



HOCHSCHULE KONSTANZ TECHNIK, WIRTSCHAFT UND GESTALTUNG
UNIVERSITY OF APPLIED SCIENCES

**Technische Grundlagen
der angewandten Informatik**

«Versuch 3»

Benedict Roth, David Kubatzki

Konstanz, 17. Mai 2015

Zusammenfassung (Abstract)

Thema: «Versuch 3»

Autoren:	Benedict Roth	beroth@htwg-konstanz.de
	David Kubatzki	dakuba@htwg-konstanz.de

Betreuer:	Prof. Dr. Matthias O. Franz	mfranz@htwg-konstanz.de
	Jürgen Keppler	juergen.keppler@htwg-konstanz.de
	Martin Miller	martin.miller@htwg-konstanz.de

«In diesem Versuch wird die Fourieranalyse auf akustische Signale und Systeme angewandt.»

Inhaltsverzeichnis

Abbildungsverzeichnis	IV
Tabellenverzeichnis	V
Listingverzeichnis	VI
Abkürzungsverzeichnis	VII
1 Einleitung	1
2 Versuch 1	2
2.1 Fragestellung, Messprinzip, Aufbau, Messmittel	2
2.1.1 Messmittel	2
2.1.2 Messprinzip	2
2.1.3 Aufbau	3
2.2 Messwerte	3
2.3 Auswertung	4
2.4 Interpretation	4
3 Versuch 2	6
3.1 Fragestellung, Messprinzip, Aufbau, Messmittel	6
3.1.1 Messprinzip	6
3.1.2 Messmittel	6
3.1.3 Aufbau	6
3.2 Messwerte	6
3.3 Auswertung	7
3.4 Interpretation	7

Anhang	10
A.1 Quellcode	10
A.1.1 Quellcode Versuch 1	10
A.1.2 Quellcode Versuch 2	10
A.1.3 Quellcode Versuch 3	10
A.1.4 Quellcode Versuch 4	10
A.2 Messergebnisse	10

Abbildungsverzeichnis

2.1	Ton	4
2.2	fouriertransformierte	5
3.1	amplitudengang klein	8
3.2	phasengang klein	8
3.3	phasengang gross	9
3.4	phasengang gross	9

Tabellenverzeichnis

Listingverzeichnis

Abkürzungsverzeichnis

1

Einleitung

Dieser Versuch besteht aus zwei Unterversuchen. Für beide wird ein Mikrofon an das Oszilloskop angeschlossen, mit welchem dann unterschiedliche Frequenzen aufgenommen. Im ersten Versuch ist die Geräuschquelle eine Mundharmonika auf welcher wir einen Ton spielen. Die gemessenen Daten werden dann vom Oszilloskop an den Laborrechner übertragen und mit einem Python Skript eingelesen. Beim zweiten Versuch benutzen wir einen Frequenz/Funktionsgenerator welcher sowohl an einen Lautsprecher als auch an das Oszilloskop angeschlossen ist. Es werden mithilfe des Generators, unterschiedliche Frequenzen im Bereich von 100 Hz bis 10 KHz erzeugt und diese Ausgangswerte dann mit den Eingangswerten verglichen. Die Phasenverschiebung und die Amplitude des Ausgangssignals wird dann von Hand notiert und anschließend am PC grafisch dargestellt. Zum Schluss wird noch ein Bode Diagramm erzeugt.

2

Versuch 1

2.1 Fragestellung, Messprinzip, Aufbau, Messmittel

Im ersten Teil war gefordert, dass wir mithilfe eines Mikrofon einen Mundharmonika Ton per Oszilloskop einlesen und anschließend die Fouriertransformierte des Signals grafisch darstellen.

2.1.1 Messmittel

Als Messmittel dient ein 250g schweres Dynamisches Mikrofon mit einer Tauchspule (engl. moving coil). Es ist Uni-direktional und reagiert auf Frequenzen im Bereich von 70Hz bis 13 KHz und hat eine Sensitivität von $-54\text{dB} \pm 3\text{dB}$. Die Ausgangs impedanz beträgt $500\Omega \pm 30\%$.

2.1.2 Messprinzip

Das Tauchspulenmikrofon (auch Tauchspulmikrofon) ist ein elektroakustischer Wandler, der nach dem elektroinduktiven Prinzip des dynamischen Mikrofons arbeitet. Es ist sowohl die Bauform des Druckgradientenmikrofons als auch die des Druckmikrofons üblich.

