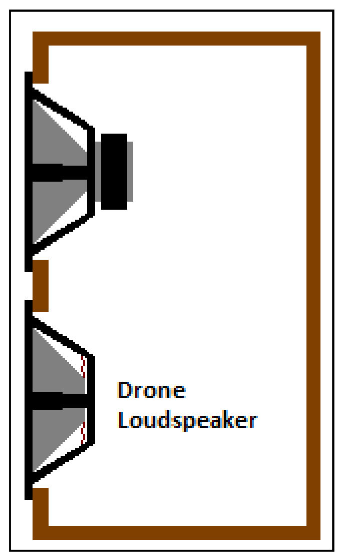

Figure 3. Passive Radiator Enclosure

cutoff frequency with smaller enclosure volumes compared to a vented system. The roll off slope is higher than a bass reflex enclosure and will typically have a better transient response. The maximum sound pressure level will typically be lower and the system is more susceptible to discrepancies. The building process also requires more attention to schematics since errors in craftsmanship can alter performance dramatically.

Active systems have the advantage of adapting accordingly and the signal can be transformed or left unaltered when needed. When combined with the accurate sound reproduction of a sealed enclosure the frequency response can be extended by transforming the cutoff frequency and system Q value or relative damping. The system Q is composed of the mechanical and electrical system at resonant frequency, Fs. This concept is the foundation of the Linkwitz transform circuit designed by Siegfried Linkwitz. In order to extend the low end frequency response a 12 dB/decade highpass crossover will be implemented, also known as a Linkwitz transform crossover circuit shown in Fig. 4 to the right. Loudspeaker behavior models a second order system having a roll off of 12dB/decade with a pair of zeroes on the s-plane origin and a pair of complex poles defined by the resonant frequency (Fs) and total Q value (Qts). By using a second order crossover (12dB/decade) the roll off will be similar to a passive sealed enclosure maintaining natural characteristics. The Linkwitz transform circuit places a pair of complex zeroes on top of the complex pole pair of the subwoofer to counteract their effects. The new pair of poles can be placed in the desirable location which is defined by the desired Fs and Q values, transforming the frequency response. The transient response is also improved which is unlike any of the passive system where transient response is reduced when frequency response is extended.

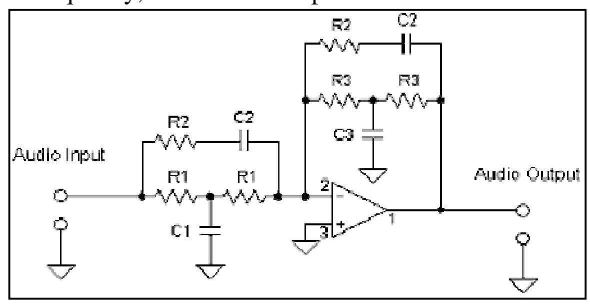

Figure 4. Linkwitz Transform Circuit

II. Analysis

A. Equations

The loudspeaker parameters can be attained from the manufacture and the desired Fs and Q value of the transformed system are achieved by calculating the proper resistor and capacitor values in the Linkwitz transform circuit. The subwoofer and sealed enclosure frequency response must be understood prior to transforming the Fs and Q values. The loudspeaker free air resonance Fs, total compliance Qts and desired enclosure volume must be known. The Q value can then be calculated by Eq. (1)

$$Q_{tc} = \frac{F_{sc} * Q_{ts}}{F_s} \tag{1}$$

The subwoofer enclosure resonance is defined by

$$F_{sc} = F_s\sqrt{\left(\frac{V_{as}}{V_b} + 1\right)} \tag{2}$$

And the cutoff frequency can be calculated by Eq. (3).

$$F_3 = F_{sc}\sqrt{\frac{\left(\left(\frac{1}{Q_{tc}^{2}}-2\right)+\sqrt{\left(\frac{1}{Q_{tc}^{2}}-2\right)^{2}+4}\right)}{2}} \tag{3}$$

The relative dB is calculated by Eq. (4) where F is the inputted frequency varying across desired range in order to plot the response of the subwoofer/enclosure system.

$$dB = 20\left(\frac{\log\frac{\left(F/F_{sc}\right)^{2}}{\sqrt{\left(\left(F/F_{sc}\right)^{2}-1\right)^{2}+\left(\frac{F/F_{sc}}{Q_{tc}}\right)^{2}}}}{\log 10}\right) \tag{4}$$

Once a target system resonance Fz and target relative damping Qz are chosen the component values can be calculated. For simplicity a value of the second capacitor will be chosen in order to calculate the remaining component values. Choosing a capacitor value in microfarads allows the following to be calculated. Resistor values are calculated in kΩ as shown below in Eqs. (5-7).

$$R_1 = \left(\frac{1}{6.2832 * F_{sc} * {}^{C_2}/_{1e6} * (2 * Q_{tc} * (1+k))}\right) * \frac{1}{1000} \tag{5}$$

$$R_2 = 2 * k * R_1 \tag{6}$$

$$R_3 = R_1 * \left(\frac{F_{sc}}{F_z}\right)^2 \tag{7}$$

The capacitor values in microfarads are calculated as shown in the following equations.

$$C_1 = C_2 * (2(Q_{tc}) * (1+k))^2 \tag{8}$$

$$C_3 = C_1 \left(\frac{F_z}{F_{tc}}\right)^2 \tag{9}$$

Now, the frequency response and phase response can be calculated for the crossover and combined system. The damping factor and angular frequency are calculated by Eqs. (10-11).

$$\beta = \frac{0.5}{Q_{tc}} \tag{10}$$

$$\omega = 2\pi f \tag{11}$$

Next the magnitude of the crossover must be calculated which is shown in Eq. (12)

$$dB_{cross} = 20\log\left(\left(\frac{\frac{R_3}{R_1}\sqrt{\left(1-\omega^2 C_1 C_2 R_1{}^2\right)^2 \left(\omega C_2 (R_2+2R_1)\right)^2}}{\sqrt{\left(1-\omega^2 C_2 C_3 R_3{}^2\right)^2 + \left(\omega C_2 (R_2+2R_3)\right)^2}}\right)\left(\frac{\sqrt{4+(\omega R_3 C_3)^2}}{\sqrt{4+(\omega R_1 C_1)^2}}\right)\right) \tag{12}$$

The phase angle in degrees is calculated by Eq. (13)

$$\varphi_{cross} = \frac{-180}{\pi}\left(\left(\tan^{-1}\left(\frac{\omega C_3 R_3}{4}\right)\right) - \tan^{-1}\left(\frac{\omega C_1 R_1}{4}\right) + \tan^{-1}\left(\frac{2\omega C_2 R_1 R_2}{\left(1-\omega^2 C_1 C_2 R_1{}^2\right)}\right) - \tan^{-1}\left(\frac{\omega C_2 (R_3+2R_3)}{\left(1-\omega^2 C_2 C_3 R_3{}^2\right)}\right)\right) \tag{13}$$

The loudspeaker frequency response and phase angle are calculated accordingly with the same varying frequency which is dependent on the angular frequency shown below.

$$dB_{speaker} = 20\log\left(\frac{\left(\frac{f}{F_{tc}}\right)^2}{\sqrt{\left(1-\left(\frac{f}{F_{tc}}\right)^2\right)^2 + \left(2\beta\frac{f}{F_{tc}}\right)^2}}\right) \tag{14}$$

$$\varphi_{speaker} = \frac{180}{\pi}\tan^{-1}\left(\frac{2\beta\frac{f}{F_{tc}}}{1-\left(\frac{f}{F_{tc}}\right)^2}\right) \tag{15}$$

The combined magnitude and phase angle can be calculated by Eqs. (16) and (17) respectively.

$$dB_{Total} = dB_{cross} dB_{speaker} \tag{16}$$

$$\varphi_{Total} = \varphi_{cross} + \varphi_{speaker} \tag{17}$$

The equations presented were implemented in a Graphical User Interface (GUI) in MATLAB in order to easily analyze the response of various loudspeakers, desired components and transformed values of Fs and Q in order to acquire the optimum frequency response of the transformed subwoofer system.

B. Transfer Function Analysis

The response in the previous section can also be obtained by using transfer function analysis, which is an advantage in the MATLAB environment. Using MATLAB's built in controls toolbox results can be generated for the various drivers and desired parameters. The transfer function for the driver and Linkwitz crossover must be known in order to calculate the combined system response. When considering only the driver and enclosure the resulting transfer function is shown in Eq. (18).

$$H(s)_{driver} = \frac{s^2}{s^2 + \frac{2\pi F_p}{Q_p}s + \left(2\pi F_p\right)^2} \tag{18}$$

Next the transfer function of the Linkwitz transform crossover is calculated by Eq. (19)

$$H(s)_{crossover} = \frac{s^2+\frac{2\pi F_p}{Q_p}s+\left(2\pi F_p\right)^2}{s^2+\frac{2\pi F_z}{Q_z}s+(2\pi F_z)^2} \tag{19}$$

Multiplying Eqs. (18) and (19) results in the transfer function of the combined transformed system.

$$H(s)_{combined} = \frac{s^2}{s^2+4\pi F_z s+(2\pi F_z)^2} \tag{20}$$

III. Theoretical Results

A. Component Selection

Various loudspeaker and amplifier parameters must be considered in order to ensure the system can deliver the low end performance. Low frequency sound wave reproduction requires larger amounts of air displacement dictating subwoofer requirements primarily cone surface area, cone excursion and power handling capabilities. The amplifier must be able to produce the extra power required, which is inversely proportionate to the frequency, and be able to dissipate the excess heat accordingly. Smaller maximum cone excursion may result in cone flexing causing distortion when limitations are reached. Also, insufficient power will limit the subwoofers mechanical system and ultimately not reproducing the sound wave the signal source intended. These parameters are all crucial to the selection of the subwoofer and amplifier which greatly affect the system performance.

B. Graphical User Interface

The purpose of the GUI is to aid in component calculations in order to optimize desired frequency response. The user must input the loudspeaker Vas, Fs, sealed enclosure volume and C_2 value which generates the frequency response, phase response and component values. By analyzing various possibilities a more desirable system can be built for a more enjoyable listening experience. The layout of the GUI is shown Fig. 5 without the graphs.

