



---

## M1 IMDEA - Digital Filtering

### Project

---

Two-ways speaker system simulation

# Introduction

## Aim of the work

The aim of the work is to simulation a two-ways speaker system using digital filters. It enables to listen to the theoretical response of the system and to design a digital crossover filters. Finally, you will design the crossover filters in Sigma Studio.

## System under study

The physical system under study is a two-ways speaker system as shown in Figure 1. The input signal  $x(t)$  is the digital audio signal that enters into a DSP where it is split into two paths: a) path 1 (woofer), b) path 2 (tweeter). The DSP implements two-way digital crossover filters. Two DSP outputs are amplified by gains  $G_1$  and  $G_2$  resulting in two voltage signals  $u_1(t)$  and  $u_2(t)$ .

These voltage signals are connected to two speakers (a woofer and a tweeter) placed in an infinite vertical ( $yz$ -axis) baffle. Since all the simulation happens in  $z = 0$ , the  $z$  axis is not anymore considerer in this work. Woofer coordinates are  $(x = 0, y = -0.3)$ , tweeter coordinates are  $(x = 0, y = 0.3)$ .

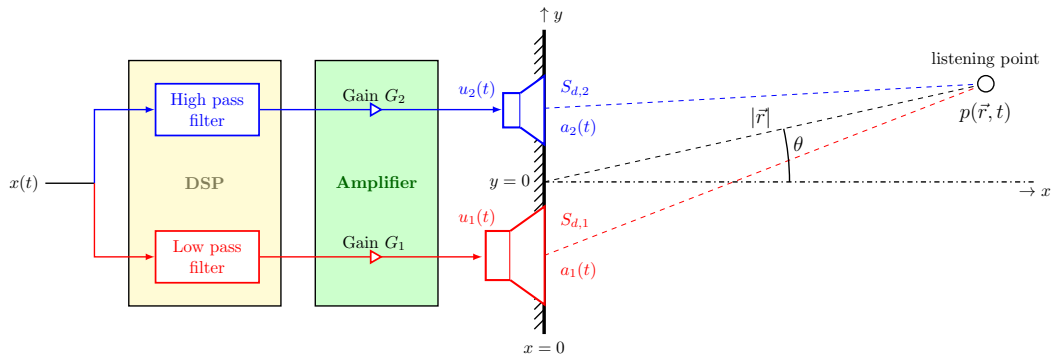


Figure 1: Principle of the two-ways speaker system.

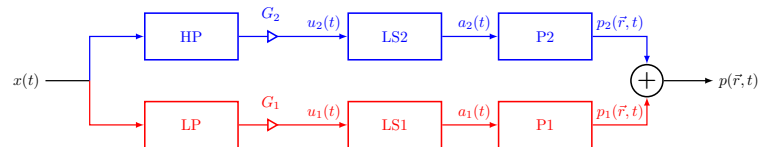


Figure 2: Electrical equivalent system of the two-ways system.

The equivalent electrical system is shown in Figure 2, with the following physical variables:

|                                    |                                                                                                                                |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| $x(t)$                             | input digital signal                                                                                                           |
| $u_1(t), u_2(t)$                   | output voltage after crossover filters (low pass and high pass respectively) and amplifiers                                    |
| $a_1(t), a_2(t)$                   | acceleration of loudspeaker membranes: $a_1(t)$ is the acceleration of the woofer and $a_2(t)$ the acceleration of the tweeter |
| $p_1(\vec{r}, t), p_2(\vec{r}, t)$ | acoustic pressures at the listening point $\vec{r}$ radiated respectively by the woofer and the tweeter                        |
| $p(\vec{r}, t)$                    | total acoustic pressure at the listening point $\vec{r}$<br>$p(\vec{r}, t) = p_1(\vec{r}, t) + p_2(\vec{r}, t)$                |

The following frequency domain notation is used in the assignment (symbol  $\xrightarrow{FT}$  stands for Fourier Transform):

$$\begin{array}{lll}
 \text{voltage} & u(t) & \xrightarrow{FT} U(\omega), \\
 \text{acceleration} & a(t) & \xrightarrow{FT} A(\omega), \\
 \text{pressure} & p(t) & \xrightarrow{FT} \hat{p}(\omega).
 \end{array}$$

Your goal is to code in Python functions and classes that simulate each part of the speaker system in the form of a digital filter. For the sake of simplicity, amplifiers are considered as a simple gain, and D/A converters at the output of DSP are having the sensitivity of 1.

