FCU ePaper

Feng Chia University
Outstanding Academic Paper by Students

Title：Digital Signal Processing: Audio Applications

Author(s): Abelardo Alvarez Moncada

Class: 1st year of Electroacoustics Master

Student ID: M0932108

Course: Digital Signal Processing

Instructor: Dr. Yu-Ting Tsai

Department: Master of Electroacoustics

Academic Year: Semester 2, 2021.

Digital Signal Processing: Audio Applications.

Abstract

Signals had let human beings analyze, model and deduce a lot of physical natural behaviors. Added to the latest technology advances, these signals found different ways to evolve through Digital Signal Processing. At this work, 5 different Audio Effects related topics are presented as Application of DSP. Noise Gate as a Dynamic Processing is explained, Linear and Circular Convolution explanation using a Measured Impulse Response, System Identification and Noise Cancelling as application of Adaptive LMS Filters is also explained and finally the Exotic Distortions with BPF filtering as a Diode Tube is explained.

Keywords： Dynamic Processing, Fourier Transform, Impulse Response, Adaptive Filters, Non-Linear Processing. Audio Effects.

Digital Signal Processing: Audio Applications.

Table of Content

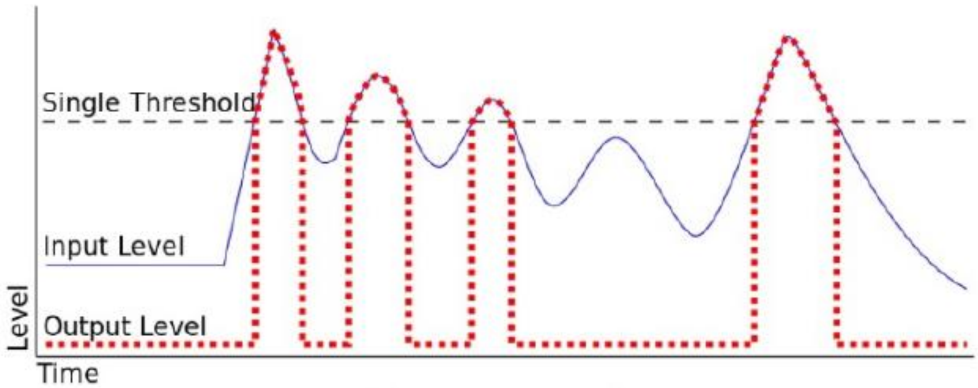
List of Figures

Chapter 1 Dynamic Processing (Noise Gate as Example)

Dynamic Processing in Audio word is well known as the capacity to modify or change the amplitude (also known as gain) according to some parameters as Threshold, Attack time and Release time as basic ones. This means every time a signal overpasses the threshold umbral, the signal will be affected according to the selected process, for this special case a Noise Gate was selected.

To understand this in a better way, an analogy is a good way to do it. We can imagine a noise gate literally as a vigilant stand in front of a door and the signal is the people that want to go through the door.

The vigilant work is to open or close the door according to a level requirement, if the signal level overpass the Threshold level, the door is open, other way the door will continue closed. Another more advanced consideration is the time that vigilant spent opening and closing the door, for this analogy the opening time is considered as the Attack time and the closing time after the signal passed through is the Release time.



[1]Figure 1. Noise Gate graphical representation

To clearly understand this, the Blue colored graph from Figure 1 (Réveillac, 2018) is the Input Signal and dotted red line is the opening, closing action taken by the Noise Gate Processor or *"The vigilant"*. We can start thinking that for this example the result is an interrupted sound and in fact it is. This make us wonder in which situations we will want to have this kind of behavior, and make us then ask, what is Noise Gate processing used for?

First of all, for two most used applications examples, we can think a Noise Gate as an automatic switch ON/OFF.

[1] Jean-Michel Réveillac, *Musical Sound Effects* (2018) p.182.

Digital Signal Processing: Audio Applications.

A very common application is in Live Sound, where a lot of open microphones are permanently opened. This generates unwanted sounds when they are not used. To illustrate this situation Figure 2 is a Stage Plot (musician's ubication on stage) of a common Latin band as example.

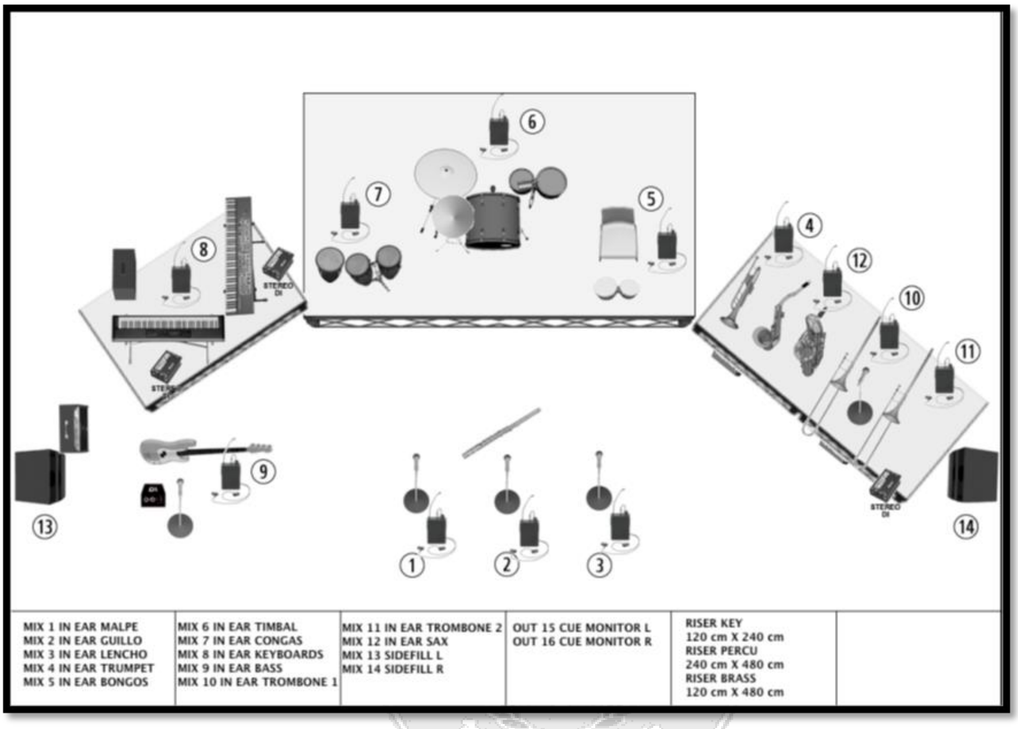


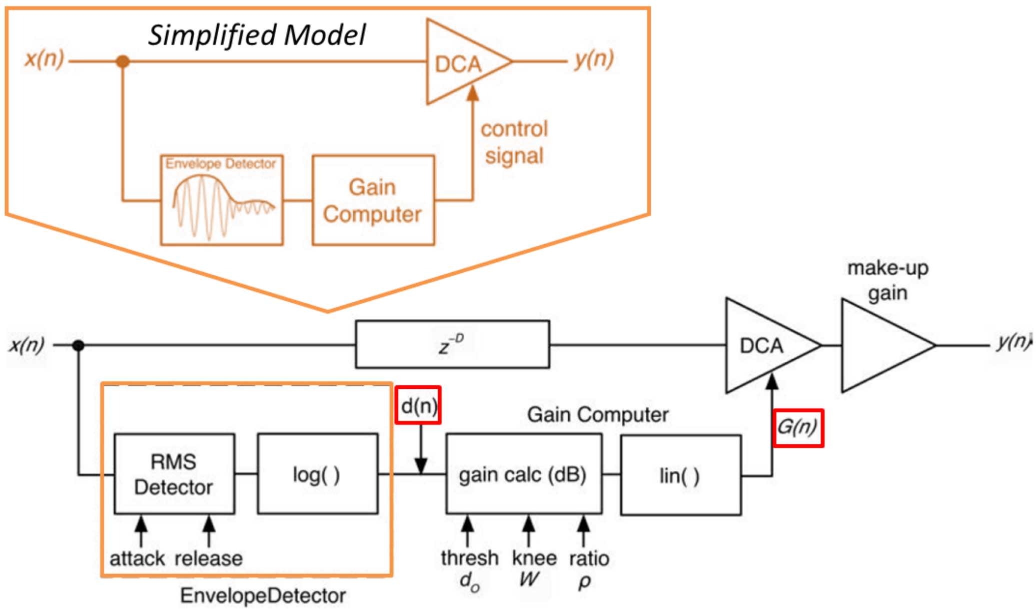
Figure 2. Latin Band Stage Plot.