Der Begriff Tauchspulenmikrofon bezieht sich auf die technische Anordnung der Bauelemente des Wandlers: Bei dem Tauchspulenmikrofon ist die Membran fest mit einer Magnet-Spule verbunden, die durch die Membranbewegung in ein statisches dauermagnetisches Feld „eintaucht“. Siehe auch: Tauchspule. Die relative Bewegung von Spule und Magnetfeld erzeugt per Induktion die Signalspannung. Diese ist proportional zur Membrangeschwindigkeit.

Tauchspulenmikrofone benötigen keine nachträgliche Impedanzanpassung und auch keine Symmetrierung; beides kann allein durch die Dimensionierung und Verschaltung der Spule erreicht werden.

Prinzipielle Nachteile: Die Schallwelle muss die Masse der Membran mit der Spule bewegen und elektrische Arbeit leisten. Tauchspulenmikrofone haben daher eine geringe Empfindlichkeit und zeigen eine Trägheit im Einschwingverhalten, wodurch feinste Details nicht erfasst werden, was jedoch erwünscht sein kann: Sie liefern ein „erdiges“, kräftiges Klangbild, hochwertige Modelle werden daher durchaus auch bei Studioaufnahmen verwendet. Tauchspulenmikrofone haben ein nicht so hohes Übertragungsspektrum wie Kondensatormikrofone und sind aufgrund ihrer geringen Empfindlichkeit für Fernaufnahmen ungeeignet. Die relativ hohe Masse des Membransystems lässt sie zudem empfindlich auf Körperschall, etwa Hantierungsgeräusche, reagieren; um solche Störungen zu verringern, ist bei hochwertigen Tauchspulenmikrofonen die gesamte technische Einheit (die Mikrofonkapsel) im Mikrofongehäuse schwingfähig gelagert.

Die Vorteile dieses Mikrofontyps zeigen sich darin, dass sie in der Regel gegenüber mechanischen Belastungen recht robust sind und hohe Schalldrücke vertragen. Auch benötigen sie keine Spannungsversorgung, was im mobilen Betrieb von Vorteil sein kann. Die einfache Bauart erlaubt preisgünstige Fertigung und macht diesen Mikrofontyp nahezu unverwundlich.

2.1.3 Aufbau

Das Mikrofon ist über ein Kabel mit dem Oszilloskop verbunden. Als Tonerzeuger benutzen wir eine handelsübliche Mundharmonika, bei welcher wir alle Löcher bis auf eines zugeklebt haben. Dies tun wir um zu verhindern, dass mehrere Töne gleichzeitig gespielt werden.

2.2 Messwerte

Während wir mit der Mundharmonika einen Ton in das Mikrofon spielen, zeichnen wir die Messwerte mithilfe des Oszilloskopes auf, indem wir den Knopf "Single Seq" drücken. Diese Momentaufnahme übertragen wir nun mithilfe eines Pythonskriptes an den PC. Hierzu importieren wir das Modul `TekTDS2000`. Dieses stellt uns die Funktion `getData()` zur Verfügung, welches uns ein Array aus Messdaten übergibt.

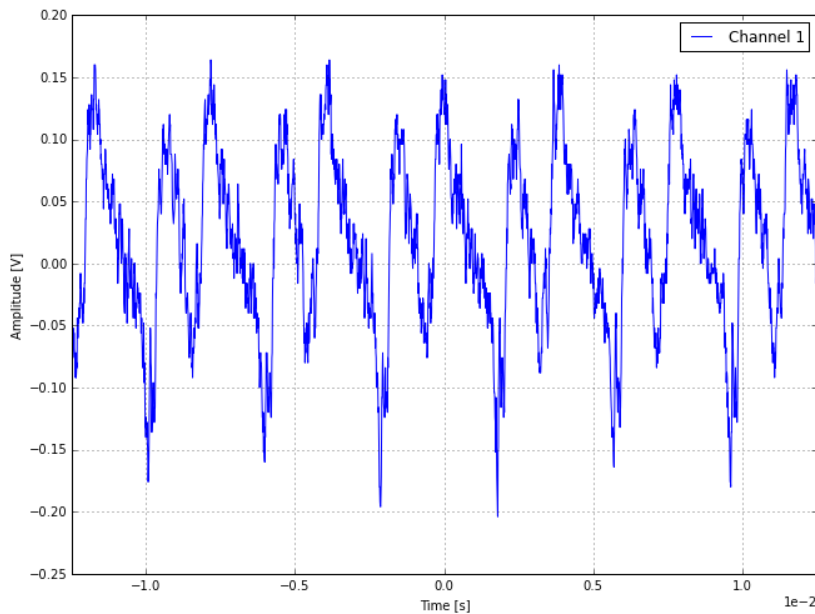


Abbildung 2.1: Ton

2.3 Auswertung

Auf die eingelesenen Werte wenden wir nun die Fouriertransformation mithilfe der numpy Funktion `fft.fft()` an. Hieraus berechnen wir nun das Amplitudenspektrum, nun haben wir noch das Problem, dass die Frequenzachse in Anzahl Schwingungen innerhalb der gesamten Signaldauer vorliegt und wir diese noch in Herz umrechnen müssen. Hierzu benutzen wir die Formel

$$f = \frac{n}{M \cdot \Delta t}$$

Nun können wir die Grundfrequenz bestimmen (252 Hz) und die dazugehörige Amplitude (0,026).

2.4 Interpretation

Durch die Fouriertransformation waren wir in der Lage die Grundfrequenz und somit die Tonhöhe zu bestimmen. Desweiteren können wir dessen Amplitude berechnen und somit die Lautstärke.

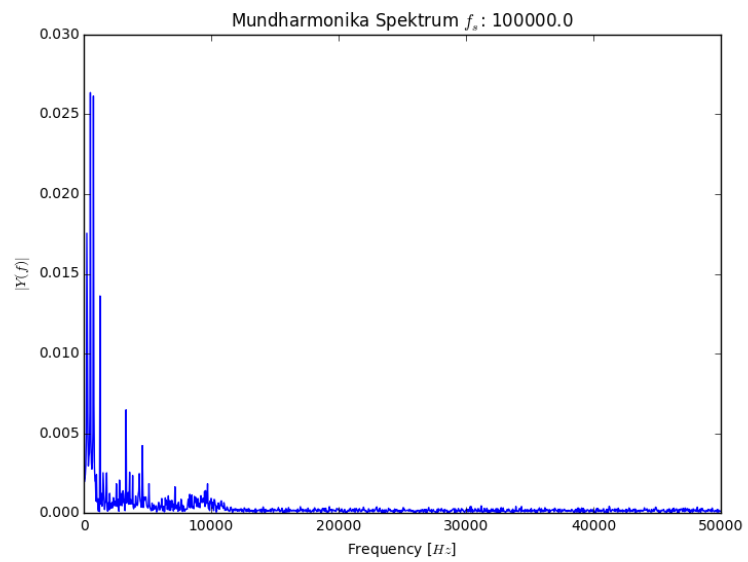


Abbildung 2.2: fouriertransformierte

3

Versuch 2

3.1 Fragestellung, Messprinzip, Aufbau, Messmittel

In Versuch 2 erzeugen wir verschiedene Frequenzen und lassen diese über einen Lautsprecher von einem Mikrofon aufnehmen. Hieraus messen wir dann die Amplitude und die Phasenverschiebung.

3.1.1 Messprinzip

Das Messprinzip entspricht dem aus Versuch 1.

3.1.2 Messmittel

Die Messmittel entsprechen auch denen aus Versuch 1.

3.1.3 Aufbau

Hier benutzen wir wieder das Oszilloskop sowie das Mikrofon aus Versuch 1. Zusätzlich haben wir nun noch einen Synthesizer und einen Lautsprecher mit dem Oszilloskop verbunden. Der Lautsprecher besitzt jeweils einen getrennten Mittel sowie einen Hochtöner.

3.2 Messwerte

Wir haben für verschiedene Frequenzen die jeweilige Amplitude und Phasenverschiebung für die entsprechenden Lautsprecher gemessen. Hierzu haben wir das Ausgangssignal mit dem

Eingangssignal mithilfe des Oszilloskops verglichen. Diese Messwerte haben wir anschließend per Hand notiert und diese mithilfe eines Python skripts in numpy arrays gespeichert.

3.3 Auswertung

Wir haben für jeden der beiden Lautsprecher ein Bode Diagramm erstellt. Hierzu rechnen wir den Amplitudengang in Dezibel um und stellen ihn in Abhängigkeit zur Frequenz dar. Desweiteren berechnen wir den Phasenwinkel nach der Formel

$$\varphi_H = -\Delta t \cdot f \cdot 360^\circ$$

und stellen diesen ebenfalls in Abhängigkeit zur Frequenz dar. Die Diagramme werden jeweils halblogarithmisch dargestellt, dies geschieht mithilfe der Funktion `semilogx()`.

3.4 Interpretation

Wenn man den jeweiligen Phasengang der beiden Lautsprecher betrachtet sieht man dass sich der Phasenwinkel bei höherer Frequenz tendenziell verringert. Anhand der Amplitudengänge kann man sehen, dass der große Lautsprecher eher niedrigere Frequenzen (ca. 200Hz) und der kleine Lautsprecher etwas höhere Frequenzen (ca. 500Hz) am stärksten verstärkt.

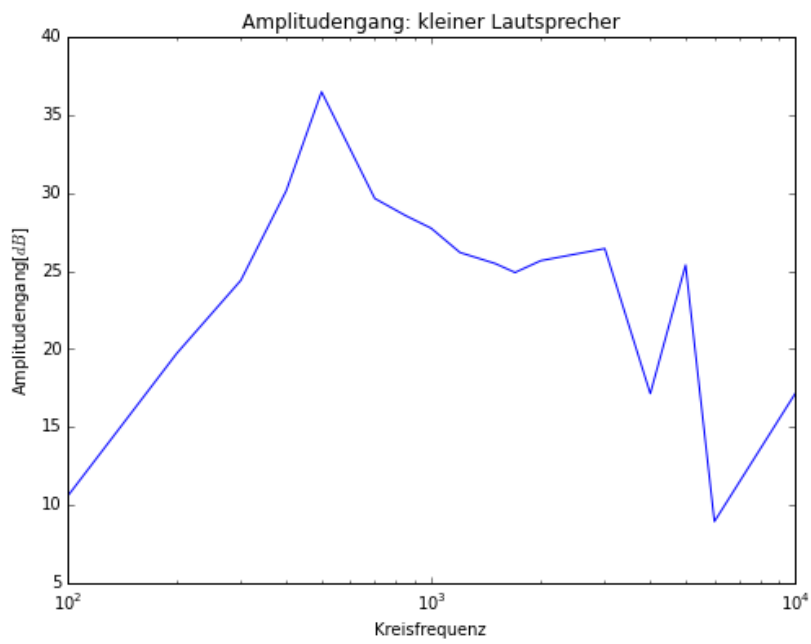


Abbildung 3.1: amplitudengang klein

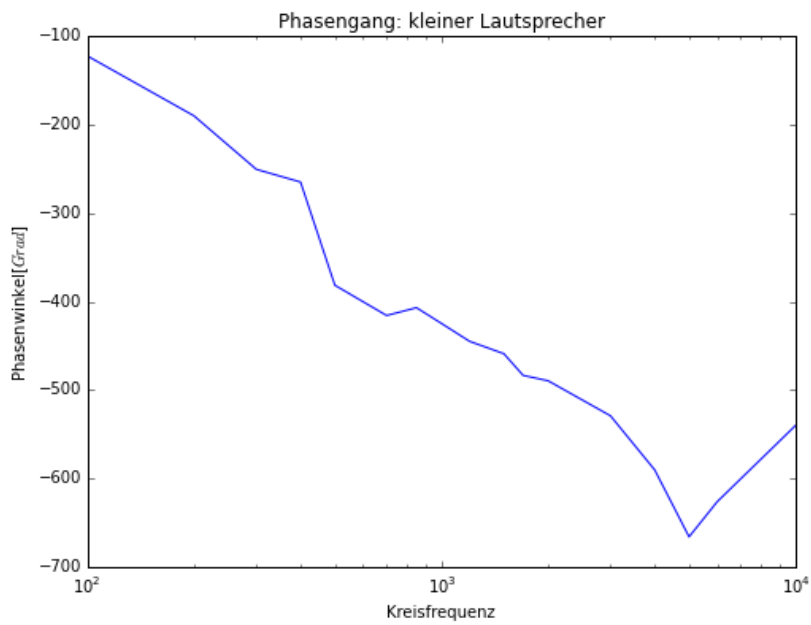


Abbildung 3.2: phasengang klein

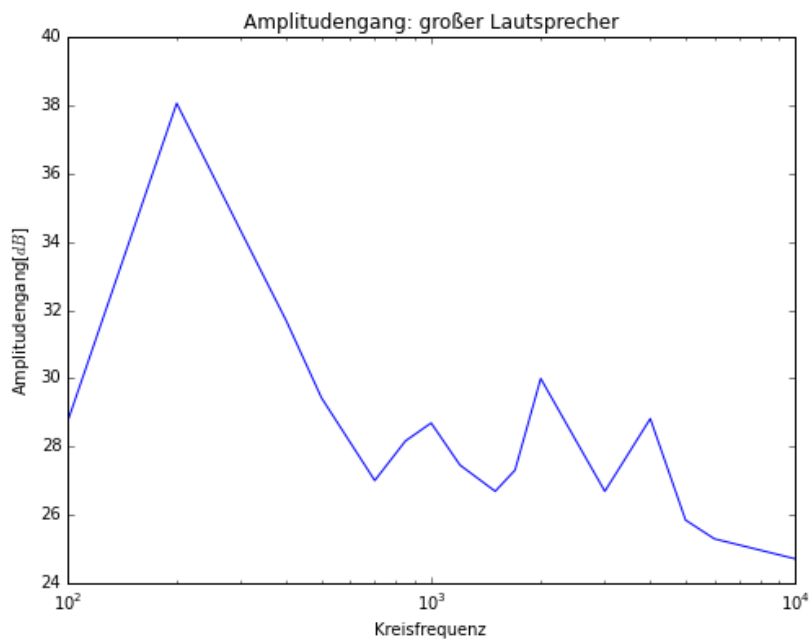


Abbildung 3.3: amplitudengang gross

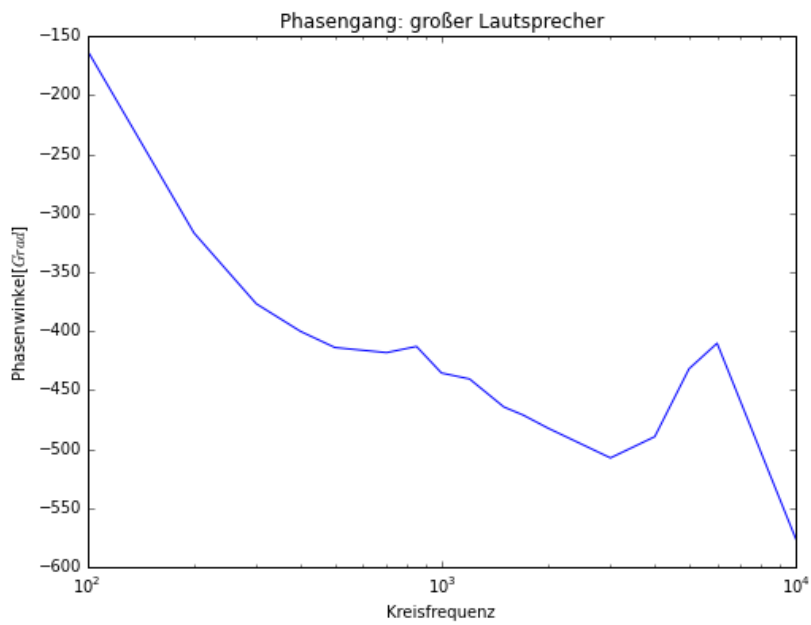


Abbildung 3.4: phasengang gross

listings

Anhang

A.1 Quellcode

A.1.1 Quellcode Versuch 1

```
# -*- coding: utf-8 -*-
"""
Spyder_Editor

This_is_a_temporary_script_file.
"""

import TekTDS2000 as tds
import numpy as np
import matplotlib.pyplot as plt
scope = tds.TekTDS2000()

x,y=scope.getData(1)
#np.savetxt("v1.csv", np.asarray([x,y]).T, delimiter=",")

dt=scope.getSamplingInterval()
spec=np.abs(np.fft.fft(y))

f=np.zeros(len(spec))

for i in range(0,len(spec)):
    f[i]=i/(len(spec)*dt)
```

```

fig , ax = plt.subplots(figsize=(800/100, 600/100), dpi=100)
ax.plot(f[0: len(spec)/2], spec[0: len(spec)/2]/ len(spec))
ax.set_xlabel('Frequency_[$Hz$]')
ax.set_ylabel('$|Y(f)|$')
ax.set_title('Mundharmonika_Spektrum_ $f_s$:_' +str(np.ceil(1/dt)))
plt.savefig('v1_fft.png', dpi=100)

print('Grundfrequenz:_' +str(np.argmax(spec[0: len(spec)/2])/( len(spec)* d

amp = spec[np.argmax(spec[0: len(spec)/2])]/ len(spec)
print('Amplitude:_' +str(amp))

