Digitale Signalverarbeitung HS13/FS14

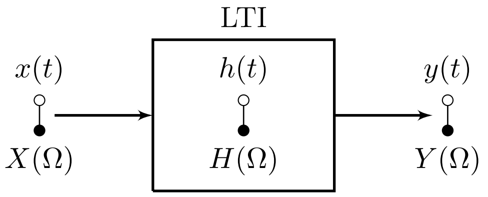
HSR Hochschule für Technik Rapperswil
Elektrotechnik

Rapperswil, 3. Mai 2016

Inhaltsverzeichnis

1 Abtastung and Rekonstruktion S.1

1.1 Analoge Signale



$\boxed{\Omega = 2\pi f}$ $[rad/s]$

Eingangssignal:

$$\boxed{\begin{aligned} X(\Omega) &= \int_{-\infty}^{\infty} x(t) \cdot e^{-j\Omega t}\, dt \\ x(t) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\Omega) \cdot e^{j\Omega t}\, d\Omega \end{aligned}}$$

Impulsantwort: $h(t)$

Übertragungsfunktion:

$$\boxed{H(\Omega) = \int_{-\infty}^{\infty} h(t) \cdot e^{-j\Omega t}\, dt}$$

Ausgangssignel:

$$\boxed{\begin{aligned} Y(\Omega) &= H(\Omega) \cdot X(\Omega) \\ y(t) &= \int_{-\infty}^{\infty} h(t-\tau) \cdot x(\tau)\, d\tau \end{aligned}}$$

Amplituden- und Phasenänderung:

$$\boxed{\begin{gathered} x(t) = e^{j\Omega t} \\ \Downarrow \\ y(t) = |H(\Omega)| \cdot e^{j\Omega t + j\, arg(H(\Omega))} \end{gathered}}$$

Linearität:

$$\boxed{\begin{gathered} x(t) = A_1 \cdot e^{j\Omega_1 t} + A_2 \cdot e^{j\Omega_2 t} \\ \Downarrow \\ y(t) = A_1 \cdot H(\Omega_1) \cdot e^{j\Omega_1 t} + A_2 \cdot H(\Omega_2) \cdot e^{j\Omega_2 t} \end{gathered}}$$

$$\boxed{\begin{gathered} X(\Omega) = 2\pi A_1 \delta(\Omega - \Omega_1) + 2\pi A_2 \delta(\Omega - \Omega_2) \\ \Downarrow \\ Y(\Omega) = 2\pi A_1 H(\Omega_1) \delta(\Omega - \Omega_1) + 2\pi A_2 H(\Omega_2) \delta(\Omega - \Omega_2) \end{gathered}}$$

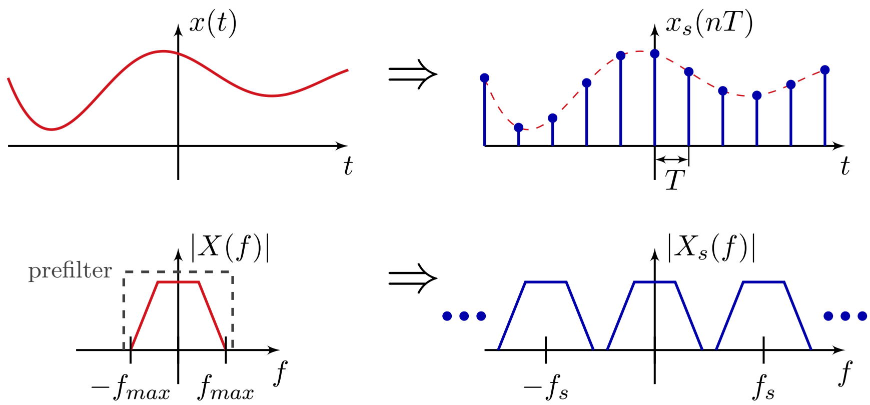
1.2 Abtasttheorem

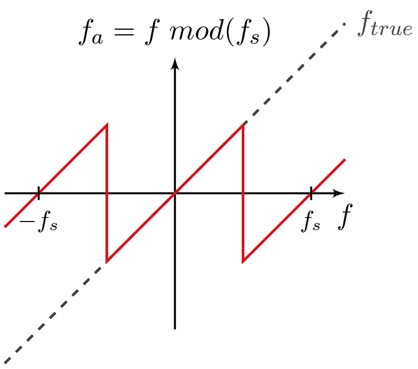
- Das Eingangssignal **muss bandbeschränkt** sein, damit es Aliasingfrei abgetastet werden kann $|X(f)| \approx 0 \quad$ für $\quad f > f_s/2$
- Wenn keine Überlappung herrscht, kann das Signal $x(t)$ mit einem idealen Tiefpassfilter (Eckfrequenz $f_c = f_s/2$) wieder herausgefiltert werden.

Nyquist-Shannon Abtasttheorem:

$\boxed{f_s \geq 2 \cdot f_{max}}$

Abtastfrequenz: $\quad f_s = \dfrac{1}{T}$





Rekonstruktion

Enthält das Signal Frequenzen ausserhalb des Nyquistbandes $\left[-\frac{f_s}{2}\,;\,\frac{f_s}{2}\right]$, so werden diese auf Frequenzen ins Nyquistband abgebildet $\rightarrow$ **Aliasing**.

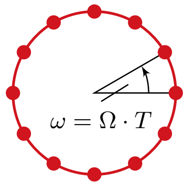
$$\boxed{f_a = f\ mod(f_s) = f - round\left(\frac{f}{f_s}\right) \cdot f_s}$$

Frequenzen mit gleichen Sampeln: $f \pm n \cdot f_s \quad n \in \mathbb{Z}$

1.3 Digitale Frequenz

Digitale Frequenz: $\boxed{\omega = \Omega \cdot T = 2\pi f T = \frac{2\pi f}{f_s}}$ $[rad/sample]$

$$x(t) = \mathrm{e}^{j2\pi f T} \quad \Rightarrow \quad x(nT) = \mathrm{e}^{j\omega n}$$

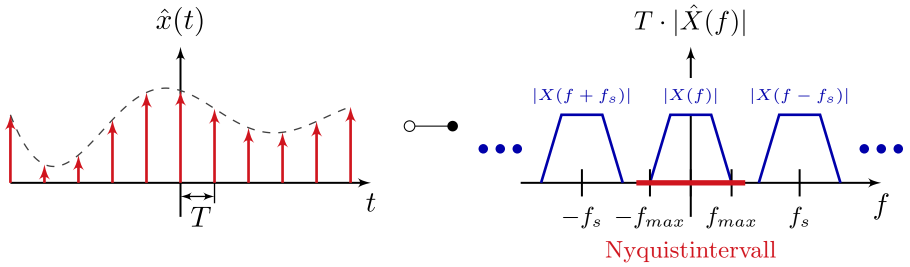


	Zeichen / Einheit		Nyquistintervall	Aliasing-Frequenzen
Natürliche Frequenz	f	$[\mathrm{Hz}] = [cycles/s]$	$-f_s/2 \ldots f_s/2$	$f \pm n \cdot f_s$
Kreisfrequenz	$\Omega = 2\pi f$	$[rad/s]$	$-\pi f_s \ldots \pi f_s$	$\Omega \pm n \cdot \Omega_s$
Digitale Frequenz	$\omega = 2\pi f / f_s$	$[rad/sample]$	$-\pi \ldots \pi$	$\omega \pm 2\pi n$
Normalisierte Frequenz	f/f_s	$[cycles/sample]$	$-1/2 \ldots 1/2$	$f/f_s \pm n$

1.4 Spektrum von abgetasteten Signalen

Discrete Time Fourier Transform (DTFT)

$$\boxed{\hat{x}(t) = \sum_{n=-\infty}^{\infty} x(nT) \cdot \delta(t - nT) \quad \circ\!\!-\!\!\bullet \quad \hat{X}(f) = \sum_{n=-\infty}^{\infty} x(nT) \cdot \mathrm{e}^{-j2\pi f T n} = \frac{1}{T} \sum_{m=-\infty}^{\infty} X(f - m f_s)}$$



$$\boxed{T \cdot \hat{X}(f) = X(f) \qquad \text{für} \quad \frac{-f_s}{2} \leq f \leq \frac{f_s}{2}}$$

1.4.1 Eigenschaften:

DTFT: Nicht zu verwechseln mit der Discrete Fourier Transform (DFT), die ein Spezialfall der DTFT ist.

Periodizität: $\hat{X}(f)$ ist eine periodische Funktion mit der Periodendauer $f_s = 1/T$. Dies ist auch aus der e-Funktion ersichtlich.

Fourierreihe: Das Spektrum des abgetasteten Signals $\hat{X}(f)$ kann auch als Fourierreihe intepretiert werden. Dabei wären die Samples $x(nT)$ die Fourierkoeffizienten.

$$\boxed{x(nT) = \frac{1}{f_s} \int_{-f_s/2}^{f_s/2} \hat{X}(f) \cdot \mathrm{e}^{j2\pi f T n}\, df = \frac{1}{2\pi} \int_{-\pi}^{\pi} \hat{X}(\omega) \cdot \mathrm{e}^{j\omega n}\, d\omega}$$

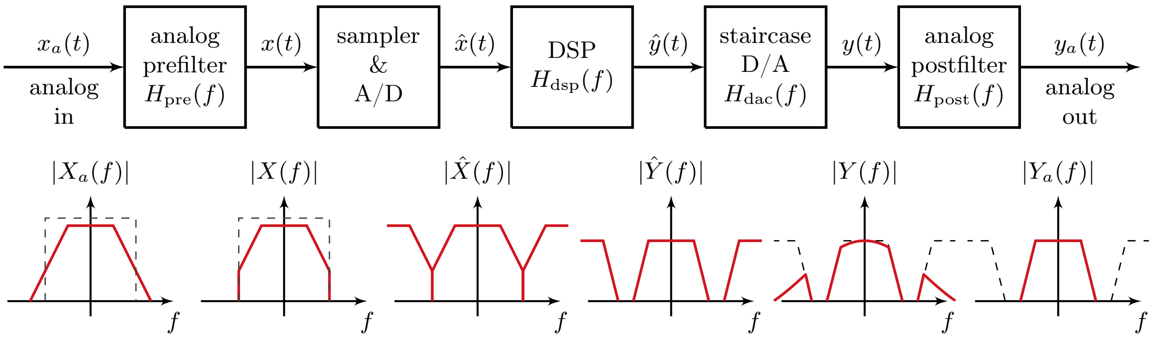
Nummerische Approximation: Das Spektrum des abgetasteten Signals $\hat{X}(f)$ kann auch als nummeriche Integralapproximation des Originalspektrums $X(f)$ betrachtet werden.

$$\boxed{X(f) = \int_{-\infty}^{\infty} x(t) \cdot \mathrm{e}^{-j2\pi f t}\, dt \approx \sum_{n=-\infty}^{\infty} x(nT) \cdot \mathrm{e}^{-j2\pi f T n} \cdot T = T \cdot \hat{X}(f)}$$

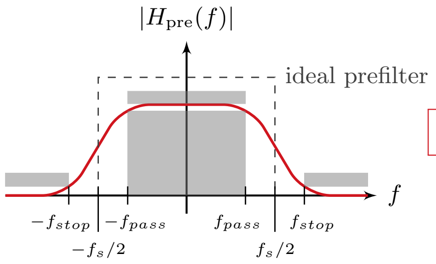
Praktische Approximation: Für die Praxis müssen noch zwei weitere Approximation gemacht werden. Die Samples müssen auf eine endliche Anzahl L beschränkt werden und die Anzahl Frequenzen ebenfalls (DFT, FFT).

$$\boxed{\hat{X}(f) \approx \hat{X}_L(f) = \sum_{n=0}^{L-1} x(nT) \cdot \mathrm{e}^{-j2\pi f T n}}$$

1.5 Prefilter und Postfilter



Antialiasing Prefilter: Das Antialiasing Prefilter ist notwentig, um das Eingangssignal bandzubeschränken. Die Stoppfrequenz und die minimale Stopdämpfung ist so zu wählen, dass die Aliasingeffekte so gering wie nötig sind. Die erste spektrale Kopie sollte dabei komplett im Stoppband liegen.



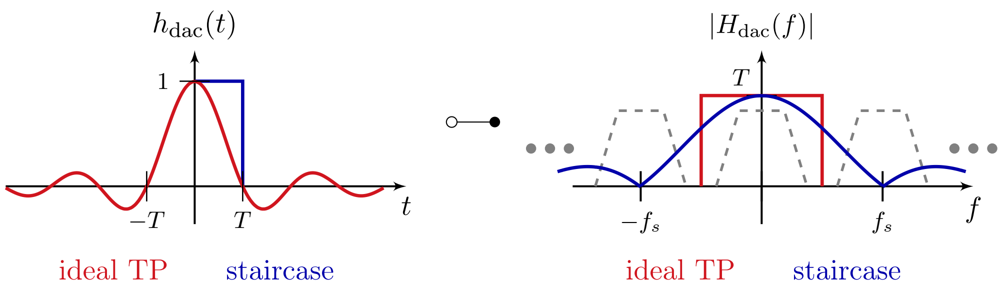
$$\boxed{f_{stop} = f_s - f_{pass}}$$

Ideales Tiefpassfilter zur Rekonstruktion:
Mit einem idealen Tiefpassfilter könnte das Originalspektrum wieder herausgefiltert werden. Die Übertragungsfunktion und Impulsantwort des Idealen Tiefpasses sind folgende:

$$\boxed{H_{dac}(f) = \begin{cases} T, & |f| \leq f_s/2 \\ 0, & \text{sonst} \end{cases}} \qquad \boxed{h_{dac}(t) = \frac{\sin(\pi\, t/T)}{\pi\, t/T} = \frac{\sin(\pi\, f_s\, t)}{\pi\, f_s\, t}}$$

Staircase Rekonstruktor: Der Staircase Rekonstruktor ist der am weitesten verbreitete. Seine Übertragungsfunktion und Impulsantwort sind folgende:

$$\boxed{H_{dac}(f) = T\,\frac{1}{j2\pi f}\cdot(1-\mathrm{e}^{-j2\pi fT}) = T\,\frac{\sin(\pi\, f\, T)}{\pi\, f\, T}\cdot \mathrm{e}^{-j\pi fT}} \qquad \boxed{h_{dac}(t) = \begin{cases} 1, & 0 \leq t \leq T \\ 0, & \text{sonst} \end{cases}}$$



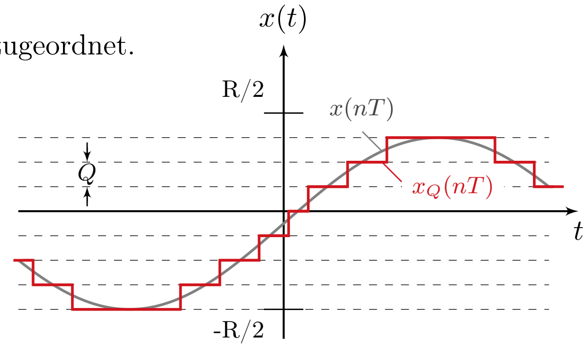
Digital Equalizer: Um die vom Staircase Rekonstruktor verursachte Abschwächung im Basisband zu kompensieren, wird vor der D/A-Wandlung das Spektrum vorverzerrt. Dabei handelt es sich um ein digitales und daher periodisches Filter. Das Filter entspricht gerade der Inversen des Staircase Rekonstruktor im Basisband.

$$\boxed{H_{eq}(f) = \frac{T}{H_{dac}(f)} = \frac{\pi\, f\, T}{\sin(\pi\, f\, T)}\cdot \mathrm{e}^{j\pi fT}} \qquad \text{für} \quad -\frac{f_s}{2} \leq f \leq \frac{f_s}{2}$$

Analog Postfilter: Das Analog Postfilter ist ein Tiefpass, der die höheren Spektralen Perioden unterdrückt.

2 Quantisierung S.61

2.1 Quantisierungsprozess



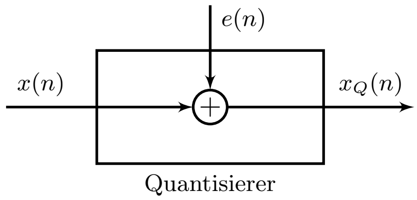
Bei der Quantisierung wird jedem Wert eine fixe Nummer zugeordnet.

- Wertebereich/Range: $-R/2 \dots R/2$
  $R/2 - Q$ ist der höchste Wert.
- Anzahl Quantisierungsintervalle: $L = 2^B$
- Anzahl Bits: B
- Quantisierungsschritt: $\boxed{Q = \frac{R}{2^B}}$
- Quantisierungsfehler: $\boxed{e(nT) = x_Q(nT) - x(nT)}$ $\quad -Q/2 \leq e \leq Q/2$

2.1.1 Quantisierungsfehler / Quantisierungsrauschen

Wenn das Signal:

- Wide-Amplitude (gleichmässig ausgesteuert)
- Wide-Band (Breitbandig)

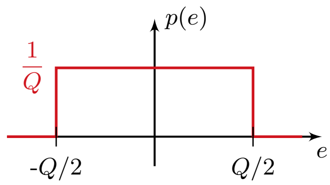


ist kann der Quantisierungsfehler als stationärer, gleichverteilter, mittelwertfreier, weisser Zufallsprozess modelliert werden. Mit diesem Modell wird der Quantisierungsfehler (nicht-linear, deterministisch) zu einem **linearen und stochasitschen** Modell.

Eigenschaften des Quanitsierungsrauschens

Gleichverteiltes Rauschen:

$$\boxed{p(e) = \begin{cases} 1/Q, & -Q/2 \leq e \leq Q/2 \\ 0, & \text{sonst} \end{cases}}$$

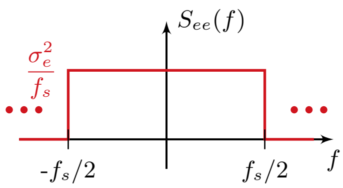


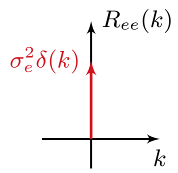
Mittelwert: $\mu_e = E[e] = 0$

Varianz: $\sigma_e^2 = E[e^2] = \frac{Q^2}{12}$

Weisses Rauschen (Leistungsdichtespektrum und Autokorrelation):

$$\boxed{S_{ee}(f) = \begin{cases} \frac{\sigma_e^2}{f_s}, & -\frac{f_s}{2} \leq f \leq \frac{f_s}{2} \\ 0, & \text{sonst} \end{cases}}$$





Unkorreliert mit dem Signal $x(n)$: $R_{ex} = E[e(n+k) \cdot x(n)] = 0$

Signal-Rausch-Abstand (SNR):

Rauschleistung:

$$P_N = E[e^2] = \frac{1}{Q} \int_{-Q/2}^{Q/2} e^2 \, de = \frac{Q^2}{12}$$

Signalleistung (voll ausgesteuert und gleichverteilt):

$$P_S = E[s^2] = \frac{1}{R} \int_{-R/2}^{R/2} s^2 \, ds = \frac{R^2}{12}$$

$$\text{SNR} = 10 \log_{10}\left(\frac{P_S}{P_N}\right) = 10 \log_{10}\left(\frac{R^2/12}{Q^2/12}\right) = 20 \log_{10}\left(\frac{R}{Q}\right) = 20 \log_{10}\left(2^B\right) = B \cdot 6\text{dB}$$

Pro Bit wird SNR um Faktor 4 besser.

$\boxed{\text{SNR} = B \cdot 6\text{dB}}$

2.2 Oversampling ⇔ Quantisierungsstufen

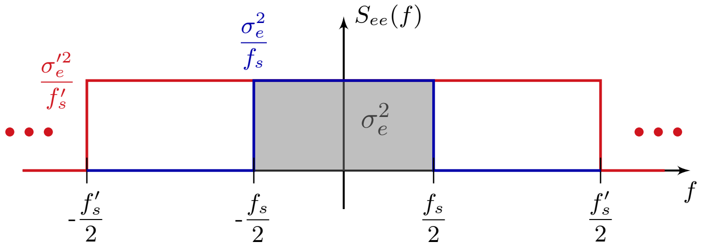
Die Idee von Oversampling ist, durch schnelleres Abstasten des Signals, weniger Bits für den Quanitsierer spendieren zu müssen aber dennoch die selbe Quantisierungsqualität beizubehalten.

$\boxed{\text{schneller Abtasten + weniger Quantisierungsintervalle = gleiche Quanitsierungsqualität}}$

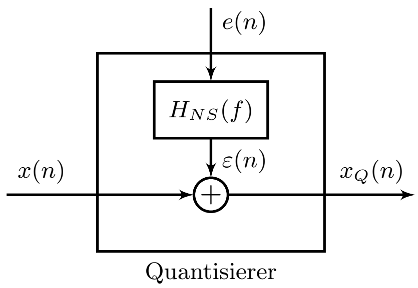
$\boxed{\text{schneller Abtasten + gleich viele Quantisierungsintervalle = bessere Quanitsierungsqualität}}$

2.2.1 Vergleich der zwei unterschiedlich schnell abgetasteten Systeme

	langsame Abtastfrequenz f_s viele Bits pro Sample B	schnelle Abtastfrequenz f'_s wenig Bits pro Sample B'
Oversamplingrate:	$\boxed{L = \frac{f'_s}{f_s}}$	
Quantisierungsintervall:	$Q = \frac{R}{2^B}$	$Q' = \frac{R}{2^{B'}}$
Rauschleistung:	$\sigma_e^2 = \frac{Q^2}{12}$	$\sigma_e'^2 = \frac{Q'^2}{12}$
Leistungsdichtespektrum:	$\boxed{\frac{\sigma_e^2}{f_s} = \frac{\sigma_e'^2}{f'_s} \quad \Rightarrow \quad \sigma_e^2 = \frac{\sigma_e'^2}{L}}$	
Bitgewinn: 1 Bit $\hat{=}$ 4 $\times$ schneller abtasten	$L = \frac{\sigma_e'^2}{\sigma_e^2} = \frac{Q'^2}{Q^2} = 2^{2(B-B')} = 2^{2\Delta B}$ $\boxed{\Delta B = 0.5 \cdot \log_2(L)}$	



2.3 Noise Shaping

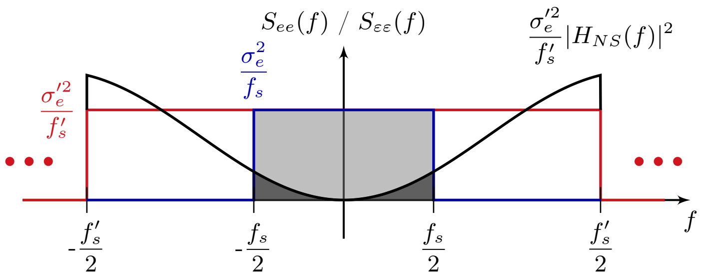


Die Idee von Noise Shaping ist, das Quantisierungsrauschen so umzuverteilen, dass im Frequenzband des langsam abgetasteten Systems möglichst wenig Rauschleistung vorhanden ist. Das Quantisierungsrauschen wird also im Prinzip zuerst gefiltert.

$$\boxed{\sigma_e^2 = \frac{\sigma_e'^2}{f'_s} \int_{-f_s/2}^{f_s/2} |H_{NS}(f)|^2 \, df}$$

$$\boxed{|H_{NS}(f)|^2 = \left| 2 \sin\left(\frac{\pi f}{f'_s}\right)\right|^{2p} \quad \text{für } -\frac{f'_s}{2} \leq f \leq \frac{f'_s}{2}}$$

$$\sigma_e^2 \overset{|f|<<f'_s/2}{\approx} \frac{\sigma_e'^2}{f'_s} \int_{-f_s/2}^{f_s/2} \left(\frac{2\pi f}{f'_s}\right)^{2p} df = \sigma_e'^2 \frac{\pi^{2p}}{2p+1} \frac{1}{L^{2p+1}} \quad \Rightarrow \quad \frac{\sigma_e^2}{\sigma_e'^2} = 2^{-2(B-B')} = 2^{-2\Delta B} = \frac{\pi^{2p}}{2p+1} \frac{1}{L^{2p+1}}$$

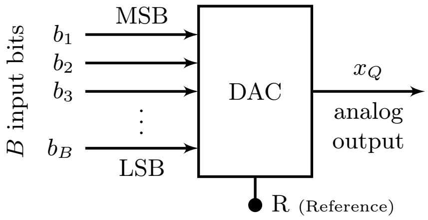


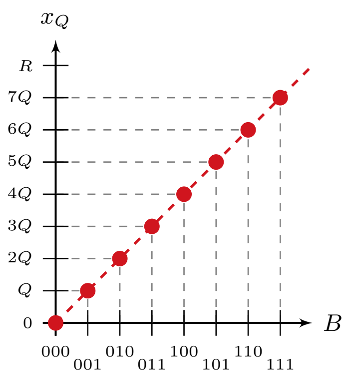
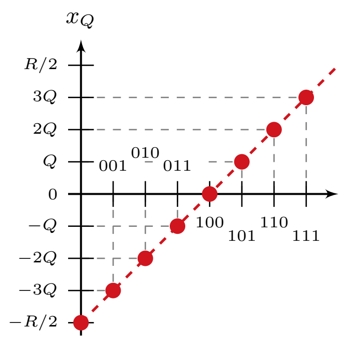
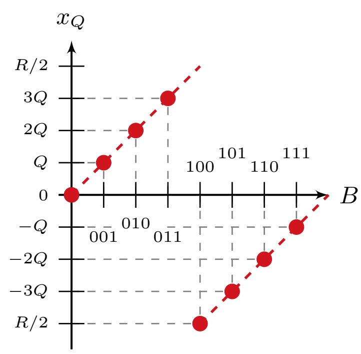
$$\boxed{\Delta B = (p + 0.5)\log_2(L) - 0.5 \log_2\left(\frac{\pi^{2p}}{2p+1}\right)}$$

2.4 Digital-Analog-Wandler

Es gibt verschiedene Arten um Bitmuster in einen analogen Wert zu wandeln.

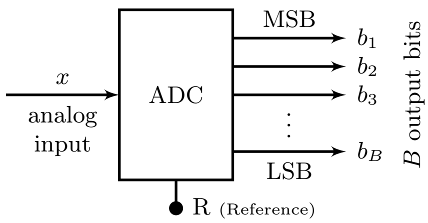
- Unipolar Natural Binary
- Bipolar Offset Binary
- Bipolar Two's Complement



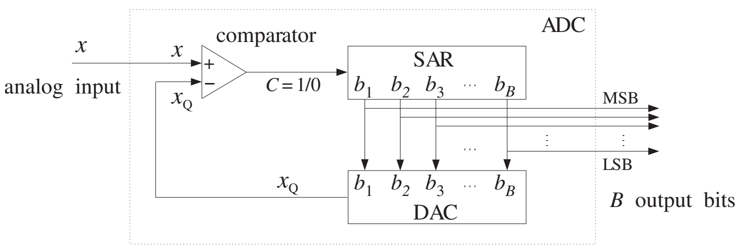
Unipolar Natural Binary	Bipolar Offset Binary	Bipolar Two's Complement
		
$\boxed{x_Q = R \sum_{n=1}^{B} b_n\, 2^{-n}}$	$\boxed{x_Q = R \left(\sum_{n=1}^{B} b_n\, 2^{-n} - \frac{1}{2} \right)}$	$\boxed{x_Q = R \left(\overline{b_1}\, 2^{-1} + \sum_{n=2}^{B} b_n\, 2^{-n} - \frac{1}{2} \right)}$
$\boxed{x_Q = Q \cdot m}$ $\boxed{m = \sum_{n=1}^{B} b_n\, 2^{B-n}}$	$\boxed{x_Q = Q \cdot m'}$ $\boxed{m' = m - 2^{B-1}}$	$\boxed{x_Q = Q \cdot m''}$ $\boxed{m'' = m + ((\text{-}1)^{b_1} - 1)\, 2^{B-1}}$
$m = 0, 1, 2, ..., 2^B - 1$	$m' = \text{-}2^{B-1}, ..., \text{-}1, 0, 1, ..., 2^{B-1} - 1$	$m'' = \text{-}2^{B-1}, ..., \text{-}1, 0, 1, ..., 2^{B-1} - 1$
$x_{Qmin} = 0$	$x_{Qmin} = -R/2$	$x_{Qmin} = -R/2$
$x_{Qmax} = R - Q$	$x_{Qmax} = R/2 - Q$	$x_{Qmax} = R/2 - Q$

2.5 Analog-Digital-Wandler

Ein Analog-Digital-Wandler quantisiert einen analogen Wert in ein digitales Bitmuster.