This is a 24-microphone configuration, with 32 total instrument channels. All open at the same time. Without the use of a Noise Gate Processor, it's very difficult to handle the noise or unwanted sounds on stage, specially on intermediates, where nobody plays any instrument and an announcer or singer is just talking.

The second example is on a meeting room. Let's suppose a table, usually for 8 to 10 people is common. If all the microphones are kept open all the time, the other side of the meeting will continuously hear not only the conversation but also the noise or unwanted sources of sound.

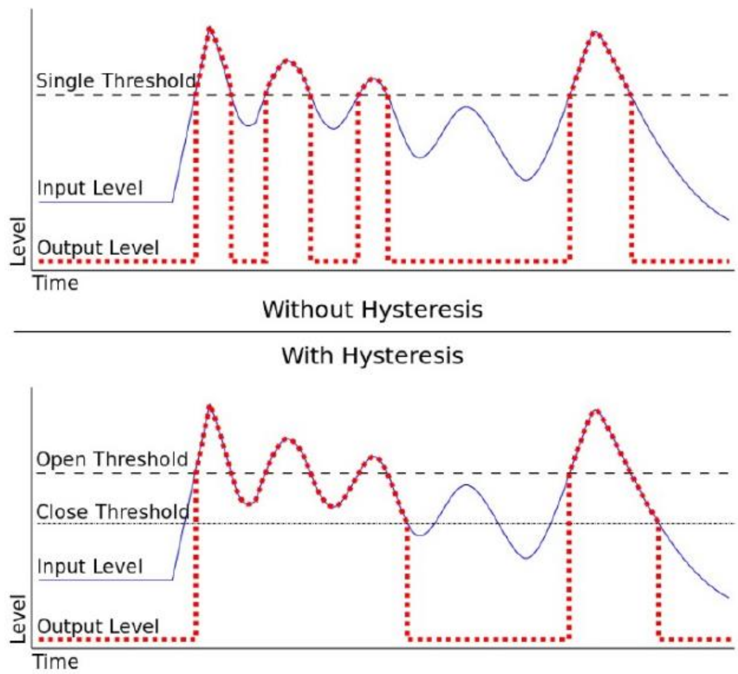
As we can imagine, in these two scenarios is very important to use Noise Gate processing on each of the microphones to avoid undesired sounds.
Going further on math, using *Matlab®* we can import a signal and process it according to what is described in *Designing Audio Effect Plugins in C++*, as shown on Figure 3 (Pirkle, 2019).

Digital Signal Processing: Audio Applications.



[2]Figure. 3 Envelope Detection

This is a quite simple method, which could be improved with a more complex methodology, for example as the included on *DAFX Digital Audio Effects*, that uses more complex time variables than described.
A graphical approach of Zoer book could be resumed on Figure 4 (Réveillac, 2018) graphs from *Musical Sound Effects* book.



[3]Figure 4. Noise Gate Hysteresis comparison graph

[2] Will C. Pirkle, *Designing Audio Effect Plugins in C++* (2019) p.357, p. 512.

[3] Jean-Michel Réveillac, *Musical Sound Effects* (2018) p.182.

The result graph of a real signal passing through a Noise Gate process is as shown on Figure 5:

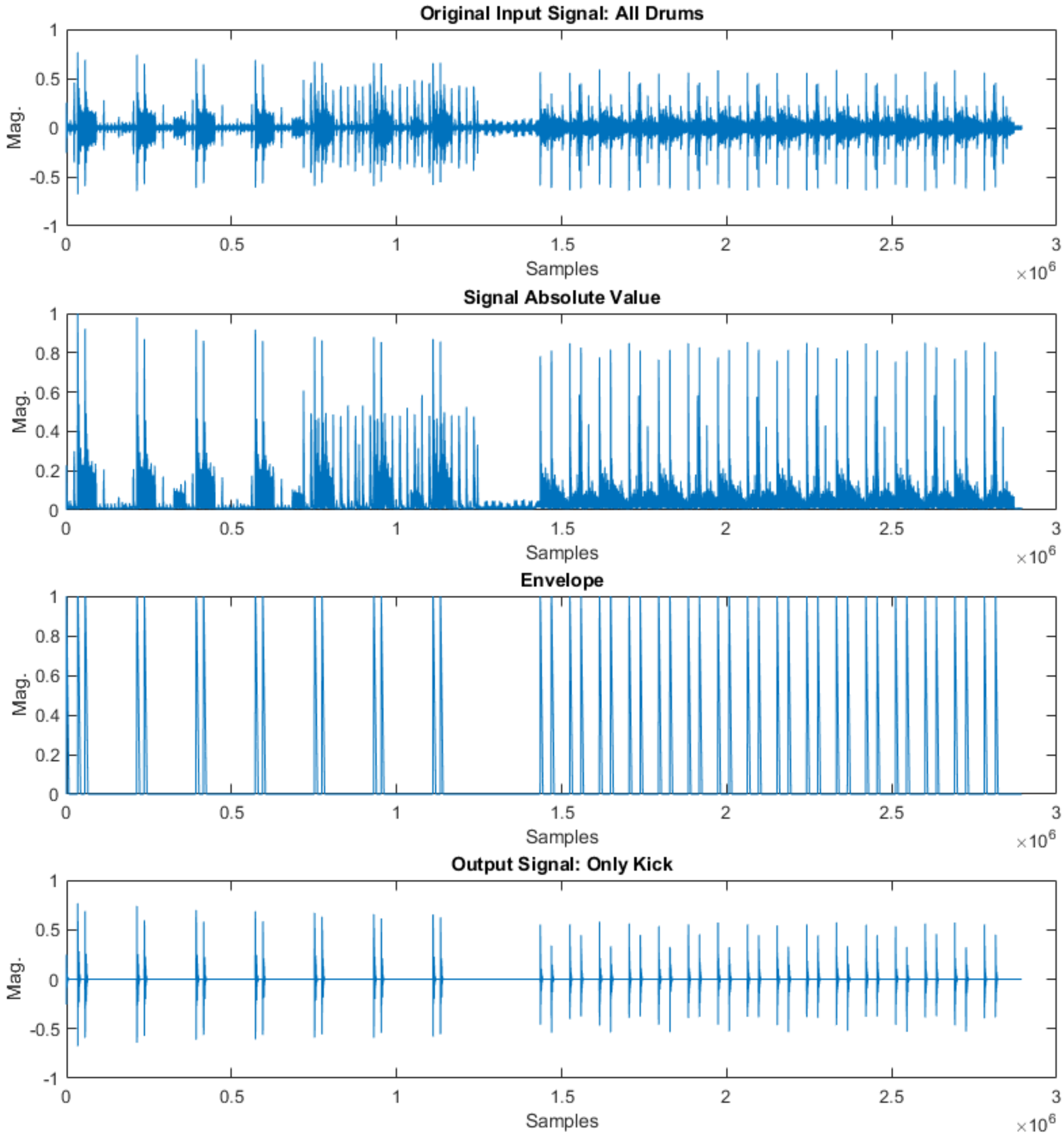


Figure 5. Noise Gate process graphs.

As we can see, not only the noise, but also not desired information of original signal was subtracted by the Noise Gate just to let the Kick Drum go through.

Chapter 2 Linear and Circular Convolution (Impulse Response Normalization and Convolution Processing)

Convolution is an expression referred to a mathematical process widely used in Signal Processing. Several books can explain mathematically the convolution process, Allan Openheim's book *Discrete Time Signal Processing* is a good example of this; but this time a music related analogy is going to be used.

In the world of music, musicians (specially guitarist) always wonder or dream with the sound of x or y guitarist. In other words, even if they have the same guitar as their favorite musician, the way that it sounds is not even close to what they wonder. This is because the musician is always looking for a sound with the color and tonal characteristic that makes him feel identified, confident and why not… happy.

Let's assume from the Figure 6 that guitar is our original input signal and is connected directly to an audio interface to be recorded, using a conventional line cable. At this moment, we have the original sound of the guitar because the cable only transmits the signal without changes.