```

A.1.2 Quellcode Versuch 2

```

# -*- coding: utf-8 -*-
"""
Created on Wed May 13 11:47:30 2015

@author: _dakubatz
"""

import numpy as np
import matplotlib.pyplot as plt
pg1 = []
pg2 = []

freq=[100,200,300,400,500,700,850,1000,1200,
      1500,1700,2000,3000,4000,5000,6000,10000]
amp1=[27.2,80,52,38.4,29.6,22.4,25.6,27.2,23.6,
      21.6,23.2,31.6,21.6,27.6,19.6,18.4,17.2]
ph1=[4.5,4.4,3.49,2.78,2.3,1.66,1.35,1.21,1.02,
      0.86,0.77,0.67,0.47,0.34,0.24,0.19,0.16]

amp2=[3.36,9.68,16.6,32.2,66.8,30.4,26.8,24.4,
      20.4,18.8,17.6,19.2,21,7.2,18.6,2.8,7.2,]

```

```
ph2=[3.4,2.64,2.32,1.84,2.12,1.65,1.33,1.18,1.03,
      0.85,0.79,0.68,0.49,0.41,0.37,0.29,0.15]
```

```
for i in range (0,17):
    pg1.append(-1*ph1[i]/1000*freq[i]*360)
    pg2.append(-1*ph2[i]/1000*freq[i]*360)
```

```
fig, ax = plt.subplots(figsize=(800/100, 600/100), dpi=100)
ax.semilogx(freq,20*np.log10(amp1))
ax.set_xlabel('Kreisfrequenz')
ax.set_ylabel('Amplitudengang[$dB$]')
ax.set_title('Amplitudengang:_gro_er_Lautsprecher')
```

```
fig, ax = plt.subplots(figsize=(800/100, 600/100), dpi=100)
ax.semilogx(freq,pg1)
ax.set_xlabel('Kreisfrequenz')
ax.set_ylabel('Phasenwinkel[$Grad$]')
ax.set_title('Phasengang:_gro_er_Lautsprecher')
```

```
fig, ax = plt.subplots(figsize=(800/100, 600/100), dpi=100)
ax.semilogx(freq,20*np.log10(amp2))
ax.set_xlabel('Kreisfrequenz')
ax.set_ylabel('Amplitudengang[$dB$]')
ax.set_title('Amplitudengang:_kleiner_Lautsprecher')
```

```
fig, ax = plt.subplots(figsize=(800/100, 600/100), dpi=100)
ax.semilogx(freq,pg2)
ax.set_xlabel('Kreisfrequenz')
ax.set_ylabel('Phasenwinkel[$Grad$]')
ax.set_title('Phasengang:_kleiner_Lautsprecher')
```

A.2 Messergebnisse

großer Box Lautsprecher 11.5.2015

Frequenz:	Amplitude:	Phasenverschiebung:
100 Hz	27,2 mV	4,5 ms
200 Hz	80 mV	4,4 ms
300 Hz	52 mV	0,16 ms + $\frac{1}{300 \text{ Hz}}$ 3,69 ms
400 Hz	38,4 mV	5,49 ms + 0,28 ms + 2,78 ms 2,78 ms
500 Hz	29,6 mV	2,3 ms
700 Hz	22,4 mV	1,66 ms
850 Hz	25,6 mV	1,35 ms
1 kHz	27,2 mV	1,21 ms
1,2 kHz	23,6 mV	1,02 ms
1,5 kHz	21,6 mV	0,86 ms
1,7 kHz	23,2 mV	0,77 ms
2 kHz	21,6 mV	0,67 ms
3 kHz	21,6 mV	0,47 ms
4 kHz	27,6 mV	0,34 ms
5 kHz	21,6 mV	0,24 ms
6 kHz	18,4 mV	0,19 ms
10 kHz	17,2 mV	0,16 ms

Abbildung 4.5: messprotokoll1

kleines Lautsprecher

Frequenz	Amplitude	Phasenverschiebung
100	3,36 mV	3,4 ms
200	9,68 mV	2,64 ms
300	16,6 mV	2,32 ms
400	32,2 mV	1,84 ms
500	66,8 mV	0,12 ms 2,72 ms
700	30,4 mV	1,65 ms
850	26,8 mV	1,53 ms
1	24,4 mV	1,18 ms
1,2	20,4 mV	1,03 ms
1,5	18,8 mV	0,85 ms
1,7	17,6 mV	0,79 ms
2	19,2 mV	0,68 ms
3	21 mV	0,49 ms
4	7,2 mV	0,41 ms
5	18,6 mV	0,37 ms
6	2,8 mV	0,29 ms
10	7,2 mV	0,15 ms

Abbildung 4.6: messprotokoll2

[?, S.21]