

R. Brigola, TH Nürnberg Georg Simon Ohm, 2014

Mathematica - Notebooks als Bonusmaterial zum Lehrbuch

**[1] Rolf Brigola Fourier-Analysis und Distributionen,
Eine Einführung mit Anwendungen,
edition swk, Hamburg 2013**

Teil 6 Entwurf von analogen Butterworth-Filtern mit *Mathematica* und Realisierung mit aktiven Sallen-Key-Schaltkreisen

Referenzen zu Kapiteln, Abschnitten, Seitenzahlen beziehen sich im Folgenden meist auf das genannte Lehrbuch des Autors. Einige wenige weitere Referenzen findet man am Ende des Notebooks.

In diesem Notebook wird zunächst gezeigt, wie man ein analoges Butterworth-Tiefpassfilter zu einem vorgegebenen Dämpfungsplan entwirft. *Dabei folgen wir der Beschreibung in [1], Kap.10S. 279-282.* Im Anschluss daran zeigen wir, wie man solche Filter mit aktiven Schaltkreisen realisieren kann.

Der Entwurf von Tiefpass-Filtern ist eine Grundaufgabe in der Schaltungstechnik, weil man aus den Frequenzgängen von Tiefpass-Filtern auch Hochpass-, Bandpass- oder Bandsperren-Filter entwickeln kann (vgl. [1], S.287 oder andere Referenzen wie [2]-[4]).

Aus einem analogen Tiefpassfilter kann man *durch bilineare Transformation oder andere Verfahren auch ein diskretes Filter* mit einem entsprechenden Frequenzgang gewinnen. Ich gehe auf diskrete Filter in einem späteren Notebook näher ein.

Wir beginnen damit, analoge Butterworth-Filter zunächst eigenständig zu berechnen und sehen am Ende, was die hierfür schon vorhandenen *Mathematica*-Routinen im Vergleich ergeben.

Anmerkung des Autors: Wie schon in vorangehenden Notebooks bemerkt, bin ich kein Experte für die schier unerschöpflichen Möglichkeiten, die ein wirklich geübter, vertiefter Umgang mit *Mathematica* bietet. Ich habe mich daher im Wesentlichen bemüht, den Stoff mit diesem Angebot zu veranschaulichen und zu zeigen, wie man mit (zum Teil wahrscheinlich oft zu umständlichen, dafür aber auch für *Mathematica*-Anfänger wie mich selbst transparenten) Anweisungen die behandelten Inhalte erschließen kann.

Die URL aller meiner *Mathematica*-Notebooks zur Fourier-Analysis ist:

<http://www.stiftung-swk.de/mathematica/>

Butterworth – Tiefpassfilter

1. Frequenzgang zu gegebenem Dämpfungsplan

Beim Butterworth-Tiefpassfilter soll das Quadrat des Amplitudengangs zum rationalen Frequenzgang $\hat{h}(\omega)$ mit der Kreisfrequenz ω unterhalb der Grenzkreisfrequenz ω_g möglichst lange horizontal verlaufen. Da dort $|\omega/\omega_g| < 1$ ist, wird diese Forderung am besten erfüllt, wenn $|\hat{h}|^2$ nur von der höchsten Potenz von ω/ω_g abhängt. Für $|\omega/\omega_g| < 1$ liefern nämlich die niedrigen Potenzen von ω/ω_g große Beiträge zum Nenner von $|\hat{h}|^2$, bewirken also eine verstärkte Amplitudenverzerrung.

Für ein Butterworth-Tiefpassfilter der Ordnung n wird $|\hat{h}|^2$ daher definiert durch

$$(1) \quad |\hat{h}(\omega)|^2 = \frac{K}{1+(\omega/\omega_g)^{2n}}.$$

Dabei ist ω_g die gewünschte *Grenzkreisfrequenz* für das sog. Passband (cutoff angular frequency).

Die Ordnung n und die Grenzkreisfrequenz ω_g sind aus einem Dämpfungsplan, den das Filter erfüllen soll, zu berechnen. Damit folgen dann die Filterkoeffizienten. Natürlich soll die Ordnung möglichst niedrig sein, da Bauteile zur Realisierung ja etwas kosten.

Um konkret zu sein, geben wir folgenden Dämpfungsplan vor:

Mit $A = -20 \log_{10}(|\hat{h}(\omega)|)$ als Dämpfungsmaß in dB geben wir für das gesuchte Filter folgende Werte vor:

Gleichspannungsverstärkung $K=1$

Passbandeckung $\omega_1=2\pi * 3000$ [1/s]

Stopbanddecke $\omega_2 = 2\pi * 5000$ [1/s]

Minimale Passbandverstärkung $h_1 = 0.891251$, d.h. maximale Dämpfung 1 dB im Passband, maximale Stopbandverstärkung $h_2 = 0.1$, d.h. mindestens Dämpfung 20 dB im Stopband.

Mit diesen Vorgaben berechnen wir ω_g , die benötigte Ordnung n des Butterworthfilters und seinen Frequenzgang wie in [1] dargestellt:

1) Man berechnet bei gegebenen ω_g und n die Nullstellen $z[1], \dots, z[n]$ von

$$(*) \quad Q(z) = 1 + (-1)^n \left(\frac{z}{\omega_g} \right)^{2n} = 1 + \left(\frac{z}{i\omega_g} \right)^{2n} \quad (i^2 = -1)$$

mit negativen Realteilen und bildet damit den Frequenzgang \hat{h} des gewünschten Butterworth-Tiefpassfilters

$$\hat{h}(\omega) = \frac{K \omega_g^n}{\prod_{k=1}^n (i\omega - z[k])}$$

2) Mit den Nullstellen $z[1], \dots, z[n]$ von Q , die negative Realteile haben, gilt dann äquivalent mit dem sog. **Butterworth-Polynom** $B_n(s) = \prod_{k=1}^n (s - z[k]/\omega_g)$ vom Grad n

$$\hat{h}(\omega) = \frac{K}{B_n(i\omega/\omega_g)}$$

Für unser Beispiel ergibt sich aus den geforderten Bedingungen nun durch Rechnung (für mehr Details vgl. man ggf. [1], S. 281)

In[400]=

```
SetDirectory["D:/Mathematica"] (*User-spez. Verzeichnis*)
```

Out[400]= D:\Mathematica

In[401]= ClearAll["Global`*"]; Remove["Global`*"]; ?Global`*

Information: No symbol matching Global`* found.

```
In[402]:= K := 1;  $\omega_1 = 2 \pi 3000$ ;  $\omega_2 = 2 \pi 5000$ ; h1 := 0.9; h2 := 0.1; (*hier die Vorgaben*)
ord := N[Log[(K^2/h1^2 - 1)/(K^2/h2^2 - 1)]/(2 * Log[ $\omega_1 / \omega_2$ ])]
n = Floor[ord] + 1 (* n = benötigte Filterordnung *)
 $\omega_g = \text{Sqrt}[\omega_1 \text{Exp}[-\text{Log}[K^2/h1^2 - 1]/(2 n)]]$   $\omega_2 \text{Exp}[-\text{Log}[K^2/h2^2 - 1]/(2 n)]$ 
 $f_g = \omega_g / (2 \pi)$  (*  $2\pi f_g = \omega_g = \text{Grenzkreisfrequenz im Frequenzgang } \hat{h}$  *)
```

Out[404]= 6

Out[405]= 21345.8

Out[406]= 3397.29

In[407]=

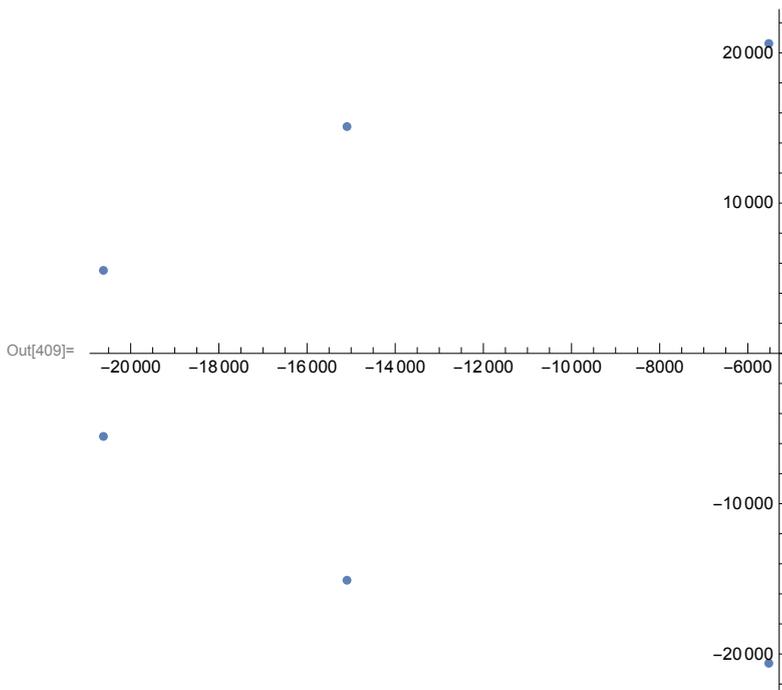
1.1 Berechnung der Polstellen für die Übertragungsfunktion, Übertragungsfunktion und Frequenzgang

Die Lösungen $z[1] \dots z[n]$ der Kreisteilungsgleichung (*) mit negativen Realteilen sind (Übungsaufgabe zum Rechnen mit komplexen Zahlen, alternativ mit Solve oder NSolve in *Mathematica* exakt oder numerisch zu lösen)

```
In[408]:= z = Table[ $\omega_g \text{Exp}[i((2k - 1)\pi/(2n) + \pi/2)]$ , {k, 1, n}]
```

```
Out[408]= {-5524.7 + 20618.5 i, -15093.8 + 15093.8 i, -20618.5 + 5524.7 i,
-20618.5 - 5524.7 i, -15093.8 - 15093.8 i, -5524.7 - 20618.5 i}
```

```
In[409]:= ListPlot[{Tooltip[{Re[#1], Im[#1]}] &} /@ z, AspectRatio -> 1]
```



Diese Nullstellen von Q sind die Polstellen der Übertragungsfunktion des Filters. Ihre Lage zeigt, dass das Filter stabil ist.