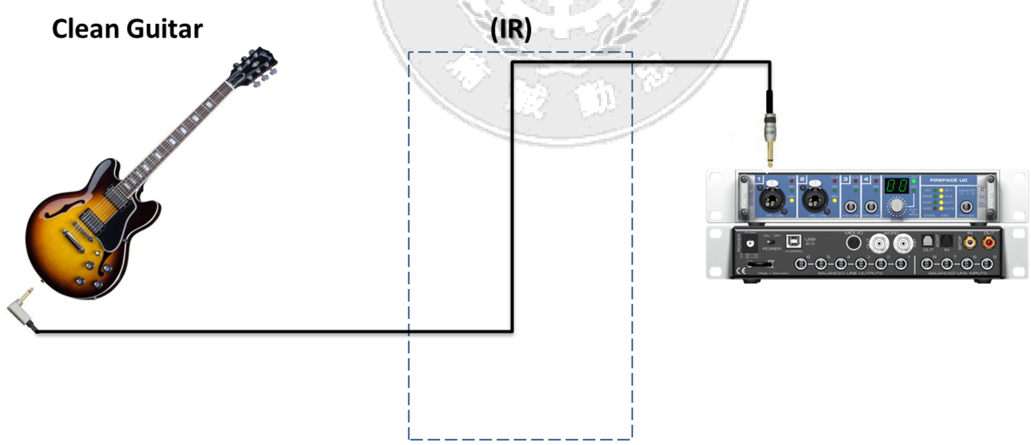
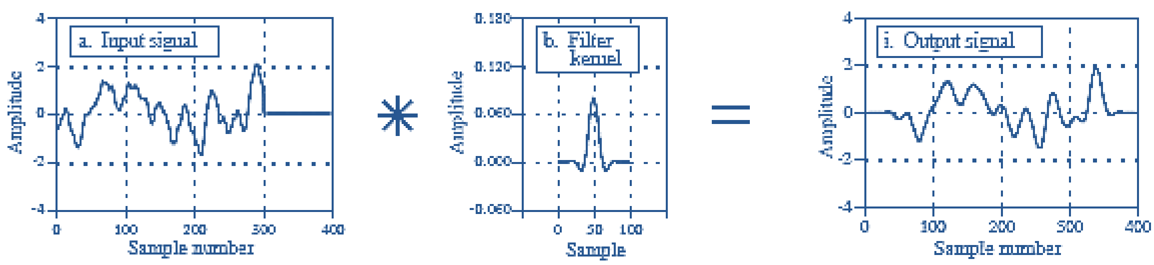


Figure 6. Unity convolution process analogy with a musical instrument.



[4]Figure 7. Convolution Process

[4] Steven W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*. (1999) p.313.

Then Convolution could be defined as the process to compare a signal, doing a point-by-point multiplication of the signal sample with the Impulse to generate a new signal in a specific time as shown on Figure 7 (Smith, 1999) If there is no variation, like the example, this Impulse is also known as Dirac or unitary impulse.

However, if we instead of using a single cable, we put a set of Head Amplifier, a Cabinet and a Microphone before going into the audio interface, the original signal of the guitar is going to be changed.

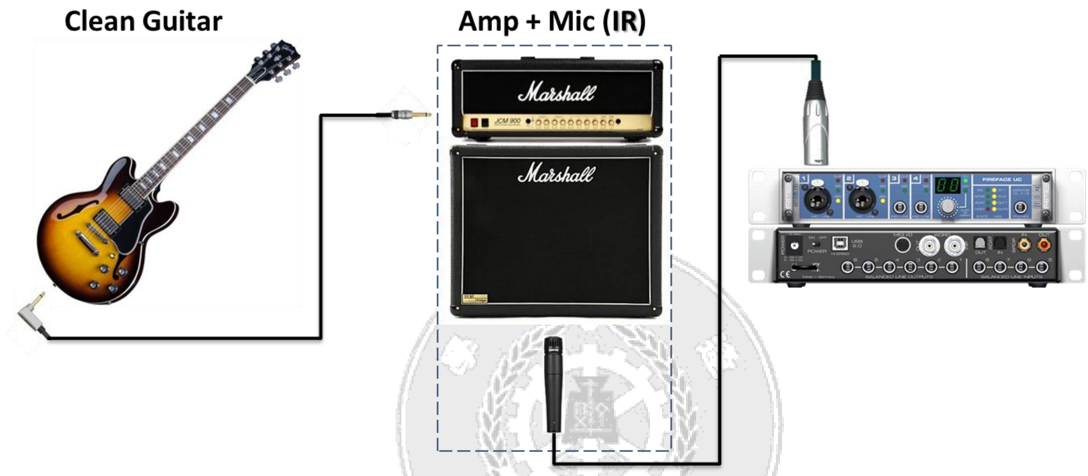


Figure 8. Convolution process analogy with a musical instrument.

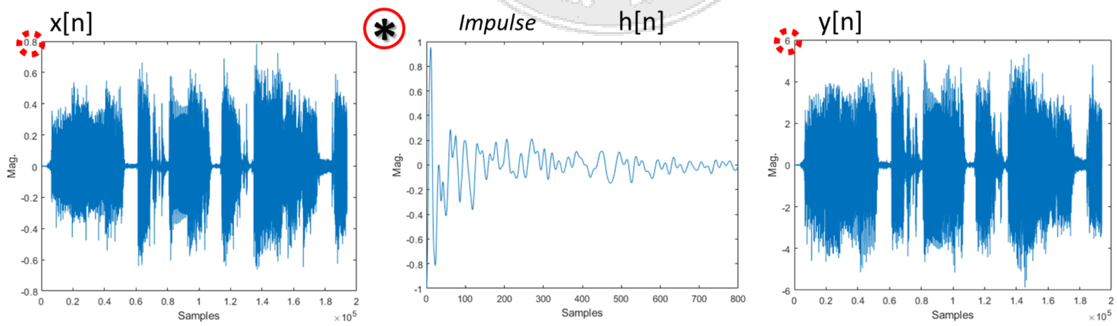


Figure 9. Convolution Process from a real case measured IR.

As we can see in Figures 8 and 9, the Impulse Response is not only affected for the Amplifier, but also by the Cabinet and the selected microphone. The Impulse Response could be analyzed in this case like human beings AND for the signal, who gives specific characteristics to an input signal.

For practical purposes, several measurements with different microphones, at different angles and different distances were performed to analyze the influence into the original signal.

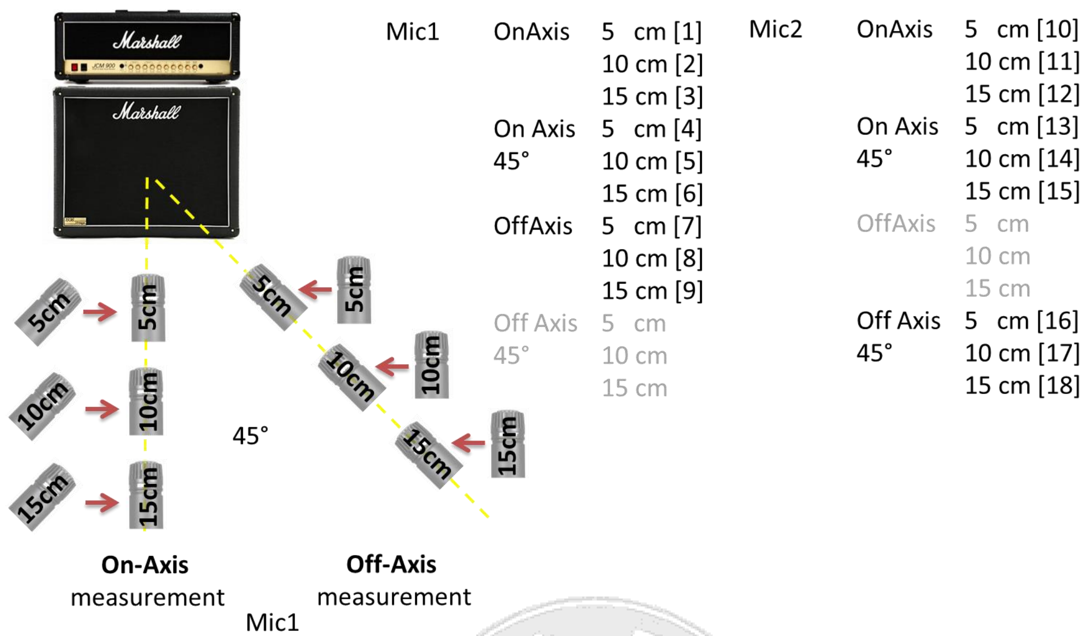


Figure 10. Impulse Response Measurements

Using the tool REW software (Room Equalization Wizard) that could be downloaded from https://www.roomeqwizard.com/ as a Free License Software, we can do different kind of measurements, for example Transfer Function for Frequency and Phase Response, Impulse Response for Acoustic Analysis and much more. For this case, we only need a two-channel audio interface with phantom power for condenser microphones and a microphone (any kind is welcome to try) with the required cables. Figure 10 shows the different points of measurement.
Once we have the Impulse Response, we can use Matlab first to cut the undesired portion of the IR and normalize the peak to 1, then perform convolution (linear and circular) with the original signal and hear the results.