## Work to do

The work is split into four following parts:

**work 1:** Acoustic propagation simulation

**work 2:** Loudspeaker simulation

**work 3:** Crossover network simulation & assembly

**work 4:** Implementation of crossovers in Sigma-Studio

# 1 Acoustic propagation simulation

The aim of this section is to simulate the effect of the acoustic propagation assuming that each loudspeaker is a baffled circular piston, as shown in Figure 3,

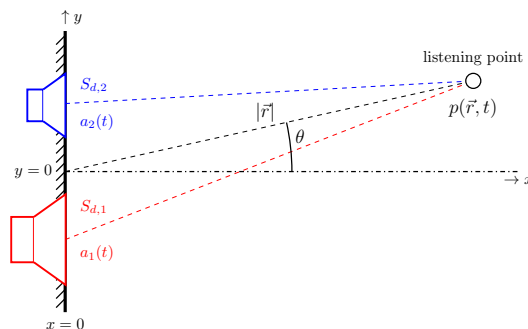


Figure 3: Equivalent acoustic system to the speaker.

for which the in-axis pressure at the distance  $r$  is written in the frequency domain as

$$\hat{p}(r, \omega) = j \frac{k \rho_0 c_0 W(\omega)}{2\pi} \left[ \frac{2J_1(k a \sin \theta)}{k a \sin \theta} \right] \frac{e^{-jkr}}{r}, \quad (1)$$

where  $k = \omega/c_0$  is the wave number,  $\omega$  the angular frequency,  $c_0$  is the speed of sound,  $\rho_0$  is the air density,  $W(\omega)$  is the piston volume velocity,  $a$  is the piston radius, and  $J_1$  is the Bessel function of the first kind. For the sake of simplicity we will consider  $k a \ll 1$  for which  $\frac{2J_1(k a \sin \theta)}{k a \sin \theta} \simeq 1$ .

1. Using Eq. (1), write the expression of the radiated pressure  $p(r)$  as a function of the loudspeaker membrane acceleration  $A(\omega)$ , and draw the amplitude and phase of the frequency response function  $H_{ap}(\omega) = \hat{p}(\omega)/A(\omega)$ . What is the main characteristic for the phase?
2. How can you write the equivalent impulse response  $h_{ap}(t)$  verifying  $p(r, t) = h_{ap}(t) * a(t)$ , where  $*$  is the convolution operator ?
3. Which digital filter enables to represent the impulse response  $h_{ap}(t)$ ?
4. Calculate the pressure  $p(r, t)$  for different angles  $\theta$  of the receiver as shown in Figure 3 using the audio file `music.wav` as accelerations  $a_1$  and  $a_2$ . Do this calculation for  $a_1 = a_2$  and  $a_1 = -a_2$ . Listen (see below python codes for reading a wave file and listening to it) to the calculated signal for each case and for different angles. What do you notice?

## Python code for reading a wave file and listening to it

```
import soundfile as sf
import sounddevice as sd

# load audio
data, fs = sf.read('music.wav')
audio_data = data[:, 1]

# play audio signal
sd.play(audio_data, fs)
sd.wait() # Wait until file is done playing
```

## 2 Loudspeakers simulation

The aim of this part is to design digital filters which simulate the response of each loudspeaker. The loudspeaker's parameters are given in appendix A (woofer) and B (tweeter). We consider, in this work, that these parameters are obtained while measured in an infinite baffle (in air).

1. Write, in hand, the analog transfer function  $\frac{A(s)}{U(s)}$  of an electrodynamic loudspeaker in the Laplace domain as a function of the usual loudspeaker parameters  $R_e$ ,  $L_e$ ,  $Bl$ ,  $M_{ms}$ ,  $R_{ms}$ , and  $K_{ms}$ .  $U(s)$  is the input voltage of the loudspeaker and  $A(s)$  is the acceleration of the membrane (considering piston motion) in the Laplace domain. Use appendix C if you do not know where to start.
2. Based on the previous result, write a Python function to get the coefficients  $B_2$ ,  $B_1$ ,  $B_0$ ,  $A_3$ ,  $A_2$ ,  $A_1$ , and  $A_0$  of an **analog** filter representing the transfer function  $\frac{A(s)}{U(s)}$  (Laplace domain) in the following form

$$\frac{A(s)}{U(s)} = \frac{B_2 s^2 + B_1 s + B_0}{A_3 s^3 + A_2 s^2 + A_1 s + A_0} \quad (2)$$