Wir halten fest, dass bei ungerader Ordnung ein Pol reell ist, die anderen treten zusammen mit den konjugiert komplexen Polstellen auf. Bei gerader Filterordnung kommt kein reeller Pol vor. Die resultierende rationale Übertragungsfunktion hat also im Nenner *ein Polynom mit reellen Koeffizienten, was für eine Realisierung ja auch wichtig ist*. Sie haben sämtlich gleiche Vorzeichen (eine notwendige Bedingung für die Stabilität des Filters).

Wir geben nun die Übertragungsfunktion H(s) und damit den Frequenzgang \hat{h} unseres Butterworth-Tiefpassfilters an:

$$\hat{h}(\omega) = H(i\omega)$$

```
In[410]:= P[s_] := Product[Expand[(s - z[[k]]) (s - Conjugate[z[[k]])], {k, 1, n/2}];
If[EvenQ[n], H[s_] = K ω_g^n / P[s], H[s_] = K ω_g^n / (P[s] (s - z[[ (n + 1) / 2 ]])];
h[ω_] = H[iω]
```

```
Out[412]= 9.45969 × 1025 / ( ((4.55644 × 108 + 0. i) + (11 049.4 + 0. i) iω + iω2)
  ((4.55644 × 108 + 0. i) + (30 187.5 + 0. i) iω + iω2)
  ((4.55644 × 108 + 0. i) + (41 237. + 0. i) iω + iω2) )
```

```
In[413]:= Product[-z[[m]], {m, 1, n}] / ω_g^n
(* das Produkt der (-z[[k]]) für k=1..n ergibt ω_g^n *)
```

```
Out[413]= 1. + 0. i
```

Damit ist der Frequenzgang so angegeben, dass wir eine Realisierungsmöglichkeit durch eine Serienschaltung von Tiefpassfiltern 2. Ordnung (sog. Biquads) sofort sehen. Aus den Koeffizienten kann eine realisierende Anlogschaltung direkt abgeleitet werden. Wir gehen später noch darauf ein.

Bemerkung: *Der Frequenzgang eines stabilen kausalen linearen Filters, dessen Übertragungsverhalten "im Zeitbereich" durch eine lineare Differentialgleichung mit konstanten Koeffizienten beschrieben wird, ist die **Fouriertransformierte \hat{h} der (kausalen) Impulsantwort h des Filters** (LTI-Systems). Umgekehrt kann man zu einer rationalen Übertragungsfunktion eine solche Differentialgleichung angeben, aus deren Koeffizienten die schaltungstechnische Realisierung abgeleitet werden kann.*

Das Nennerpolynom der Übertragungsfunktion ist dann gerade das charakteristische Polynom der Differentialgleichung, ein ggf. vorhandenes Zählerpolynom gehört zu einem linearen Differentialoperator auf der rechten Seite der Differentialgleichung mit den Koeffizienten dieses Zählerpolynoms. Mehr dazu in meinen Notebooks zur Fouriertransformation und zeitinvarianten linearen Systemen unter der gleichen URL wie dieses Material (vgl.auch [1], Kap. 9-10).

1.2 Grafische Darstellungen des Amplitudengangs, des Phasengangs, der Verzögerung (Delay) und der Gruppenlaufzeit (Group Delay)

Zunächst zum Test ein paar Werte: $\text{Abs}[H[0]]$ dc-gain, $-20\text{Log}[10,\text{Abs}[H[i \omega_1]]]$ Dämpfung an der Passbandecke, $-20\text{Log}[10,\text{Abs}[H[i \omega_2]]]$ Dämpfung an der Stopbandecke, $-20\text{Log}[10,\text{Abs}[H[i \omega_g]]]$ bei der Grenzkreisfrequenz, alle in dB gemessen. Danach der Amplitudengang für Kreisfrequenzen $\omega \in [0, 4 \omega_g]$. Wegen der Symmetrie $|H(i\omega)|=|H(-i\omega)|$ genügt es, den Amplitudengang nur für positive ω darzustellen.

```
In[414]:= Abs[H[0]] (* Gleichspannungsverstärkung, "dc-gain" *)
-20 Log[10, Abs[H[i ω1]]] (* Dämpfung in dB an der Passbandecke *)
-20 Log[10, Abs[H[i ω2]]] (* an der Stopbandecke *)
-20 Log[10, Abs[H[i ωg]]] (* und bei der Grenzkreisfrequenz *)
```

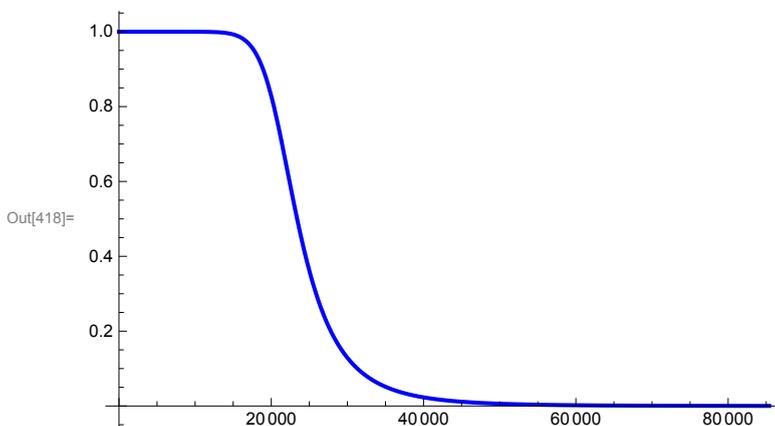
Out[414]= 1.

Out[415]= 0.880768

Out[416]= 20.1823

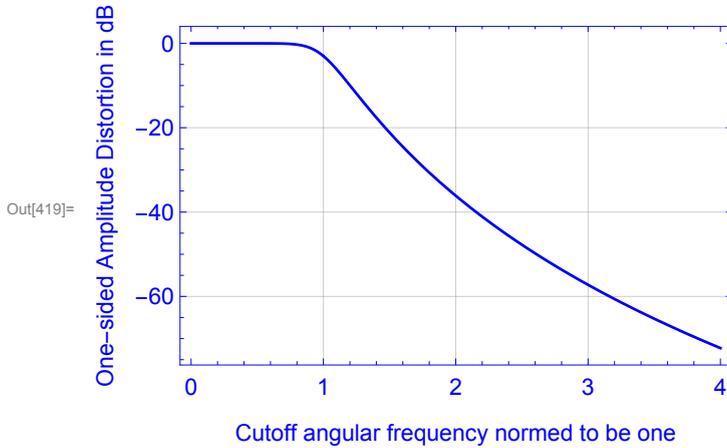
Out[417]= 3.0103

```
In[418]:= p1 = Plot[Abs[H[i ω]], {ω, 0, 4 ωg},
PlotStyle → Directive[Blue, Thickness[0.006]], PlotRange → All]
```



In der Ingenieurliteratur wird die Amplitudenverzerrung meist so angegeben, dass die Grenzkreisfrequenz auf Eins normiert wird und die Verzerrung in Dezibel gezeigt wird. Diese Darstellung daher hier wie folgt. Sie zeigt sofort, dass das Filter die gewünschte Spezifikation erfüllt.

```
In[419]:= p2 = Plot[20 Log[10, Abs[H[i s ωg]]], {s, 0, 4}, Frame → True,
  FrameStyle → Directive[Blue, FontSize → 12, FontWeight → Plain],
  FrameLabel -> {"One-sided Amplitude Distortion in dB", ""},
  {"Cutoff angular frequency normed to be one", ""}},
  GridLines → Automatic, PlotStyle → {Blue, Thickness[0.005]}, PlotRange → All]
```



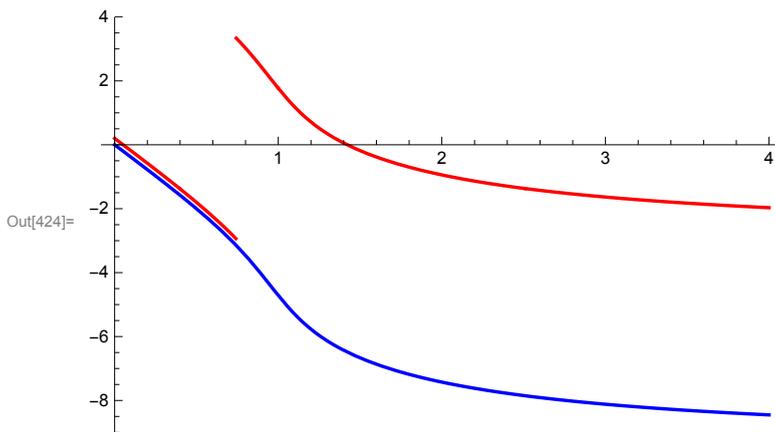
Nun der Phasengang, wieder für Kreisfrequenzen $\omega \in [0,4 \omega_g]$ und ω_g auf Eins normiert.

