

Impulse Response Measurement Toolbox

By:

Edgar Berdahl

Julius Smith

Impulse Response Measurement Toolbox

By:

Edgar Berdahl

Julius Smith

Online:

< <http://cnx.org/content/col10519/1.5/> >

C O N N E X I O N S

Rice University, Houston, Texas

This selection and arrangement of content as a collection is copyrighted by Edgar Berdahl, Julius Smith. It is licensed under the Creative Commons Attribution 2.0 license (<http://creativecommons.org/licenses/by/2.0/>).

Collection structure revised: June 23, 2008

PDF generated: February 4, 2011

For copyright and attribution information for the modules contained in this collection, see p. 21.

Table of Contents

1 Impulse Response Measurement Using Golay Complementary Sequences	1
2 Swept Sine Impulse Response Measurement	5
3 Measuring Impulse Responses Using Golay Complementary Sequences and Swept Sinusoids	9
Bibliography	19
Index	20
Attributions	21

Chapter 1

Impulse Response Measurement Using Golay Complementary Sequences¹

1.1 Introduction

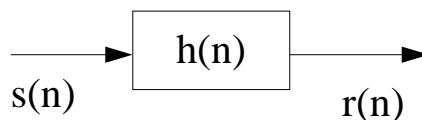


Figure 1.1: Linear system to be measured

Figure 1.1 depicts a linear system characterized by an impulse response $h(n)$, driven by an input signal $s(n)$, and producing the output signal $r(n)$. The **system identification** problem is to estimate $h(n)$ given known input/output signals $s(n)$ and $r(n)$. A practical method for identifying finite impulse responses uses **Golay complementary sequences** to excite the linear system as described below. Maximum length sequences² may be alternatively used for this type of measurement, as they are also noise signals consisting of 1's and -1's, but perfectly inverse filtering the measurement is more computationally intensive.

1.2 Golay Complementary Sequences

The length L bilevel sequences $a(n)$ and $b(n)$ are **Golay complementary sequences** if and only if the following condition holds, where $\delta(n)$ denotes the autocorrelation operator [1]:

$$a(n)a(n) + b(n)b(n) = 2L\delta(n) \quad (1.1)$$

and $\delta(n)$ is the Kronecker delta function. Many references in the audio signal processing literature refer to such sequences as Golay codes; however, to avoid confusion with Golay error-correcting codes used in digital

¹This content is available online at <http://cnx.org/content/m15947/1.10/>.

²http://en.wikipedia.org/wiki/Maximum_length_sequence

communication, we call the sequences **Golay complementary sequences**. Recall that ((1.1)) can also be written using $*$, the convolution operator:

$$a(-n) * a(n) + b(-n) * b(n) = 2L\delta(n) \quad (1.2)$$

Given that $a_L(n)$ and $b_L(n)$ are Golay, it turns out that $a_{2L}(n) = \begin{bmatrix} a_L(n) & b_L(n) \end{bmatrix}$ and $b_{2L}(n) = \begin{bmatrix} a_L(n) & -b_L(n) \end{bmatrix}$ are also Golay. This means that Golay complementary sequences can be constructed recursively given seed sequences such as $a_2(n) = \begin{bmatrix} 1 & 1 \end{bmatrix}$ and $b_2(n) = \begin{bmatrix} 1 & -1 \end{bmatrix}$. See the MATLAB³/Octave⁴ source code `generate_golay.m`⁵ for details. Notice also that the resulting bilevel sequences consist of only 1's and -1's. This means that the signal contains the maximum possible power level given that $|s(n)| \leq 1 \forall n$. This property is helpful for minimizing measurement noise.

Let $r_a(n) = a(n) * h(n)$ be the response due to the input $a(n)$, and let $r_b(n) = b(n) * h(n)$ be the response due to the input $b(n)$. Due to ((1.1)), the impulse response $h(n)$ may be determined as follows:

$$h(n) = \frac{1}{2L} (a(n)r_a(n) + b(n)r_b(n)) \quad (1.3)$$

See `golay_response.m`⁶ for more details.

1.3 Measurement Procedure Using Golay Complementary Sequences

1. Generate the Golay complementary sequences `golayA.wav`⁷ and `golayB.wav`⁸ using `generate_golay.m`⁹.
2. Open the pd patch `golay.pd`¹⁰, in pd.
3. Ensure that the patch is not in editing mode, and check the “compute audio” box in the main pd window.
4. Adjust the “Output Volume” so that when you click on “Record Response to Golay A”, the system under test is behaving linearly (i.e. not clipping), but so that the input signal to the sound interface is not too noisy.
5. If there is an input volume on the sound interface, adjust it so that the levels approximately match those shown in Figure 1.2 when you click on “Record Response to Golay A” and “Record Response to Golay B.” If the sound interface has no input volume, then you will need to adjust the “Output Volume” accordingly.