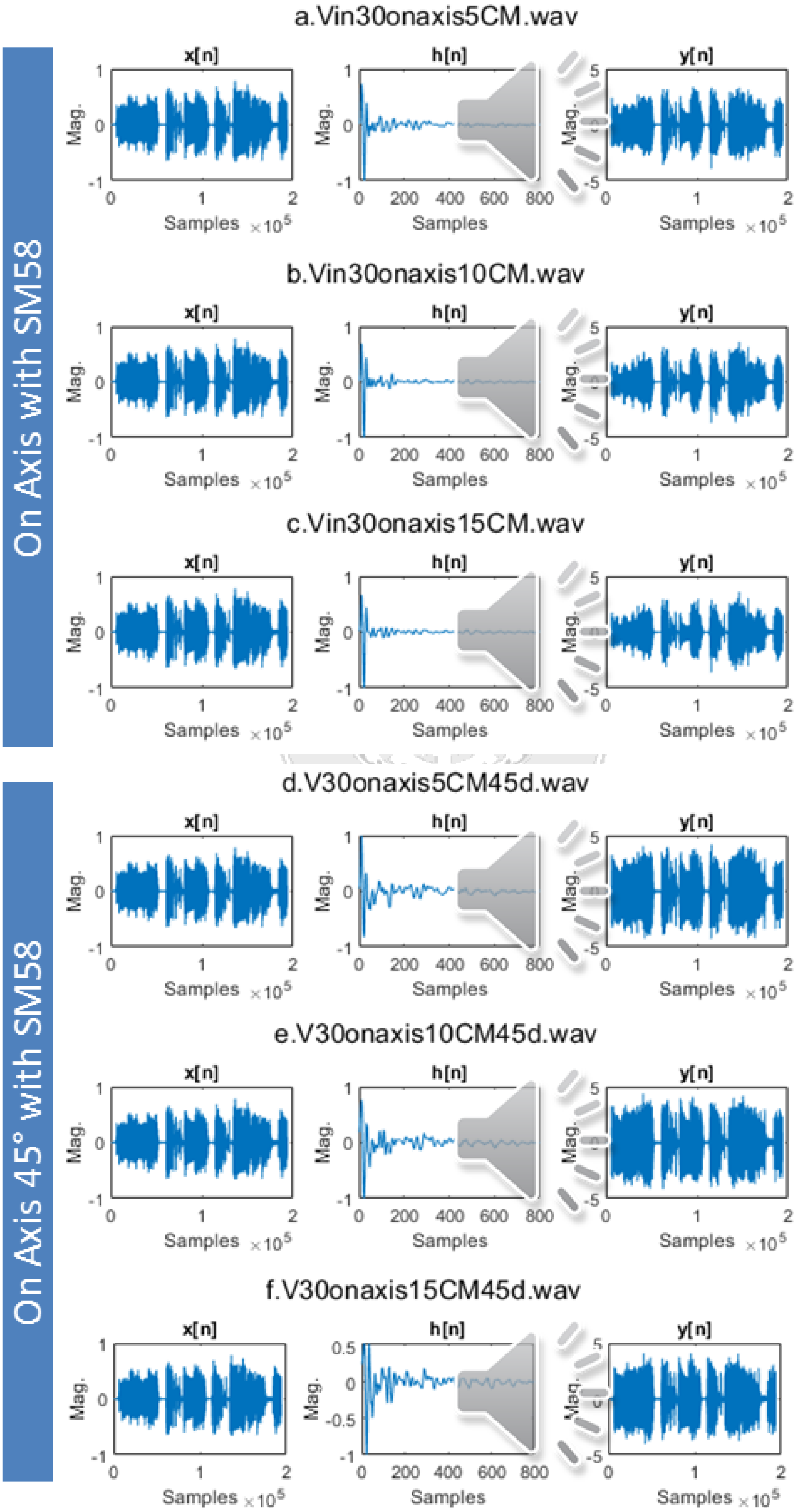


Figure 11a. IR Measurements Methods and Convolution

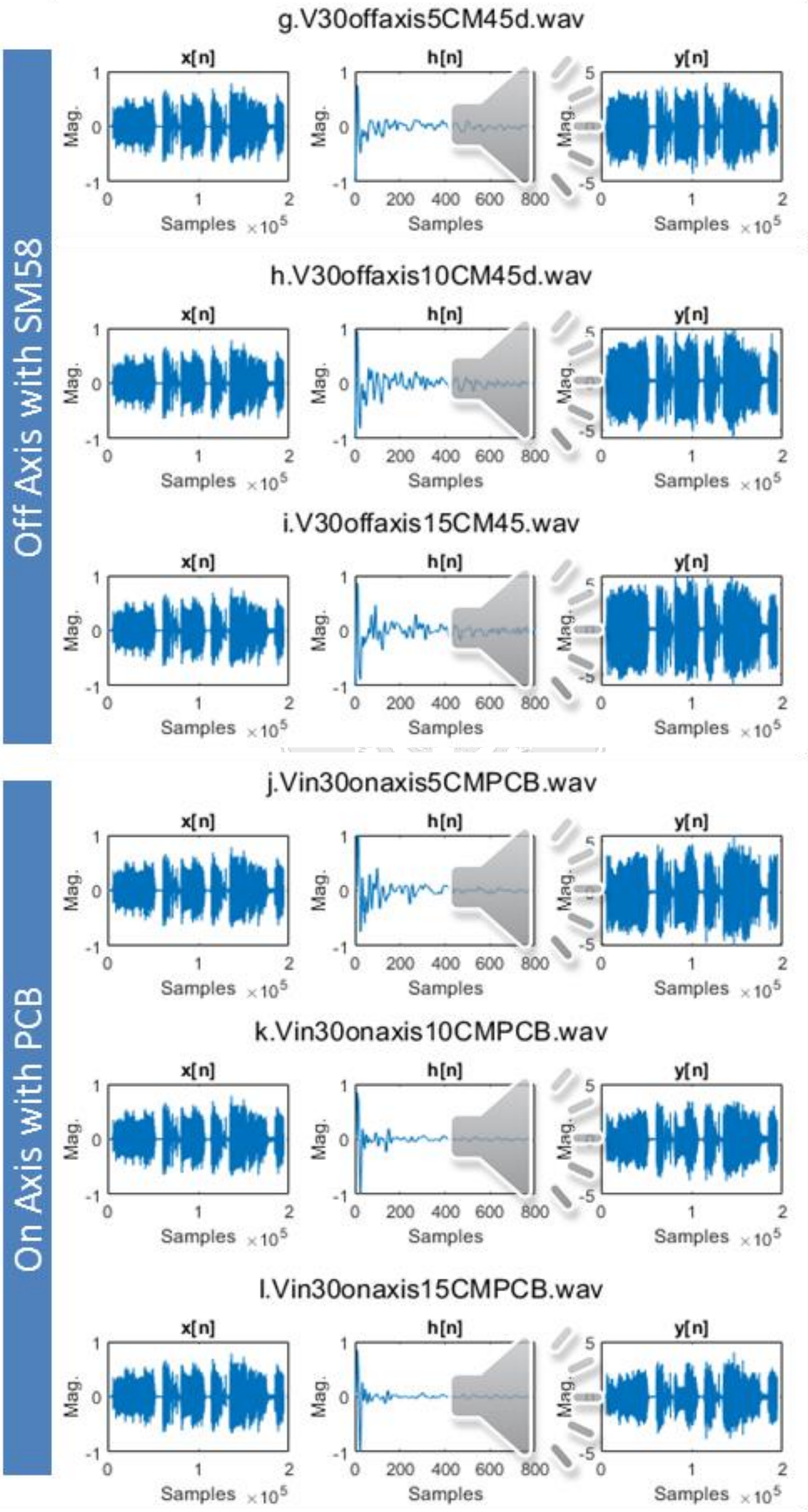


