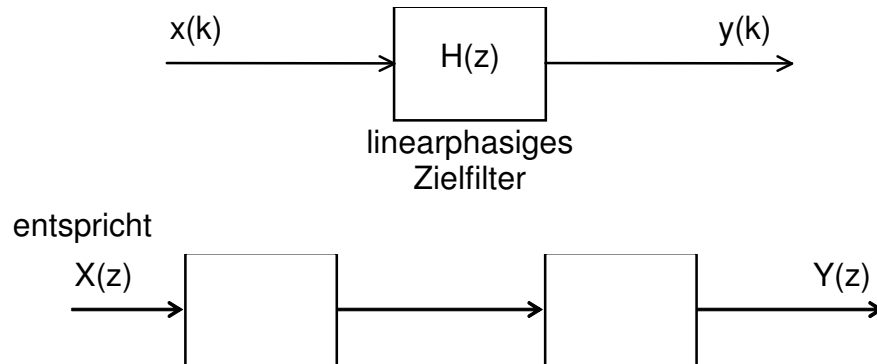


7 Entwurf von FIR-Systemen

7.1 Entwurf linearphasiger FIR-Filter nach der Fenstermethode

7.1.1 Theorie

Ansatz:



- Beschränkung auf geradzahlige N
- Entwurfsfilter: $H_0(z)$
- Zielfilter: $H(z) = z^{-N/2} H_0(z)$

Entwurfsfilter/Zielfilter:

$$H(z) = \underbrace{z^{-N/2}}_{\text{Verzögerung}} \underbrace{\left(b_0 z^{N/2} + b_1 z^{N/2-1} + \dots + b_{N-1} z^{-N/2+1} + b_N z^{-N/2} \right)}_{\text{Nullphasenfilter (Entwurfsfilter), } H_0(z)}$$

Linearphasiges Zielfilter
 $b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N}$

Substitution: $B_{N/2-j} = b_j$:

$$H_0(z) = B_{N/2} z^{N/2} + B_{N/2-1} z^{N/2-1} + \dots + B_{-N/2+1} z^{-N/2+1} + B_{-N/2} z^{-N/2}$$

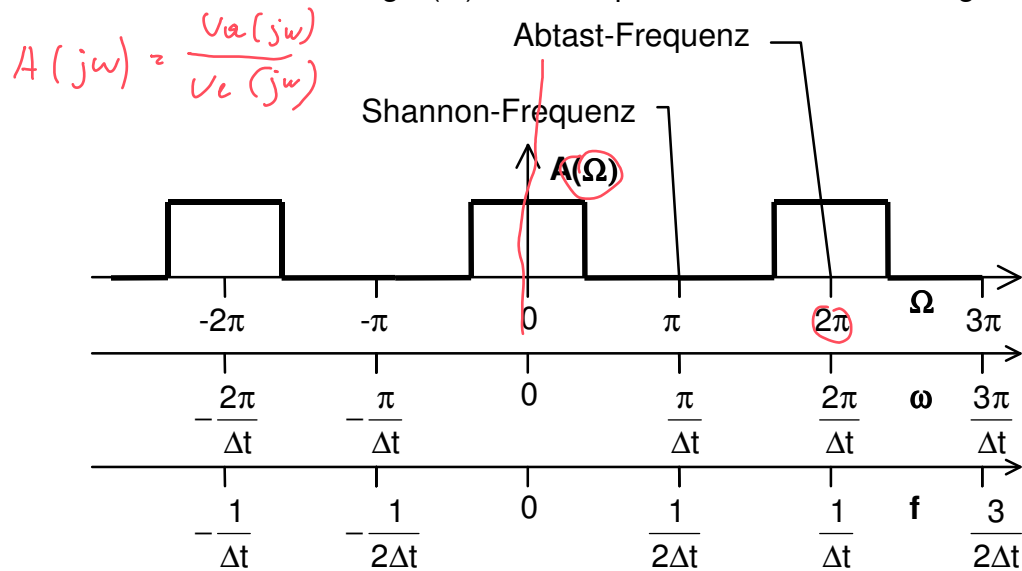
$$H_0(z) = \sum_{n=-N/2}^{N/2} B_n z^n$$

Frequenzgang Entwurfsfilter:

Spezifikation eines Wunschfrequenzgangs:

$$H_{0, \text{Wunsch}}(e^{j\Omega}) = A(\Omega) \quad (\text{Nullphasen-Filter!}), \text{ reellwertig,} \\ \Omega = \omega \Delta t \text{ (diskrete Frequenz)}$$

Grafische Darstellung $A(\Omega)$ incl. der periodischen Fortsetzung



Entwicklung $A(\Omega)$ als Fourierreihe

Idee: $A(\Omega)$ periodisch, kann als Fourierreihe angegeben werden.

Komplexe Fourierreihe:

$$x(t) = \sum_{n=-\infty}^{\infty} C_n e^{jn\omega_0 t} \quad \text{mit} \quad C_n = \frac{1}{T} \int_{-T/2}^{+T/2} x(t) e^{-jn\omega_0 t} dt$$

„Fourierreihe“ „Fourierkoeffizienten“

„periodische Zeit-Funktion $x(t)$ kann als gewichtete Summe von Drehzeigern $e^{jn\omega_0 t}$ (Harmonische der Grundfrequenz ω_0) dargestellt werden“

Angewendet auf $A(\Omega)$:

$$\begin{aligned} x &\rightarrow A \\ t &\rightarrow \omega \\ T &\rightarrow 2\pi \end{aligned}$$

$$\omega_0 \rightarrow 1$$

$$\omega_0 = \frac{2\pi}{T}$$

Darstellung des Wunschfrequenzgangs als Fourierreihe:

$$A(\Omega) = \sum_{n=-\infty}^{\infty} C_n e^{jn\Omega}$$

$$C_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} A(\Omega) e^{-jn\Omega} d\Omega$$

Vergleich mit Frequenzgang $H_0(e^{j\Omega})$ des Entwurfsfilters:

$$A(\Omega) = \sum_{n=-N/2}^{N/2} B_n e^{jn\Omega}$$

Fazit:

Die gesuchten Filterkoeffizienten B_n des Entwurfsfilters sind gerade die Fourierkoeffizienten C_n des Wunschfrequenzgangs

In der Praxis endliche Fourierreihe.

Vereinfachung

da $A(\Omega)$ i.a. achsensymmetrisch:

$$A(\Omega) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(n\Omega) \quad \text{mit} \quad a_n = \frac{2}{\pi} \int_0^{\pi} A(\Omega) \cos(n\Omega) d\Omega$$

Zusammenhang zwischen a_n und C_n : $C_n = C_{-n} = a_n/2$

Zuordnung a... -> Filterkoeffizienten b...

a_{2/2}	a_{1/2}	a_{0/2}	a_{1/2}	a_{2/2}
C ₋₂	C ₋₁	C ₀	C ₁	C ₂
B ₋₂	B ₋₁	B ₀	B ₁	B ₂
b ₄	b ₃	b ₂	b ₁	b ₀
b₀	b₁	b₂	b₃	b₄
b ₀	b ₁	b ₂	b ₁	b ₀

Symmetrielinie

Die letzten 2 Zeilen gelten, da für Nullphasenfilter $b_i = b_{N-i}$ gilt (Symmetrie bzgl. mittlerem Koeffizienten, hier b_2).

Siehe nachfolgende Bemerkung.

- Da $A(\Omega)$ reellwertig und gerade ist (Frequenzgang realer Filter ist immer „konjugiert komplex“) sind die C_n und damit die B_n ebenfalls reellwertig und gerade. Für die b_n des linearphasigen Zielfilters bedeutet dies, dass sie um den Index N/2 (hier 2) symmetrisch sind.

Rezept für Filterentwurf1) Spezifikation

Aufstellen des Wunschfrequenzgangs $A(f)$ und $A(\Omega)$ incl. periodischer Fortsetzung. I.A. werden als Vorgabewerte nur ideale Filter zugrundegelegt (z.B. idealer Tiefpass).