³<http://www.mathworks.com/>

⁴<http://www.octave.org/>

⁵http://ccrma.stanford.edu/real/simple/imp_meas/generate_golay.m

⁶http://ccrma.stanford.edu/real/simple/imp_meas/golay_response.m

⁷http://ccrma.stanford.edu/real/simple/imp_meas/golayA.wav

⁸http://ccrma.stanford.edu/real/simple/imp_meas/golayB.wav

⁹http://ccrma.stanford.edu/real/simple/imp_meas/generate_golay.m

¹⁰http://ccrma.stanford.edu/real/simple/imp_meas/golay.pd

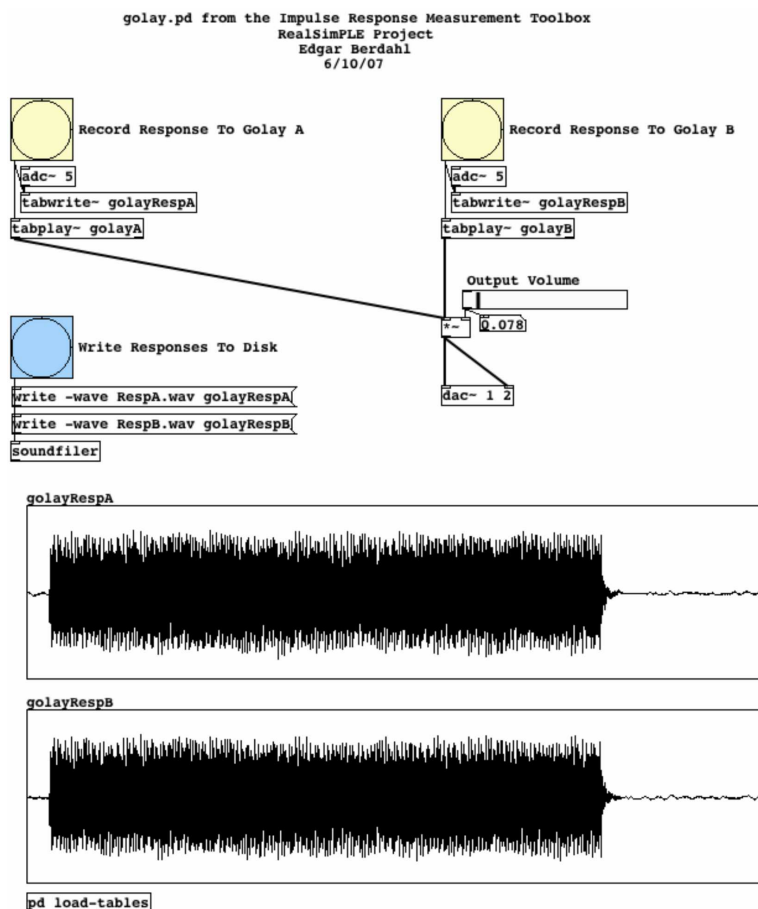


Figure 1.2: golay.pd after making a measurement with an appropriate input level

6. Once you are satisfied with the results, click the “Write Responses to Disk” button.
7. pd will write the files RespA.wav and RespB.wav to disk. Rename these files so that the names match the measurement you just made. For instance, you might rename them to hpfRespA.wav¹¹ and hpfRespB.wav¹² if they corresponded to the measurement of a high-pass filter.
8. Run `golay_response('hpf')`¹³ in MATLAB¹⁴ or Octave¹⁵ to analyze the measured response. Plots will be created, and the file hpfImpResp.wav¹⁶ will be written to disk.

¹¹http://ccrma.stanford.edu/realsimple/imp_meas/hpfRespA.wav

¹²http://ccrma.stanford.edu/realsimple/imp_meas/hpfRespB.wav

¹³http://ccrma.stanford.edu/realsimple/imp_meas/golay_response.m

¹⁴<http://www.mathworks.com/>

¹⁵<http://www.octave.org/>

¹⁶http://ccrma.stanford.edu/realsimple/imp_meas/hpfImpResp.wav

Chapter 2

Swept Sine Impulse Response Measurement¹

2.1 Introduction

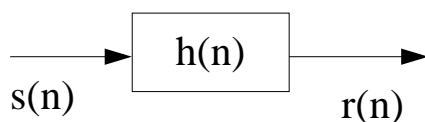


Figure 2.1: Linear system to be measured

Figure 2.1 depicts a linear system characterized by an impulse response $h(n)$, driven by an input signal $s(n)$, and producing the output signal $r(n)$. The **system identification** problem is to estimate $h(n)$ given known input/output signals $s(n)$ and $r(n)$. A practical method for identifying finite impulse responses is the **swept-sine** measurement technique, described below.

2.2 Sine Sweep Measurement Theory

In some cases, it is desirable to relax the power-maximizing constraint $|s(n)| = 1 \forall n$ in favor of obtaining some other desirable measurement system properties. For example, we may care more about the accuracy of the measurement at lower frequencies compared to higher frequencies, so we would like the excitation signal $s(n)$ to contain more energy at lower frequencies [2]. We might also be measuring a mechanical or acoustical system in which the motor controlled by $s(n)$ behaves weakly nonlinearly. If the nonlinearity is memoryless and is **NOT** preceded by any filtering, then the system to be measured matches the Hammerstein model shown in Figure 2.2. The goal is to measure $h(n)$, independently of the motor nonlinearity $f(s)$. Performing the measurement is complicated by the fact that superposition no longer holds.

