Daniel Sanz Ausín

Experimental Investigation Of Loudspeaker Power Requirements

Master’s thesis, January 2015
Experimental Investigation Of Loudspeaker Power Requirements

Author:
Daniel Sanz Ausín

Supervisors:
Michael A. E. Andersen
Henrik Schneider

DTU Elektro
Technical University of Denmark
2800 Kgs. Lyngby
Denmark

studieadministration@elektro.dtu.dk

Project period: 25/08/2014-25/01/2015
ECTS: 30
Class: Public
Edition: 1. edition
Abstract

The components of a sound system, whether it is a small system for an electronic device or a high power one, need to be designed to meet certain power requirements. For this, manufacturers often design these components using test-signals, which have an unchanging amplitude. These are very different from real audio signals, because, in audio, the amplitude is continuously changing over the time, and high power peaks can be found, which have a small duration in comparison with the continuous low power values.

This project follows the work done by a group of PhD students and professors of the Technical University of Denmark, where the power requirements of a loudspeaker were investigated when it is reproducing different audio files. For this, over 400 songs were used, and the simulations were done with the mathematical linear model of the loudspeaker, which will also be explained in this document. The results of the simulations showed, as expected, that loudspeakers have a high power consumption for short amounts of time, and a low continuous power consumption. The goal of the investigation is to avoid oversizing and unnecessary costs when designing all the units of a sound system, such as the power supply unit, the audio amplifier and the loudspeaker driver.

The main goal of this project is to validate the functionality of the mathematical model of the loudspeaker, to make sure that the simulations on this model give the same results as measurements in a real loudspeaker. For this, a circuit has been designed and built, which allows to send an audio signal to a real loudspeaker system and measure the power consumption of the loudspeaker driver. Measurements taken on this circuit have been compared to simulations using the linear and non-linear models, which are also explained in the document. The results show, indeed, that the models simulate the behaviour of the real loudspeaker with high precision, so the results from the simulations are absolutely valid. Moreover, the difference between the measurements on the linear and the non-linear models is very small, which confirms that the linear model is enough for the simulations, which is more simple than the non-linear one.

The validity of the mathematical model has been confirmed making measurements with various audio files belonging to different musical styles. The power requirements give very similar results both for the real loudspeaker and the simulation models.

Finally, a database of over 100 loudspeaker drivers of the same size has been taken, and simulations have been done on each one of them: all the loudspeakers must have the same sound pressure level response, so a filter is applied to each one to make them all sound the same.
Then, simulations have been run and the power requirements have
been analysed. The results show how the range of power consumption
can be quite wide for the loudspeakers, even if they are similar and
are forced to have the same sound response. This explains why design
for the worst case needs to be done.
# Contents

1 Introduction .................................................. 1
   1.1 Thesis Objective ........................................... 1
   1.2 Organisation of the Thesis ............................... 2

2 Loudspeaker Modelling ........................................ 4
   2.1 Loudspeaker Parts And Operation ....................... 4
   2.2 Domains and Conversions ................................ 5
      2.2.1 Electrical Domain .................................... 5
      2.2.2 Mechanical Domain .................................. 5
      2.2.3 Acoustical Domain .................................. 6
   2.3 Electro-Mechanical Conversion and Final Linear Model 7
   2.4 Non-Linearities ............................................ 9
   2.5 Final Non-Linear Model ................................... 11

3 Loudspeaker Power Measurement Circuit .................... 13
   3.1 Audio Power, RMS and Peak Power ..................... 13
   3.2 Sampling Theorem and Aliasing ......................... 14
   3.3 Power Measurement Circuit: Main Components ........ 16
      3.3.1 The Loudspeaker: Monacor SPH-170TC ............ 18
      3.3.2 The Amplifier: Texas Instruments TPA3116D2 .... 18
      3.3.3 The 10mΩ Current Measurement Resistor .......... 19
      3.3.4 The Instrumentation Amplifier: Texas Instruments INA2128 ........................................ 19
      3.3.5 Operational Amplifiers for Filter design: MC33079 .................................................. 19
      3.4 Testing Anti-Aliasing Filters ........................... 21
      3.5 ADC & Software: Matlab & National Instruments USB-6356 ................................................. 24
         3.5.1 Matlab ............................................... 24
         3.5.2 NI USB-6356 ......................................... 24
      3.6 Audio Amplifier Model ................................... 25

4 Loudspeaker System Measurements, Simulations & Comparison ..................................... 29
   4.1 Preparing the Audio File .................................. 29
   4.2 Power Measurements at two Sound Pressure Levels .... 30
      4.2.1 Circuit Measurements ................................ 30
      4.2.2 Simulations: Linear and Non-Linear Loudspeaker Models ............................................. 30
   4.3 Comparison & Conclusions .................................. 31
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Power Window Sweep</td>
<td>35</td>
</tr>
<tr>
<td>5.1</td>
<td>Power Window Sweep Algorithm</td>
<td>35</td>
</tr>
<tr>
<td>5.2</td>
<td>Comparing Results &amp; Conclusions</td>
<td>37</td>
</tr>
<tr>
<td>6</td>
<td>Loudness Normalisation &amp; Measurements with Different Songs</td>
<td>41</td>
</tr>
<tr>
<td>6.1</td>
<td>Loudness Normalisation</td>
<td>41</td>
</tr>
<tr>
<td>6.2</td>
<td>Simulations with Different Songs</td>
<td>42</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Preparation of the Songs</td>
<td>43</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Measurements</td>
<td>43</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Power Window Sweep</td>
<td>44</td>
</tr>
<tr>
<td>7</td>
<td>Statistical Analysis of Loudspeaker Power Requirements</td>
<td>49</td>
</tr>
<tr>
<td>7.1</td>
<td>Thiele-Small Parameters Database</td>
<td>49</td>
</tr>
<tr>
<td>7.2</td>
<td>Target Sound Pressure Level &amp; Filter design</td>
<td>50</td>
</tr>
<tr>
<td>7.3</td>
<td>Simulation &amp; Power Window Sweep</td>
<td>51</td>
</tr>
<tr>
<td>7.4</td>
<td>Comparison &amp; Conclusions</td>
<td>54</td>
</tr>
<tr>
<td>8</td>
<td>Conclusion</td>
<td>57</td>
</tr>
<tr>
<td>9</td>
<td>Future Work</td>
<td>59</td>
</tr>
<tr>
<td>Appendices</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>Matlab Files</td>
<td>c</td>
</tr>
<tr>
<td>B</td>
<td>Other Files</td>
<td>e</td>
</tr>
</tbody>
</table>
## List of Figures

2.1 Loudspeaker Driver. Source: [5] ................................. 4  
2.2 Electrical model of the voice coil. Source: [3] ................. 5  
2.3 Mechanical model of the loudspeaker. Source: [3] ............... 6  
2.4 Electro-Mechanical equivalent circuit of the loudspeaker. Source: [3] 7  
2.5 Linear model of the loudspeaker in Simulink ........................ 8  
2.6 Linear model of the loudspeaker used in LTspice .................. 9  
2.7 *force factor* ($Bl$) vs *diaphragm displacement* ($x$). Source: [3] 10  
2.8 *voice coil inductance* ($Le$) vs *diaphragm displacement* ($x$). Source: [3] 10  
2.9 *mechanical suspension compliance* ($Cm$) vs *diaphragm displacement* ($x$). Source: [3] 10  
2.10 Non-linear model of the loudspeaker in Simulink ................. 11  
3.1 Typical waveform of an audio file, showing high peak power and low rms values. Source: [7] ................................. 14  
3.2 Correctly and incorrectly sampled signals. Source: [8] ............ 14  
3.3 Aliasing: frequencies being mirrored due to wrong sampling. Source: [8] ......................................................... 15  
3.4 Typical frequency response of a first order low-pass filter ......... 16  
3.5 Block diagram of the whole power measurement circuit ............ 17  
3.6 Current measurement resistor in series with the loudspeaker 17  
3.7 Picture of the power measurement circuit .......................... 18  
3.8 *Sallen-Key* configuration for a 2nd order low-pass filter ....... 20  
3.9 Frequency response of the $8^{th}$ order low-pass filter, using the *Sallen-Key* configuration 21  
3.10 FFT of the unfiltered signal in the output of the DAC .......... 22  
3.11 FFT of the low-pass filtered signal in the output of the DAC 22  
3.12 FFT of the unfiltered signal in the output of the instrumentation amplifier (current channel) ............................ 23  
3.13 FFT of the low-pass filtered signal in the output of the instrument amplifier (current channel) 23  
3.14 Gain of the audio amplifier over the frequency, depending on the connected load ............................................. 26  
3.15 Matlab figure showing the constant gain of the amplifier for a high level signal: Resistive Load = Red, Loudspeaker Load = Blue. (for more precision, only the peaks of the sinusoidal waveforms are shown) 27  
3.16 Matlab figure showing the constant gain of the amplifier for a lower level signal: Resistive Load = Red, Loudspeaker Load = Blue. 28  
4.1 Band-pass filter used for the audio file on Simulink ............... 29
4.2 Comparison of the voltage between the real measurements (green) and the simulation on the linear model (blue) and non-linear model (red) for low level sound pressure level. The red line cannot be seen because it is exactly the same as the blue one. ................................................................. 31

4.3 Comparison of the current between the real measurements (green) and the simulation on the linear model (blue) and non-linear model (red) for low level sound pressure level. ..................................................... 32

4.4 Comparison of the voltage between the real measurements (green) and the simulation on the linear model (blue) and non-linear model (red) for high level sound pressure level. The red line cannot be seen because it is exactly the same as the blue one. ................................................................. 32

4.5 Comparison of the current between the real measurements (green) and the simulation on the linear model (blue) and non-linear model (red) for high level sound pressure level. ..................................................... 33

4.6 Comparison of the power curves of the real measurements (green) and the simulation on the linear model (blue) and non-linear model (red) ................................................................. 34

5.1 Window sweep over the power curve of the loudspeaker with different window sizes. Source: [1] ................................................................. 36

5.2 Maximum $rms$ values as a function of the window size. Source: [1] 37

5.3 Power window sweep algorithm applied to high level measurements. Real measurement (green), simulation on the linear model (blue) and non-linear model (red). ................................................................. 38

5.4 Power window sweep algorithm applied to low level measurements. Real measurement (green), simulation on the linear model (blue) and non-linear model (red). ................................................................. 38

5.5 Difference between the real measurements and each one of the simulation model for high level (difference with linear model = green, difference with non-linear model = red). ................................. 40

5.6 Difference between the real measurements and each one of the simulation model for low level (difference with linear model = green, difference with non-linear model = red). ................................. 40

6.1 Frequency response of the two filters that compose the $K$-filter. Source: [1] ................................................................. 42

6.2 Short period of song no. 1 showing the comparison of the power consumption (for high level). Real measurement (green), simulation on the linear model (blue) and non-linear model (red). 44
6.3 Short period of song no. 4 showing the comparison of the power consumption (for high level). Real measurement (green), simulation on the linear model (blue) and non-linear model (red).