3. Use the bilinear transform (`scipy.signal.bilinear()`, see appendix D) to get the coefficients of a **digital** filter from the analog coefficients  $B_2$ ,  $B_1$ ,  $B_0$ ,  $A_3$ ,  $A_2$ ,  $A_1$ , and  $A_0$ .  
Note, that `scipy.signal.bilinear()` requires the analog coefficients in descending order  $B = [B_2, B_1, B_0]$  and  $A = [A_3, A_2, A_1, A_0]$ .
4. Apply previous steps to get the filter coefficients for each loudspeaker (LS1 = woofer, LS2 = tweeter). Which kind of digital filters do you build here (IIR, FIR)?
5. Plot the frequency response functions of both the digital and the analog filter and compare them. You can use the following functions `scipy.signal.freqs` and `scipy.signal.freqz` to calculate the frequency response functions of analog and digital filter respectively (see the `scipy` documentation).
6. Using the audio file `music.wav` as the loudspeakers input voltage  $u_1(t)$  and  $u_2(t)$  to listen to the acceleration  $a_1(t)$  and  $a_2(t)$ . Use `scipy.signal.lfilter` and the filters coefficients in this step. Do you hear the effect of each virtual loudspeaker?

### 3 Crossover filters design

The aim of this part is to design digital filters for crossovers.

#### Crossover design

Create a function `my_filter` (you can choose any name) that returns parameters `b` and `a` of a IIR filter for crossover filters (Butterworth, or Linkwitz–Riley<sup>1</sup>). The function takes following arguments:

`f_name` [string] filter name ("Butter", "LR")  
`f_order` [integer] filter order (1, 2, or 4)  
`f_type` [string] filter type (e.g. "low", "high")  
`fc` [float] filter cut-off frequency (in Hz)  
`inv` [boolean] inverted polarity (False / True)  
`fs` [integer] sample rate (in Hz)

An example of how to call this function to create a Linkwitz–Riley 4th order low-pass filter ( $f_c = 1$  Hz,  $f_s = 48$  kHz) is:

```
b, a = my_filter("LR", 4, "low", 1e3, False, 48e3)
```

#### Test the function

Use this function to create the following sets of crossovers with and plot their FRF and the FRF of their sum ( $f_c = 1$  kHz and  $f_s = 48$  kHz)

1. Butterworth 1st order
2. Butterworth 2nd order
3. Linkwitz–Riley 2nd order
4. Linkwitz–Riley 4th order

---

<sup>1</sup>Linkwitz–Riley of order  $N$  is made of two same Butterworth filters of order  $N/2$  connected in series. The transfer function  $H(z)$  of two filters in series is calculated as the multiplication of both transfer functions of the respective individual filters  $H_a(z)$ ,  $H_b(z)$ . Consequently, the  $b$  coefficients (of the  $H(z)$  filter) can be calculated as the convolution of the  $b_a$  and  $b_b$  coefficients of the respective individual filters  $H_a(z)$ ,  $H_b(z)$ , and the coefficients  $a$  (of the filter  $H_{tot}(z)$ ) can be calculated as the convolution of the coefficients  $a_a$  and  $a_b$  of the respective individual filters  $H_a(z)$ ,  $H_b(z)$

## Assemble all parts

You should now have all parts ready for final assembly of the 2-way system (Fig. 1). Choose a crossover filter, choose a cut-off frequency and simulate the whole chain (Fig. 2) with input signal  $x(t)$  and output signal the pressure  $p(t)$  at the listening point. Set the distance  $|r| = 1$  m and angle  $\theta = 0$ .

- Plot both the FRF of each way of the 2-way system. Use `freqz` and coefficients of all filters (crossover, speaker, propagation).
- Plot the final FRF of the 2-way system (sum of both speakers). Use `freqz` and coefficients of all filters (crossover, speaker, propagation).
- Chose  $x(t)$  as the audio signal from `music.wav` and listen to the signals  $p_1(t)$ , and  $p_2(t)$  and to the sum  $p(t)$ .

