

Aufgabenstellung: Übertragungsfunktion eines analogen Filters

Aus definierten Variablen sollen 4 Diagramme erstellt werden.

Jeweils der Amplitudengang und der Phasengang der Butterworth und Besselfunktion Charakteristik und die Pol-Nullstellen der Charakteristiken soll gezeichnet werden.

Contents

Funktion Bodediagramm.m	1
Funktion Transfer.m	3
Funktion hjw.m	4

Funktion Bodediagramm.m

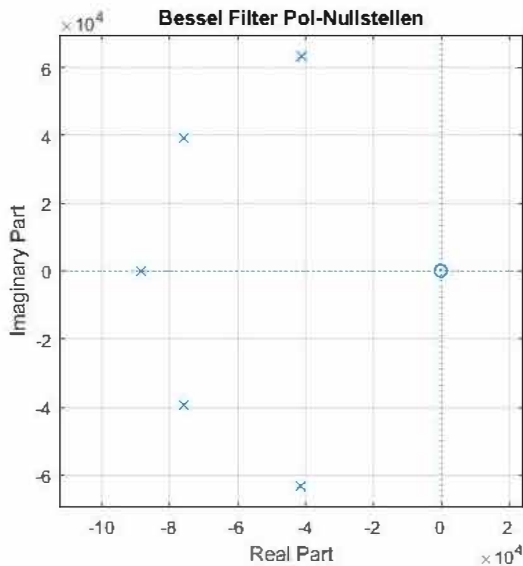
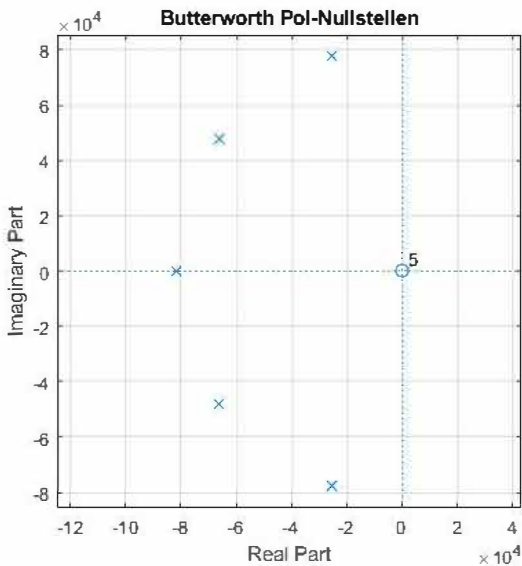
Variablen definieren

```
lnr = 13;
order = 4+mod(lnr,3);
fg = 1000 * lnr;
wg = fg*2*pi;
fch = 'butter';
ftyp = 'high';
fmin = 1e-2*fg;
fmax = 10*fg;
n=500;
```

Pol-Nullstellen Diagramme zeichnen

```
figure('Position', [100, 100, 1024, 1024]);
subplot(2,2,1);
[hdb_butter,hdeg_butter,f_butter] = Transfer(fch,ftyp,fmin,fmax,n,order,wg);

fch = 'besself';
subplot(2,2,2);
[hdb_bessel,hdeg_bessel,f_bessel] = Transfer(fch,ftyp,fmin,fmax,n,order,wg);
```

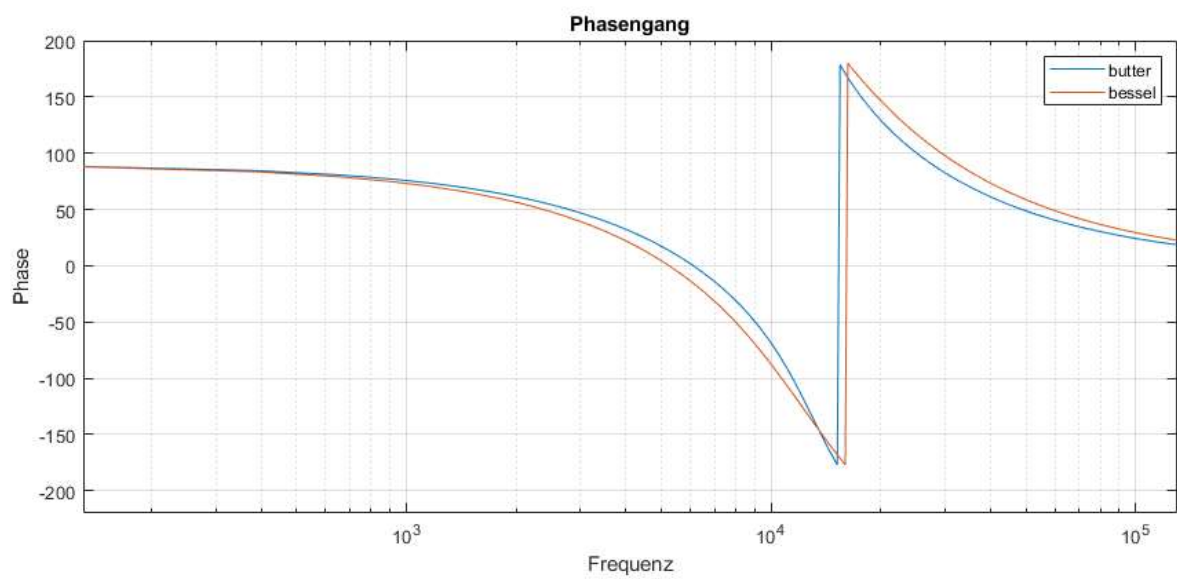
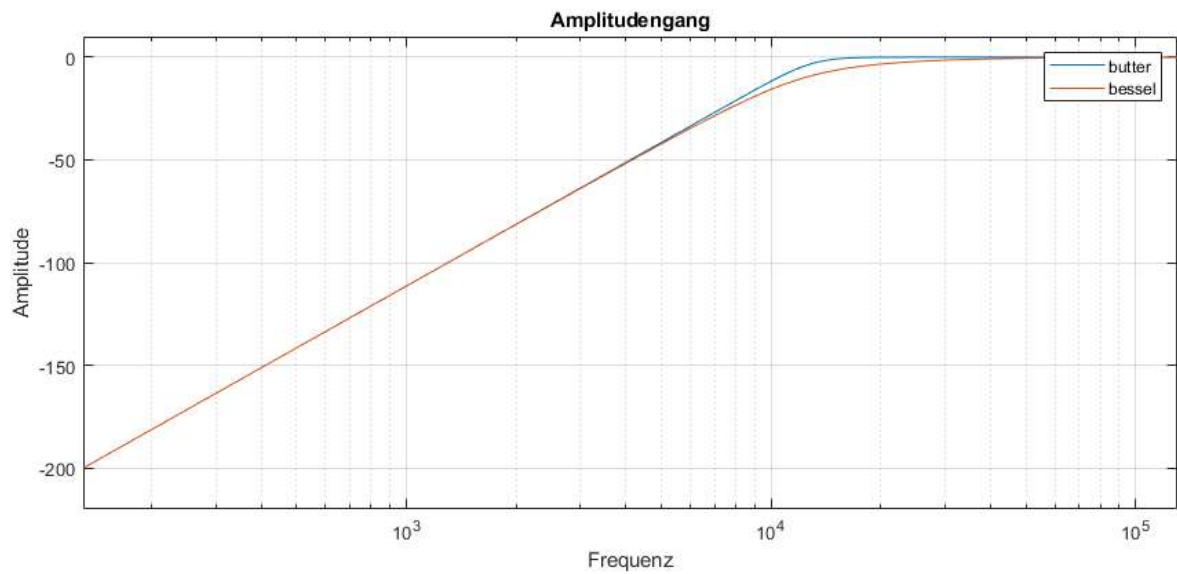


Amplitudengänge über die Frequenz zeichnen

```
figure('Position', [100, 100, 1024, 1024]);
subplot(2,1,1);
semilogx(f_butter,hdb_butter, f_bessel,hdb_bessel);
axis([fmin fmax -220 10]);
title('Amplitudengang');
xlabel('Frequenz');
ylabel('Amplitude');
legend('butter','bessel');
grid on;
```

Phasengänge über die Frequenz zeichnen

```
subplot(2,1,2);  
semilogx(f_butter,hdeg_butter, f_bessel,hdeg_bessel);  
axis([fmin fmax -220 200]);  
title('Phasengang');  
xlabel('Frequenz');  
ylabel('Phase');  
legend('butter','bessel');  
grid on;
```



Funktion Transfer.m

```
function [hdb,hdeg,f] = Transfer(fch,ftyp,fmin,fmax,n,order,wg)
```

```
%hdb Vektor Amplitudengang  
%hdeg Vektor Phasengang  
%f Frequenzvektor  
%fch Charakteristik ('butter','cheby1','cheby2','bessel','Cauer')  
%ftyp 'low','high','bandpass','stop'  
%fmin minmale Frequenz  
%fmax maximale Frequenz  
%n Anzahl der Frequenzpunkte  
%order Filterordnung  
%wg Kreisgrenzfrequenz
```

Mit switch wird ermittelt welche Funktionscharakteristik angewandt wird

```
switch fch  
    case 'butter'  
        [bn,an] = butter(order, wg, ftyp, 's');  
        type='Butterworth Pol-Nullstellen';  
    case 'besself'  
        [bn,an] = besself(order, wg, ftyp);  
        type='Bessel Filter Pol-Nullstellen';  
    otherwise  
end
```

Pol-Nullstellen Diagramm erstellen

```
%Wurzel der Filterkoeffizienten bn und an nehmen  
rbn = roots(bn);  
ran = roots(an);  
  
zplane(rbn, ran);  
title(type);  
grid on;
```

dB und Grad Werte für alle Frequenzen in Vektor abspeichern

```
u=0;  
f = fmin :(fmax-fmin)/n: fmax;  
  
for i = f  
    u = u + 1;  
    [hdb(u), hdeg(u)] = hjw(bn, an, i);  
end
```

```
end
```

Funktion hjw.m

```
function [hdb,hdeg]=hjwt(bn, an, f)
```

```
%Calculate sinus function for angle vector phi  
%A... Amplitude of sinus function  
%phi... angle vector  
%k... number of periods  
%example y=skript_sinus(phi,A,k)  
%length(x) berechnet Länge des Vektors  
%hd Zählerpolynom  
%hn Nennerpolynom  
%h Übertragungsfunktion  
%hdb Übertragungsfunktion in dB  
%hdeg Übertragungsfunktion in Grad
```

Nennerpolynom

$$h_n = \sum_{i=0}^n b_i * (jw)^i$$

```
hn = 0;  
for n = 1 :1: length(an)  
    hn = hn* 1i*2*pi*f + an(n);  
end
```

Zählerpolynom berechnen

$$h_d = \sum_{i=0}^n b_i * (jw)^i$$

```
hd = 0;  
for n = 1 :1: length(bn)  
    hd = hd* 1i*2*pi*f + bn(n);  
end
```

Gesamtübertragungsfunktion berechnen

$$h(j\omega) = \frac{h_d}{h_n}$$

$$h(j\omega)_{dB} = 20 * \log|h(j\omega)|$$

$$\varphi(h(j\omega)) = \frac{180^\circ}{\pi} * \arg(h(j\omega))$$

```
h=hd/hn;  
hdb = 20* log10(abs(h));  
hdeg = 180/pi * angle(h);
```

```
end
```