Figure 11b. IR Measurements Methods and Convolution

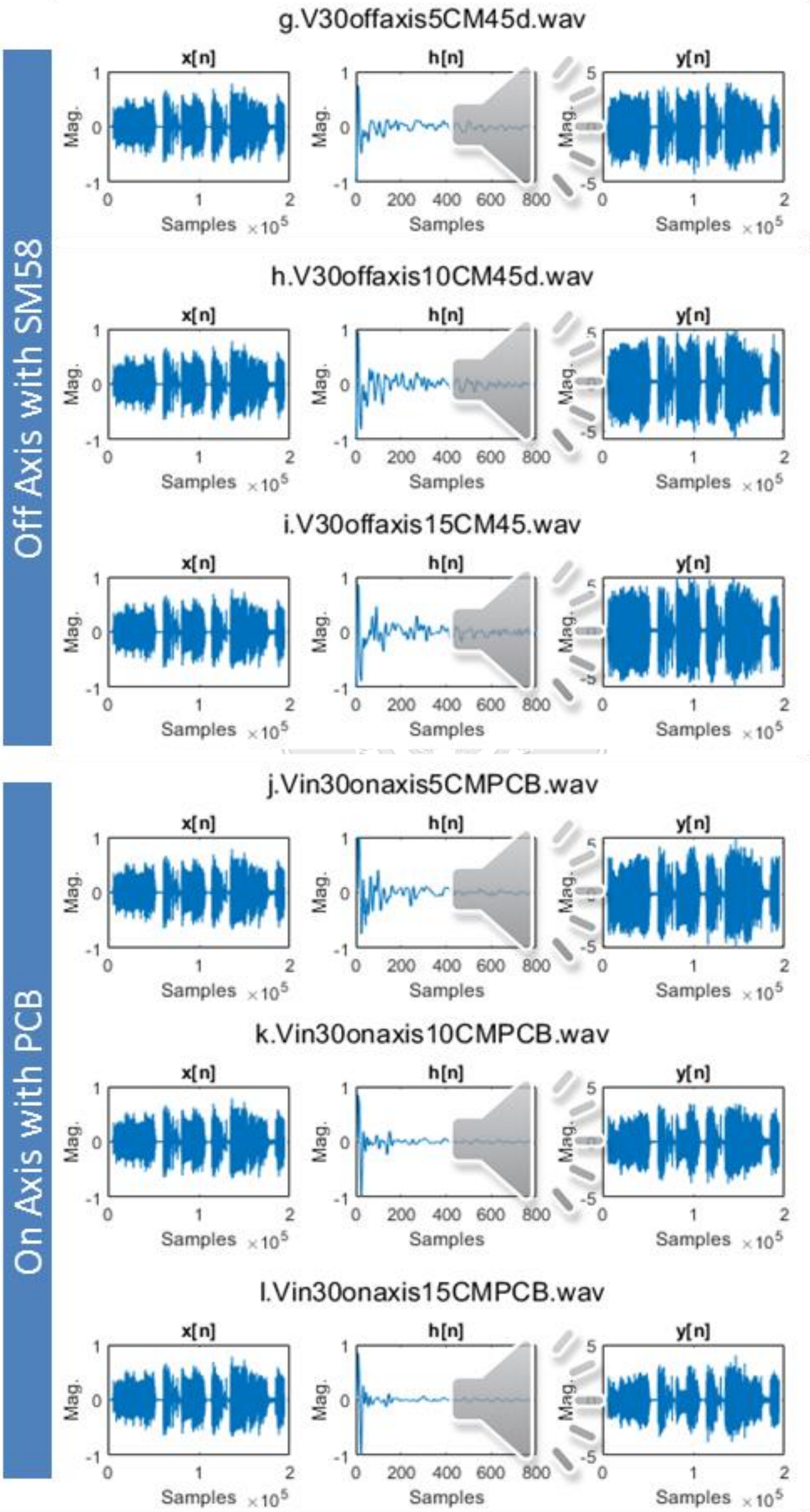
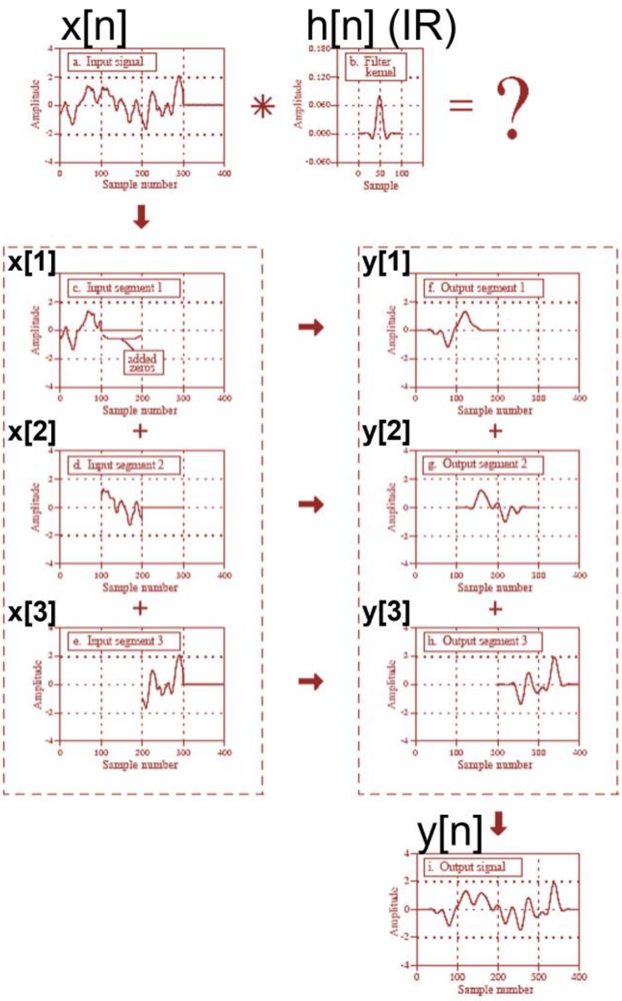