¹This content is available online at <<http://cnx.org/content/m15945/1.4/>>.

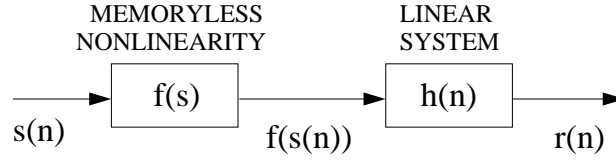


Figure 2.2: Hammerstein Model

Mathematically, the Hammerstein system behaves as follows:

$$r(n) = (f(s) * h)(n) \quad (2.1)$$

It turns out that we can obtain both of these desirable measurement system properties by using a new excitation signal $s(n)$. This signal is a sine wave with a frequency that is exponentially increased from ω_1 to ω_2 over T seconds [3]:

$$s(n) = \sin \left[K \left(e^{-n/Lf_s} - 1 \right) \right] \quad (2.2)$$

where $K = \frac{\omega_1 T}{\ln \frac{\omega_2}{\omega_1}}$ and $L = \frac{T}{\ln \frac{\omega_2}{\omega_1}}$. The MATLAB² /Octave³ code `generate_sinesweeps.m`⁴ generates the appropriate sine sweep.

Now we consider how to extract the linearized impulse response from a measurement. In essence, we need to inverse filter⁵ the measurement by the excitation signal.

To this end, we realize that a useful property of $s(n)$ is that the time delay Δt_N between any sample n_0 and a later point with instantaneous frequency N times larger than the instantaneous frequency at $s(n_0)$ is constant:

$$\Delta t_N = T \frac{\ln(N)}{\ln \frac{\omega_2}{\omega_1}} \quad (2.3)$$

This characteristic implies that after inverse-filtering the measured response, the signals due to the nonlinear terms in $f(s)$ are located at specific places in the final response signal. Consequently, the linear contribution to the response, which is proportional to $h(n)$ can be separated from the other nonlinear terms. We can thus measure a linear system even if it is being driven by a weakly nonlinear motor.

Because the frequency of $s(n)$ increases exponentially, the system is excited for longer periods of time at lower frequencies. This means that the inverse filter averages measurements at lower frequencies longer, so this measurement technique is better suited to especially low-pass noise sources.

2.3 Sine Sweep Measurement Procedure

1. Generate the sine sweeps⁶ using `generate_sinesweeps.m`⁷
2. Open the pd patch `sinesweeps.pd`⁸ in pd.
3. Ensure that the patch is not in editing mode, and check the “compute audio” box in the main pd window.

²<http://www.mathworks.com/>

³<http://www.octave.org/>

⁴http://ccrma.stanford.edu/real/simple/imp_meas/generate_sinesweeps.m

⁵http://ccrma.stanford.edu/~jos/pasp/Inverse_Filtering.html

⁶http://ccrma.stanford.edu/real/simple/imp_meas/sinesweeps.wav

⁷http://ccrma.stanford.edu/real/simple/imp_meas/generate_sinesweeps.m

⁸http://ccrma.stanford.edu/real/simple/imp_meas/sinesweeps.pd

4. Adjust the “Output Volume” so that when you click on “Record Response To The Sine Sweeps,” the system under test is behaving linearly (i.e. not clipping), but so that the input signal to the sound interface is not too noisy.
5. If there is an input volume on the sound interface, adjust it so that the levels approximately match those shown in Figure 2.3 when you click on “Record Response To The Sine Sweeps.” If the sound interface has no input volume, then you will need to adjust the “Output Volume” accordingly.

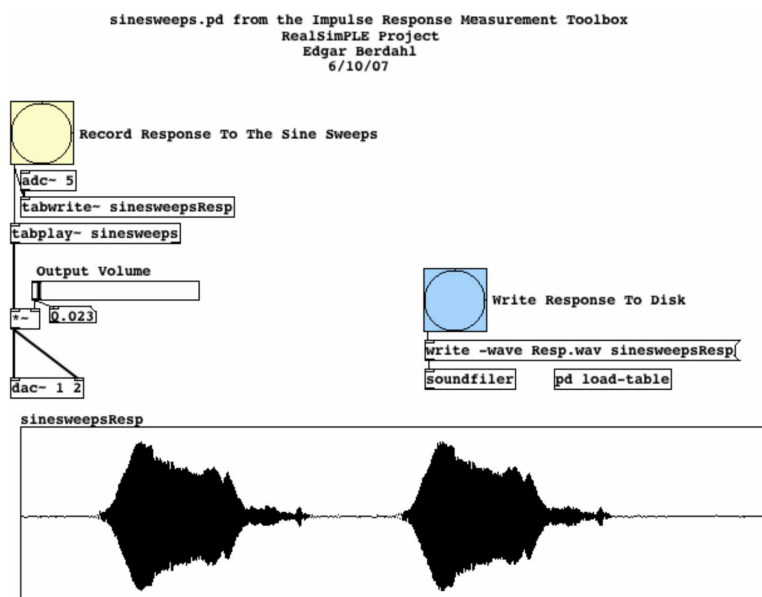


Figure 2.3: sinesweeps.pd after making a measurement with an appropriate input level