Einmal mit phi1 mit einem Phasensprung um 2π , danach bei der Darstellung mit phi die beiden Zweige zu einer stetigen Funktion zusammengefügt. (Die Funktion phi1 (rot) ist mit einem offset von +0.2 zur besseren Sichtbarkeit gezeichnet, phi ist blau.)

Dabei benutzen wir die Darstellung des Nenners von $H[i\omega]$ als Produkt von Linearfaktoren der Form $(i\omega - z[[k]])$, $k=1,\dots,n$. (Der Zähler hat dann das Argument Null.)

```
In[420]:= phi1[ω_] := Assuming[Element[ω, Reals], Arg[H[i ω]]] (* mit Phasensprung *)
phi[ω_] = -Sum[Arg[i ω - z[[k]]], {k, 1, n}] (* ohne Phasensprung *)
p3 := Plot[phi[s ωg], {s, 0, 4},
  PlotStyle → Directive[Blue, Thickness[0.005]], PlotRange → All]
p4 := Plot[phi1[s ωg] + 0.2, {s, 0, 4},
  PlotStyle → Directive[Red, Thickness[0.005]], PlotRange → All]
Show[
  p3,
  p4]
```

```
Out[421]= -Arg[(5524.7 - 20618.5 i) + i ω] - Arg[(5524.7 + 20618.5 i) + i ω] -
Arg[(15093.8 - 15093.8 i) + i ω] - Arg[(15093.8 + 15093.8 i) + i ω] -
Arg[(20618.5 - 5524.7 i) + i ω] - Arg[(20618.5 + 5524.7 i) + i ω]
```



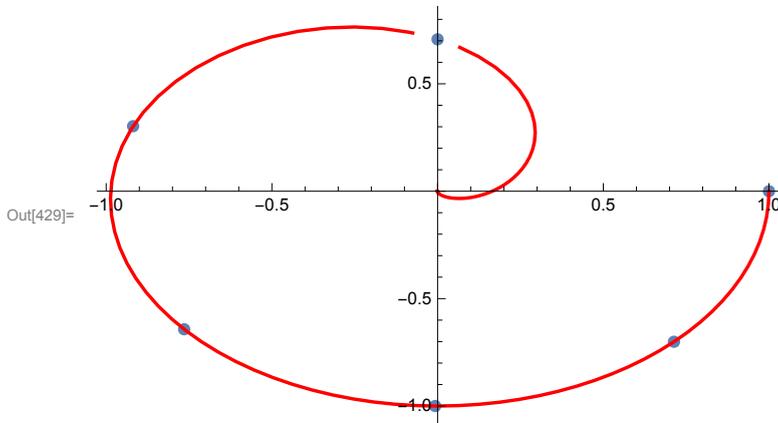
```
In[425]:= phi1[ωg]
phi[ωg]
```

```
Out[425]= 1.5708
```

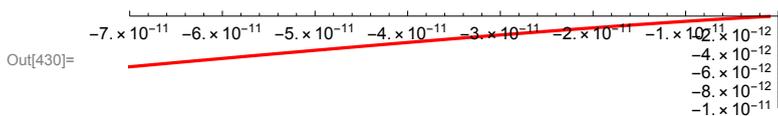
```
Out[426]= -4.71239
```

In der Ingenieurliteratur findet man auch häufig die Darstellung des Frequenzgangs als Ortskurve in der komplexen Ebene. Bei einem **stabilen** Tiefpass-Filter beginnt sie für $\omega=0$ auf der positiven Halbachse und dreht sich **im Uhrzeigersinn für $\omega \rightarrow \infty$ zum Nullpunkt, wobei sie genau so oft Quadranten durchläuft wie die Ordnung des Filters angibt** (hier $n=6$; ein Zoom für große ω zeigt, dass sich die Ortskurve von links unten aus Südwest dem Nullpunkt nähert). Die Punkte auf der Kurve zeigen die komplexen Frequenzgang-Werte für einige ω an. Die geringen Betrags- und Phasenverzerrungen des Filters bis in die Nähe der Grenzkreisfrequenz sind auch hier gut zu erkennen.

```
In[427]:= ortskurve := ParametricPlot[{Re[H[i s ωg]], Im[H[i s ωg]]},
  {s, 0, 50}, PlotStyle → Directive[Red, Thickness[0.005]]];
punkte := Table[{Re[H[i s ωg]], Im[H[i s ωg]]}, {s, 0, 1, 0.2}];
Show[{ListPlot[punkte, PlotStyle → PointSize[.018]], ortskurve}, PlotRange → All]
```



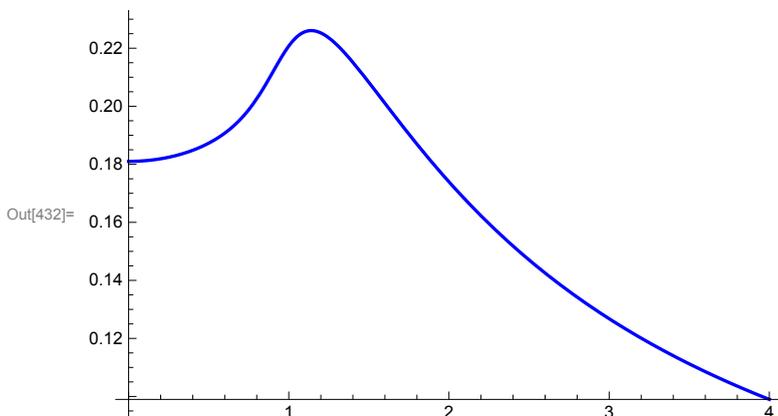
```
In[430]:= ParametricPlot[{Re[H[i s ωg]], Im[H[i s ωg]]},
  {s, 49, 100}, PlotStyle → Directive[Red, Thickness[0.005]],
  PlotRange → {{-7 × 10-11, 0}, {-10-11, 0}}]
(* Die Ortskurve für sehr hohe Frequenzen *)
```



Nun die **Verzögerung $-\text{phi}[\omega]/\omega$ in ms (phase delay)** für Kreisfrequenzen $\omega \in [0,4 \omega_g]$ mit Normierung wie oben.

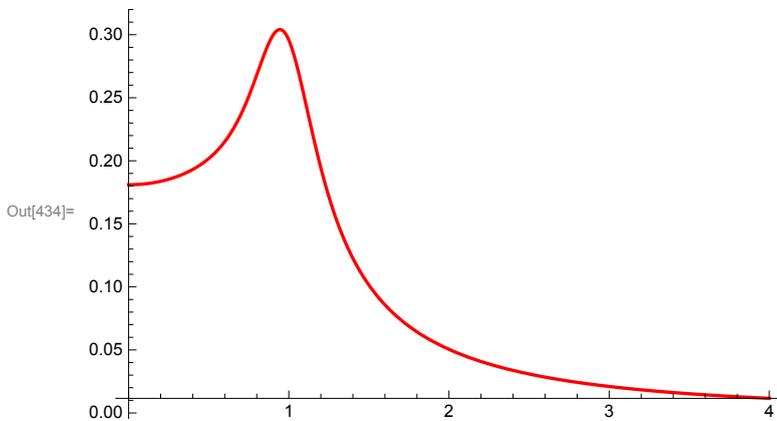
Unterschiedliche Verzögerungen im Frequenzband von Audio-Aufnahmen ab etwa 1 ms empfinden wir bereits als reichlich störend - ein gutes Gehör vorausgesetzt. Butterworth-Filter ergeben wie nachfolgend erkennbar gute Ergebnisse.

```
In[431]:= delayms[ω_] := -1000 phi[ω] / ω
p5 = Plot[delayms[s ωg], {s, 0, 4},
  PlotStyle → Directive[Blue, Thickness[0.005]], PlotRange → All]
```



Und schließlich die **Gruppenlaufzeit $-D[\text{phi}]$ in ms (group delay)** für $\omega \in [0,4 \omega_g]$ in Hz. Hier das Ergebnis:

```
In[433]:= groupdelay[w_] := -Sum[1 / (Re[z[[k]]] + (w - Im[z[[k]])2 / Re[z[[k]]]), {k, 1, n}]
p6 = Plot[1000 groupdelay[s ωg], {s, 0, 4},
PlotStyle → Directive[Red, Thickness[0.005]], PlotRange → All]
```



Zur Veranschaulichung der Verzögerung betrachten wir als Input-Signal $\sin(\omega t)$ für $\omega=200 \pi$ und plotten das gefilterte Signal.