Successive Approximation:



1. Alle bits auf null setzen
2. b_1 auf 1 setzen
   $x_Q > x \quad \rightarrow \quad b_1 = 0$
   $x_Q < x \quad \rightarrow \quad b_1 = 1$
3. b_2 auf 1 setzen
   $x_Q > x \quad \rightarrow \quad b_2 = 0$
   $x_Q < x \quad \rightarrow \quad b_2 = 1$
4. mit allen Bits wiederholen
5. b_B auf 1 setzen
   $x_Q > x \quad \rightarrow \quad b_B = 0$
   $x_Q < x \quad \rightarrow \quad b_B = 1$

Weil der SAR-AD-Wandler immer abrundet, muss vor dem Quantisieren zum Eingangswert eine halber Quantisierungsschritt dazugezählt werden

$$\boxed{y = x + \frac{1}{2} Q}$$

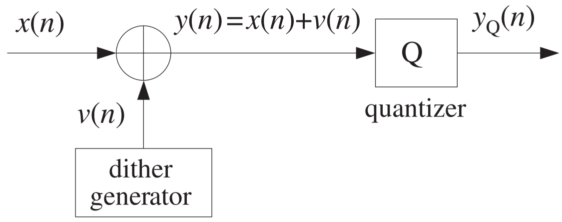
2.6 Dither

Dither ist ein kleines weisses Rauschen, dass vor dem Quantisieren zum Signal addiert wird.
Dies wird gemacht um:

- Quantisierungsverzerrungen (Muster) zu eliminieren
- Das Quantisierungsrauschen weisser scheinen lassen

Fehler:

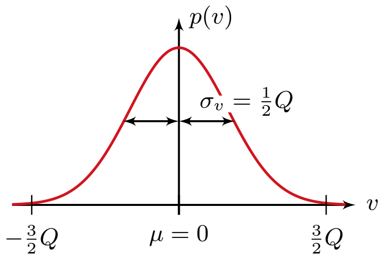
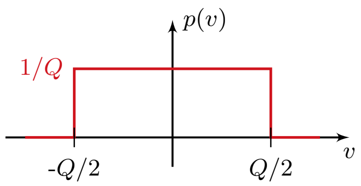
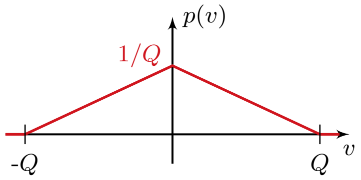
Da das Quantisierungsrauschen $e(n)$ und das Ditherrauschen $v(n)$ unkorreliert sind, wird der Fehler zwischen dem Eingangssignal und dem quanitsierten Signal zu:



$\boxed{\varepsilon(n) = y_Q(n) - x(n) = v(n) + e(n)}$

$$\boxed{\sigma_\varepsilon^2 = \sigma_e^2 + \sigma_v^2 = \frac{Q^2}{12} + \sigma_v^2}$$

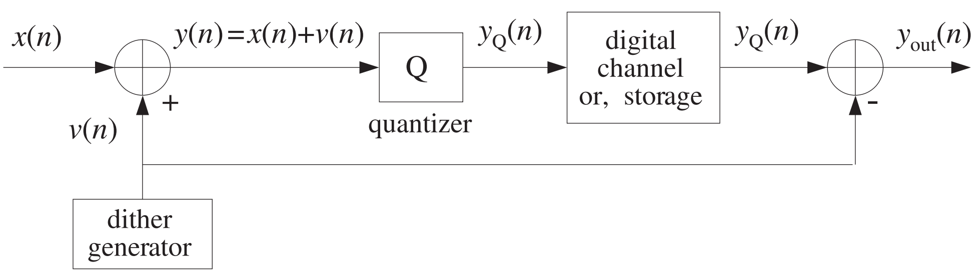
2.6.1 Formen von Ditherrauschen

Gausssches Rauschen	**Gleichverteiltes Rauschen**	**Dreieckverteiltes Rauschen**
		
$p(v) = \dfrac{\mathrm{e}^{-v^2/(2\sigma_v^2)}}{\sqrt{2\pi\sigma_v^2}}$	$p(v) = \begin{cases} \dfrac{1}{Q}, & -\frac{1}{2}Q \le v \le \frac{1}{2}Q \\ 0, & \text{sonst} \end{cases}$	$p(v) = \begin{cases} \dfrac{Q-\lvert v\rvert}{Q^2}, & -Q \le v \le Q \\ 0, & \text{sonst} \end{cases}$
$\sigma_v^2 = \frac{1}{4}Q^2$	$\sigma_v^2 = \frac{1}{12}Q^2$	$\sigma_v^2 = \frac{1}{6}Q^2$
$\boxed{\sigma_\varepsilon^2 = \frac{1}{12}Q^2 + \frac{1}{4}Q^2 = \frac{1}{3}Q^2}$	$\boxed{\sigma_\varepsilon^2 = \frac{1}{12}Q^2 + \frac{1}{12}Q^2 = \frac{1}{6}Q^2}$	$\boxed{\sigma_\varepsilon^2 = \frac{1}{12}Q^2 + \frac{1}{6}Q^2 = \frac{1}{4}Q^2}$
$4\times$ mehr Rauschen (6dB)	$2\times$ mehr Rauschen (3dB)	$3\times$ mehr Rauschen (4.8dB)

2.6.2 Subtractives Dither-Rauschen

Da das Dither-Rauschen selber erzeugt wurde kann es nach dem Quantisieren auch wieder abgezogen werden. Der Fehler reduziert sich dadurch wieder auf den Fehler des Quantisierungsrauschens.

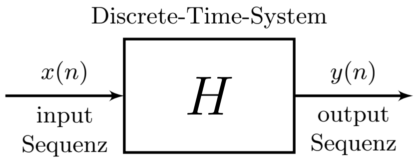
$\boxed{\varepsilon(n) = y_{out}(n) - x(n) = e(n)}$ $\boxed{\sigma_\varepsilon^2 = \frac{1}{12}Q^2}$



3 Discrete-Time Systems S.95

Ein Discrete-Time-System ist ein System, welches eine Eingangssequenz in eine Ausgangssequenz transformiert.
Die Transformation kann dabei auf zwei Arten erfolgen:

- Sample für Sample
- Blockweise



Sample für Sample

$$\{x_0, x_1, x_2, ..., x_n, ...\}$$
$$\downarrow H$$
$$\{y_0, y_1, y_2, ..., y_n, ...\}$$

Bsp:

$$y(n) = 2\,x(n) + 3\,x(n-1) + 4\,x(n-2)$$

Blockweise

$$\begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_n \\ \vdots \end{bmatrix} \xrightarrow{H} \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \\ \vdots \end{bmatrix}$$

Bsp: $\vec{y} = H \cdot \vec{x}$

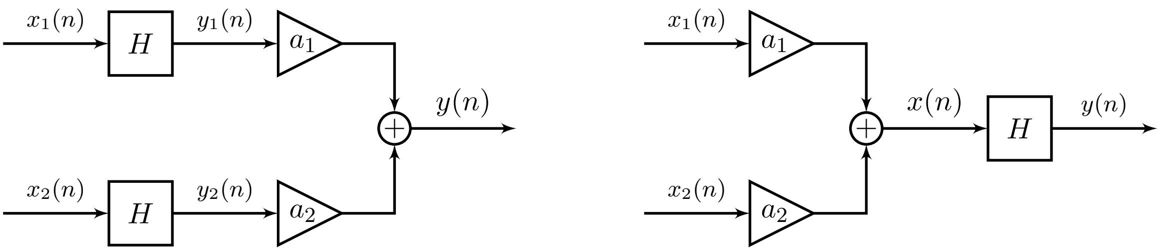
$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 \\ 3 & 2 & 0 \\ 4 & 3 & 2 \\ 0 & 4 & 3 \\ 0 & 0 & 4 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$$

3.1 Linearität und Zeitinvarianz

Linearität:

Für ein lineares System muss die Superpositionsbedingung erfüllt sein:

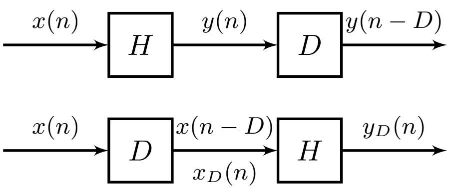
$$\boxed{y(n) = a_1\,H[x_1(n)] + a_2\,H[x_2(n)] = H[a_1\,x_1(n) + a_2\,x_2(n)]}$$

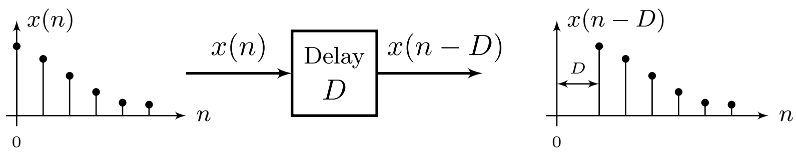


Zeitinvarianz:

Ein Zeitinvariantes System ändert seine Eigenschaften über die Zeit nicht.

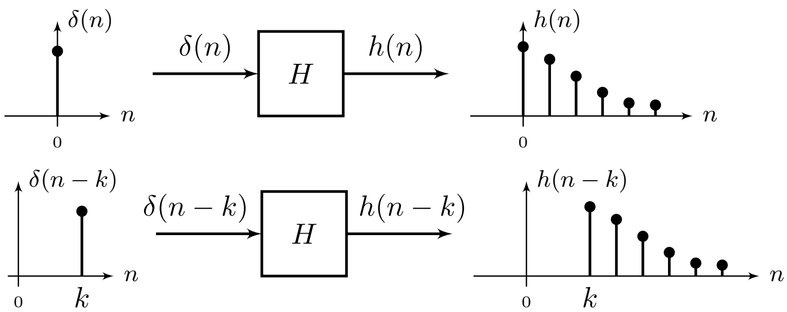
$$\boxed{y_D(n) = y(n-D)}$$





3.2 Impulsantwort

Ein lineares, zeitinvariantes System, ist durch seine Impulsantwort vollständig bestimmt. Wird am Eingang ein Impuls $\delta(n)$ angelet, so antwortet das System mit der Impulsantwort $h(n)$.



LTI-Form:

Eingangssequenz ist Folge von gewichteten und verzögerten Diracs. Ausgangssequenz ist Überlagerung der gewichteten und verzögerten Impulsantworten.

$$\boxed{y(n) = \sum_m x(m) \cdot h(n-m)}$$

Direkt Form:

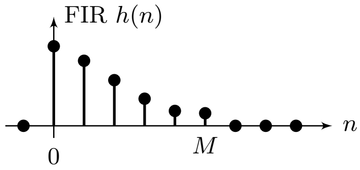
Faltung der Eingangssequenz mit der Impulsantwort

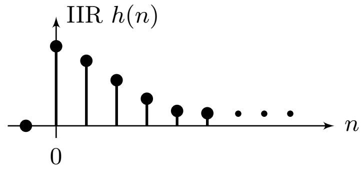
$$\boxed{y(n) = \sum_m h(m) \cdot x(n-m)}$$

3.3 FIR und IIR Filter

Diskrete LTI-Systeme werden über ihre Impulsantwort klassifiziert

- **FIR Filter:** Finite Impulse Response
- **IIR Filter:** Infinite Impulse Response





FIR Filter:

Ein FIR Filter hat eine endliche Anzahl Koeffizienten, die nicht Null sind.

Filterkoeffizienten	$\boxed{\{h_0, h_1, h_2, ..., h_M, 0, 0, 0, ...\}}$
Ordung:	$\boxed{M}$
Filterlänge:	$\boxed{L_h = M + 1}$

Für FIR Filter vereinfacht sich die Direkte Faltungsform zu:

$$y(n) = h_0\, x(n) + h_1\, x(n-1) + h_2\, x(n-2) + ... + h_M\, x(n-M)$$

$$\boxed{y(n) = \sum_{m=0}^{M} h(m) \cdot x(n-m)}$$

IIR Filter:

Ein IIR Filter hat eine unendliche Anzahl Koeffizienten, die nicht Null sind. Da solche Filter nicht realisiert werden können, wird er Fokus auf eine wichtige Unterklasse der IIR Filter gelegt:
IIR Filter die den Ausgang zurückführen (Feedback-Loop)

Die Impulsantwort solcher IIR Filter kann in folgender Form geschrieben werden:

$$\boxed{h(n) = \underbrace{\sum_{i=1}^{M} a_i\, h(n-i)}_{\text{IIR-Teil}} + \underbrace{\sum_{j=0}^{L} b_j\, \delta(n-j)}_{\text{FIR-Teil}}}$$

und die Ein-Ausgangsgleichung folgendermassen:

$$y(n) = \underbrace{a_1\, y(n-1) + a_2\, y(n-2) + ... + a_M\, y(n-M)}_{\text{IIR-Teil}} + \underbrace{b_0\, x(n) + b_1\, x(n-1) + b_2\, x(n-2) + ... + b_L\, x(n-L)}_{\text{FIR-Teil}}$$

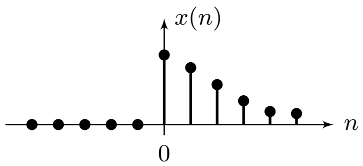
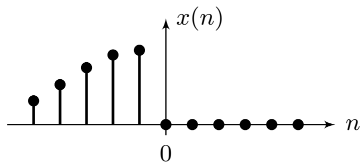
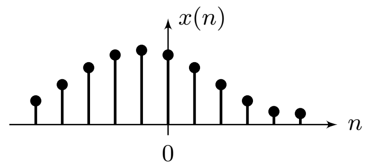
$$\boxed{y(n) = \underbrace{\sum_{i=1}^{M} a_i\, y(n-i)}_{\text{IIR-Teil}} + \underbrace{\sum_{j=0}^{L} b_j\, x(n-j)}_{\text{FIR-Teil}}}$$

3.4 Kausalität und Stabilität

Kausalität

Es gibt drei Arten von Signalen und demzufolge auch drei Arten von LTI-Systemen.

- Kausale Signale / LTI-Systeme
- Antikausale Signale / LTI-Systeme
- Gemischte Signale / LTI-Systeme

Kausal	**Antikausal**	**Gemischt**
		
ungleich Null für $n >= 0$	ungleich Null für $n <= -1$	ungleich Null alle n

Stabilität:

Ein LTI-System ist stabil, wenn ein begrenzter Eingang nur einen begrenzten Ausgang generiert.

$$\boxed{\sum_{n=-\infty}^{\infty} |h(n)| < \infty}$$

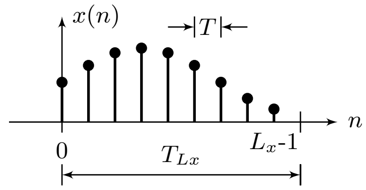
Stabilität und Kausalität:

- Stabilität und Kausalität sind unabhänging voneinander
- Stabilität ist zwingend Kausalität nicht
- Um ein stabiles, antikausales oder gemischtes LTI-System zu realisieren, wird der Ausgang und die Anzahl Samples verzögert, die in der Zukunft liegen würden.
  Geht die Impulsantwort bis ins unendliche in die Zukunft, so wird diese einfach abgeschnitten, wenn die Koeffizienten genug klein geworden sind → Approximation

4 FIR Filter und Faltung S.121

4.1 Blockverarbeitungsmethoden

Oft werden analoge Signale eine bestimmte Zeit lang abgetastet und in einem Vektor gespeichert. Dieser wird dann als Block verarbeitet.



Messdauer: $\boxed{T_{Lx} = L_x \cdot T}$

Anzahl Samples: $\boxed{L_x = T_{Lx} \cdot f_s}$

4.1.1 Blocklängen und Grenzen der Faltungssumme

Gegeben:

- Eingangsvektor x der Länge L_x
- Kausales FIR-Filter h der Ordung M und Länge L_h
  $\boxed{L_h = M + 1}$
- Ausgangsvektor der Länge L_y
  $\boxed{L_y = L_x + L_h - 1 = L_x + M}$

$$h = \underbrace{\boxed{h_0, h_1, h_2, ..., h_M}}_{M+1}$$

$$x = \underbrace{\boxed{x_0, x_1, x_2, x_3, ..., x_{L_x-1}}}_{L_x}$$

$$y = h * x = \underbrace{\underbrace{\boxed{y_0, y_1, y_2, ..., y_{L_x-1}}}_{L_x}\underbrace{\boxed{y_{L_x}, ..., y_{L_x+M}}}_{M}}_{L_y}$$

Grenzen der Faltungssumme

Für jedes $y(n)$ müssen die Grenzen der Summe festgelegt werden. Dabei müssen folgende zwei Bedinungen eingehalten werden:

$$(0 \leq m \leq L_x - 1 \quad \cap \quad 0 \leq n - m \leq M) \qquad \cup \qquad (0 \leq m \leq M \quad \cap \quad 0 \leq n - m \leq L_x - 1)$$

Berechnung des Ausgangsvektors $y(n)$:

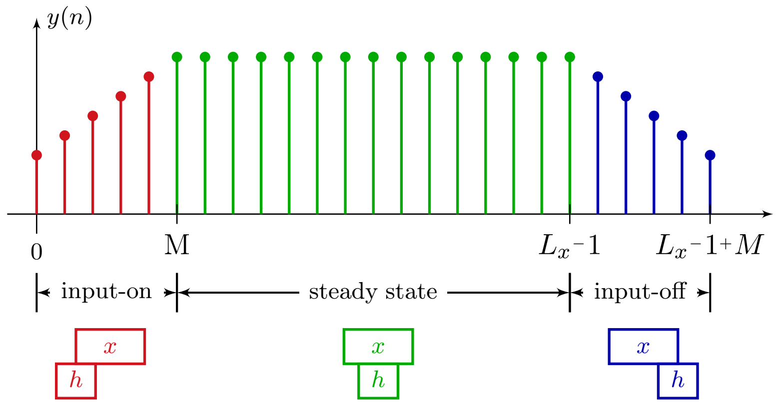
LTI-Form: $\boxed{y(n) = \sum_{m=max\{0,n-M\}}^{min\{n,L_x-1\}} x(m) \cdot h(n-m)}$ für $n = 0, 1, 2, ..., L_x + M - 1$

Direct-Form: $\boxed{y(n) = \sum_{m=max\{0,n-L_x+1\}}^{min\{n,M\}} h(m) \cdot x(n-m)}$ für $n = 0, 1, 2, ..., L_x + M - 1$

4.1.2 Transienten und Steady-State

Bei der Faltung zwischen dem Eingang $x(n)$ und der Impulsantwort $h(n)$ gibt es drei Phasen:

- Input-On Transiente
- Steady State
- Input-Off Transiente



$$\boxed{y(n) = \begin{cases} \sum_{m=0}^{n} h(m) \cdot x(n-m) & 0 \leq n < M & \text{Input-On Transiente} \\ \sum_{m=0}^{M} h(m) \cdot x(n-m) & M \leq n \leq L_x - 1 & \text{Steady State} \\ \sum_{m=n-L_x+1}^{M} h(m) \cdot x(n-m) & L_x - 1 < n \leq L_x - 1 + M & \text{Input-Off Transiente} \end{cases}}$$

4.1.3 Faltungstabelle (Convolution table)

Die Faltung kann in Form einer Tabelle geschrieben werden.

Für den n-ten Wert der Ausgangssequenz $y(n)$ müssen die Werte der entsprechenden Diagonale zusammengezählt werden.

$$\boxed{y(n) = \sum_{\substack{i,j \\ i+j=n}} h(i) \cdot x(j)}$$

Bsp: $y(5) = \sum_{\substack{i,j \\ i+j=5}} h(i) \cdot x(j) = h(3)\,x(2) + h(2)\,x(3) + h(1)\,x(4)$

$\longrightarrow j$, $\downarrow i$

	x_0	x_1	x_2	x_3	x_4
h_0	$h_0 x_0$	$h_0 x_1$	$h_0 x_2$	$h_0 x_3$	$h_0 x_4$
h_1	y_0 $h_1 x_0$	$h_1 x_1$	$h_1 x_2$	$h_1 x_3$	$h_1 x_4$
h_2	y_1 $h_2 x_0$	$h_2 x_1$	$h_2 x_2$	$h_2 x_3$	$h_2 x_4$
h_3	y_2 $h_3 x_0$	$h_3 x_1$	$h_3 x_2$	$h_3 x_3$	$h_3 x_4$
	y_3	y_4	y_5	y_6	y_7

4.1.4 LTI-Form Faltung

Der Eingangsvektor $x(n)$ kann in gewichtete und verzögerte Diracs aufgesplittet werden. Diese werden einzeln durch das LTI-System geschickt und deren gewichteten und verzögerten Impulsantworten summiert.

$$\begin{aligned} x &= x_0\,[1,\,0,\,0,\,0,\,0] \\ &+ x_1\,[0,\,1,\,0,\,0,\,0] \\ &+ x_2\,[0,\,0,\,1,\,0,\,0] \\ &+ x_3\,[0,\,0,\,0,\,1,\,0] \\ &+ x_4\,[0,\,0,\,0,\,0,\,1] \end{aligned} \quad \xrightarrow{H} \quad \begin{aligned} y &= x_0\,[h_0,\,h_1,\,h_2,\,h_3,\,0,\,0,\,0,\,0] \\ &+ x_1\,[0,\,h_0,\,h_1,\,h_2,\,h_3,\,0,\,0,\,0] \\ &+ x_2\,[0,\,0,\,h_0,\,h_1,\,h_2,\,h_3,\,0,\,0] \\ &+ x_3\,[0,\,0,\,0,\,h_0,\,h_1,\,h_2,\,h_3,\,0] \\ &+ x_4\,[0,\,0,\,0,\,0,\,h_0,\,h_1,\,h_2,\,h_3] \end{aligned}$$

	h_0	h_1	h_2	h_3	0	0	0	0
x_0	x_0h_0	x_0h_1	x_0h_2	x_0h_3	0	0	0	0
x_1	0	x_1h_0	x_1h_1	x_1h_2	x_1h_3	0	0	0
x_2	0	0	x_2h_0	x_2h_1	x_2h_2	x_2h_3	0	0
x_3	0	0	0	x_3h_0	x_3h_1	x_3h_2	x_3h_3	0
x_4	0	0	0	0	x_4h_0	x_4h_1	x_4h_2	x_4h_3
	y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7

$$\boxed{y(n) = \sum_m x(m) \cdot h(n-m)}$$

4.1.5 Matrix Form

Die Faltung zwischen Eingangsvektor $x(n)$ und Impulsantwort $h(n)$ kann auch als Matrixmultiplikation geschriben werden. Dabei gibt es zwei Formen:

- Impulsantwortmatrix $\boxed{\vec{y} = H \cdot \vec{x}}$
- Eingangssequenzmatirx $\boxed{\vec{y} = X \cdot \vec{h}}$

Impulsantwortmatrix H $((L_x + M) \times L_x)$

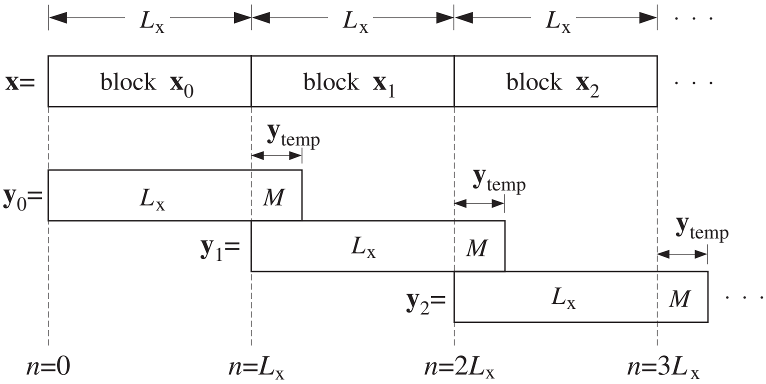
$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \underbrace{\begin{bmatrix} h_0 & 0 & 0 & 0 & 0 \\ h_1 & h_0 & 0 & 0 & 0 \\ h_2 & h_1 & h_0 & 0 & 0 \\ h_3 & h_2 & h_1 & h_0 & 0 \\ 0 & h_3 & h_2 & h_1 & h_0 \\ 0 & 0 & h_3 & h_2 & h_1 \\ 0 & 0 & 0 & h_3 & h_2 \\ 0 & 0 & 0 & 0 & h_3 \end{bmatrix}}_{H} \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Eingangssequenzmatirx X $((L_x + M) \times (M + 1))$

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \underbrace{\begin{bmatrix} x_0 & 0 & 0 & 0 \\ x_1 & x_0 & 0 & 0 \\ x_2 & x_1 & x_0 & 0 \\ x_3 & x_2 & x_1 & x_0 \\ x_4 & x_3 & x_2 & x_1 \\ 0 & x_4 & x_3 & x_2 \\ 0 & 0 & x_4 & x_3 \\ 0 & 0 & 0 & x_4 \end{bmatrix}}_{X} \cdot \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{bmatrix}$$

4.1.6 Overlap-Add Blockfaltung

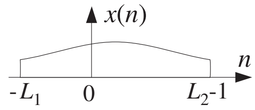
Wenn die Eingangssequenz sehr lange wird muss diese in Blöcke der Länge L_x aufgeteilt werden. Jeder dieser Blöcke wird einzeln verarbeitet. Da durch die Faltung aber Ausgangblöcke entstehen, die länger sind als die Eingangsblöcke, müssen diese auf die Länge der Eingangsblöcke L_x zugeschnitten werden. der Rest des Ausgangsblockes y_{temp} zum nächsten Blockes addiert.

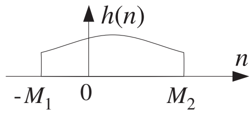


4.1.7 Generelle Faltung

Für akausale Eingangssequenzen und akausale Impulsantworten gilt folgende allgemeine Faltungsformel:

Direct-Form: $$\boxed{y(n) = \sum_{m=max\{-M_1, n-L_2+1\}}^{min\{n+L_1, M_2\}} h(m) \cdot x(n-m)}$$





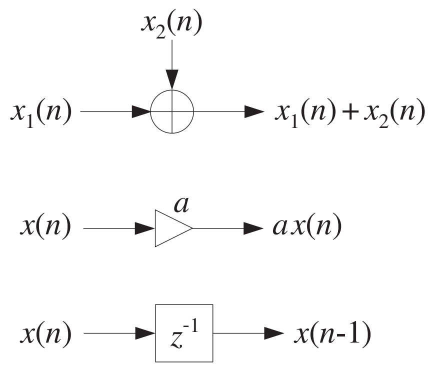
4.1.8 DC-Gain

Der DC-Gain eines stabilen Filters ist der Wert, auf den der Ausgang konvergiert, wenn am Eingang der Einheitsschritt angelegt wird $x(n) = u(n)$

DC-Gain $$\boxed{y_{DC} = \sum_m h(m)}$$

4.2 Sample für Sample Verarbeitung

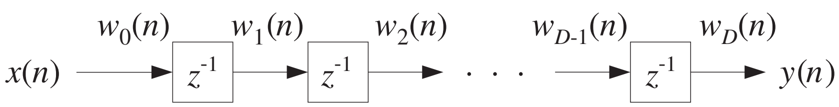
- Die Sample für Sample Verarbeitung wird angewendet wenn zwischen Ein- und Ausgang nur eine kurze Verzögerungszeit liegen darf (Echtzeitsysteme).
- Diese Verarbeitung kann in Signalflussdiagrammen dargestellt werden. Dazu werden die drei Elemente Addierer, Verstärker und Verzögerer verwendet.



4.2.1 Verzögerung (Pure Delays)

Um die verzögerte Werte speichern zu können werden **Zustände** definiert.
Ein Zustand $w(n)$ ist ein Register, welches den vorhergehenden Wert (Zustand) speichert.

Achtung:
Die Reihenfolge der Register-Updates ist sehr wichtig!



I/O Gleichungen

$w_0(n) = x(n)$
$y(n) = w_D(n)$

for $i = D, D-1, .., 1$ do:
$w_i(n+1) = w_{i-1}(n)$

Algorithums

$w[0] = x;$
$y = w[D];$

for $i = D, D-1, .., 1$ do:
$w[i] = w[i-1];$

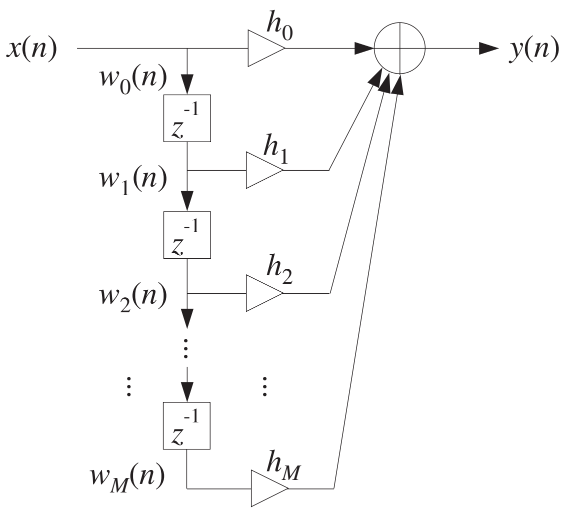
4.2.2 FIR filtern mit Direktform

Die Direktform I/O Faltungsformel für ein FIR-Filter der Ordung M lautet:

$$\begin{aligned} y(n) &= h_0 x(n) + h_1 x(n-1) + ... + h_M x(n-M) \\ &= h_0 w_0(n) + h_1 w_1(n) + ... + h_M w_M(n) \end{aligned}$$

Diese Form kann direkt in ein Signalflussdiagramm ungesetzt werden. Dabei gelten folgende Beziehungen:

$$\begin{aligned} & w_0(n) = x(n) \\ & w_1(n) = x(n-1) = w_0(n-1) \\ & w_2(n) = x(n-2) = w_1(n-1) \\ & \vdots \\ & w_M(n) = x(n-M) = w_{M-1}(n-1) \end{aligned}$$



- Durch die Verwendung von internen Zuständen kann der Ausgang aus dem aktuellen Eingangswert und den Zuständen berechnet werden.
- Die input-on und input-off Transienten können anhand des Signalflussdiagrammes leicht erklärt werden. Sie entsprechen jeweils der Zeit (Anzahl Samples), bis die Verzögerungslinie (Zustände) gefüllt bzw. geleert sind.