2) Filterordnung N festlegen:

Beschränkung auf $N+1$ Koeffizienten $b_0 \dots b_N$

bzw. $N/2$ Koeffizienten $a_0 \dots a_{N/2}$

3) Bestimmung der Fourierkoeffizienten a_n

$$a_n = \frac{2}{\pi} \int_0^{\pi} A(\Omega) \cos(n\Omega) d\Omega$$

4) Fourierkoeffizienten -> Filterkoeffizienten

Umkopieren der Fourierkoeffizienten $a_{...}$ auf die Filterkoeffizienten $b_{...}$ Bsp. $N=4$:

Fourier-koeffizienten $a_{2/2} \quad a_{1/2} \quad a_{0/2} \quad a_{1/2} \quad a_{2/2}$
Filterkoeffizienten

5) Fensterung der Filterkoeffizienten

Zur Reduktion der Welligkeit im Amplitudengang werden die Koeffizienten aus Schritt 4) mit einem Gewichtungsfenster (Hanning, Bartlett, Kaiser ...) multipliziert, siehe Abschnitt 7.1.3.

6) DC-Gain-Korrektur (Nur für Tiefpässe)

Falls für einen Tiefpass DC-Gain = 1 gewünscht wird, muss folgende Korrektur vorgenommen werden:

- DC-Gain = $H(1) = \underbrace{b_0 + b_1 + b_2 + b_3 + b_4 \dots + b_N}_{\text{Summe der Filterkoeffizienten}}$
- Alle Filterkoeffizienten durch diesen Wert teilen
=> neue DC-Gain = 1

7) Amplitudengang zeichnen/überprüfen

Achtung: reales Filter mit endlichem N ist nur Näherung an Wunschfrequenzgang => Aufzeichnen des tatsächlichen $A(\Omega)$, z.B. über Fourierreihe

$$A(\Omega) = \frac{a_0}{2} + \sum_{n=1}^{N/2} a_n \cos(n\Omega)$$

oder mit Programm (MATLAB) und Vergleich mit Wunsch. Evtl. N erhöhen und Entwurf noch mal wiederholen -> 3).

Anmerkung: Die Filterordnung N ist der einzige Freiheitsgrad für den Filterentwurf (unter der Annahme, dass Wunschfrequenzgang als „ideal“ vorgegeben ist).

Merksatz

Die Filterkoeffizienten b_i entsprechen den Fourierkoeffizienten der Fourierreihe für den Wunschamplitudengang. Sie müssen ggf. zusätzlich gefenstert und DC-Gain-korrigiert werden.

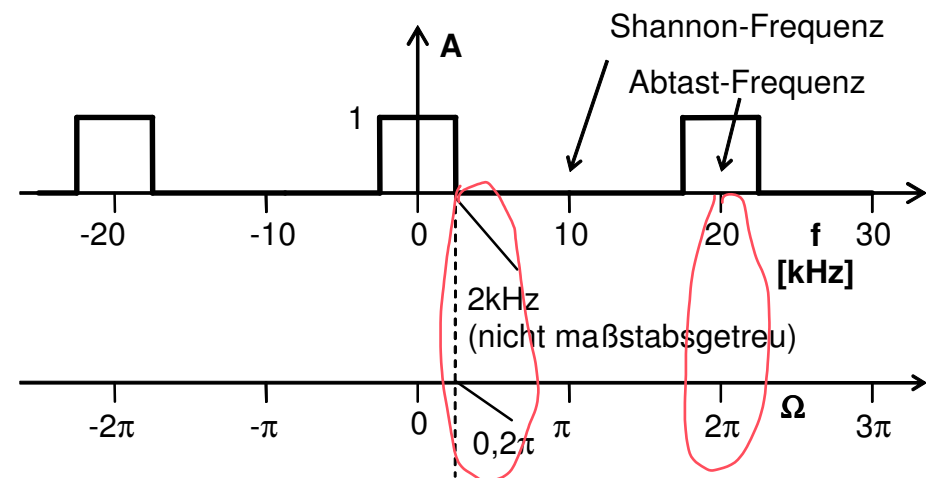
Anmerkungen

- Hier: linearphasige Filter mit symmetrischem Koeffizientensatz $b_i = b_{N-i}$.
- Allgemein: auch antisymmetrischer Koeffizientensatz möglich: $b_i = -b_{N-i}$. Bsp.: „Primitiv-Differenzierer“

$$y(k) = \frac{1}{\Delta t} [x(k) - x(k-1)].$$
- FIR-Filter mit antisymmetrischem Koeffizientensatz haben immer DC-Gain=0.
- FIR-Filter mit symmetrischem Koeffizientensatz können (müssen aber nicht) DC-Gain=0 haben.
 Bsp. für Hochpass mit symmetrischem Koeffizientensatz: Zweite Ableitung numerisch durch Reihenschaltung zweier Primitiv-Differenzierer.
- FIR-Filterentwurf nach der Fenstermethode mit MATLAB: Routine FIR1.

7.1.2 BeispielParameter:

- Abtastfrequenz = 20 kHz
- Wunsch: Idealer Tiefpass mit Grenzfrequenz 2 kHz

Wunschamplitudengang

Fourierkoeffizienten:

$$a_n = \frac{2}{\pi} \int_0^{0,2\pi} \cos(n \cdot \Omega) d\Omega \Rightarrow \text{Rechnen}$$

$$\frac{2 \cdot \sin(628,319 \cdot 10^{-3} \cdot n)}{n \cdot \pi}$$

Einsetzen: $n=0, 1, 2, 3, 4$

$$A(\Omega) = \underbrace{0,4}_{n=0} + \underbrace{0,374}_{n=1} \cdot \cos(\Omega) + 0,302 \cdot \cos(2\Omega) + 0,12 \cdot \cos(3\Omega) + 0,0835 \cdot \cos(4\Omega)$$

Filterkoeffizienten (N=8):*N... Filterordnung*

	$a_4/2$	$a_3/2$	$a_2/2$	$a_1/2$	$a_0/2$	$a_1/2$	$a_2/2$	$a_3/2$	$a_4/2$
1)	0,0468	0,1009	0,1514	0,1871	0,2000	0,1871	0,1514	0,1009	0,0468
2)	0,0399	0,0861	0,1291	0,1596	0,1706	0,1596	0,1291	0,0861	0,0399
3)	0,0051	0,0294	0,1107	0,2193	0,2710	0,2193	0,1107	0,0294	0,0051
	b_0	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8

1) DC-Gain=1,1723

2) nach DC-Gain Korrektur \Rightarrow DC-Gain=1

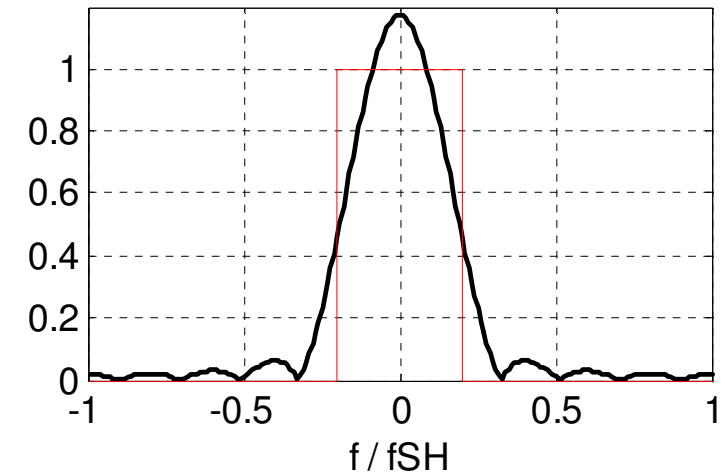
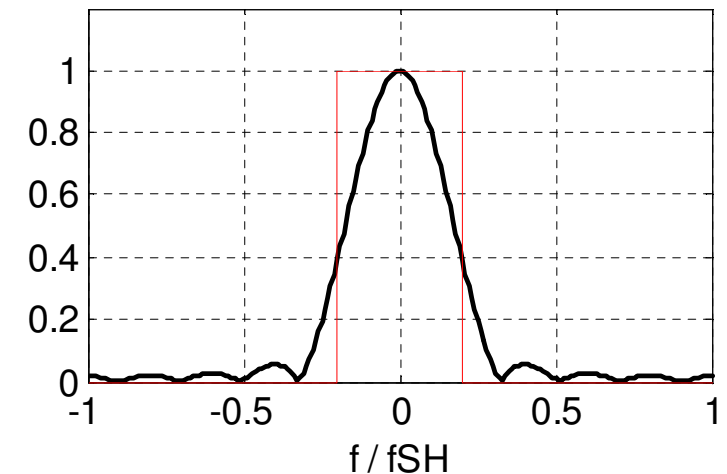
3) nach Koeffizienten-Fensterung und DC-Gain Korrektur

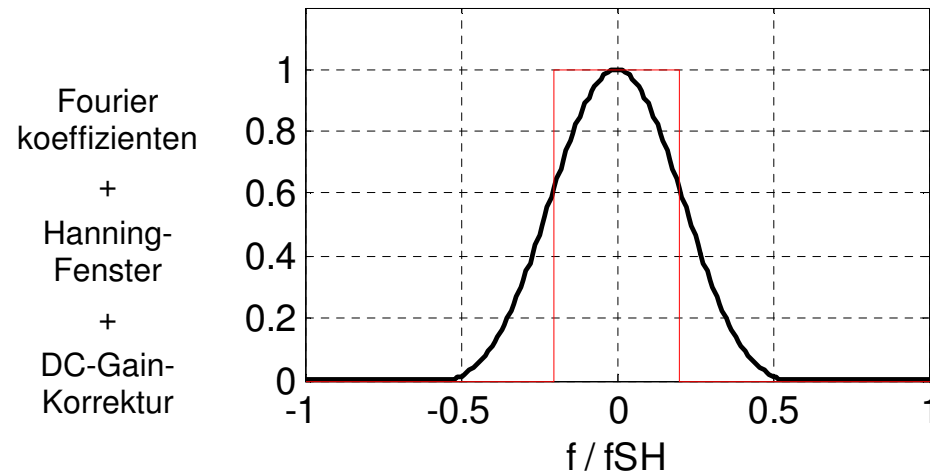
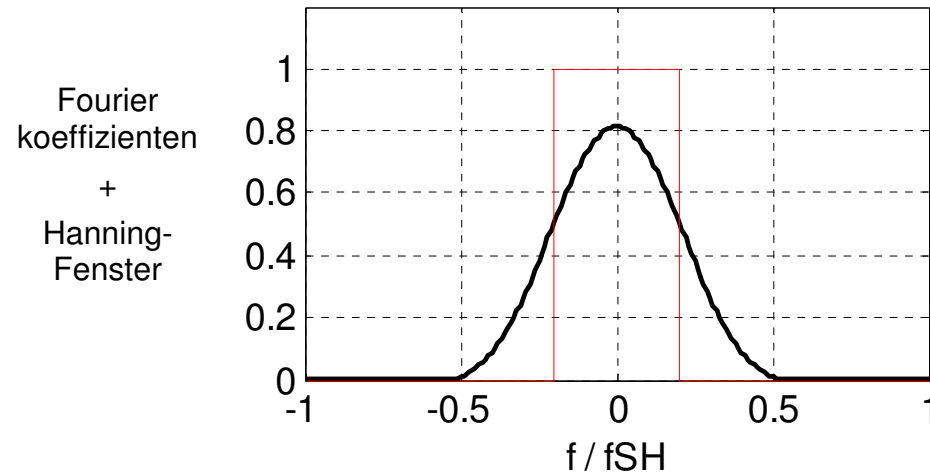
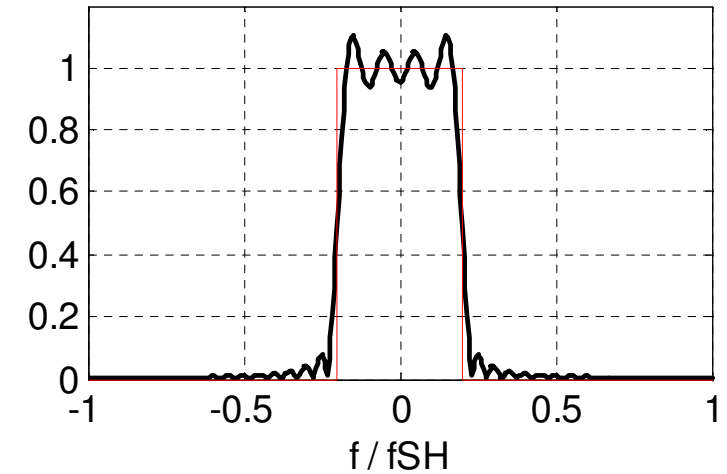
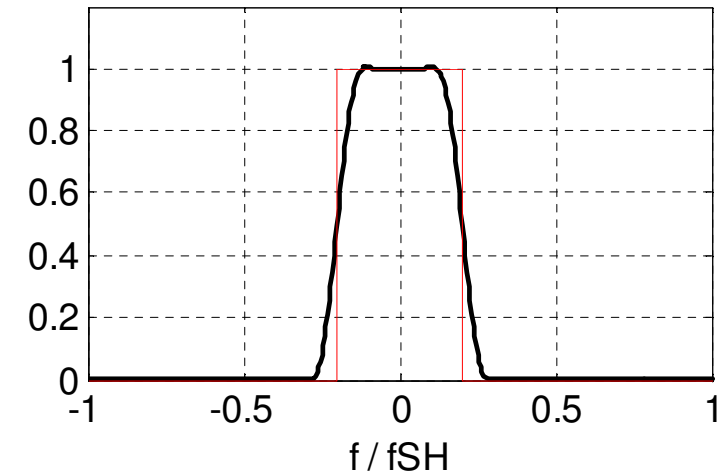
Filter-Übertragungsfunktion (Zielfilter):

$$H(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} + b_4 z^{-4} + b_5 z^{-5} + b_6 z^{-6} + b_7 z^{-7} + b_8 z^{-8}$$

Amplitudengänge (reales Filter):

Fourierkoeffizienten pur

Fourierkoeffizienten
+
DC-Gain-Korrektur

Beispiel Filterordnung N=40Fourierkoeffi-
zienten purFourier
koeffizienten
+
Hanning-
Fenster
+
DC-Gain-
Korrektur

MATLAB-Programm**% Parameter**

```
N=40; % Filterordnung
OmegaD=0.2; % Omega_Grenz/Omega_Shan
```

% Berechnung der Filterkoeffizienten

```
b=fir1(N,OmegaD,Hanning(N+1))
% Koeffizienten b0...bN,
% mit Hanning-Fenster
% und Verst.-korr.
a=[1]; % FIR-Filter ->a0=1
```

% realen Amplitudengang berechnen

```
f=linspace(-1,1,201); % Normierter Frequenz-
% bereich, 201 Pkt.
amp=abs(freqz(b,a,pi*f)); % Amplitudengang (neg.
% + pos. Frequ.)
% Range bis Shannon-
% Frequenz
```

% Wunschamplitudengang

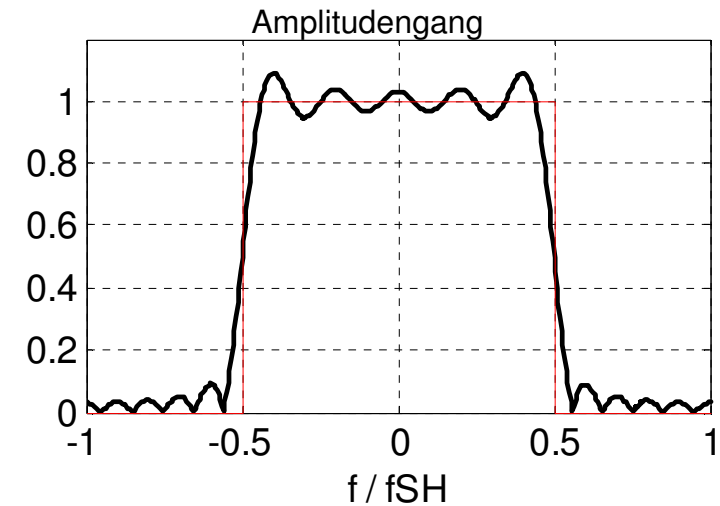
```
fideal=[-1 -OmegaD -OmegaD OmegaD OmegaD 1];
% Frequenzliste
% für Vorgabe
ampideal=[0 0 1 1 0 0]; % Amplitudengangliste
% für Vorgabe
```

% Amplitudengänge malen

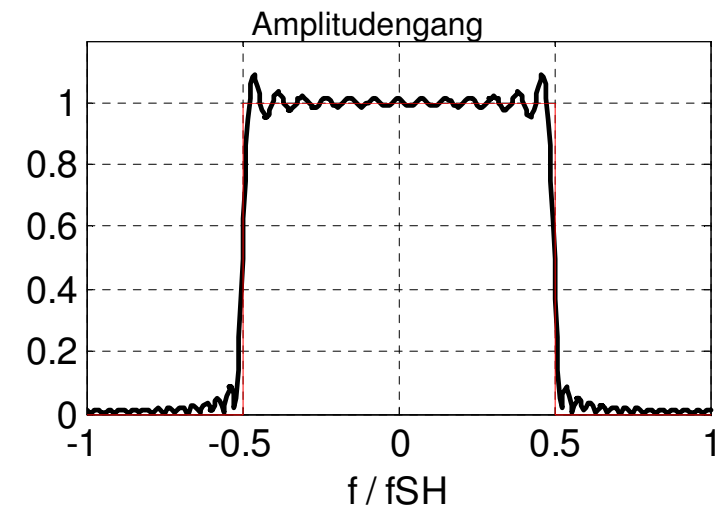
```
plot(f,amp,fideal,ampideal);
% Bild
grid; % mit Gitter
```

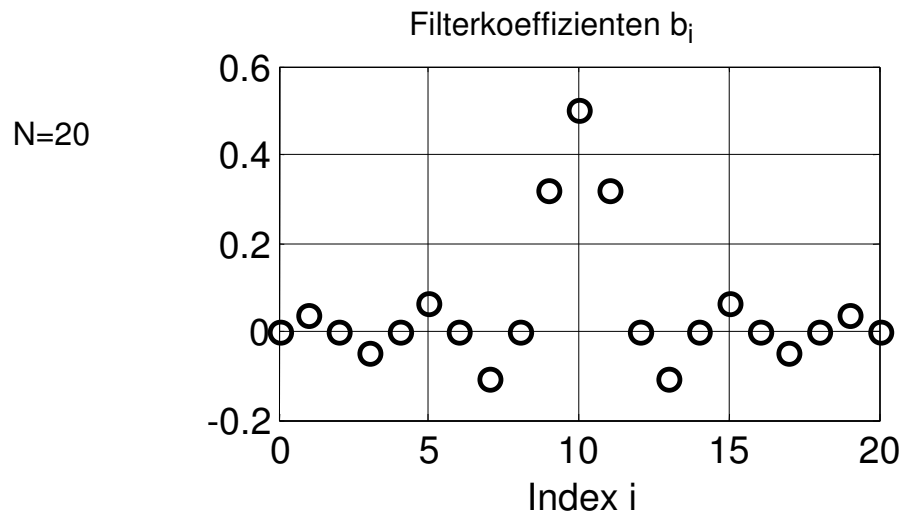
7.1.3 Gewichtungsfenster für die FenstermethodeTiefpass $f_D = 0,5 f_{SH}$ ohne Gewichtungsfenster

N=20



N=50





Problembeschreibung:

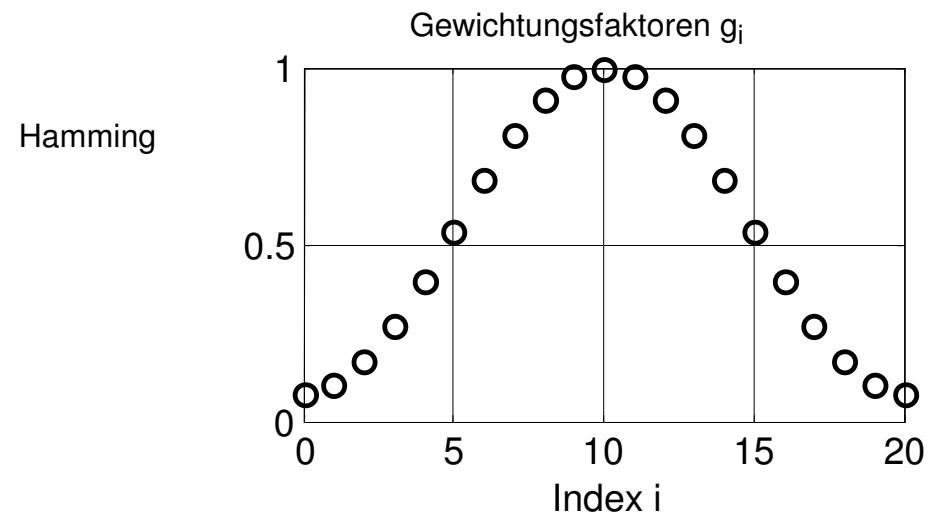
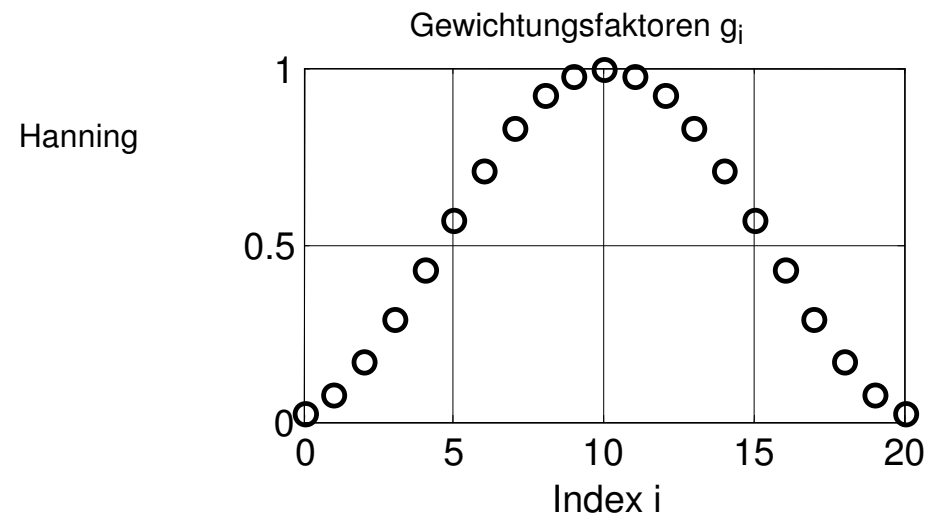
Durch Abbruch der Fourierreihe entsteht Welligkeit insbesondere an steilen Übergängen. Die wellenförmigen Bereiche werden mit zunehmendem N zwar immer schmaler, die Amplitude aber bleibt (siehe oben $N=20 \rightarrow N=50$). Bekanntes Phänomen bei der Fourierreihe: „Gibbs-Phänomen“

Abhilfe:

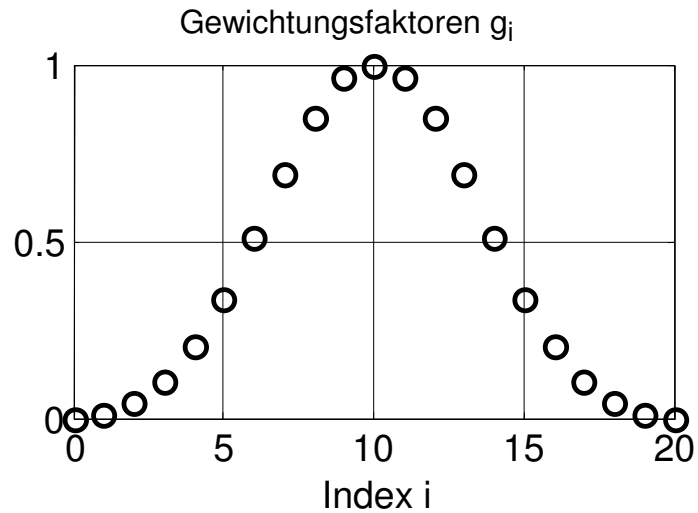
Multiplikation der Fourierkoeffizienten bzw. Filterkoeffizienten mit einer Gewichtungsfunktion $G(i)$, $i = 0..20$, die am Rande des Koeffizientenbereichs Null wird.

Gewichtungsfunktionen („Gewichtungsfenster“):

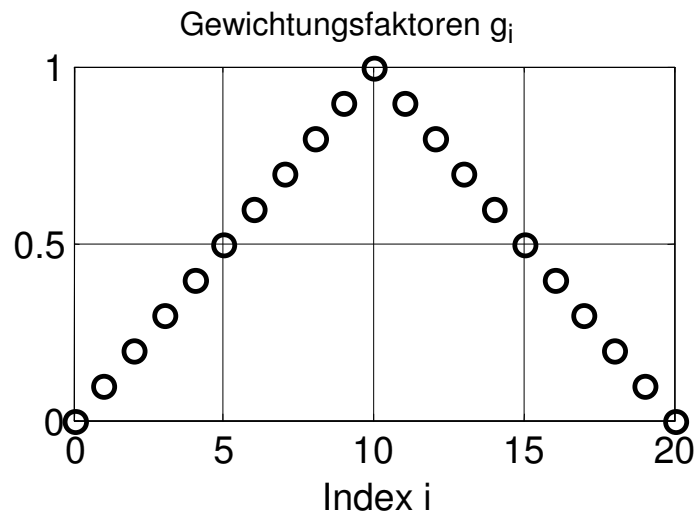
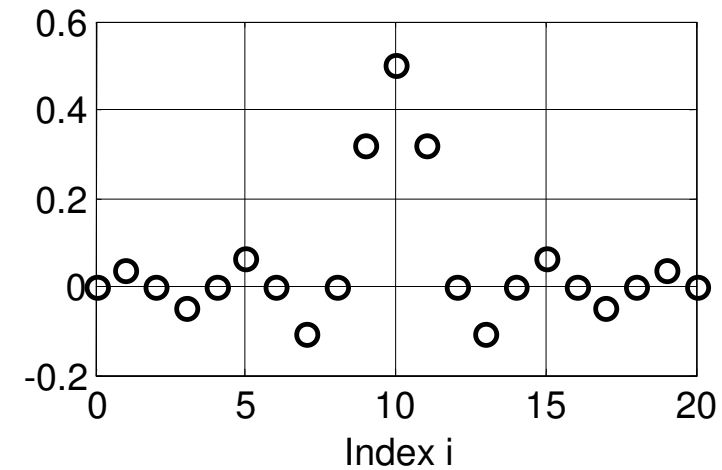
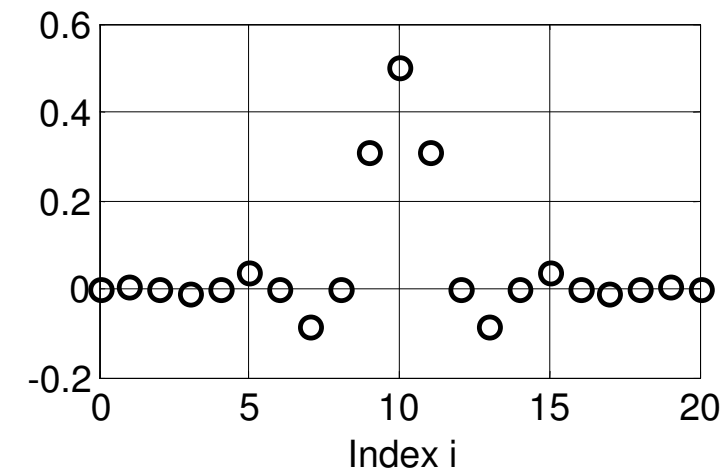
Alle Beispiele für $N=20$, d.h. 21 Taps $b_0 \dots b_{20}$



Blackman



Bartlett

Anwendung (Filter Beispiel von oben)Filterkoeffizienten b_i , ohne GewichtungsfensterFilterkoeffizienten b_i mit Gewichtungsfenster *)

$$*) b_{i,mit} = g_i \cdot b_{i,ohne}$$

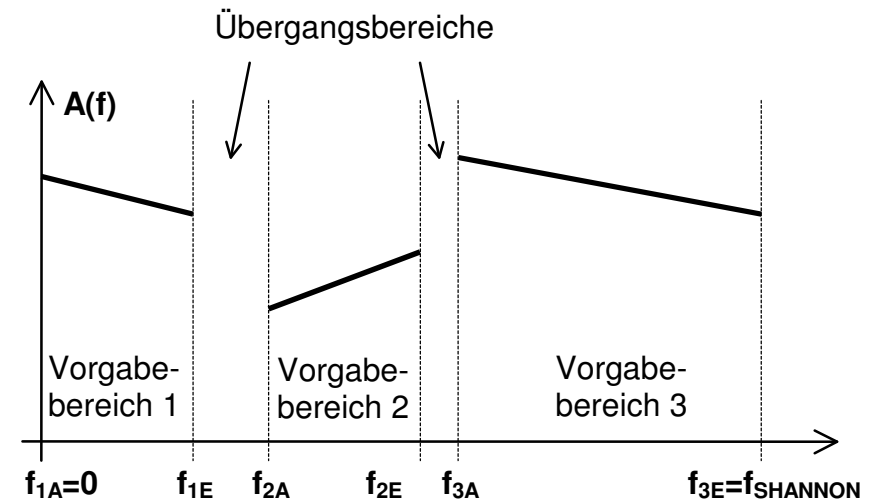
Eigenschaften der Gewichtung:

- Rippel werden reduziert
- Übergang zwischen Durchlassbereich und Sperrbereich wird weniger steil
- Standard: Hanning Fenster (Formel siehe Kapitel 9)

Ähnliches Problem bei FFT, wenn Signal stationär ->
Verwendung derselben Gewichtungsfenster!

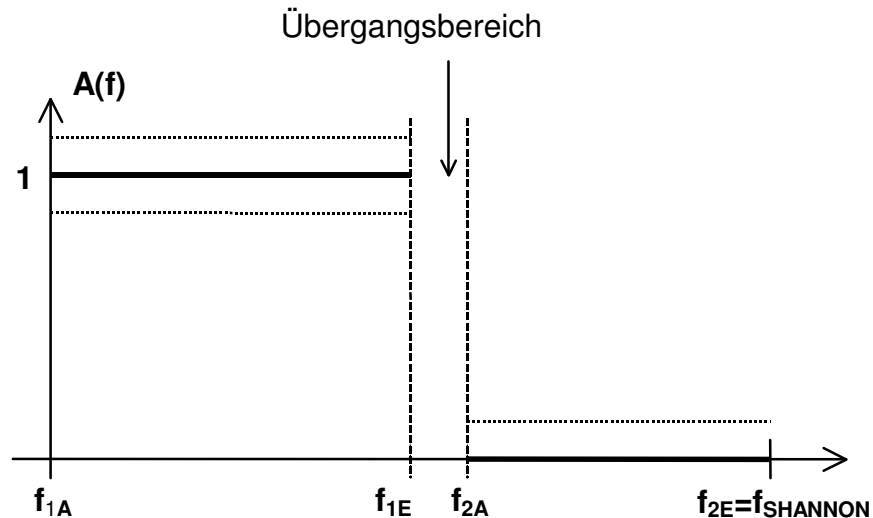
7.2 Verfahren von Parks-Mc Clellan**7.2.1 Idee**Spezifikation:

- Einteilung der Frequenzachse in n Bereiche $[f_{iA}, f_{iE}]$ $i=1, 2 \dots n$ (Vorgabebereiche).
- Vorgabe: Wunsch-Amplitudengang in den Vorgabebereichen als Geraden.
- Zwischen jeweils 2 benachbarten Vorgabebereichen liegt ein Übergangsbereich, in dem der Amplitudengang nicht spezifiziert wird.

Entwurfsidee:

- Ansatz: Optimierungsverfahren
„In den Vorgabebereichen möglichst nah ran“
- Erinnerung an Botschaft des elliptischen Filters:
Erlaube gleichmäßige Welligkeit in den Vorgabebereichen
=> maximal steiflankiges Filter möglich
- Optimierungsziel 1:
Minimale aber gleichmäßige Welligkeit in den Vorgabebereichen

- Verfeinerung:
Um die Vorgabebereichen unterschiedlich zu gewichten, kann für jeden Vorgabebereich durch einen Faktor angegeben werden, mit welcher Gewichtung der Amplitudengangsfehler in diesem Vorgabebereich bei der Gesamt-Optimierung berücksichtigt wird. Beispiel Specs. für einen Tiefpass:



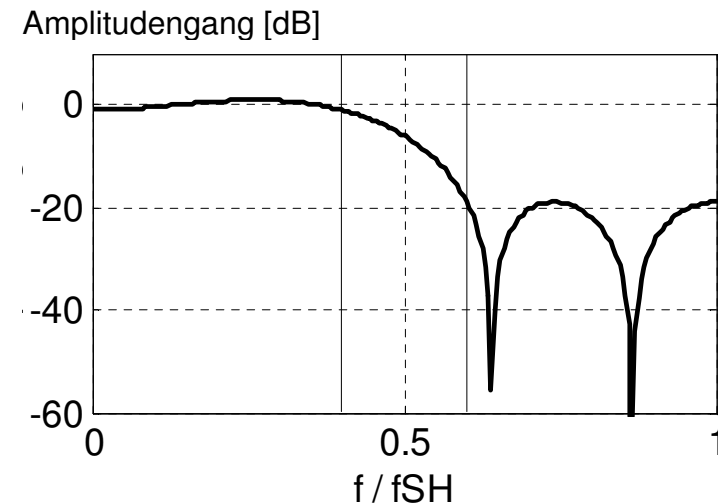
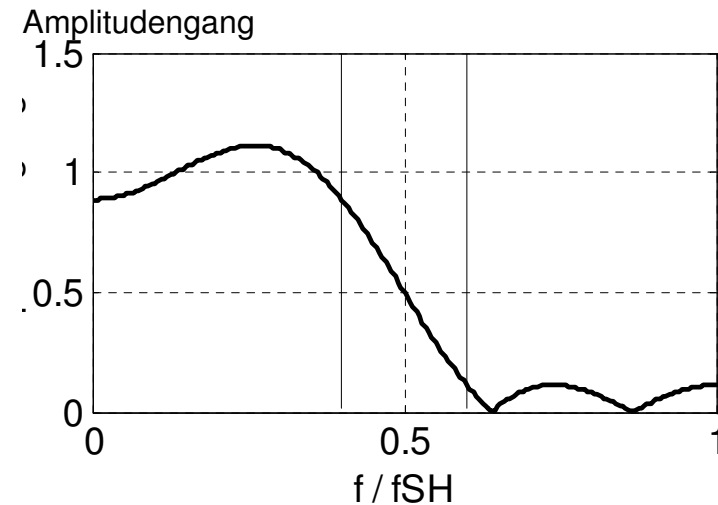
Amplitudengangschwankung Durchlassbereich $\pm 0,1$
Sperrdämpfung 40dB (Schwankung $\pm 0,01$)

- => Gewichtungsfaktor 1 für Durchlassbereich
- Gewichtungsfaktor 10 für Sperrbereich
- => Der gewichtete Amplitudengangsfehler ist dann in beiden Bereichen 0,1.

7.2.2 Beispiel Tiefpass

Beispiel 1:

$$N=8 \quad \tilde{f}_{1E}=0,4 \quad \tilde{f}_{2A}=0,6 \quad \underline{w} = [1 \ 1]$$



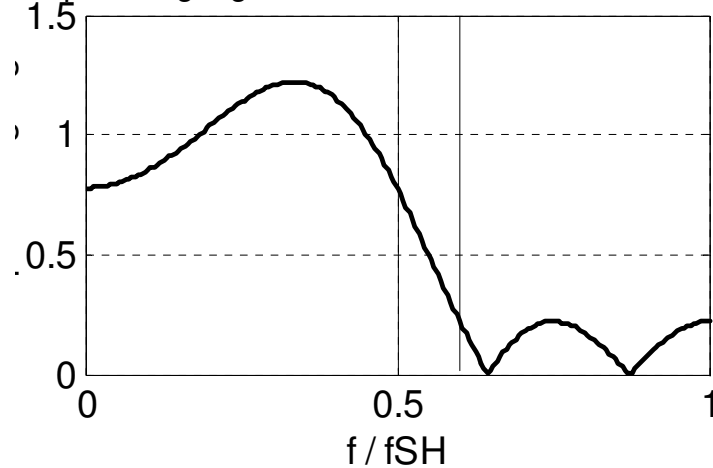
Beispiel 2:

$$N=8 \quad \tilde{f}_{1A}=0 \quad \tilde{f}_{1E}=0,5 \quad \tilde{f}_{2A}=0,6 \quad \tilde{f}_{2E}=1$$

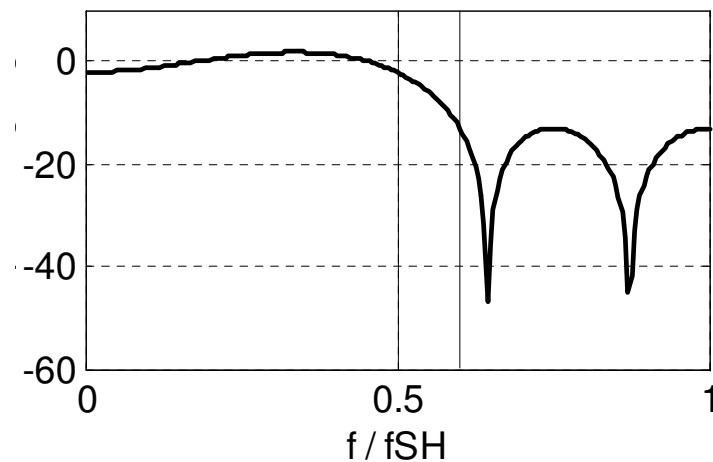
(Frequenzen auf Shannon-Frequenz bezogen: $\tilde{f} = f/f_{SH}$)

Fehlergewichtungsvektor $\underline{w} = [1 \ 1]$

Amplitudengang



Amplitudengang [dB]

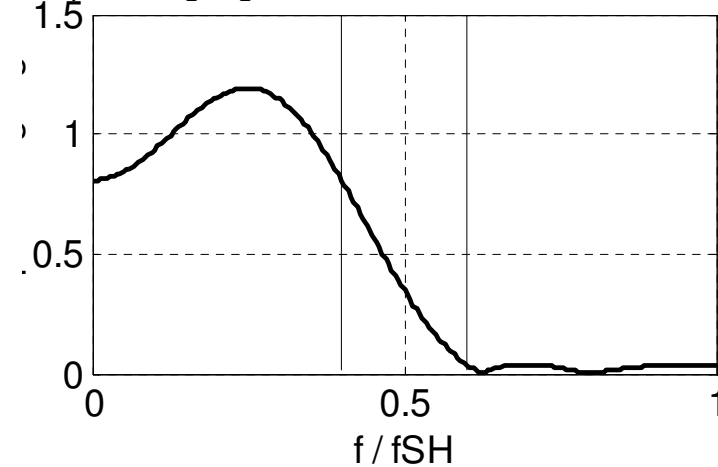


Fazit: Übergangsbereich breiter => Welligkeit kleiner

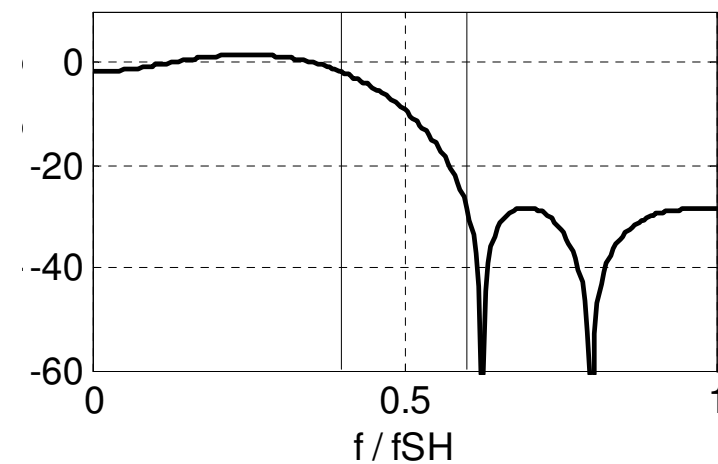
Beispiel 3:

$$N=8 \quad \tilde{f}_{1E}=0,4 \quad \tilde{f}_{2A}=0,6 \quad \underline{w} = [1 \ 5]$$

Amplitudengang



Amplitudengang [dB]

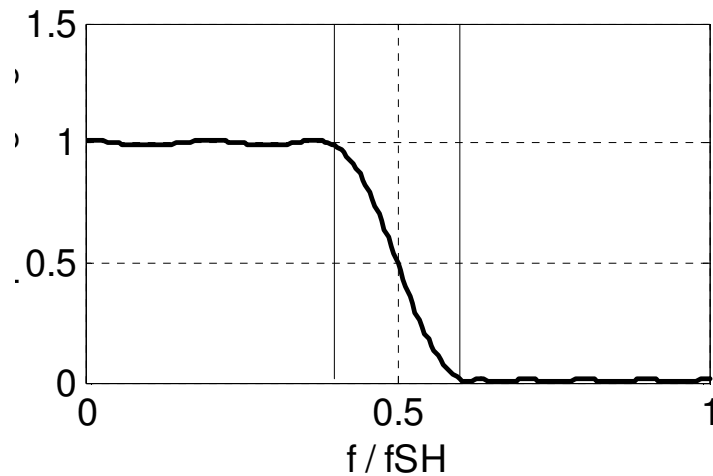


Fazit: Hohe Gewichtung im Vorgabebereich 2 reduziert Fehler in 2 zu Lasten von Vorgabebereich 1

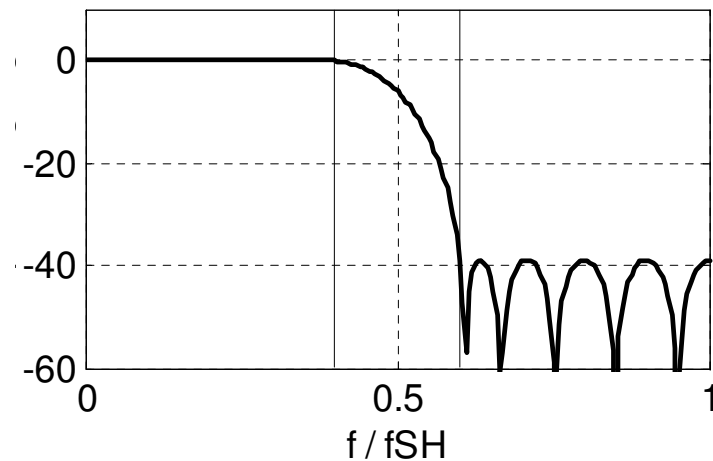
Beispiel 4:

$$N=20 \quad \tilde{f}_{1E}=0,4 \quad \tilde{f}_{2A}=0,6 \quad \underline{w} = [1 \ 1]$$

Amplitudengang [dB]



Amplitudengang [dB]

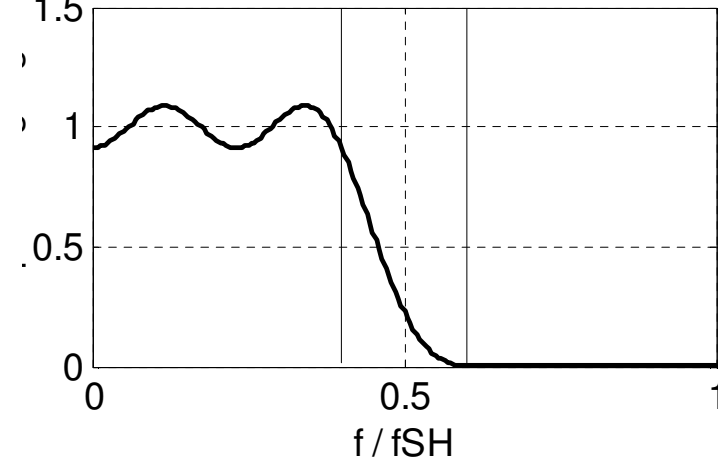


Fazit: Filterordnung größer => Welligkeit kleiner

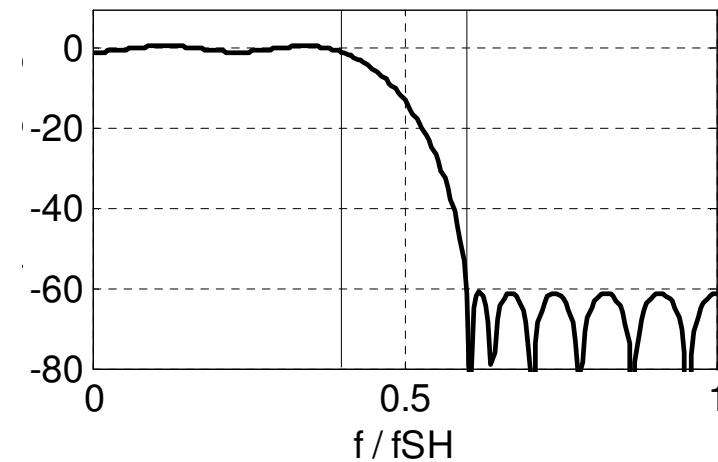
Beispiel 5:

$$N=20 \quad \tilde{f}_{1E}=0,4 \quad \tilde{f}_{2A}=0,6 \quad \underline{w} = [1 \ 100]$$

Amplitudengang [dB]



Amplitudengang [dB]



Fazit: Hohe Gewichtung im Vorgabebereich 2 reduziert Fehler in 2 zu Lasten von Vorgabebereich 1

MATLAB-Programm**% Parameter**

```

N=8;                % Filterordnung
OmegaD = 0.4;        % Endfrequenz Durchlassbereich
OmegaS = 0.6;        % Anfangsfrequenz Sperrbereich
                    % beide bezogen auf
                    % Shannonfrequenz

```

% Amplitudengangsvorgabe in den Durchlassbereichen

```

FS=[0 OmegaD OmegaS 1]; % 2 Vorgabebereiche
AS=[1 1 0 0];           % Amplitudenwerte
                    % (Geradenendpunkte)
W=[1 100];              % Gewichtung in
                    % 2 Durchlassbereichen

```

% Filterberechnung

```

b=remez(N,FS,AS,W);     % Berechnung Filter-
                    % Koeffizienten bi
a=[1];                  % FIR-Filter => A(z)=1

```

% Amplitudengang berechnen

```

Omega=linspace(0,1,201); % Frequenzachse
                    % bez. auf Shannonfrequ.
                    % (pos. + neg. Bereiche)
amp=abs(freqz(b,a,pi*Omega));
                    % A(Omega) für
                    % berechnetes Filter

```

% Amplitudengang zeichnen

```

plot(Omega,amp,'r');
axis([0 1 0 1.5]);
ylabel('Amplitudengang');
xlabel('Omega/OmegaShannon');
grid;

```

7.2.3 Beispiel BandsperreDaten für alle Beispiele:

$$\begin{aligned}
 \tilde{f}_{1A} &= 0 & \tilde{f}_{1E} &= 0,2 \\
 \tilde{f}_{2A} &= 0,3 & \tilde{f}_{2E} &= 0,5 \\
 \tilde{f}_{3A} &= 0,6 & \tilde{f}_{2E} &= 1,0
 \end{aligned}$$

Fehlergewichtungsvektor $\underline{w}=[1 \ 1 \ 1]$

Für MATLAB:

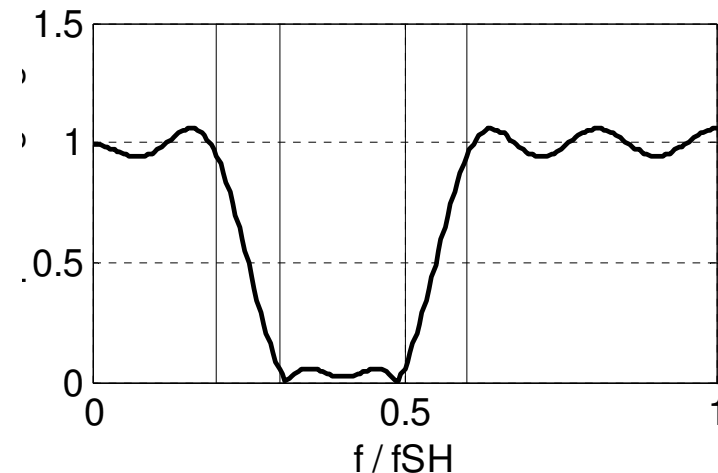
FS = [

AS = [

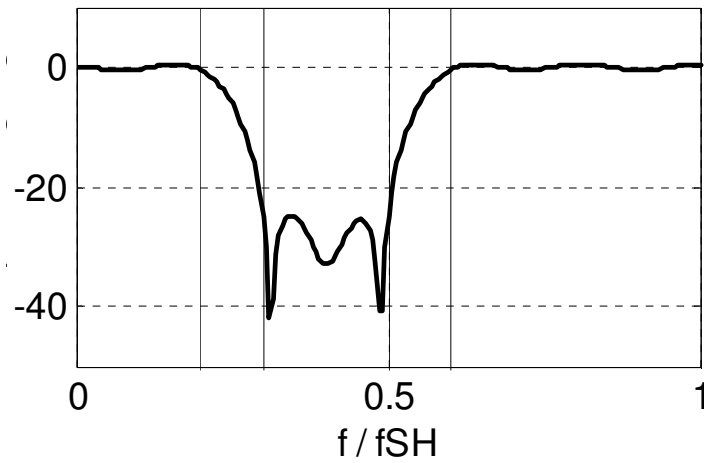
Beispiel 1:

Zahl Filtertaps N=20

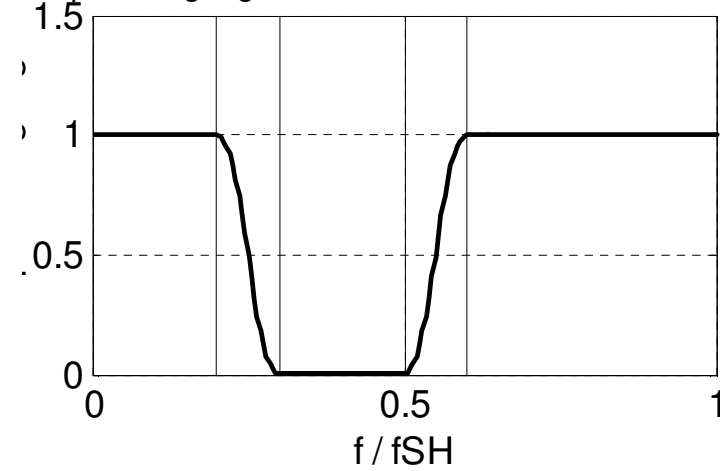
Amplitudengang



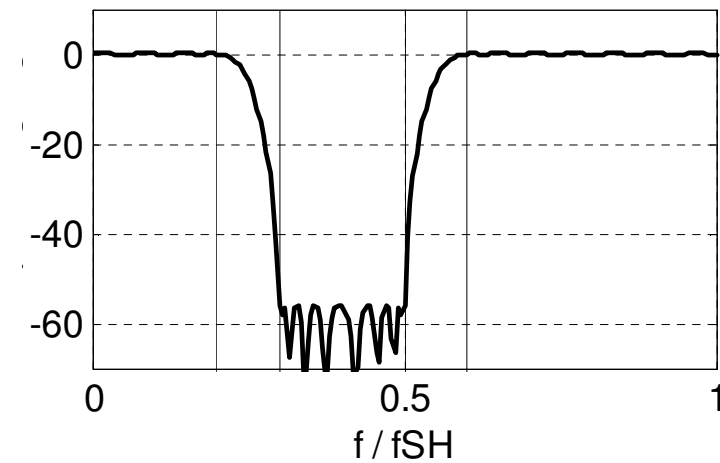
Amplitudengang [dB]

Beispiel 1:Zahl Filtertaps $N=40$

Amplitudengang



Amplitudengang [dB]



7.3 Vergleich IIR/FIR-Systeme

	IIR	FIR
Entwurf allg.	<u>Glatt:</u> BW-Filter + Bilinertransformation <u>Rippel:</u> TS, Ellip. Filter + Bilinertransformation	<u>Glatt:</u> Fenstermethode (Fourierreihe mit Fensterung der Koeffizienten) <u>Rippel:</u> Parks-Mc-Clellan („Equi-Rippel“)
Realisierungsaufwand	Niedrig: Typ. 8. Ordnung -> 2x8 MAC ^{*)} - Operationen	Hoch: Typ 100 Taps -> 100 MAC ^{*)} - Operationen
Abhängigkeit Realisierungsaufwand von Δt	nicht abhängig	in 1. Näherung prop. $1/\Delta t$
strukturstabil	Nein	Ja
linearphasig	Nein	Möglich (symm. Koeff.)
minimalphasig ^{**)}	Möglich	Nein

^{*)} Multiply and Accumulate

^{**)} Minimal mögliche Phasenverzögerung für vorgegebenen Amplitudengang (günstig für RET-Anwendungen)

Spezialverfahren:

- Notch-Filter (nur IIR)
=> Strauss-Artikel, Praktikum
- Analyse/Synthese-Filter für Datenkompression (nur FIR)
=> Übung 11

Lernziele

- Sie können anhand der DGL FIR-Systeme von IIR-Systemen unterscheiden. Ferner erkennen Sie linearphasige Systeme anhand der Filterkoeffizienten.
- Sie wissen, dass die Gruppenlaufzeit eines linearphasigen FIR-Filters der halben Ordnung (Ordnung=Tap- 1) entspricht.
- Sie kennen die DGL-Formeln für „Mittelwertfilter“ und „Primitiv-Differenzierer“.
- Sie können anhand eines Ersatzschaltbildes mit Hilfe einer Frequenzgangsberechnung das Prinzip des noise-shapings von Sigma-Delta-Wandlern erklären.
- Sie können den Frequenzgang einfacher FIR-Filter von Hand berechnen.
- Sie wissen, wie man FIR-Koeffizienten nach dem Verfahren der Impulsantwort berechnet.
- Sie können das Prinzip des Filterentwurfs mit der Fenstermethode skizzieren. Sie können die Wirkung einer anschließenden Gewichtung (Fensterung) der Filterkoeffizienten erläutern.
- Sie können die Grundgedanken des Verfahrens von Parks-Mc-Clellan, insbesondere auch die Form der Spezifikation erläutern.
- Sie können FIR-Filter und IIR-Filter in wichtigen Punkten miteinander vergleichen und können die wichtigsten Entwurfsverfahren benennen.
- Sie können unter Vorlage der wichtigsten MATLAB-Befehle einen MATLAB Skript-File schreiben, der FIR-Filterkoeffizienten nach der Fenstermethode bzw. nach dem Verfahren von Parks-Mc-Clellan bestimmt, sowie den Amplitudengang des gefundenen Filters anschließend in natürlichen Frequenzen grafisch darstellt.