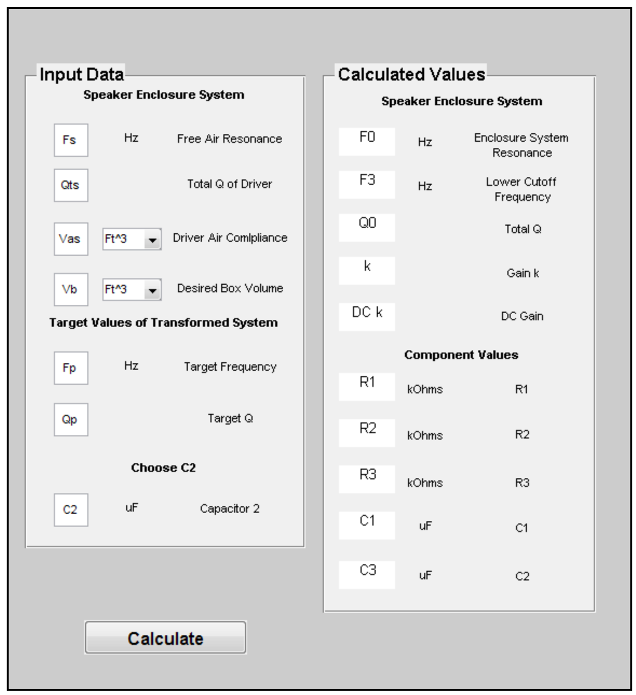

Figure 5. Input and Calculated Values of GUI

The left column is the data inputted by the user which are the loudspeaker parameters, desired transformed values and the second capacitor value. Once the "calculate" push button is selected the components, frequency response and phase response will be calculated and plotted for visual comparison. The k and DC k values are ratios of the original system to the new transformed system and serve as a comparison in order to eliminate a transformed system that cannot be physically built by its initial and desired parameters. Therefore k should be a positive value and DC k should not exceed a value of 20. C_2 can also be varied in order to raise or lower the other circuit component values in Linkwitz transform circuit. This option generates flexibility by giving the option to use resistor and capacitor values that are more common.

Since the human hearing is limited to the range of 20 Hz to 20 kHz an ideal audio system will have a flat response across the entire human hearing spectrum. However, the actual listening range varies between each person, especially with the elderly. In order to show the capabilities of the Linkwitz transform circuit the goal of the system is to achieve a target frequency of 20 Hz with an undersized enclosure volume when compared to the recommended manufacture sealed enclosure volume. Decreasing the enclosure volume causes the resonant frequency to increase therefore, truly testing the crossover performance. The size of the driver, Fs, V_{as}, P_{RMS}, impedance and price were all considered during driver selection. After implementing candidates into the program the Alpine SWE-1243 12" Type E subwoofer with a 4 ohm voice coil generated the best results considering the price.

Fig. 6 shows the inputted values and the calculated results for a Type E Alpine Subwoofer in a 0.75 Ft^3 enclosure volume.

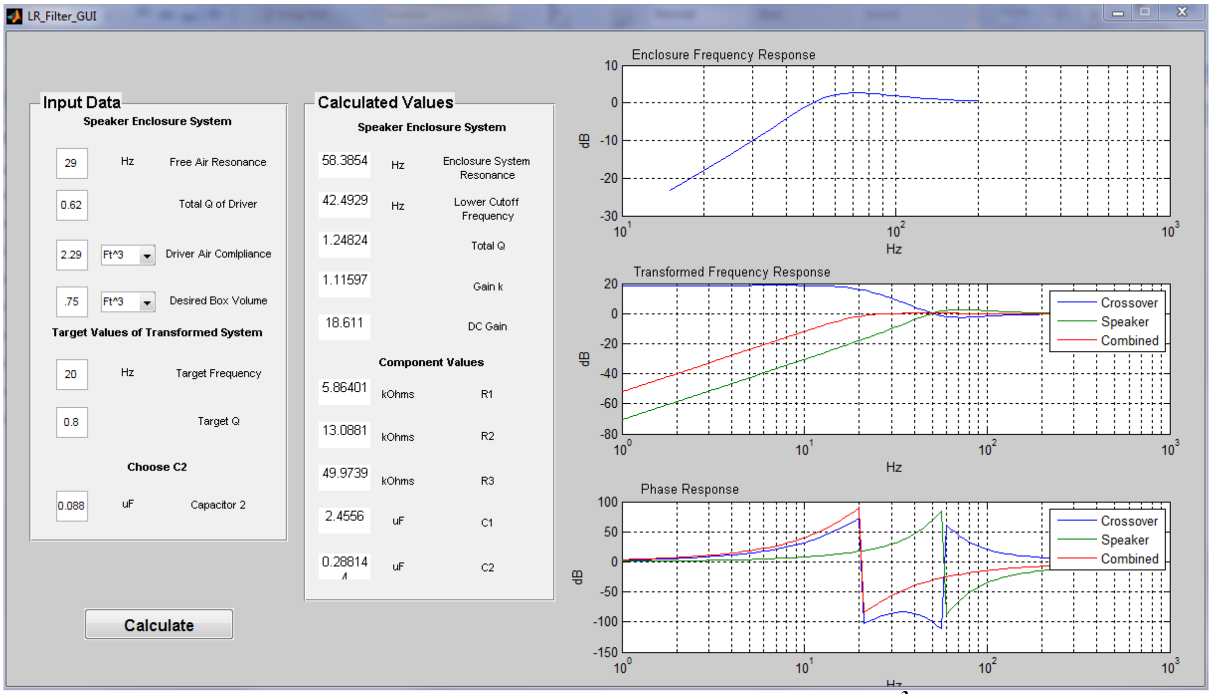

Figure 6. GUI with calculated values for an 12 "Alpine Type E Subwoofer with a 0.75 Ft^3 Enclosure Volume

The exact component values will not be attainable but if actual values are within 5% from the calculated value, desirable functionality will be maintained. From the frequency response of driver system and combined system, it is apparent the desired response can be reached with a DC gain of 18.61 and k value of 1.116 showing that it is physically possible. Maximum SPL level is sacrificed which is shown by a decrease in the hump prior to the cutoff frequency of the subwoofer and enclosure system without the crossover. Like with most designs a series of tradeoffs must be considered and here SPL is being traded for decrease in cutoff frequency which is reduced from 42.5 Hz to 20 Hz. The trade off is of added benefit since an increase in SPL at a specific frequency will result in unequal volume across the response. The roll off slope of the combined system is unaltered preserving the natural characteristic of 12dB/decade.

The resonant frequency and total Q of the loudspeaker are based off electrical and mechanical components and cannot be physically changed by the active network. The input signal is altered or transformed which is noticeable in the phase response showing a signal shift at the locations of the original and transformed cutoff frequency.

C. Transfer Function Results

Generating the system transfer function will yield information that is crucial to understanding the system behavior. The transfer function of the driver, crossover and combined system were shown previously in Eqs. (18-20). Using the controls toolbox in MATLAB various plots were generated for the driver, crossover and combined system. While the GUI aids in choosing adequate driver, enclosure volume and crossover components to achieve the desired frequency response, the values can then be inputted into the transfer function as a supplementary analysis. The area of interest is the frequency response of the transformed system but transfer function analysis aids in the comparison of the previous method for frequency response calculations along with transient response and root locus techniques. The resonant frequency (F_P =58 Hz) and damping factor (Q_p=1.25) of the driver and enclosure system were inputted into Eq. (18) yielding the transfer function shown below

$$\mathrm{TF}_{\mathrm{driver}} = \frac{s^2}{s^2+291.5s+1.328e5} \tag{21}$$

Combing F_p and Q_p with the desired transformed values of F_z =20 Hz and Q_z=0.8 Eq. (19) is calculated to be

$$TF_{crossover} = \frac{s^2+291.5s+1.328e5}{s^2+157.1s+1.579e4} \tag{22}$$

The combined transfer function for a single input single output (SISO) system is shown in Eq. (23).

$$TF_{combined} = \frac{s^2}{s^2+251.3+1.579e4} \quad (23)$$

The pole-zero and root locus plots are shown in Fig. 7, which shows how the Linkwitz transform crossover alters the input to the desired output serving as an open loop signal controller. The driver and enclosure system is composed of two complex poles and a set of real zeros. The Linkwitz filter takes the input of the driver system and cancels the driver complex poles by placing two complex zeros on top which is apparent in Fig. 7. A set of poles are placed closer to the real axis to achieve the desired combined system with real zeros at the origin and real poles. A similar response can be generated with a lower damping factor lowering the Linkwitz filter poles even more towards the real axis yielding better transient response but the value of $Q_z = 0.8$ was chosen to retain more natural characteristics of sound reproduction. A lower Q value will dampen at much faster rate but will have a higher resemblance to a square wave which is an unnatural sound by acoustic musical instruments.

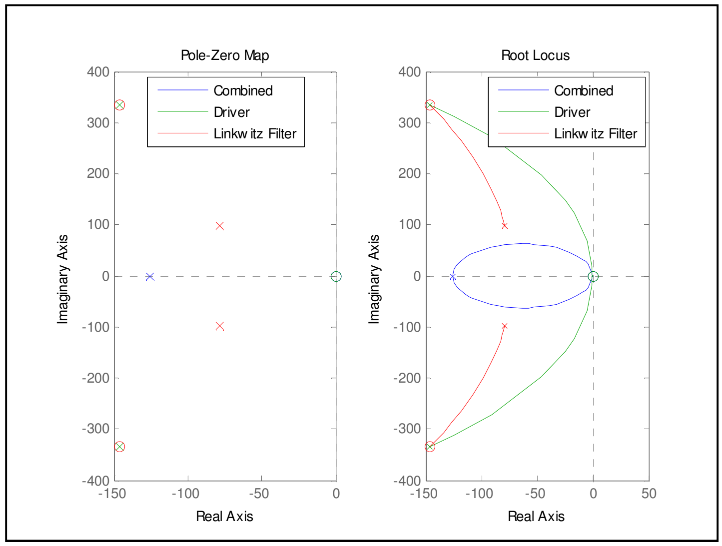

Figure 7. a. Pole Zero Plot b. Root Locus Plot

The step and impulse response of the system is essential to making improvements of the acoustic characteristics. In passive systems the trade off to gain low end frequency response and overall SPL is a delayed transient response and less accurate sound reproduction. Using a Linkwitz transform circuit in conjunction with a sealed enclosure sacrifices overall SPL to extend frequency response but does not sacrifice the transient response of the system. Figure 8 shows the step and impulse response using the same values of F_z and Q_z.

The combined system has a smaller overshoot in both plots which indicates a more accurate sound reproduction and faster response. Various Q_z values were used to view changes in step response. It was found that the settling time can be decreased by decreasing the value of Q but resulted in dramatic changes in the phase response generating audible distortion. The values chosen result in a desirable overall response and improvements in all areas covered besides sound pressure level which is an expected tradeoff.