Figure 11c. IR Measurements Methods and Convolution

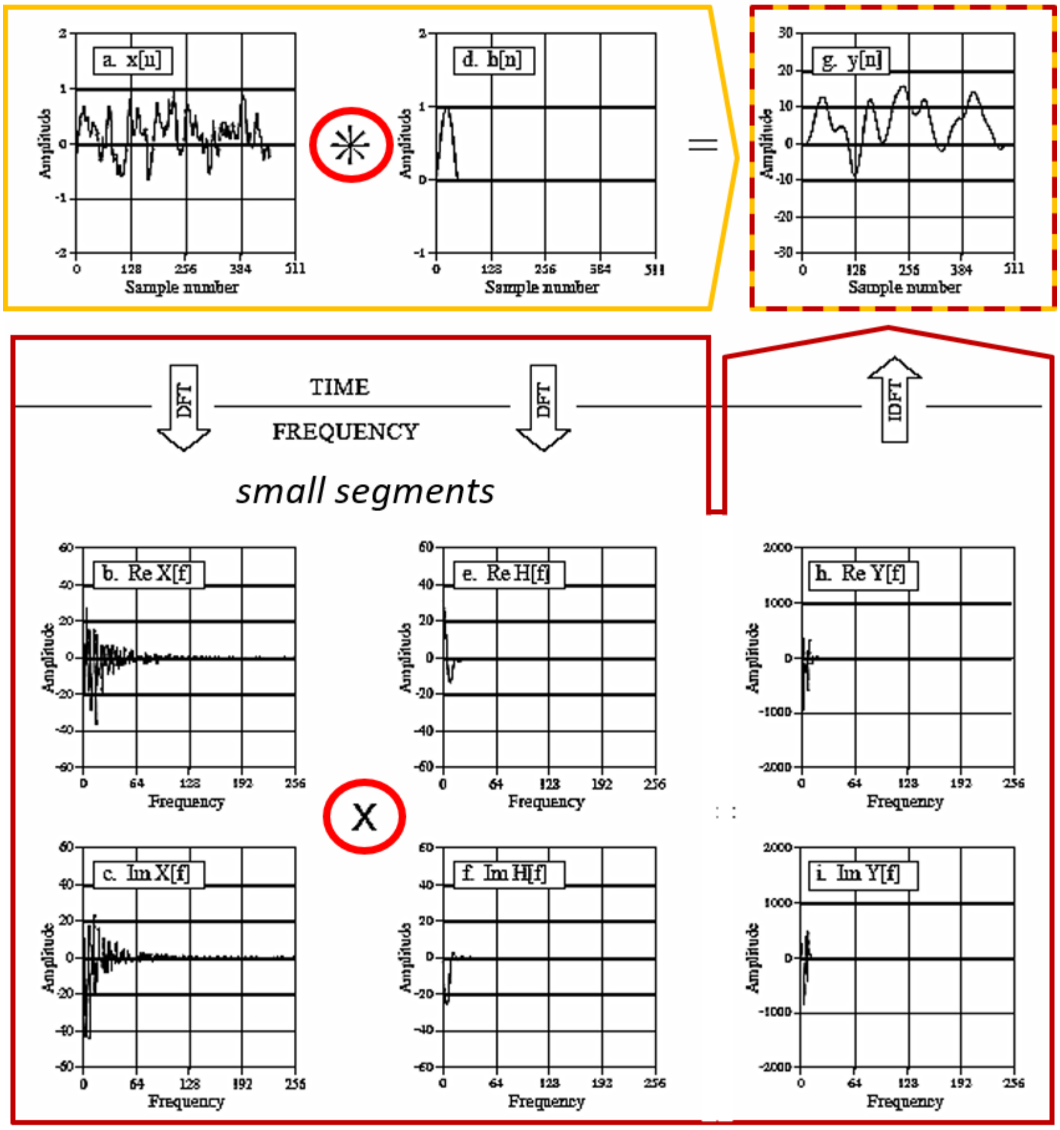
Even if results are the same for linear and circular convolution, the required time for each one is different. As we have seen, the linear convolution is a Time Domain performed operation, however the FFT (Fast Fourier Transform) is performed in the Frequency Domain and in small portions of the signal and finally use the Inverse FFT to get the time domain signal representation as illustrated on Figure 12 (Smith, 1999).



[5]Figure 12. FFT Convolution process

From this we can understand the FFT is a process that let us obtain the frequential content of the signal, also known as Frequency Domain; decomposing this into all the frequencies contained in a signal. Until now, we have used the FFT, but this is used for continuous signals only, while DFT is used for discrete signals. In Figure 13 (Smith, 1999), the yellow content is linear convolution process, while the red content is the Fourier Transform way.

[5] Steven W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing.* (1999) p.313.



[6]Figure 13. Convolution process comparison graph

[6] Steven W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing.* (1999) p.181.

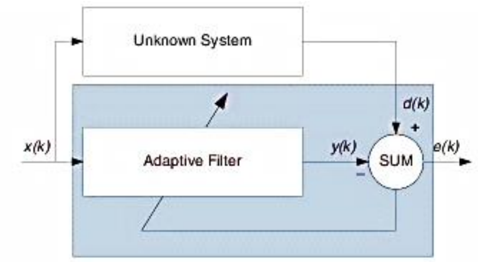
Digital Signal Processing: Audio Applications.

Chapter 3 Adaptive LMS (How to quit noise from a signal using System Identification and Noise Cancelling as selected methods)

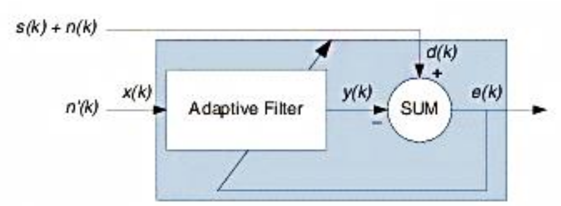
LMS is the abbreviation of Least Mean Square, a widely used algorithm in Adaptive Filters. In other words, adaptive means the filter is changing according to the desired result.

Additionally, according to the noisy given signal, two different methods were used to compare the application of each one.

The left one is the Adaptive LMS Filter for Unknown systems or also named as System Identification, as shown on figure 14a (Mathworks, s.a).



[7]Figure 14a.
System Identification



[8]Figure 14b.
Noise Cancellation

The right one showed on Figure 14b (Mathworks, s.a) is the Adaptive LMS Filter for Noise Cancellation, which requires a known noise source signal to be subtracted from the original one.

The easiest way to understand this is thinking the Adaptive LMS as a continuation of the convolution, as follows:

Convolution

Clean Guitar * IR = Desired Sound [Need the IR]

Desired Sound - Clean Guitar = Unknow System (**Adaptive**) [Need the Desired Sound]

Wiener Filter

Just to remember, with the original sound of the guitar and the Impulse Response of the equipment convolved, the Desired Sound is found.

However, this time for the Adaptive LMS Filter we already have the

---

7 Overview of Adaptive Filters and Applications, https://www.mathworks.com/help/dsp/ug/overview-of-adaptive-filters-and-applications.html, https://www.mathworks.com/help/dsp/ug/adapt_sysid.png

8 https://www.mathworks.com/help/dsp/ug/adapt_cancelnoise.png

desired sound and the original sound of the guitar, so the main objective of this filter is to find the Response of the Equipment.
Two methods were used to understand the main differences between them. The first used case is the System Identification, which from a desired signal, using a reference signal, we can obtain the coefficients of the desired filter as illustrated on Figure 15. System Identification helps to **predict** based on a desired signal.
For this practice a clean guitar signal was used as Desired Signal and the same signal with noise was used as a Reference Signal. At this point is important to notice that both correspond to a clean guitar signal, only with modifications. In other words, the adaptive filter signal should be working with correlated or known signals.

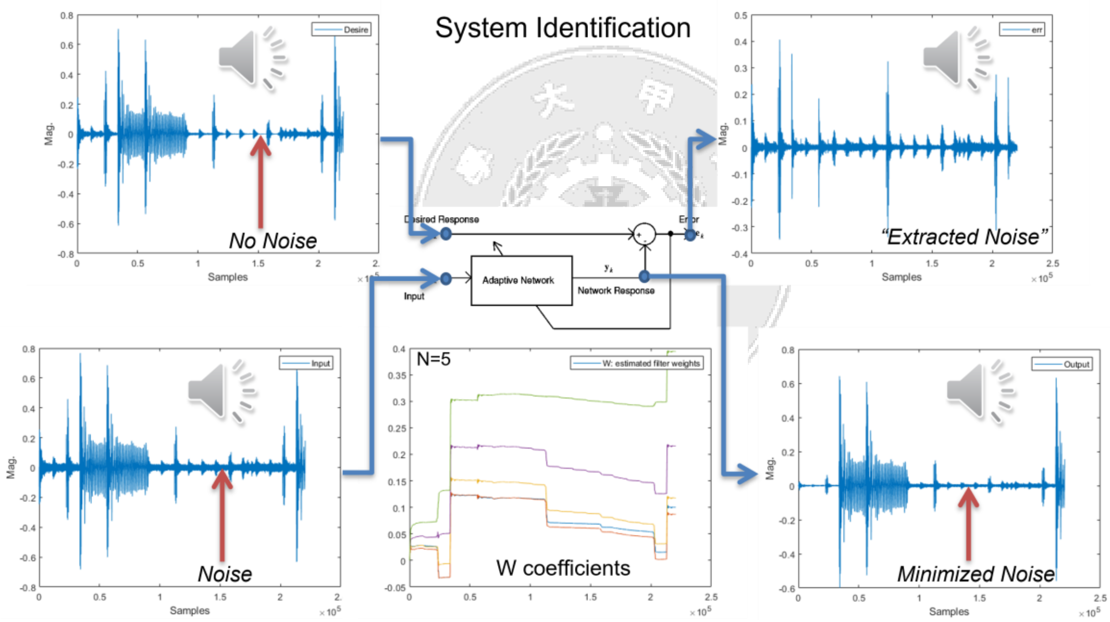


Figure 15. Adaptive LMS System Identification filter process graph

Even if this is process is quite responsive and accurate, the obtained signal is a good representation of the Desired signal, remains a small portion of the reference noisy signal, which could be considered as a Minimized Noise. Because of this reason, a different method was considered and compared.
The second used case corresponds to a different application because this method uses a signal affected by a known or previously measured noise.
Noise Cancelling helps to **substract** based on a "known" Signal.