I/O Gleichungen

$w_0(n) = x(n)$
$y(n) = h_0 w_0(n) + h_1 w_1(n) + ... + h_M w_M(n)$

for $i = M, M-1, .., 1$ do:
$w_i(n+1) = w_{i-1}(n)$

Algorithums

$w[0] = x;$
$y = h[0]\, w[0] + h[1]\, w[1] + ... + h[M]\, w[M];$

for $i = M, M-1, .., 1$ do:
$w[i] = w[i-1];$

5 z-Transformation S.183

Definition:

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) \cdot z^{-n}$$

Linearität: Die z-Transformation ist linear.

$$a_1\, x_1(n) + a_2\, x_2(n) \quad \xrightarrow{Z} \quad a_1\, X_1(z) + a_2\, X_2(z)$$

Verzögerung: Verzögerung in der Zeit entsprich einer Multipikation mit z^{-D} in der z-Transformation.

$$x(n) \xrightarrow{Z} X(z) \quad \Rightarrow \quad x(n-D) \xrightarrow{Z} X(z) \cdot z^{-D}$$

Faltung: Die Faltung in der Zeit entsprich der Multipikation in der z-Transformation

$$y(n) = h(n) * x(n) \quad \xrightarrow{Z} \quad Y(z) = H(z) \cdot X(z)$$

5.1 Konvergenzbereich (ROC)

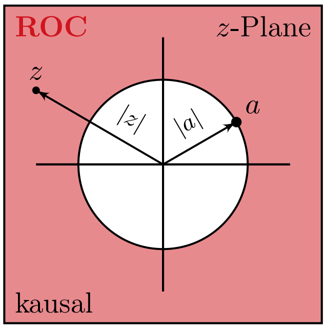
Zum Konvergenzbereich (Region of Convergence) gehören alle komplexen z-Werte bei denen die z-Transformation einen endlichen Wert hat.

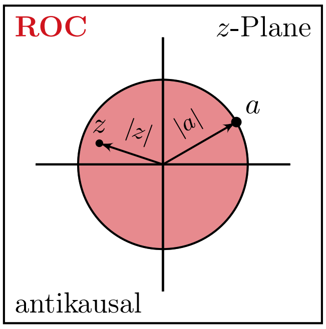
Region of Convergence: $$\text{ROC} = \left\{ z \in \mathbb{C} \;\middle|\; X(z) = \sum_{n=-\infty}^{\infty} x(n) \cdot z^{-n} \neq \pm\infty \right\}$$

Zu einer z-Transformation muss **immer die ROC angegeben** werden. Ist keine ROC angegeben wird angenommen, dass das Zeitsignal kausal war. Hat eine z-Transformation keine ROC, so existiert diese nicht!

$$a^n\, u(n) \xrightarrow{Z} \frac{1}{1 - a\, z^{-1}} \qquad \text{ROC} = \{ z \in \mathbb{C} \mid |z| > |a| \}$$ → kausales Signal

$$-a^n\, u(-n-1) \xrightarrow{Z} \frac{1}{1 - a\, z^{-1}} \qquad \text{ROC} = \{ z \in \mathbb{C} \mid |z| < |a| \}$$ → antikausales Signal

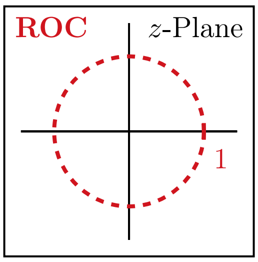




5.2 Kausalität und Stabilität

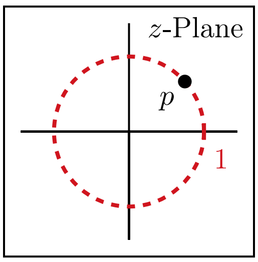
Stabilitätsbedingung

Der **Einheitskreis** der z-Ebene muss **innerhalb** der ROC liegen!



Grenzstabil

Der Einheitskreis bildet die Grenze der der ROC → Pol liegt genau auf dem Einheitskreis!



5.2.1 Kausale Signale

Generelles kausales Signal:

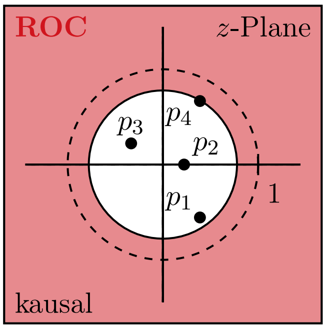
$$x(n) = A_1\, p_1^n\, u(n) + A_2\, p_2^n\, u(n) + \ldots$$
$$\downarrow Z$$
$$X(z) = \frac{A_1}{1 - p_1\, z^{-1}} + \frac{A_2}{1 - p_2\, z^{-1}} + \ldots$$

ROC:

$$\text{ROC} = \left\{ z \in \mathbb{C} \;\middle|\; |z| > \max_i |p_i| \right\}$$

Stabilitätsbedingung:

$$\max_i |p_i| < 1$$



5.2.2 Antikausale Signale

Generelles antikausales Signal:

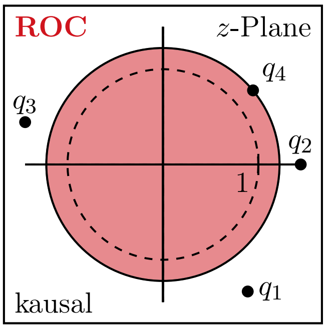
$$x(n) = \text{-}B_1\, q_1^n\, u(\text{-}n\text{-}1) - B_2\, q_2^n\, u(\text{-}n\text{-}1) - \ldots$$
$$\downarrow Z$$
$$X(z) = \frac{B_1}{1 - q_1\, z^{-1}} + \frac{B_2}{1 - q_2\, z^{-1}} + \ldots$$

ROC:

$$\text{ROC} = \left\{ z \in \mathbb{C} \;\middle|\; |z| < \min_i |q_i| \right\}$$

Stabilitätsbedingung:

$$\min_i |q_i| > 1$$



5.2.3 Gemischte Signale

Generelles gemischtes Signal:

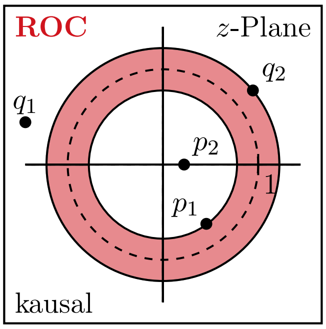
$$x(n) = A_1\, p_1^n\, u(n) + A_2\, p_2^n\, u(n) + \ldots$$
$$\text{-}B_1\, q_1^n\, u(\text{-}n\text{-}1) - B_2\, q_2^n\, u(\text{-}n\text{-}1) - \ldots$$
$$\downarrow Z$$
$$X(z) = \frac{A_1}{1 - p_1\, z^{-1}} + \frac{A_2}{1 - p_2\, z^{-1}} + \ldots$$
$$+ \frac{B_1}{1 - q_1\, z^{-1}} + \frac{B_2}{1 - q_2\, z^{-1}} + \ldots$$

ROC:

$$\text{ROC} = \left\{ z \in \mathbb{C} \;\middle|\; |z| > \max_i |p_i| \quad \cap \quad |z| < \min_i |q_i| \right\}$$

Stabilitätsbedingung:

$$\max_i |p_i| < 1 \quad \cap \quad \min_i |q_i| > 1$$



5.3 Inverse z-Transformation

Um eine z-Transformation, die als **gebrochen rationale Funktion** dargestellt werden kann, in die Zeit zurückzutransformieren sind folgende Schritte notwendig:

- Die z-Transformation in Partialbrüche der Form $\dfrac{1}{1 - a\, z^{-1}}$ zerlegen
- Jeden Partialbruch mit Hilfe folgender Beziehung und dem Wissen, ob der Pol kausal oder anitkausal ist, zurückzutransformieren.

kausales Signal	$\rightarrow$	$a^n\, u(n)$	$\xrightarrow{Z}$	$\dfrac{1}{1 - a\, z^{-1}}$	$\text{ROC} = \{z \in \mathbb{C} \mid \|z\| > \|a\|\}$
antikausales Signal	$\rightarrow$	$-a^n\, u(-n-1)$	$\xrightarrow{Z}$	$\dfrac{1}{1 - a\, z^{-1}}$	$\text{ROC} = \{z \in \mathbb{C} \mid \|z\| < \|a\|\}$

- Bei Doppelpolen folgende Rücktransformationen verwenden.

kausales Signal	$\rightarrow$	$(n+1)\, a^n\, u(n)$	$\xrightarrow{Z}$	$\dfrac{1}{(1 - a\, z^{-1})^2}$	$\text{ROC} = \{z \in \mathbb{C} \mid \|z\| > \|a\|\}$
antikausales Signal	$\rightarrow$	$-(n+1)\, a^n\, u(-n-1)$	$\xrightarrow{Z}$	$\dfrac{1}{(1 - a\, z^{-1})^2}$	$\text{ROC} = \{z \in \mathbb{C} \mid \|z\| < \|a\|\}$

- Konstanten werden zu gewichteten Diracs zurücktransformiert. $A_0 \delta(n) \xrightarrow{Z} A_0$

5.3.1 Partialbruchzerlegung bei einfachen Polen

Bei der Partialbruchzerlegung einer gebrochen rationalen Funktion werden drei Fälle unterschieden:

- $M > N$ echt gebrochen
- $M = N$ unecht gebrochen
- $M < N$ unecht gebrochen

$$X(z) = \frac{N(z)}{D(z)} = \frac{b_0 + b_1\, z^{-1} + b_2\, z^{-2} + \cdots + b_N\, z^{-N}}{1 + a_1\, z^{-1} + a_2\, z^{-2} + \cdots + a_M\, z^{-M}}$$

<table>
<tr><td>$M > N$</td><td>

$X(z) = \frac{N(z)}{D(z)} = \frac{N(z)}{(1-p_1\, z^{-1})\,\ldots\,(1-p_M\, z^{-1})} = \frac{A_1}{(1-p_1\, z^{-1})} + \cdots + \frac{A_M}{(1-p_M\, z^{-1})}$

$A_i = \left[(1-p_i\, z^{-1})\cdot X(z)\right]_{z=p_i} = \left[\frac{N(z)}{\prod\limits_{j\neq i}(1-p_j\, z^{-1})}\right]_{z=p_i}$

</td></tr>
<tr><td>$M = N$</td><td>

$X(z) = \frac{N(z)}{D(z)} = \frac{N(z)}{(1-p_1\, z^{-1})\,\ldots\,(1-p_M\, z^{-1})} = A_0 + \frac{A_1}{(1-p_1\, z^{-1})} + \cdots + \frac{A_M}{(1-p_M\, z^{-1})}$

$A_0 = X(z)|_{z=0}$ A_i siehe Fall $M > N$

</td></tr>
<tr><td>$M < N$</td><td>

$X(z) = \frac{N(z)}{D(z)} \overset{\text{Polynom-}}{\underset{\text{division}}{=}} Q(z) + \frac{R(z)}{D(z)}$

$Q(z) = B_0 + B_1\, z^{-1} + ... + B_{N-M}\, z^{-(N-M)} \xrightarrow{Z^{-1}} B_0\, \delta(n) + \delta(n\text{-}1) + ... + B_{N-M}\, \delta(n\text{-}(N\text{-}M))$

$\frac{R(z)}{D(z)} = \frac{R(z)}{(1-p_1\, z^{-1})\,\ldots\,(1-p_M\, z^{-1})} = \frac{A_1}{(1-p_1\, z^{-1})} + \cdots + \frac{A_M}{(1-p_M\, z^{-1})}$

A_i siehe Fall $M > N$

</td></tr>
<tr><td></td><td>

$X(z) = \frac{N(z)}{D(z)} = N(z)\cdot W(z)$

Nur $W(z)$ wie im Fall $M > N$ zurücktransformierten $\Rightarrow$ $w(n)$

$W(z) = \frac{1}{D(z)} = \frac{1}{(1-p_1\, z^{-1})\,\ldots\,(1-p_M\, z^{-1})} = \frac{A_1}{(1-p_1\, z^{-1})} + \cdots + \frac{A_M}{(1-p_M\, z^{-1})}$

A_i siehe Fall $M > N$

$X(z) = N(z)\cdot W(z) = b_0\cdot W(z) + b_1\, z^{-1}\cdot W(z) + b_2\, z^{-2}\cdot W(z) + \cdots + b_N\, z^{-N}\cdot W(z)$

$x(n)$ ist die Summe der mit b_i gewichteten und um i verzögerten $w(n)$

</td></tr>
</table>

5.4 Frequenzspektrum

Das Frequenzspektrum (DTFT) eines diskreten Signals $x(n)$ und ist definiert als:

$$\boxed{X(\omega) = \sum_{n=-\infty}^{\infty} x(n)\, \mathrm{e}^{-j\omega n}}$$

und die inverse DTFT als:

$$\boxed{x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega)\, \mathrm{e}^{j\omega n}\, d\omega}$$

5.4.1 Vergleich zwischen analogem Signal $x(nT)$ und digitalen Signal $x(n)$

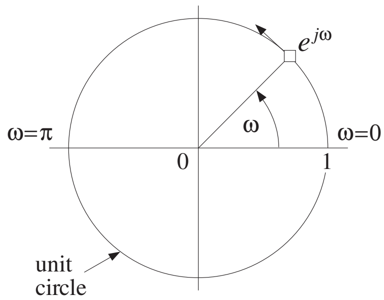
	Analoges Signal $x(nT)$	**Diskretes Signal $x(n)$**
DTFT	$\hat{X}(f) = \sum_{n=-\infty}^{\infty} x(nT) \cdot \mathrm{e}^{-j2\pi f T n}$	$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)\, \mathrm{e}^{-j\omega n}$
Periode	f_s	$2\,\pi$
inverse DTFT	$x(nT) = \frac{1}{f_s} \int_{-f_s/2}^{f_2/2} \hat{X}(f) \cdot \mathrm{e}^{j2\pi f T n}\, df$	$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega)\, \mathrm{e}^{j\omega n}\, d\omega$

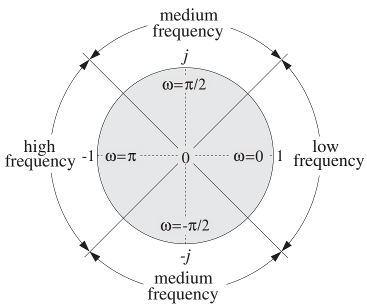
5.4.2 Geometrische Interpretation des Frequenzganges

Der Frequenzgang $X(\omega)$ entspricht der Übertragungsfunktion $X(z)$ auf dem Einheitskreis:

$$\boxed{z = \mathrm{e}^{j\omega}} \quad \Rightarrow \quad \boxed{X(\omega) = X(z)|_{z=\mathrm{e}^{j\omega}}}$$

Damit der Frequenzgang existiert, muss der Einheitskreis **innerhalb der Region of Convergence (ROC)** liegen.

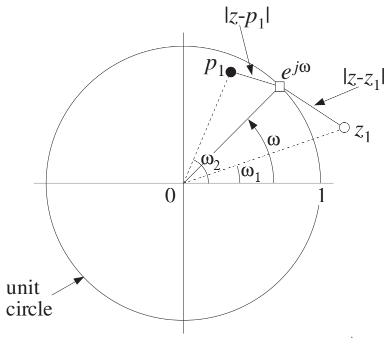


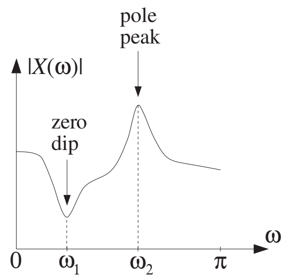


- Wenn ω von $-\pi$ bis π variert, wendert z auf dem Einheitskreis von -1 über 0 nach -1.
- Die tiefen Frequenzen sind bei $z \approx 1$ und die hohen bei $z \approx -1$. Bei $z \approx \pm j$ sind die mittleren Frequenzen.

Anhand der Pol- und Nullstellen in der z-Ebene, kann der Frequenzgang $|X(\omega)|$ skizziert werden.

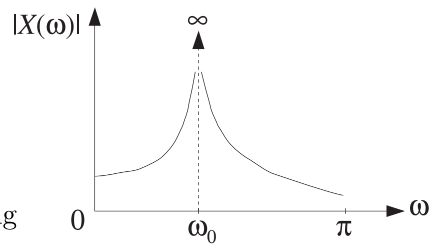
$$\boxed{|X(\omega_0)| = \frac{\prod_i \text{Abstand von } \mathrm{e}^{j\omega_0} \text{ zur Nullstelle } z_i}{\prod_j \text{Abstand von } \mathrm{e}^{j\omega_0} \text{ zum Pol } p_j}}$$





Daraus lassen sich folgende Grundsätze ableiten

- Je näher der Punkt auf dem Einheitskreis einer Nullstelle kommt, desto kleiner wird der Frequenzgang.
- Je näher der Punkt auf dem Einheitskreis einem Pol kommt, desto grösser wird der Frequenzgang.
- Sitzt der Pol genau auf dem Einheitskreis, so wird der Frequenzgang an dieser stelle unendlich gross.



5.4.3 Fourierpärchen

	$x(n)$	$X(\omega)$ (Nyquistintervall)
Komplexe Schwingung	$\mathrm{e}^{j\omega_0 n}$	$2\pi\, \delta(\omega - \omega_0)$
Cosinusschwingung	$\cos(\omega_0 n)$	$\pi\, \delta(\omega - \omega_0) + \pi\, \delta(\omega + \omega_0)$
Sinusschwingung	$\sin(\omega_0 n)$	$-j\pi\, \delta(\omega - \omega_0) + j\pi\, \delta(\omega + \omega_0)$

Satz des Parseval:

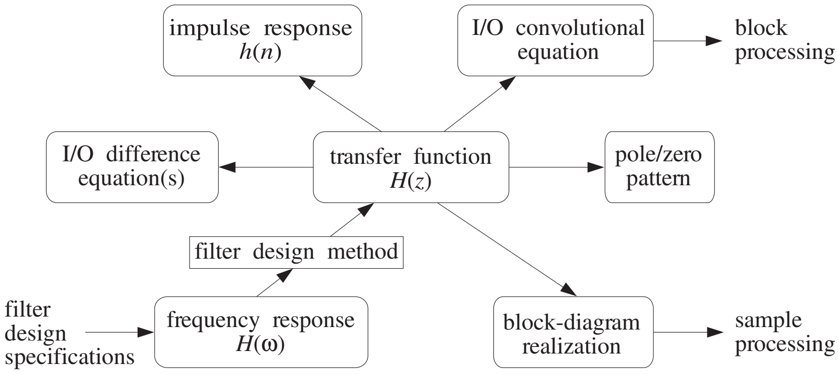
$$\boxed{E = \sum_{n=-\infty}^{\infty} |x(n)|^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |H(\omega)|^2\, d\omega}$$

6 Übertragungsfunktionen S.214

6.1 Äquivalente Beschreibungen für digitale Filter

Um digitale Filter zu beschreiben, gibt es diverse Möglichkeiten.

- Übertragungsfunktion $H(z)$
- Impulsantwort $h(n)$
- Frequenzgang $H(\omega)$
- I/O Differenzengleichung
- Pol/Nullstellen-Diagramm
- Realisation mit Blockdiagrammen für Sampleverarbeitung
- I/O Faltungsgleichungen für Blockverarbeitung



6.1.1 IIR und FIR Filter-Übertragungsfunktionen

- IIR Filter können im allgemeinen Fall als gebrochen rationale Funktionen dargestellt werden.

IIR $$\boxed{H(z)=\frac{N(z)}{D(z)}=\frac{b_0+b_1\,z^{-1}+b_2\,z^{-2}+\cdots+b_N\,z^{-N}}{1+a_1\,z^{-1}+a_2\,z^{-2}+\cdots+a_M\,z^{-M}}}$$

- Das FIR Filter ist ein Spezialfall des IIR Filters ($a_i=0$).

FIR $$\boxed{H(z)=N(z)=b_0+b_1\,z^{-1}+b_2\,z^{-2}+\cdots+b_N\,z^{-N}}$$

6.1.2 Impulsantwort $h(n)$

$$\boxed{H(z)\quad\xrightarrow{Z^{-1}}\quad h(n)}\qquad\qquad\boxed{h(n)\quad\xrightarrow{Z}\quad H(z)}$$

6.1.3 I/O Differenzengleichung

$H(z)$ umformen, dass es bruchfrei und wird und $H(z)$ alleine auf der linken Seite steht.

$$H(z)=\frac{N(z)}{D(z)}\qquad\Rightarrow\qquad H(z)\cdot\underbrace{\left[1+a_1\,z^{-1}+a_2\,z^{-2}+...+a_M\,z^{-M}\right]}_{D(z)}=\underbrace{\left[b_0+b_1\,z^{-1}+b_2\,z^{-2}+...+b_N\,z^{-N}\right]}_{N(z)}$$

$$\boxed{H(z)=\underbrace{\left[b_0+b_1\,z^{-1}+b_2\,z^{-2}+...+b_N\,z^{-N}\right]}_{N(z)}-H(z)\cdot\underbrace{\left[a_1\,z^{-1}+a_2\,z^{-2}+...+a_M\,z^{-M}\right]}_{D(z)-1}}$$

$$\Big\downarrow Z$$

$$\boxed{h(n)=[b_0\,\delta(n)+b_1\,\delta(n-1)+b_2\,\delta(n-2)+...+b_N\,\delta(n-N)]-[a_1\,h(n-1)+a_2\,h(n-2)+...+a_M\,h(n-M)]}$$

$$\delta(n)=x(n)\;\Big\downarrow\;h(n)=y(n)$$

$$\boxed{y(n)=[b_0\,x(n)+b_1\,x(n-1)+b_2\,x(n-2)+...+b_N\,x(n-N)]-[a_1\,y(n-1)+a_2\,y(n-2)+...+a_M\,y(n-M)]}$$

6.1.4 Frequenzgang $H(\omega)$

In der Übertragungsfunktion z durch $\mathrm{e}^{j\omega}$ ersetzen

$$\boxed{z = \mathrm{e}^{j\omega}} \quad \rightarrow \quad \boxed{H(\omega) = H(z)|_{z=\mathrm{e}^{j\omega}} = \frac{b_0 + b_1\,\mathrm{e}^{-j\omega} + b_2\,\mathrm{e}^{-j\omega 2} + \dots + b_N\,\mathrm{e}^{-j\omega N}}{1 + a_1\,\mathrm{e}^{-j\omega} + a_2\,\mathrm{e}^{-j\omega 2} + \dots + a_M\,\mathrm{e}^{-j\omega M}}}$$

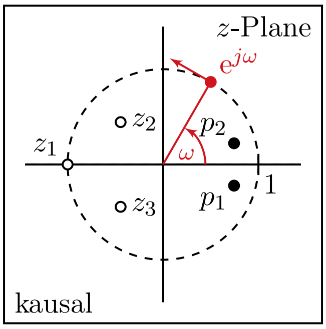
Für den Betrag der Übertragungsfunktion $|H(\omega)|$ können Terme der Form $|1 - a\,\mathrm{e}^{-j\omega}|$ noch vereinfacht werden.

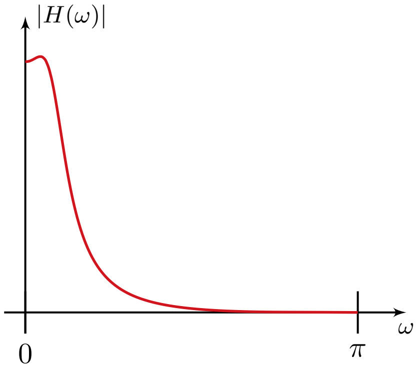
$$\boxed{|1 - a\,\mathrm{e}^{-j\omega}| = \sqrt{1 - 2a\cos(\omega) + a^2}}$$

$$|H(\omega)| = \frac{|1 - z_1\,\mathrm{e}^{-j\omega}|\,|1 - z_2\,\mathrm{e}^{-j\omega}|\,\dots\,|1 - z_N\,\mathrm{e}^{-j\omega}|}{|1 - p_1\,\mathrm{e}^{-j\omega}|\,|1 - p_2\,\mathrm{e}^{-j\omega}|\,\dots\,|1 - p_M\,\mathrm{e}^{-j\omega}|} = \frac{\sqrt{1 - 2z_1\cos(\omega) + z_1^2}\,\dots\,\sqrt{1 - 2z_N\cos(\omega) + z_N^2}}{\sqrt{1 - 2p_1\cos(\omega) + p_1^2}\,\dots\,\sqrt{1 - 2p_M\cos(\omega) + p_M^2}}$$

6.1.5 Pol/Nullstellen - Diagramm

- Die Pole (Nullstellen von $D(z)$) und Nullstellen (Nullstellen von $N(z)$) können in der z-Ebene eingezeichnet werden.
- Der Frequenzgang kann aus dem Pol/Nullstellen - Diagramm sehr einfach gezeichnet werden, indem ein man sich einen Punkt vorstellt, der von 0 bis π auf dem Einheitskreis wandert.
  - Punkt fährt nahe am Pol vorbei $\rightarrow$ Frequenzgang wird gross.
  - Punkt fährt nahe an Nullstelle vorbei $\rightarrow$ Frequenzgang wird klein.





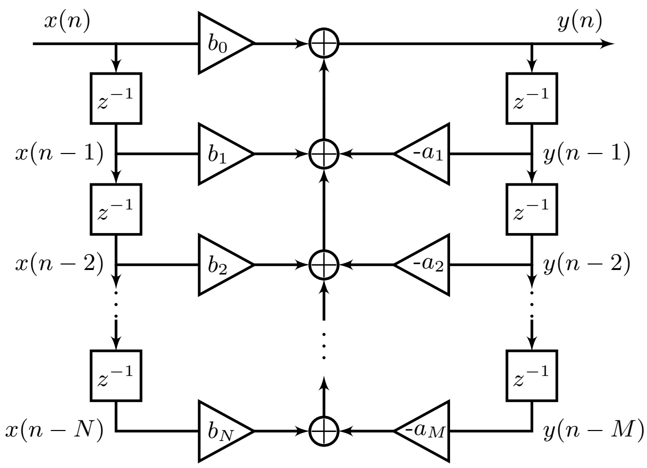
6.1.6 Realisation mit Blockdiagrammen

Es werden vier Arten von Blockdiagrammen unterschieden

- Direktform (Siehe Seite 217)
- Parallelform (Siehe Seite 219)
- Kanonische Form (Siehe Seite 221)
- Transpolierte Form (Siehe Seite 222)

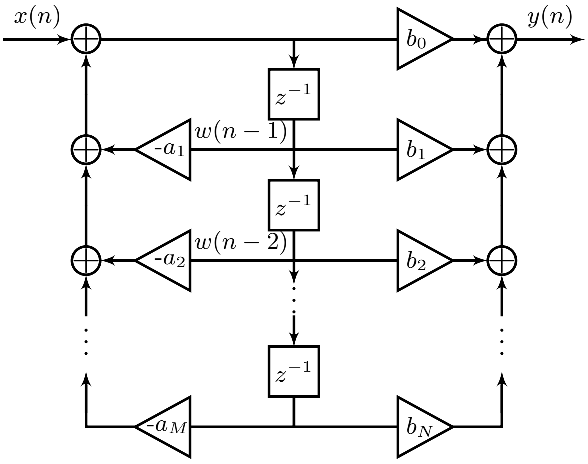
Direktform

Aus der I/O Differenzengleichung kann direkt das Blockdiagramm der Direktform aufgezeichnet werden. Der Nachteil an dieser Variante ist, dass sehr viele Zustände gespeichert werden müssen.



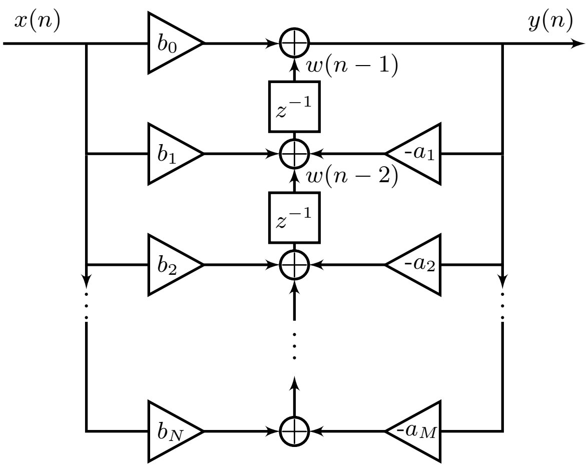
Kanonische Form

Die Kanonische Form kann aus der Direktform abgeleitet werden. Dazu muss die rechte und linke Seite des Blockdiagrammes getauscht werden. Der Vorteil an dieser Variante ist, dass viel weniger Zustände gespeichert werden müssen.