6. Once you are satisfied with the results, click the “Write Responses to Disk” button.
7. pd will write the file `Resp.wav` to disk. Rename this file so that the name matches the measurement you just made. For instance, you might rename it to `nonlinear2Resp.wav`⁹ if it corresponded to the second time you measured the transfer function of a weakly nonlinear system.
8. Run `sinesweeps_response('nonlinear2,100,0.4')`¹⁰ in MATLAB¹¹ or Octave¹² to analyze the measured response. This means that the inverse filter will be restricted to a dynamic range of 100 (40 dB), which helps avoid exaggerating problems beneath ω_1 and above ω_2 , where the excitation signal has little energy. 0.4 refers to the length in seconds of the linear impulse response term to be extracted. Plots will be generated, and the file `nonlinear2ImpResp.wav`¹³ will be written to disk.

⁹http://ccrma.stanford.edu/realsimple/imp_meas/nonlinear2Resp.wav

¹⁰http://ccrma.stanford.edu/realsimple/imp_meas/sinesweeps_response.m

¹¹<http://www.mathworks.com/>

¹²<http://www.octave.org/>

¹³http://ccrma.stanford.edu/realsimple/imp_meas/nonlinear2ImpResp.wav

Chapter 3

Measuring Impulse Responses Using Golay Complementary Sequences and Swept Sinusoids¹

3.1 Introduction

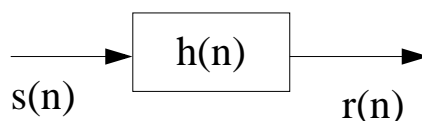


Figure 3.1: Linear system to be measured

Figure 3.1 depicts a linear system characterized by an impulse response $h(n)$, driven by an input signal $s(n)$, and producing the output signal $r(n)$. The **system identification** problem is to estimate $h(n)$ given known input/output signals $s(n)$ and $r(n)$. This module illustrates the **swept-sine** and the **Golay complementary sequence** techniques for identifying finite impulse responses. The first example measures the impulse response of a highpass filter using the Golay method, and the second example measures the impulse response of a weakly nonlinear loudspeaker driver using the swept-sine method.

3.2 Summary Of Objectives

- To provide a general framework for **characterizing single-input, single-output linear systems**.
- To demonstrate how to measure the **impulse response** of a linear system using **Golay complementary sequences**.
- To explain some of the limitations of making measurements using standard sound interfaces.
- To demonstrate how to find the **minimum-phase spectrum** corresponding to a complex spectrum.
- To explain how to measure the impulse response of a system even if the excitation source **motor is weakly nonlinear**. This measurement technique uses a **sine sweep** test signal.

¹This content is available online at <<http://cnx.org/content/m15946/1.6/>>.

3.3 Installation

1. Install a sound card, sound interface, or other full-duplex data acquisition card.
2. Consider testing the data acquisition card viewing the soundcard set-up instructions².
3. Install either MATLAB³ or Octave⁴.
4. Install pd⁵. (Alternatively, you may use other software that is capable of recording a system's output for a given input excitation signal. The software should also be capable of reading and writing WAV files. For example, any multitrack recording software should be fine.)
5. Download tf_meas.zip⁶ and unzip the contents into a conveniently located local directory.

3.4 Linear System

Consider the causal, single-input single-output (SISO) system shown in Figure 3.2. For simplicity, we will take the system to be linear and discrete-time, so that it is characterized by its impulse response $h(n)$ or equivalently its transfer function $H(z)$, which is the z transform of $h(n)$.

$$h(n) \leftrightarrow H(z) \quad (3.1)$$

We will assume that both $h(n)$ and $H(z)$ exist so that we can discuss measuring them interchangeably. We will further assume that $h(n)$ has finite length so that we can measure the response to an input signal in a finite amount of time. The goal of this document is to explain how to excite the system with a signal $s(n)$, measure the response $r(n)$, and use $s(n)$ and $r(n)$ to determine $h(n)$ (and equivalently $H(z)$). In particular, it is useful to pick a signal $s(n)$ that contains a large amount of energy so that measurement noise will not significantly corrupt the measurement results.

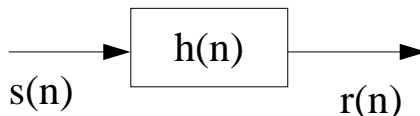


Figure 3.2: Linear system

3.5 Limitations Of Sound Interfaces

Sound cards and sound interfaces are not designed for making transfer function measurements. They merely provide a cost-effective solution since almost all computers have sound cards. We demonstrate these weaknesses given measurements made on a PreSonus Firepod sound interface. The output from channel 1 was directly connected to the line input on channel 1, and the sampling rate was $f_S = 44.1\text{kHz}$.

1. Sound interfaces attenuate frequencies near DC and near half of the sampling rate $f_S/2$. Figure 3.3 shows the measured magnitude response of the sound interface. Ideally it would be constant for all frequencies. However, Figure 3.3 shows that the -3dB point for the DC blocker lies near 5Hz. The

²http://ccrma.stanford.edu/realsimple/soundcard_test/

³<http://www.mathworks.com/>

⁴<http://www.octave.org/>

⁵<http://cra.ucsd.edu/msp/software.html>

⁶http://ccrma.stanford.edu/realsimple/imp_meas/tf_meas.zip

anti-aliasing filters cause the magnitude to roll off sharply such that the high frequency -3dB roll-off point is about 20.9kHz.