6.4 Song Number 1: Max RMS Power VS Window Size HIGH LEVEL (Real Measurement = green, Simulation on Linear Model = blue, Simulation on Non-Linear Model = red).

6.5 Song Number 4: Max RMS Power VS Window Size HIGH LEVEL (Real Measurement = green, Simulation on Linear Model = blue, Simulation on Non-Linear Model = red).

6.6 Power window sweep comparison for all the songs on high level. Curves correspond to real measurements on the circuit (1-red, 2-blue, 3-green, 4-yellow, 5-magenta, 6-black).

6.7 Power window sweep comparison for all the songs on low level. Curves correspond to real measurements on the circuit (1-red, 2-blue, 3-green, 4-yellow, 5-magenta, 6-black).

7.1 Bode diagram showing the SPL of each loudspeaker over the frequency, theoretically measured at 1m distance with a 2.83V input voltage.

7.2 Bode diagram of the target SPL transfer function.

7.3 Bode diagram of the filter designed for each loudspeaker.

7.4 Simulink model used, where the linear model is combined with the filter of each loudspeaker.

7.5 Power consumption of each loudspeaker for the same SPL. Only a short period of the audio curve is shown.

7.6 Result of power window sweep algorithm applied to 128 loudspeakers.

7.7 histfit function used to display the distribution of the peak apparent power values.

7.8 histfit function used to display the distribution of the rms apparent power values for the 128 loudspeakers.

7.9 Ratio of the peak apparent power to the rms apparent power for the 128 loudspeakers.

List of Tables

1 Data from the measurement of the loudspeaker system when playing an audio file.

2 Peak and rms power values and their ratio, the crest factor, for each song.
1 Introduction

Nowadays loudspeakers can be found everywhere, from small electronic devices such as MP3 players and mobile phones, to professional level sound systems. The range of power requirements of these loudspeakers can also be very extensive, and the components of the system, such as the power supply, the cables, the amplifier, the filters and the loudspeakers need to be designed to meet these power specifications. Before the design of all these parts, it is important to perform investigation of different loudspeakers and configurations when they are playing audio files, to get knowledge about the power requirements.

The project presented here follows the work done by a group of PhD students and professors of the Technical University of Denmark: *Requirements Specification for Amplifiers and Power Supplies in Active Loudspeakers* [1]. Here, a mathematical model for the loudspeaker is described, which can be used to simulate the behaviour of the loudspeaker in order to perform measurements. This paper also introduces a method to get the power requirements of the loudspeaker more clearly, when playing an audio file.

For this project, a circuit to measure the power of a real loudspeaker has been designed and built in order to check that the model of the loudspeaker introduced in the previous paragraph gives similar results to the real loudspeaker. After that, different measurements have been performed and the results of the real loudspeaker and the models have been compared.

1.1 Thesis Objective

The main motivation of the project is to avoid oversizing and unnecessary cost in the elements of a sound system, such as the power supply, the amplifier and the loudspeaker driver. For this, a lot of research needs to be done, and this project focuses in part of this investigation.

The paper mentioned before [1] explains how loudspeakers have a high power consumption for very short time periods and low consumption for longer periods (high power peaks and low continuous values). However, manufacturers usually test the power requirements of their loudspeakers using test-signals that have a constant power, which is not the case of the common audio files, so their loudspeakers are oversized. It is believed that this oversizing can be avoided and costs can be reduced if it is known for how long these peaks of power last and how often they are repeated, because, this way, a sound system that meets the power requirements of real audio files can be designed.

In [1], a research of the power requirements of over 400 audio files is
explained, and the mathematical model of the loudspeaker is used to make simulations. The main goal of this project is to compare results of measurements in mathematical models with real measurements, to check that the models are valid. Two models are used: the linear one and the non-linear one, which is more complex. It will be analysed if the linear model is valid to simulate the behaviour of the real loudspeaker or if, instead, the non-linear model is needed.

Finally, an statistical analysis of power requirements of multiple similar loudspeakers is performed, in order to know what is the power consumption range of all of them. The goal of this is to be able to design the power supply unit and the rest of the parts of the sound system so that they meet the power requirements of all the loudspeakers, so design for the worst case needs to be done. The design of these components is not part of this project, but would be a future step.

1.2 Organisation of the Thesis

The chapters of this document are explained in this section:

In Chapter 2, the dynamic loudspeaker is explained. First, its physical parts are explained, and after that the electrical, mechanical and acoustical domains are analysed. All the domains are included in the mathematical linear model of the loudspeaker. Finally, the non-linearities are introduced to the model.

Chapter 3 introduces first the concept of audio power. After that, the Sampling Theorem and aliasing are explained. It is also in this chapter where the circuit to make the loudspeaker power measurements is explained: the components and software tools used are introduced, and the design of the anti-aliasing filters is described. Finally, the linearity of the audio amplifier over the frequency is measured.

In Chapter 4 the first measurements are done in the circuit described in the previous chapter. These measurements are afterwards compared to results obtained with the linear and non-linear models of the loudspeaker.

Chapter 5 explains first the power window sweep algorithm used in this project to compare the results of the power measured with different methods. First, the algorithm is explained, and then it is applied to the power consumption curves measured in Chapter 4.

In Chapter 6, loudness normalisation is explained. Then, it is applied to six different songs to make them sound equally loud. Finally, the same measurements and comparisons explained in Chapters 5 and 6 are made for these six songs.
Chapter 7 is a little bit different than the rest. Here, a database of more than 100 similar loudspeakers is taken and a target sound pressure level response is designed, to make all of them sound equally loud. For this, one filter is designed for each one of them, that will make all the loudspeakers have the same sound response. Finally, the same measurements (only simulations) and comparisons as in Chapters 5 and 6 are applied to all the loudspeakers, and their power requirements are compared.

At the end, the final conclusions of the project are stated, and the future work that follows is explained.

The appendices explain the files that are related to this project. However, these files are not needed for the understanding of this document.
2 Loudspeaker Modelling

2.1 Loudspeaker Parts And Operation

[2] The loudspeaker is an electroacoustic transducer that converts an electrical signal to an acoustic signal. The main parts of the dynamic loudspeaker are the voice coil, the pole piece, the magnet, the diaphragm or cone, the spider, the dust cap, the outer suspension, the top plate and the basket.

![Loudspeaker Driver](image)

Figure 2.1: Loudspeaker Driver. Source:[5]

When the loudspeaker receives an electrical signal, the current flows in the voice coil and an electro-magnetic force is induced in the permanent magnetic field in the air gap. This force causes the displacement of the voice coil, which is then transferred to the diaphragm and this one radiates sound. The loudspeaker can be split into three domains:

- Electrical domain
- Mechanical domain
- Acoustical domain

Each one of these domains and the conversion between them is explained in the next section.
2.2 Domains and Conversions

2.2.1 Electrical Domain

[2] The analogous electrical circuit of the loudspeaker is formed with the impedance of the voice coil, which is modelled as a resistor $R_e$ in series with an inductor $L_e$. $u(t)$ is the input voltage signal, which introduces a current in the voice coil, $i(t)$. $Blv(t)$ is the force induced to the mechanical domain in the voice coil, which makes the diaphragm move with a velocity $v(t)$. $Bl$ is the force factor where $B$ is the magnetic flux density in the air gap and $l$ is the effective length of the voice coil.

The parameters mentioned before, as well as some others, are called Thiele-Small Parameters, and are usually found in the data sheets of the loudspeakers. These electro-mechanical parameters define the performance of a loudspeaker driver.

![Electrical model of the voice coil. Source: [3]](image)

Applying Kirchhoff’s voltage law to the previous circuit, the transfer function of the electrical circuit is shown in equation (2.1).

$$u(t) = R_e i(t) + j\omega L_e i(t) + Bl v(t)$$  \hspace{1cm} (2.1)

where $\omega = 2\pi f$ and $f$ is the frequency of the input signal.

This project is focused in Woofer loudspeakers, thus, in the low frequency range. The previous electrical model is sufficient for this frequency range, but, if higher frequencies were being analysed, the effect of Eddy currents would have to be taken into account and a resistor $R'_e$ would need to be added to the model in parallel with $L_e$.

2.2.2 Mechanical Domain

The mechanical circuit of the loudspeaker includes: $C_m$; the suspension compliance, $M_m$; the mass of the diaphragm, and $R_m$; the mechanical damping
factor. $Bli(t)$ is the force induced from the electrical domain into the mechanical domain, which causes the diaphragm to move with a velocity $v(t)$ due to the input current $i(t)$. Finally, $Z_A$ is the representation of the acoustic impedance in the mechanical domain. When the loudspeaker is not mounted in any kind of box (free air configuration), the term $Z_A$ can be neglected.

When representing the mechanical domain in an analogous electrical circuit, $R_m$ acts as a resistor, $M_m$ as an inductor and $c_m$ as a capacitor. The force applied by the current running through the voice coil is represented as a voltage source, and the velocity it generates in the diaphragm is represented as current flowing through the circuit. This way, the analogous electrical circuit can be represented as in figure 2.3.

![Figure 2.3: Mechanical model of the loudspeaker. Source: [3]](image)

Again, applying Kirchoff’s voltage law, the transfer function is as in equation (2.2).

$$Bli(t) = R_m v(t) + j\omega M_m v(t) + \frac{1}{j\omega c_m} v(t) + Z_A$$ \hspace{1cm} (2.2)

As explained before, the term $Z_A$ can be neglected when the loudspeaker is in free air configuration. In this project, all the measurements and simulations have been done using this configuration, so from now on, the term $Z_A$ will be neglected.

### 2.2.3 Acoustical Domain

The most important acoustic parameters of a loudspeaker system are those that are determined by the enclosure of the speaker. This is why, when working in the free air configuration, no model is needed for the Acoustical Domain.

The acoustic pressure level at a distance $r$ from the loudspeaker is defined as in equation (2.3).
\[ p(r) = \frac{\rho}{2\pi r} s U_d(s) \]  

(2.3)

\( U_d(s) \) is the displaced air volume, and is calculated as \( U_d(s) = u_d(s) S_D \), where \( u_d(s) \) is the velocity of the diaphragm calculated in the mechanical domain, and \( S_D \) is the effective surface area of the loudspeaker driver, usually provided in the data sheet. \( \rho \) is the density of air.

The previous equation would give as a result the acoustic pressure in a pressure unit, such as Pa. However, when working with sound, the pressure level is often expressed in dB, due to the fact that this is more ‘realistic’, in the sense that it adjusts better to the human hearing. The sound pressure level in dB (SPL) is calculated as shown in equation (2.4).

\[ SPL(dB) = 20 \log \left( \frac{p(r)}{p_0} \right) \]  

(2.4)

where \( p_0 \) is the reference sound pressure level. Usually \( p_0 = 20\mu Pa \) is taken, as this value corresponds to the human hearing threshold (lowest pressure value that a human can hear).

### 2.3 Electro-Mechanical Conversion and Final Linear Model

Integrating the two circuits explained before into a single one, the loudspeaker can be modelled as it is shown in figure 2.4.

![Electro-Mechanical equivalent circuit of the loudspeaker.](image)

Figure 2.4: Electro-Mechanical equivalent circuit of the loudspeaker. Source:[3]

The force factor \((Bl)\) is the parameter for the conversion of energy from the electrical domain to the mechanical domain: the current flowing through the voice coil induces a velocity in the diaphragm in the mechanical domain.

Equations (2.1) and (2.2) can also be written in the Laplace domain as represented in equations (2.5) and (2.6).
\[ u(s) = (R_e + L_e s) i(t) + B l v(t) \]  \hspace{1cm} (2.5)

\[ B li(s) = \left( R_m + M_m s + \frac{1}{C_m s} \right) v(t) \]  \hspace{1cm} (2.6)

In this project, all the simulations have been done using the software Matlab. To integrate the previous model in Matlab, the third party software Simulink has been used inside Matlab. Following equations (2.5) and (2.6), the Simulink linear model of the loudspeaker used for this project is shown in figure 2.5.

\[ u(s) = \left( R_e + L_e s + \frac{1}{R_{el}} + \frac{1}{L_{el} s} + C_{el} s \right) i(s) \]  \hspace{1cm} (2.7)

\[ R_{el}, \ L_{el} \ \text{and} \ C_{el} \ \text{are the electrical equivalents of} \ R_m, \ C_m \ \text{and} \ M_m \ \text{respectively, and can be calculated as explained in equations (2.8) to (2.10).} \]

\[ R_{el} = \frac{(Bl)^2}{R_m} \]  \hspace{1cm} (2.8)

\[ L_{el} = (Bl)^2 C_m \]  \hspace{1cm} (2.9)
\[ C_{el} = \frac{M_m}{(Bl)^2} \]  

2.4 Non-Liniarities

The previously described linear model of the loudspeaker is valid for small sound pressure levels, where the diaphragm displacement is short. However, when the sound level is higher and the displacement of the diaphragm is longer, non-linearities appear in the model: several loudspeaker parameters are dependent on the displacement, so their value will change with the position of the diaphragm. The most important non-linear parameters are the force factor $Bl$, the electrical inductance $Le$ and the mechanical compliance $C_m$.

In the case of $Bl$, when the diaphragm moves long distances (some mm) in and out from its center position, the $l$ parameter (effective length of the voice coil) changes, so $Bl$ changes depending on the position $x(t)$ of the diaphragm. In the case of $Le$, the inductance value changes with the position of the voice coil as it moves in and out. Finally the mechanical compliance values change when the voice coil is driven far away from its center position.

Figures 2.7 to 2.9 show the non-linearities of this parameters that depend on the displacement.
Figure 2.7: force factor ($BL$) vs diaphragm displacement ($x$). Source: [3]

Figure 2.8: voice coil inductance ($Le$) vs diaphragm displacement ($x$). Source: [3]
2.5 Final Non-Linear Model

For the non-linear model, the non-linearities of $B_l$, $L_e$ and $C_m$ have to be introduced in the model explained before. In this project, the Klippel analysis software has been used to determine the coefficients for the equations of these three parameters as a function of the position of the piston. The resulting non-linear model is shown in figure 2.10. The main difference with the linear model is that $B_l$, $C_m$ and $L_e$ are now calculated as polynomials of the position, using the coefficients of the Klippel analyser.
Figure 2.10: Non-linear model of the loudspeaker in Simulink
3 Loudspeaker Power Measurement Circuit

3.1 Audio Power, RMS and Peak Power

Audio power is the electrical power transferred from the audio amplifier to the loudspeaker. Although it is usually measured in Watts, when dealing with AC signals, that unit corresponds to the active power. In this project the apparent power is measured, so the unit used is VAR. The electrical power delivered to the loudspeaker determines the sound pressure level generated by this one.

The power of a loudspeaker can be calculated if the voltage in the terminals of the loudspeaker and the current passing through it are known. The average power of an audio signal played in a loudspeaker can be calculated as in equation (3.1).

\[
P_{\text{avg}} = \frac{1}{T} \int_{0}^{T} v(t)i(t) \, dt
\]  
(3.1)

In this project, both the measurements and the simulations have been done using discrete signals, with a sampling frequency of 44.1kHz, which is the typical sampling frequency used in audio. In most cases, instead of calculating the average power in the way shown above, the instantaneous power has been calculated, which corresponds to the apparent power of each sample, as in equation (3.2).

\[
P_i = V_i I_i
\]  
(3.2)

With the power values of each sample, the \textit{rms} Power and the \textit{Peak Power} can be calculated. For \( n \) values \( \{P_1, P_2, ..., P_n\} \), these parameters are calculated as in equations (3.3) and (3.4).

\[
P_{\text{rms}} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} P_i^2}
\]  
(3.3)

\[
P_{\text{peak}} = \max\{P_1, P_2, ..., P_n\}
\]  
(3.4)

These two parameters are very important when dealing with audio signals, because these signals have a wide amplitude range (large variation of \textit{rms} values over time). This means that the \textit{rms} value and the peak value will be very different depending on the amount of samples taken. Varying the window size (amount of samples taken) and the position of the window through the audio file can give extremely different results, due to the fact...
that audio signals typically have a very high \textit{crest factor}, which is calculated as the ratio of the peak power of the audio signal to the \textit{rms} value of it.

![Typical waveform of an audio file](image)

Figure 3.1: Typical waveform of an audio file, showing high peak power and low \textit{rms} values. Source:[7]

### 3.2 Sampling Theorem and Aliasing

\[ f_s \geq 2f_{\text{max}} \]  

Figure 3.2 shows the different resulting waveforms depending on whether the sampling theorem has been accomplished or not.

![Correctly and incorrectly sampled signals](image)

Figure 3.2: Correctly and incorrectly sampled signals. Source:[8]
The sampling theorem is also applied to audio signals: the human hearing frequency range goes from 20 Hz to 20 kHz, which is the bandwidth of the audio signals. The sampling frequency must be, then, over 40 kHz. This is why audio signals are often sampled at 44.1 kHz or above.

Aliasing is the effect that causes a signal to become indistinguishable when it is sampled without the sampling theorem being accomplished. Aliasing can also happen if the sampling theorem is accomplished: sampling generates low-frequency aliases, but their amplitude levels are very low, so they do not mean a problem. But if the signal has frequency components higher than $f_s/2$ (sampling theorem not accomplished), these frequencies will be mirrored back into the bandwidth of the sampling frequency. In this case, there is wrong information about the measured signal.

![Figure 3.3: Aliasing: frequencies being mirrored due to wrong sampling. Source: [8]](image)

Aliasing is avoided using low-pass filters. The signal is filtered with a cutoff frequency ($f_c$) below $f_s/2$, and this way higher frequency components are eliminated. However, analog filters are not ideal and they still let some signal pass through even if its frequency is higher than $f_c$, but the amplitude is reduced. The amount of reduction applied to higher frequencies is determined by the order of the filter. Each increment of the order has a roll-off of 20 dB/decade. So a higher order filter will be more effective to reduce frequency components over $f_s$.

In this project, the sampling frequency used was 44.1 kHz, so the sampling theorem was accomplished. However, anti-aliasing low-pass filters were used, and there are two main reasons for this:

The first one is that the signals may contain frequency components over the audio bandwidth (20 Hz to 20 kHz), such as noise and other sources. These components might be mirrored back to the signal frequency spectrum due to sampling, so they need to be cut off.

The second reason is something that can be understood more easily if the next sections of this paper are read. Basically, there are two sampling steps
in this project: the first one, when the signal is converted from digital to the analog domain, to send it to the amplifier and then to the loudspeaker. In this first step, there is no problem with aliasing, as the signal’s original bandwidth remains unaffected. However, images higher than the original bandwidth will appear due to sampling. The second stage of sampling is when the current and voltage signals of the loudspeaker are sent back to the computer. These two signals have the mirrored frequency components due to the first stage of sampling, and if no filter is used, these components will be mirrored in the original signal’s frequency bandwidth (audio frequency range), affecting it wrongly.

3.3 Power Measurement Circuit: Main Components

In this section the circuit built for this project will be explained shortly, and the main components and their functionality will be explained.

As explained before, a circuit has been built to measure the power of the loudspeaker when it is playing a certain audio file. For this, the signal is sent from the computer using a DAC and is first low-pass filtered, before it goes in the audio amplifier. After that, the signal goes into the loudspeaker, which is in series with a current-measuring resistor. A dual instrumentation amplifier is used to measure the voltage in the terminals of the loudspeaker and the current passing through it. The outputs of the instrumentation amplifier are low-pass filtered again and finally stored in the computer via an ADC.
The signal flow of this circuit is shown in figure 3.5. Figure 3.6 shows how a small resistor is placed in series with the loudspeaker (which in this case is simplified as an 8Ω resistor) to measure the current. The voltage measured in the terminals of this resistor is proportional to the current through it (Ohm’s Law). Finally, figure 3.7 shows a picture of the measurement circuit, showing its most important parts.

Figure 3.5: Block diagram of the whole power measurement circuit

Figure 3.6: Current measurement resistor in series with the loudspeaker
3.3.1 The Loudspeaker: *Monacor SPH-170TC*

The loudspeaker chosen for this project is the *SPH-170 TC*, by *Monacor*. It is a dual 6.5" Woofer (only one channel has been used) with a power handling of $40\, \text{W}_{\text{rms}}$ and $60\, \text{W}_{\text{max}}$. It has an $8\, \Omega$ impedance. Its resonant frequency is at $33\, \text{Hz}$ and its frequency bandwidth goes up to $5000\, \text{Hz}$.

3.3.2 The Amplifier: *Texas Instruments TPA3116D2*

The *TPA3116D2*, by *Texas Instruments*, is a Class-D stereo amplifier. Its maximum output power is $50\, \text{W}$, and it needs a $24\, \text{V}$ single power supply. It is important to know that this is a full-bridge amplifier, which means that its output is floating. This fact affects other components.
3.3.3 The $10m\Omega$ Current Measurement Resistor

A $10m\Omega$ current measuring resistor, placed in series with the loudspeaker, has been used to measure the current going through it. This is done measuring the voltage at the terminals of the resistor, which is directly proportional to the current.

3.3.4 The Instrumentation Amplifier: *Texas Instruments INA2128*

The *INA2128* dual instrumentation amplifier has been used to get the voltage and current signals of the loudspeaker and send them to the computer. Each one of the channels is designed with three operational amplifiers, and has individual gain control using only one external resistor. The gain can be set from 1 to 1000. This instrumentation amplifier is supplied at $\pm12V$, and voltage regulators have been used to avoid any unexpected change in the supply voltage.

The reason for using this component is because both the voltage and current signals are floating, and their amplitude is too high for the ADC. So the functionality of the instrumentation amplifier is to make the signals ground-referenced and have the proper amplitude to be sent to the ADC.

The gain of each channel is set by the following formula: $G = 1 + \frac{50k}{R_G}$. To set this value, it must be known that the input voltage range for the ADC is $\pm10V$, so the outputs of the instrumentation amplifier must be between these limits. For the current channel, the voltage in the terminals of the $10m\Omega$ resistor is always smaller than $50mV$, so it needs a high gain. A resistor of $270\Omega$ is used, which, according to the formula before, gives a gain of around 186. For the voltage channel, however, the signal should be reduced, because it can go up to $20V$ in the output of the amplifier. For this, a voltage divider is placed before the instrumentation amplifier, and then the signal is amplified again. The gain is set to be 4.125 using a resistor of $16k\Omega$.

3.3.5 Operational Amplifiers for Filter design: *MC33079*

As explained before, when converting audio signals from the digital to the analog domain and vice versa, aliasing problems might appear due to improper sampling. This is exactly the case of this project, and this is why anti-aliasing filters need to be included.

In the power measurement circuit, aliasing can appear due to two elements mainly: the first one is the DAC/ADC: the sampling frequency for this one is $44.1kHz$ (typical for audio signals). The second one is the audio amplifier, which has a switching frequency of $400kHz$. 
To avoid this, two low-pass filters are used: one at the input of the amplifier and one (per channel) at the output of the instrumentation amplifier. These two filters use the Sallen-Key topology for low pass filters [10]. The software Filter Wizard, provided by Analog Devices [11] was also used as an initial help for the design. Figure 3.8 shows the typical Sallen-Key configuration 2nd order low-pass filter.

![Sallen-Key configuration for a 2nd order low-pass filter](image)

Figure 3.8: Sallen-Key configuration for a 2nd order low-pass filter

The main parameters of this filter are the cut-off frequency $f_c$ and the quality factor $Q$. They are calculated as explained in equations (3.6) and (3.7).

$$f_c = \frac{1}{2\pi \sqrt{R_1R_2C_1C_2}} \quad (3.6)$$

$$Q = \frac{\sqrt{R_1R_2C_1C_2}}{C_2(R_1 + R_2)} \quad (3.7)$$

For this project, the components chosen are:

- $R_1 = 7.5k\Omega$
- $R_1 = 130k\Omega$
- $C_1 = 680pF$
- $C_2 = 56pF$

This gives $f_c = 26.12kHz$ and $Q = 0.791$. $f_c$ is chosen to be around $25kHz$ to reduce the images due to the sampling frequency of the ADC at $44.1kHz$ and the switching frequency of the audio amplifier at $400kHz$, but bearing in mind that all the audio signal, below $20kHz$, must remain unaffected.
The operational amplifier chosen for the implementation of the filters is the MC33079. It is a low noise quad operational amplifier with a high bandwidth, up to 15 MHz, which makes it ideal for this application.

In the input of the amplifier, a 4th order filter is needed: to reach a 4th order filter with the Sallen-Key topology, two stages as the one shown before are needed, connected in series. For this, two channels of the MC33079 quad operational amplifier were used. At the output of each one of the channels of the instrumentation amplifier, another 4th order low-pass filter is needed, so all the four channels of the operational amplifier are used (two per output of the instrumentation amplifier).

As a result, the images due to the sampling frequency are reduced with a 8th order low-pass filter (two 4th order in series), and the images due to the switching frequency of the amplifier are reduced with a 4th order filter. The resulting frequency response of an 8th order filter is shown in figure 3.9.

![Frequency response of the 8th order low-pass filter, using the Sallen-Key configuration](image)

**Figure 3.9:** Frequency response of the 8th order low-pass filter, using the Sallen-Key configuration

### 3.4 Testing Anti-Aliasing Filters

After including the filters in the circuit, measurements are taken to check if they are efficient to avoid aliasing. For this, the LeCroy Wavesurfer 10 oscilloscope is used. This oscilloscope has the option to do the Fourier Frequency Transform (FFT) of an input signal, and it is used to see if there
are images near the two frequencies mentioned before due to aliasing. The measurements confirmed that aliasing is happening (before the filters), as it can be seen in the next figures, even if the amplitude of the mirrored signals is very small.

The figure 3.10 shows the FFT of the output signal from the DAC. This signal has low voltage because it is passed through the amplifier yet. In the figure, images at the right side of 44.1 kHz can be seen, which are a mirror of the frequencies in the left, but with a decreased amplitude.

![Figure 3.10: FFT of the unfiltered signal in the output of the DAC](image)

After the 4th order low-pass filter applied at the output of the DAC, it can be seen in figure 3.11 how the frequencies around 44.1 kHz have disappeared. This is very important because, even if the images have a small amplitude, they could produce sampling when the signal is amplified and sent back to the computer via the ADC, sampled at 44.1 kHz.

![Figure 3.11: FFT of the low-pass filtered signal in the output of the DAC](image)
The same explanation is applied for the 400kHz switching frequency of the audio amplifier: figure 3.12 shows the images around 400kHz. The signal corresponds to the output signal of the instrumentation amplifier channel measuring the current through the loudspeaker.

![Figure 3.12: FFT of the unfiltered signal in the output of the instrumentation amplifier (current channel)](image)

After the filter, it can be seen again in figure 3.13 how no images appear around 400kHz so aliasing is avoided when the signal is sent to the ADC.

![Figure 3.13: FFT of the low-pass filtered signal in the output of the instrumentation amplifier (current channel)](image)

After seeing the previous images, the importance of the first filter is obvious. However, it is not clear if the second filter is absolutely necessary due to the fact that the images around 400kHz have a very small amplitude, and after the filter there is still some very small amplitude noise which cannot be eliminated by the filter.
3.5 ADC & Software: Matlab & National Instruments USB-6356

The main system used in this project to send, receive and store audio signals to and from the power measurement system is composed by the USB-6356 data acquisition toolbox, by National Instruments, and the software Matlab.

3.5.1 Matlab

Matlab is a numerical computing environment and high level programming language. It is developed by Mathworks. It allows plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with other programs and software devices.

In this project Matlab is used with the previously mentioned data acquisition device called USB-6356. Matlab includes some functions that let it communicate with devices of National Instruments. A review of these functions is done in the next section.

The other use of Matlab in this project is the realisation of simulations using the loudspeaker model explained in the second chapter of this document. For the simulations, the software Simulink is used, which is one of the Matlab’s 3rd party software tools. Simulink is a data flow graphical programming language tool. It includes a graphical block diagraming tool, which is used here to implement the different loudspeaker models.

3.5.2 NI USB-6356

The USB-6356 by National Instruments is the data acquisition device used at this project. It can be used for several purposes, but in this project it is mainly used as an Digital-to-Analog and Analog-to-Digital Converter and data storing element, always in communication with Matlab. Some of the most important parameters of the device are:

- 8 Analog Inputs (floating or ground referenced, DC coupling)
- 2 Analog Outputs (ground referenced, DC coupling)
- Maximum Input/Output range: ±10V
- Input and output resolution: 16bits
- Maximum sampling frequency: 1.25MHz
The next few lines show some basic functions to use the National Instruments devices on Matlab. In this simple example, a sinusoidal signal is sent to the first output of the device and is then sent back and stored in Matlab via the first input.

```matlab
%Initial example: 300 Hz sinusoidal output
%1st output is directly connected to 1st input

%Detect and list the devices connected to the computer
d = daq.getDevices;
%Establish connection to the National Instruments device
s = daq.createSession('ni');

%Initialize first input and output. Voltage measurement
s.addAnalogInputChannel('Dev1',0,'Voltage');
s.addAnalogOutputChannel('Dev1',0,'Voltage');

%Set the sampling frequency for both input and output channels
fs = 44.1e3;
s.Rate = fs;

%Set the maximum voltage range
s.Channels(1).Range = [-10 10];

%Create sinusoidal signal

t = 0:1/fs:10-1/fs; %Run for 10 seconds
signal = sin(2*pi*300*t);

%Output signal. The conversion starts and runs for the specified time
s.queueOutputData(signal');

%Get data from device input and store it in a Matlab variable
data = s.startForeground;
```

### 3.6 Audio Amplifier Model

The power measurement circuit has two main components: the audio amplifier and the loudspeaker. When sending an audio signal to the circuit, this one is amplified and played in the loudspeaker. For the simulations, the signal sent to the loudspeaker must be the same as in the real measurements, so a model of the amplifier needs to be included in the simulation.

It is very important to check if the TPA3116D2 audio amplifier has a linear gain over the frequency. For this, a high precision audio analyser is used: this device sends a frequency-swept signal to the amplifier and determines if the output voltage of it is constant in amplitude over the frequency, or instead, if the gain is different depending on the frequency. Three different
measurements have been done: first an 8Ω purely resistive load is connected in the output of the amplifier. Secondly the loudspeaker is connected, and, lastly, no load is connected. The results are shown in the next figure 3.14.

![Figure 3.14: Gain of the audio amplifier over the frequency, depending on the connected load](image)

The figure shows that the gain of the amplifier is almost constant for an purely resistive 8Ω load in the audio frequency range (20\(Hz\) to 20\(kHz\)). However, when the loudspeaker is connected, the results show a small resonance peak around 60\(Hz\). Around 500\(Hz\) the gain is the same as in the resistive load, but at 1\(kHz\) it begins to fall, and finally raises exponentially when passed 10\(kHz\). The results without a load show a higher gain in general, but this case is not so important for this project.

Although the amplifier is not linear, only the range from around 200\(Hz\) to 2\(kHz\) is important for this project, because this is more or less the frequencies that a woofer loudspeaker will reproduce. In the mentioned range, the gain is almost linear, even if it drops when it passes 1\(kHz\). This means that in the simulations the amplifier can be modelled using only a linear gain block of 25.7\(dB\).

To check this, measurements have been done in the power measurement circuit: first a frequency-swept sinusoidal signal, from 20\(Hz\) to 4\(kHz\) is sent to the loudspeaker and the output voltage of the amplifier is stored. Then the same is done with a 8Ω resistor load, instead of the amplifier. This is done for both a high level voltage signal and a lower level voltage signal. The results of the comparison between the output voltage of each measurement show that they are very similar and that the gain is very constant in the
(a) Low frequency range (around 250Hz)

(b) Higher frequency range (around 1.7kHz)

Figure 3.15: Matlab figure showing the constant gain of the amplifier for a high level signal: Resistive Load = Red, Loudspeaker Load = Blue. (for more precision, only the peaks of the sinusoidal waveforms are shown)

interesting frequency range. All of this confirms that, in the simulations, the amplifier can be modelled simply with a gain block. The difference errors in both cases are under 1%.
(a) Low frequency range (around 250 Hz)

(b) Higher frequency range (around 1.7 kHz)

Figure 3.16: Matlab figure showing the constant gain of the amplifier for a lower level signal: Resistive Load = Red, Loudspeaker Load = Blue.
4  Loudspeaker System Measurements, Simulations & Comparison

After the power measurement circuit has been built and tested, measurements are taken sending audio signals to the loudspeaker and measuring the power consumption of the driver. The results of these measurements are then compared to results of simulations using the linear and non-linear model of the loudspeaker.

4.1 Preparing the Audio File

Before playing any audio file in the loudspeaker, it needs to be taken into account that it is a woofer. In a two-way loudspeaker system, the woofer would be the part of the system reproducing the lower frequencies, and the tweeter driver would reproduce the higher ones. In a three-way loudspeaker, there would be a subwoofer, a woofer and a tweeter. This means that each one of these drivers does not have to reproduce the entire audio frequency range: instead, crossover filters are used, which divide the signal into the ways of the sound system and filter them for the proper frequency range for each driver. For example, as explained before, the audio signal of a woofer would be band-pass filtered between 200Hz and 2kHz, more or less.

For this project, the same is applied: the audio file is band-pass filtered before sending it to the loudspeaker system or the simulations. For this, Simulink plug-in is used on Matlab: the wav audio file is loaded on a variable and then sent to Simulink, where it is band-pass filtered and sent back to the workspace, to be finally stored in a new wav file. The figure 4.1 shows the Simulink diagram used. The commercial song used for most of the measurements is 'Give Life Back To Music', by 'Daft Punk', and it is chosen due to being a quite ’standard’ song nowadays, that could easily be heard on the media such as the radio, television or music websites.

![Figure 4.1: Band-pass filter used for the audio file on Simulink](image-url)
4.2 Power Measurements at two Sound Pressure Levels

After the audio file has been filtered, the measurements are done. For all the measurements (real and simulations) two sound pressure levels are used: the low level corresponds to background listening level, where the listener can easily have a normal conversation while listening to the background music, for example. The high level corresponds to when the listener wants to focus only on the music and wants to listen to it at a considerable volume, but not enough to be painful. The voltage sent to the amplifier in the lower level measurements is 10 times smaller (in linear scale) than the one in the higher level measurements. The gain of the audio amplifier is set at the maximum for all the simulations.

4.2.1 Circuit Measurements

First of all, real measurements are taken on the circuit: the filtered audio file is cut and done only 20 seconds long, for time saving reasons. Then it is sent to the power measurement circuit and the data from the instrumentation amplifier measuring the current and the voltage through the loudspeaker is sent back to the computer. After doing the necessary unit conversions, the current and voltage are stored into variables.

Table 1 shows some important parameters of the current and voltage signals stored for the two levels. As discussed before, it can be seen how the peak values are much higher than the \textit{rms} values (high \textit{crest factor}).

<table>
<thead>
<tr>
<th>Level</th>
<th>RMS Voltage (V)</th>
<th>Peak Voltage (V)</th>
<th>RMS Current (A)</th>
<th>Peak Current (A)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>0.1895</td>
<td>1.0664</td>
<td>0.0271</td>
<td>0.1461</td>
</tr>
<tr>
<td>High</td>
<td>1.8905</td>
<td>10.5369</td>
<td>0.2660</td>
<td>1.4240</td>
</tr>
</tbody>
</table>

Table 1: Data from the measurement of the loudspeaker system when playing an audio file

4.2.2 Simulations: Linear and Non-Linear Loudspeaker Models

Simulations have done with the same 20 seconds of the audio file using the linear and non-linear models of the loudspeaker in \textit{Simulink} and the current and voltage through the loudspeaker model are extracted, to be compared with the results of the real measurements. The peak values of current and
voltage are very similar to those of the real measurements, in both low and high sound pressure levels. The comparison between the results and the precision of them is discussed in the next section.

4.3 Comparison & Conclusions

There are several different ways to compare the results form the measurements in the circuit and in the simulations with different models. In figures 4.2 to 4.5, the voltage and current curves of the three different methods of measurement (real, linear model and non-linear model) are compared directly. In order to see the small differences with higher detail, only a small area is shown for these figures: from $t = 6.954s$ to $t = 6.964s$.

Figure 4.2: Comparison of the voltage between the real measurements (green) and the simulation on the linear model (blue) and non-linear model (red) for low level sound pressure level. The red line cannot be seen because it is exactly the same as the blue one.
Figure 4.3: Comparison of the current between the real measurements (green) and the simulation on the linear model (blue) and non-linear model (red) for low level sound pressure level.

Figure 4.4: Comparison of the voltage between the real measurements (green) and the simulation on the linear model (blue) and non-linear model (red) for high level sound pressure level. The red line cannot be seen because it is exactly the same as the blue one.
As it can be seen in the figures, the results from the different types of measurements are very satisfactory: the curves are very similar, both the voltage and the current. They both follow the same waveform, although they have slightly different values when peaks are reached. In general, the current measurements seem to be less precise: this makes sense, because more components have been used to measure the current through the loudspeaker than those to the voltage. All the components used in the measurements have small tolerances. However, all these tolerances have to be summed up: this means that it is possible that the small differences appreciated in the figures are introduced by the lack of precision of the components, rather than by the linear and non-linear models.

If the voltage and current waveforms are multiplied sample by sample, the apparent power can be displayed and compared. Figure 4.6 shows the apparent power curve compared for the 3 methods of measurement. It can be appreciated how the differences between the real and simulated measurements increase. This can be seen most clearly in the peak values, where the peaks of the real measurement can sometimes be slightly higher than those of the simulations. This errors appear because, already in the voltage and current curves, there were some tiny errors, so when these two curves are multiplied, the errors raise with a power of two.

It is difficult to explain exactly why these small errors appear mostly on
Figure 4.6: Comparison of the power curves of the real measurements (green) and the simulation on the linear model (blue) and non-linear model (red) the peaks. They could also appear due to a lack of precision of the Thiele-Small parameters, because their values can change when they are heated up due to high power. The best example of this is the change of the resistance value, $R_e$, which has not been included in the model. However, it can be seen clearly how the three curves follow the same waveform continuously, which ensures that both the linear and the non-linear model have a high precision and can simulate the behaviour of a real loudspeaker with high fidelity.

Even if some conclusions can be taken when comparing the apparent power curves, an algorithm is introduced in the next section, which is more suitable for comparing the different power curves and taking conclusions about loudspeaker power requirements.
5 Power Window Sweep

In chapter 4 some measurements have been done on the power measurement circuit and then compared to results of simulations with the linear and non-linear model of the loudspeaker. For this, the resulting current and voltage curves have been compared directly, and then multiplied to compare the power curve. Even if this can give some useful information about the power, it is difficult to see the differences between the three methods of measurement, so an algorithm must be implemented so as to show the differences of the power requirement measurement more clearly in the three cases. The algorithm will show to be very useful to see how, when the loudspeaker is playing an audio file, commonly the power peaks are very high in comparison with the $\text{rms}$ value of the power.

What it is explained in the next section is included in the content of a Convention Paper for the Audio Engineering Society (AES) called 'Requirements Specification for Amplifiers and Power Supplies in Active Loudspeakers' [1], written by a group of professors and PhD students of the Technical University of Denmark.

5.1 Power Window Sweep Algorithm

Let’s assume that the voltage and current signals of the loudspeaker are currently properly stored. As the signals have been digitalised with a certain sampling frequency, they are stored in a discrete way: each value corresponds to a sample. The instantaneous power can, then, be calculated multiplying the current and voltage values that correspond to the same sample. After doing this, the values of power of every sample are stored and the $\text{rms}$ values can be calculated. In the case of $n$ values $\{P_1, P_2, ..., P_n\}$:

$$P_{\text{rms}} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} P_i^2} \quad (5.1)$$

The way this algorithm is implemented is by changing the number of samples taken to calculate the $\text{rms}$ value of the power, which from now on will be called the window size. Assuming that $n$ is the window size and $N$ is the total length or the audio file (the total number of samples), all the possible windows are analysed for a given window size: this is done calculating first the $\text{rms}$ power of a window that starts on the $1^{\text{st}}$ sample and finishes on the sample $N - n$. Then, the window starts with the $2^{\text{nd}}$ sample, then with the $3^{\text{rd}}$ and so on, until the last window starts on the sample $N - n$ and finishes.
on the sample $N$. The maximum value of all the $rms$ values taken will be stored as the maximum $rms$ value for a window size $n$.

The final goal of this algorithm is to get the maximum $rms$ value for several different window sizes and compare them: first, a window size of only one sample is taken. In this particular case, the $rms$ value of the window will be the exactly the value of the sample. The maximum value will also be the highest power peak.

The window size is then increased and, thus, the $rms$ values of the windows will begin to be lower and lower. The last case is when the window size is equal to the length of the audio file, and there will be only one window, whose value is the $rms$ value of the song.

The importance of the algorithm resides in comparing the maximum $rms$ values of different window sizes and plotting them in the same graph. This will show something similar to what is shown in figure 5.2, where the maximum $rms$ power is high for short window sizes, but goes down very fast as the window size is increased. From a particular window size and on, the
maximum values will be almost the same.

![Graph showing power vs. window width](image)

Figure 5.2: Maximum $rms$ values as a function of the window size. Source:[1]

This algorithm gives important information about the power consumption of a loudspeaker when reproducing an audio file: it shows how the value of the power is very high for very short amounts of time. However, when a longer time window is taken, the $rms$ value of the power decreases exponentially, which means that the power peaks are very short and high compared to the power curve’s $rms$ value.

### 5.2 Comparing Results & Conclusions

The algorithm explained in 5.1 has been implemented on Matlab and used to analyse the measurements taken on the real loudspeaker system and the simulations with linear and non-linear models. To save time, the window size has been increased logarithmically, and not linearly. This way there is more information on smaller window sizes, where the maximum $rms$ power decreases exponentially. For these measurements, only 5 seconds of the power curves have been analysed, using 1000 different window sizes.

The figures 5.3 to 5.6 show the results of applying the power window sweep algorithm to both low and high sound pressure levels.
Figure 5.3: Power window sweep algorithm applied to high level measurements. Real measurement (green), simulation on the linear model (blue) and non-linear model (red).

Figure 5.4: Power window sweep algorithm applied to low level measurements. Real measurement (green), simulation on the linear model (blue) and non-linear model (red).
The figures show satisfactory results, that confirm that the linear and non-linear models of the loudspeaker are trustable models for the behaviour of the real loudspeaker driver. Both figures show how the curve corresponding to the non-linear model has higher peak values than the linear model. This is due to the fact that non-linearities affect most on the highest power peaks, when the position of the piston of the loudspeaker moves a considerable distance from its center position, and this makes some of the Thiele-Small parameters of the loudspeaker change. For larger window sizes, the curves of the two models are very similar.

Both for low and high sound pressure levels it can be seen how the power curve of the real measurements is very similar to the simulations for short window sizes (It should be closer to the one of the non-linear model than to the one of the linear model, but maybe this difference is due to errors of precision of the components in the circuit). As the window size increases, the real measurement shows to have higher maximum $rms$ power values than the simulations, but this difference is very small. This can be seen most clearly when the window size is $10^2$ samples. Finally, for very long window sizes, real measurements and simulations give very similar results. All of this can be seen more clearly in figures 5.5 and 5.6, where the difference of the power curve of the real measurement and each simulation is directly compared. It can be appreciated how the difference remains very low for both cases, again, confirming the fidelity of both models.
Figure 5.5: Difference between the real measurements and each one of the simulation model for high level (difference with linear model = green, difference with non-linear model = red)

Figure 5.6: Difference between the real measurements and each one of the simulation model for low level (difference with linear model = green, difference with non-linear model = red)
6 Loudness Normalisation & Measurements with Different Songs

In the previous section it was demonstrated how the linear and non-linear models can be used to simulate the behaviour of the loudspeaker driver. However, all the measurements before have been done using the same audio file. The aim of this project is not to compare the power requirements of different musical styles, but it is still important to check that the model is working for different audio files, belonging to various musical styles and with changing waveform characteristics. This is why the same simulations as before have been run with six audio files of different musical styles.

However, the audio files cannot be directly compared due to the fact that the music pieces belong to different styles: they are from different times and recorded in different ways. This means that the loudness perceived when we listen to them is not the same, and thus, they need to be loudness normalised.

6.1 Loudness Normalisation

Nowadays sound is important in almost every communication media: from songs on conventional loudspeakers to the television, the radio, the mobile phones, electronic devices, and so on. However, sometimes the audio files that all these devices reproduce have different loudness levels, and the listener finds himself adjusting the sound volume continuously. To avoid this, loudness normalisation needs to be applied to the audio files, so that they are all perceived with identical loudness level by the listener.

With digital audio, all the songs are peak normalised: their peak value is below $0\,\text{dB}$ (digitally, not in SPL), but this does not mean that all the digital audio files have the same loudness level, because for example their $\text{rms}$ values can be very different.

To perform loudness normalisation, the loudness level is first measured with a $K$-weighted filter curve, which is based on how humans perceive the loudness level of sound. This filter consists of two filters: the first one simulates the acoustic effect of the human head as a rigid sphere, while the second one, a high-pass filter, refers to the low frequency $b$-curve used to simulate the frequency sensitivity of the human hearing. The $K$-weighting method is an open standard defined by The International Telecommunication Union: ITU BS.1770. This filter is shown in figure 6.1.

After the $K$-weighting filter, the loudness is measured using three similar terms: $\text{LKFS}$, $\text{LUFS}$ and $\text{LU}$. $\text{LKFS}$ is an abbreviation of Loudness $K$-weighted Full Scale and one unit of $\text{LKFS}$ is equal to $1\,\text{dB}$. $\text{LUFS}$ stands
for Loudness Units Full Scale: despite having a different name, it is identical to LKFS, but it is used in different standards. Finally, LU stands for Loudness Unit and is a relative loudness measurement unit (the other two were absolute measures).

The audio file, measured in LKFS, is split into 400 ms blocks with an overlapping of 75%. Gating filters are used to remove audio content that should not be taken into account, such as quiet parts or very loud exceptional noises. Two threshold values are used for this, an absolute one and a relative one. In this project the absolute value used is $-70 \, dB$ and the relative one is $-10 \, dB$ (compared to the average loudness of the song).

Finally, this algorithm gives a value of the perceived loudness in $dB$ for the audio file analysed. In order for two or more audio files to be loudness normalised, this loudness value in $dB$ needs to be the same for all of them, so the lowest one is taken as a reference, and a ‘reduction gain’ is be applied to all the others to make them match the reference loudness value (if the reference value taken is higher than the one corresponding to the audio file with the lowest loudness, when applying a gain, saturation would be introduced, because the highest peak of the lowest one would be over 0 dB).

In this project the algorithm explained above has been implemented using the loudness itu function in Matlab. This function follows the ITU-R BS.1770-2 specification.

6.2 Simulations with Different Songs

The algorithm explained above has been applied to six songs of different musical styles. Then the steps explained in the previous chapters have been
applied and the power requirements of the loudspeaker driver reproducing the six songs has been analysed. Simulations have been compared to real measurements.