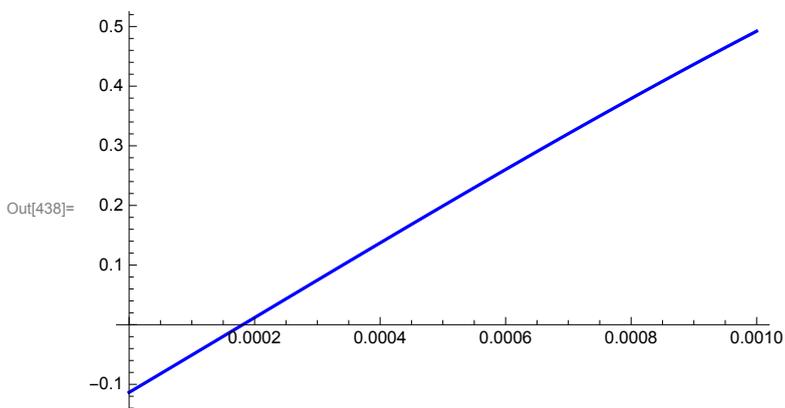
```
In[435]:= w := 200 π;
input[t_] := 1 / (2 i) (Exp[i w t] - Exp[-i w t]);
(* Sinus-input als komplexe Fourierreihe *)
output[t_] =
FullSimplify[Expand[1 / (2 i) (H[i w] Exp[i w t] - H[-i w] Exp[-i w t])], Trig]
```

Out[436]= $-0.113496 \cos[200 \pi t] + 0.993538 \sin[200 \pi t]$

```
In[437]:= delaysms[w] (* Berechnete Verzögerung in ms für die 100 HZ-Schwingung *)
```

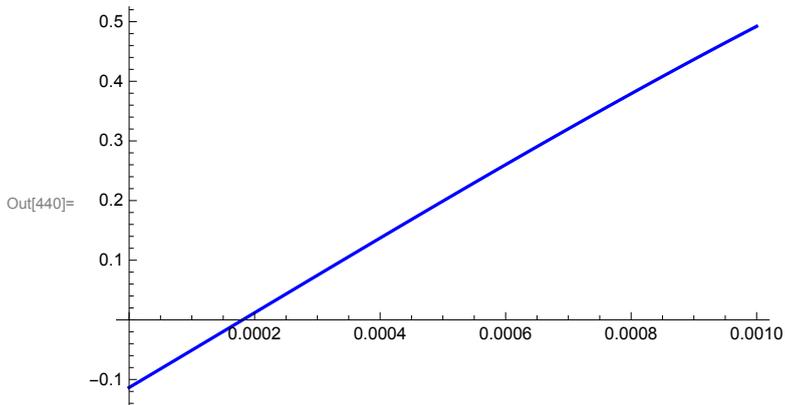
Out[437]= 0.181024

```
In[438]:= Plot[output[t], {t, 0, 0.001}, PlotStyle → Directive[Blue, Thickness[0.005]]]
(* hier ist die Verzögerung der Sin-Schwingung am Output zu sehen *)
```



Das gleiche Ergebnis erhält man natürlich mit

```
In[439]:= output2[t_] = Sin[w (t - delays[w] / 1000)];
Plot[output2[t], {t, 0, 0.001}, PlotStyle -> Directive[Blue, Thickness[0.005]]]
```



1.3 Mathematica Transfer Function Models

Wegen der großen Bedeutung linearer Filter in der gesamten Übertragungstechnik, welche heute viele Lebensbereiche durchdringt, bietet *Mathematica* eine ganze Reihe sog. *Transfer Function Models* an, mit denen ganz unterschiedliche Filtertypen, und zwar sowohl analoge als auch diskrete Filter mit gewünschten Eigenschaften schnell berechnet werden können. Man siehe hierzu die Webseiten

<https://reference.wolfram.com/mathematica/tutorial/AnalogFilterDesign.html>

<https://reference.wolfram.com/mathematica/tutorial/DigitalFilterDesign.html>

Wir betrachten nur kurz als Beispieltyp unser Butterworth-Tiefpassfilter mit den verwendeten Passband- und Stopband-Ecken sowie den dort im Ergebnis berechneten Dämpfungswerten. Das *Mathematica*-Modell ergibt dann die gleiche Übertragungsfunktion:

```
In[441]:= bf = ButterworthFilterModel[
  {"Lowpass", {ω1, ω2}, {-20 Log[10, Abs[H[i ω1]]], -20 Log[10, Abs[H[i ω2]]]}]
```

Out[441]=
$$\left(\frac{9.45969 \times 10^{25}}{\left((5524.7 - 20618.5 i) + s \right) \left((5524.7 + 20618.5 i) + s \right) \left((15093.8 - 15093.8 i) + s \right) \left((15093.8 + \right.} \right.$$

Zum Vergleich noch einmal die übereinstimmenden Pole unserer oben berechneten Übertragungsfunktion

```
In[442]:= z
```

Out[442]=
$$\{-5524.7 + 20618.5 i, -15093.8 + 15093.8 i, -20618.5 + 5524.7 i, \\ -20618.5 - 5524.7 i, -15093.8 - 15093.8 i, -5524.7 - 20618.5 i\}$$

In[443]:=

2. Zur Realisierung von Tiefpassfiltern in der Schaltungstechnik

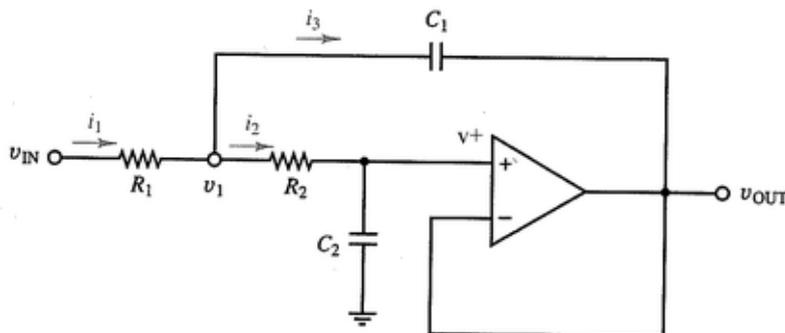
Eine Realisierungsmöglichkeit eines wie oben besprochenen Butterworth-Tiefpassfilters ist je nach Filterordnung die *Kaskadierung mehrerer Filter 2. Ordnung (Biquads) und dazu in Serie ggf. ein Tiefpass erster Ordnung*. Das Filter erster Ordnung kann man z.B. passiv als RC-Tiefpass realisieren mit Frequenzgang $\hat{h}_1(\omega) = 1 / (RC i\omega + 1)$, etwa durch einen RC-Spannungsteiler.

Eine typische aktive Schaltung für ein Tiefpass-Biquad-Filter mit "dc-gain" 1 ist ein sog. **Sallen-Key-Tiefpassfilter zweiter Ordnung** mit Operationsverstärker wie nachfolgend gezeigt (vgl. auch [5]):

In[444]:=

```
biquad = Import["Sallen-Key-Biquad.png", ImageSize -> Large]
```

Out[444]:=



Zu lösen sind:

- 1) Wie lauten die Übertragungsfunktion bzw. der Frequenzgang der Schaltung?
- 2) Wie sind die Widerstände und Kapazitäten zu wählen, um ein Filter 2. Ordnung in einer Kaskadierung für ein Butterworth-Tiefpassfilter zu erhalten?