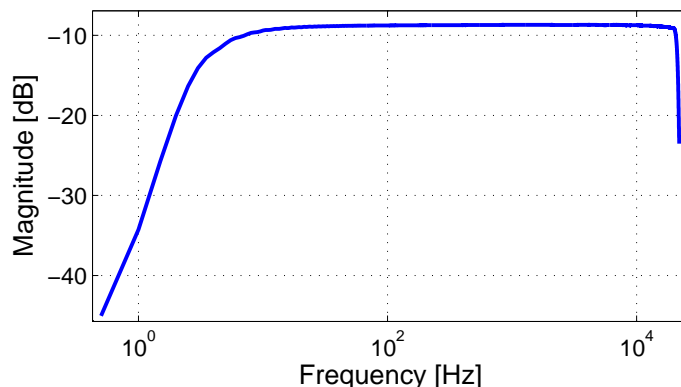


Figure 3.3: Magnitude response of the PreSonus FirePOD sound interface when the channel 1 output is directly connected to the channel 1 line input

2. The same sound interface filters cause the phase response measurement to be incorrect as well. Although ideally the phase response would be 0 radians for all frequencies, Figure 3.4 reveals that the phase measurement is distorted in roughly the same regions where the magnitude measurement is distorted.

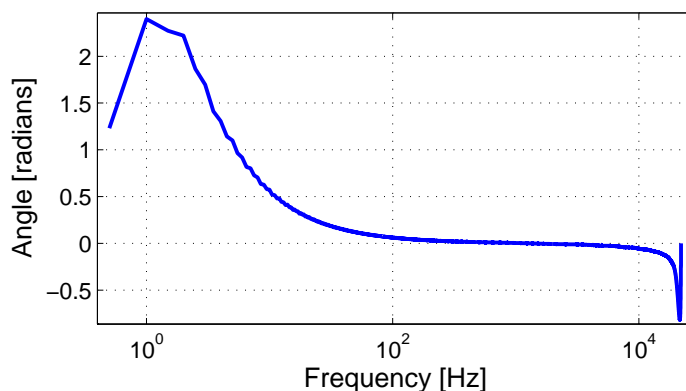


Figure 3.4: Phase response of the PreSonus FirePOD sound interface when the channel 1 output is directly connected to the channel 1 line input

3. Even when a computer is programmed to pass an input signal flowing into the sound interface input out of the sound interface output as fast as possible, there is a **delay** or **latency**. This delay is typically on the order of tens of milliseconds. Figure 3.5 shows the impulse response measured on the PreSonus card. The latency is approximately 62ms. The latency could have been decreased some by adjusting software settings in pd . The impulse response also rings noticeably due to the anti-aliasing filters.

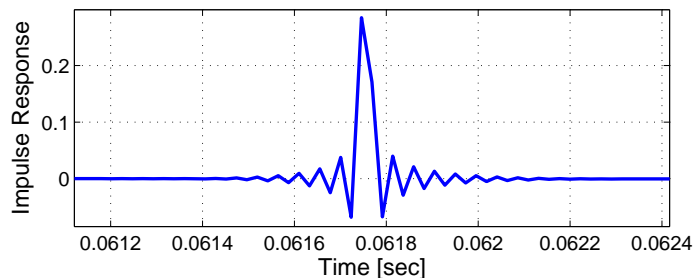


Figure 3.5: Impulse response of the PreSonus FirePOD sound interface when the channel 1 output is directly connected to the channel 1 line input

For reference, we provide the measured impulse response⁷ shown in Figure 3.5.

3.6 Minimum Phase Systems

One further consequence of the delay is that determining the phase response of the measured system is more complicated. The delay is responsible for a linear phase term since $\delta(n - k) \leftrightarrow e^{-j2\pi fk/f_s}$. If the delay is known (or measured), then it may be removed by multiplying the measured spectrum by $e^{j2\pi fk/f_s}$. However, if the system being measured is known to be minimum phase, then the **folded-cepstrum method**⁸ may be applied to find the minimum phase frequency response corresponding to the measured frequency response.

The transfer function measurement toolbox assumes that the system being measured is minimum phase. This is a valid assumption in many cases. For instance, all strictly positive real transfer functions are minimum phase. Dissipative systems are strictly positive real (and therefore minimum phase) if the appropriate quantity is measured and the sensor and motor are collocated. For example, if $s(n)$ controls a motor exerting a force on a dissipative system, and $r(n)$ is the velocity at that same point, the corresponding transfer function will be minimum phase. This holds for other dual variable pairs such as torque and angular velocity, voltage and current, and pressure and fluid flow.

For systems that are not minimum phase, such as systems involving a transmission delay between the input and output quantities, the phase plotted by the transfer function measurement toolbox is not the system phase response, but rather the minimum phase response corresponding to the measured system phase response.