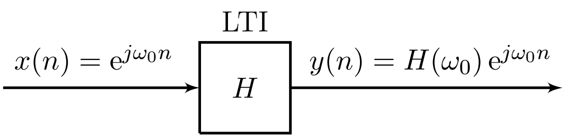
Transpolierte Form

Die Transpolierte Form kann aus der Kanonischen Form abgeleitet werden. Dazu muss das Blockdiagramm transponiert werden, d.h. Addierer mit Knoten und Knoten mit Addierer ersetzen, alle Flussrichtungen umkehren und Ein- und Ausgang vertauschen.



6.2 Systemantwort auf sinusförmiges Eingangssignal

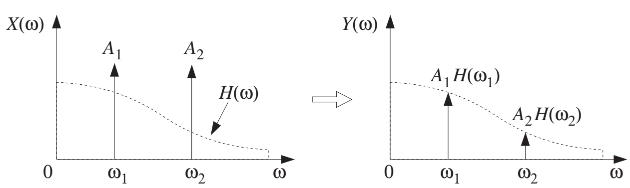
Wenn in ein LTI-System ein sinusförmiges Eingangssignal gegeben wird, so wird auch ein sinusförmiges Ausgangssignal derselben Frequenz herauskommen.



$$\boxed{\begin{array}{lcl} e^{j\omega_0 n} & \xrightarrow{H} & H(\omega_0)\, e^{j\omega_0 n} \\ e^{j\omega_0 n} & \xrightarrow{H} & |H(\omega_0)|\, e^{j\omega_0 n + j arg(H(\omega_0))} \end{array}}$$

Ein LTI System:

- erzeugt keine neuen Frequenzen
- dämpft/verstärkt das Signal mit $|H(\omega_0)|$
- verzöger das Signal um $arg(H(\omega_0))$
- erscheint am Ausgang mit derselben Frequenz



$$\boxed{\begin{array}{lcl} \cos(\omega_0 n) & \xrightarrow{H} & |H(\omega_0)| \cdot \cos(\omega_0 n + arg(H(\omega_0))) \\ \sin(\omega_0 n) & \xrightarrow{H} & |H(\omega_0)| \cdot \sin(\omega_0 n + arg(H(\omega_0))) \end{array}}$$

6.2.1 Phasen- und Gruppenverzögerung

Phasenverzögerung

Die Phasenverzögerung beschreibt, welche Frequenz um wie viele Samples verzögert wird. Die Anzahl verzögerter Samples muss nicht ganzzahlig sein.

$$\boxed{d(\omega) = -\frac{arg(H(\omega))}{\omega}} \quad \rightarrow \quad e^{j\omega n} \xrightarrow{H} |H(\omega)|\, e^{j\omega(n-d(\omega))}$$

Verzögert das Filter alle Frequenzen um die selbe Anzahl Samples, handelt es sich um ein linearphasiges Filter

$$\boxed{\text{linearphasiges Filter} \quad \Leftrightarrow \quad d(\omega) = D = \text{konstant}}$$

→ FIR Filter können sehr einfach als linearphasige Filter designed werden.

Gruppenverzögerung

$$\boxed{d_g(\omega) = -\frac{d\, arg(H(\omega))}{d\omega}}$$

Wenn in einem Frequenzintervall alle Frequenzen dieselbe Phasenverzögerung haben, ist die Gruppenverzögerung eine Konstante.

6.2.2 Einschwingvorgang

Wird eine Schwinung eingeschaltet, so braucht das Filter eine gewisse Zeit, bis es eingeschwungen ist (im steady state).

Einschaltsignal:

$$x(n) = e^{j\omega_0 n}\, u(n)$$

$$X(z) = \frac{1}{1 - e^{j\omega_0}\, z^{-1}} \qquad \text{ROC } |z| > 1$$

Filter:

$$H(z) = \frac{N(z)}{D(z)} = \frac{N(z)}{(1 - p_1\, z^{-1})\,(1 - p_1\, z^{-1})\, \ldots\, (1 - p_M\, z^{-1})}$$

Ausgangssignal:

$$Y(Z) = \frac{H(\omega_0)}{1 - e^{j\omega_0}\, z^{-1}} + \frac{B_1}{1 - p_1\, z^{-1}} + \frac{B_2}{1 - p_2\, z^{-1}} + \cdots + \frac{B_M}{1 - p_M\, z^{-1}}$$

$$\boxed{y(n) = H(\omega_0)\, e^{j\omega_0 n} + \underbrace{B_1\, p_1^n + B_2\, p_2^n + \cdots + B_M\, p_M^n}_{\text{Einschwingvorgang}}}$$

Übergang in Steady State

$$\boxed{\text{Alle Pole } |p_i| < 1 \quad \Rightarrow \quad \lim_{n\to\infty} y(n) = H(\omega_0)\, e^{j\omega_0 n}}$$

Einschwingzeit:

Der Pol am nächsten beim Einheitskreis dominiert die Zeitdauer des Einschwingvorganges

$$\boxed{\rho = \max_i \{|p_i|\}}$$

Der Steady State gilt als erreicht, wenn der Beitrag des langsamste Pols unter 1% gefallen ist.

$$\boxed{\rho^{n_{\text{eff}}} = \epsilon = 0.01} \quad \Rightarrow \quad \textbf{Zeitkonstante des Filters:} \quad \boxed{n_{\text{eff}} = \frac{\ln(\epsilon)}{\ln(\rho)} = \frac{\ln(1/\epsilon)}{\ln(1/\rho)}}$$

6.2.3 DC-Gain und AC-Gain

Das Einschwingverhalten des Einheitssprunges $u(n)$ resultiert im DC-Gain

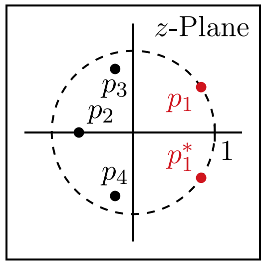
$$\boxed{\lim_{n\to\infty} y(n) = H(\omega_0)\, e^{j\omega_0 n}\big|_{\omega_0=0} = H(0)} \qquad \textbf{DC-Gain:} \quad \boxed{H(0) = H(z)|_{z=1} = \sum_{n=0}^{\infty} h(n)}$$

Das Einschwingverhalten des alternierenden Einheitssprunges $(-1)^n\, u(n)$ resultiert im AC-Gain

$$\boxed{\lim_{n\to\infty} y(n) = H(\omega_0)\, e^{j\omega_0 n}\big|_{\omega_0=\pi} = H(\pi)\,(-1)^n} \qquad \textbf{AC-Gain:} \quad \boxed{H(\pi) = H(z)|_{z=-1} = \sum_{n=0}^{\infty} (-1)^n\, h(n)}$$

6.2.4 Einschwingverhalten von Grenzstabilen Systemen

Wird an ein Grenzstabiles System, mit einem konjugiert komplexen Polpaar auf dem Einheitskreis, eine Einschaltschwinung angelegt, so resultiert folgende Ausgangsschwingung.



$$\boxed{y(n) = H(\omega_0)\, e^{j\omega_0 n} + \underbrace{B_1\, p_1^n + B_1^*\, p_1^{*n}}_{\text{grenzstabile Pole}} + \underbrace{B_2\, p_2^n + B_3\, p_3^n + \cdots + B_M\, p_1^n}_{\text{Einschwingvorgang}}}$$

Nach dem Übergang in den Steady State resultiert folgende Ausgangsschwingung:

$$\boxed{\lim_{n\to\infty} y(n) = H(\omega_0)\, e^{j\omega_0 n} + \underbrace{B_1\, p_1^n + B_1^*\, p_1^{*n}}_{\text{grenzstabile Pole}} = H(\omega_0)\, e^{j\omega_0 n} + \underbrace{B_1\, e^{j\theta_1 n} + B_1^*\, e^{-j\theta_1 n}}_{\text{Sinusschwingung}}}$$

- Das Ausgangssignal klingt nicht ab sondern schwingt.
- Die Frequenz der Eingangsschwingung darf nicht dieselbe sein, wie die der Pole auf dem Eingheitskreis!
- $e^{j\omega_0 n} = p_1 = e^{j\theta_1 n} \quad \Rightarrow \quad$ Doppelter Pol $\quad \Rightarrow \quad$ System ist in Resonanz, Ausgang wird immer grösser!

6.2.5 Einschwingverhalten von FIR Filtern

Bei FIR Filtern ist der Steady State nach M Samples erreicht (siehe Kapitel 4.1.2)

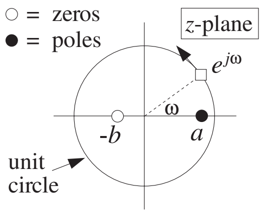
6.3 Pol/Nullstellen Filterdesign

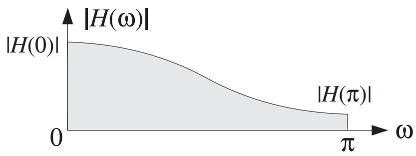
Durch das Platzieren von Polen und Nullstellen können intuitive Filter designed werden. Durch die Pol-und Nullstellen ist alles, bis auf einen Verstärkungsfaktor, definiert.

6.3.1 Filter erster Ordnung

Allgemeine Form: $\boxed{H(z) = G\,\dfrac{1 + b\, z^{-1}}{1 - a\, z^{-1}}}$

mit $|a| < 1$ und $|b| \leq 1$





Designparameter des Filters:

- Steilheit (Stärke) des Filters $\rightarrow$ Verhältnis von AC- zu DC-Gain

$$\boxed{\frac{H(\pi)}{H(0)} = \frac{(1-b)\,(1-a)}{(1+b)\,(1+a)} \quad \Rightarrow \quad \frac{H(\pi)}{H(0)} \begin{cases} < 1 & \text{Tiefpassfilter} \\ > 1 & \text{Hochpassfilter} \end{cases}}$$

- Maximale Zeitkonstante (Einschwingzeit) von N Samples

$$\boxed{n_{\text{eff}} = \frac{\ln(\epsilon)}{\ln(|a|)} \leq N}$$

- Verstärkungsfaktor G wird oft so gewählt, dass bei einem Tiefpassfilter $|H(0)| = 1$ ist und bei einem Hochpassfilter $|H(\pi)| = 1$ ist.

6.3.2 Resonator

Mittels eines konjugiert komplexen Polpaares nahe am Einheitskreis kann ein Resonator gebaut werden. Dabei gelten folgende grundlegenden Zusammenhänge:

$\boxed{\text{Schmalere Bandbreite} \quad \Leftrightarrow \quad \text{Pole näher am Einheitskreis} \quad \Leftrightarrow \quad \text{Längere Einschwingzeit des Filters}}$

Allgemeine Form:

$$\boxed{H(z) = \frac{G}{(1 - R\, e^{j\omega_0}\, z^{-1})\,(1 - R\, e^{-j\omega_0}\, z^{-1})} = \frac{G}{(1 + a_1\, z^{-1} + a_2\, z^{-2})})} \qquad \text{mit} \quad p = R\, e^{j\omega_0} \text{ und } |R| \leq 1$$

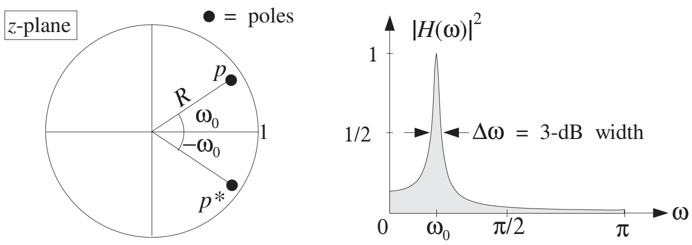
$$\boxed{a_1 = -2R\cos(\omega_0) \qquad a_2 = R^2} \qquad \boxed{|H(\omega_0)| = 1 \quad \Rightarrow \quad G = (1-R)\sqrt{1 - 2R\cos(2\omega_0) + R^2}}$$

Bandbreite als Designparameter:

Die 3dB Bandbreite ist die Breite auf der halben maximalen Höhe des quadrierten Frequenzganges.

$$\boxed{\Delta\omega = \omega_2 - \omega_1} \qquad \boxed{|H(\omega_{1,2})|^2 = \frac{|H(\omega_0)|^2}{2}}$$

$$\Rightarrow \quad \boxed{10\log\left(\frac{|H(\omega_{1,2})|^2}{|H(\omega_0)|^2}\right) = 10\log\left(\frac{1}{2}\right) = -3\text{dB}}$$

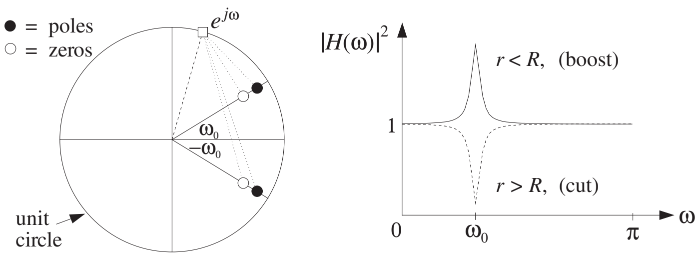


Ist der Pol nahe am Einheitskreis gilt: $\boxed{\Delta\omega \approx 2(1-R)}$

6.3.3 Notch- und Comb-Filter

Durch das "Hintereinanderlegen" von Polen ($p = R\,e^{j\omega_0}$) und Nullstellen ($z = r\,e^{j\omega_0}$) können gezielt bestimmte Frequenzen verstärkt bzw. unterdrückt werden.

- Pol näher am Einheitskreis ($R > r$)
  $\Rightarrow$ Verstärkung dieser Frequenz (Comb)
- Nullstelle näher am Einheitskreis ($R < r$)
  $\Rightarrow$ Unterdrückung dieser Frequenz (Notch)



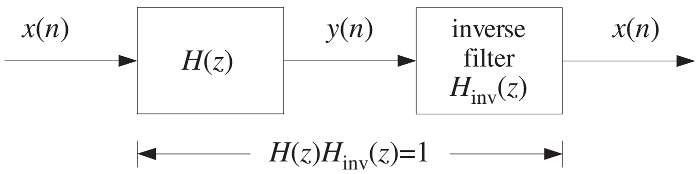
Allgemeine Form:

$$\boxed{H(z) = \frac{N(z)}{D(z)} = \frac{\prod_{i=1}^{N}(1 - r\,e^{j\omega_i}\,z^{-1})}{\prod_{i=1}^{N}(1 - R\,e^{j\omega_i}\,z^{-1})}}$$

$R < r \quad \rightarrow \quad$ Notchfilter
$R > r \quad \rightarrow \quad$ Combfilter
$|R| < 1 \quad \cap \quad |r| \leq 1$

6.3.4 Inverse Filterung

Oft ist gefordert, eine bestimmte Filterung rückgängig zu machen (z.B. Kanalausgleichung). Dazu kann grundsätzlich die Inverse Filterfunktion verwendet werden.



Inversefilter: $\boxed{H_{\text{inv}}(z) = \frac{1}{H(z)}}$

Die Inversefilterung hat jedoch zwei bedeutende Probleme:

- Das Rauschen wird auch inverse gefiltert.
- Das Inversefilter muss stabil sein.

Rauschen:

Wird das verrauschte Signal inverse gefiltert, so wird das Rauschen in den Frequenzbereichen stark verstärkt, in denen das ursprüngliche Filter stark dämpfte $\rightarrow$ Verschlechterung der SNR!

$$\boxed{Y(z) = H(z)\cdot X(z) + N(z) \qquad \Rightarrow \qquad \hat{X}(z) = H_{\text{inv}}(z)\cdot Y(z) = X(z) + \frac{N(z)}{H(z)}}$$

Stabilität des Inversefilters:

Ist das ursprüngliche, kausale Filter $H(z)$ stabil (alle Pole innerhalb des Einheitskreises) so heisst dies nicht das das inverse Filter $H_{\text{inv}}(z)$ auch stabil ist. Durch die Invertierung werden alle Pole zu Nullstellen und alle Nullstellen zu Polen. Dies bedeutet, das Nullstellen, die vorher ausserhalb des Einheiskreises lagen zu Polen werden und das Filter im kausalen Fall dadruch instabil wird. Die Lösung dafür lautet, dass in diesem Fall die akausale Impulsantwort des Inversfilters genommen wird, diese um D Samples verzögert und wenn sie genügend abgeklungen ist (nach D Samples) einfach abgeschnitten wird. Mit dieser Lösung ist das Filter jedoch nur noch eine Approximation des eigentlichen Inversefilters.

7 Realisierung digitaler Filter S.265

7.1 Direktform

- Blockdiagramm kann direkt aus der I/O Differenzengleichung aufgezeichnet werden.
- Es sind sehr viele Speicherstellen notwendig.
- Innere Zustände (v_i, w_i) definieren, damit der Ausgang aus dem aktuellen Eingang und den Zuständen berechnet werden kann.
- Innere Zustände mit Null initialisieren $v_i = 0$ und $w_i = 0$

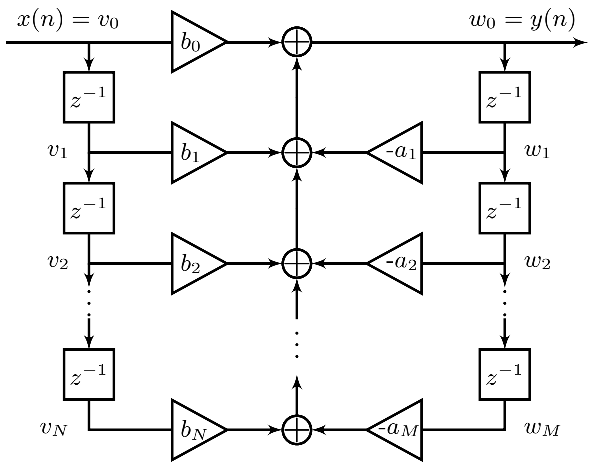
Achtung:
v_0 und w_0 sind keine Zustände, sondern Ein- und Ausgangssignal.

Übertragungsfunktion

$$H(z) = \frac{N(z)}{D(z)} = \frac{b_0 + b_1\, z^{-1} + b_2\, z^{-2} + \cdots + b_N\, z^{-N}}{1 + a_1\, z^{-1} + a_2\, z^{-2} + \cdots + a_M\, z^{-M}}$$

I/O Differenzengleichung

$$\begin{aligned} y(n) = \; & \big[b_0\, x(n) + b_1\, x(n-1) + \ldots + b_N\, x(n-N)\big] \\ & -\big[a_1\, y(n-1) + \ldots + a_M\, y(n-M)\big] \end{aligned}$$



Algorithmus

for each input x do:

$$\begin{aligned} v_0 &= x \\ w_0 &= b_0 v_0 + b_1 v_1 + \ldots + b_N v_N - a_1 w_1 - \ldots - a_M w_M \\ y &= w_0 \\ v_i &= v_{i-1} \qquad i = N, N-1, \ldots, 1 \\ w_i &= w_{i-1} \qquad i = M, M-1, \ldots, 1 \end{aligned}$$

7.2 Kanonische Form

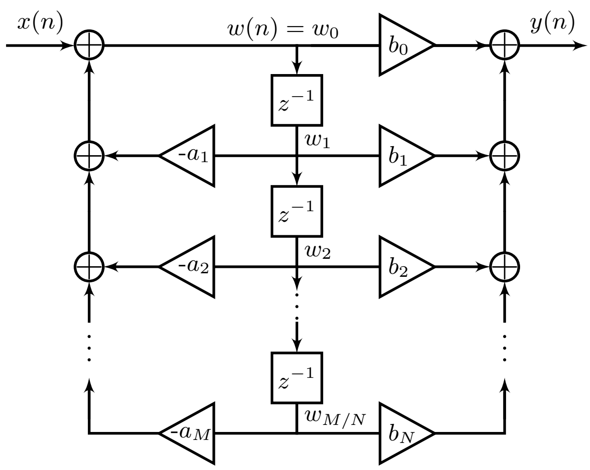
- Die Kanonische Form kann aus der Direktform abgeleitet werden. Dazu muss die rechte und linke Seite des Blockdiagrammes getauscht werden.
- Es sind viel weniger Speicherstellen notwendig.
- Innere Zustände (w_i) definieren, damit der Ausgang aus dem aktuellen Eingang und den Zuständen berechnet werden kann.
- Innere Zustände mit Null initialisieren $w_i = 0$

Übertragungsfunktion

$$H(z) = \frac{N(z)}{D(z)} = \frac{b_0 + b_1\, z^{-1} + b_2\, z^{-2} + \cdots + b_N\, z^{-N}}{1 + a_1\, z^{-1} + a_2\, z^{-2} + \cdots + a_M\, z^{-M}}$$

I/O Differenzengleichung

$$\begin{aligned} w(n) &= x(n) - a_1\, w(n-1) - \ldots - a_M\, w(n-M) \\ y(n) &= b_0\, w(n) + b_1\, w(n-1) + \ldots + b_N\, w(n-N) \end{aligned}$$



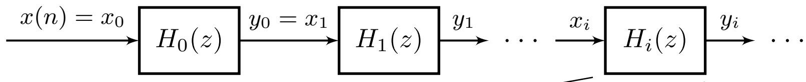
Algorithmus

for each input x do:

$$\begin{aligned} w_0 &= x - a_1 w_1 - a_2 w_2 - \ldots - a_M w_M \\ y &= b_0 w_0 + b_1 w_1 + b_2 w_2 + \ldots + b_N w_N \\ w_i &= w_{i-1} \qquad i = K, K-1, \ldots, 1 \\ & \qquad\qquad\quad K = \max\{N, M\} \end{aligned}$$

7.3 Kaskaden Form

- Filter mit reellen Impulsantworten (reelle Koeffizienten) können als kaskadierte Second Order Sections (SOS) geschrieben werden.
- Eine Second Order Section (IIR-Filter 2. Ordnung) kann in einer beliebigen Form umgesetzt werden. Häufig wird sie jedoch in der Kanonischen Form realisiert.

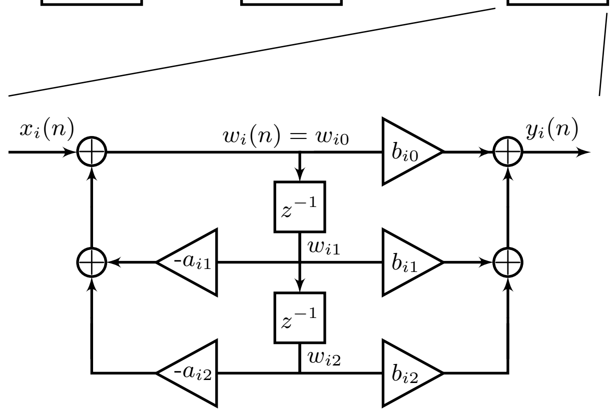


Übertragungsfunktion

$$\boxed{H(z) = \prod_{i=0}^{K-1} H_i(z) = \prod_{i=0}^{K-1} \frac{b_{i0} + b_{i1}\, z^{-1} + b_{i2}\, z^{-2}}{1 + a_{i1}\, z^{-1} + a_{i2}\, z^{-2}}}$$

Faktorisierung in SOS

$$\boxed{H(z) = \frac{N(z)}{D(z)} = \prod_{i=0}^{K-1} H_i(z)}$$



- Die Nullstellen der beiden reellen Polynome $N(z)$ und $D(z)$ finden.

$$\begin{aligned} D(z) &= 1 + a_1\, z^{-1} + a_2\, z^{-2} + \ldots + a_M\, z^{-M} \\ &= (1 - p_1 z^{-1}) \cdot (1 - p_2 z^{-1}) \ldots (1 - p_M z^{-1}) \end{aligned}$$

- Konjugiert-Komplexe Pole zusammenfassen $p_2 = p_1^*$

$$\begin{aligned} (1 - p_1 z^{-1}) \cdot (1 - p_2\, z^{-1}) &= (1 - (p_1 + p_2)\, z^{-1} + p_1 p_2\, z^{-2}) \\ &= (1 - (p_1 + p_1^*)\, z^{-1} + p_1 p_1^*\, z^{-2}) \\ &= (1 - 2\mathrm{Re}(p_1)\, z^{-1} + |p_1|^2\, z^{-2}) \\ &= (1 - 2R\cos(\theta)\, z^{-1} + R^2\, z^{-2}) \end{aligned}$$

- SOS von $D(z)$ und von $N(z)$ nach belieben zusammennehmen.

8 Digitale Waveform Generatoren S.316

Das Ziel ist mit einem Filter $H(z)$ bzw. dessen Impulsantwort $h(n)$ eine gewünschte Waveform zu generieren. Dazu gibt es zwei Möglichkeiten:

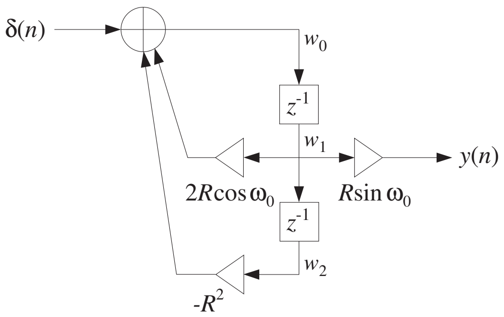
- Die Waveform Sample für Sample berechnen (Impulsantwort $h(n)$ des Filters $H(z)$)
- Waveform vorab berechnen und in einer Wavetable abspeichern. Die Frequenz periodischer Signale kann so über die Auslesegeschwindigkeit gesteuert werden

8.1 Sinus-Generator

Sinus-Signal-Generator

$$\boxed{h(n) = R^n \sin(\omega_0 n)\, u(n)}$$

$$\boxed{H(z) = \frac{R \sin(\omega_0)\, z^{-1}}{1 - 2\, R \cos(\omega_0)\, z^{-1} + R^2\, z^{-2}}}$$

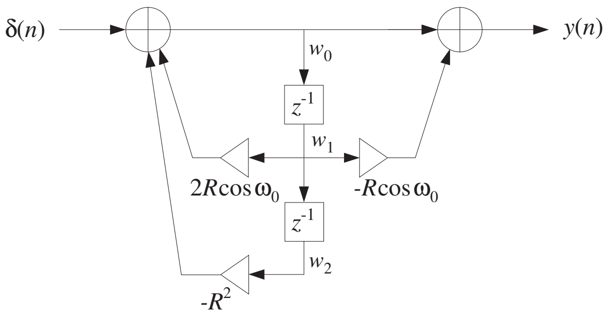


$$\begin{aligned} &for\ n = 0, 1, 2, \dots\ do: \\ &\quad w_0 = 2R\cos(\omega_0)\, w_1 - R^2\, w_2 + \delta(n) \\ &\quad y = R\sin(\omega_0)\, w_1 \\ &\quad w_2 = w_1 \\ &\quad w_1 = w_0 \end{aligned}$$

Cosinus-Signal-Generator

$$\boxed{h(n) = R^n \cos(\omega_0 n)\, u(n)}$$

$$\boxed{H(z) = \frac{1 - R \cos(\omega_0)\, z^{-1}}{1 - 2\, R \cos(\omega_0)\, z^{-1} + R^2\, z^{-2}}}$$



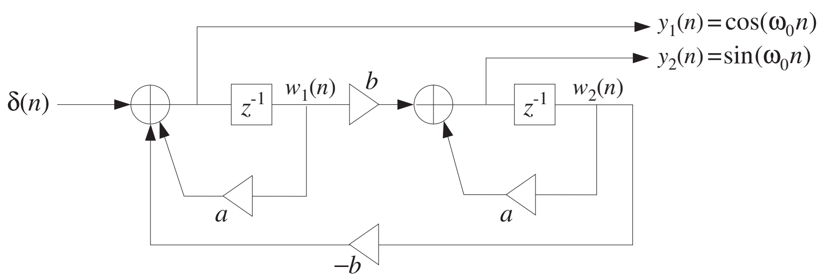
$$\begin{aligned} &for\ n = 0, 1, 2, \dots\ do: \\ &\quad w_0 = 2R\cos(\omega_0)\, w_1 - R^2\, w_2 + \delta(n) \\ &\quad y = w_0 - R\cos(\omega_0)\, w_1 \\ &\quad w_2 = w_1 \\ &\quad w_1 = w_0 \end{aligned}$$

Gekoppelter-Sinus-Cosinus-Generator

$$\boxed{h_1(n) = R^n \cos(\omega_0 n)\, u(n)} \qquad \boxed{h_2(n) = R^n \sin(\omega_0 n)\, u(n)}$$

$$\boxed{H_1(z) = \frac{1 - a\, z^{-1}}{1 - 2\, a\, z^{-1} + (a^2 + b^2)\, z^{-2}}} \qquad \boxed{H_2(z) = \frac{b\, z^{-1}}{1 - 2\, a\, z^{-1} + (a^2 + b^2)\, z^{-2}}}$$

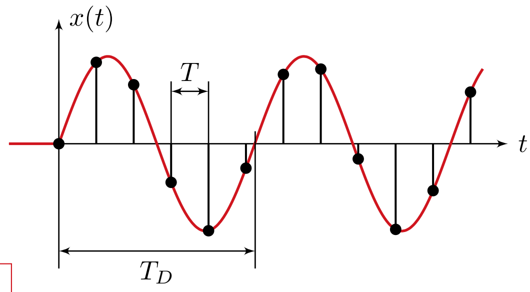
$$\boxed{a = R \cos(\omega_0)} \qquad \boxed{b = R \sin(\omega_0)}$$



$$\begin{aligned} &for\ n = 0, 1, 2, \dots\ do: \\ &\quad y_1 = a\, w_1 - b\, w_2 + \delta(n) \\ &\quad y_2 = a\, w_2 + b\, w_1 \\ &\quad w_1 = y_1 \\ &\quad w_2 = y_2 \end{aligned}$$

8.2 Periodische Waveform Generatoren

8.2.1 Periodische Signale



Achtung:
Die abgetastete Version $x(n)$ eines periodischen Signals $x(t)$ ist nicht zwingend periodisch!