6.2.1 Preparation of the Songs

Six songs have been chosen for this part of the project. The style of these audio pieces ranges from classical music to dubstep, which means that they are from very different seasons and their loudness values will be very different. The songs chosen are:

- 1 - Waking On A Pretty Daze - Kurt Vile
- 2 - Wilkie - Roman Flgel
- 3 - Creature Fear - Bon Iver
- 4 - Up Down Up Down - Koreless
- 5 - Bangarang feat. Sirah - Skrillex
- 6 - Requiem for soloists, chorus and orchestra, K. 626: Requiem aeternam - Helmuth Rilling, Mozart

These songs have been loudness normalised as explained in section 6.1, using the loudness_itu function on Matlab. All of them have been matched to the loudness level of the 6th song, which showed to be the one with the lowest loudness level. After that, all the songs have been filtered to be suitable for a 6.5" woofer driver, as explained in section 4.1.

6.2.2 Measurements

The steps followed to perform the measurements in the circuit and the simulations of the loudspeaker playing these audio files are the same as the ones explained in section 4.2. In this case, the voltage and current values of each sample taken from the real simulations, the linear and the non-linear models of the loudspeaker have been directly multiplied, and the comparison of the apparent power consumption of the loudspeaker for two of the songs is shown in figures 6.2 and 6.3. The reason for showing only the results of songs 1 and 4 is because they are the most ‘extreme’ cases, with the highest and lowest peak values. The measurements have been done again in 2 different sound pressure levels, but the results shown here correspond only to the high level.

Even if they belong to very different musical styles, it can be clearly appreciated how the waveforms of the real measurements and the simulations
still match almost perfectly. Again, some small differences can be found in the highest peaks, but the models behave almost exactly as the real loudspeaker.

Figure 6.2: Short period of song no. 1 showing the comparison of the power consumption (for high level). Real measurement (green), simulation on the linear model (blue) and non-linear model (red).

6.2.3 Power Window Sweep

The same power window sweep algorithm as the one explained in chapter 5 is applied here. In this case, only 4 seconds of each song are taken, for time saving reasons, and the amount of window sizes taken for each audio piece is 1000. Figures 6.4 and 6.5 show the result of applying the algorithm to songs 1 and 4 (again, the most extreme cases). However, it needs to be taken into account that only a small piece of the whole audio file is taken, so the peak values do not correspond to the maximum peaks of the whole files, but only of the parts taken.
Figure 6.3: Short period of song no. 4 showing the comparison of the power consumption (for high level). Real measurement (green), simulation on the linear model (blue) and non-linear model (red).

Figure 6.4: Song Number 1: Max RMS Power VS Window Size HIGH LEVEL (Real Measurement = green, Simulation on Linear Model = blue, Simulation on Non-Linear Model = red)

It can be seen how the curves of the real measurements and the simula-
tions match again, as expected. In this case, the real measurement shows higher power consumption than the simulations, but with a very small difference. Comparing the results of the two simulation models, the non-linear one shows higher peaks due to the displacement of the loudspeaker diaphragm.

Finally, figures 6.6 and 6.7 show the comparison of the algorithm applied to all the songs, each one for one sound pressure level. The range of power consumption is very different for each one of the songs. This can be due to two reasons: first one, as loudness normalisation has been applied to all the songs, it is possible that the maximum peaks of some of them have been reduced. However, in this case, the \textit{rms} values of all the songs should be quite similar because they all have the same theoretical loudness, but this is not true due to the second reason: only a small part of the audio file has been considered (4 seconds), and it is unknown if this part corresponds to a high or low level part of each song (a low level part could be the introduction of the song, while a high level part would be the chorus, for example).

In order to display the data of each song more clearly, \textit{table} 2 shows the \textit{rms} power, peak power, and the ratio of the peak to the \textit{rms}, which is the crest factor, for each one of the songs. In this case, 30 seconds of the whole audio piece are analysed, and not only 4 seconds, to give a much better idea of these parameters in longer time. The large variation of the crest factor is
due to the different musical styles.

Figure 6.6: Power window sweep comparison for all the songs on high level. Curves correspond to real measurements on the circuit (1-red, 2-blue, 3-green, 4-yellow, 5-magenta, 6-black).
Figure 6.7: Power window sweep comparison for all the songs on low level. Curves correspond to real measurements on the circuit (1-red, 2-blue, 3-green, 4-yellow, 5-magenta, 6-black).

<table>
<thead>
<tr>
<th>Song</th>
<th>RMS Power (VAR)</th>
<th>Peak Power (VAR)</th>
<th>Crest Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.3968</td>
<td>6.0026</td>
<td>15.1279</td>
</tr>
<tr>
<td>2</td>
<td>0.3167</td>
<td>7.3907</td>
<td>23.3357</td>
</tr>
<tr>
<td>3</td>
<td>1.1519</td>
<td>18.7844</td>
<td>16.3080</td>
</tr>
<tr>
<td>4</td>
<td>0.1296</td>
<td>2.0323</td>
<td>15.6812</td>
</tr>
<tr>
<td>5</td>
<td>0.2732</td>
<td>3.6056</td>
<td>13.1965</td>
</tr>
<tr>
<td>6</td>
<td>0.2980</td>
<td>5.0153</td>
<td>16.8322</td>
</tr>
</tbody>
</table>

Table 2: Peak and \( \text{rms} \) power values and their ratio, the crest factor, for each song.
7 Statistical Analysis of Loudspeaker Power Requirements

This section describes a work done in this project that differs a little bit from the rest of the sections. For the statistical analysis of loudspeaker power requirements, 128 loudspeakers with similar characteristics have been taken, and their power consumption is compared.

This is done defining a target sound pressure level that every loudspeaker should have. Then, the power consumption of each one of them is analysed. This gives an idea on the range of power consumption values of similar loudspeakers, and their efficiency compared with the rest.

7.1 Thiele-Small Parameters Database

First of all, the Thiele-Small parameters of many loudspeakers are needed. However, in order to make reasonable comparisons between them, they need to be quite similar. To achieve this, only woofers of 6.5\textquotesingle\ diameter have been analysed. The Thiele-Small parameters have been taken from [15], and an Excel file with 128 loudspeakers of 6.5\textquotesingle\ diameter has been created. The parameters that are needed are:

- $S_d$ : piston area of the diaphragm ($cm^2$)
- $R_e, L_e$ : electrical resistance and inductance of the voice coil ($\Omega, mH$)
- $M_m, C_m, R_m$ : mechanical mass, suspension compliance and damping factor of the diaphragm ($g, N/m, N*s/m$)
- $B_l$ : force factor ($T*m$)

All these parameters are loaded on Matlab in order to define one transfer function for each loudspeaker: the input to this transfer function is the voltage applied to the loudspeaker, and the output is the sound pressure level. So, this transfer function defines the behaviour and efficiency of the loudspeaker driver: it determines what the sound pressure level is going to be for a certain input voltage signal depending on its frequency and amplitude. This can be much easier understood looking at figure 7.1, where the sound pressure level of each loudspeaker is displayed for the audio frequency spectrum in a bode diagram, from 20Hz up to 20kHz. The SPL is measured theoretically at 1m from the loudspeaker, for a 2.83V$_{rms}$ input.

Looking at figure 7.1 it can be seen how the frequency response and the efficiency of each loudspeaker can change several dBs in comparison to others.
This is why a ‘target SPL curve’ response has been implemented, and a filter has been designed for each loudspeaker in order to achieve this same ‘target SPL’ for all of them. This is explained in detail in the next section.

7.2 Target Sound Pressure Level & Filter design

To design the target sound pressure level curve, it has been taken into account that all the loudspeakers being analysed are 6.5” woofer drivers. The typical frequency range that this kind of loudspeaker reproduces is below 2kHz.

The target SPL curve has been designed as two filters in series: a 4th order low-pass filter with a cut-off frequency of 1.5kHz and a 2nd order high-pass filter with a cut-off at 60Hz. A gain of 90dB has also been applied, because this is a typical SPL value for loudspeakers at 1m distance with 2.83Vrms signal. Figure 7.2 shows the target SPL frequency response curve.

For each of the loudspeaker analysed before, one filter is designed: placing this filter before the loudspeaker would make it have the same SPL response over frequency as the target. Depending on the loudspeaker, this filter will have a different curve, which can either apply or remove some gain for a certain frequency. In figure 7.3, the filter designed for each loudspeaker can be appreciated. It can be seen how in most of the cases the gain applied is around 0, which means that the SPL of that loudspeaker and the target SPL were quite similar before applying the filter.
Figure 7.2: *Bode* diagram of the target SPL transfer function

Figure 7.3: *Bode* diagram of the filter designed for each loudspeaker

### 7.3 Simulation & Power Window Sweep

In order to get an interesting comparisons on how each loudspeaker behaves and what efficiency it has, a simulation has been run using *Simulink*. The same filtered song that was used in other simulations in chapters 4 and 5 has
been amplified and cut to 10 seconds. This audio piece has been simulated in each loudspeaker using their linear model. To get all the loudspeakers to have the same theoretical sound pressure level, the filter designed in the section before has been applied to the audio signal before sending it to the loudspeaker. Figure 7.4 shows the Simulink file, where the filter is placed just before the loudspeaker. Each loudspeaker has its own numerator and denominator coefficients for the filter, calculated as explained in the previous section.

Figure 7.4: Simulink model used, where the linear model is combined with the filter of each loudspeaker

As it can be seen on figure 7.4, the current is extracted from each simulation. This current is then multiplied with the input voltage to the loudspeaker in order to compare the apparent power of each loudspeaker: figure 7.5 shows a short period of the power curve of all the loudspeakers. It can be appreciated how the power consumption of each one of them has very different values, even if the sound pressure level will be the same thanks to the filter.

In figure 7.5 it can be difficult to distinguish the different ranges of power consumption for each loudspeaker. This is why the power window sweep algorithm explained in Chapter 5 has been also applied to each one of these curves of power. In this case, only 5 seconds of the audio power curve have been analysed in the algorithm, and the number of different window sizes taken is 500, in order for the simulation not to take too long (the algorithm needs to be applied once for each loudspeaker, with a total of 128 times).