1) Zur Berechnung des Frequenzgangs bemühen wir komplexe Wechselstromrechnung:

Mit den komplexen Impedanzen $Z_k = 1/(i\omega C_k)$ und den üblichen Bezeichnungen der Wechselstromrechnung mit Großbuchstaben erhalten wir

$$(1) \quad I_1 = I_2 + I_3$$

$$(2) \quad \frac{V_{IN} - V_1}{R_1} = \frac{V_1}{R_2 + Z_2} + \frac{V_1 - V_{OUT}}{Z_1}$$

Die Spannung v_+ am Operationsverstärker findet man wie bei einem Spannungsteiler mittels v_1

$$(3) \quad V_+ = V_1 \frac{Z_2}{R_2 + Z_2}$$

Der "op-amp" ist gegengekoppelt, d.h. im eingeschwungenen Zustand wird die Eingangsspannungsdifferenz des op-amp Null, also gilt dann $v_{OUT} = v_+$ (Spannungsfolger, engl. "voltage follower";). Aus (3) folgt durch Umstellung also

$$(4) \quad V_1 = V_{OUT} \frac{R_2 + Z_2}{Z_2}$$

Dies eingesetzt in (2) und dann mit etwas Geduld aufgelöst nach V_{OUT} ergibt:

$$(5) \quad V_{OUT} = V_{IN} \frac{Z_1 Z_2}{Z_1 Z_2 + Z_1(R_1 + R_2) + R_1 R_2}$$

Wir erhalten daraus nun den gesuchten Frequenzgang (*Leser sollten nochmal nachrechnen*)

$$(6) \quad \hat{h}(\omega) = V_{OUT} / V_{IN} = \frac{1}{1 + i\omega C_2(R_1 + R_2) - \omega^2 R_1 R_2 C_1 C_2}$$

2) Zur Berechnung der Widerstände bei gegebenen Kapazitäten

In der Praxis wird man für jedes verwendete Biquad-Filter die Kapazitäten C_1 und C_2 vorgeben und dann die Widerstände für die op-amp-Beschaltung durch Koeffizientenvergleich mit dem Frequenzgang des berechneten Filtertyps ermitteln. Wir betrachten zum Abschluss als einfaches Beispiel die Berechnung für ein Butterworth-Tiefpassfilter der Ordnung 3 und Gleichspannungsverstärkung 1:

Der Nenner des Filter-Frequenzgangs eines Butterworth-Tiefpasses der Ordnung 3 ist $(1 + s_n)(1 + s_n + s_n^2)$ mit $s_n = i\omega/\omega_g$. Durch **Koeffizientenvergleich** folgen die gesuchten Widerstände bei gegebenen Kapazitäten C für den ersten Linearfaktor, C_1, C_2 wie oben im Schaltungsbild. Wir bezeichnen nachfolgend nun die Grenzkreisfrequenz mit $\Omega = \omega_g$:

Für den ersten Linearfaktor bei Frequenzgang $1/(1 + \Omega RC s)$ also $R = 1/(\Omega C)$. Wir setzen $C_1 \geq 4 C_2$, um reelle Lösungen R_1, R_2 beim Koeffizientenvergleich zu erhalten. Wegen der Symmetrie des Frequenzgangs in den Variablen R_1 und R_2 ergeben sich zwei mögliche Paare von Lösungen R_1, R_2 .

In[445]= Assuming [$C_1 \geq 4 C_2$ && $C_2 > 0$ && $\Omega > 0$,

Simplify[Solve [$\Omega C_2 (R_1 + R_2) == 1$ && $\Omega^2 R_1 R_2 C_1 C_2 == 1$, $\{R_1, R_2\}$]]]

$$\text{Out[445]= } \left\{ \left\{ R_1 \rightarrow -\frac{-1 + \sqrt{1 - \frac{4C_2}{C_1}}}{2 \Omega C_2}, R_2 \rightarrow \frac{1 + \sqrt{1 - \frac{4C_2}{C_1}}}{2 \Omega C_2} \right\}, \left\{ R_1 \rightarrow \frac{1 + \sqrt{1 - \frac{4C_2}{C_1}}}{2 \Omega C_2}, R_2 \rightarrow -\frac{-1 + \sqrt{1 - \frac{4C_2}{C_1}}}{2 \Omega C_2} \right\} \right\}$$

Etwas allgemeiner: Ein quadratischer Faktor im Nenner eines Tiefpass-Frequenzgangs hat unabhängig vom Filtertyp (etwa Butterworth- oder Tschebyscheff- oder Bessel-Filter) die Form $P(s) := 1 + a s + b s^2$;

Bei dem Sallen – Key – Biquad mit voll gegengekoppeltem "op-amp" und "dc-gain" 1 wie oben ergeben sich dann die Widerstände bei gegebenen Kapazitäten C_1 und C_2 mit

der Bedingung $C_1 \geq \frac{4b}{a^2} C_2$ durch

In[446]= Assuming [$a > 0$ && $b > 0$ && $C_2 > 0$ && $C_1 \geq 4 b C_2 / a^2$ && $\Omega > 0$,

Simplify[Solve [$\Omega C_2 (R_1 + R_2) == a$ && $\Omega^2 R_1 R_2 C_1 C_2 == b$, $\{R_1, R_2\}$]]]

$$\text{Out[446]= } \left\{ \left\{ R_1 \rightarrow \frac{a - \sqrt{a^2 - \frac{4bC_2}{C_1}}}{2 \Omega C_2}, R_2 \rightarrow \frac{a + \sqrt{a^2 - \frac{4bC_2}{C_1}}}{2 \Omega C_2} \right\}, \left\{ R_1 \rightarrow \frac{a + \sqrt{a^2 - \frac{4bC_2}{C_1}}}{2 \Omega C_2}, R_2 \rightarrow \frac{a - \sqrt{a^2 - \frac{4bC_2}{C_1}}}{2 \Omega C_2} \right\} \right\}$$

Verwendet man für Realisierungen andere Bauteile, dann kann man nach Berechnung des zugehörigen Frequenzgangs die Beschaltung ganz analog durch Koeffizientenvergleiche mit den Filterpolynomen erhalten.

Abgeleitete Hochpass-, Bandpass- und Bandsperren-Filter

1. Tiefpass - Hochpass - Transformation

Wir betrachten noch ein Beispiel der sog. Tiefpass-Hochpass-Transformation, um zu sehen, wie man aus dem oben berechneten Butterworth-Tiefpass-Frequenzgang den Frequenzgang eines Hochpass-Filters mit der gleichen Grenzkreisfrequenz ω_g erhält.

Wir notieren das Butterworth-Polynom der Ordnung 6 von oben durch $B[n,s]$ (s steht für den normierten Parameter $s = i\omega/\omega_g$, $n=6$). Der Frequenzgang HP des abgeleiteten Hochpassfilters ist dann $HP[\omega] = 1/B[n, -i\omega_g/\omega]$, d.h. der normierte Parameter s wird transformiert in $1/s$. Die Pole z_k der Tiefpass-Übertragungsfunktion werden dabei transformiert in die Pole $\omega_g^2/z_k = \omega_g^2 \bar{z}_k / |z_k|^2$ der Hochpass-Übertragungsfunktion. Wegen der Symmetrie der z_k bzgl. der reellen Achse (siehe oben) liegen die ω_g^2/z_k im Gesamtbild wieder auf denselben Strahlen verteilt wie die Pole z_k , haben also insbesondere negative Realteile, d.h. das gewonnene Hochpassfilter ist stabil. Für die Pole z_k von Butterworth-Filtern gilt $\omega_g = |z_k|$, die Pole des gewonnenen Hochpass-Filters sind daher \bar{z}_k . Das Hochpass-Filter hat für diesen Fall dann die

gleiche Gruppenlaufzeit wie das Tiefpass-Filter.

Ergänzung: Für einen Tiefpass-Filteranteil erster Ordnung der Form $1/(1+as)$ ergibt diese Transformation die rationale Funktion $1/(1+a/s)=s/(s+a)$, für einen "Biquad-Anteil" der Form $1/(1+bs+cs^2)$ dann $1/(1+b/s+cs^2) = s^2/(s^2+bs+c)$. Bei positiven Koeffizienten a,b,c eines stabilen Tiefpassfilters folgt im Produkt solcher Terme dann wieder eine Übertragungsfunktion $HP(s)$ eines stabilen Hochpassfilters mit positiven Koeffizienten. Die Filterordnung bleibt bei der Tiefpass-Hochpass-Transformation erhalten. Die zugehörige Übertragungsfunktion ist die *Laplace-Transformierte der kausalen Impulsantwort einer Differentialgleichung der Form $P(D)u=Q(D)f$* . Die charakteristischen Polynome zu den Differentialoperatoren P bzw. Q sind dann das Nenner- bzw. Zählerpolynom von HP . **Der Frequenzgang ist die Fouriertransformierte der kausalen Impulsantwort der Differentialgleichung (vgl. [1], Kap. 10), d.h. der eindeutig bestimmten kausalen Lösung von $P(D)h=Q(D)\delta$, wobei δ die Dirac-Distribution ist.**

Wir plotten den Amplituden- und den Phasengang.