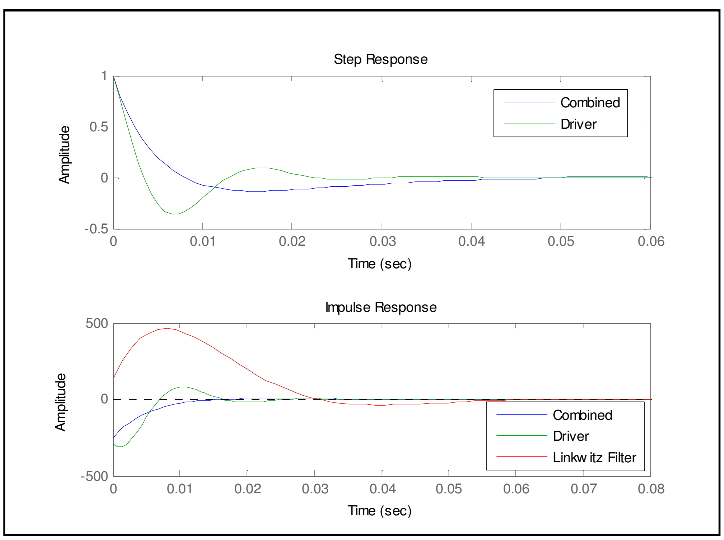

Figure 8. a. Step Response b. Impulse Response

IV. Building Crossover and System Integration

Once the system reaches a satisfactory response the Linkwitz transform crossover can be built for integration and test with an audio system. The final circuit used is composed of the Linkwitz transform circuit along with a signal phase inverter to gain greater control during the audio tuning process. The circuit is shown in Fig. 9 below.

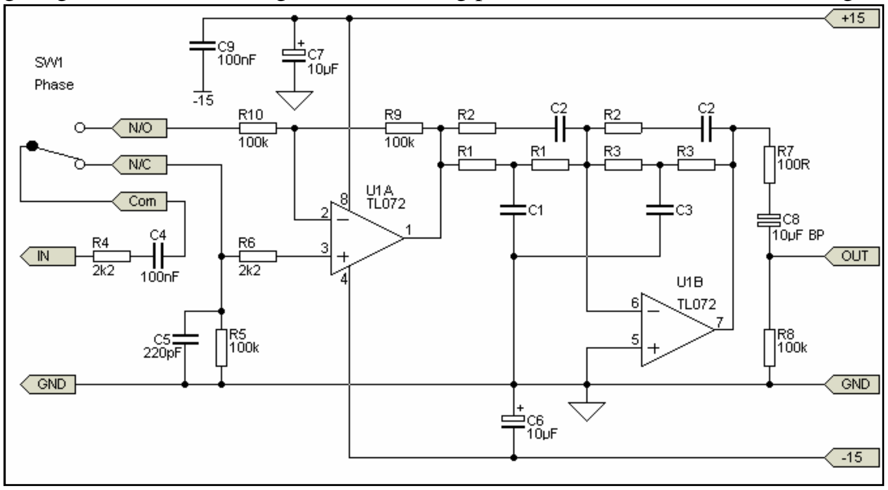

Figure 9. Linkwitz Transform Crossover with Phase Inverter

The normally closed operates in a 0 degree phase and normally open is 180 degrees out of phase of the input signal. The input signal is connected on the left and output signal is on the right of the circuit noted by OUT and GND. The circuit requires a suitable power supply with +/- 15 V output. So the following power supply was implemented using a 16 VAC @ 1amp wall transformer as the supply leads to the power supply. The circuit diagram is shown below in Fig. 10 for clarification. The power supply components were selected based on the desired output voltage and the auxiliary output can be used to supply a relay. This circuit is a common power supply designed by Rod Elliot of Elliot Sound Products, which uses a positive and negative voltage regulator and R4A/B and R6A/B dictate the output voltages which were chosen to be 2.2 kΩ to generate the desired +/-15 V.

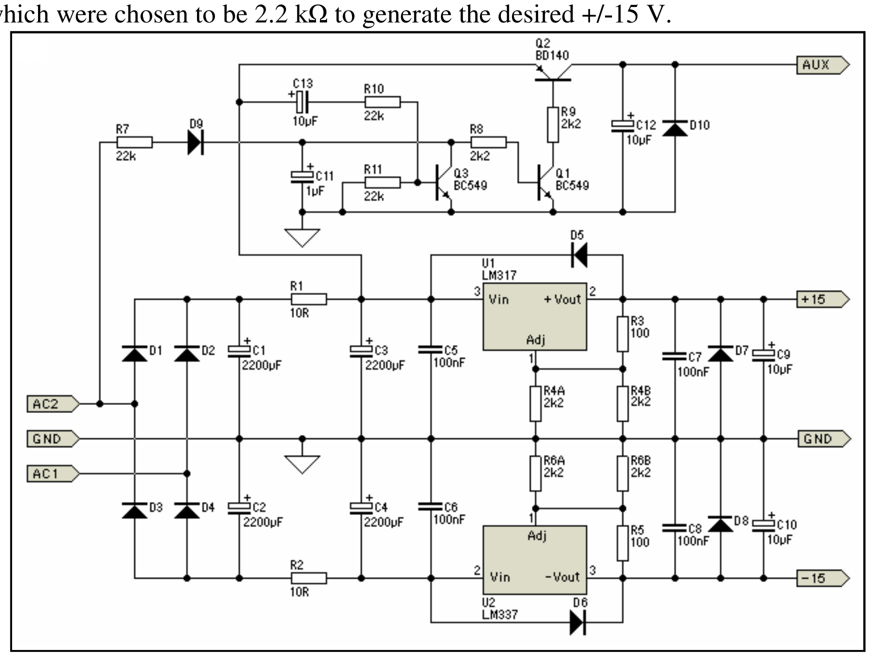

Figure 10. Power Supply Circuit for Crossover

The printed circuit boards were supplied by Elliot Sound Products and components were purchased at The Electronic Parts Supermart in Santa Maria, California to desired specification. Audio systems are very sensitive to electrical noise so precise and high quality components were considered for the circuit. 1% tolerance resistors were preferred but due to the limited supply of values in the local area 2 % tolerance were purchased. The actual values versus calculated resistor and capacitor values are shown in table 1 along with the percent discrepancy.

Component	Calculated Value	Value Used	% Error
R_1	5.86 kΩ	5.9 kΩ	0.68 %
R_2	13.06 kΩ	13 kΩ	0.46%
R_3	49.99 kΩ	49.9 kΩ	0.18%
C_1	2.459 µF	1 + 1 + 0.47 µF	0.447 %
C_2	0.088 µF	0.08+0.0068 µF	0.9 %
C_3	0.288 µF	0.22 + 0.06 µF	2.78 %

Table 1. Calculated and Actual Component Values

A goal of 5 % discrepancy was the requirement for resistor and capacitor selection which was achieved by combining resistors in series and capacitor in parallel noted by a + sign in the “value used” column. By combining components the 5% tolerance requirement was easily achieved. The other components used are listed in the appendix corresponding with circuit number in the diagram.

The components were measured and soldered to the printed circuit boards and the final completed boards are shown in Fig. 11 below. The top board is the power supply and the bottom is the Linkwitz transform circuit used for the final integration.

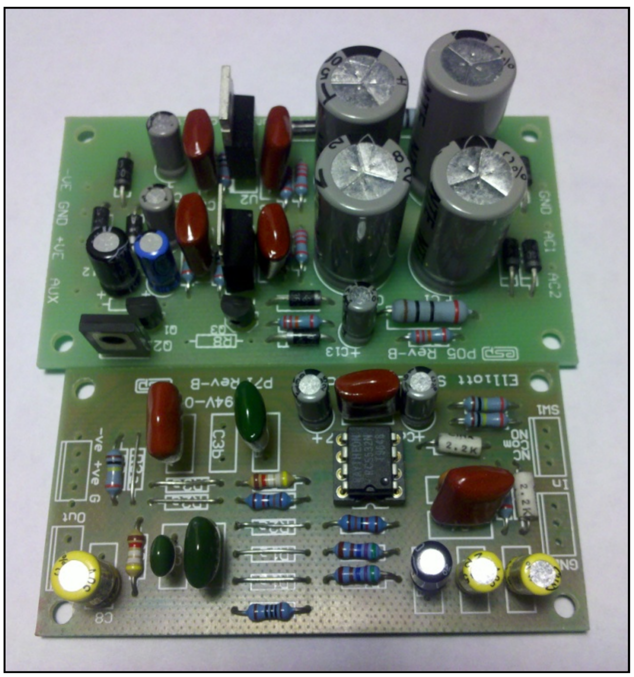

Figure 11. Completed Linkwitz Crossover and Power Supply

Prior to connecting the unit to the subwoofer amplifier, the voltage output was measured on the power supply and a dummy load was connected to the crossover to ensure adequate performance.

To truly enjoy the transformed subwoofer system the Linkwitz crossover was integrated with a home theater system to create a unique audio experience. Since lower frequencies are usually felt more than heard, building an audio system inside of a couch can generate great results even at low volumes. Due to the limited internal space of a couch generates a great application where low frequencies can be properly produced with an undersized sealed driver enclosure.

One of the biggest considerations was price versus performance of home theater supplementary components. After various considerations, researching products and reviews from other buyers the final supplementary home theater components were chosen according to the block diagram in Fig. 12.

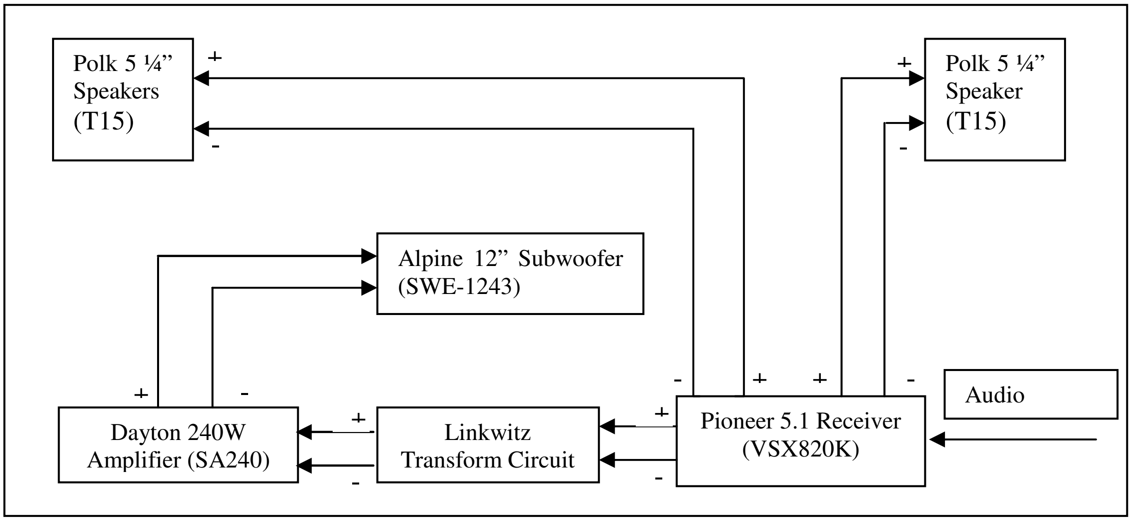

Figure 12. Component Block Diagram

A platform was built inside the couch, along with an amp rack and custom subwoofer enclosure. A Pioneer 550 Watt 5.1 Dolby Digital surround sound receiver was used to amplify the pair of Polk Audio T15 bookshelf speakers using 5 ¼" midrange drivers with a 1" silk dome tweeter. The subwoofer signal is then passed out of the receiver into the Linkwitz transform circuit, then to a Dayton 240 watt (RMS @ 4 Ω) subwoofer amplifier and finally to the subwoofer enclosure. Figure 13 below shows the completed components and custom enclosures integrated inside of the couch.