Figure 7.6 shows the result of the algorithm applied to all the loudspeakers. It can be seen how from $10^4$ samples window and on, the maximum rms power remains constant. The figure shows how there is a quite wide range of power consumption for all the loudspeakers: In the one with the highest
Figure 7.5: Power consumption of each loudspeaker for the same SPL. Only a short period of the audio curve is shown.

consumption, the peak power (that corresponds to the maximum $rms$ for a window size of one sample) goes over 12 VAR, while in others it is around only 2 VAR. To compare the results with higher precessions, other functions of *Matlab* have been used in the next section.
Figure 7.6: Result of power window sweep algorithm applied to 128 loudspeakers

7.4 Comparison & Conclusions

Although it might not be easy to compare the power consumption of all the 128 loudspeakers from the figures in the previous section, some important conclusions can be extracted. For this, the statistical function histfit of Matlab has been used. This function has firstly been used to compare all the peak apparent power values of the loudspeakers, which are equal to the maximum rms power values with a window size of only one sample. Figure 7.7 shows how the mean value of the peak power consumption is around 6 VAR, but the range is very wide: from the ones with the highest efficiency that have around 2 VAR peak power, to the less efficient ones, with a peak power around 12 VAR.

As the window size is increased, the maximum rms value goes down as expected, with the same form for all the loudspeakers, because all the simulations are done using the same audio file. Figure 7.8 shows the results of the maximum rms apparent power when the window size is 220500 samples long (5 seconds of the audio file), which is the highest window size taken. This value is very similar to the rms value of the whole audio file, and it has a mean value around 0.3 VAR.

Finally, figure 7.9 shows the ratio of the peak apparent power to the rms apparent power (crest factor) for each one of the 128 loudspeakers analysed.
Figure 7.7: *histfit* function used to display the distribution of the peak apparent power values.

Figure 7.8: *histfit* function used to display the distribution of the *rms* apparent power values.

It can be seen how, even if they have different power consumption ranges, the ratios remain constant: all of them are between 16.5 and 18.5. This confirms that the power consumption is proportional to the input audio file for all the loudspeakers, whatever the power consumption is.
Figure 7.9: Ratio of the peak apparent power to the $rms$ apparent power for the 128 loudspeakers
8 Conclusion

The main objective of this thesis was to make measurements on a real loudspeaker and compare the results obtained with those of simulations using the linear and non-linear models of the loudspeaker, to investigate about their power requirements. Different measurements have been done: first, the power measurement circuit was built and tested, and measurements taken on this circuit were compared to simulations, using the same audio file. After that, six different audio files, belonging to various musical styles, were used to perform the same comparisons. Finally, a database of over a hundred loudspeakers was used to create models for each one of them and compare their power requirements using simulations.

The first part of the project shows clearly how the linear and non-linear models of the loudspeaker behave in a very similar way to the real loudspeaker that is being modelled. The results achieved playing the same audio file are very satisfactory, as it can be seen how the power curves belonging to measurements on the real circuit, the linear model and the non-linear model follow each other very closely, with some small differences when peaks are reached. These small errors could also be due to the precision errors of the components used in the circuit. The power window sweep algorithm has also been applied to these measurements, and it can be clearly appreciated how audio files usually have very short peaks, which are much larger than the \( \text{rms} \) power of the whole audio piece (high \textit{crest factor}).

The comparison between the two simulation models shows how similar they are: the only difference between the curves of power obtained with the two models is that the non-linear model is slightly more precise when high power peaks are reached, due to the fact that it takes into account how some of the \textit{Thiele-Small} parameters change when the diaphragm of the driver has a high displacement. However, these are minor errors, and it has to be taken into account that the non-linear coefficients need to be included in this model, which are usually not provided in the loudspeaker datasheets (they need to be obtained experimentally using the \textit{Klippel} analyser or a similar tool). This is why the linear model is clearly enough to simulate the behaviour of the loudspeaker, and there is no need for a non-linear model, unless very high precision is needed.

The results obtained from the measurements using six songs from different musical styles show that the linear and non-linear models of the loudspeaker behave almost exactly as the real loudspeaker, it does not matter what audio file used is. Even if loudness normalisation has been applied to all the files to make them sound equally loud, it can be seen how the \textit{crest factor} is always very high, although it changes for the different musical styles.
Finally, the statistical analysis of loudspeaker power requirements shows that the range of power consumption for the same sound pressure level response can be very wide, even if the loudspeakers have similar characteristics, because they have different efficiencies. This means that, when designing a power supply for a certain kind of loudspeaker, it has to be taken into account that the power consumption may vary widely, so the worst case needs to be taken into account, which would be the loudspeaker with the highest power consumption, or the lowest efficiency.
9 Future Work

In this project an extensive investigation of loudspeaker power requirements has been performed and important conclusions have been extracted. However, this does not mean that it is finished here: there is a great amount of work that can be done following this project, from measurements of power requirements on different loudspeakers and different configurations, to the design of power supplies, amplifiers and loudspeakers that take into account all the results of this work.

First of all, all the measurements and simulations in this project have been done for woofer loudspeakers. The linear and non-linear models are only valid for the low-frequency range. However, it would be very interesting to do a similar investigation for tweeter loudspeaker drivers, using models that are valid for the high frequency range. This way, the power requirements for the entire audio spectrum would be known.

The measurements have been done using the free air configuration, where the loudspeaker does not have any enclosure. Although this gives important information, the power requirements might change when the loudspeaker is inside an enclosure, such as in the closed box or ported box configurations. In these cases it is important to take into account the acoustical domain of the loudspeaker when creating its model. It would be interesting to investigate and compare the power requirements of loudspeakers in these configurations, and check that the models are also valid for them.

The final goal after so much investigation would be to design loudspeaker systems with power supplies, amplifiers and loudspeaker drivers that take into account the results obtained. Right now, loudspeaker system manufacturers test their equipment using test-signals such as sinusoidal waveforms, so they design for constant power consumption. But these signals are very different to real audio signals, as in audio, typically, power requirements are very different over time, because the crest factor is very high. This means that loudspeaker systems that require a high maximum power but a low continuous power should be designed, which would reduce the cost.
References


Appendices

A Matlab Files

In this appendix, the Matlab files related to each section of this document are listed. These files are not necessary to understand this document, but the list is provided in case the reader wants to see the work done with more detail, or use some of the Matlab functions used during this project. The name of each file starts with the section it belongs to, inside each chapter.

• Chapter 2 - Loudspeaker Modelling
  - section3_Loudspeaker_Linear_Model.slx: linear model of the loudspeaker in Simulink
  - section5_Loudspeaker_Non_Linear_Model.slx: non-linear model of the loudspeaker in Simulink

• Chapter 3 - Loudspeaker Power Measurement Circuit
  - section5_Sin_Output_NIdevice.m: simple example of the functions of the NI USB-6356 data acquisition device.
  - section6_Sinusoidals_Loudspeaker_Load.m: sinusoidal waveforms on the loudspeaker to check the linearity of the audio amplifier.
  - section6_Sinusoidals_Resistive_Load.m: sinusoidal waveforms on a resistive load to check the linearity of the audio amplifier.
  - section6_Comparison.m: comparison of the linearity on the resistive load and the loudspeaker.

• Chapter 4 - Loudspeaker System Measurements, Simulations & Comparison
  - section1_Applying_filter_to_song.m: filter applied to the audio file to make it suitable for a woofer loudspeaker.
  - section2_Measurements_Circuit_2_Levels.m: measurements of current and voltage of the loudspeaker circuit playing the filtered audio file.
  - section3_Simulations_Comparison.m: same measurements on the linear and non-linear models using Simulink, and comparison of results.
• Chapter 5 - Power Window Sweep
  – section2_Applying_Window.m: power window sweep algorithm applied to the power curves obtained in the measurements of chapter 4.

• Chapter 6 - Loudness Normalisation & Measurements with Different Songs
  – section2_1_Loudness_Normallization_Songs.m: loudness normalisation of the six songs, using the loudness_itu function.
  – section2_2_Applying_Filter_Songs.m: filter applied to the loudness normalised songs.
  – section2_3_Circuit_Measurements.m: measurements taken on the loudspeaker circuit for the six songs.
  – section2_4_Simulations_And_Comparison.m: measurements with the linear and non-linear model for the six songs, and comparison of all the results.
  – section2_5_Applying_Window.m: power window sweep algorithm applied to the power consumption curves of all the songs.
  – section2_6_Normal_Distribution.m: statistical analysis of the peak and $rms$ power values for each song.

• Chapter 7 - Statistical Analysis of Loudspeaker Power Requirements
  – section1_Target_Filter.m: reading of all the Thiele-Small parameters from the loudspeakers database Excel file, design of target sound pressure level curve, and design for the filter for each loudspeaker.
  – section3_Simulations.m: simulations playing the filtered song on each one of the loudspeaker models of the database, with their filter.
  – section3_Applying_WIndow.m: power window sweep algorithm applied to the power consumption curves of all the loudspeakers.
  – section4_Normal_Distribution.m: statistical analysis of the power characteristics of each loudspeaker.
B Other Files

In this appendix, other files related to some chapters of this document can be found.

- **Chapter 3 - Loudspeaker Power Measurement Circuit**
  - *Datasheets folder*: datasheets of all the components used in the power measurement circuit
  - *Whole Circuit - LTSpice.pdf*: schematic of the power measurement circuit on LTSpice
  - *LoudspeakerLTSPICE.pdf*: complex model of the loudspeaker on LTSpice

- **Chapter 7 - Statistical Analysis of Loudspeaker Power Requirements**
  - *6.5 Woofer Data.xlsx*: database with the Thiele-Small parameters of the 128 loudspeakers analysed (the ones used are on sheet 3)