Three different ways were tried.

1. Using a random generated noise.
2. Using a looped noisy section from reference signal.
3. Using white noise added to the desire signal to be computed.

After doing the three ways, to have results working with these filters, correlation between signals should be considered. In other words, the noise affecting the signal should be determined previously. In Figure 16 the correlated noise example is presented.

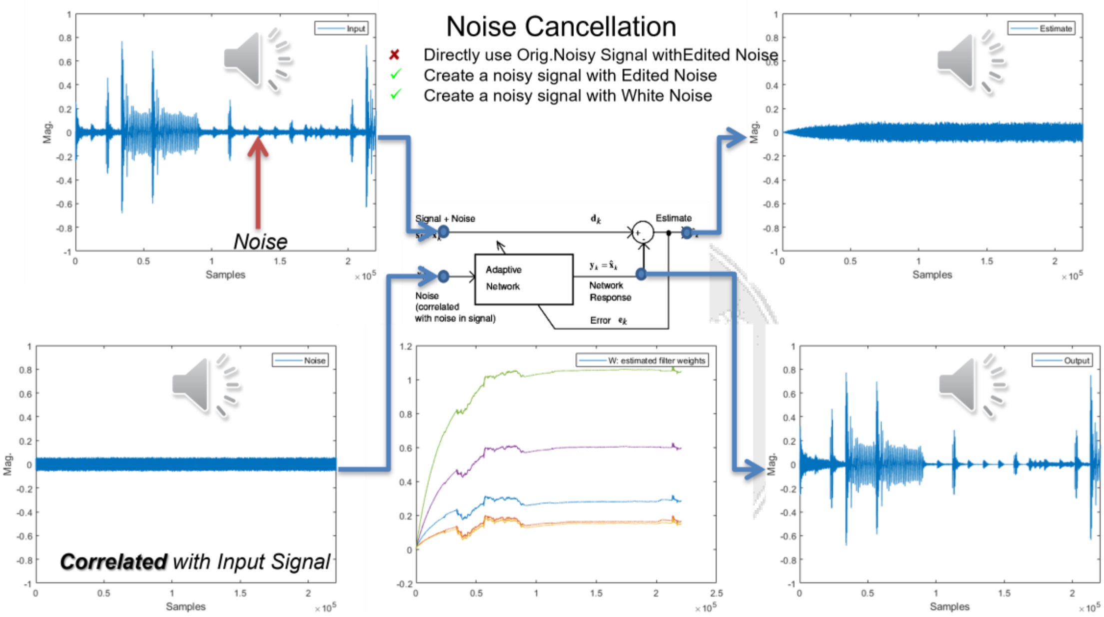
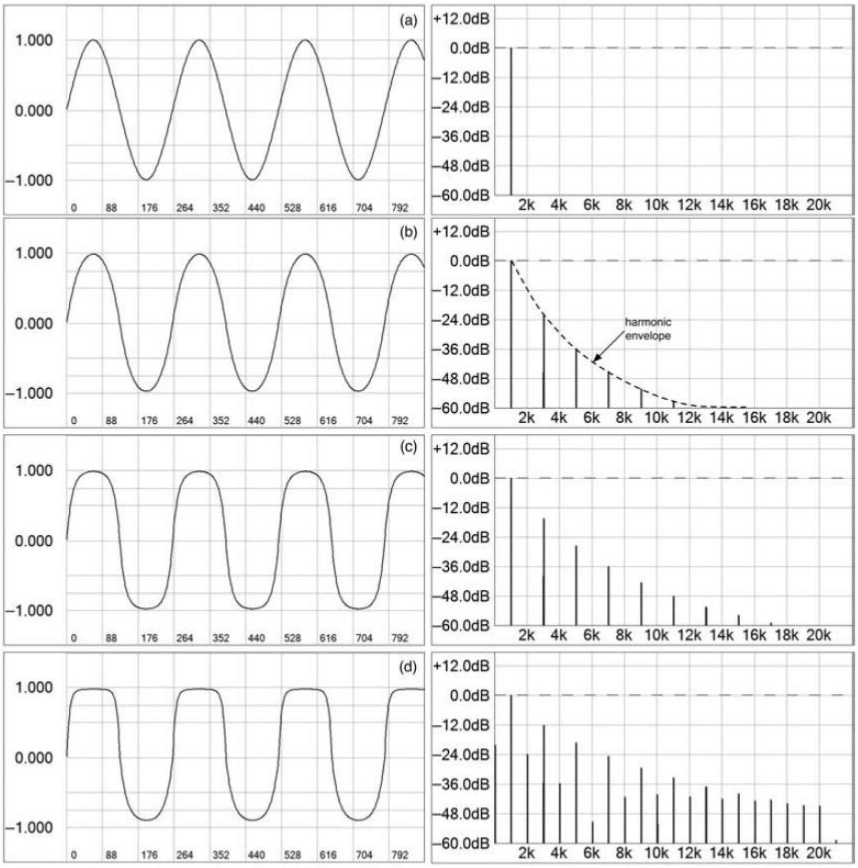


Figure 16. Adaptive LMS Noise Cancellation filter process graph

The easiest way to understand the term distortion is as illustrated on Figure 17, to start a pure signal without distortion in Time Domain is represented on the left side while his Frequency Domain representation corresponds to the 1KHz on the right side.

For the next graph a 1KHz tone is represented slightly different in Time Domain, but now on the Frequency Domain there's not only the 1KHz representation, there's also 3KHz, 5KHz, 7KHz, 9KHz and a little bit of 11KHz which means that this signal is a 1KHz signal with new *"children"* caused because of distortion. As we can see these *"children"* or better called harmonics are an odd version of the Fundamental 1KHz, but there are also even versions of Fundamental, or a combination of both odd-even harmonics as seen on the last graph of Figure 17 (Pirkle,1999).

Finally, the sum of these harmonics represents what is called as Distortion.



[9]Figure 17. Distortion representation

[9] Will C. Pirkle, *Designing Audio Effect Plugins in C++* (2019) p.537.

Digital Signal Processing: Audio Applications.

Several ways of distortion can be modeled using *Matlab®* and using the function described in *Designing Audio Effect Plugins in C++* book, page 548.

For this work, only the exotic distortion graphical representation could be observed on Figure 18, where different original waveforms in blue and distorted by the function on red are showed on the left side. Also, the same case from an original and changed signal containing a classic guitar content on the right side is displayed.