Figure 13. Supplementary Components

V. Testing and Apparatus

After system functionality was attained, testing is required to verify the performance of the transformed system. Two methods of testing were considered which were a real time analyzer and the human ear. A Real Time Analyzer (RTA) can gather empirical data for comparison but human ears will ultimately decide if the new system has really improved or not. However, the RTA can measure and dictate exactly where the improvements have been made along the frequency response while the human ear can only aid in decided if the entire system is better or not.

A Real Time Analyzer measures and displays the frequency spectrum of an audio signal in real time. There are many variations of this very useful device ranging in the number of measurements per octave. Hardware and software RTA's exist and due to the increased resolution and reduced price of software RTA TrueRTA by True Audio software was used to acquire data. The software has the advantage of taking 24 measurements per octave which is a much higher resolution than a typical 1/3 or 1/6 octave RTA. A Pioneer calibration microphone was used as the acquisition sensor and connected into a Creative Labs Sound Blaster X-Fi Go external sound card for improved audio sampling. The sound card was calibrated to remove its response in order achieve more accurate readings and the data was gathered via text file and exported during each run to compare and analyze in MATLAB.

During data acquisition, the couch and audio components were set up away from walls to reduce wave reflections in order to reduce skewed measurements. A quick sweep across the audible spectrum was generated, which is heard as a "chirp", then the microphone gathers and plots the data. Multiple runs were conducted at various microphone locations, then averaged and smoothed out for consistency. The smoothing process reduces outliers and produces more desirable data. The microphone data was gathered with and without the Linkwitz transform circuit to compare performance in MATLAB.

VI. Results and Discussion

Multiple sweeps were generated and measured at various locations in attempt to generate unbiased results. The data gathered was smoothed and the response with and without the crossover is shown in Fig 14. The area of interest is about 500 Hz and below or were the transition between midrange drivers and subwoofer occurs. The area of interest is the cutoff frequency F_3 which is the measured frequency of -3dB at roll off.

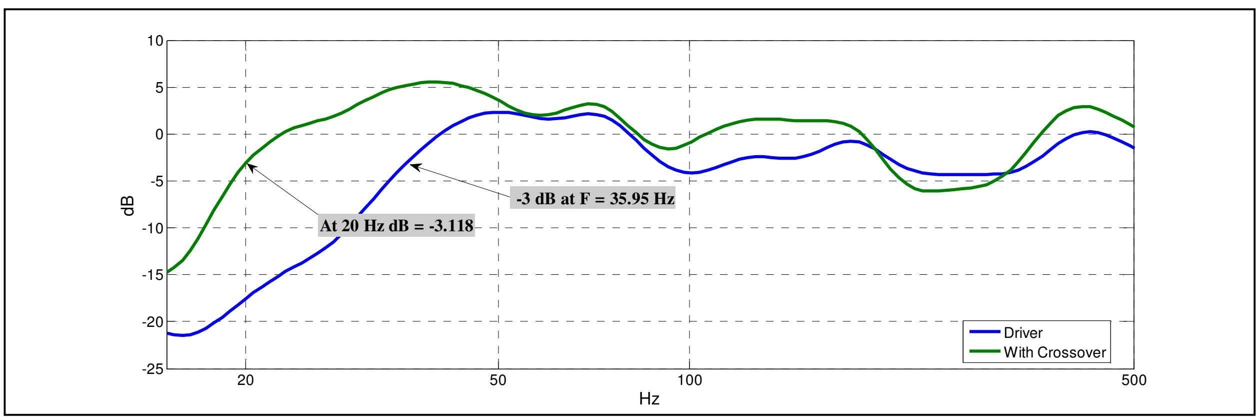

Figure 14. Measure Response with and without Linkwitz crossover

Figure 14 shows the actual achieved F_3 just above 20 Hz which is very close to the 20Hz goal. The cutoff frequency before the addition of the crossover was measured to be 35.95 Hz. The data reveals the crossover performance producing tremendous results easily heard at low frequencies. The optimum frequency response of an audio system would be as flat as possible along the 0 dB axis. The system response shows dips and humps but both are consistent implying other parameters are affecting the response. There are many contributors to alterations in the system response which include the audio components, room characteristics, speaker placement and calibration microphone used. The subwoofer system produced results that were easily detected by ear, adding more realism and more natural characteristics to the sound. The gain on the amplifier was able to be reduced and still perform well but without the reduction in amplifier gain a small distortion was audible but was reduced once adjusted.

The addition of an equalizer will benefit the system by offering greater control of frequency levels. The primary purpose of an equalizer is to boost or cut specific center frequencies and these adjustment taper off to neighboring frequencies depending on the Q value and center frequency being adjusted. A parametric equalizer can vary the Q value, center frequency band and level. While a graphic equalizer has set Q values and center frequency bands and only level is controlled, so even a graphic equalizer with a decent number of bands would help reduce discrepancies from the 0 dB axis generating the desirable flat frequency response across the hearing spectrum.

The Linkwitz transform circuit extended the frequency response as desired next, specific frequencies were produced with and without the active crossover to test performance. 30 Hz was chosen due to the inadequate sound reproduction at this region. A lot of cone movement was noticed but fairly week audible tone was heard compared to frequencies higher than the cutoff frequency. Frequencies between 20-30 Hz were very weak in the driver system without the crossover, so a 30 Hz sine wave was chosen due to its increased popularity in music over a 20 Hz tone. The advantages of this system can also be enjoyed during movies where explosions, gunshots and high impact scenarios produce low frequencies, making them more realistic for improved viewer engagement. TrueRTA was used to generate and measure the 30 Hz sine wave and the results are shown below in Fig. 15.

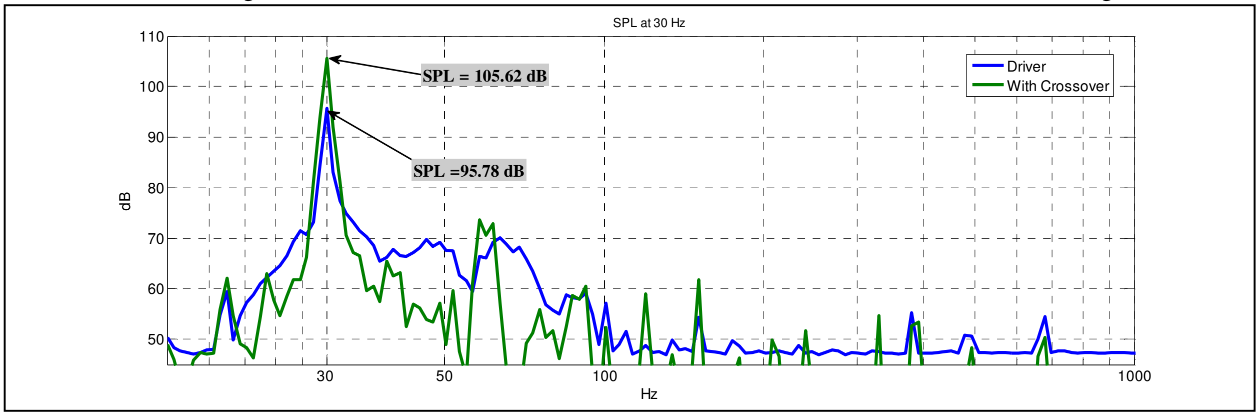

Figure 15. SPL for 30 Hz Audio Signal

The maximum values are noted on the graph where an impressive 9.84 dB difference was measured. This value corresponds to more than 16 times the sound intensity when compared to the system without the Linkwitz filter. The differences was easily heard and felt throughout the room shaking walls and rattling windows. The receiver was set to about ¾ of the maximum power output. These measurements were not intended to test the maximum level attainable. The purpose was to compare the original system and system with the Linkwitz filter while producing 30 Hz signal at the same volume output from the receiver. This also shows the subwoofer and amplifier was chosen properly to handle the increase in power required to produce lower frequencies. A third harmonic was not measured or heard showing proper subwoofer selection, where inadequate subwoofer will produce a third harmonic tone of the frequency attempting to produce.

The overall system combined with the Polk Audio bookshelf speakers yielded impressive results when compared to other surround sound system. The placement of the subwoofer amplifies bass drum kicks and listener enjoyment. Taking into account the price and the number of speakers in the system it out performs many 5.1 and 7.1 surround sound systems. Ultimately showing the versatility and performance of the Linkwitz transform crossover circuit making it a great addition to any home theater system if the subwoofer and amplifier contain the headroom for lower frequency reproduction.

VII. Conclusion

The Linkwitz transform crossover was shown to be a great addition to an existing home theater system. The ability to transform the complex poles to real axis poles is the foundation of the circuitry yielding a fairly simple but effective and inexpensive circuit. Other passive systems tend to require a larger enclosure volume and transient response is sacrificed for sound pressure level and lower frequency reproduction. The Linkwitz crossover sacrifices slight sound pressure level but the results are well worth the tradeoff especially when space is an issue. The GUI designed aided in the component value, driver and enclosure selection to build a prototype from a theoretical system. The crossover performed to a very satisfactory level and integration of the system inside a couch generated great results that any audiophile can appreciate. For future consideration the addition of an equalizer would minimize variations from the 0 dB axis in order to optimize and build a dream home theater system for audio enjoyment.

Appendix

A. Component List With Cost

Linkwitz Transform Circuit					
Component	Value	Price	Component	Value	Price
R1	5.86 kΩ	0.5	C1	2.459 µF	0.81
R2	13.06 kΩ	0.5	C2	0.088 µF	0.81
R3	49.99 kΩ	0.5	C3	0.288 µF	2.43
R4,R6	2.2 kΩ	0.5	C4,C9	100 nF	0.83
R5,R8,R9,R10	100 kΩ	0.5	C5	220 pF	0.81
U1 Op Amp	NE5532	0.775	C6,C7	10 µF	1.14
PCB		18	C8	10 µF	0.89

Power Supply					
Component	Value	Price	Component	Value	Price
R1,R2	10 Ω @ 1W	1.14	C1,C2,C3,C4	2200 µF	0.89
R3,R5	100 Ω	0.25	C5,C6,C7,C8	100 nF	0.81
R4A/B,R6 A/B, R8	2.2 kΩ	0.25	C9,C10,C12,C13	10 µF	0.81
R7, R10, R11	22 kΩ	0.25	C11	1 µF	0.81
R8,R9	2.2 kΩ	0.25	Q1,Q3	BC549	1.16
D1-D10	1N4004	0.1	U1	LM317T	2.78
PCB		18	Q2	BD140	2.26

