



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

Digitale Signalverarbeitung

---

## Labor Nr. 3: FIR Filter

---

*Autoren:*

Tommy JAHNKE  
Nils PARCHE

*Professor:*

Prof. Dr. VOLLMER

21. November 2017

---

# Inhaltsverzeichnis

<b>1</b>	<b>Beschreibung</b>	<b>1</b>
<b>2</b>	<b>Attachments</b>	<b>2</b>
2.1	A1 Tiefpassentwurf mit <code>fir()</code> . . . . .	2
2.2	A2 Tiefpassentwurf mit <code>firpm()</code> . . . . .	5
2.3	B Bandpass-Filterentwurf . . . . .	9
2.4	C1 Analoge Übertragungscharakteristik des DSK Boards . . . . .	10
2.5	C2 Echtzeit-Festkomma-Implementierung des FIR-Filters . . . . .	10
2.6	C3 Vergleich des Amplitudengangs vom FIR-Filter Matlab - DSK Board . . . . .	10
2.7	D Profiling FIR-ISR . . . . .	10
2.8	E Weichenfilter Transformation mit $h_{TP} \rightarrow h_{HP}$ ..... . . . .	10
2.9	F Weichenfilter Amplitudengang Hoch- und Tiefpass . . . . .	10
2.10	G Weichenfilter Transformation mit $h_{TP} \rightarrow h_{HP}$ ..... . . . .	10
<b>3</b>	<b>Fazit</b>	<b>12</b>

---

# 1 Beschreibung

Die Labordurchführung wurde nach der Praktikumsbeschreibung „FIR Filter Implementierung in MATLAB und in C“ bearbeitet. In dieser Beschreibung wird davon ausgegangen, dass die Laborbeschreibung vorliegt.

In dem nachfolgenden Bericht wird der FIR-Filter theoretisch untersucht und als  $h_{TP}$ ,  $h_{HP}$  und  $h_{BP}$  auf einem DSP Implementiert und untersucht.

## 2 Attachements

### 2.1 A1 Tiefpassentwurf mit fir()

Mit der Matlab Funktion `fir()` ist ein FIR-Tiefpassfilter zu entwerfen. Als Defaulteinstellung wird als Fensterfunktion das Hamming-Fenster verwendet. Um die geforderten Grenzwerte einzuhalten muss zunächst die Filterordnung mit dem M-File `Kaiser_Order_01.m` bestimmt werden. Die Koeffizienten werden mit dem M-File `fir_1.m` gemäß Listing 2 bestimmt. Außerdem wird der Amplitudengang ( $x$ =normiert auf  $F_s/2$ ), das Zeitsignal und der Frequenzgang vor sowie nach dem Filter in einem Diagramm ausgegeben. In den Abbildungen 2 und 3 kann gut erkannt werden, dass die ripple im Pass- und Stopband eingehalten werden. Die normierten Filterkoeffizienten (normiert auf  $\pm 1$ ) müssen für die spätere Implementierung in den DSP auf 16-Bit Integer werte angepasst werden. Dazu werden die Koeffizienten mit einem Korrekturfaktor versehen. In Abbildung 1 sind die Änderungen von Listing 2 aufgeführt.

Korrekturwert maximal  $1 \approx 32767 \rightarrow 1\text{-Bit Vorzeichen} + 15\text{-Bit Wertebereich}$ .

$$b_k(x) = b(x) * 2^{15} - 1 \quad (1)$$

Parameter	Wert
Eckfrequenz Durchlassbereich	1800 Hz
Eckfrequenz Sperrbereich	2600 Hz
Maximaler Ripple im Durchlassbereich	0.5 db
Minimale Sperrdämpfung	40 db
Abtastfrequenz	8000 Hz

```

1 N=23; %FIR1 requires filter order (N) to be EVEN when gain = 1 at Fs/2.
2 % Normierte Eckfrequenz im Durchlassbereich auf Fs/2. 1800 Hz/4000Hz = 9/20
3 % = 0.45
4 % Normierte Eckfrequenz im Sperrbereich auf Fs/2. 2600 Hz/4000 Hz = 13/20 = 0.65
5 W=(9/20); %Specify TP filter with stop band
6 B=fir1(N,W,'DC-1') %Design FIR Filter using default Hamming window.
7 correction = 32767; % Correction for 16-bit integer normalized 1
8 B_correction =int16(B*correction) %cast B to 16 bit short Int
9 %create header file fir_coef.h (FIR filter coefficients)

```

Listing 1: `fir_2a.m` Matlab-File Auszug - Tiefpassfilter Ordnung 23

```

1 #define N 24
2 short h[N]={
3     -38,    73,    123,   -145,   -422,    167,   1080,    126,
4     -2332, -1431,  5700,  13482,  13482,  5700,  -1431, -2332,
5     126,   1080,   167,   -422,   -145,    123,    73,    -38,
6 };

```

Listing 2: FIR-Filter Koeffizienten Ordnung 23

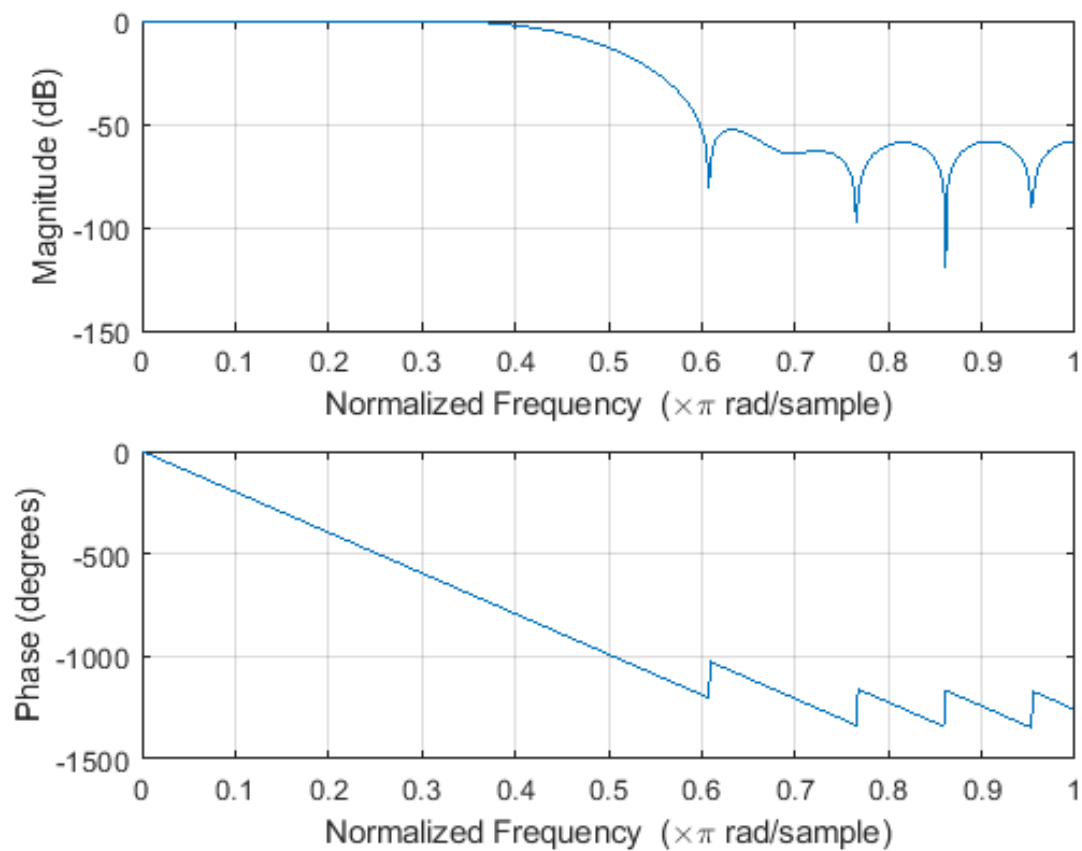


Abbildung 1: Amplituden und Phasengang - FIR-Filter Tiefpass

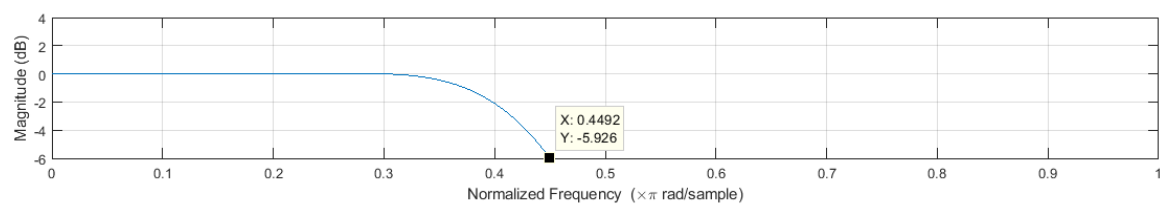


Abbildung 2: Amplitudengang skaliert auf das Passband

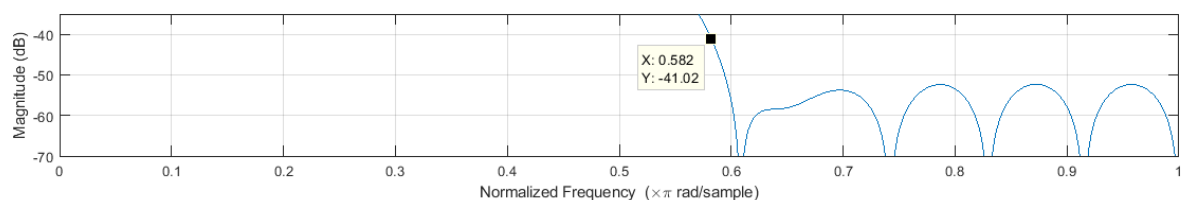


Abbildung 3: Amplitudengang skaliert auf das Stopband

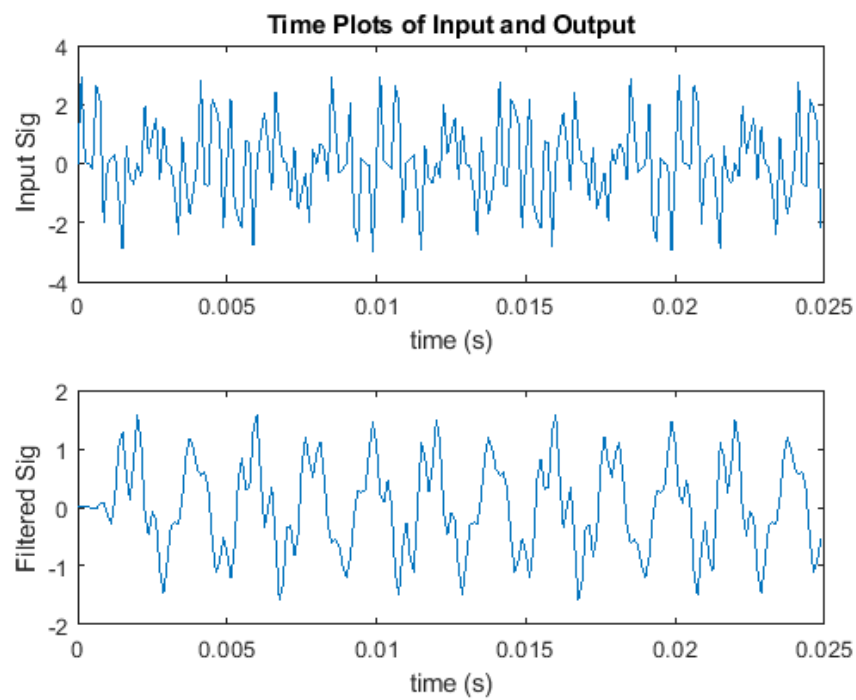


Abbildung 4: Eingangs- und gefiltertes Ausgangszeitensignal - FIR-Filter Tiefpass

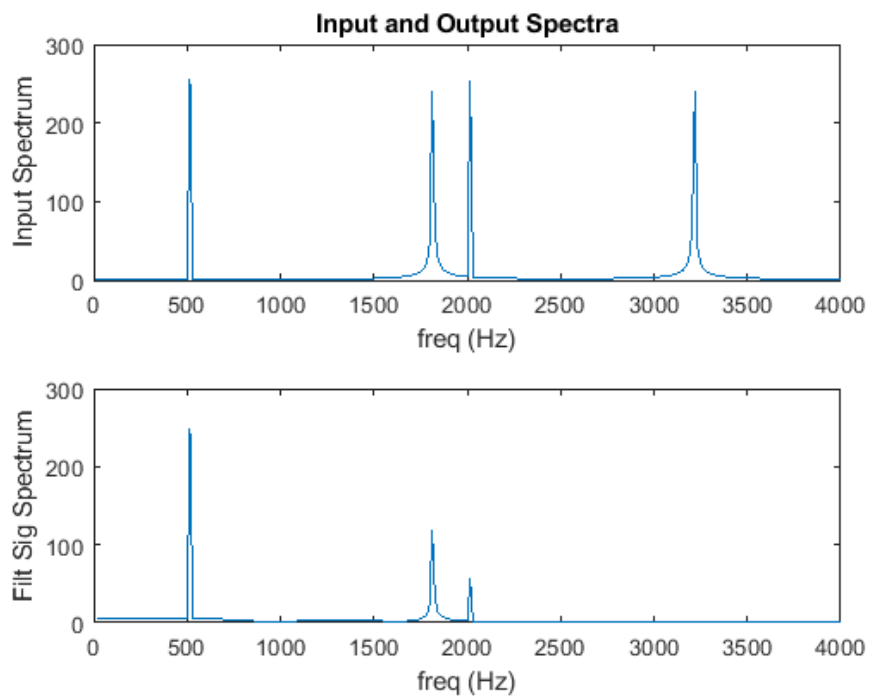


Abbildung 5: Eingangs- gefiltertes Ausgangs Frequenzspektrum

## 2.2 A2 Tiefpassentwurf mit firpm()

Alternativ zur Funktion fir1() soll jetzt die Funktion firpm() verwendet werden um einen FIR-Tiefpassfilter in MATLAB zu entwerfen. In Code-Listing 3 sind die Funktionsparameter beschrieben. Die Koeffizienten müssen wie in Kapitel 2.1 angepasst werden.

```

1 % fir_2b.m
2 % FIR filter design example using the MATLAB firpm function
3 % Autor: Nils Parche, 19.11.2017
4
5 Fs=8e3; %Specify Sampling Frequency
6 Ts=1/Fs; %Sampling period.
7 Ns=512; %No of time samples to be plotted.
8 F=[1800 2600]; %Cutoff frequenzcy
9 A=[1 0]; % Desired amplitude
10 RP = 0.5; % Passband ripple
11 RS = 40; % Stopband ripple
12 DEV = [(10^(RP/20)-1)/(10^(RP/20)+1) 10^(-RS/20)];
13
14 [N, F0, A0, W] = firpmord(F, A, DEV, Fs);
15 N = N+2; % Korrektur der Filterkoeffizienten um die Sperrdämpfung zu erreichen
16 B=firpm(N, F0, A0, W) %Design FIR Filter using default Hamming window.
17 correction = 32767;
18 %B_correction = cast((B*correction), 'uint16') %cast B to 16 bit short Int
19 B_correction = floor(B*correction);

```

Listing 3: fir\_2b.m Matlab-File Auszug - Tiefpassfilter Ordnung 16/18

```

1 #define N 17
2 short h[N]={
3     188,   -728,   -820,   1152,   1017,  -2848,  -1258,  10203,
4     17728, 10203,  -1258,  -2848,   1017,   1152,   -820,   -728,
5     188,};

```

Listing 4: FIR-Filter Koeffizienten Ordnung 16

Mit der Koeffizientenanzahl, die von der Funktion firpmord zurückgeliefert wird kann die geforderte Sperrdämpfung von 40 db nicht eingehalten werden. Der Amplitudengang ist in Abbildung 6, 7 und 8 dargestellt. **Dieser Zustand ist uns erst in der Versuchsnachbereitung aufgefallen, sodass wir alle folgenden Kapitel mit den Filterkoeffizienten der Filterordnung 16 bearbeitet haben.** In den nachfolgenden Kapiteln werden wir diesen Zustand nicht weiter behandeln.

Um die geforderten Parameter dennoch einhalten zu können müsste man die Koeffizienten um 2 erhöhen. Dadurch würde es ein Filtersystem 18 Ordnung ergeben und die in den Abbildungen 9, 10 und 11 gezeigten Amplitudenverläufe zeigen.

Des weiteren ist zu beobachten, dass die Eckfrequenz des Passbereiches sich auf etwa  $\approx 2.15$  kHz verschoben hat. Die Eckfrequenz des Sperrbereiches allerdings fest auf 2.6 kHz geblieben ist.

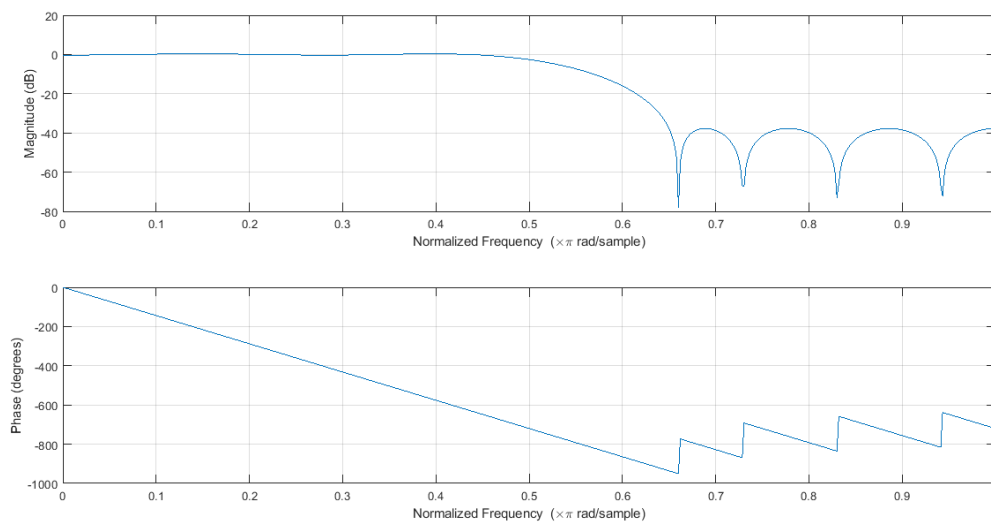


Abbildung 6: Amplituden und Phasengang - FIR-Filter Tiefpass - Ordnung: 16

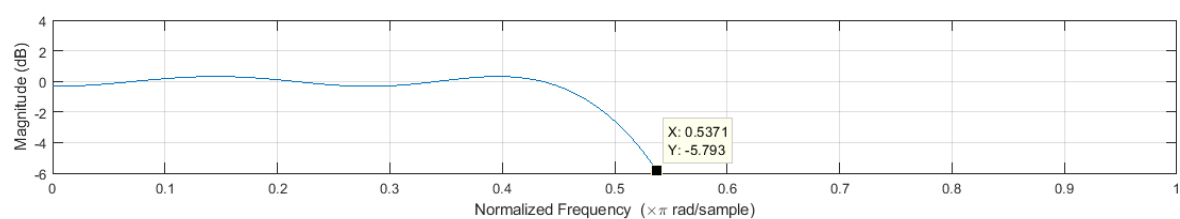


Abbildung 7: Amplitudengang skaliert Passband FIR-Filter Tiefpass - Ordnung: 16

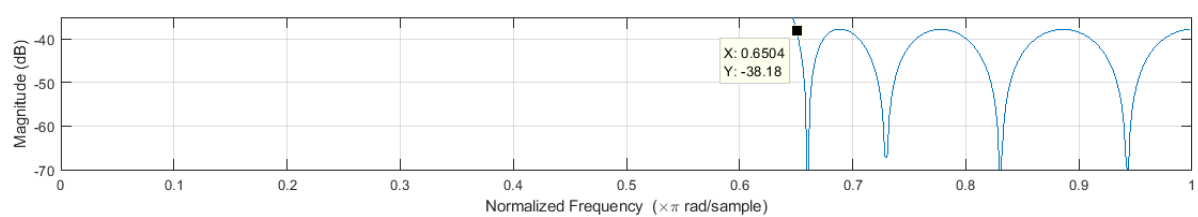


Abbildung 8: Amplitudengang skaliert Stopband FIR-Filter Tiefpass - Ordnung: 16



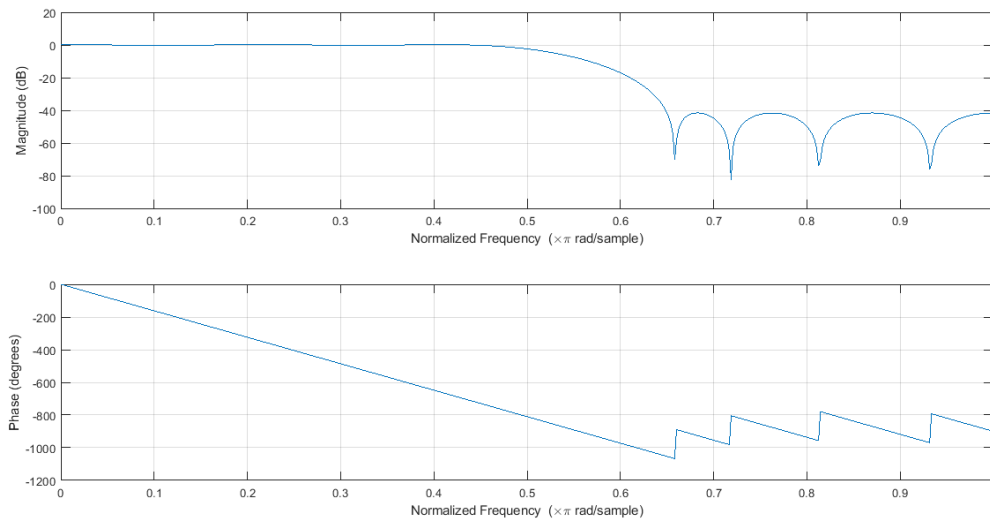


Abbildung 9: Amplituden und Phasengang - FIR-Filter Tiefpass - Ordnung: 18

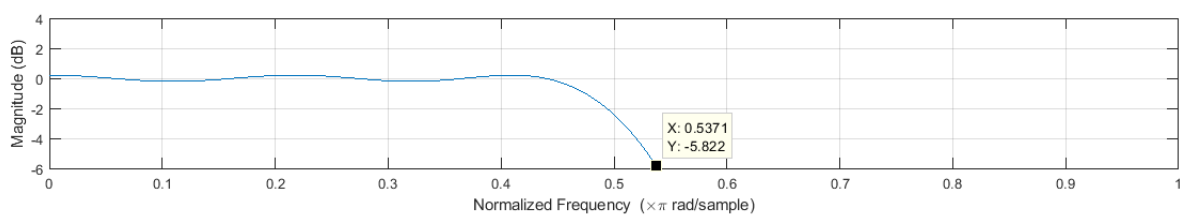


Abbildung 10: Amplitudengang skaliert Passband FIR-Filter Tiefpass - Ordnung: 18

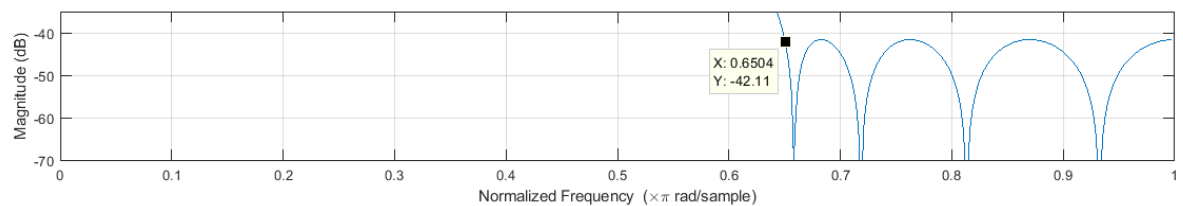


Abbildung 11: Amplitudengang skaliert Stopband FIR-Filter Tiefpass - Ordnung: 18

```

1 #define N 19
2 short h[N]={
3     291,   538,  -523,  -724,  1219,  1030, -2887, -1247,
4     10217, 17712, 10217, -1247, -2887,  1030,  1219,  -724,
5     -523,   538,   291,};

```

Listing 5: FIR-Filter Koeffizienten Ordnung 18

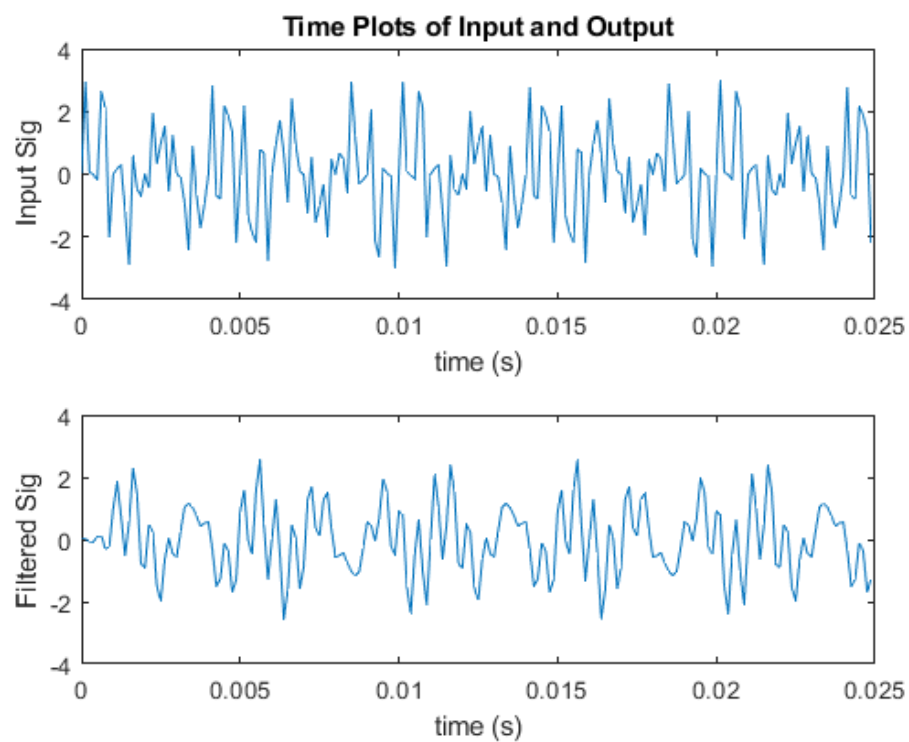


Abbildung 12: Eingangs- und gefiltertes Ausgangszeit-signal - FIR-Filter Tiefpass

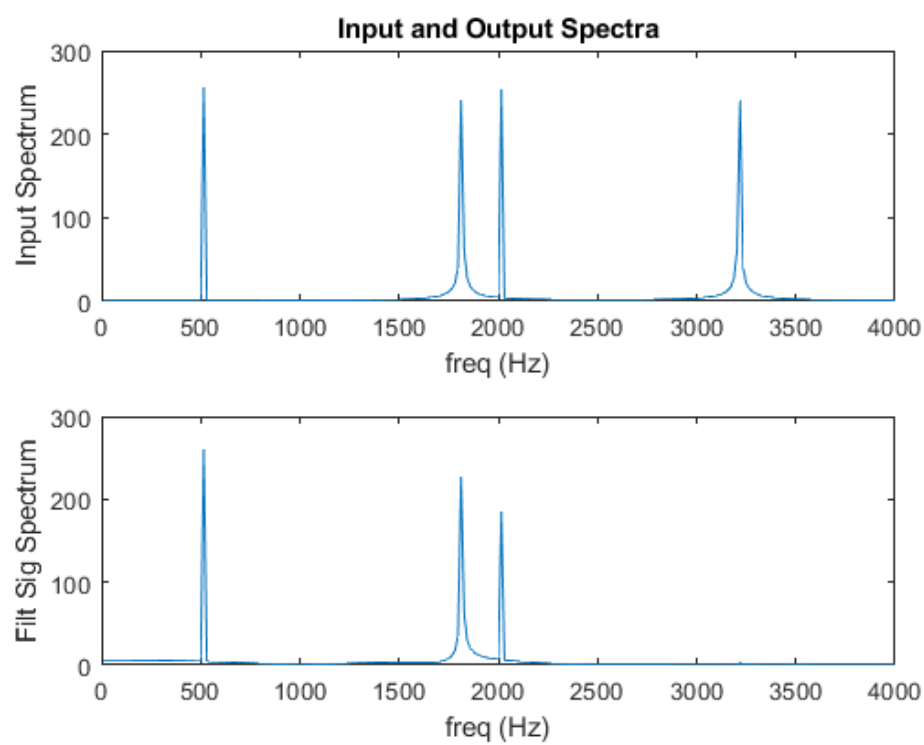


Abbildung 13: Eingangs- gefiltertes Ausgangs Frequenzspektrum

## 2.3 B Bandpass-Filterentwurf

```

1 % fir_3.m
2 % FIR filter design example using the MATLAB firpm function
3 % Autor: Nils Parche, 19.11.2017
4
5 Fs=8e3; %Specify Sampling Frequency
6 Ts=1/Fs; %Sampling period.
7 Ns=512; %No of time samples to be plotted.
8 F=[500 800 2400 2700]; %Cutoff frequenzcy
9 A=[0 1 0]; % Desired amplitude
10 RP = 0.4; % Passband ripple
11 RS = 40; % Stopband ripple
12 DEV = [10^(-RS/20) (10^(RP/20)-1)/(10^(RP/20)+1) 10^(-RS/20)];
13
14 [N, F0, A0, W] = firpmord(F, A, DEV, Fs);
15 N = N+5;
16 B=firpm(N, F0, A0, W) %Design FIR Filter using default Hamming window.
17 %create header file fir_coef.h (FIR filter coefficients)
18 correction = 32767;
19 B_correction =uint16(B*correction) %cast B to 16 bit short Int

```

Listing 6: fir\_3.m Matlab-File Auszug - Bandpassfilter Ordnung 45

```

1 #define N 51
2 short h[N]={
3     -228,    -67,    267,    202,    20,    423,    188,    -342,
4         53,   -186,   -929,   -236,   131,   -595,    462,   1542,
5        218,    709,   1998,   -968,  -2072,    306,  -4446,  -8341,
6       4330, 15371,   4330,  -8341,  -4446,    306,  -2072,   -968,
7       1998,    709,    218,   1542,    462,   -595,    131,   -236,
8      -929,   -186,    53,   -342,    188,    423,    20,    202,
9        267,    -67,   -228,};

```

Listing 7: FIR-Filter Koeffizienten Ordnung 45

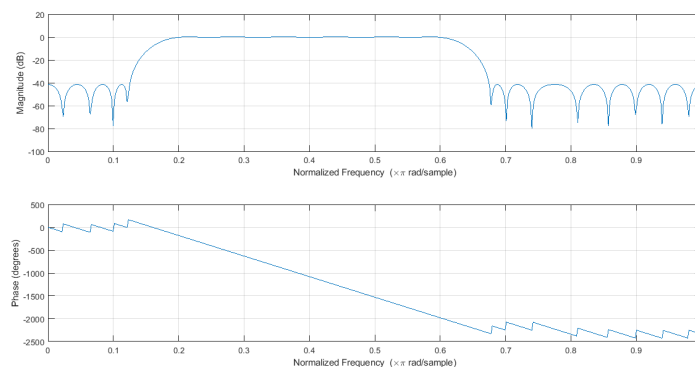


Abbildung 14: Amplituden und Phasengang - FIR-Filter Bandpass

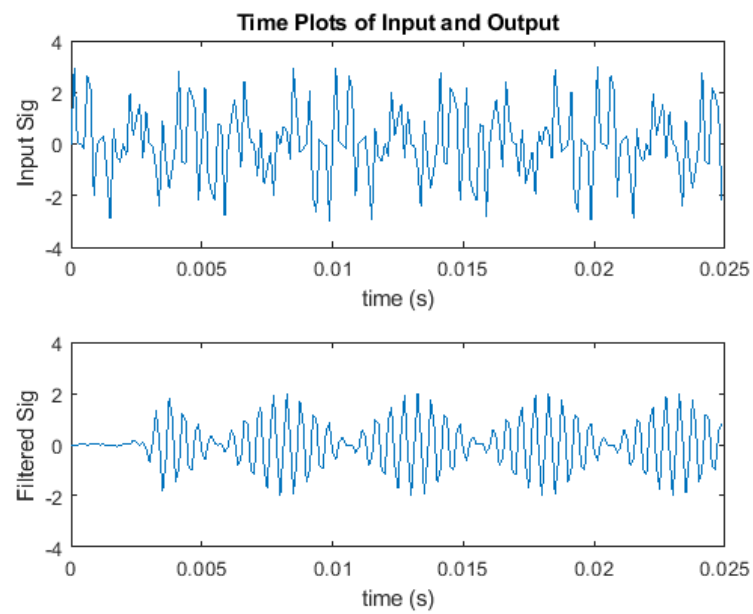


Abbildung 15: Eingangs- und gefiltertes Ausgangszeit-signal - FIR-Filter Bandpass

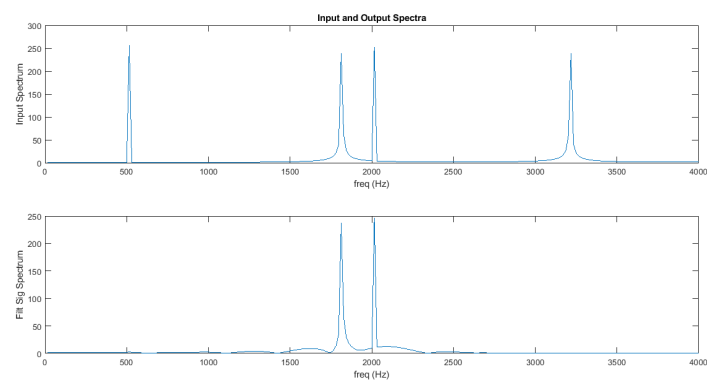


Abbildung 16: Eingangs- gefiltertes Ausgangs Frequenzspektrum

- 
- 2.4 C1 Analoge Übertragungscharakteristik des DSK Boards**
  - 2.5 C2 Echtzeit-Festkomma-Implementierung des FIR-Filters**
  - 2.6 C3 Vergleich des Amplitudengangs vom FIR-Filter Matlab  
- DSK Board**
  - 2.7 D Profiling FIR-ISR**
  - 2.8 E Weichenfilter Transformation mit  $h_{TP} \rightarrow h_{HP}$ .....**
  - 2.9 F Weichenfilter Amplitudengang Hoch- und Tiefpass**
  - 2.10 G Weichenfilter Transformation mit  $h_{TP} \rightarrow h_{HP}$ .....**

### 3 Fazit

FIR-Filter:

- Phase: Linearer abfallender Verlauf Zwischen zwei Polstellen. An den Polstellen sind Phasensprünge zu beobachten.
- Filterkoeffizienten sind immer  $N+1$ , wenn  $N$  die Ordnung beschreibt.
- Die Eckfrequenz eines digitalen Filters liegt bei -6db.
- 16-Filterordnung Sperrdämpfung zu gering, Theorie Matlab mit 18 Ordnung Sperrdämpfung wird erreicht, im DSP sollte verhalten ähnlich sein.