## 4 Implementation in Sigma Studio

The aim of the last homework is to use parts of previous homeworks to implement a two-way crossover filter on SigmaDSP ADAU-1701 using Sigma Studio.

1. Prepare the Frequency Response Function (pressure over voltage) of each loudspeaker (without crossover filters) as if it were measured with a microphone set to  $r = 1 \text{ m}$ ,  $\theta = 0$ . Export the FRFs (with logarithmic frequency axis) of both speakers in two files in MLSSA format that can be read by Sigma Studio.
2. Open Sigma Studio, create a new project for ADAU1701 and import these two loudspeakers. Note that Sigma Studio normalizes the FRF to 0 dB (maximum of the FRF), you will need to add a gain to compensate for the normalization.
3. In Sigma Studio, design a two-way crossover for these two loudspeakers. Use Stimuli and Probe to plot the woofer path, tweeter path and sum of both. Adjust the crossovers so that the sum of the two paths has an approximately flat FRF.
4. Add a parametric equalizer to compensate for design imperfections. Use only a few filters, so don't over-complicate the parametric equalizer.
5. Save this Sigma Studio project and upload it on umtice together with the two MLSSA files and the python code.

# A Woofer characteristics



Revision:2 | Date: 2014-10-11 20:12:55

## Woofer

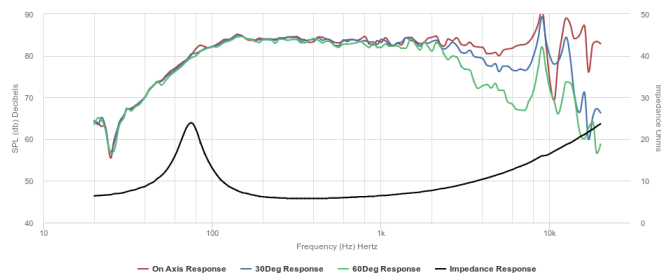
This 4 inch 8 ohm driver is a member of the high performance HDS family. A powerful ferrite magnet system is coupled to a finite element analysis designed suspension system, containing both a linear spider design and a rubber surround. The motor contains an aluminium shorting ring, which reduces coil inductance, thus providing both extended frequency response performance and reduced distortion. The long-throw voice coil ensures linear high excursion performance, needed for signal clarity. The cone necks are vented so as to reduce air compression effects under high excursion conditions. The cast aluminium basket offers structural rigidity, heat sinking capacity for the motor, and additional air venting under the spider so as to again reduce air compression effects. The cone in this model is aluminium, offering a unique visual and acoustic experience.



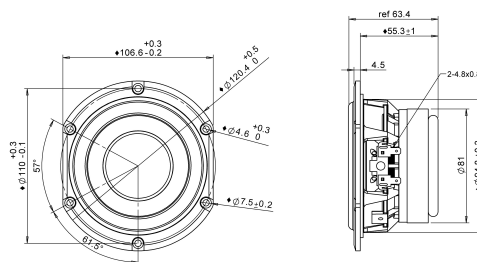
HDS-P835023

| SPECIFICATIONS                     |                  |            |             |             |                            |         |                               |       |  |
|------------------------------------|------------------|------------|-------------|-------------|----------------------------|---------|-------------------------------|-------|--|
| DC Resistance                      | Rev              | $\Omega$   | 5.32        | $\pm 5.0\%$ | Moving Mass                | Mms     | g                             | 7.3   |  |
| Minimum Impedance                  | Zmin             | $\Omega$   | 5.78        | $\pm 7.5\%$ | Suspension Compliance      | Cms     | um/N                          | 514.3 |  |
| Voice Coil Inductance              | Le               | mH         | 0.26        | -           | Effective Cone Diameter    | D       | cm                            | 8.5   |  |
| Resonant Frequency                 | fs               | Hz         | 82.18       | 15%         | Effective Piston Area      | Sd      | cm <sup>2</sup>               | 56.8  |  |
| Mechanical Q Factor                | Qms              | -          | 2.81        | -           | Equivalent Volume          | Vas     | L                             | 2.33  |  |
| Electrical Q Factor                | Qes              | -          | 0.8         | -           | Motor Force Factor         | BL      | T·m                           | 5.01  |  |
| Total Q Factor                     | Qts              | -          | 0.62        | -           | Motor Efficiency Factor    | $\beta$ | (T·m <sup>2</sup> )/ $\Omega$ | 4.7   |  |
| Ratio                              | fs/Qts           | -          | 132.33      | -           | Voice Coil Former Material | VCfm    | -                             | ASV   |  |
| Half Space Sensitivity             | dB@2.83V/1m      | dB         | 84.02       | $\pm 1.01$  | Voice Coil Inner Diameter  | VCd     | mm                            | 25.73 |  |
| Sensitivity                        | 1W/1m            | dB         | 82.6        | $\pm 1.01$  | Gap Height                 | Gh      | mm                            | 6     |  |
| Rated Noise Power (IEC 268-5 18.1) | P                | W          | 30          | -           | Maximum Linear Excursion   | Xmax    | mm                            | 3.1   |  |
| Test Spectrum Bandwidth            | 1300 Hz - 20k Hz | 12 dB/Oct  | 20Hz - 5kHz | -           | Ferrofluid Type            | FF      | -                             | -     |  |
| Energy Bandwidth Product           | EBP              | (1/Qes)·fs | -           | -           | Transducer Size            | -       | -                             | 4 in  |  |
|                                    |                  |            |             |             | Transducer Mass            | -       | Kg                            | 0.91  |  |

FREQUENCY & IMPEDANCE RESPONSE



MECHANICAL 2D DRAWING



© 2015 Tympany HK Ltd.  
Address: 1307-8 Dominion Centre, 43-59 Queen's Rd. East., Wanchai, Hong Kong  
Email: [sales@tympany.com](mailto:sales@tympany.com)

# B Tweeter characteristics



Revision:3 | Date: 2014-10-14 23:24:06

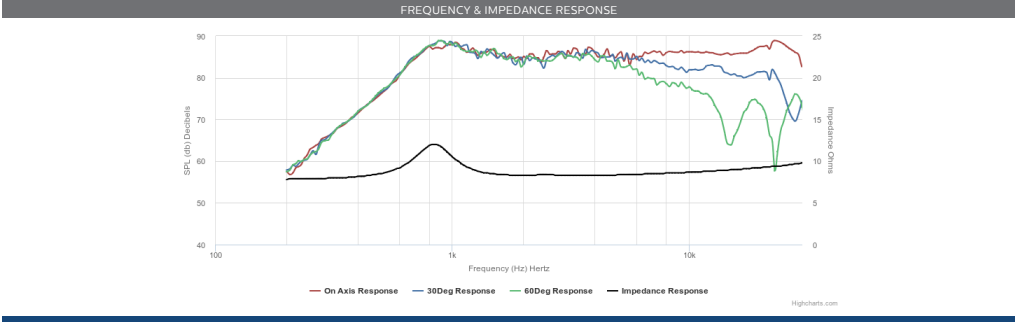
## Tweeter

This DX family tweeter features a 19 mm 8 ohm voice coil, a damped silk diaphragm with large surround roll to produce a low resonant frequency, and a ferrite magnet motor which adds heat sinking for power handling capacity. The faceplate features recessed mounting holes to minimize reflections.



DX20BF00-08

| SPECIFICATIONS                     |                  |            |              |       |                            |      |          |
|------------------------------------|------------------|------------|--------------|-------|----------------------------|------|----------|
| DC Resistance                      | Revc             | Ω          | 7.52         | ±5.0% | Moving Mass                | Mms  | g        |
| Minimum Impedance                  | Zmin             | Ω          | 8.27         | ±7.5% | Suspension Compliance      | Cms  | um/N     |
| Voice Coil Inductance              | Le               | mH         | 0.02         | -     | Effective Cone Diameter    | D    | cm       |
| Resonant Frequency                 | fs               | Hz         | 810          | 15%   | Effective Piston Area      | Sd   | cm2      |
| Mechanical Q Factor                | Qms              | -          | 2.86         | -     | Equivalent Volume          | Vas  | L        |
| Electrical Q Factor                | Qes              | -          | 4.23         | -     | Motor Force Factor         | BL   | T•m      |
| Total Q Factor                     | Qts              | -          | 1.71         | -     | Motor Efficiency Factor    | β    | (T•m2)/Ω |
| Ratio                              | fs/Qts           | -          | 474.24       | -     | Voice Coil Former Material | VCfm | -        |
| Half Space Sensitivity             | dB@2.83V/1m      | dB         | 86.12        | ±1.01 | Voice Coil Inner Diameter  | VCdi | mm       |
| Sensitivity                        | 1W/1m            | dB         | 86.3         | ±1.01 | Gap Height                 | Gh   | mm       |
| Rated Noise Power (IEC 268-5 18.1) | P                | W          | 60           | -     | Maximum Linear Excursion   | Xmax | mm       |
| Test Spectrum Bandwidth            | 1300 Hz - 20k Hz | 12 dB/Oct  | 2.5KHz-20KHz | -     | Ferrofluid Type            | FF   | -        |
| Energy Bandwidth Product           | EBP              | (1/Qes)•fs | -            | -     | Transducer Size            | -    | mm       |
|                                    |                  |            |              | -     | Transducer Mass            | -    | Kg       |



© 2015 Tymphany HK Ltd.  
Address: 1307-8 Dominion Centre, 43-59 Queen's Rd. East., Wanchai, Hong Kong  
Email: [sales@tymphany.com](mailto:sales@tymphany.com)

## C Modeling the response of a loudspeaker

We consider the loudspeaker with parameters :

- Electrical parameters
  - Coil resistance  $R_e$  [ $\Omega$ ]
  - Coil inductance  $L_e$  [H]
- Mechanical parameters
  - Mechanical compliance  $C_{ms}$  [m/N]
  - Mechanical moving mass  $M_{ms}$  [kg]
  - Mechanical resistance  $R_{ms}$  [kg/s]
- Coupling parameters
  - Coupling parameter  $B\ell$  [T.m] [N/A]
  - Effective diaphragm area  $S_d$  [m<sup>2</sup>]

The fundamental equations are

$$u = R_e i + L_e \frac{di}{dt} + B\ell \frac{dx}{dt} \quad (3)$$

$$B\ell i = K_{ms}x + R_{ms} \frac{dx}{dt} + M_{ms} \frac{d^2x}{dt^2}. \quad (4)$$

### Laplace Domain analysis

In the Laplace domain the set of equation becomes

$$U(s) = (R_e + sL_e)I(s) + sB\ell X(s) \quad (5)$$

$$B\ell I(s) = (K_{ms} + sR_{ms} + s^2M_{ms})X(s), \quad (6)$$

Taking into account the acceleration  $A(s) = s^2X(s)$ , you can derive

$$\frac{A(s)}{U(s)} = \dots \quad (7)$$

## D `scipy.signal.bilinear`

```
b, a = scipy.signal.bilinear(B, A, fs=1.0)
```

Return a digital IIR filter from an analog one using a bilinear transform. In other words, transform a set of poles and zeros from the analog s-plane to the digital z-plane.

---

|                    |                              |                                                                                            |
|--------------------|------------------------------|--------------------------------------------------------------------------------------------|
| <b>Parameters:</b> | <b>b</b> <i>[array_like]</i> | Numerator of the analog filter transfer function.                                          |
|                    | <b>a</b> <i>[array_like]</i> | Denominator of the analog filter transfer function.                                        |
|                    | <b>fs</b> <i>[float]</i>     | Sample rate, as ordinary frequency (e.g., hertz). No pre-warping is done in this function. |
| <b>Returns:</b>    | <b>z</b> <i>[ndarray]</i>    | Numerator of the transformed digital filter transfer function.                             |
|                    | <b>p</b> <i>[ndarray]</i>    | Denominator of the transformed digital filter transfer function.                           |

---

Example (NOT RELATED THE PROJECT):

```
import numpy as np
import scipy.signal as signal
import matplotlib.pyplot as plt

fs = 48e3 # sample rate
fc = 1e3 # filter cut-off frequency

# analog filter design
B, A = signal.butter(2, 2*np.pi*fc, analog=True)
b, a = signal.butter(2, fc/(fs/2))

# bilinear transform
b, a = signal.bilinear(B, A, fs=fs)

# Frequency response functions
Ws, Hs = signal.freqs(B, A)
Fs = Ws/(2*np.pi)
Wz, Hz = signal.freqz(b, a)
Fz = Wz/(2*np.pi) * fs

fig, ax = plt.subplots()
ax.semilogx(Fs, 20*np.log10(Hs), label='analog filter')
ax.semilogx(Fz, 20*np.log10(Hz), label='digital filter')
ax.set(xlabel='Frequency [Hz]', ylabel='Magnitude [dB]')
ax.legend()
```