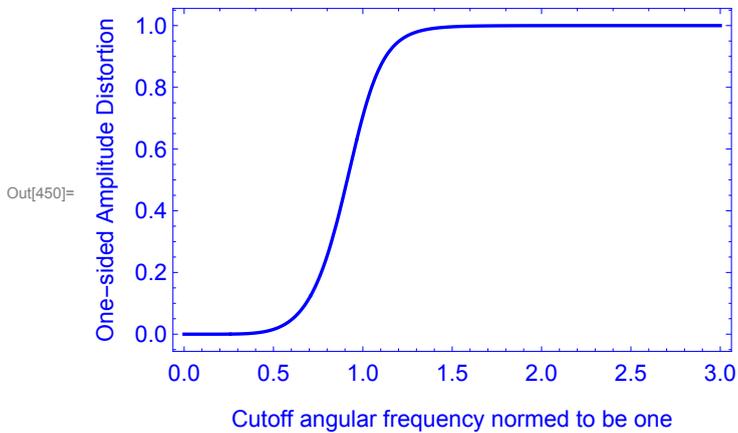
```
In[447]:= B[n_, s_] =  $\prod_{k=1}^{n/2} \text{Expand}[(s - z[[k]]/\omega_g)(s - \text{Conjugate}[z[[k]])/\omega_g]$ 
(* unser Butterworth-Polynom vom geraden Grad 6 *)

Out[447]=  $\frac{((1. + 0. i) + (0.517638 + 0. i) s + s^2)((1. + 0. i) + (1.41421 + 0. i) s + s^2)((1. + 0. i) + (1.93185 + 0. i) s + s^2)}$ 

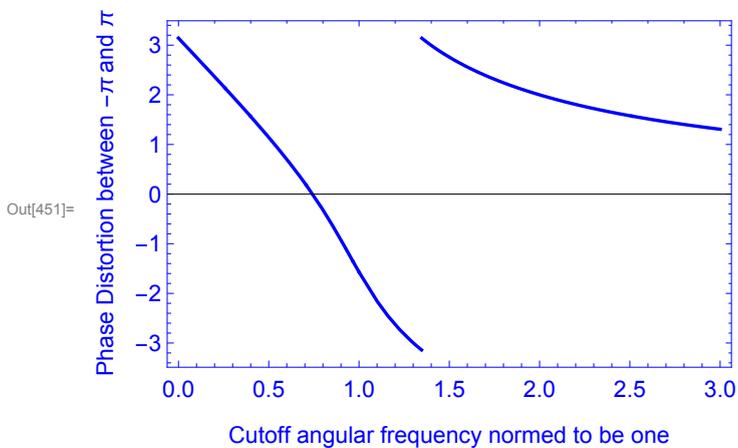
In[448]:= HP[\omega_] = Factor[1/B[n, -i \omega_g / \omega]]
(* Der mit der Tiefpass-Hochpass-Transformation erzeugte Hochpass-Frequenzgang *)
phiHP[\omega_] := Assuming[Element[\omega, Reals], Arg[HP[\omega]]];
(* und der zugehörige Phasengang *)

Out[448]=  $\frac{(1. + 0. i) \omega^6}{(((-20618.5 - 5524.7 i) + (1. + 0. i) \omega)((-15093.8 - 15093.8 i) + (1. + 0. i) \omega))(((-5524.7 - 20618.5 i) + (1. + 0. i) \omega)((5524.7 - 20618.5 i) + (1. + 0. i) \omega))((15093.8 - 15093.8 i) + (1. + 0. i) \omega)((20618.5 - 5524.7 i) + (1. + 0. i) \omega)}$ 
```

```
In[450]:= Plot[Abs[HP[s  $\omega_g$ ]], {s, 0, 3}, Frame -> True,
  PlotStyle -> Directive[Blue, Thickness[0.006]],
  FrameStyle -> Directive[Blue, FontSize -> 12, FontWeight -> Plain],
  FrameLabel -> {"One-sided Amplitude Distortion", ""},
  {"Cutoff angular frequency normed to be one", ""}]
```

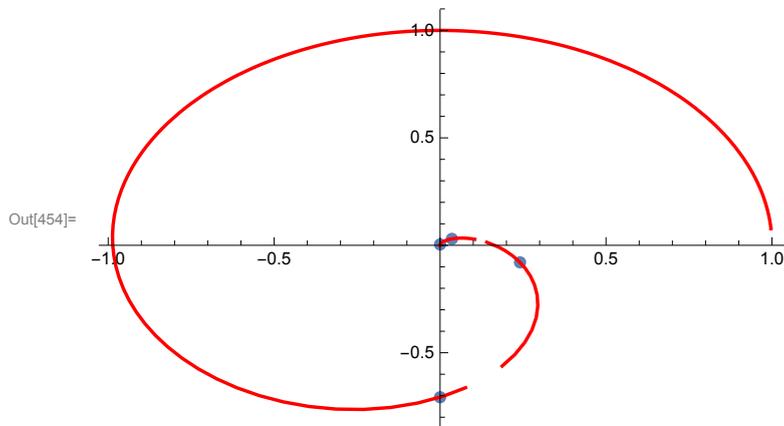


```
In[451]:= Plot[phiHP[s  $\omega_g$ ], {s, 0, 3}, Frame -> True, PlotStyle -> Directive[Blue, Thickness[0.006]],
  FrameStyle -> Directive[Blue, FontSize -> 12, FontWeight -> Plain],
  FrameLabel -> {"Phase Distortion between  $-\pi$  and  $\pi$ ", ""},
  {"Cutoff angular frequency normed to be one", ""}]
```



Hier der Frequenzgang wieder als komplexe Ortskurve mit Werten für einige Kreisfrequenzen. Die Kurve läuft für $\omega \rightarrow \infty$ nach Eins und durchläuft 6 Quadranten beginnend im Nordwest-Quadranten.

```
In[452]:= ortskurve2 := ParametricPlot[{Re[HP[s ωg]], Im[HP[s ωg]]},
    {s, 0, 50}, PlotStyle → Directive[Red, Thickness[0.005]]]
punkte := Table[{Re[HP[s ωg]], Im[HP[s ωg]]}, {s, 0.4, 1, 0.2}];
Show[{ListPlot[punkte, PlotStyle → PointSize[.018]], ortskurve2}, PlotRange → All]
```



Zum Vergleich wieder das in *Mathematica* bereits implementierte Modell, das zum gleichen Ergebnis führt, wenn wir dieselben Dämpfungswerte vorgeben, die unser gerade berechnetes Hochpass-Filter aufweist:

```
In[455]:= -20 Log[10, Abs[HP[ω1]]]
-20 Log[10, Abs[HP[ω2]]]
```

Out[455]= 7.36217

Out[456]= 0.0418452

```

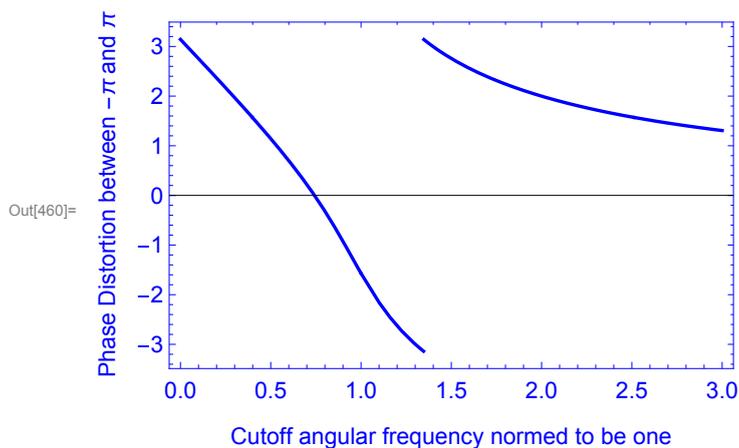
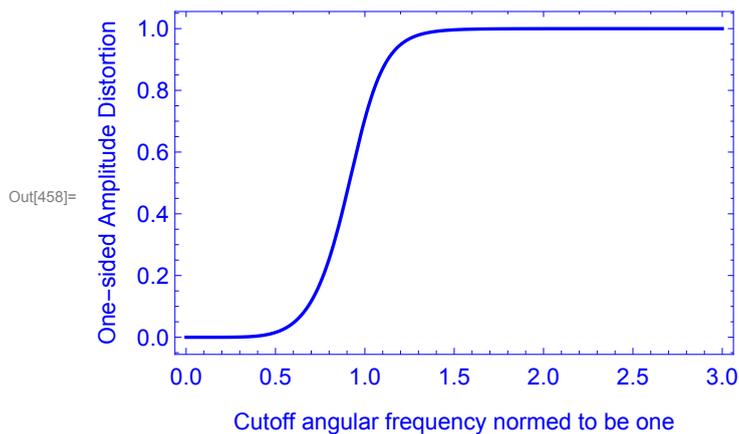
In[457]:= hp = ButterworthFilterModel[
  {"Highpass", { $\omega_1$ ,  $\omega_2$ }, {-20 Log[10, Abs[HP[ $\omega_1$ ]]], -20 Log[10, Abs[HP[ $\omega_2$ ]]]}}]
(* Hier sind die oben benötigten Berechnungen schon implementiert
   und wir sehen gleichen Amplituden- und Phasengang *)
Plot[Abs[hp[t  $\mathbb{i}$   $\omega_g$ ]], {t, 0, 3}, Frame  $\rightarrow$  True,
  PlotStyle  $\rightarrow$  Directive[Blue, Thickness[0.006]],
  FrameStyle  $\rightarrow$  Directive[Blue, FontSize  $\rightarrow$  12, FontWeight  $\rightarrow$  Plain],
  FrameLabel  $\rightarrow$  {"One-sided Amplitude Distortion", ""},
  {"Cutoff angular frequency normed to be one", ""}]
phihp[ $\omega_+$ ] := Assuming[Element[ $\omega$ , Reals], Arg[hp[ $\mathbb{i}$   $\omega$ ]];
Plot[phihp[s  $\omega_g$ ], {s, 0, 3}, Frame  $\rightarrow$  True,
  PlotStyle  $\rightarrow$  Directive[Blue, Thickness[0.006]],
  FrameStyle  $\rightarrow$  Directive[Blue, FontSize  $\rightarrow$  12, FontWeight  $\rightarrow$  Plain],
  FrameLabel  $\rightarrow$  {"Phase Distortion between  $-\pi$  and  $\pi$ ", ""},
  {"Cutoff angular frequency normed to be one", ""}]