Damit das abgetastete Signal $x(n)$ periodisch ist muss gelten:

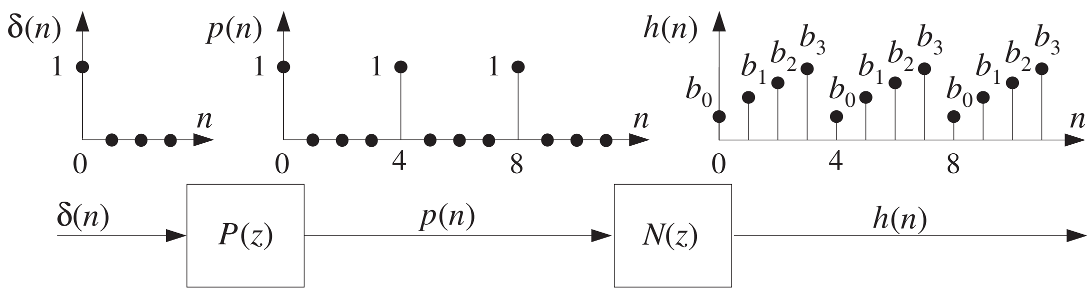
T_D ist ein ganzzahliges Vielfaches der Abtastperiode T $\quad \boxed{T_D = D \cdot T} \quad \Rightarrow \quad \boxed{f = \frac{f_s}{D}}$

analoge Periodizität: $\boxed{x(t) = x(t + T_D)}$

digitale Periodizität: $\boxed{x(nT) = x(nT + DT) = x(nT + T_D)}$

8.2.2 Realisierungsformen periodischer Waveform Generatoren

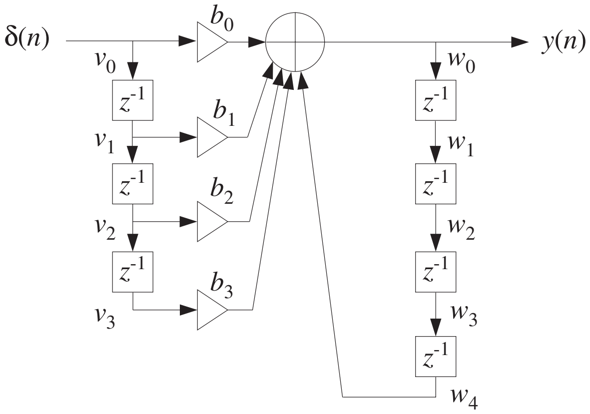
- Die ersten D-Samples $(h(0), ..., h(D-1))$ mit einem FIR-Filter spezifizieren und das FIR-Filter periodisch mit Impulsen anregen $(\delta(n), \delta(n-D), \delta(n-2D), ...)$.
- FIR-Filter $N(z)$: $\boxed{N(z) = b_0 + b_1\, z^{-1} + b_2\, z^{-2} + ... + b_{D-1}\, z^{-(D-1)}}$
- Periodische Anregung $P(z)$: $\boxed{P(z) = 1 + z^{-D} + z^{-2D} + z^{-3D} + ... = \frac{1}{1 - z^{-D}}}$
- Übertragungsfunktion $H(z)$: $\boxed{H(z) = P(z) \cdot N(z) = \frac{b_0 + b_1\, z^{-1} + b_2\, z^{-2} + ... + b_{D-1}\, z^{-(D-1)}}{1 - z^{-D}}}$
- Impulsantwort $h(n)$: $\boxed{h(n) = \underbrace{h(n-D)}_{\text{weitere Perioden}} + \underbrace{b_0\, \delta(n) + b_1\, \delta(n-1) + ... + b_{D-1}\, \delta(n-(D-1))}_{\text{1. Periode}}}$



Implementierung mit Direktform

- Zustände v_i und w_i werden mit Null initialisiert.
- FIR-Filter-Teil der angeregt wird, wodurch die ersten D Samples generiert werden.
- IIR-Filter-Teil der die D Samples wiederholt.

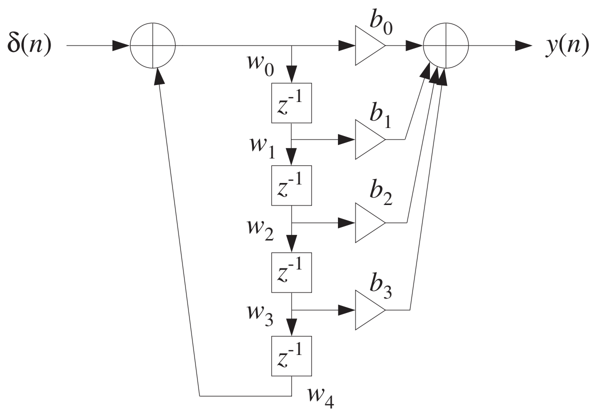
$$
\begin{array}{lcll}
\multicolumn{4}{l}{for\ n = 0, 1, 2, ...\ do:} \\
v_0 & = & \delta(n) & \\
w_0 & = & b_0\, v_0 + b_1\, v_1 + b_2\, v_2 + ... + b_{D-1}\, v_{D-1} & \\
y & = & w_0 & \\
v_i & = & v_{i-1} & i = D-1, \ldots, 1 \\
w_i & = & w_{i-1} & i = D, D-1, \ldots, 1
\end{array}
$$



n	v_0	v_1	v_2	v_3	w_0	w_1	w_2	w_3	w_4	$y = w_0$
0	1	0	0	0	b_0	0	0	0	0	b_0
1	0	1	0	0	b_1	b_0	0	0	0	b_1
2	0	0	1	0	b_2	b_1	b_0	0	0	b_2
3	0	0	0	1	b_3	b_2	b_1	b_0	0	b_3
4	0	0	0	0	b_0	b_3	b_2	b_1	b_0	b_0
5	0	0	0	0	b_1	b_0	b_3	b_2	b_1	b_1
6	0	0	0	0	b_2	b_1	b_0	b_3	b_2	b_2
7	0	0	0	0	b_3	b_2	b_1	b_0	b_3	b_3
8	0	0	0	0	b_0	b_3	b_2	b_1	b_0	b_0

Implementierung mit kanonischer Form

- Zustände w_i werden mit Null initialisiert.
- FIR-Filter-Teil der periodisch angeregt wird, wodurch die D Samples der Impulsantwort immer wieder generiert werden.
- IIR-Filter-Teil der den Impuls $\delta(n)$ alle D Samples wiederholt.



n	w_0	w_1	w_2	w_3	w_4	$y = b_0w_0 + b_1w_1 + b_2w_2 + b_3w_3$
0	1	0	0	0	0	b_0
1	0	1	0	0	0	b_1
2	0	0	1	0	0	b_2
3	0	0	0	1	0	b_3
4	1	0	0	0	1	b_0
5	0	1	0	0	0	b_1
6	0	0	1	0	0	b_2
7	0	0	0	1	0	b_3

$$
\begin{array}{lcll}
\multicolumn{4}{l}{for\ n = 0, 1, 2, \dots\ do:} \\
w_0 & = & w_D + \delta(n) & \\
y & = & b_0\, w_0 + b_1\, w_1 + b_2\, w_2 + \dots + b_{D-1}\, w_{D-1} & \\
w_i & = & w_{i-1} & i = D, D-1, \dots, 1
\end{array}
$$

8.2.3 Spektrum periodischer Waveform Generatoren

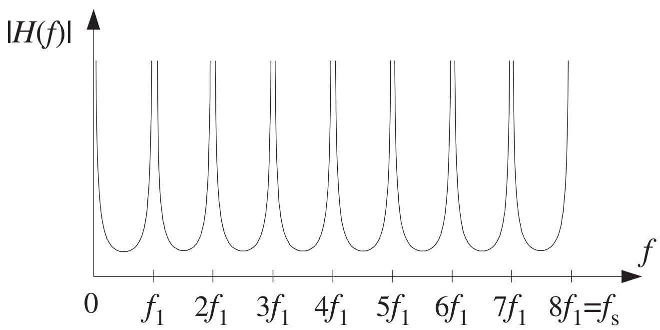
- Da die Signalsequenzen $x(n)$ kausal (einseitig) sind, hat das Spektrum keine klaren Spektrallinien, sondern dominante Peaks bei der Grundfrequenz f und dessen Harmonischen mf

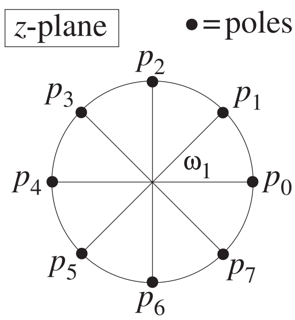
$$H(\omega) = \frac{N(\omega)}{1 - \mathrm{e}^{-j\omega D}} = \frac{N(\omega)}{2j\mathrm{e}^{j\omega D/2} \sin\left(\frac{\omega D}{2}\right)}$$

$$\Rightarrow \quad \boxed{|H(\omega)| = \frac{|N(\omega)|}{2\left|\sin\left(\frac{\omega D}{2}\right)\right|}} \quad \boxed{|H(f)| = \frac{|N(f)|}{2\left|\sin\left(\frac{\pi f D}{f_s}\right)\right|}}$$

- Die Peaks kommen von den Polstellen von $|H(f)|$.

$$\sin\left(\frac{\pi f D}{f_s}\right) = 0 \quad \Rightarrow \quad \frac{\pi f D}{f_s} = m\pi \quad \Rightarrow \quad \boxed{f_m = m\frac{f_s}{D} \qquad \omega_m = \frac{2\pi m}{D}} \qquad m = 0, 1, \dots, D-1$$





$$\boxed{z = p_m = \mathrm{e}^{j\omega_m}}$$

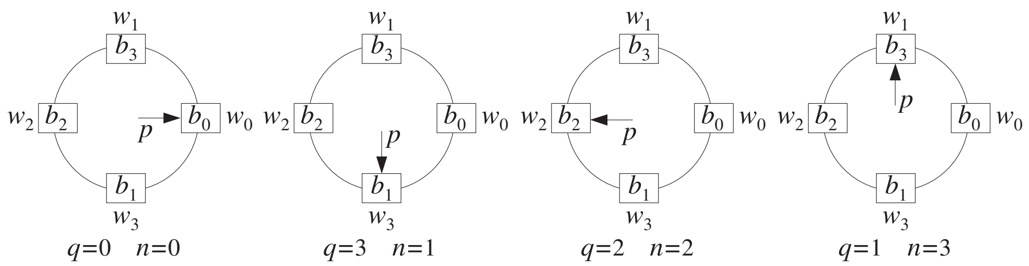
8.3 Wavetable Generatoren

- Statt zu filtern, können die D Samples einer Periode auch in einer Tabelle abgespeichert werden und anschliessend immer wieder "abgespielt" werden.
- Die Tabelle wird ein einem zirkularen Buffer $\boldsymbol{w} = [w_0, ..., w_{D-1}]$ gespeichert und über einen Pointer $^*p = w[q]$, der rückwärts im Kreis wandert, ausgelesen.
- Das Signal kann ganz einfach mit dem Startpunkt des Pointers p verzögert werden:
  nicht verzögern: $\boxed{p = \&w[0]}$ um m-Samples verzögern: $\boxed{p = \&w[m]}$
- Der Buffer wird rückwärts gefüllt

$$\boxed{\begin{array}{l} for\ i = 0, 1, ..., D-1\ do: \\ \quad w[(D-i)\%D] = b[i]; \end{array}} \quad \text{oder} \quad \boxed{\begin{array}{l} for\ i = 0, 1, ..., D-1\ do: \\ \quad w[i] = b[(D-i)\%D]; \end{array}}$$

- Das Auslesen aus dem Buffer kann folgendermassen gemacht werden.

$$\boxed{\begin{array}{l} q_0 = 0; \\ p = \&w[q_0]; \\ do\ forever: \\ \quad y = *p; \\ \quad q_{n+1} = (q_n - 1)\%D; \\ \quad p = \&w[q_{n+1}]; \end{array}}$$



8.3.1 Frequenz-Änderungen durch Wavetable-Manipulation

- Die Abtastfrequenz f_s, die Länge der Wavetable D und die Periode T_D bzw. die fundamentale Frequenz f hängen folgendermassen zusammen:

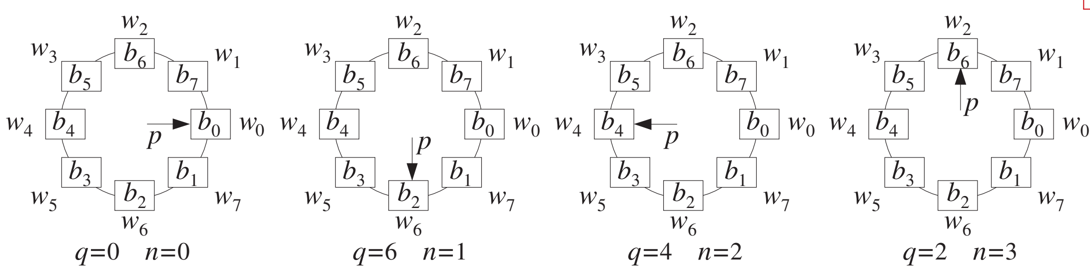
$$\boxed{T_D = D\,T \quad \Rightarrow \quad f = \frac{f_s}{D}}$$

- Die fundamentale Frequenz f kann verändert werden, indem die Länge der Wavetable D verändert wird.

$$\boxed{T_d = d\,T \quad \Rightarrow \quad f = \frac{f_s}{d}} \quad d \le D$$

- Wenn D ein ganzzahliges Vielfaches von d ist werden einfach Samples in der Wavetable übersprungen.
  Beispiel:
  Nur jedes zweite Sample ausgeben $\Rightarrow$ fundamentale Frequenz f verdoppeln.

$$\boxed{c = \frac{D}{d}} \quad c \in \mathbb{Z}$$



- Ist D kein ganzzahliges Vielfaches von d entstehen keine ganzzahligen Offsets und damit keine ganzzahligen Indizes q_n. Folglich müssen
  - Die Indizes aufgerundet, abgerundet oder normal gerundet werden.
  - Die Koeffizienten b_i der Indizes davor und danach $(w[\lceil q_n \rceil], w[\lfloor q_n \rfloor])$ linear interpoliert werden.
- Der Update des Indexes wird folgendermassen gemacht

$$\boxed{y(n) = {}^*p = w[q_n]} \quad \boxed{q_{n+1} = (q_n - c)\%D}$$

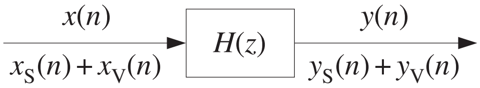
Achtung:
Damit das Abtasttheorem nicht verletzt wird muss f innerhalb des Nyquist-Intervalls bleiben und für c gilt damit:

$$\boxed{-\frac{D}{2} \le c \le \frac{D}{2}}$$

9 Rausch-Reduktion und Signal-Verbesserung S.382

9.1 Rausch-Reduktions-Filter

Ziel:
Das Ziel eines Rausch-Reduktions-Filters $H(z)$ ist es, das eigentliche Signal $x_S(n)$ aus dem Messsignal $x(n)$ herauszufiltern und dabei das Rauschen $x_V(n)$ zu unterdrücken.



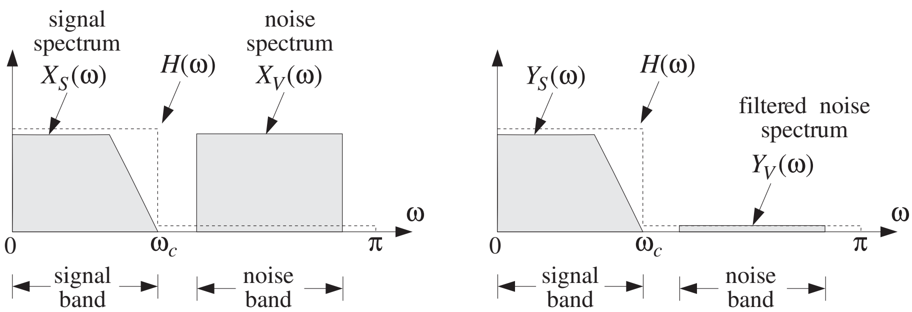
$$\boxed{\begin{aligned} y_S(n) &= x_S(n-delay) \\ y_V(n) &= 0 \end{aligned}}$$

9.1.1 Ideales Rausch-Reduktions-Filter

Damit das Rauschen vollkommen unterdrückt werden kann ohne das eigentliche Signal $x_S(n)$ zu verändern muss im Frequenzbereich folgendes gelten:

$$\boxed{\begin{aligned} Y_S(\omega) &= H(\omega)\,X_S(\omega) = X_S(\omega) \\ Y_V(\omega) &= H(\omega)\,X_V(\omega) = 0 \end{aligned}}$$

$\Rightarrow$ **Die beiden Spektren $X_S(\omega)$ und $X_V(\omega)$ dürfen nicht überlappen!**

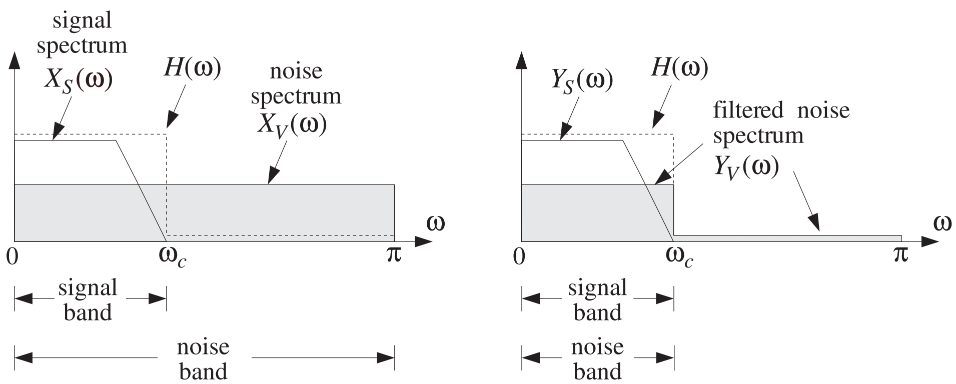


9.1.2 Bestmögliches Rausch-Reduktions-Filter

Ist das ideale Rausch-Reduktions-Filter nicht realisierbar, weil die beiden Spektren $X_S(\omega)$ und $X_V(\omega)$ überlappen ist die bestmögliche Lösung ein **idealer Bandpass** (hier Tiefpass).

- Das eigentliches Signal $x_S(n)$ wird nicht verzerrt.
- Das Rauschen $x_V(n)$ wird maximal unterdrückt.

$$\boxed{Y_S(\omega) = H(\omega)\,X_S(\omega) = X_S(\omega)}$$



Rauschen:

Für ein Mittelwertfreies-Weisses-Rauschen $x_V(n)$ gelten folgende Beziehungen:

Leistungsdichtespektrum und Leistung des Mittelwertfreiem-Weissem-Rauschen $x_V(n)$

$$\boxed{S_{X_V X_V}(\omega) = \sigma_{X_V}^2}$$

Leistungsdichtespektren des Farbigen-Ausgangsrauschen $y_V(n)$

$$\boxed{S_{Y_V Y_V}(\omega) = \left|H(\omega)\right|^2 S_{X_V X_V}(\omega) = \left|H(\omega)\right|^2 \sigma_{X_V}^2}$$

Leistung des Farbigen-Ausgangsrauschen

$$\boxed{\sigma_{Y_V}^2 = \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{Y_V Y_V}(\omega)\,d\omega = \sigma_{X_V}^2 \frac{1}{2\pi}\int_{-\pi}^{\pi} \left|H(\omega)\right|^2 d\omega = \sigma_{X_V}^2\, NRR}$$

Noise-Reduction-Ratio NRR (sollte möglichst klein sein)

$$\boxed{NRR = \frac{\sigma_{Y_V}^2}{\sigma_{X_V}^2} = \frac{1}{2\pi}\int_{-\pi}^{\pi} \left|H(\omega)\right|^2 d\omega = \sum_n h(n)^2}$$

Signal-Rausch-Abstand SNR:

Durch das Rausch-Reduktions-Filter ergeben sich folgende Signal-Rausch-Abstände:

SNR am Eingang $\boxed{\text{SNR}_{in} = \frac{E[x_S(n)^2]}{E[x_V(n)^2]}}$ SNR am Ausgang $\boxed{\text{SNR}_{out} = \frac{E[y_S(n)^2]}{E[y_V(n)^2]}}$

SNR-Verbesserung
$$\boxed{\frac{\text{SNR}_{out}}{\text{SNR}_{in}} = \frac{E[y_S(n)^2]}{E[y_V(n)^2]} \cdot \frac{E[x_V(n)^2]}{E[x_S(n)^2]} = \underbrace{\frac{E[x_V(n)^2]}{E[y_V(n)^2]}}_{1/NRR} \cdot \frac{E[y_S(n)^2]}{E[x_S(n)^2]}}$$

$\Rightarrow$ Wenn das Signal $x_S(n)$ durch das Filter $H(\omega)$ nicht verändert wird gilt:

$$\boxed{\frac{\text{SNR}_{out}}{\text{SNR}_{in}} = \frac{1}{NRR}}$$

NRR von idealen Filtern:

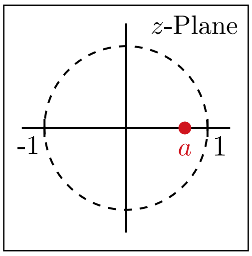
Idealer Tiefpass
$$\boxed{NRR = \frac{\sigma_{Y_V}^2}{\sigma_{X_V}^2} = \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} 1\, d\omega = \frac{2\omega_c}{2\pi} = \frac{\omega_c}{\pi}}$$

Idealer Bandpass
$$\boxed{NRR = \frac{\sigma_{Y_V}^2}{\sigma_{X_V}^2} = \frac{2}{2\pi}\int_{\omega_a}^{\omega_b} 1\, d\omega = \frac{\omega_b - \omega_a}{\pi}}$$

9.1.3 Reale Rausch-Reduktions-Filter

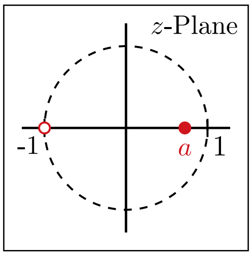
Ideale Filter mit einer unendlich kurzen Übergangsbandbreite können nicht implementiert werden. Reale Rausch-Reduktions-Filter haben eine gewisse Übergangsbandbreite und sind oft linearphasig.

IIR-Smoother erster Ordnung



- Geeignet für DC-Signale $x_S(n)$
- Mittelung über alle bisherigen Samples, wobei die Samples mit dem Faktor $a^d\, b\, x(n-d)$ gewichtet werden. Je weiter ein Sample zurückliegt, desto weniger Einfluss hat es. $\boxed{y(n) = b\,x(n) + a\,y(n-1)}$ $\boxed{b = 1-a}$
- Signalband: $\boxed{\omega = 0}$
- Übertragungsfunktion: $\boxed{H(z) = \frac{b}{1-a\,z^{-1}} \quad\Rightarrow\quad \left|H(\omega)\right|^2 = \frac{b^2}{1-2\,a\cos(\omega)+a^2}}$
- Grenzfrequenz: $\boxed{\left|H(\omega_c)\right|^2 = \frac{1}{2} \quad\Rightarrow\quad \cos(\omega_c) = 1 - \frac{(1-a)^2}{2\,a}}$
- DC- und AC-Gain: $\boxed{\text{DC: } H(z)\big|_{z=1} = \frac{b}{1-a} = 1 \qquad \text{AC: } H(z)\big|_{z=-1} = \frac{1-a}{1+a}}$
- Noise-Reduction-Ratio: $\boxed{NRR = \frac{1-a}{1+a}}$ Einschwingzeit: $\boxed{n_{\text{eff}} = \frac{\ln(\epsilon)}{\ln(a)}}$

IIR-Smoother erster Ordnung mit bestimmter Grenzfrequenz

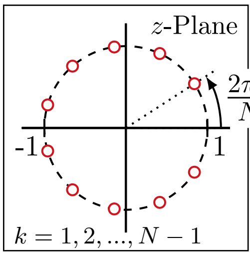


- Geeignet für DC-Signale $x_S(n)$
- Mittelung über alle bisherigen Samples, wobei die Samples mit dem Faktor $(a^d + a^{d-1})\, b\, x(n-d)$ gewichtet werden. Je weiter ein Sample zurückliegt, desto weniger Einfluss hat es. $\boxed{y(n) = b\,x(n) + b\,x(n-1) + a\,y(n-1)}$

$$\boxed{a = \frac{1-\tan(\omega_c/2)}{1+\tan(\omega_c/2)}} \qquad \boxed{b = \frac{1-a}{2}}$$

- Signalband: $\boxed{\omega = 0}$
- Übertragungsfunktion: $\boxed{H(z) = \frac{b\,(1+z^{-1})}{1-a\,z^{-1}} \quad \Rightarrow \quad \left|H(\omega)\right|^2 = \frac{2b^2(1+\cos(\omega))}{1-2\,a\cos(\omega)+a^2}}$
- Grenzfrequenz: $\boxed{\left|H(\omega_c)\right|^2 = \frac{1}{2} \quad \Rightarrow \quad \cos(\omega_c) = \frac{2\,a}{1+a^2}}$
- DC- und AC-Gain: $\boxed{\text{DC: } H(z)\big|_{z=1} = \frac{2b}{1-a} = 1 \qquad \text{AC: } H(z)\big|_{z=-1} = 0}$
- Noise-Reduction-Ratio: $\boxed{NRR = \frac{1-a}{2}}$ Einschwingzeit: $\boxed{n_{\text{eff}} = \frac{\ln(\epsilon)}{\ln(a)}}$

FIR-Mittelungs-Filter



- Geeignet für DC-Signale und tieffrequente Signale $x_S(n)$
- Mittelung über die letzten N Samples, wobei alle gleich stark gewichtet werden $(1/N)$, da dadurch die NNR minimiert wird.

$$\boxed{y(n) = \frac{1}{N}\left(x(n) + x(n-1) + x(n-2) + ... + x(n-(N-1))\right)}$$

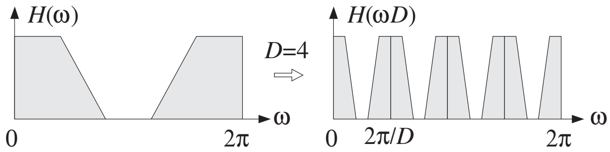
- Signalband: $\boxed{0 \le \omega \le \omega_c}$
- Übertragungsfunktion: $\boxed{H(z) = \frac{1}{N}\left(1 + z^{-1} + ... + z^{-(N-1)}\right) = \frac{1}{N}\,\frac{1-z^{-N}}{1-z^{-1}}}$
- Frequenzgang: $\boxed{\left|H(\omega)\right|^2 = \frac{1}{N^2}\,\frac{\sin^2(N\omega/2)}{\sin^2(\omega/2)}}$
- Grenzfrequenz: $\boxed{\left|H(\omega_c)\right|^2 \overset{\text{N gross}}{\approx} \frac{4}{\pi^2} \approx 0.41 \;\widehat{=}\; 3.9\text{dB} \quad \Rightarrow \quad \omega_c = \frac{\pi}{N}}$
- DC- und AC-Gain: $\boxed{\text{DC: } H(z)\big|_{z=1} = 1 \qquad \text{AC: } H(z)\big|_{z=-1} = \begin{cases} 0, & N \text{ gerade} \\ 1/N, & N \text{ ungerade} \end{cases}}$
- Noise-Reduction-Ratio: $\boxed{NRR = \frac{1}{N}}$ Einschwingzeit: $\boxed{n_{\text{eff}} = N}$

9.2 Notch- und Comb-Filter

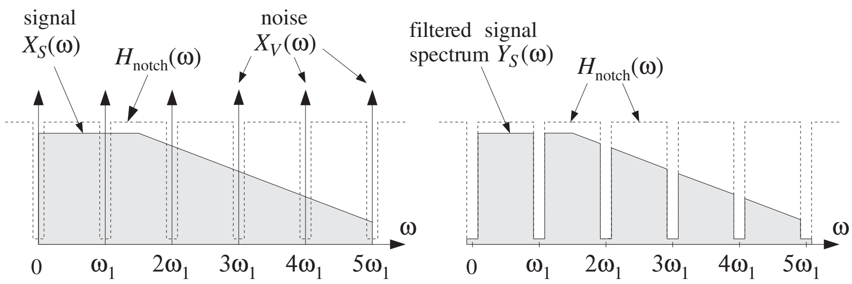
Diese Filter sind gut geeignet, wenn das Rauchen $x_V(n)$ oder das Signal $x_S(N)$ periodisch sind.