B. TrueRTA Data

Driver		Driver with Crossover		Driver SPL		Driver with Crossover SPL	
Frequency	dB	Frequency	dB	Frequency	dB	Frequency	dB
10.293	-5.178	10.293	-3.334	10.293	75.394	10.293	76.408
10.595	-6.074	10.595	-4.249	10.595	75.394	10.595	76.408
10.905	-7.501	10.905	-5.676	10.905	66.149	10.905	66.505
11.225	-9.356	11.225	-7.471	11.225	56.903	11.225	56.698
11.554	-11.412	11.554	-9.367	11.554	52.605	11.554	51.353
11.892	-13.473	11.892	-11.142	11.892	48.306	11.892	46.007
12.241	-15.374	12.241	-12.631	12.241	49.306	12.241	46.722
12.599	-17.007	12.599	-13.756	12.599	50.306	12.599	47.524
12.968	-18.324	12.968	-14.514	12.968	47.652	12.968	45.61
13.348	-19.331	13.348	-14.961	13.348	47.467	13.348	42.726
13.74	-20.072	13.74	-15.172	13.74	47.512	13.74	41.083
14.142	-20.603	14.142	-15.204	14.142	47.496	14.142	44.24
14.557	-20.982	14.557	-15.081	14.557	48.068	14.557	47.397
14.983	-21.243	14.983	-14.784	14.983	50.46	14.983	49.018
15.422	-21.4	15.422	-14.268	15.422	48.328	15.422	45.944

15.874	-21.443	15.874	-13.491	15.874	47.703	15.874	40.909
16.339	-21.351	16.339	-12.438	16.339	47.463	16.339	42.627
16.818	-21.106	16.818	-11.139	16.818	47.009	16.818	45.776
17.311	-20.708	17.311	-9.672	17.311	47.339	17.311	47.384
17.818	-20.176	17.818	-8.142	17.818	47.883	17.818	47.085
18.34	-19.549	18.34	-6.658	18.34	47.941	18.34	47.214
18.877	-18.874	18.877	-5.305	18.877	54.813	18.877	55.833
19.431	-18.193	19.431	-4.124	19.431	59.441	19.431	62.105
20	-17.534	20	-3.118	20	49.913	20	54.69
20.586	-16.908	20.586	-2.259	20.586	54.682	20.586	49.031
21.189	-16.315	21.189	-1.511	21.189	57.238	21.189	48.362
21.81	-15.75	21.81	-0.85	21.81	58.834	21.81	46.255
22.449	-15.207	22.449	-0.269	22.449	60.965	22.449	54.433
23.107	-14.688	23.107	0.225	23.107	62.214	23.107	62.997
23.784	-14.192	23.784	0.624	23.784	63.38	23.784	57.772
24.481	-13.712	24.481	0.93	24.481	64.496	24.481	54.588
25.198	-13.228	25.198	1.165	25.198	66.584	25.198	58.447
25.937	-12.711	25.937	1.371	25.937	69.324	25.937	61.798
26.697	-12.129	26.697	1.593	26.697	71.405	26.697	61.814
27.479	-11.455	27.479	1.871	27.479	70.67	27.479	66.191
28.284	-10.677	28.284	2.222	28.284	73.1	28.284	81.225
29.113	-9.804	29.113	2.641	29.113	84.331	29.113	93.751
29.966	-8.862	29.966	3.101	29.966	95.782	29.966	105.617
30.844	-7.884	30.844	3.564	30.844	83.07	30.844	91.975
31.748	-6.904	31.748	3.995	31.748	77.284	31.748	80.995
32.678	-5.946	32.678	4.371	32.678	74.922	32.678	70.575
33.636	-5.019	33.636	4.687	33.636	73.135	33.636	67.155
34.621	-4.125	34.621	4.945	34.621	71.439	34.621	66.531
35.636	-3.261	35.636	5.156	35.636	70.218	35.636	59.59
36.68	-2.43	36.68	5.324	36.68	68.528	36.68	60.454
37.755	-1.639	37.755	5.446	37.755	65.391	37.755	57.359
38.861	-0.9	38.861	5.515	38.861	66.19	38.861	65.411
40	-0.227	40	5.522	40	67.824	40	62.556
41.172	0.37	41.172	5.465	41.172	66.52	41.172	63.17
42.379	0.888	42.379	5.345	42.379	66.447	42.379	52.453
43.62	1.33	43.62	5.168	43.62	67.169	43.62	56.963
44.898	1.698	44.898	4.936	44.898	68.146	44.898	56.154
46.214	1.991	46.214	4.649	46.214	69.786	46.214	53.849
47.568	2.202	47.568	4.305	47.568	68.412	47.568	53.345
48.962	2.324	48.962	3.906	48.962	69.136	48.962	57.179
50.397	2.35	50.397	3.466	50.397	67.576	50.397	48.974
51.874	2.285	51.874	3.015	51.874	67.427	51.874	59.55
53.394	2.148	53.394	2.596	53.394	62.624	53.394	47.565

54.958	1.971	54.958	2.257	54.958	61.537	54.958	42.314
56.569	1.794	56.569	2.039	56.569	59.5	56.569	61.045
58.226	1.659	58.226	1.968	58.226	66.429	58.226	73.587
59.932	1.599	59.932	2.046	59.932	75.708	59.932	87.261
61.688	1.627	61.688	2.252	61.688	69.12	61.688	72.873
63.496	1.734	63.496	2.539	63.496	70.018	63.496	57.176
65.357	1.889	65.357	2.843	65.357	68.883	65.357	44.709
67.272	2.039	67.272	3.092	67.272	67.346	67.272	38.873
69.243	2.122	69.243	3.212	69.243	68.244	69.243	37.493
71.272	2.08	71.272	3.144	71.272	65.845	71.272	49.241
73.36	1.868	73.36	2.858	73.36	63.52	73.36	51.256
75.51	1.467	75.51	2.358	75.51	60.114	75.51	55.922
77.723	0.886	77.723	1.685	77.723	56.745	77.723	50.306
80	0.163	80	0.911	80	55.767	80	51.724
82.344	-0.645	82.344	0.128	82.344	54.924	82.344	46.205
84.757	-1.473	84.757	-0.574	84.757	58.803	84.757	52.464
87.241	-2.254	87.241	-1.119	87.241	58.263	87.241	58.604
89.797	-2.937	89.797	-1.456	89.797	78.006	89.797	91.363
92.428	-3.48	92.428	-1.57	92.428	59.162	92.428	60.574
95.137	-3.865	95.137	-1.474	95.137	54.886	95.137	43.967
97.924	-4.083	97.924	-1.209	97.924	48.951	97.924	36.063
100.794	-4.144	100.794	-0.829	100.794	57.146	100.794	52.406
103.747	-4.065	103.747	-0.391	103.747	47.649	103.747	39.595
106.787	-3.873	106.787	0.054	106.787	48.879	106.787	37.491
109.916	-3.6	109.916	0.466	109.916	51.597	109.916	36.874
113.137	-3.286	113.137	0.82	113.137	47.139	113.137	34.661
116.452	-2.976	116.452	1.103	116.452	47.751	116.452	43.249
119.865	-2.71	119.865	1.313	119.865	48.729	119.865	58.946
123.377	-2.521	123.377	1.453	123.377	47.368	123.377	44.459
126.992	-2.427	126.992	1.534	126.992	47.483	126.992	34.659
130.713	-2.424	130.713	1.566	130.713	46.917	130.713	40.527
134.543	-2.485	134.543	1.56	134.543	49.793	134.543	46.985
138.486	-2.566	138.486	1.526	138.486	47.833	138.486	41.006
142.544	-2.616	142.544	1.477	142.544	48.101	142.544	36.762
146.721	-2.586	146.721	1.426	146.721	47.609	146.721	45.116
151.02	-2.443	151.02	1.389	151.02	54.282	151.02	61.675
155.445	-2.181	155.445	1.376	155.445	47.643	155.445	34.259
160	-1.825	160	1.384	160	47.481	160	36.784
164.688	-1.43	164.688	1.388	164.688	47.348	164.688	41.185
169.514	-1.069	169.514	1.335	169.514	47.11	169.514	40.018
174.481	-0.816	174.481	1.16	174.481	49.668	174.481	41.695
179.594	-0.733	179.594	0.795	179.594	48.604	179.594	46.292
184.856	-0.849	184.856	0.199	184.856	47.212	184.856	35.477

190.273	-1.155	190.273	-0.628	190.273	47.369	190.273	32.733
195.849	-1.612	195.849	-1.632	195.849	47.727	195.849	32.216
201.587	-2.154	201.587	-2.718	201.587	47.218	201.587	34.211
207.494	-2.712	207.494	-3.77	207.494	47.399	207.494	49.795
213.574	-3.224	213.574	-4.68	213.574	47.743	213.574	46.6
219.833	-3.648	219.833	-5.373	219.833	47.429	219.833	33.767
226.274	-3.966	226.274	-5.822	226.274	47.111	226.274	34.108
232.905	-4.178	232.905	-6.047	232.905	48.728	232.905	38.083
239.729	-4.298	239.729	-6.104	239.729	47.19	239.729	51.665
246.754	-4.35	246.754	-6.062	246.754	47.565	246.754	40.576
253.984	-4.356	253.984	-5.979	253.984	46.882	253.984	40.086
261.426	-4.34	261.426	-5.896	261.426	47.437	261.426	34.664
269.087	-4.319	269.087	-5.821	269.087	47.858	269.087	38.689
276.972	-4.307	276.972	-5.736	276.972	47.674	276.972	40.331
285.088	-4.306	285.088	-5.606	285.088	46.992	285.088	41.779
293.441	-4.307	293.441	-5.385	293.441	47.316	293.441	40.683
302.04	-4.29	302.04	-5.032	302.04	47.244	302.04	37.963
310.89	-4.225	310.89	-4.512	310.89	47.019	310.89	32.619
320	-4.076	320	-3.813	320	47.727	320	30.071
329.377	-3.814	329.377	-2.943	329.377	47.51	329.377	54.723
339.028	-3.424	339.028	-1.941	339.028	47.209	339.028	32.864
348.962	-2.911	348.962	-0.869	348.962	47.193	348.962	32.94
359.188	-2.309	359.188	0.191	359.188	47.146	359.188	40.198
369.713	-1.67	369.713	1.152	369.713	47.194	369.713	31.841
380.546	-1.056	380.546	1.939	380.546	55.34	380.546	52.633
391.697	-0.525	391.697	2.503	391.697	47.231	391.697	53.356
403.175	-0.124	403.175	2.828	403.175	47.274	403.175	37.999
414.989	0.123	414.989	2.931	414.989	47.238	414.989	36.915
427.149	0.208	427.149	2.851	427.149	47.397	427.149	35.681
439.665	0.138	439.665	2.633	439.665	47.564	439.665	35.003
452.548	-0.07	452.548	2.317	452.548	47.633	452.548	37.901
465.809	-0.395	465.809	1.93	465.809	47.23	465.809	34.377
479.458	-0.809	479.458	1.491	479.458	50.846	479.458	40.594
493.507	-1.273	493.507	1.017	493.507	50.578	493.507	48.379
507.968	-1.741	507.968	0.528	507.968	47.407	507.968	35.453
522.853	-2.155	522.853	0.048	522.853	47.368	522.853	36.204
538.174	-2.455	538.174	-0.395	538.174	47.254	538.174	32.456
553.943	-2.592	553.943	-0.778	553.943	47.407	553.943	34.27
570.175	-2.537	570.175	-1.085	570.175	47.364	570.175	40.683
586.883	-2.294	586.883	-1.312	586.883	47.259	586.883	42.539
604.08	-1.903	604.08	-1.466	604.08	47.31	604.08	29.687
621.78	-1.436	621.78	-1.562	621.78	47.393	621.78	35.949
640	-0.981	640	-1.626	640	47.216	640	35.478

658.753	-0.628	658.753	-1.69	658.753	50.056	658.753	46.683
678.056	-0.444	678.056	-1.786	678.056	54.47	678.056	50.381
697.925	-0.462	697.925	-1.944	697.925	47.366	697.925	33.122
718.376	-0.677	718.376	-2.179	718.376	47.647	718.376	41.699
739.426	-1.043	739.426	-2.491	739.426	47.711	739.426	42.085
761.093	-1.493	761.093	-2.861	761.093	47.371	761.093	35.896
783.394	-1.955	783.394	-3.259	783.394	47.186	783.394	32.169
806.349	-2.363	806.349	-3.647	806.349	47.45	806.349	32.513
829.977	-2.673	829.977	-3.991	829.977	47.375	829.977	34.175
854.298	-2.865	854.298	-4.265	854.298	47.262	854.298	30.834
879.33	-2.937	879.33	-4.454	879.33	47.212	879.33	34.182
905.097	-2.903	905.097	-4.556	905.097	47.423	905.097	35.552
931.618	-2.787	931.618	-4.574	931.618	47.32	931.618	31.746
958.917	-2.615	958.917	-4.517	958.917	47.445	958.917	33.205
987.015	-2.415	987.015	-4.397	987.015	47.3	987.015	33.717
1015.937	-2.211	1015.937	-4.224	1015.937	47.229	1015.937	31.177
1045.706	-2.027	1045.706	-4.006	1045.706	47.4	1045.706	37.073
1076.347	-1.88	1076.347	-3.749	1076.347	47.395	1076.347	30.747
1107.887	-1.78	1107.887	-3.448	1107.887	47.21	1107.887	31.733
1140.35	-1.727	1140.35	-3.101	1140.35	47.431	1140.35	33.314
1173.765	-1.709	1173.765	-2.7	1173.765	47.259	1173.765	29.252
1208.159	-1.703	1208.159	-2.245	1208.159	47.289	1208.159	31.888
1243.561	-1.683	1243.561	-1.742	1243.561	47.413	1243.561	29.719
1280	-1.619	1280	-1.211	1280	47.287	1280	27.584
1317.507	-1.489	1317.507	-0.675	1317.507	47.238	1317.507	28.368
1356.113	-1.284	1356.113	-0.164	1356.113	47.276	1356.113	28.133
1395.85	-1.003	1395.85	0.298	1395.85	47.166	1395.85	31.446
1436.751	-0.658	1436.751	0.699	1436.751	47.149	1436.751	44.381
1478.851	-0.263	1478.851	1.039	1478.851	47.344	1478.851	30.341
1522.185	0.168	1522.185	1.332	1522.185	47.384	1522.185	32.755
1566.789	0.623	1566.789	1.599	1566.789	47.37	1566.789	28.803
1612.699	1.092	1612.699	1.859	1612.699	47.114	1612.699	30.854
1659.955	1.561	1659.955	2.121	1659.955	47.295	1659.955	34.119
1708.595	2.009	1708.595	2.377	1708.595	47.295	1708.595	28.093
1758.661	2.413	1758.661	2.606	1758.661	47.34	1758.661	33.934
1810.193	2.741	1810.193	2.772	1810.193	47.386	1810.193	43.809
1863.236	2.965	1863.236	2.842	1863.236	47.367	1863.236	29.874
1917.833	3.066	1917.833	2.785	1917.833	47.48	1917.833	27.607
1974.03	3.036	1974.03	2.589	1974.03	47.384	1974.03	34.115
2031.873	2.881	2031.873	2.259	2031.873	47.37	2031.873	33.179
2091.412	2.62	2091.412	1.819	2091.412	47.438	2091.412	34.945
2152.695	2.28	2152.695	1.306	2152.695	47.356	2152.695	27.319
2215.774	1.892	2215.774	0.76	2215.774	47.47	2215.774	29.325

2280.701	1.483	2280.701	0.215	2280.701	47.489	2280.701	27.936
2347.53	1.071	2347.53	-0.306	2347.53	47.397	2347.53	28.467
2416.318	0.66	2416.318	-0.799	2416.318	47.408	2416.318	32.655
2487.122	0.246	2487.122	-1.275	2487.122	47.392	2487.122	28.905
2560	-0.189	2560	-1.751	2560	47.322	2560	28.732
2635.014	-0.664	2635.014	-2.243	2635.014	47.415	2635.014	30.933
2712.226	-1.196	2712.226	-2.757	2712.226	47.329	2712.226	36.879
2791.7	-1.791	2791.7	-3.287	2791.7	47.527	2791.7	31.495
2873.503	-2.441	2873.503	-3.822	2873.503	47.431	2873.503	31.802
2957.703	-3.124	2957.703	-4.343	2957.703	47.288	2957.703	33.779
3044.37	-3.807	3044.37	-4.841	3044.37	47.312	3044.37	28.983
3133.577	-4.455	3133.577	-5.31	3133.577	47.349	3133.577	28.383
3225.398	-5.035	3225.398	-5.758	3225.398	47.419	3225.398	32.441
3319.909	-5.526	3319.909	-6.194	3319.909	47.372	3319.909	27.087
3417.19	-5.922	3417.19	-6.63	3417.19	47.291	3417.19	34.296
3517.321	-6.23	3517.321	-7.076	3517.321	47.477	3517.321	32.772
3620.387	-6.469	3620.387	-7.535	3620.387	47.364	3620.387	30.465
3726.472	-6.661	3726.472	-8.002	3726.472	47.386	3726.472	27.623
3835.666	-6.832	3835.666	-8.467	3835.666	47.221	3835.666	29.584
3948.06	-7.005	3948.06	-8.907	3948.06	47.24	3948.06	27.223
4063.747	-7.198	4063.747	-9.296	4063.747	47.341	4063.747	30.148
4182.824	-7.421	4182.824	-9.6	4182.824	47.31	4182.824	32.586
4305.39	-7.678	4305.39	-9.788	4305.39	47.4	4305.39	37.891
4431.547	-7.961	4431.547	-9.839	4431.547	47.392	4431.547	26.942
4561.401	-8.254	4561.401	-9.749	4561.401	47.329	4561.401	28.203
4695.061	-8.534	4695.061	-9.533	4695.061	47.362	4695.061	26.351
4832.636	-8.771	4832.636	-9.227	4832.636	47.521	4832.636	31.555
4974.244	-8.938	4974.244	-8.877	4974.244	47.402	4974.244	27.273
5120	-9.009	5120	-8.527	5120	47.422	5120	27.213
5270.027	-8.972	5270.027	-8.217	5270.027	47.382	5270.027	27.346
5424.451	-8.828	5424.451	-7.969	5424.451	47.277	5424.451	25.085
5583.4	-8.595	5583.4	-7.789	5583.4	47.358	5583.4	29.036
5747.006	-8.307	5747.006	-7.667	5747.006	47.483	5747.006	27.273
5915.406	-8	5915.406	-7.581	5915.406	47.407	5915.406	25.749
6088.74	-7.708	6088.74	-7.502	6088.74	47.407	6088.74	32.173
6267.154	-7.452	6267.154	-7.4	6267.154	47.45	6267.154	35.092
6450.796	-7.236	6450.796	-7.253	6450.796	47.376	6450.796	28.298
6639.819	-7.053	6639.819	-7.051	6639.819	47.353	6639.819	26.227
6834.38	-6.891	6834.38	-6.808	6834.38	47.371	6834.38	28.798
7034.643	-6.746	7034.643	-6.556	7034.643	47.415	7034.643	26.029
7240.773	-6.628	7240.773	-6.34	7240.773	47.394	7240.773	28.332
7452.944	-6.568	7452.944	-6.212	7452.944	47.356	7452.944	28.783
7671.332	-6.603	7671.332	-6.213	7671.332	47.372	7671.332	32.02

7896.119	-6.768	7896.119	-6.367	7896.119	47.477	7896.119	36.944
8127.493	-7.082	8127.493	-6.675	8127.493	47.451	8127.493	28.308
8365.647	-7.54	8365.647	-7.118	8365.647	47.36	8365.647	28.793
8610.779	-8.11	8610.779	-7.665	8610.779	47.398	8610.779	28.209
8863.094	-8.746	8863.094	-8.277	8863.094	47.434	8863.094	27.613
9122.803	-9.399	9122.803	-8.918	9122.803	47.458	9122.803	30.215
9390.121	-10.025	9390.121	-9.559	9390.121	47.465	9390.121	28.923
9665.273	-10.597	9665.273	-10.174	9665.273	47.496	9665.273	28.249
9948.487	-11.102	9948.487	-10.747	9948.487	47.521	9948.487	42.528
10240	-11.541	10240	-11.266	10240	47.443	10240	33.212
10540.055	-11.918	10540.055	-11.723	10540.055	47.459	10540.055	36.811
10848.902	-12.244	10848.902	-12.114	10848.902	47.412	10848.902	28.215
11166.799	-12.53	11166.799	-12.44	11166.799	47.465	11166.799	28.23
11494.011	-12.795	11494.011	-12.711	11494.011	47.493	11494.011	27.459
11830.812	-13.069	11830.812	-12.952	11830.812	47.422	11830.812	30.309
12177.481	-13.39	12177.481	-13.199	12177.481	47.477	12177.481	31.26
12534.308	-13.802	12534.308	-13.501	12534.308	47.402	12534.308	29.967
12901.592	-14.349	12901.592	-13.911	12901.592	47.464	12901.592	28.669
13279.637	-15.062	13279.637	-14.473	13279.637	47.457	13279.637	30.896
13668.76	-15.947	13668.76	-15.21	13668.76	47.452	13668.76	27.159
14069.285	-16.986	14069.285	-16.117	14069.285	47.405	14069.285	30.502
14481.547	-18.129	14481.547	-17.153	14481.547	47.498	14481.547	37.197
14905.889	-19.303	14905.889	-18.25	14905.889	47.415	14905.889	27.789
15342.664	-20.418	15342.664	-19.315	15342.664	47.43	15342.664	28.095
15792.239	-21.382	15792.239	-20.253	15792.239	47.419	15792.239	26.837
16254.987	-22.101	16254.987	-20.959	16254.987	47.43	16254.987	27.778
16731.294	-22.542	16731.294	-21.397	16731.294	47.421	16731.294	26.087
17221.559	-22.837	17221.559	-21.691	17221.559	47.373	17221.559	28.42
17726.189	-23.026	17726.189	-21.883	17726.189	47.433	17726.189	28.284
18245.606	-23.15	18245.606	-22.009	18245.606	47.411	18245.606	27.001
18780.243	-23.231	18780.243	-22.092	18780.243	47.448	18780.243	29.202
19330.546	-23.283	19330.546	-22.146	19330.546	47.41	19330.546	35.665
19896.974	-23.318	19896.974	-22.183	19896.974	47.416	19896.974	29.986
20480	-23.341	20480	-22.209	20480	47.461	20480	28.049
21080.11	-23.358	21080.11	-22.228	21080.11	47.396	21080.11	28.841
21697.804	-23.371	21697.804	-22.244	21697.804	47.402	21697.804	21.723
22333.598	-23.383	22333.598	-22.261	22333.598	34.897	22333.598	11.193
22988.023	-23.397	22988.023	-22.283	22988.023	-139.477	22988.023	-139.477
23661.623	-23.415	23661.623	-22.316	23661.623	-139.608	23661.623	-139.608
24354.962	-23.445	24354.962	-22.372	24354.962	-139.73	24354.962	-139.73
25068.617	-23.499	25068.617	-22.476	25068.617	-139.857	25068.617	-139.857
25803.183	-23.598	25803.183	-22.661	25803.183	-139.984	25803.183	-139.984
26559.274	-23.763	26559.274	-22.961	26559.274	-140.108	26559.274	-140.108

27337.52	-23.998	27337.52	-23.373	27337.52	-140.232	27337.52	-140.232
28138.571	-24.285	28138.571	-23.853	28138.571	-140.359	28138.571	-140.359
28963.094	-24.582	28963.094	-24.324	28963.094	-140.483	28963.094	-140.483
29811.777	-24.846	29811.777	-24.716	29811.777	-140.607	29811.777	-140.607
30685.329	-25.046	30685.329	-24.991	30685.329	-140.738	30685.329	-140.738
31584.478	-25.176	31584.478	-25.156	31584.478	-140.858	31584.478	-140.858
32509.974	-25.249	32509.974	-25.239	32509.974	-140.987	32509.974	-140.987
33462.588	-25.283	33462.588	-25.275	33462.588	-141.11	33462.588	-141.11
34443.117	-25.297	34443.117	-25.288	34443.117	-141.235	34443.117	-141.235
35452.378	-25.303	35452.378	-25.293	35452.378	-141.363	35452.378	-141.363
36491.211	-25.305	36491.211	-25.296	36491.211	-141.487	36491.211	-141.487
37560.486	-25.307	37560.486	-25.298	37560.486	-141.612	37560.486	-141.612
38661.092	-25.308	38661.092	-25.3	38661.092	-141.74	38661.092	-141.74
39793.948	-25.309	39793.948	-25.302	39793.948	-141.863	39793.948	-141.863
40960	-25.31	40960	-25.303	40960	-141.989	40960	-141.989
42160.22	-25.311	42160.22	-25.305	42160.22	-142.113	42160.22	-142.113
43395.608	-25.312	43395.608	-25.307	43395.608	-142.241	43395.608	-142.241
44667.197	-25.313	44667.197	-25.309	44667.197	-142.365	44667.197	-142.365
45976.045	-25.315	45976.045	-25.311	45976.045	-142.49	45976.045	-142.49
47323.246	-25.316	47323.246	-25.313	47323.246	-142.616	47323.246	-142.616