```

```

Out[457]= 
$$\frac{\left( (5524.7 - 20618.5 \mathbb{i}) + (1. + 0. \mathbb{i}) \zeta \right) \left( (5524.7 + 20618.5 \mathbb{i}) + (1. + 0. \mathbb{i}) \zeta \right) \left( (15093.8 - 15093 \mathbb{i}) + (1. + 0. \mathbb{i}) \zeta \right)}{\dots}$$

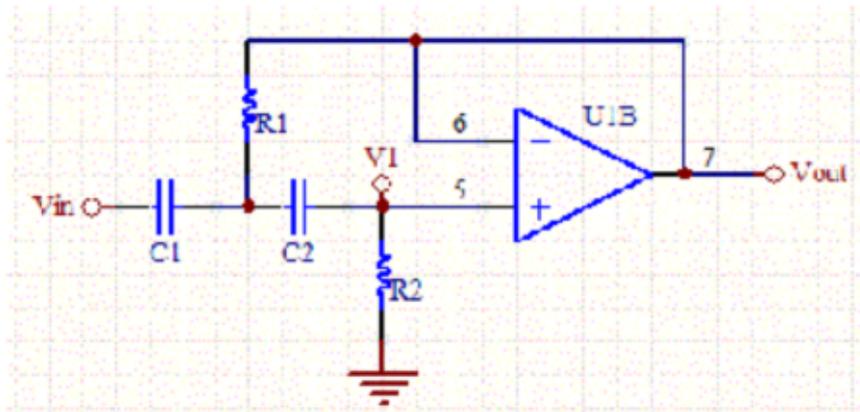

```



Realisierung mit Hardware

Ein "RC-Active Sallen-Key-Highpass-Filter" zweiter Ordnung ist nachfolgend dargestellt:

biquadhigh = Import["Sallen-Key-HP-Filter.png", ImageSize -> Large]



Interessierte Leser seien dazu angeregt, die Übertragungsfunktion $HP(s)$ der Schaltung selbst zu überlegen und für gegebenen Kapazitäten C_1 , C_2 die benötigten Widerstände zur Realisierung eines bestimmten Hochpass-Frequenzgangs auszurechnen.

Die Übertragungsfunktion lautet:

$$HP(s) = s^2 / (s^2 + s(1/(R_2 C_1) + 1/(R_2 C_2)) + 1/(R_1 R_2 C_1 C_2))$$

2. Tiefpass-Bandpass-Transformation

Eine Möglichkeit, einen Bandpass zu entwerfen, ist die sog. Tiefpass-Bandpass-Transformation. Dabei ersetzt man in einem Tiefpass-Frequenzgang den normierten Parameter $i\omega_n = i\omega / \omega_g$ (ω_g die Grenzkreisfrequenz) durch den Parameter $1/B (i\omega_n + 1/(i\omega_n))$. B ist dabei die frei wählbare normierte Bandbreite des Bandpassfilters. Wir beginnen als Demo-Beispiel nachfolgend mit einem Butterworth-Tiefpass 2. Ordnung mit Grenzfrequenz 150 Hz, wählen $B=1/2$ und gewinnen ein Bandpassfilter 4. Ordnung.

```

In[461]:=  $\omega_g := 300 \pi$ ; (* 150 Hz als Grenzfrequenz eines Butterworthtiefpasses 2. Ordnung *)
B := 1/2; (* B=1/2 als normierte Bandbreite des Bandpassfilters *)
htp[s_] := 1 / (1 +  $\sqrt{2}$  s + s2);
(* die Übertragungsfunktion des Butterworth-Tiefpassfilters 2. Ordnung mit s =
   $i\omega/\omega_g$  *)
hbp[s_] = Together[B2 s2 / (1 +  $\sqrt{2}$  B s + (2 + B2) s2 +  $\sqrt{2}$  B s3 + s4)]
(* die Übertragungsfunktion des Bandpassfilters nach Tiefpass-
  Bandpass-Transformation mit s= $i\omega/\omega_g$  *)

```

Out[464]=

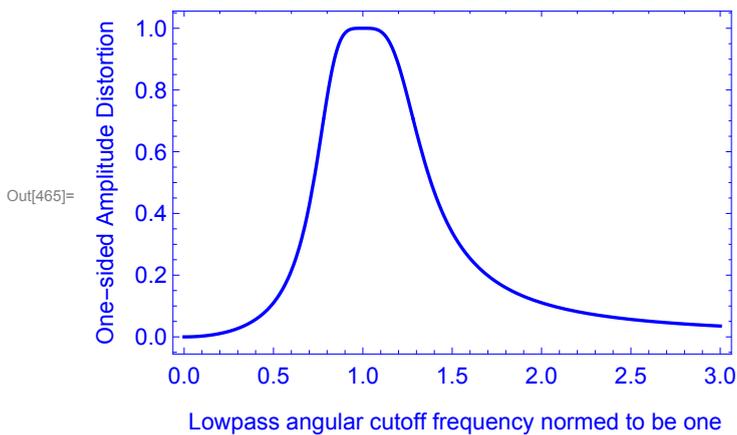
$$\frac{s^2}{4 + 2\sqrt{2}s + 9s^2 + 2\sqrt{2}s^3 + 4s^4}$$

Nun Amplituden- und Phasengang des berechneten Bandpassfilters.

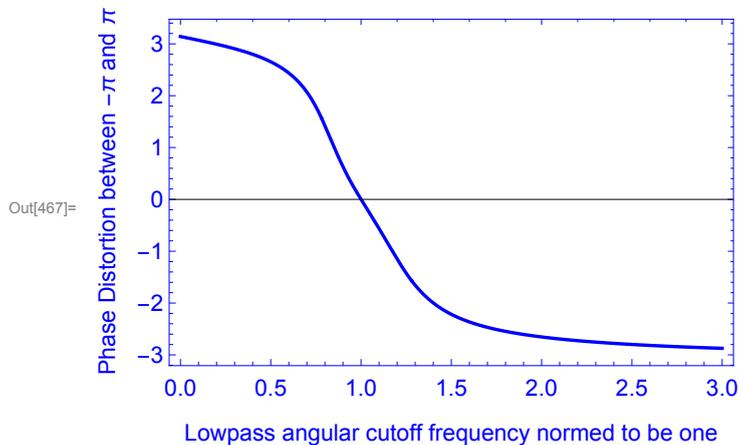
```

In[465]:= Plot[Abs[hbp[ $i$  t]], {t, 0, 3}, Frame -> True,
PlotStyle -> Directive[Blue, Thickness[0.006]],
FrameStyle -> Directive[Blue, FontSize -> 12, FontWeight -> Plain],
FrameLabel -> {"One-sided Amplitude Distortion", ""},
{"Lowpass angular cutoff frequency normed to be one", ""}, PlotRange -> All]

```

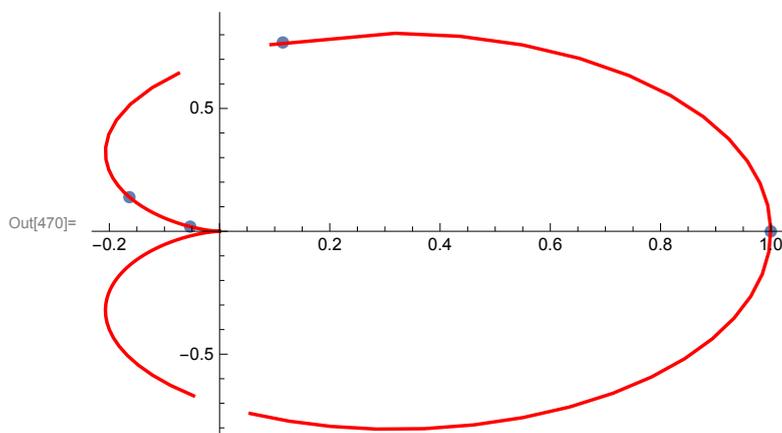


```
In[466]:= phihbp[r_] := Assuming[Element[r, Reals], Arg[hbp[i r]]];
Plot[phihbp[r], {r, 0, 3}, Frame → True, PlotStyle → Directive[Blue, Thickness[0.006]],
FrameStyle → Directive[Blue, FontSize → 12, FontWeight → Plain],
FrameLabel -> {"Phase Distortion between  $-\pi$  and  $\pi$ ", ""},
{"Lowpass angular cutoff frequency normed to be one", ""}]
```



Und auch hier noch die Darstellung des Frequenzgangs als Ortskurve.
Man kann sie als Kombination einer Ortskurve zu einem Hochpass 2. Ordnung und der eines Tiefpassfilters 2. Ordnung sehen (vgl. die Ortskurven weiter oben).

```
In[468]:= ortskurve3 := ParametricPlot[{Re[hbp[i s]], Im[hbp[i s]]},
{s, 0, 50}, PlotStyle → Directive[Red, Thickness[0.005]]];
punkte := Table[{Re[hbp[i s]], Im[hbp[i s]]}, {s, 0.4, 1, 0.2}];
Show[{ListPlot[punkte, PlotStyle → PointSize[.018]], ortskurve3}, PlotRange → All]
```



Die 3 dB-Grenzen f_1 und f_2 des Passbandes in Hz

```
In[471]:= f1 = N[(sqrt(1 + B^2/4) - B/2) * omega_g / (2 * pi)]
f2 = N[(sqrt(1 + B^2/4) + B/2) * omega_g / (2 * pi)]
```

Out[471]= 117.116

Out[472]= 192.116

```
In[473]:= 20 Log [10, Abs [hbp [ i 2 π f1 / ωg ] ] ]
          20 Log [10, Abs [hbp [ i 2 π f2 / ωg ] ] ]
```

```
Out[473]= - 3. 0103
```

```
Out[474]= - 3. 0103
```

Selbstverständlich lässt sich ein Bandpassfilter auch durch serielle Verknüpfung eines Tiefpass- und eines Hochpassfilters konstruieren. Realisierungsbeispiele hatten wir oben schon gesehen. Diese Methode scheint auch dem Transfer-Modell für Bandpassfilter in *Mathematica* zugrunde zu liegen. Testen Sie das selbst. Andere Schaltungsbeispiele finden Sie in der Quelle [2].

3. Tiefpass-Bandsperren-Transformation

Bei der Tiefpass-Bandsperren-Transformation ersetzt man den normierten Parameter $i\omega/\omega_g$ im Frequenzgang eines Tiefpassfilters durch $B/(i\omega/\omega_g - i\omega_g/\omega)$. B ist dabei die normierte 3 dB-Bandbreite der Bandsperre, ω_g wird Nullstelle des Frequenzgangs. Je kleiner B ist, desto steiler fällt die Verstärkung in der Umgebung von ω_g ab. Als Beispiel gehen wir aus von einem Butterworth-Tiefpassfilter 2. Ordnung wie oben mit der Übertragungsfunktion $h_{tp}[s]$ und $\omega_g=300\pi$. $B=1/2$. Durch die Transformation entsteht ein Bandsperren-Filter (engl. Notch filter) 4. Ordnung.

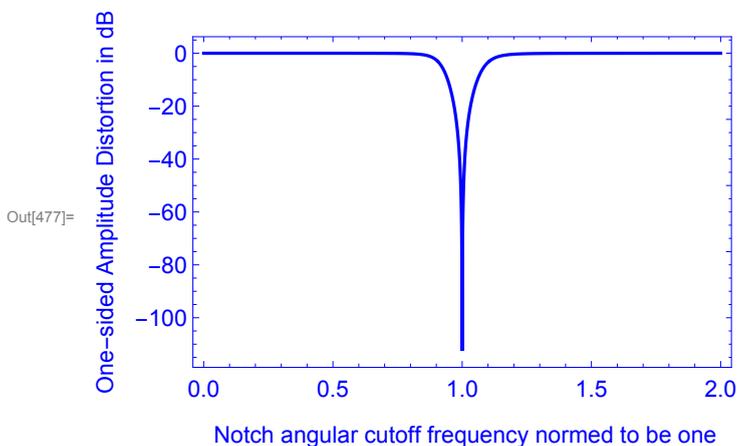
```
In[475]:= B := 1 / 5;
          hbsp[s_] = Together[htp[B / (s + 1 / s)]]
```

```
Out[476]= 
$$\frac{25 (1 + s^2)^2}{25 + 5\sqrt{2} s + 51 s^2 + 5\sqrt{2} s^3 + 25 s^4}$$

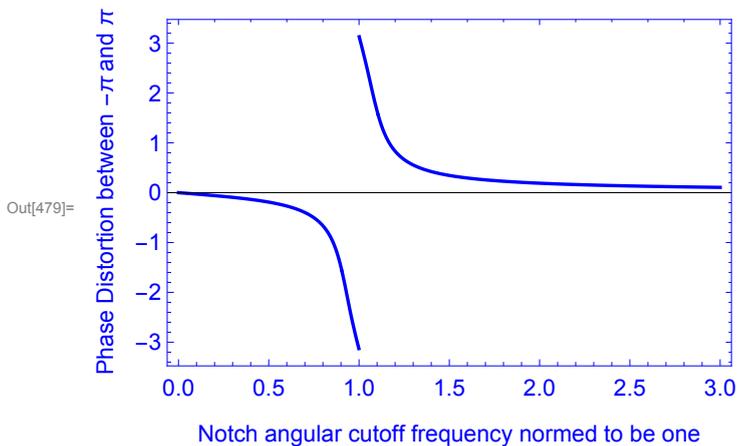
```

Nachfolgend zum Abschluss der Amplitudengang, der Phasengang und die zugehörige Ortskurve

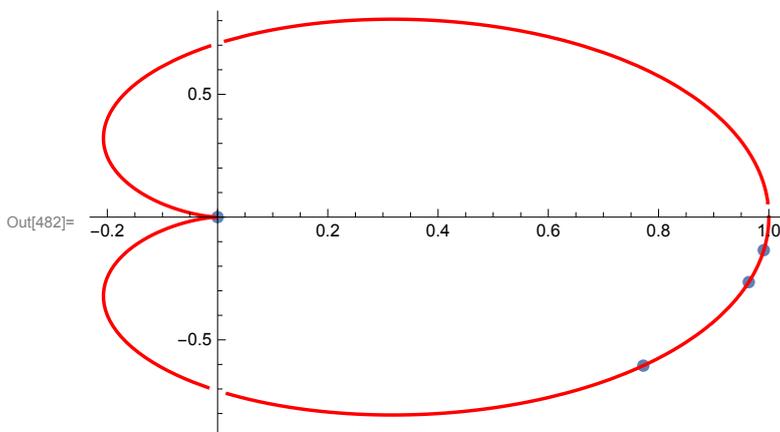
```
In[477]:= Plot[20 Log[10, Abs[hbsp[ i t ] ]], {t, 0, 2},
             Frame → True, PlotStyle → Directive[Blue, Thickness[0.006]],
             FrameStyle → Directive[Blue, FontSize → 12, FontWeight → Plain],
             FrameLabel -> {"One-sided Amplitude Distortion in dB", ""},
             {" Notch angular cutoff frequency normed to be one", ""}], PlotRange → All]
```



```
In[478]:= phihbsp[r_] := Assuming[Element[r, Reals], Arg[hbsp[i r]]];
Plot[phihbsp[r], {r, 0, 3}, Frame → True,
PlotRange → All, PlotStyle → Directive[Blue, Thickness[0.006]],
FrameStyle → Directive[Blue, FontSize → 12, FontWeight → Plain],
FrameLabel -> {"Phase Distortion between  $-\pi$  and  $\pi$ ", ""},
{"Notch angular cutoff frequency normed to be one", ""}]
```



```
In[480]:= ortskurve4 := ParametricPlot[{Re[hbsp[i s]], Im[hbsp[i s]]},
{s, 0, 5}, PlotStyle → Directive[Red, Thickness[0.005]]];
punkte := Table[{Re[hbsp[i s]], Im[hbsp[i s]]}, {s, 0.4, 1, 0.2}];
Show[{ListPlot[punkte, PlotStyle → PointSize[.018]], ortskurve4}, PlotRange → All]
```



Konkrete Schaltungsbeispiele für Bandsperrfilter findet man in der unten genannten Referenz [2]. Mehr zum Thema, auch über weitere spezielle konkrete Filterschaltungen und Tabellen mit den Koeffizienten unterschiedlicher Filtertypen (Tschebyscheff-, Bessel-Filter u.a.) findet man ebenfalls in [2] - [6]. Im praktischen Einsatz findet man Elektronik-Bauteile, mit denen verschiedene Filtertypen wie oben besprochen realisiert werden, in einer immensen Vielfalt von Geräten, vom Wäschetrockner, in Audiosystemen, E-Bikes, in der Telekommunikation bis zum Space-Shuttle. Interessierten Lesern seien daher die nachfolgend genannten Quellen zur vertiefenden Lektüre ans Herz gelegt.

Weitere empfehlenswerte Referenzen:

- [2] U. Tietze, Ch. Schenk Halbleiter-Schaltungstechnik
Springer, 2002
- [3] H. Wupper, U. Niemeyer Elektronische Schaltungen 2
Springer, 1996
- [4] R. Unbehauen Systemtheorie
Oldenbourg, 2002
- [5] R.P. Sallen, E.L. Key A Practical Method of Designing RC Active Filters
IRE Transactions on Circuit Theory, Vol. CT-2, 74-85, 1955
- [6] T. Kugelstadt Active Filter Design Technique
<http://focus.ti.com/lit/ml/sloa088/sloa088.pdf>

u.v.a.m., die Sie im Literaturverzeichnis am Ende von [1] finden können.