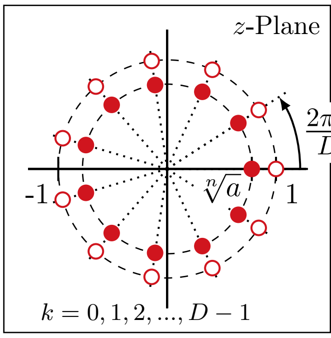
- **Notch-Filter:** Das periodische Rauschen (50 Hz Netzbrummen) wird durch Notches (Kerben) bei den Rauschfrequenzen vernichtet.
- **Comb-Filter:** Die Spektrallinien des periodische Signal werden durch die Combs herausgefiltert und das umliegende Rauschen wird vernichtet.
- Die Abtastfrequenz f_s wird am besten auf ein ganzzahliges Vielfaches der fundamentalen Frequenz f (Signal- oder Rauschfrequenz) gesetzt. $\boxed{f_s = D \cdot f}$
- Ein Multi-Notch/Comb-Filter kann aus einem einfachen Filter erster Ordnung abgeleitet werden, indem z durch z^D ersetzt wird. Dadurch wird das periodische Spektrum um den Faktor D zusammen gestaucht.

$$\boxed{z \;\rightarrow\; z^D \quad \Rightarrow \quad H(\omega) \;\rightarrow\; H(\omega D)}$$



9.2.1 Notch-Filter



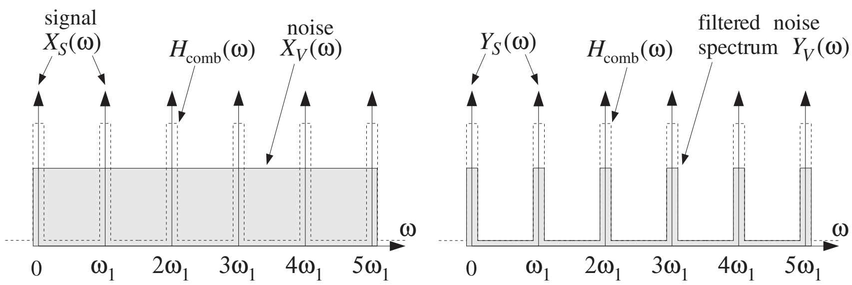


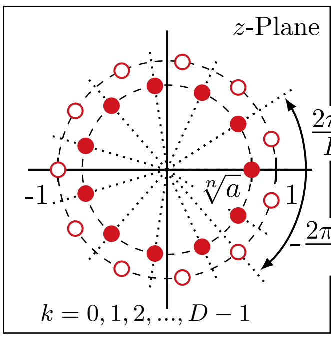
3 dB-Bandbreite $\Delta\omega$: $\boxed{\beta = \tan\left(\frac{D\Delta\omega}{4}\right), \qquad a = \frac{1-\beta}{1+\beta}, \qquad b = \frac{1}{1+\beta}}$

$$\begin{matrix} 0 \le a < 1 \\ 0 < \beta \le 1 \end{matrix} \quad \Rightarrow \quad \Delta\omega \le \frac{\pi}{D} \quad \Rightarrow \quad \Delta f \le \frac{f_s}{2D}$$

Übertragungsfunktion: $\boxed{H_{\text{notch}}(z) = b\frac{1-z^{-D}}{1-a\,z^{-D}}}$ $\boxed{\left|H_{\text{notch}}(\omega)\right|^2 = \frac{\tan^2(\omega D/2)}{\tan^2(\omega D/2)+\beta^2}}$ $\boxed{NRR = b = \frac{1+a}{2}}$

9.2.2 Comb-Filter





3 dB-Bandbreite $\Delta\omega$: $\boxed{\beta = \tan\left(\frac{D\Delta\omega}{4}\right), \qquad a = \frac{1-\beta}{1+\beta}, \qquad b = \frac{\beta}{1+\beta}}$

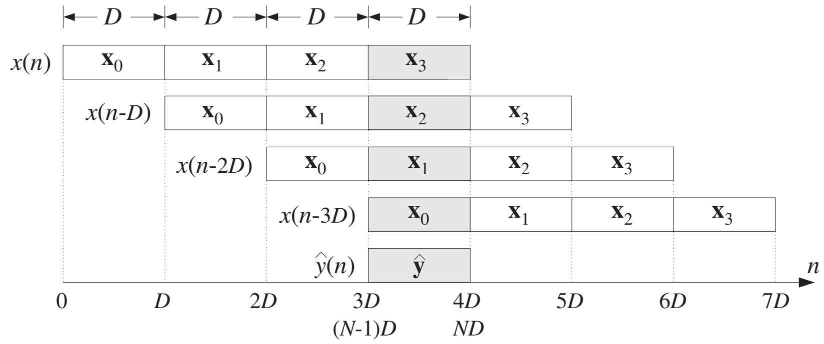
$$\begin{matrix} 0 \le a < 1 \\ 0 < \beta \le 1 \end{matrix} \quad \Rightarrow \quad \Delta\omega \le \frac{\pi}{D} \quad \Rightarrow \quad \Delta f \le \frac{f_s}{2D}$$

Übertragungsfunktion: $\boxed{H_{\text{comb}}(z) = b\frac{1+z^{-D}}{1-a\,z^{-D}}}$ $\boxed{\left|H_{\text{comb}}(\omega)\right|^2 = \frac{\beta^2}{\tan^2(\omega D/2)+\beta^2}}$ $\boxed{NRR = b = \frac{1-a}{2}}$

Die Noise-Reduction-Ratio bleibt gleich, wenn das Spektrum nur gestaucht wird. D.h. Die NNR der Notch- und Comb-Filter entspricht der NNR der entsprechenden Filter erster Ordnung.

9.3 Signal-Mittelung

Ziel:
Von einem periodischen verrauschten Signal werden die Perioden gemittelt um das Rauschen zu unterdrücken. Diese Methode der Rauschunterdrückung resultiert in einem FIR-Comb-Filter und wird für die Verarbeitung von endlich langen Signal-Aufzeichnungen verwendet.



$\Rightarrow$ **Mittelung über N Perioden mit der Länge D**

Gemitteltes Signal: $\boxed{\widehat{y}(n) = \frac{1}{N}\sum_{i=0}^{N-1} x(n - i\,D) = \frac{1}{N}\sum_{i=0}^{N-1} x_i(n)}$ $\quad n = 0, 1, 2, ..., D-1$

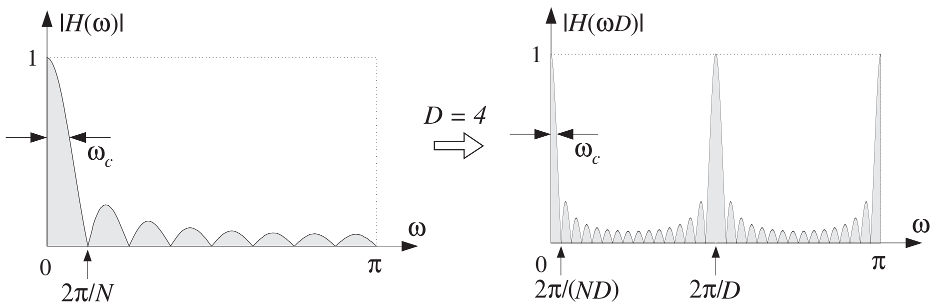
Übertragungsfunktion: $\boxed{H(z) = \frac{1}{N}(1 + z^{-D} + z^{-2D} + ... + z^{-(N-1)D}) = \frac{1}{N}\frac{1 - z^{-ND}}{1 - z^{-D}}}$

Frequenzgang: $\boxed{H(\omega) = \frac{1}{N}\frac{\sin(ND\omega/2)}{\sin(D\omega/2)}}$ $\quad \boxed{\left|H(\omega)\right|^2 = \frac{1}{N^2}\frac{\sin^2(ND\omega/2)}{\sin^2(D\omega/2)}}$

Grenzfrequenz: $\boxed{\omega_c = \frac{\pi}{ND}}$

Noise-Reduction-Ratio: $\boxed{NRR = \frac{1}{N}}$

Dieses Filter erhält man auch, wenn man das Spektrum des FIR-Mittelungs-Filter $y(n) = [x(n) + x(n-1) + ... + x(n-(N-1))]/N$ um den Faktor D staucht (z durch z^D ersetzen)



Es resultiert ein Ausgangssignal das um den Faktor $1/N$ weniger stark rauscht.

$$\boxed{\begin{aligned}\widehat{y}(n) &= y_S(n) + y_V(n)\\ &= x_S(n) + y_V(n)\end{aligned}} \qquad \boxed{\sigma^2_{Y_V} = \frac{1}{N}\sigma^2_{X_V}}$$

9.4 Savitzky-Golay-Smoothing-Filter

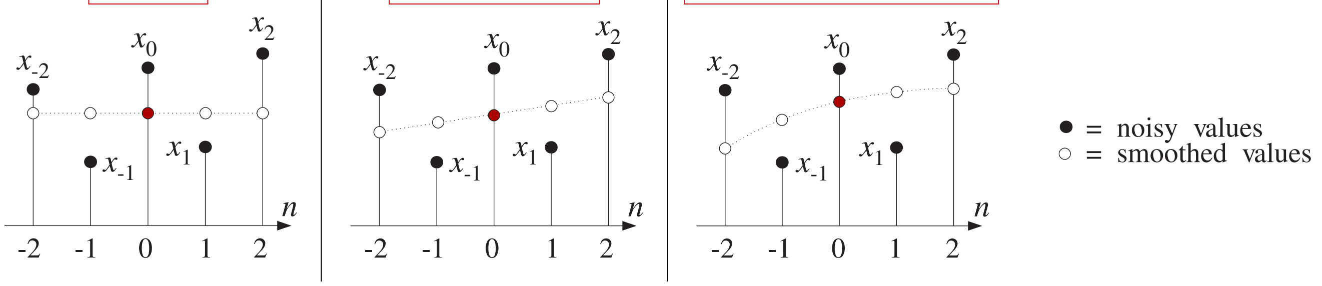
Ziel:
Die Idee ist, ein Polynom p-ter Ordnung optimal (Least-Square) in $N = 2M+1$ gemessene Samples einzupassen und anschliessend das mittlere Sample durch den Wert des Polynoms an dieser Stelle zu ersetzen. Dadurch wird das Rauschen reduziert. Zudem muss gelten $\boxed{N \geq p+1}$

Polynom-Beispiele mit N=5 Samples:

Konstante: $\boxed{\hat{x}_m = c_0}$

Linear: $\boxed{\hat{x}_m = c_0 + c_1\, m}$

Quadratisch: $\boxed{\hat{x}_m = c_0 + c_1\, m + c_2\, m^2}$



9.4.1 Optimierungsproblem

Das Ziel ist ein Polynom p-ten Grades zu finden, welches den kleinsten quadratischen Fehler hat.

Minimiere:
$$\mathcal{J} = \sum_{m=-M}^{M} e_m^2 = \sum_{m=-M}^{M} (x_m - \hat{x}_m)^2 = \sum_{m=-M}^{M} (x_m - (c_0 + c_1\, m + c_2\, m^2 + \ldots + c_p\, m^p))^2$$

oder in Matrixschreibweise:
$$\mathcal{J} = \vec{e}^{\mathsf{T}}\vec{e} = (\vec{x} - S\,\vec{c})^{\mathsf{T}}(\vec{x} - S\,\vec{c}) = \vec{x}^{\mathsf{T}}\vec{x} - 2\,\vec{c}^{\mathsf{T}}S^{\mathsf{T}}\vec{x} + \vec{c}^{\mathsf{T}}S^{\mathsf{T}}S\,\vec{c}$$

mit
$$\vec{c} = \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_p \end{bmatrix}, \qquad \vec{x} = \begin{bmatrix} x_{-M} \\ \vdots \\ x_{-1} \\ x_0 \\ x_1 \\ \vdots \\ x_M \end{bmatrix}, \qquad \vec{\hat{x}} = \begin{bmatrix} \hat{x}_{-M} \\ \vdots \\ \hat{x}_{-1} \\ \hat{x}_0 \\ \hat{x}_1 \\ \vdots \\ \hat{x}_M \end{bmatrix} = S\,\vec{c}, \qquad \vec{e} = \vec{x} - \vec{\hat{x}}$$

und
$$S = \begin{bmatrix} \vec{s}_0 & \vec{s}_1 & \vec{s}_2 & \cdots & \vec{s}_p \end{bmatrix} = \begin{bmatrix} 1 & -M & (-M)^2 & \cdots & (-M)^p \\ \vdots & \vdots & \vdots & \iddots & \vdots \\ 1 & -2 & 4 & \cdots & (-2)^p \\ 1 & -1 & 1 & \cdots & (-1)^p \\ 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 1 & \cdots & 1 \\ 1 & 2 & 4 & \cdots & 2^p \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & M & M^2 & \cdots & M^p \end{bmatrix}, \qquad \vec{s}_i(m) = m^i$$

Daraus folgt:
$$\boxed{\frac{\partial \mathcal{J}}{\partial \vec{c}} = -2\,S^{\mathsf{T}}\vec{x} + 2\,S^{\mathsf{T}}S\,\vec{c} = \mathbf{0}} \quad \Rightarrow \quad \boxed{\vec{c} = (S^{\mathsf{T}}S)^{-1}S^{\mathsf{T}}\vec{x} = G^{\mathsf{T}}\vec{x}}$$

Die gefilterten Samples $\hat{x}_m$ (Polynomwerte) können also folgendermassen berechnet werden:

$$\Rightarrow \quad \boxed{\vec{\hat{x}} = S\,\vec{c} = S\,G^{\mathsf{T}}\vec{x} = S\,(S^{\mathsf{T}}S)^{-1}S^{\mathsf{T}}\vec{x} = B\,\vec{x}}$$

mit
$$B = B^{\mathsf{T}} = S\,G^{\mathsf{T}} = S\,(S^{\mathsf{T}}S)^{-1}S^{\mathsf{T}} = S\,F^{-1}S^{\mathsf{T}} = \begin{bmatrix} \vec{b}_{-M} & \cdots & \vec{b}_{-1} & \vec{b}_0 & \vec{b}_1 & \cdots & \vec{b}_M \end{bmatrix}$$

und
$$G = S\,(S^{\mathsf{T}}S)^{-1} = S\,F^{-1} = \begin{bmatrix} \vec{g}_0 & \vec{g}_1 & \vec{g}_2 & \cdots & \vec{g}_p \end{bmatrix} \quad \text{und} \quad \boxed{F = S^{\mathsf{T}}S}$$

9.4.2 Matrizen für $M = 2$ und $p = 2$

Matrix S:

$$S = \begin{bmatrix} 1 & -2 & 4 \\ 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \end{bmatrix}$$

Matrix F:

$$F = S^{\mathsf{T}} S \begin{bmatrix} 5 & 0 & 10 \\ 0 & 10 & 0 \\ 10 & 0 & 34 \end{bmatrix}$$

Inverse Matrix F:

$$F^{-1} = \frac{1}{35} \begin{bmatrix} 17 & 0 & -5 \\ 0 & 3.5 & 0 \\ -5 & 0 & 2.5 \end{bmatrix}$$

Matrix G:

$$G = S\,F^{-1} = \frac{1}{35} \begin{bmatrix} -3 & -7 & 5 \\ 12 & -3.5 & -2.5 \\ 17 & 0 & -5 \\ 12 & 3.5 & -2.5 \\ -3 & 7 & 5 \end{bmatrix}$$

$$= \begin{bmatrix} \vec{g}_0 & \vec{g}_1 & \vec{g}_2 \end{bmatrix}$$

Matrix B:

$$B = S\,G^{\mathsf{T}} = \frac{1}{35} \begin{bmatrix} 31 & 9 & -3 & -5 & 3 \\ 9 & 13 & 12 & 6 & -5 \\ -3 & 12 & 17 & 12 & -3 \\ -5 & 6 & 12 & 13 & 9 \\ 3 & -5 & -3 & 9 & 31 \end{bmatrix}$$

$$= \begin{bmatrix} \vec{b}_{-2} & \vec{b}_{-1} & \vec{b}_0 & \vec{b}_1 & \vec{b}_2 \end{bmatrix}$$

9.4.3 FIR-Smoothing-Filter

- Die gefilterten Samples $\widehat{x}_m$ (Polynomwerte) können nun folgendermassen berechnet werden:

$$\boxed{\vec{\widehat{x}} = B\,\vec{x} = B^{\mathsf{T}}\vec{x} = \begin{bmatrix} \vec{b}_{-M}^{\mathsf{T}}\vec{x} \\ \vdots \\ \vec{b}_{-1}^{\mathsf{T}}\vec{x} \\ \vec{b}_0^{\mathsf{T}}\vec{x} \\ \vec{b}_1^{\mathsf{T}}\vec{x} \\ \vdots \\ \vec{b}_M^{\mathsf{T}}\vec{x} \end{bmatrix}}$$

Oder wenn nur ein bestimmtes Sample, meistens das mittlere ($m = 0$), von Interesse ist:

$$\boxed{\widehat{x}_m = b_m^{\mathsf{T}}\,\vec{x}} \quad m = -M, ..., -2, -1, 0, 1, 2, ..., M$$

- Die Polynom-Koeffizienten können ebenfalls folgendermassen berechnet werden:

$$\boxed{\vec{c} = G^{\mathsf{T}}\vec{x} = \begin{bmatrix} \vec{g}_0{}^{\mathsf{T}}\vec{x} \\ \vec{g}_1{}^{\mathsf{T}}\vec{x} \\ \vec{g}_2{}^{\mathsf{T}}\vec{x} \\ \vdots \\ \vec{g}_p{}^{\mathsf{T}}\vec{x} \end{bmatrix}} \qquad \boxed{c_i = g_i{}^{\mathsf{T}}\vec{x}} \quad i = 0, 1, 2, ..., p$$

- Das Savitzky-Golay-Smoothing-Filter (für das mittlere Sample) kann also folgendermassen als FIR-Filter implementiert werden:

$$\boxed{y_0 = \widehat{x}_0 = c_0 = b_0{}^{\mathsf{T}}\vec{x} = g_0{}^{\mathsf{T}}\vec{x} = \sum_{m=-M}^{M} b_0(m)\,x_m}$$

Oder konkret für $M = 2$ und $p = 2$:

$$\boxed{y(n) = \frac{-3\,x(n-2) + 12\,x(n-1) + 17\,x(n) + 12\,x(n+1) - 3\,x(n+2)}{35}}$$

- Das resultierende FIR Filter hat einen DC-Gain von 1 und keine Phase. Durch eine Verzögerung von M Samples wird es kausal und linearphasig.

- Das FIR-Filter hat eine Noise-Reduction-Ratio von:

$$\boxed{NRR = \vec{b}_0^{\mathsf{T}}\vec{b}_0 = \sum_{m=-M}^{M} b_0(m)^2 = b_0(0) = \frac{17}{35} = 0.49}$$

9.4.4 Ableiten und integrieren mit Savitzky-Golay-Smoothing-Filter

Das das Savitzky-Golay-Smoothing-Filter ein Polynom in die gemessenen Samples hineinpasst ist es sehr gut geeignet, um das Signal mittels dieses Polynoms abzuleiten oder zu integrieren

Ableiten mit Savitzky-Golay-Smoothing-Filter:

Polynom:

$$\boxed{y_m = \widehat{x}_m = c_0 + c_1\, m + c_2\, m^2 + ... + c_p\, m^p}$$

erste Ableitung:

$$\dot{y}_m = c_1 + 2\, c_2\, m + 3\, c_3\, m^2 + ... + p\, c_p\, m^{p-1} \quad \Rightarrow \quad m = 0 \quad \Rightarrow \quad \boxed{\dot{y}_0 = c_1 = {g_1}^\mathsf{T} \vec{x}}$$

zweite Ableitung:

$$\ddot{y}_m = 2\, c_2 + 6\, c_3\, m + ... + p\,(p-1)\, c_p\, m^{p-2} \quad \Rightarrow \quad m = 0 \quad \Rightarrow \quad \boxed{\ddot{y}_0 = 2\, c_2 = 2\, {g_2}^\mathsf{T} \vec{x}}$$

i-te Ableitung:

$$y_m^{(i)} = i!\, c_i + (i-1)!\, c_{i-1}\, m + ... + \frac{p!}{(p-i-1)!}\, c_p\, m^{p-i} \quad \Rightarrow \quad m = 0 \quad \Rightarrow \quad \boxed{y_0^{(i)} = i!\, c_i = i!\, {g_i}^\mathsf{T} \vec{x}}$$

Ableitung für $M = 2$ und $p = 2$

$$\boxed{\dot{y}_n = \frac{1}{35}\big(-7\, x_{n-2} - 3.5\, x_{n-1} + 3.5\, x_{n+1} + 7\, x_{n+2}\big)} \quad \boxed{\ddot{y}_n = \frac{2}{35}\big(5\, x_{n-2} - 2.5\, x_{n-1} - 5\, x_n - 2.5\, x_{n+1} + 5\, x_{n+2}\big)}$$

Integrieren mit Savitzky-Golay-Smoothing-Filter:

Polynom:

$$\boxed{y_m = \widehat{x}_m = c_0 + c_1\, m + c_2\, m^2 + ... + c_p\, m^p}$$

ein Integral-Intervall:

$$\begin{aligned}
i[n] &= \int_{-1/2}^{1/2} \widehat{x}_m\, dm = \int_{-1/2}^{1/2} c_0 + c_1\, m + c_2\, m^2 + ... + c_p\, m^p\, dm \\
&= \left[c_o\, m + \frac{c_1}{2} m^2 + \frac{c_2}{3} m^3 + \cdots + \frac{c_p}{p+1} m^{p+1} \right]_{-1/2}^{1/2} \\
&= \begin{cases} \begin{bmatrix} 1 & 0 & \frac{1}{3\cdot 2^2} & 0 & \frac{1}{5\cdot 2^4} & \cdots & 0 & \frac{1}{(p+1)\cdot 2^p} \end{bmatrix} \cdot \vec{c}, & p \text{ ungerade} \\ \begin{bmatrix} 1 & 0 & \frac{1}{3\cdot 2^2} & 0 & \frac{1}{5\cdot 2^4} & \cdots & \frac{1}{p\cdot 2^{p-1}} & 0 \end{bmatrix} \cdot \vec{c}, & p \text{ gerade} \end{cases} \\
&= \begin{cases} \begin{bmatrix} 1 & 0 & \frac{1}{3\cdot 2^2} & 0 & \frac{1}{5\cdot 2^4} & \cdots & 0 & \frac{1}{(p+1)\cdot 2^p} \end{bmatrix} \cdot G^\mathsf{T} \vec{x}, & p \text{ ungerade} \\ \begin{bmatrix} 1 & 0 & \frac{1}{3\cdot 2^2} & 0 & \frac{1}{5\cdot 2^4} & \cdots & \frac{1}{p\cdot 2^{p-1}} & 0 \end{bmatrix} \cdot G^\mathsf{T} \vec{x}, & p \text{ gerade} \end{cases}
\end{aligned}$$

alle Integral-Intervalle aufsummiert:

$$\boxed{I[n] = \sum_{k=0}^{n} T \cdot i[k] = \sum_{k=0}^{n-1} T \cdot i[k] + T \cdot i[n] = I[n-1] + T \cdot i[n]}$$

10 DFT/FFT Algorithmus S.464

Die Diskrete-Fourier-Transformation (DFT) und die Fast-Fourier-Transformation (FFT) ist für Folgendes von grosser Bedeutung:

- Nummerische Berechnung des Frequenzspektrums eines Signals.
- Effiziente Berechnung der Faltung mittels FFT.
- Effiziente Speicherung und Übertragung von von Waveform-Signalen (Bilder und Sprache/Musik).

10.1 Frequenzauflösung und Windowing

- Unendliches Signal $x(n)$ mittels eines Fensters auf eine endliche Länge reduzieren, um das Spektrum berechnen zu können.

$$\boxed{x_L(n) = x(n) \cdot w(n)} \qquad \boxed{T_L = L \cdot T = \frac{L}{f_s}}$$

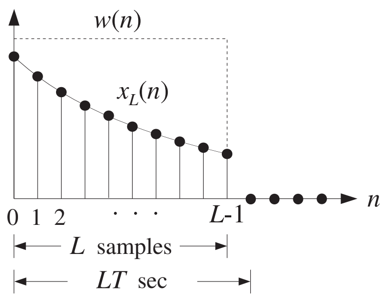
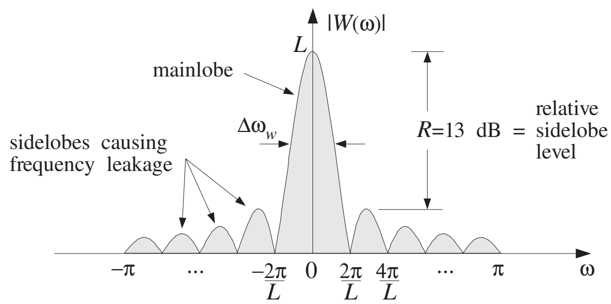
$w(n)$: Fenster
$x(n)$: unendliches Signal
$x_L(n)$: endliches Signal

- Die DTFT des windowed Signals kommt immer näher and die des orginalen unendlichen Signals, je länger das Fenster ist, bzw. je mehr Samples L verwendet werden.

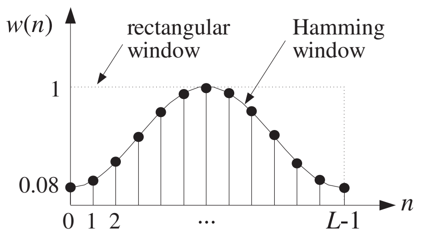
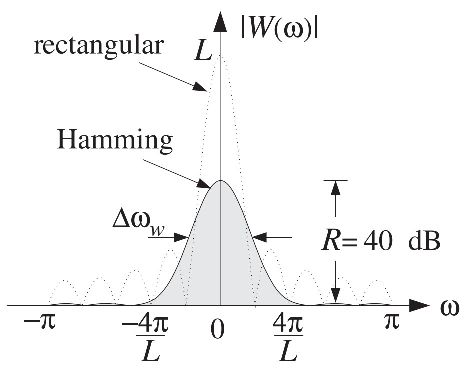
$$\boxed{X(\omega) = \sum_{n=-\infty}^{\infty} x(n)\,\mathrm{e}^{-j\omega n} \quad \approx \quad X_L(\omega) = \sum_{n=-\infty}^{\infty} x_L(n)\,\mathrm{e}^{-j\omega n} = \sum_{n=0}^{L-1} x(n)\cdot w(n)\,\mathrm{e}^{-j\omega n}}$$

- Durch das Windowing wird die **Frequenzauflösung reduziert**.
- Das Windowing erzeugt **hohe Frequenzanteile im Spektrum** (frequency leakage)

10.1.1 Rechteck-Fenster

Zeitbereich		**Frequenzbereich**
		
$w(n) = \begin{cases} 1, & 0 \le n \le L-1 \\ 0, & \text{sonst} \end{cases}$	$\circ\!\!-\!\!\bullet$	$W(\omega) = \dfrac{1-\mathrm{e}^{-jL\omega}}{1-\mathrm{e}^{-j\omega}} = \dfrac{\sin(\omega L/2)}{\sin(\omega/2)} \cdot \mathrm{e}^{-j\omega(L-1)/2}$

10.1.2 Hamming-Fenster

Zeitbereich	**Frequenzbereich**
	

$$\boxed{w(n) = \begin{cases} 0.54 - 0.46\cos\left(\dfrac{2\pi n}{L-1}\right), & 0 \le n \le L-1 \\ 0, & \text{sonst} \end{cases}}$$

10.1.3 Eigenschaften der Fenster-Spektren

Die Wahl des Fensters ist ein **Tradeoff** zwischen der Mainlobe-Width (Frequenzauflösung) und der Sidelobe-Suppression.

	Rechteck-Fenster	**Hamming-Fenster**
Mainlobe	Je grösser die Anzahl Samples L, desto höher und schmaler wird die Mainlobe. (Die Sidelobes wachsen proportional mit.)	
Mainlobe-Width	$\boxed{\Delta\omega_W = c \cdot \frac{2\pi}{L} = c \cdot \frac{2\pi \Delta f_W}{f_s}}$ $\boxed{\Delta f_W = c \cdot \frac{f_s}{L} = c \cdot \frac{1}{LT} = c \cdot \frac{1}{T_L}}$ $c \geq 1$	
	$c = 1 \Rightarrow$ **Beste Frequenzauflösung**	$c = 2$
Sidelobes	Sind sehr gross, weil das Fenster im Zeitbereich scharf abschneidet.	Sind eher klein, weil das Fenster im Zeitbereich langsam hinein und hinaus fährt.
Sidelobe-Suppression	$\boxed{R = \left\|\frac{W(\omega)}{W(0)}\right\|_{\omega=3\pi/L} \simeq \frac{2}{3\pi} \mathrel{\hat{=}} -13.46\text{dB}}$	$\boxed{R = -40\text{dB}}$ $\Rightarrow$ **gute Sidelobe-Suppression**

10.1.4 Frequenzauflösung

Die Multiplikation des Signals $x(n)$ mit dem Fenster $w(n)$ entspricht im Frequenzbereich einer Faltung, wodurch das **Spektrum des Signals** $X(\omega)$ **verschmiert wird**.

$$\boxed{x_L(n) = x(n) \cdot w(n) \quad \circ\!\!-\!\!\bullet \quad X_L(\omega) = X(\omega) * W(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega') \cdot W(\omega - \omega')\, d\omega'}$$

Beispiel:

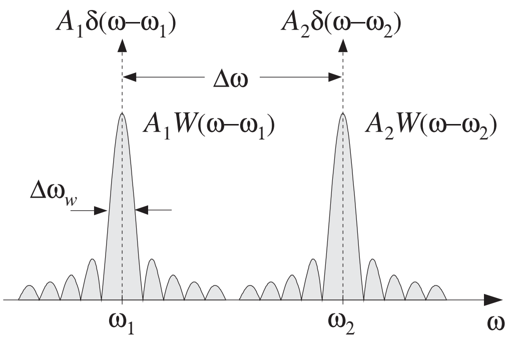
$$\begin{aligned}
x(n) &= A_1\, e^{j\omega_1 n} + A_2\, e^{j\omega_2 n} & \circ\!\!-\!\!\bullet \quad X(\omega) &= \underline{\underline{2\pi\, A_1\, \delta(\omega-\omega_1) + 2\pi\, A_2\, \delta(\omega-\omega_2)}} \\
x_L(n) &= A_1\, e^{j\omega_1 n} \cdot w(n) + A_2\, e^{j\omega_2 n} \cdot w(n) & \circ\!\!-\!\!\bullet \quad X_L(\omega) &= \int_{-\pi}^{\pi} A_1\, \delta(\omega'-\omega_1) \cdot W(\omega-\omega')\, d\omega' \\
& & &+ \int_{-\pi}^{\pi} A_2\, \delta(\omega'-\omega_2) \cdot W(\omega-\omega')\, d\omega' \\
& & &= \underline{\underline{A_1\, W(\omega-\omega_1) + A_2\, W(\omega-\omega_2)}}
\end{aligned}$$

$\Rightarrow$ **Die Mainlobe muss schmaler sein, als der kleinste noch auflösbare Frequenzunterschied (Auflösung).**

$$\boxed{\Delta\omega \geq \Delta\omega_W = c \cdot \frac{2\pi}{L} = c \cdot \frac{2\pi \Delta f_W}{f_s}}$$

$$\boxed{\Delta f \geq \Delta f_W = c \cdot \frac{f_s}{L} = c \cdot \frac{1}{LT} = c \cdot \frac{1}{T_L}}$$

$$\boxed{L \geq c \cdot \frac{f_s}{\Delta f} = c \cdot \frac{2\pi}{\Delta\omega}}$$



10.2 Discrete-Fourier-Transformation (DFT)

- Die Discrete-Time-Fourier-Transformation (DTFT) eines L-Sample langen Signals ist definiert als:

$$\boxed{X(\omega) = \sum_{n=0}^{L-1} x(n)\,\mathrm{e}^{-j\omega n}}$$

- Die Discrete-Fourier-Transformation (DFT) ist die DTFT, welche nur an bestimmten (diskreten) Frequenzpunkten ausgewertet wird.

10.2.1 DFT für einzelne Frequenzen

- Berechnung der DFT für einzelne bestimmte Frequenzen ω_i mittels der Übertragungsfunktion $H(z)$

$$\boxed{X(\omega_i) = \sum_{n=0}^{L-1} x(n)\,\mathrm{e}^{-j\omega_i n} = \sum_{n=0}^{L-1} x(n)\,z^{-n}\big|_{z=\mathrm{e}^{j\omega_i}} = X(z)|_{z=\mathrm{e}^{j\omega_i}}}$$

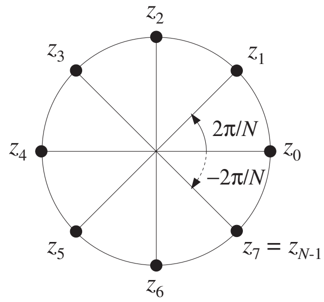
$$X(z) = x_0 + z^{-1}\cdot(x_1 + z^{-1}\cdot(x_2 + z^{-1}x_3))$$

10.2.2 DFT für N-Frequenzen

- N-Point DFT eines L-Sample langen Signals mit N gleichmässig verteilten Frequenzpunkten.

$$\boxed{X(\omega_k) = \sum_{n=0}^{L-1} x(n)\,\mathrm{e}^{-j\omega_k n}} \qquad \text{mit} \quad \omega_k = \frac{2\pi k}{N} \qquad \text{oder} \quad f_k = \frac{k\,f_s}{N} \qquad k = 0, 1, ..., N-1$$

- Die N-Werte $X(\omega_k)$ können auch als Werte der z-Transformation $X(z)$ auf dem Einheitskreis betrachtet werden



$$\boxed{X(\omega_k) = X(z_k) = \sum_{n=0}^{L-1} x(n)\,z_k^{-n}}$$

$$z_k = \mathrm{e}^{j\omega_k} = \mathrm{e}^{2\pi jk/N} \qquad k = 0, 1, ..., N-1$$

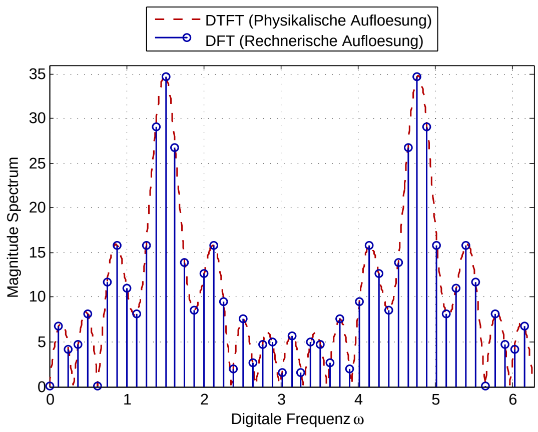
- Sie sind auch die N-ten Einheitswurzeln $\boxed{z^n = 1}$

10.2.3 Anzahl Frequenzpunkte versus Anzahl Samples

- $\boldsymbol{L}$: Anzahl Samples des Zeit-Signals $x(n)$
- $\boldsymbol{N}$: Anzahl Samples des Frequenzspektrums $X(\omega)$
- $\boldsymbol{L = N}$: Häufiger Spezialfall $\Rightarrow$ alles okej
- $\boldsymbol{L < N}$: Zeitsequenz ist zu kurz $\Rightarrow$ am Ende des Signals Nullen anhängen (zero padding)
- $\boldsymbol{L > N}$: Zeitsequenz ist zu lang $\Rightarrow$ überzählige Samples am Anfang addieren (modulo-N-wrapping siehe Kapitel 10.2.6)

10.2.4 Physikalische versus rechnerische Auflösung

Physikalische Auflösung	Rechnerische Auflösung
Frequenz-Differenz zwischen zwei noch unterscheidbaren Schwingungen aufgrund der Mainlobe-Width.	Frequenz-Differenz zwischen zwei Samples des abgetasteten Spektrums
Beeinflussbar durch Anzahl Samples L des Zeit-Signals	Beeinflussbar durch Anzahl Samples N des Spektrums
$\boxed{\Delta\omega_W = \frac{2\pi}{L}}$ $\boxed{\Delta f_W = \frac{f_s}{L}}$	$\boxed{\Delta\omega_{\mathrm{bin}} = \frac{2\pi}{N}}$ $\boxed{\Delta f_{\mathrm{bin}} = \frac{f_s}{N}}$



Mögliche Probleme des Frequenzspektrums

- Die Samples der DFT fallen nicht zwangsweise auf die Peaks der DTFT. Denn die Samples stimmen nur dann exakt, wenn k (DFT Index) eine ganze Zahl ist.
  $\Rightarrow$ Rechnerische Auflösung erhöhen (Anzahl Samples N).

  $$\boxed{k = N\frac{f}{f_s}}$$

- Der Frequenz Bias Fehler entsteht, wenn eine Mainlobe von fremden Sidelobes gestört wird (Interferenz) und sich dadurch der Peak der Mainlobe leicht verschiebt.
  $\Rightarrow$ Physikalische Auflösung erhöhen (Anzahl Samples L).

10.2.5 DFT als Matrix-Multiplikation

- Die Diskrete-Fourier-Transformation kann auch als Matrix-Multiplikation geschrieben werden:

$$\boxed{\boldsymbol{X} = \mathrm{DFT}(\boldsymbol{x}) = \boldsymbol{A}\cdot\boldsymbol{x}}$$

Matrixelement: $A_{kn} = \mathrm{e}^{-j\omega_k n} = \mathrm{e}^{-j2\pi kn/N} = W_N^{kn}$

Twiddle-Faktor: $W_N = \mathrm{e}^{-j2\pi/N}$

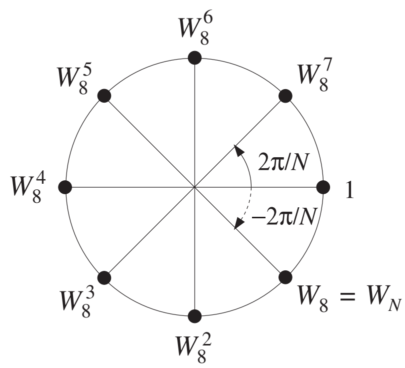
mit $$\boldsymbol{X} = \begin{bmatrix} X(\omega_0) \\ X(\omega_1) \\ \vdots \\ X(\omega_{N-1}) \end{bmatrix} = \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{bmatrix}$$ und $$\boldsymbol{x} = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(L-1) \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{L-1} \end{bmatrix}$$

und $$\boldsymbol{A} = \begin{bmatrix} W_N^{0\cdot 0} & W_N^{0\cdot 1} & \dots & W_N^{0\cdot(L-1)} \\ W_N^{1\cdot 0} & W_N^{1\cdot 1} & \dots & W_N^{1\cdot(L-1)} \\ \vdots & \vdots & \ddots & \vdots \\ W_N^{(N-1)\cdot 0} & W_N^{(N-1)\cdot 1} & \dots & W_N^{(N-1)\cdot(L-1)} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & W_N & \dots & W_N^{(L-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{(N-1)} & \dots & W_N^{(N-1)(L-1)} \end{bmatrix}$$

- Die Ermittlung der Matrix $\boldsymbol{A}$ bzw. der Twiddle-Faktoren W_N^{kn} geht am einfachsten über den Einheitskreis der z-Ebene.

$$\boxed{W_N^k = \mathrm{e}^{-j2\pi k/N} = z_{-k}}$$

1. Matrix Zeilenweise auffüllen
2. Rechts bei 1 beginnen und im Uhrzeigersinn die Werte des Einheitskreises eintragen.
3. Für die nächste Zeile wiederum rechts bei 1 beginnen und im Uhrzeigersinn die Werte des Einheitskreises eintragen jedoch die Sprunglänge um eins erhöhen.
4. Schritt 3. so oft wiederholen bis die Matrix voll ist.



- Beispiel für eine DFT mit $N, L = 2$ und 4

$$\begin{bmatrix} X_0 \\ X_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} x_0 + x_1 \\ x_0 - x_1 \end{bmatrix}$$

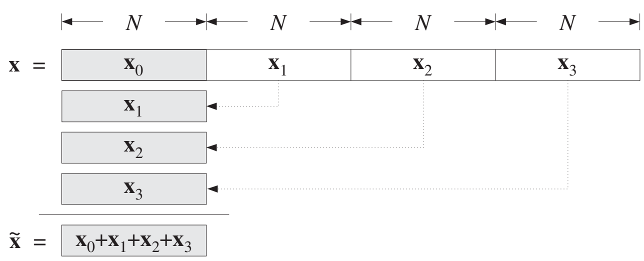
$$\begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

10.2.6 Modulo-N Reduktion

- Eine Modulo-N Reduktion des Eingangssignals (Eingangsvektor $\boldsymbol{x}$) ist notwendig wenn:

$$\boxed{\text{Anzahl Zeit-Samples } L > \text{Anzahl Frequenz-Samples } N}$$

- Beim wrapping-Prozess wird der Vektor $\boldsymbol{x}$ in $N-$dimensionale Subvektoren unterteilt, die dann zusammengezählt werden. Wenn L kein ganzzahliges Vielfaches von N ist, wird $\boldsymbol{x}$ mit Nullen aufgefüllt.



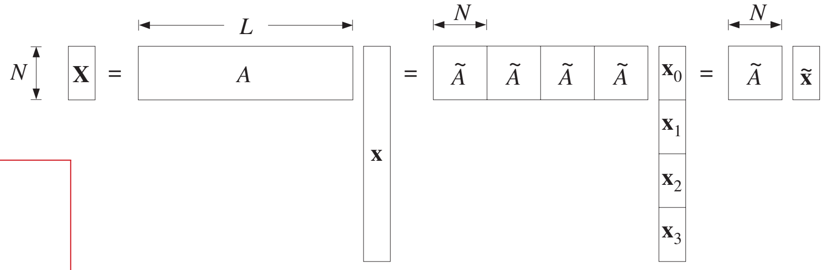
$$\widetilde{x}(n) = x_0(n) + x_1(n) + \ldots + x_{L/N-1}(n)$$
$$= x(n) + x(N+n) + \ldots + x((\tfrac{L}{N}-1)\cdot N + n)$$

$$\Rightarrow \boxed{\widetilde{x}(n) = \sum_{m=0}^{L/N-1} x(m\,N+n)} \quad n = 0, 1, ..., N-1$$

$$\Rightarrow \boxed{\widetilde{\boldsymbol{x}} = \boldsymbol{x}_0 + \boldsymbol{x}_1 + \ldots + \boldsymbol{x}_{(L/N)-1}}$$

- Die N-Point-DFT des originalen, L-langen Signals $x(n)$ und die N-Point-DFT des modulo-N-gewrappten Signals $\widetilde{x}(n)$ sind identisch.

$$\boxed{X(\omega_k) = \widetilde{X}(\omega_k) \quad \Rightarrow \quad X_k = \widetilde{X}_k}$$



Also gilt auch:

$$\boxed{\begin{aligned} \boldsymbol{X} \;&=\; \boldsymbol{A}\,\boldsymbol{x} = \begin{bmatrix} \widetilde{\boldsymbol{A}} & \widetilde{\boldsymbol{A}} & \cdots \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{x}_1 \\ \vdots \end{bmatrix} \\ &=\; \widetilde{\boldsymbol{A}}\,(\boldsymbol{x}_0 + \boldsymbol{x}_1 + \ldots) \;=\; \widetilde{\boldsymbol{A}}\,\widetilde{\boldsymbol{x}} \;=\; \widetilde{\boldsymbol{X}} \end{aligned}}$$

Dies ist bewiesen, wenn gezeigt werden kann, dass: $\boxed{\boldsymbol{A} = \begin{bmatrix} \widetilde{\boldsymbol{A}} & \widetilde{\boldsymbol{A}} & \widetilde{\boldsymbol{A}} & \cdots \end{bmatrix} \quad \text{bzw.} \quad A_{kn} = A_{k(mN+n)}}$

$$\underline{\underline{A_{k(mN+n)}}} = W_N^{k(mN+n)} = W_N^{kmN} \cdot W_N^{kn} = (\mathrm{e}^{-j2\pi/N})^{kmN} \cdot W_N^{kn} = \underbrace{\mathrm{e}^{-j2\pi km}}_{1} \cdot W_N^{kn} = W_N^{kn} = \underline{\underline{A_{kn}}}$$

- Verschiedene Signale der Länge L mit der selben modulo-N-Reduktion haben die gleiche N-Point-DFT.

$$\boldsymbol{x} \neq \boldsymbol{y} \,\cap\, \widetilde{\boldsymbol{x}} = \widetilde{\boldsymbol{y}} \quad \Rightarrow \quad X_k = Y_k$$

Diese Signale haben in der z-Domäne folgende Beziehung.

$$F(z_k) = X(z_k) - Y(z_k) = X_k - Y_k = 0 \qquad k = 0, 1, ..., N-1$$

Die N-Nullstellen auf dem Einheitskreis sind auch Nullstellen des Differenzpolynoms $F(z)$, womit gilt:

$$1 - z^{-N} = \prod_{k=0}^{N-1} (1 - z_k\, z^{-1}) \quad \Rightarrow \quad F(z) = X(z) - Y(z) = (1 - z^{-N})\, Q(z)$$

$$\Rightarrow \quad \boxed{X(z) = Y(z) + (1 - z^{-N})\, Q(z)} \qquad \text{Ordnung von } Q(z) \text{ ist } (L-1) - N$$

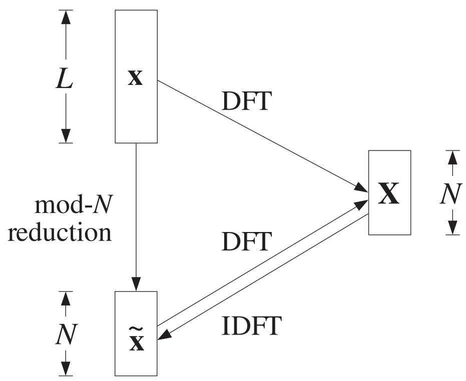
$\Rightarrow$ **Zwei Sequenzen $x(n)$ und $y(n)$, die folgende Gleichung erfüllen haben, die selbe N-Point-DFT!**

$$\Rightarrow \quad \boxed{x(n) = y(n) + q(n) - q(n-N)} \qquad n = 0, 1, ..., L-1$$

10.2.7 Inverse-Discrete-Fourier-Transformation (IDFT)

- Die Discrete-Fourier-Transformation kann geschrieben werden als:
  $\boldsymbol{X} = \boldsymbol{A}\,\boldsymbol{x} = \widetilde{\boldsymbol{A}}\,\widetilde{\boldsymbol{x}}$
- Damit ist die Inverse-Discrete-Fourier-Transformation folgendes:

$$\boxed{\widetilde{\boldsymbol{x}} = \mathrm{IDFT}(\boldsymbol{X}) = \widetilde{\boldsymbol{A}}^{-1}\,\boldsymbol{X}}$$



Achtung:
Damit die Matrix $\boldsymbol{A}$ invertierbar ist, muss sie symetrisch sein, was bei der Matrix $\widetilde{\boldsymbol{A}}$ gegeben ist. Somit kann die IDFT nur das modulo-N-gewrappte Signal $\widetilde{x}$ bzw. $\widetilde{x}(n)$ rekonstruieren!

Dabei gilt jedoch: $\boxed{\begin{array}{ll} \widetilde{\boldsymbol{x}} = \boldsymbol{x} & \text{wenn } N \geq L \\ \widetilde{\boldsymbol{x}} \neq \boldsymbol{x} & \text{wenn } N < L \\ & \Rightarrow \text{ Aliasing im Frequenzbereich!} \end{array}}$

- Aufgrund der Beziehung $\boxed{\dfrac{1}{N}\,\widetilde{\boldsymbol{A}}\,\widetilde{\boldsymbol{A}}^* = \boldsymbol{I}_N \quad \Rightarrow \quad \widetilde{\boldsymbol{A}}^{-1} = \dfrac{1}{N}\,\widetilde{\boldsymbol{A}}^*}$

  kann die IDFT vereinfacht werden zu $\boxed{\widetilde{\boldsymbol{x}} = \mathrm{IDFT}(\boldsymbol{X}) = \dfrac{1}{N}\,\widetilde{\boldsymbol{A}}^*\,\boldsymbol{X}}$

  oder mit der DFT geschrieben werden als $\boxed{\widetilde{\boldsymbol{x}} = \mathrm{IDFT}(\boldsymbol{X}) = \dfrac{1}{N}\,\left[\mathrm{DFT}(\boldsymbol{X}^*)\right]^*}$

10.2.8 DTFT/IDTFT versus DFT/IDFT

	DTFT/IDTFT	**DFT/IDFT**
Fourier-Transformation:	• L Zeit-Samples • ∞ Frequenz-Samples • $\boxed{X(\omega) = \sum_{n=0}^{L-1} x(n)\, \mathrm{e}^{-j\omega n}}$	• L Zeit-Samples • N Frequenz-Samples • $\boxed{X(\omega_k) = \sum_{n=0}^{L-1} x(n)\, \mathrm{e}^{-j\omega_k n}}$
Inverse-Fourier-Transformation:	• Rekonstruiert das gesamte originale Signal $x(n)$ • $\boxed{x(n) = \frac{1}{2\pi} \int_0^{2\pi} X(\omega)\, \mathrm{e}^{j\omega n}\, d\omega}$	• Rekonstruiert nur das modulo-N-gewrappte Signal $\widetilde{x}(n)$ • $\boxed{\widetilde{x}(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(\omega_k)\, \mathrm{e}^{j\omega_k n}}$
IDFT als Approximation der IDTFT:	$x(n) \;=\; \frac{1}{2\pi} \int_0^{2\pi} X(\omega)\, \mathrm{e}^{j\omega n}\, d\omega \;\simeq\; \sum_{k=0}^{N-1} X(\omega_k)\, \mathrm{e}^{j\omega_k n} \frac{\Delta\omega_{\text{bin}}}{2\pi} \;=\; \frac{1}{N} \sum_{k=0}^{N-1} X(\omega_k)\, \mathrm{e}^{j\omega_k n}$	

10.3 Fast-Fourier-Transformation (FFT)

- Die Fast-Fourier-Transformation (FFT) ist eine schnelle Implementation der Discrete-Fourier-Transformation (DFT), welche N^2 komplexe Multiplikation benötigen würde. **Somit sind die Resultate der FFT und der DFT identisch.**
- Die FFT berechnet die DFT mit der "Teile-und-Hersche" Strategie, daher ist es von Vorteil, wenn die Anzahl Frequenz-Samples eine Zweierpotenz ist.

$$\boxed{N = 2^B \qquad \Rightarrow \qquad B = \log_2(N)}$$

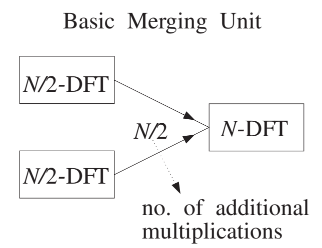
- Bei der FFT muss die Anzahl Zeit-Samples L und die Anzahl Frequenz-Samples N gleich gross sein. Ist dies nicht der Fall, so muss das Zeitsignal vorverarbeitet werden (Zero-Padding oder modulo-N-wrapping).

$$\boxed{N = L}$$

10.3.1 Teile-und-Hersche Strategie

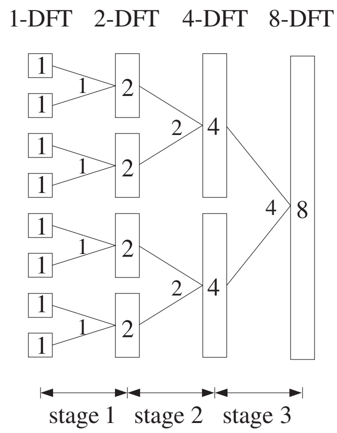
- Das Problem der Berechnung einer N-Point-DFT wird reduziert auf die Berechnung von zwei $N/2$-Point-DFTs.

$$\boxed{N \text{ - Point - DFT} = 2 \cdot \frac{N}{2} \text{ - Point - DFT} + \frac{N}{2} \text{ komplexe Multiplikationen}}$$



- Die $N/2$-Point-DFTs wiederum in $N/4$-Point-DFTs aufteilen, usw.
- Wenn die N-Point-DFT über m Stufen aus $N/2^m$-Point-DFTs berechnet wird, sind M komplexe Multiplikation notwendig.

$$\boxed{M \;=\; \frac{N^2}{2^m} + \frac{N}{2} m \;=\; \underbrace{2^m \left(\frac{N}{2^m}\right)^2}_{\text{DFT}} + \underbrace{2^{m-1} \frac{N}{2^m} + \ldots + 4\frac{N}{8} + 2\frac{N}{4} + \frac{N}{2}}_{\text{Stufen-Multiplikationen}}}$$



- Wenn mit 1-Point-DFTs gestartet wird sind $m = B = \log_2(N)$ Stufen notwendig und die Berechnung der 1-Point-DFTs entfällt, weil diese gerade sich selbst entsprechen. Damit entfällt der erste Term und es müssen nur noch die Stufen-Multiplikationen berechnet werden.

$$\boxed{M \;=\; \frac{N}{2} m \;=\; \frac{1}{2} N B \;=\; \frac{1}{2} N \log_2(N)}$$

10.3.2 N-Point-DFT aus zwei $N/2$-Point-DFTs berechnen

- Aufsplittung der DFT $X(k)$ in zwei Terme $G(k)$ und $H(k)$ welche jeweils die geraden bzw. die ungeraden Samples der Zeit-Sequenz berücksichtigen.

$$\begin{aligned}
X(k) &= \sum_{n=0}^{N-1} x(n)\,\mathrm{e}^{-j\omega_k n} = \sum_{n=0}^{N-1} W_N^{kn}\,x(n) \\
&= \sum_{n=0}^{N/2-1} W_N^{k(2n)}\,x(2n) + \sum_{n=0}^{N/2-1} W_N^{k(2n+1)}\,x(2n+1) && \Big| \ \boxed{\begin{aligned} g(n) &= x(2n) \\ h(n) &= x(2n+1) \end{aligned}} \quad n = 0, 1, ..., \tfrac{N}{2}-1 \\
&= \sum_{n=0}^{N/2-1} W_N^{k(2n)}\,g(n) + \sum_{n=0}^{N/2-1} W_N^{k(2n+1)}\,h(n) && \Big| \ W_N^2 = W_{N/2} \\
&= \sum_{n=0}^{N/2-1} W_{N/2}^{kn}\,g(n) + \sum_{n=0}^{N/2-1} W_N^k\,W_{N/2}^{kn}\,h(n) \\
&= \underline{\underline{G(k) + W_N^k\,H(k)}}
\end{aligned}$$

$$\Rightarrow \quad \boxed{X(k) = G(k) + W_N^k\,H(k)} \quad \text{mit} \quad \boxed{\begin{aligned} G(k) &= \sum_{n=0}^{N/2-1} W_{N/2}^{kn}\,g(n) \\ H(k) &= \sum_{n=0}^{N/2-1} W_{N/2}^{kn}\,h(n) \end{aligned}}$$

N – periodisch $\rightarrow \quad k = 0, 1, ..., N-1$

$N/2$ – periodisch $\rightarrow \quad k = 0, 1, ..., \frac{N}{2}-1$

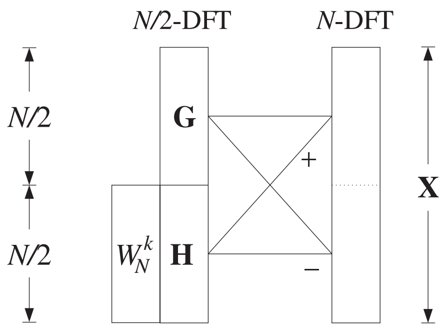
- Um die Anzahl Multiplikation mit W_N^k noch zu halbieren, wird die Eigenschaft ausgenutzt, dass $G(k)$ und $H(k)$ mit $N/2$ periodisch sind.

$$\begin{aligned}
X(k) &= \underline{G(k) + W_N^k\,H(k)} && \Big| \ k = 0, 1, ..., \tfrac{N}{2}-1 \\
X(k+\tfrac{N}{2}) &= G(k+\tfrac{N}{2}) + W_N^{k+N/2}\,H(k+\tfrac{N}{2}) && \Big| \ G(k+\tfrac{N}{2}) = G(k) \quad \text{und} \quad H(k+\tfrac{N}{2}) = H(k) \\
&= G(k) + W_N^k\,W_N^{N/2}\,H(k) && \Big| \ W_N^{N/2} = -1 \\
&= \underline{G(k) - W_N^k\,H(k)}
\end{aligned}$$

$$\Rightarrow \quad \boxed{\begin{aligned} X(k) &= G(k) + W_N^k\,H(k) \\ X(k+\tfrac{N}{2}) &= G(k) - W_N^k\,H(k) \end{aligned}} \qquad k = 0, 1, ..., \tfrac{N}{2}-1$$

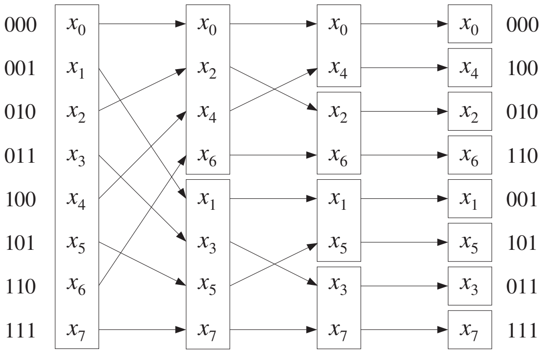
- Somit kann eine N-Point-DFT folgendermassen aus zwei $N/2$-Point-DFTs berechnet werden.

$$\boxed{X(k) = \begin{cases}
\begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N/2-1} \end{bmatrix} = \begin{bmatrix} G_0 \\ G_1 \\ \vdots \\ G_{N/2-1} \end{bmatrix} + \begin{bmatrix} H_0 \\ H_1 \\ \vdots \\ H_{N/2-1} \end{bmatrix} \times \begin{bmatrix} W_N^0 \\ W_N^1 \\ \vdots \\ W_N^{N/2-1} \end{bmatrix} \\
\begin{bmatrix} X_{N/2} \\ X_{N/2+1} \\ \vdots \\ X_{N-1} \end{bmatrix} = \begin{bmatrix} G_0 \\ G_1 \\ \vdots \\ G_{N/2-1} \end{bmatrix} + \begin{bmatrix} H_0 \\ H_1 \\ \vdots \\ H_{N/2-1} \end{bmatrix} \times \begin{bmatrix} W_N^0 \\ W_N^1 \\ \vdots \\ W_N^{N/2-1} \end{bmatrix}
\end{cases}}$$



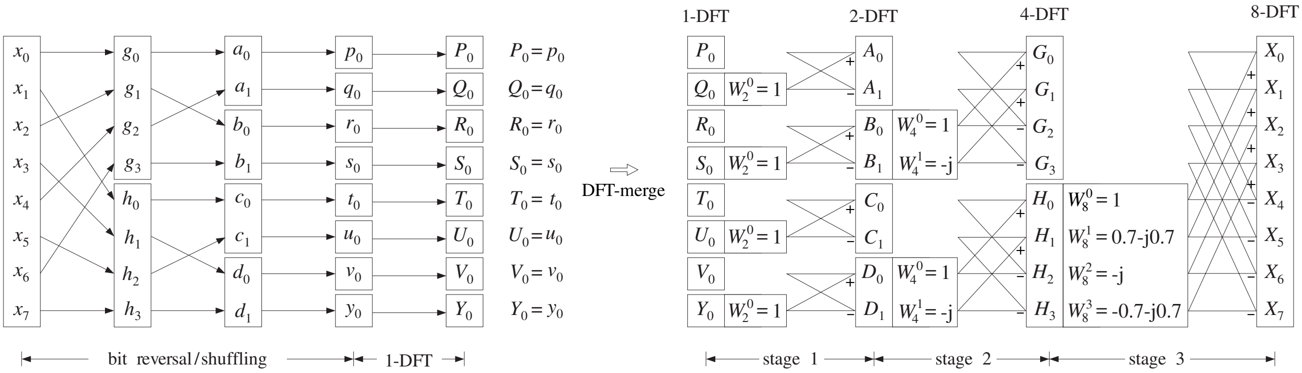
10.3.3 FFT-Algorithmus

- Der Eingang muss in gerade und ungerade Sequenzen unterteilt werden. Dies muss so oft wiederholt werden, bis nur noch ein Element übrig bleibt.