C. MATLAB Code

Graphical User Interface

```
function varargout = LR_Filter_GUI(varargin)
%LR_FILTER_GUI M-file for LR_Filter_GUI.fig
%      LR_FILTER_GUI, by itself, creates a new LR_FILTER_GUI or raises the existing
%      singleton*.
%
%      H = LR_FILTER_GUI returns the handle to a new LR_FILTER_GUI or the handle to
%      the existing singleton*.
%
%      LR_FILTER_GUI('Property','Value',...) creates a new LR_FILTER_GUI using the
%      given property value pairs. Unrecognized properties are passed via
%      varargin to LR_Filter_GUI_OpeningFcn.  This calling syntax produces a
%      warning when there is an existing singleton*.
%
%      LR_FILTER_GUI('CALLBACK') and LR_FILTER_GUI('CALLBACK',hObject,...) call the
%      local function named CALLBACK in LR_FILTER_GUI.M with the given input
%      arguments.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help LR_Filter_GUI

% Last Modified by GUIDE v2.5 31-Oct-2010 22:30:00

% Begin initialization code - DO NOT EDIT
```

```
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @LR_Filter_GUI_OpeningFcn, ...
                   'gui_OutputFcn',  @LR_Filter_GUI_OutputFcn, ...
                   'gui_LayoutFcn',  [], ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
   gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before LR_Filter_GUI is made visible.
function LR_Filter_GUI_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   unrecognized PropertyName/PropertyValue pairs from the
%            command line (see VARARGIN)

% Choose default command line output for LR_Filter_GUI
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes LR_Filter_GUI wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = LR_Filter_GUI_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in Calculate_pushbutton.
function Calculate_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to Calculate_pushbutton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Execute Main Calculates to calculate values and plot responses

%% Data to input into Calculator

%%%%%%%%%
%%INPUT%%
%%%%%%%%%
format short
```

```
% Speaker Parameters

Fs = str2num(get(handles.Fs_edit,'String'))
Qts = str2num(get(handles.Qts_edit,'String'))
Vas = str2num(get(handles.Vas_edit,'String'))
Vb = str2num(get(handles.Vb_edit,'String'))

%Fs = 24; % [Hz] Free Air Resonance Frequency
% Qts = 0.28; % Total Q of driver
% Vas =  4.732; %[ft^3] Driver air compliance
% Vb  = 1.413; %[ft^3]  Desired box volume

% Note: Conversion: Liters/28.317 = ft^3

%% Enclosure Calculations



alpha = Vas/Vb;
Fsc = Fs*sqrt(alpha+1); %[Hz] System Resonance
Qtc = Fsc*Qts/Fs; % Total System Q
F3 = Fsc*sqrt(((1/Qtc^2-2)+sqrt((1/Qtc^2-2)^2+4))/2); % [Hz] Lower cut off frequency

%Update Handles structure
set(handles.F0_text,'String',Fsc);
set(handles.Q0_text,'String',Qtc);
set(handles.F3_text,'String',F3);
guidata(hObject, handles);

% Frequency variation to generate response of driver in the enclosure
Freq = 15:5:200; %[Hz]
Fh = Freq/Fsc;
Fq = Fh.^2;

Mag = Fq./sqrt((Fq-1).^2 + (Fh/Qtc).^2); % Magnitude
Resp = 20*(log10(Mag)/log10(10)); % [dB] Response

axes(handles.Enclosure_system_response)
semilogx(Freq,Resp)
set(handles.Enclosure_system_response,'XMinorTick','on')
grid on
set(get(handles.Enclosure_system_response,'Xlabel'),'string','Hz');
set(get(handles.Enclosure_system_response,'Ylabel'),'string','dB');

% semilogx(handles.Enclosure_system_response,Freq,Resp)
% set(handles.Enclosure_system_response,'XMinorTick','on')
% grid on
% set(get(handles.Enclosure_system_response,'Xlabel'),'string','Frequency (Hz)');
% set(get(handles.Enclosure_system_response,'Ylabel'),'string','dB');

%set(get(handles.Enclosure_system_response,'title'),'string','Response of Speaker
Enclosure');

% grid on
% title('Frequency Response of Driver and Enclosure')
% xlabel('Frequency')
% ylabel('dB')
% axis([min(Freq) max(Freq) min(Resp)-5 max(Resp)+5])

%% Calculate Resistor and Capacitor Values

%%%%%%%%%
%%INPUT%%
```

```
%%%%%%%%%

Fp = str2num(get(handles.Fp_edit,'String'))
Qp = str2num(get(handles.Qp_edit,'String'))
C2_m = str2num(get(handles.C2_edit,'String'))


% Fp = 20; %[Hz] Target System Resonance (Fsc) of transformed system
% Qp = 0.8; % Taget Sysgtem Q (Qtc) of transformed System
% % Choose Capacitor #2 value in microFarads
% C2_m = 0.068; %[MicroFarads]

F0 = Fsc; %[Hz] System resonance
Q0 = Qtc; % Total System Q

k = (F0/Fp-Q0/Qp)/(Q0/Qp-Fp/F0); % Should be greater than 0

% Calculate Resistor and Capacitor Component Values
R1_k = (1/(6.2832*F0*(C2_m/1000000)*(2*Q0*(1+k))))/1000; %[kOhms]
R2_k = 2*k*R1_k; %[kOhms]
R3_k = R1_k*(F0/Fp)^2; %[kOhms]

C1_m = C2_m*(2*(Q0)*(1+k))^2; %[MicroFarads]
C3_m = C1_m*(Fp/F0)^2; %[MicroFarads]

DC_Gain = 40*log10(F0/Fp);


% Update Handles Structure
set(handles.k_text,'String',k);
set(handles.DCk_text,'String',DC_Gain);
set(handles.R1_text,'String',R1_k);
set(handles.R2_text,'String',R2_k);
set(handles.R3_text,'String',R3_k);
set(handles.C1_text,'String',C1_m);
set(handles.C3_text,'String',C3_m);

guidata(hObject, handles);


%Note: DC Gain should be lower than 20 dB
%Increase C2 to lower values of R1, R2 and R3

% Frequency Response Calculations

% Convert R and C to ohms and Farads
R1 = R1_k*1000;
R2 = R2_k*1000;
R3 = R3_k*1000;

C1 = C1_m*10^-6;
C2 = C2_m*10^-6;
C3 = C3_m*10^-6;

beta = 0.5/Q0; % Damping Factor for closed box speaker

F = ones(1,100);
for i = 2:length(F)
    F(i) = F(i-1)*10^0.025;
end

w = 2*pi*F; %[rad/sec] Angular Frequency
```

```
% Preallocate Variable
H_cross = zeros(1,length(F));
H_cross_log = zeros(1,length(F));
phi_cross = zeros(1,length(F));
H_speaker = zeros(1,length(F));
H_speaker_log = zeros(1,length(F));
phi_speaker = zeros(1,length(F));
H_tot = zeros(1,length(F));
H_tot_log = zeros(1,length(F));
phi_tot = zeros(1,length(F));

for i = 1:length(F)
    % H of Crossover Network
    H_cross(i) =(R3/R1).*((1.-
w(i).^2.*C1*C2*R1^2).^2.+(w(i).*C2*(R2+2*R1)).^2).^0.5...
        ./((1.-
w(i).^2.*C2*C3*R3^2).^2.+(w(i).*C2*(R2+2*R3)).^2).^0.5.*(4.+(w(i).*R3*...
        C3).^2).^0.5./(4.+(w(i).*C1*R1).^2).^0.5;

    H_cross_log(i) = 20*log10(H_cross(i));

    % phi of crossover network
    phi_cross(i) = -180/pi*(atan(w(i)*C3*R3/4)-atan(w(i)*C1*R1/4)+atan((w(i)*...
        C2*(2*R1+R2))/(1-w(i).^2*C1*C2*R1^2))-atan((w(i)*C2*(R2...
        +2*R3))/(1-w(i).^2*C2*C3*R3^2)));
    % H of speaker system
    H_speaker(i) = (F(i)/F0).^2/((1-(F(i)/F0).^2)^2+(2*beta*F(i)/F0).^2).^0.5;
    H_speaker_log(i) = 20*log10(H_speaker(i));

    % phi of speaker
    phi_speaker(i) = 180/pi*atan((2*beta*F(i)/F0)/(1-(F(i)/F0).^2));

    H_tot(i) = H_cross(i)*H_speaker(i);

    H_tot_log(i) = 20*log10(H_tot(i));

    phi_tot(i) = phi_cross(i) + phi_speaker(i);

end

% Find maximum and minimum values
maxH = [max(H_cross_log),max(H_speaker_log),max(H_tot_log)];
minH = [min(H_cross_log),min(H_speaker_log),min(H_tot_log)];

maxphi = [max(phi_cross),max(phi_speaker),max(phi_tot)];
minphi = [min(phi_cross),min(phi_speaker),min(phi_tot)];

%% Plots

axes(handles.Transformed_system_response)
semilogx(F,H_cross_log,F,H_speaker_log,F,H_tot_log)
set(handles.Enclosure_system_response,'XMinorTick','on')
grid on
set(get(handles.Transformed_system_response,'Xlabel'),'string','Hz');
set(get(handles.Transformed_system_response,'Ylabel'),'string','dB');
legend('Crossover','Speaker','Combined')


%
semilogx(handles.Transformed_system_response,F,H_cross_log,F,H_speaker_log,F,H_tot_log
)
% set(get(handles.Transformed_system_response,'Xlabel'),'string','Frequency (Hz)');
% set(get(handles.Transformed_system_response,'Ylabel'),'string','dB');
```

```
% set(handles.Transformed_system_response,'XMinorTick','on')
% grid on
% set(get(handles.Transformed_system_response,'title'),'string','Response of Speaker
Enclosure');

axes(handles.Phase_response)
semilogx(F,phi_cross,F,phi_speaker,F,phi_tot)
set(handles.Phase_response,'XMinorTick','on')
grid on
set(get(handles.Phase_response,'Xlabel'),'string','Hz');
set(get(handles.Phase_response,'Ylabel'),'string','dB');
legend('Crossover','Speaker','Combined')

% semilogx(handles.Phase_response,F,phi_cross,F,phi_speaker,F,phi_tot)
% %axis([min(F) max(F) min(minphi)-5 max(maxphi)+5])
% % title('Phase Response')
% xlabel('Frequency')
% ylabel('deg')
% %legend('Crossover','Speaker w/ Enclosure','Total','Location','best')
% % set(handles.frequency_axes,'XMinorTick','on')
% grid on

% --- Executes during object creation, after setting all properties.
function Qts_edit_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Qts_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function Fs_edit_Callback(hObject, eventdata, handles)
% hObject    handle to Fs_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Fs_edit as text
%        str2double(get(hObject,'String')) returns contents of Fs_edit as a double
Fs_input = str2num(get(hObject,'String'));

%checks to see if input is empty. if so, default input1_editText to zero
if (isempty(Fs_input))
     set(hObject,'String','0')
end
guidata(hObject, handles);



% --- Executes during object creation, after setting all properties.
function Fs_edit_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Fs_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function Vas_edit_Callback(hObject, eventdata, handles)
% hObject    handle to Vas_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Vas_edit as text
%        str2double(get(hObject,'String')) returns contents of Vas_edit as a double
Vas_input = str2num(get(hObject,'String'));

%checks to see if input is empty. if so, default input1_editText to zero
if (isempty(Vas_input))
     set(hObject,'String','0')
end
guidata(hObject, handles);



% --- Executes during object creation, after setting all properties.
function Vas_edit_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Vas_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function Vb_edit_Callback(hObject, eventdata, handles)
% hObject    handle to Vb_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Vb_edit as text
%        str2double(get(hObject,'String')) returns contents of Vb_edit as a double
Vb_input = str2num(get(hObject,'String'));

%checks to see if input is empty. if so, default input1_editText to zero
if (isempty(Vb_input))
     set(hObject,'String','0')
end
guidata(hObject, handles);



% --- Executes during object creation, after setting all properties.
function Vb_edit_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Vb_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
```

```
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function Fp_edit_Callback(hObject, eventdata, handles)
% hObject    handle to Fp_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Fp_edit as text
%        str2double(get(hObject,'String')) returns contents of Fp_edit as a double
Fp_input = str2num(get(hObject,'String'));

%checks to see if input is empty. if so, default input1_editText to zero
if (isempty(Fp_input))
     set(hObject,'String','0')
end
guidata(hObject, handles);



% --- Executes during object creation, after setting all properties.
function Fp_edit_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Fp_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function Qp_edit_Callback(hObject, eventdata, handles)
% hObject    handle to Qp_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Qp_edit as text
%        str2double(get(hObject,'String')) returns contents of Qp_edit as a double
Qp_input = str2num(get(hObject,'String'));

%checks to see if input is empty. if so, default input1_editText to zero
if (isempty(Qp_input))
     set(hObject,'String','0')
end
guidata(hObject, handles);



% --- Executes during object creation, after setting all properties.
function Qp_edit_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Qp_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
```

```
% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function C2_edit_Callback(hObject, eventdata, handles)
% hObject    handle to C2_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of C2_edit as text
%        str2double(get(hObject,'String')) returns contents of C2_edit as a double
C2_input = str2num(get(hObject,'String'));

%checks to see if input is empty. if so, default input1_editText to zero
if (isempty(C2_input))
     set(hObject,'String','0')
end
guidata(hObject, handles);


% --- Executes during object creation, after setting all properties.
function C2_edit_CreateFcn(hObject, eventdata, handles)
% hObject    handle to C2_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on selection change in Vas_unit_popupmenu.
function Vas_unit_popupmenu_Callback(hObject, eventdata, handles)
% hObject    handle to Vas_unit_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns Vas_unit_popupmenu contents as cell
array
%        contents{get(hObject,'Value')} returns selected item from Vas_unit_popupmenu
% Determine the selected data set.

% %gets the selected option
% switch get(handles.popupmenu1,'String')
%     case 1
%         handles.Vas_volume = handles.Vas_volume;
%     case 2
%         handles.Vas_volume = handles.Vas_volume/28.329;
%     otherwise
% end

str = get(hObject, 'String');
val = get(hObject,'Value');
% Set current data to the selected data set.
switch str{val};
```

```
case 'Ft^3' % User selects cubic feet.
   handles.Vas_input = handles.Vas_input;
   1
case 'Liters' % User selects liters.
   handles.Vas_input = handles.Vas_input/28.33;
   2
end
% Save the handles structure.
guidata(hObject,handles)


% --- Executes during object creation, after setting all properties.
function Vas_unit_popupmenu_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Vas_unit_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on selection change in Vb_units_popupmenu.
function Vb_units_popupmenu_Callback(hObject, eventdata, handles)
% hObject    handle to Vb_units_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns Vb_units_popupmenu contents as cell
array
%        contents{get(hObject,'Value')} returns selected item from Vb_units_popupmenu
% Determine the selected data set.
str = get(hObject, 'String');
val = get(hObject,'Value');
% Set current data to the selected data set.
switch str{val};
case 'Ft^3' % User selects cubic feet.
   handles.Vb_volume = handles.cubic_ft;
case 'Liters' % User selects liters.
   handles.Vb_volume = handles.liters;
end
% Save the handles structure.
guidata(hObject,handles)

% --- Executes during object creation, after setting all properties.
function Vb_units_popupmenu_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Vb_units_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function Qts_edit_Callback(hObject, eventdata, handles)
```

```
% hObject    handle to Qts_edit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Qts_edit as text
%        str2double(get(hObject,'String')) returns contents of Qts_edit as a double

Qts_input = str2num(get(hObject,'String'));

%checks to see if input is empty. if so, default input1_editText to zero
if (isempty(Qts_input))
     set(hObject,'String','0')
end
guidata(hObject, handles);
```

Transfer Function Analysis

```
%% Linkwitz Transform Transfer Function Plots

clear all
close all
clc

%Driver Properties
Fp = 58;
Qp = 1.25;

%Transform Properties
Fz = 20;
Qz = 0.8;

%Combined system 1
num1 = [1 0 0];
den1 = [1 4*pi*Fz (2*pi*Fz)^2];
sys1 =tf(num1,den1)

% Driver Enclosure 2
num2 = [1 0 0];
den2 = [1 2*pi*Fp/Qp (2*pi*Fp)^2];
sys2 = tf(num2,den2)

% Linkwitz Equilizer 3
num3 = [1 (2*pi*Fp)/Qp (2*pi*Fp)^2];
den3 = [1 (2*pi*Fz)/Qz (2*pi*Fz)^2];
sys3 = tf(num3,den3)

figure('name','Pole-Zero and Root Locus')
subplot(1,2,1)
pzmap(sys1,sys2,sys3)
legend('Combined','Driver','Linkwitz Filter','Location','best')

subplot(1,2,2)
rlocus(sys1,sys2,sys3)
legend('Combined','Driver','Linkwitz Filter','Location','best')

figure('name','Bode Plot')
h1 = bodeplot(sys1,sys2,sys3);
setoptions(h1,'FreqUnits','Hz')

figure('name','Step and Impulse Response')
subplot(2,1,1)
stepplot(sys1,sys2)
```

```
%axis([0 0.04 -1.25 9.25])
legend('Combined','Driver','Location','best')

subplot(2,1,2)
impulseplot(sys1,sys2,sys3)
legend('Combined','Driver','Linkwitz Filter','Location','SouthEast')
```

References

[1] Akoustikos Henry George Liddell, Robert Scott, *A Greek-English Lexicon*, at Perseus

[2]Linkwitz Lab, "Active Filters", URL: http://www.linkwitzlab.com/filters.htm [cited 2 Dec 2010]

[3]Elliot, Rod, Elliot Sound Products, *Project 71,* URL: http://sound.westhost.com/project71.htm [cited 2 Dec 2010]

[4]Dickason Vance, *Loudspeaker Design Cookbook*, 7[ed]ed. Audio Amateur Press, New Hampshire 2006

[5]Eargle, John M., *Loudspeaker Handbook*, 1[st] ed., Chapman and Hall 1997

[6]Steele, Brian, *The Subwoofer DIY Page,* URL: http://www.diysubwoofers.org/ [cited 3 Dec 2010]

[7]Nise, Norman S., 4[th] ed., John Wiley and Sons, Inc, New Jersey, 2004

[8]MATLAB, Ver. 7.7.0.471 (R2008b)The Mathworks, MA, 2008

[9]TrueRTA, Ver.3.5.0, True Audio, TN, 2010