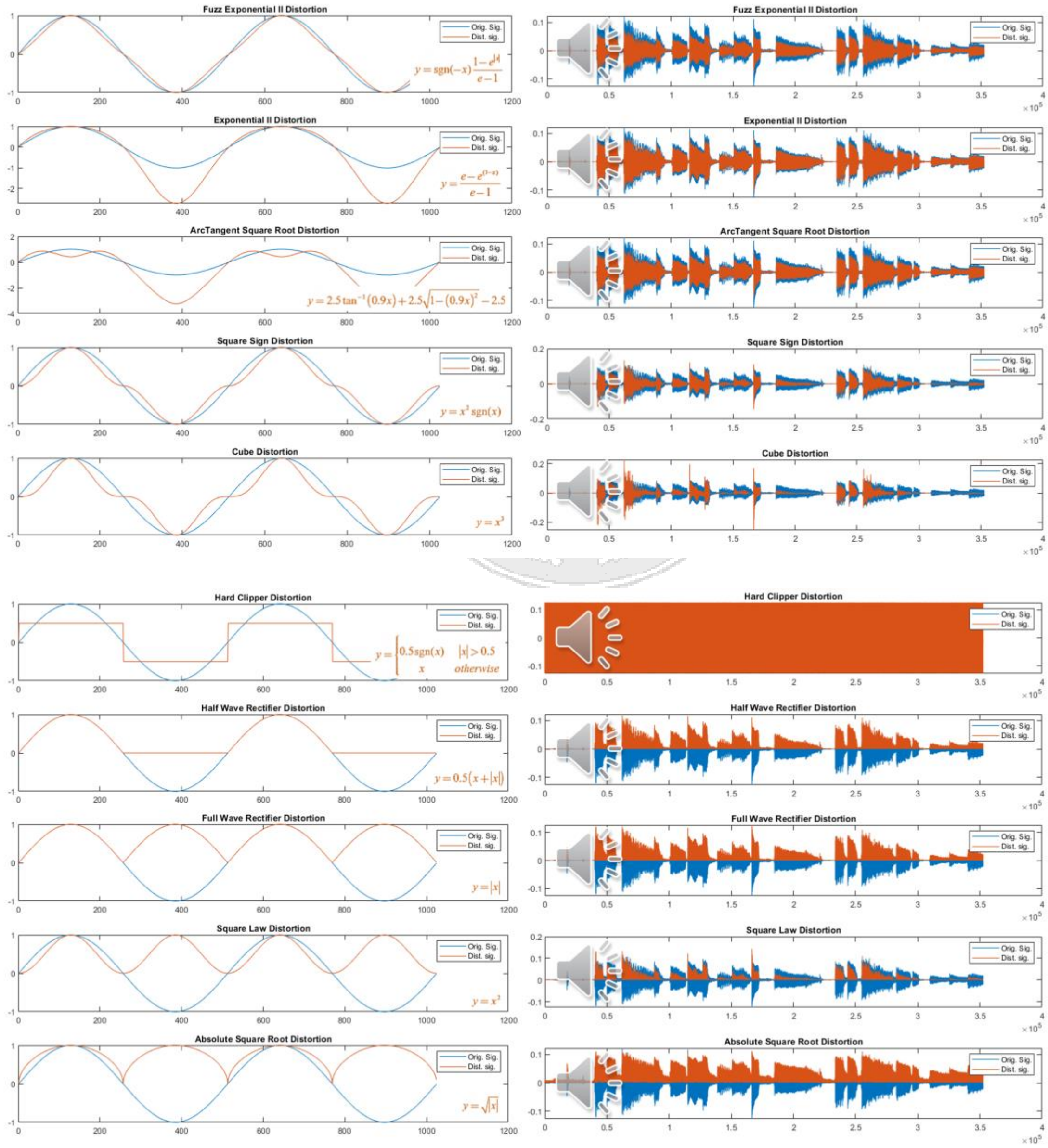
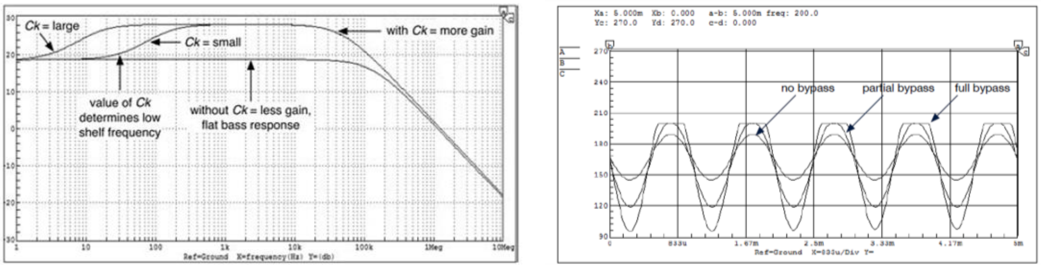


Figure 18. Exotic Distortion Graphs Representation

Digital Signal Processing: Audio Applications.

For this work the use of one exotic distortion with a band pass filter is modeled and presented to get an approximation of a triode class A amplifier shown on Figure 19 (Pirkle, 2019) for theoretical case and Figure 20 for practical case.



[10]Figure 19. Frequency Response and Time Domain output of a single Triode class A Amplifier.

ArcTangent [TANH] $y = \dfrac{\arctan(kx)}{\arctan(k)}$

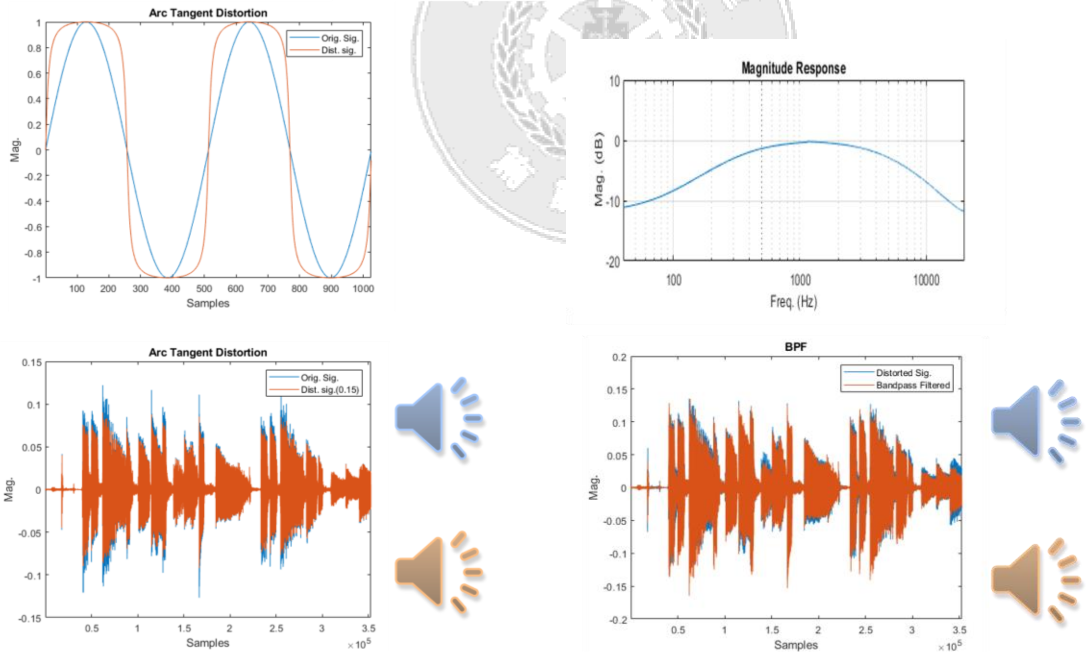


Figure 20. Exotic Distortion + Band Pass Filter

[10] Will C. Pirkle, *Designing Audio Effect Plugins in C++* (2019) p.539.

Chapter 5 Conclusions

- Five different topics were presented on a simple but effective way to understand how signal processing could be used and applied using music related signals.
- A Noise Gate processor was explained using a graph with detailed process and results on a selected signal.
- The hysteresis analysis on a Noise Gate makes a big difference when processing, because compared with the decay time, let the processor have suitable flexibility for the signal closing time. In other words, because of musical signals dynamics once it passed the single threshold, immediately close the gate, but it doesn't really mean all the is the required closing time, translated in a cut off output signal.
- IR and FFT Convolution were explained and compared as a mathematical process and proving the process with a selected signal.
- A measurement process was explained to obtain the Impulse Response of a Head Amplifier, Cabinet and Microphone system to do the convolution process with a guitar signal. Also, different IR measurements from different positions and angles were compared.
- Adaptive LMS Filter was described using two different application methods: system identification and noise cancelling for a desired and reference signal.
- Adaptive LMS system identification Filter and Adaptive LMS noise cancelling Filter were compared to analyze their application and accuracy with different parameters configuration.
- The Non-Linear Distortion was described using Exotic Distortion as an application.
- Using a Bandpass filter with the ArcTangent distortion a triode class A amplifier was modeled and compared with a musical signal.

References

- **Will C. Pirkle** (2019) *Designing Audio Effect Plugins in C++*. 2nd Edition. Focal Press.
- **Sophocles J. Orfanidis** (2010) *Introduction to Signal Processing.* Prentice-Hall.
- **Joshua D. Reiss; Andrew P. McPherson** (2015) *Audio Effects: Theory, Implementation and Application.* CRC Press.
- **Udo Zölzer** (2011) *DAFX: Digital Audio Effects.* John Wiley & Sons Ltd.
- **Steven W. Smith** (1999) *The Scientist and Engineer's Guide to Digital Signal Processing.* 2nd Edition. California Technical Publishing.
- **Jean-Michelle Réveillac** (2018) *Musical Sound Effects. Analog and Digital Sound Processing.* ISTE, Wiley.
- **Alan Oppenheim**. RES.6-008 Digital Signal Processing. Spring 2011. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu.