

# Kommunikationstechnik II – Wintersemester 08/09

---

Prof. Dr. Stefan Weinzierl

## Musterlösung: 4. Aufgabenblatt

Lösung in der Rechenübung am 11.12.2008

### 1. Aufgabe: Up-/Downsampling

Die Abtastfolge  $x[n]$  wird mit dem Faktor  $M$  unter- und dem Faktor  $L$  überabgetastet.



a) Stellen Sie die beiden Blockdiagramme mit Hilfe eines analytischen Ausdrucks dar. Wie groß ist in beiden Fällen die neue Abtastrate  $f_s$  und das Abtastintervall  $T_s$ ?

Im Fall von Downsampling:

$$x[n] = x(nT_s) \rightarrow y_d[n] = x(nMT_s)$$
$$f_s = \frac{1}{T_s} \rightarrow f_s' = \frac{f_s}{M} = \frac{1}{MT_s}$$
$$T_s' = MT_s = \frac{M}{f_s}$$

Im Fall von Upsampling:

$$x[n] = x(nT_s) \rightarrow y_u[n] = \begin{cases} x\left(\frac{nT_s}{L}\right), & n = 0, \pm L, \pm 2L, \dots \\ 0, & \text{sonst} \end{cases}$$
$$f_s = \frac{1}{T_s} \rightarrow f_s' = Lf_s = \frac{L}{T_s}$$
$$T_s' = \frac{T_s}{L} = \frac{1}{Lf_s}$$

b) Stellen Sie die Über- und Unterabtastung in Matlab dar für ein Signal mit der Frequenz 500 Hz, das ursprünglich mit 9 kHz abgetastet wird. Es sei  $M = L = 3$ . Tipp: Füllen Sie bei der Überabtastung im Signalvektor Nullen für die neuen Abtastwerte ein.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Musterloesung zum 4. Aufgabenblatt
%
%-----%

% Aufgabe 1: Up-/Downsampling

%%%%%%%%% b) %%%%%%%%%

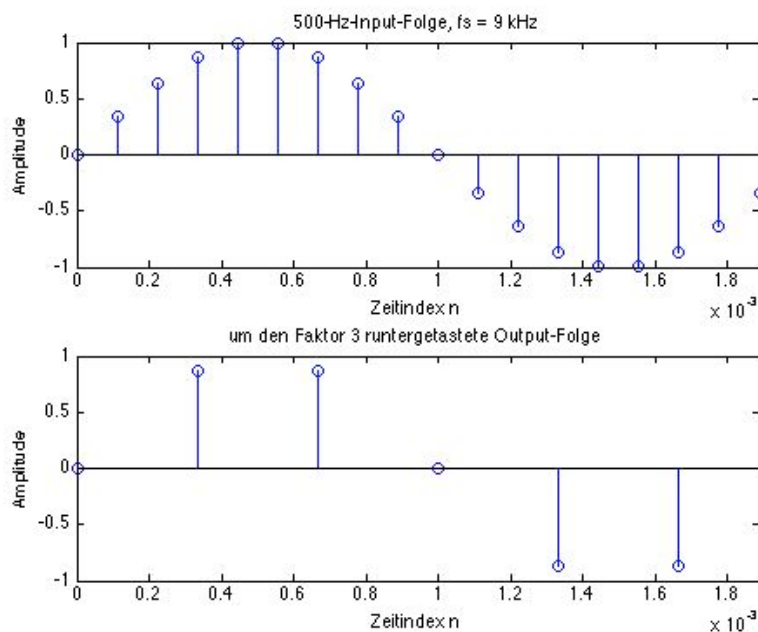
%-- Downsampling --%

% Erzeuge Sinusfolge
fs = 9000;
f = 500;
n = 0:1/fs:1-1/fs;
x = sin(2*pi*f*n);

% Erzeuge runtergetastete Folge mit zugehoerigem Zeitvektor n_M
M = 3;
y_down = x(1 : M : length(x));
n_M = 0:M/fs:1-M/fs;

% Plot des Input- und des Outputsignals ueber eine Periode des Signals
index_x = round(fs/f);
index_ydown = round((fs/M)/f);
figure
subplot(2,1,1)
stem(n(1:index_x), x(1:index_x));
title('500-Hz-Input-Folge, fs = 9 kHz')
xlabel('Zeitindex n'), ylabel('Amplitude')
axis([0 n(index_x) -1 1])
subplot(2,1,2)
stem(n_M(1:index_ydown), y_down(1:index_ydown));
title('um den Faktor 3 runtergetastete Output-Folge')
xlabel('Zeitindex n'), ylabel('Amplitude')
axis([0 n(index_x) -1 1])

```

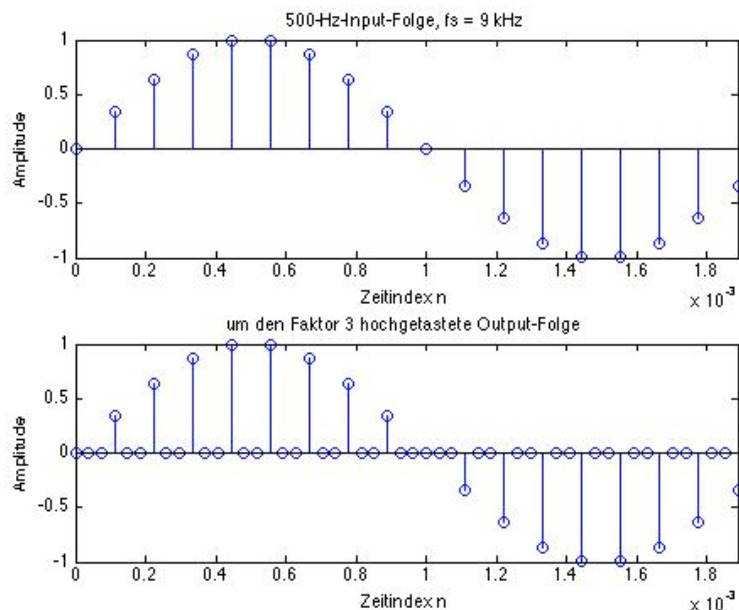


Nach der Unterabtastung liegt das Signal  $M$  mal weniger samples pro Periode vor. Um Aliasing zu vermeiden, muss ein Tiefpassfilter vorgeschaltet werden (siehe Aufgabe c)).

```
%-- Upsampling --%

% Erzeuge hochgetastete Folge mit zugehoerigem Zeitvektor n_L
L = 3;
y_up = zeros(1, L*length(x));
y_up(1:L:length(y_up)) = x;
n_L = 0:1/(fs*L):1-1/(fs*L);

% Plot des Input- und des Outputsignals ueber eine Periode des Signals
index_yup = round((fs*L)/f);
figure
subplot(2,1,1)
stem(n(1:index_x), x(1:index_x));
title('500-Hz-Input-Folge, fs = 9 kHz')
xlabel('Zeitindex n'), ylabel('Amplitude')
axis([0 n(index_x) -1 1])
subplot(2,1,2)
stem(n_L(1:index_yup), y_up(1:index_yup));
title(['um den Faktor 3 hochgetastete Output-Folge'])
xlabel('Zeitindex n'), ylabel('Amplitude')
axis([0 n(index_x) -1 1])
```



Nach der Überabtastung liegt das Signal  $L$  mal mehr samples pro Periode vor. Die zusätzlichen samples werden vorerst mit Nullen aufgefüllt, sie müssen jedoch anschließend durch Tiefpassfilterung zu Abtastwerten interpoliert werden (siehe Aufgabe c)).

c) Stellen Sie das ursprüngliche und die beiden neu abgetasteten Signale im Frequenzbereich dar (in Matlab oder als Skizze). Erklären Sie anhand des Ergebnisses, welche Filter zusätzlich nötig sind.

```

##### c) #####

% normierte FFT der Signale
X = abs(fft(x))/max(abs(fft(x)));
Y_down = abs(fft(y_down))/max(abs(fft(y_down)));
Y_up = abs(fft(y_up))/max(abs(fft(y_up)));

% Plots der FFTs
N = length(X);
N_M = length(Y_down);
N_L = length(Y_up);
f_indexx = 0:fs/N:fs-fs/N;
f_indexydown = 0:(fs/M)/N_M:(fs/M)-(fs/M)/N_M;
f_indexyup = 0:(fs*L)/N_L:(fs*L)-(fs*L)/N_L;

figure
subplot(3,1,1), plot(f_indexx,20*log10(X))
xlabel('Frequenz [Hz]'), ylabel('Betrag [dBFS]')
title('Betragsspektrum einer 500-kHz-Sinusfolge, fs = 9 kHz')
axis([0 fs -350 80])
subplot(3,1,2), plot(f_indexyup,20*log10(Y_up))
xlabel('Frequenz [Hz]'), ylabel('Betrag [dBFS]')
title('Betragsspektrum der um den Faktor 3 hochgetasteten Sinusfolge')
axis([0 fs*L -350 80])
subplot(3,1,3), plot(f_indexydown,20*log10(Y_down))
xlabel('Frequenz [Hz]'), ylabel('Betrag [dBFS]')
title('Betragsspektrum der um den Faktor 3 runtergetasteten Sinusfolge')
axis([0 fs/M -350 80])

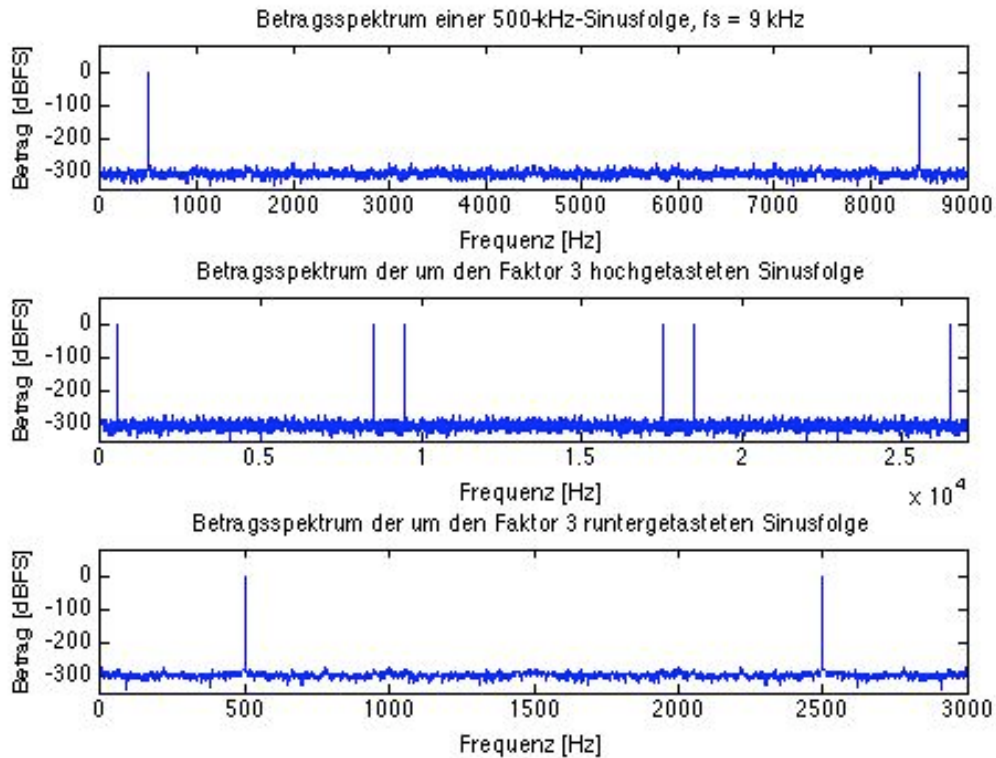
```

### Upsampling:

Durch die erste Abtastung des Signals mit 9 kHz tritt eine Spiegelfrequenz bei  $f_S - f$  auf, also bei 8,5 kHz. Wird dieses Signal nun überabgetastet, so ergeben zusätzliche Spiegelfrequenzen um jedes Vielfache der alten Abtastfrequenz.

Eine anschließende Tiefpassfilterung bei der halben alten Abtastfrequenz bewirkt im Frequenzbereich die Unterdrückung der entstandenen Spiegelfrequenzen zwischen  $f_{S\_alt}/2$  und  $f_{S\_neu} - f_{S\_alt}/2$ . Im Zeitbereich erfolgt die Interpolation zwischen den durch die eingefügten Nullen „getrennten“ samples. Das Tiefpassfilter wird deshalb Interpolations- oder Anti-Imaging-Filter genannt. Der Vorgang der Überabtastung mit anschließender Tiefpassfilterung wird Interpolation genannt.

Das bei der Rückwandlung in ein analoges Signal notwendige Rekonstruktionsfilter (siehe Aufgabenblatt 2) unterdrückt auch noch die Signalanteile zwischen  $f_{S\_alt}/2$  und  $f_{S\_neu}$ , es muss jedoch dank der Überabtastung nicht besonders steilflankig sein.



### Downsampling:

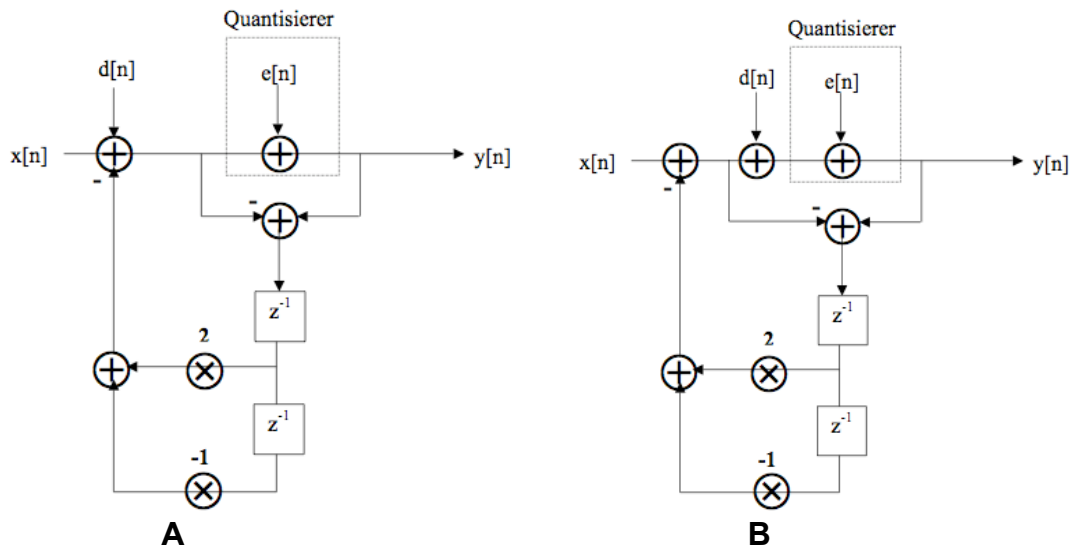
Beim Downsampling wird die Abtastrate und somit auch die Spiegelfrequenzen bei  $f_s - f$  in tiefere Frequenzbereiche verschoben. Um zu vermeiden, dass durch diese Aliasing auftritt, dass also  $f_{s\_neu} - f < f$  bzw.  $f_{s\_neu} < 2f$ , muss das Signal vor dem Downsampling mit einem Anti-Aliasing- oder Dezimationsfilter bandbegrenzt werden. Der Vorgang der Unterabtastung mit vorhergehender Tiefpassfilterung wird Dezimation genannt.

d) Nennen Sie die zwei Hauptgründe für Überabtastung.

Der erste Grund für Überabtastung liegt im Aufwand für den Entwurf des analogen Antialiasing-Filters. Dieses müsste im Normalfall sehr steiflankig sein, damit die gesamte Bandbreite des Nutzsignals bis zur Nyquistfrequenz genutzt werden kann. Wird die Abtastrate aber höher gesetzt, so verschiebt sich auch die Nyquistfrequenz zu einer höheren Frequenz, das Nutzband bleibt jedoch gleich. Es ist also ein flacheres Filter ausreichend, um trotzdem die gesamte Bandbreite des Nutzsignals zu übertragen. Ebenso kann das Rekonstruktionsfilter deutlich flacher ausfallen (siehe Aufgabe c)).

Der zweite Grund bezieht sich auf den Quantisierungsfehler. Dessen Rauschleistung ist – bei guter Aussteuerung und ausreichender Wortbreite – über das gesamte Spektrum gleichverteilt, gleichzeitig ist sie jedoch nicht abhängig von der Abtastfrequenz. Steigt also die Bandbreite des Übertragungssignals mit der Abtastrate, so verteilt sich die Rauschleistung über einen größeren Frequenzbereich. Nach erneuter (digitaler) Tiefpassfilterung resultiert ein größerer SNR, der mit ca. 3 dB pro Frequenzverdoppelung steigt.

## 2. Aufgabe: Noiseshaping



a) Stellen Sie die Differenzgleichungen für die beiden dargestellten Noiseshaping-Systeme auf und berechnen Sie jeweils die Signal- und die Rauschübertragungsfunktion.

### System A:

$$y[n] = x[n] + d[n] + e[n] - 2e_0[n-1] + e_0[n-2]$$

$$e_0[n] = x[n] + d[n] + e[n] - (x[n] + d[n]) = e[n]$$

somit ist:

$$y[n] = x[n] + d[n] + e[n] - 2e[n-1] + e[n-2]$$

im z-Bereich:

$$Y(z) = X(z) + D(z) + E(z) - 2z^{-1}E(z) + z^{-2}E(z) = X(z) + \underbrace{D(z) + E(z)}_{\text{Rauschen}} \underbrace{[1 - 2z^{-1} + z^{-2}]}_{\text{spektrale Formung}}$$

Rausch- und Signalübertragungsfunktion:

$$H_X(z) = \frac{Y(z)}{X(z)} = H_D(z) = \frac{Y(z)}{D(z)} = 1$$

$$H_E(z) = \frac{Y(z)}{E(z)} = 1 - 2z^{-1} + z^{-2} = (1 - z^{-1})^2 \quad \rightarrow \text{Noiseshaping 2. Ordnung}$$

### System B:

$$y[n] = x[n] + d[n] + e[n] - 2e_0[n-1] + e_0[n-2]$$

$$e_0[n] = x[n] + d[n] + e[n] - x[n] = d[n] + e[n]$$

somit ist:

$$y[n] = x[n] + d[n] + e[n] - 2d[n-1] + d[n-2] - 2e[n-1] + e[n-2]$$

im z-Bereich:

$$\begin{aligned} Y(z) &= X(z) + D(z) + E(z) - 2z^{-1}D(z) + z^{-2}D(z) - 2z^{-1}E(z) + z^{-2}E(z) \\ &= X(z) + D(z)[1 - 2z^{-1} + z^{-2}] + E(z)[1 - 2z^{-1} + z^{-2}] = X(z) + \underbrace{[D(z) + E(z)]}_{\text{Rauschen}} \underbrace{[1 - 2z^{-1} + z^{-2}]}_{\text{spektrale Formung}} \end{aligned}$$

Rausch- und Signalübertragungsfunktion:

$$H_X(z) = \frac{Y(z)}{X(z)} = 1$$

$$H_E(z) = \frac{Y(z)}{E(z)} = H_D(z) = \frac{Y(z)}{D(z)} = 1 - 2z^{-1} + z^{-2} = (1 - z^{-1})^2 \rightarrow \text{Noiseshaping 2. Ordnung}$$

b) Welches System ist sinnvoller? Begründen Sie Ihre Antwort.

Im System A wird zwar der Quantisierungsfehler spektral geformt, nicht aber das Dithersignal. Dieses wird genau wie das Nutzsignal mit  $H(z) = 1$  übertragen. Im System B hingegen wird auch das Dithersignal bei der Spektralformung berücksichtigt. Dieser Algorithmus wird also bevorzugt.

c) Stellen Sie die Rauschübertragungsfunktion des in b) gewählten Systems in Matlab dar.

```
##### b) #####

w = 0:2*pi/1024:2*pi;
H = 1-2*exp(-i*w)+exp(-2*i*w);

figure
plot(w(1:512),abs(H(1:512)),'r')
title('Noiseshaping zweiter Ordnung')
xlabel('Frequenz'), ylabel('Betrag des Rauschens')
axis([0 pi 0 4])
set(gca,'XTick',[0 pi/3 0.5*pi pi])
set(gca,'XTickLabel',{'0';'fs/6';'fs/4';'fs/2'})
grid on
hold on
plot(w(1:512),ones(1,512))
legend('spektrale Formung','H(z)=1','Location','NW')
```