3.7 High Pass Filter Measurement

The circuit shown in Figure 3.6 was measured using the Golay complementary sequence method to show how the sound interface non-idealities affect a measurement. V_{IN} was connected to the output of channel 1 of the PreSonus sound interface, and V_{OUT} was connected to the line input of channel 1 of the interface.

⁷http://ccrma.stanford.edu/realsimple/imp_meas/direct2ImpResp.wav

⁸http://ccrma.stanford.edu/jos/filters/Matlab_listing_mps_m_test.html

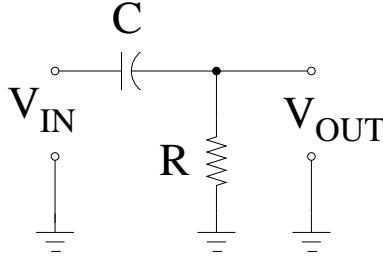


Figure 3.6: High pass filter electrical circuit

The analog transfer function $H(f)$ can be determined analytically using the voltage divider rule:

$$H(f) = \frac{V_{OUT}(f)}{V_{IN}(f)} = \frac{R}{R + \frac{1}{j2\pi fC}} = \frac{j2\pi fRC}{j2\pi fRC + 1} \quad (3.2)$$

In this case, $R = 1\text{k}\Omega$ and $C = 0.47\mu\text{F}$, so the -3dB point is about $f_{3dB} = \frac{1}{2\pi RC} \approx 340\text{Hz}$. Figure 3.7 and Figure 3.8 show that the frequency response is accurately measured in the range of about 10Hz to about $9f_S/20$.

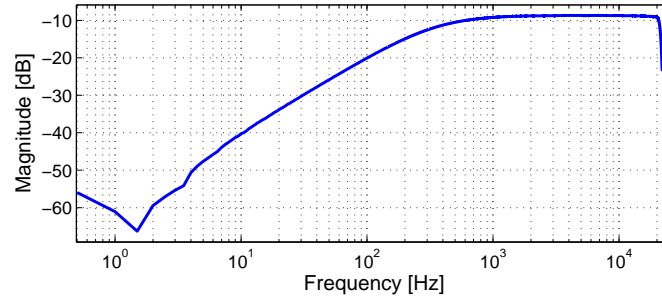


Figure 3.7: Measured magnitude response of the high pass filter

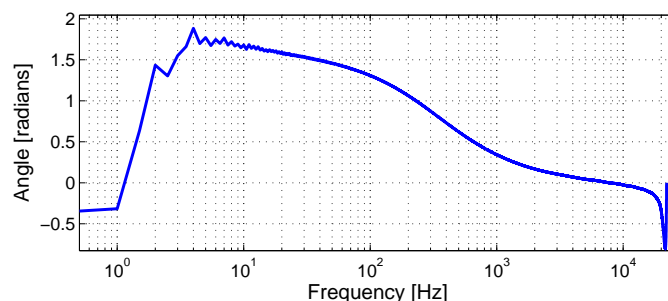


Figure 3.8: Measured phase response of the high pass filter

The ringing in the measured impulse response distracts from the more subtle characteristics of the ideal high pass filter impulse response. For transfer functions that pass large amounts of energy at high frequencies, it may be more instructive to inspect the frequency domain measurement results.

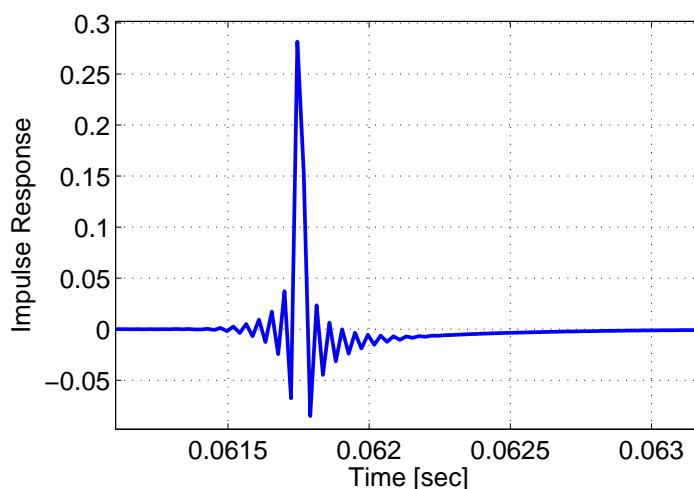


Figure 3.9: Impulse response

3.8 Sine Sweep Measurement of a Weakly Nonlinear Loudspeaker Driver

To exereaggerate the nonlinearity of a loudspeaker, we cut the cone of a mishandled driver as shown in Figure 3.10. We monitored the sound pressure several centimeters in front of the dustcap using an Audio Technica AT4049a microphone, which has a flat magnitude response to within 3dB from 100Hz to 5kHz.

The output from channel 1 of the PreSonus sound interface was connected to the speaker via a power amplifier, and the microphone was connected to the microphone input of channel 1 on the sound interface. The following results are typical of sine sweep measurements⁹ to show how with a weakly nonlinear motor.



Figure 3.10: Mishandled driver with additional cuts in the cone

Inverse filtering the measured response results in Figure 3.11, which is a plot of `nonlinear2ImpResp.wav`¹⁰. The linear contribution corresponds to the spike at the beginning, while the weakly nonlinear terms are clustered closer to the end of the response.

⁹http://ccrma.stanford.edu/real/simple/imp_meas/

¹⁰http://ccrma.stanford.edu/real/simple/imp_meas/nonlinear2ImpResp.wav

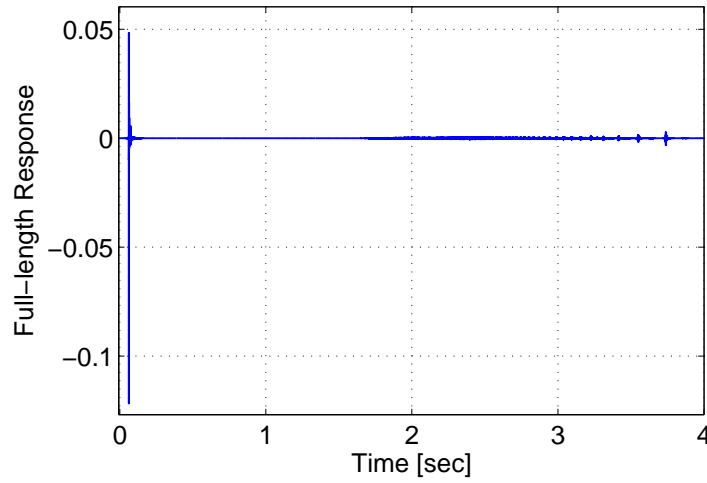


Figure 3.11: Full-length response

The main linear contribution is cut out and plotted in Figure 3.12. The measurement was not made in an anechoic chamber, so there is a reflection about 15ms after the main impact.

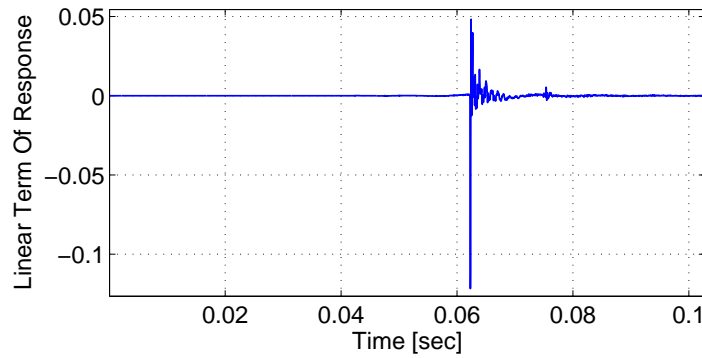


Figure 3.12: Linear impulse response term

The nonlinear terms are shown magnified in Figure 3.13. The lower order nonlinear terms toward the right have larger magnitude but overlap less in time (see Figure 3.13). Note that (2.3) implies that the overlapping could be reduced by increasing the total length T of the sweep excitation signal.

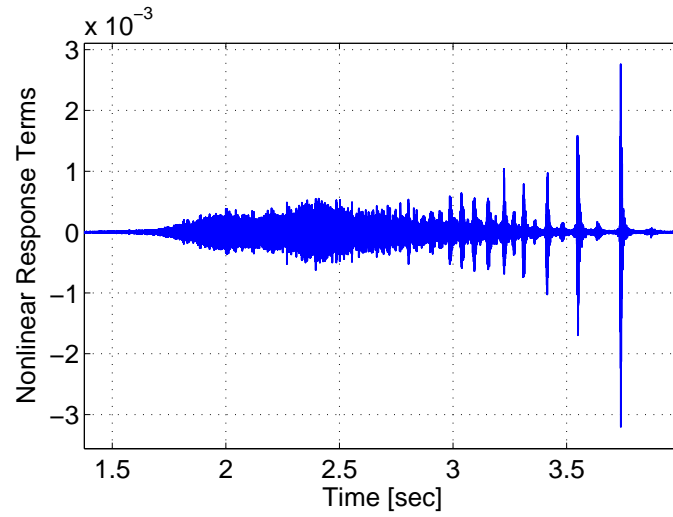


Figure 3.13: Nonlinear response terms

The magnitude and phase responses corresponding to the linear impulse response term from Figure 3.12 are shown in Figure 3.14 and Figure 3.15 in **blue**. For comparison, another sine sweep measurement was made at a lower level so that the speaker behaved approximately linearly. Decreasing the level also resulted in more noise and even some systematic error, as is evidenced by the **red** curves in Figure 3.14 and Figure 3.15. This comparison demonstrates that making measurements at larger levels can reduce the effects of noise, while nonlinear motor effects can be overcome with the sine sweep measurement technique.

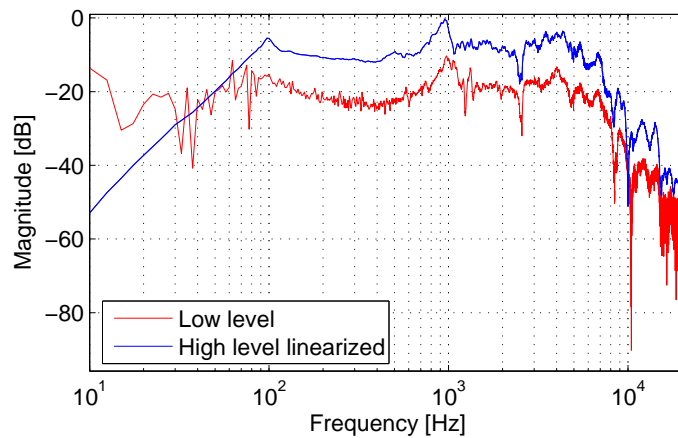


Figure 3.14: Proportional to magnitude response of $h(n)$

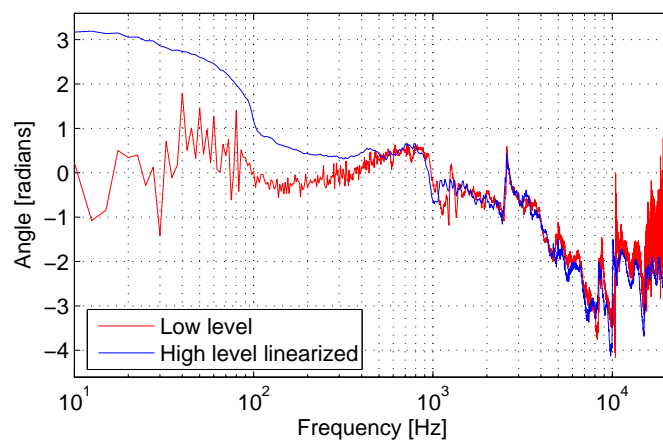


Figure 3.15: Phase response of $h(n)$

Bibliography

- [1] J. Abel and D. Berners. *Signal Processing Techniques for Digital Audio Effects*. <http://ccrma.stanford.edu/courses/424/>, 2005.
- [2] J. Abel and D. Berners. *Signal Processing Techniques for Digital Audio Effects*. <http://ccrma.stanford.edu/courses/424/>, 2005.
- [3] Angelo Farina. Simultaneous measurement of impulse response and distortion with a swept-sine technique. *Audio Engineering Society Convention*, Preprint 5093, Feb. 2000.

Index of Keywords and Terms

Keywords are listed by the section with that keyword (page numbers are in parentheses). Keywords do not necessarily appear in the text of the page. They are merely associated with that section. *Ex.* apples, § 1.1 (1) **Terms** are referenced by the page they appear on. *Ex.* apples, 1

F frequency response measurement, § 3(9)

G Golay, § 1(1)
Golay complementary sequences, § 3(9)

I impulse response measurement, § 3(9)

L loudspeaker, § 3(9)

S swept sine, § 2(5), § 3(9)
swept sinusoid, § 3(9)
system identification, § 1(1), § 2(5), § 3(9)

Attributions

Collection: *Impulse Response Measurement Toolbox*

Edited by: Edgar Berdahl, Julius Smith

URL: <http://cnx.org/content/col10519/1.5/>

License: <http://creativecommons.org/licenses/by/2.0/>

Module: "Impulse Response Measurement Using Golay Complementary Sequences"

By: Edgar Berdahl, Julius Smith

URL: <http://cnx.org/content/m15947/1.10/>

Pages: 1-3

Copyright: Edgar Berdahl, Julius Smith

License: <http://creativecommons.org/licenses/by/2.0/>

Module: "Swept Sine Impulse Response Measurement"

By: Edgar Berdahl, Julius Smith

URL: <http://cnx.org/content/m15945/1.4/>

Pages: 5-7

Copyright: Edgar Berdahl, Julius Smith

License: <http://creativecommons.org/licenses/by/2.0/>

Module: "Measuring Impulse Responses Using Golay Complementary Sequences and Swept Sinusoids"

By: Edgar Berdahl, Julius Smith

URL: <http://cnx.org/content/m15946/1.6/>

Pages: 9-18

Copyright: Edgar Berdahl, Julius Smith

License: <http://creativecommons.org/licenses/by/2.0/>

Impulse Response Measurement Toolbox

The Impulse Response Measurement Toolbox is a simple, convenient, and free open-source solution for measuring impulse responses, magnitude spectra, and phase spectra of single-input, single-output (SISO) linear systems. Two impulse-response measurement methods are explained and demonstrated. The Golay code measurement technique is particularly robust to additive white noise, while the swept sine measurement technique is robust to weak nonlinearities.

About Connexions

Since 1999, Connexions has been pioneering a global system where anyone can create course materials and make them fully accessible and easily reusable free of charge. We are a Web-based authoring, teaching and learning environment open to anyone interested in education, including students, teachers, professors and lifelong learners. We connect ideas and facilitate educational communities.

Connexions's modular, interactive courses are in use worldwide by universities, community colleges, K-12 schools, distance learners, and lifelong learners. Connexions materials are in many languages, including English, Spanish, Chinese, Japanese, Italian, Vietnamese, French, Portuguese, and Thai. Connexions is part of an exciting new information distribution system that allows for **Print on Demand Books**. Connexions has partnered with innovative on-demand publisher QOOP to accelerate the delivery of printed course materials and textbooks into classrooms worldwide at lower prices than traditional academic publishers.