$\Rightarrow$ Das richtige Anordnen der Werte für die FFT wird auch als **Bit-Reversal** bezeichnet, weil der binäre Startindex rückwärts gelesen gerade dem binären Zielindex entspricht.
Bsp: 011 $\rightarrow$ 110

- Wenn alle Eingangssamples richtig geordnet sind, kann der FFT-Algorithmus darauf angewendet werden.



10.4 Schnelle Faltung mit FFT

- Eine fundamentale Beziehung zwischen Zeit- und Frequenzbereich ist

$$\boxed{\boldsymbol{y} = \boldsymbol{h} * \boldsymbol{x} \quad \Leftrightarrow \quad Y(\omega) = H(\omega) \cdot X(\omega)}$$

- Um also die Faltung möglichst effizient berechnen zu können, kann die FFT/IFFT verwendet werden.

$$\boxed{\tilde{\boldsymbol{y}} = \widetilde{\boldsymbol{h} * \boldsymbol{x}} = \text{IDFT}\big[\text{DFT}(\boldsymbol{h}) \cdot \text{DFT}(\boldsymbol{x})\big]}$$

- Da mit der FFT/IFFT jedoch nur die **Zirkulare Faltung** (modulo-N-gewrappte Version) berechnet werden kann, müssen die beiden Eingangssignale $x(n)$ und $h(n)$ mit Nullen aufgefüllt werden.

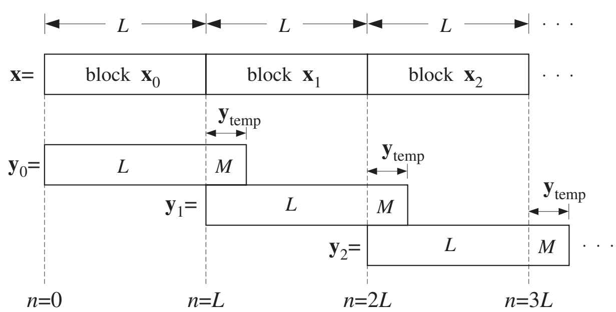
Signal:	ursprüngliche Länge	Zero-Padded Länge
$\boldsymbol{x}$	L	$L_y = L + M$
$\boldsymbol{h}$	$M + 1$ (Ordnung M)	$L_y = L + M$

$$\Rightarrow \quad \boxed{\tilde{\boldsymbol{y}} = \boldsymbol{y} \quad \text{wenn } N \geq L_y = L + M}$$

- Die rechnerischen Kosten dafür sind $\frac{3}{2} N \log_2(N) + N$ komplexe Multiplikationen

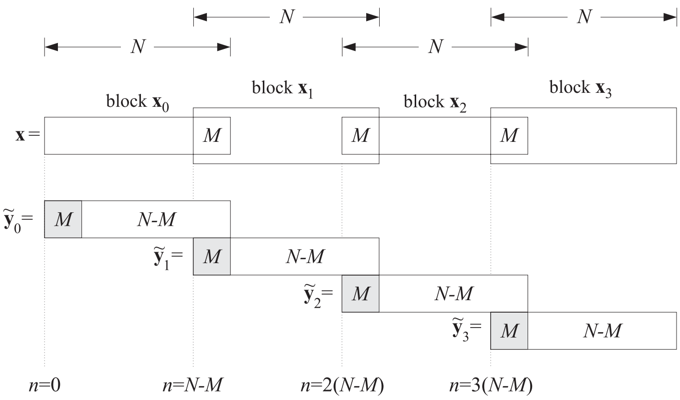
10.4.1 Overlap-Add-Faltung

- Unendlich langes Signal $x(n)$ muss mit einem Filter $h(n)$ gefaltet werden.
- Signal wird in L lange Eingangsblöcke zerlegt, die einzeln mittels FFT und IFFT gefaltet werden.
- Durch die Faltung entstehen $L + M$ lange Ausgangsblöcke, von welchen nur die ersten L Samples direkt verwendet werden und die letzten M Samples in einem Buffer $\boldsymbol{y}_{\text{temp}}$ zwischengespeichert werden.
- Die Samples des Zwischenspeichers $\boldsymbol{y}_{\text{temp}}$ werden beim nächsten Ausgangsblock vorne addiert.



10.4.2 Overlap-Save-Faltung

- Unendlich langes Signal $x(n)$ muss mit einem Filter $h(n)$ gefaltet werden.
- Signal wird in L lange Eingangsblöcke zerlegt, welche jeweils um M Samples überlappen.
- Die L langen Eingangsblöcke werden ohne Zero-Padding mittels FFT und IFFT gefaltet, wodurch, aufgrund der Circularen Faltung, L lange Ausgangsblöcke entstehen. Bei diesen Ausgangsblöcken sind jedoch die ersten M Samples falsch.
- Diese falschen Samples werden jedoch verworfen, da die richtigen Samples bereits vom vorhergehenden Ausgangsblock berechnet wurden.

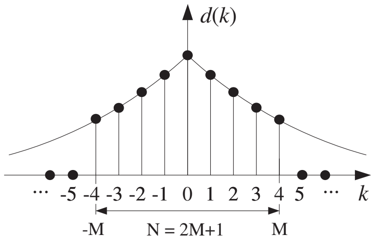


11 FIR Digital Filter Design S.532

Die Fenster-Methode ist eine sehr simple Methode um FIR-Filter zu designen.

- Der ideale Frequenzgang $D(\omega)$ wird in eine unendlich lange Impulsantwort $d(k)$ umgerechnet.

$$\boxed{D(\omega) = \sum_{k=-\infty}^{\infty} d(k)\, \mathrm{e}^{-j\omega k} \quad \Leftrightarrow \quad d(k) = \frac{1}{2\pi} \int_{-\pi}^{\pi} D(\omega)\, \mathrm{e}^{j\omega k}\, d\omega}$$



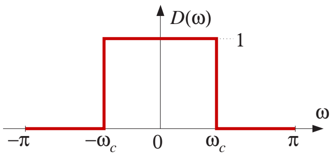
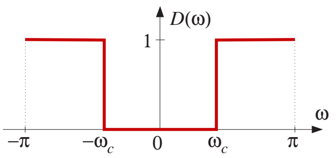
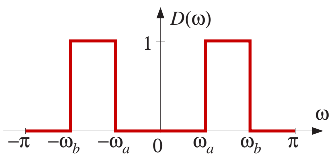
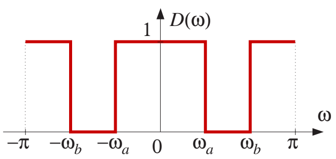
- Die unendlich lange Impulsantwort $d(k)$ wird symmetrisch mittels eines Fensters $w(n)$ bei $\pm M$ begrenzt. Dadurch ist die totale Anzahl Filterkoeffizienten ungerade. $\boxed{N = 2M+1}$

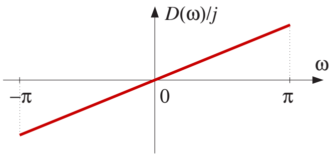
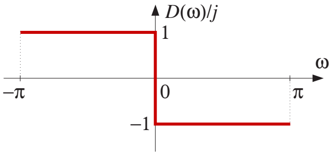
$$\boxed{\widehat{D}(\omega) = \sum_{k=-M}^{M} d(k)\, \mathrm{e}^{-j\omega k} \quad \Leftrightarrow \quad d(k) = \frac{1}{2\pi} \int_{-\pi}^{\pi} D(\omega)\, \mathrm{e}^{j\omega k}\, d\omega \qquad -M \leq k \leq M}$$

- Damit das FIR-Filter $h(n)$ kausal wird, muss die begrenzte Impulsantwort $d(k)$ noch um M Samples verzögert werden.

$$\boxed{H(\omega) = \mathrm{e}^{-j\omega M}\, \widehat{D}(\omega) \quad \Leftrightarrow \quad h(n) = d(n-M) \qquad n = 0, 1, ..., N-1}$$

- Diese Verzögerung um M Samples resultiert in einem linearphasigen Filter.
- Die Fenster-Methode ist sehr gut geeignet für einfache Frequenzgänge wie die folgender Filter.

Filter	**Frequenzgang $D(\omega)$ und Impulsantwort $d(k)$**	**Eigenschaften**
Tiefpass 	$\boxed{D(\omega) = \begin{cases} 1, & -\omega_c \leq \omega \leq \omega_c \\ 0, & \text{sonst} \end{cases}}$ $\boxed{d(k) = \frac{\sin(\omega_c k)}{\pi k}}$	Der Frequenzgang ist: - reell - gerade - symmetrisch Die Impulsantwort ist: - zweiseitig (akausal) - unendlich lang - reell - gerade - symmetrisch
Hochpass 	$\boxed{D(\omega) = \begin{cases} 0, & -\omega_c \leq \omega \leq \omega_c \\ 1, & \text{sonst} \end{cases}}$ $\boxed{d(k) = \delta(k) - \frac{\sin(\omega_c k)}{\pi k}}$	
Bandpass 	$\boxed{D(\omega) = \begin{cases} 1, & -\omega_b \leq \omega \leq -\omega_a \;\cup\; \omega_a \leq \omega \leq \omega_b \\ 0, & \text{sonst} \end{cases}}$ $\boxed{d(k) = \frac{\sin(\omega_b k) - \sin(\omega_a k)}{\pi k}}$	
Bandsperre 	$\boxed{D(\omega) = \begin{cases} 0, & -\omega_b \leq \omega \leq -\omega_a \;\cup\; \omega_a \leq \omega \leq \omega_b \\ 1, & \text{sonst} \end{cases}}$ $\boxed{d(k) = \delta(k) - \frac{\sin(\omega_b k) - \sin(\omega_a k)}{\pi k}}$	

Filter	**Frequenzgang $D(\omega)$ und Impulsantwort $d(k)$**	**Eigenschaften**
Differenzierer 	$\boxed{D(\omega) = j\omega}$ $\boxed{d(k) = \frac{\cos(\pi k)}{k} - \frac{\sin(\pi k)}{\pi k^2}}$	Der Frequenzgang ist: - imaginär - ungerade - antisymmetrisch Die Impulsantwort ist: - zweiseitig (akausal) - unendlich lang - reell - ungerade - antisymmetrisch
Hilbert-Transformator 	$\boxed{D(\omega) = -j\,\mathrm{sgn}(\omega)}$ $\boxed{d(k) = \frac{1-\cos(\pi k)}{\pi k}}$	

11.1 Eigenschaft der linearen Phase

Wenn FIR-Filter mit der Fenster-Methode designt werden resultiert ein **linearphasiges Filter**, d.h. das Filter verzögert alle Frequenzen um genau gleich viele (M) Samples $\Rightarrow$ konstante Gruppenlaufzeit!

Dies kommt daher, dass ...

- ... **symmetrische Impulsantworten keine Phase (Nullphasig) und damit keinen Delay haben.**
- ... die Begrenzung einer symmetrischen/antisymmetrischen Impulsantwort die symmetrie/antisymmetrie beibehaltet und der Frequenzgang nach wie vor rein reell/imaginär ist.

  Achtung:
  Impulsantwort muss symmetrisch begrenzt werden, da sonst die Symmetrie und damit die Linearphasigkeit aufgegeben wird.

- ... durch die eingeführte Zeitverzögerung (um das Filter kausal zu machen) aus dem nullphasigen Filter ein linearphasiges Filter wird.

Die Linearphasigkeit ist in folgenden Formeln erkennbar wobei

$$\boxed{\beta(\omega) = \frac{1-\mathrm{sgn}(\widehat{D}(\omega))}{2} = \begin{cases} 0, & \widehat{D}(\omega) > 0 \\ 1, & \widehat{D}(\omega) < 0 \end{cases}}$$

Symmetrischer Fall (rein reell):

$$H(\omega) = e^{-j\omega M}\,\widehat{D}(\omega) = e^{-j\omega M}\,\mathrm{sgn}(\widehat{D}(\omega))\,\big|\widehat{D}(\omega)\big| = e^{-j\omega M}\,e^{j\pi\beta(\omega)}\,\big|\widehat{D}(\omega)\big| = \underline{e^{j(-\omega M + \pi\beta(\omega))}\,\big|\widehat{D}(\omega)\big|}$$

$$\Rightarrow \quad \boxed{\big|H(\omega)\big| = \big|\widehat{D}(\omega)\big|, \qquad \arg H(\omega) = -\omega M + \underbrace{\pi\beta(\omega)}_{\text{Vorzeichenwechsel}}}$$

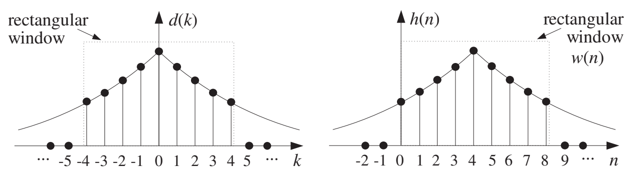
Antysymmetrischer Fall (rein imaginär):

$$\Rightarrow \quad \boxed{\big|H(\omega)\big| = \big|\widehat{D}(\omega)\big|, \qquad \arg H(\omega) = -\omega M + \underbrace{\frac{\pi}{2}}_{\text{Offset für } j} + \underbrace{\pi\beta(\omega)}_{\text{Vorzeichenwechsel}}}$$

11.2 Rechteck-Fenster

Das simpelste Fenster ist das Rechteck-Fenster

$$\boxed{w(n) = \begin{cases} 1, & 0 \le n \le N-1 \\ 0, & \text{sonst} \end{cases}}$$



Hierbei sind für das FIR-Filter-Design folgende Schritte notwendig:

1. Wähle eine ungerade Länge $N = 2M + 1$
2. Berechne die N Koeffizienten von $d(k)$ mit $k = -M \le k \le M$. **Aufpassen bei $k = 0$!**
3. Verzögere $d(k)$ um M-Samples um ein kausales Filter zu erhalten $\boxed{h(n) = d(n-M)}$ $n = 0, ..., N-1$

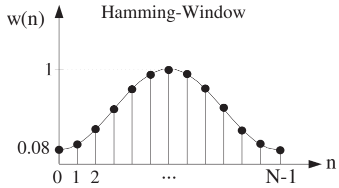
11.3 Hamming- und Hann-Fenster

Etwas komplexere Fenster sind das Hamming-Fenster und das Hann-Fenster.

Hamming-Fenster

$$\boxed{w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right)}$$

$n = 0, 1, ..., N-1$

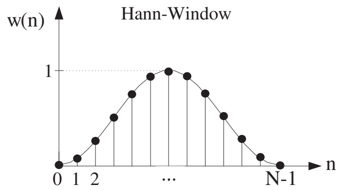


Hann-Fenster

$$\boxed{w(n) = 0.5 - 0.5\cos\left(\frac{2\pi n}{N-1}\right)}$$

$n = 0, 1, ..., N-1$

Hierbei sind für das FIR-Filter-Design folgende Schritte notwendig:



1. Wähle eine ungerade Länge $N = 2M + 1$
2. Berechne die N Koeffizienten von $d(k)$ mit $k = -M \leq k \leq M$.
   Aufpassen bei $k = 0$!
3. Verzögere $d(k)$ um M-Samples um ein kausales Filter zu erhalten und multipliziere die verzögerte Sequenz mit dem Fenster $w(n)$

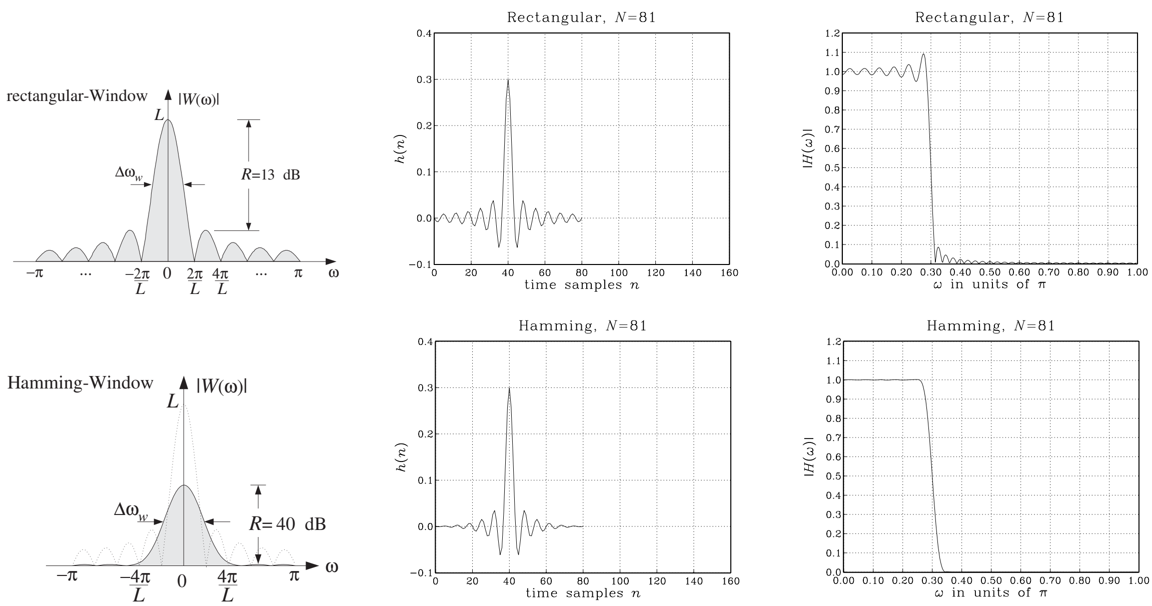
   $\boxed{h(n) = w(n)\,d(n-M)}$ $\quad n = 0, ..., N-1$

11.4 Frequenzgang der Filters mit dem Rechteck/Hamming-Fenster

- Beim Design mit der Fenster-Methode wird die verschobene Impulsantwort $d(n-M)$ mit einem Fenster multipliziert. Dies entspricht im Frequenzbereich einer Faltung.

$$\boxed{h(n) = w(n)\,d(n-M) \quad \Leftrightarrow \quad H(\omega) = \frac{1}{2\pi}\int_{-\pi}^{\pi} W(\omega-\omega')\,\mathrm{e}^{-j\omega' M}\,D(\omega')\,d\omega'}$$

- Durch die Faltung werden die Rippel der Impulsantwort des Fensters $W(\omega)$ integriert, was zu Rippel und Überschwinger im Frequenzgang $H(\omega)$ führt.



- Der Frequenzgang $H(\omega)$ ist nur durch die Anzahl Samples N beeinflussbar, dabei gilt:

$$\boxed{N \Uparrow \quad \Rightarrow \quad \text{Übergangsbandbreite } \Delta\omega \Downarrow \quad \cap \quad \text{schnelleres abklingen der Überschwinger}}$$

- Die Höhe der Überschwinger ist fensterabhängig und damit bei fixen Fenstern immer gleich gross

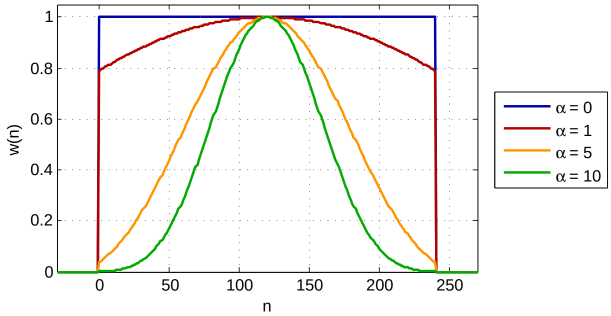
	Höhe Überschwinger	Stoppband-Dämpfung
Rechteck-Fenster	8.9%	$-20\log(0.089) = 21\text{dB}$
Hamming-Fenster	0.2%	$-20\log(0.002) = 54\text{dB}$

- Der Frequenzgang hat bei ω_c immer die halbe Höhe $\boxed{\left|H(\omega_c)\right| = 0.5}$

11.5 Kaiser-Fenster

Das Kaiser-Fenster ist eine Fenster-Familie welches zwei Parameter hat. Dadurch können folgende zwei Dinge eingestellt werden.

- Die Übergangsbandbreite $\Delta\omega$ über die Anzahl Samples N
- Die Höhe der Überschwinger bzw. die Stoppband-Dämpfung A_{stop} über den Parameter α



$$\boxed{\begin{aligned} w(n) &= \frac{I_0(\alpha\sqrt{1-(n-M)^2/M^2})}{I_0(\alpha)} \\ &= \frac{I_0(\alpha\sqrt{n(2M-n)}/M)}{I_0(\alpha)} \end{aligned}}$$

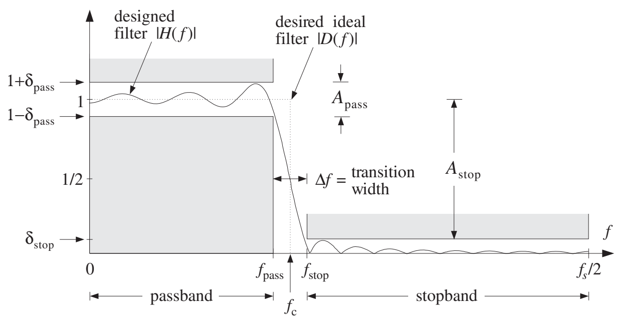
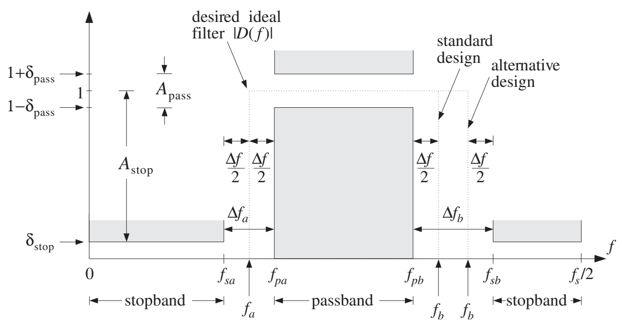
$n = 0, 1, ..., N-1$

$I_0(...)$: modifizierte Besselfunktion erster Gattung, 0-ter Ordnung.

$w(M) = 1$ und $w(0) = w(N-1) = 1/I_0(\alpha)$

$\alpha = 0 \quad \Rightarrow \quad$ Rechteck-Fenster

Für das FIR-Filter-Design mit Kaiser-Fenster müssen folgende Dinge gegeben sein:

Tiefpass / Hochpass	Bandpass / Bandsperre
- Stopp- und Druchlass-Frequenz $\boxed{f_{stop}\ ,\ f_{pass}}$	- Stopp- und Druchlass-Frequenz für beide Seiten $\boxed{f_{sa}\ ,\ f_{pa}\ ,\ f_{sb}\ ,\ f_{pb}}$
- Stopp- und Passband-Dämpfung $\boxed{A_{stop}\ ,\ A_{pass}}$	- Stopp- und Passband-Dämpfung $\boxed{A_{stop}\ ,\ A_{pass}}$
	

Für das FIR-Filter-Design folgende Schritte notwendig:

1. Berechnung der Übergangsbandbreite Δf

Tiefpass / Hochpass	Bandpass / Bandsperre		
$\boxed{\Delta f = \lvert f_{stop} - f_{pass}\rvert}$	$\boxed{\Delta f = \min\{\Delta f_a, \Delta f_b\}}$	$\boxed{\Delta f_a = \lvert f_{pa} - f_{sa}\rvert}$	$\boxed{\Delta f_b = \lvert f_{sb} - f_{pb}\rvert}$

2. Berechnung der Eckfrequenz f_c bzw. f_a und f_b

Tiefpass / Hochpass	Bandpass / Bandsperre		
$\boxed{f_c = \frac{1}{2}(f_{stop} + f_{pass})}$	Standard-Design:	$\boxed{f_a = f_{pa} - \frac{1}{2}\Delta f}$	$\boxed{f_b = f_{pb} + \frac{1}{2}\Delta f}$
	Alternatives-Design:	$\boxed{f_a = f_{sa} + \frac{1}{2}\Delta f}$	$\boxed{f_b = f_{sb} - \frac{1}{2}\Delta f}$

3. Berechnung der digitalen Eckfrequenz ω_c bzw. ω_a und ω_b

Tiefpass / Hochpass	Bandpass / Bandsperre	
$\boxed{\omega_c = \frac{2\pi f_c}{f_s}}$	$\boxed{\omega_a = \frac{2\pi f_a}{f_s}}$	$\boxed{\omega_b = \frac{2\pi f_b}{f_s}}$

4. Berechnung der Stopp- und Passband-Dämpfung δ_{stop} und δ_{pass}

$$\boxed{\delta_{stop} = 10^{-A_{stop}/20} \quad ; \quad \delta_{pass} = \frac{10^{A_{pass}/20} - 1}{10^{A_{pass}/20} + 1}}$$

$$\boxed{A_{stop} = -20\log_{10}(\delta_{stop}) \quad ; \quad A_{pass} = 20\log_{10}\left(\frac{1+\delta_{pass}}{1-\delta_{pass}}\right)}$$

5. Berechnung der Dämpfung δ und A

$$\boxed{\delta = \min\{\delta_{stop}, \delta_{pass}\}} \qquad \boxed{A = -20\log_{10}(\delta) \qquad \delta = 10^{-A/20}}$$

6. Berechnung des Parameter α und des Filterflankenfaktors D, welcher anzeigt, wie viel breiter die Mainlobe des Kaiser-Fenster ist als jene des Rechteck-Fensters (Schmale Mainlobe $\Rightarrow$ schmale Übergangsbandbreite Δf).

$$\boxed{\alpha = \begin{cases} 0.1102(A-8.7), & A \geq 50 \\ 0.5842(A-21)^{0.4} + 0.07886(A-21), & 21 < A < 50 \\ 0, & A \leq 21 \end{cases}} \qquad \boxed{D = \begin{cases} \dfrac{A-7.95}{14.36}, & A > 21 \\ 0.922, & A \leq 21 \end{cases}}$$

7. Berechnung der Filterlänge N mit anschliessendem Aufrunden auf die nächste ungerade Zahl und Berechnung der Verschiebung M.

$$\boxed{\Delta f = \frac{D f_s}{N-1} \quad \Leftrightarrow \quad N = \frac{D f_s}{\Delta f} + 1} \qquad \boxed{N = 2M+1} \qquad \boxed{M = \frac{N-1}{2}}$$

8. Berechnung des Kaiser-Fensters $w(n)$ für $n = 0, 1, ..., N-1$.

9. Berechnung der Impulsantwort $h(n)$ aus der Impulsantwort des gewünschten Filters $d(k)$.

$$\boxed{h(n) = w(n)\, d(n-M)} \qquad n = 0, ..., N-1$$

wobei $h(M) = \omega_c/\pi$ bzw. $h(M) = (\omega_b - \omega_a)/\pi$ ist

Damit ergeben sich im Vergleich zum Rechteck- und Hamming-Fenster folgende Filter-Kennwerte

Fenster	δ	A_{stop}	A_{pass}	D
Rechteck	8.9%	21 dB	1.55 dB	0.92
Hamming	0.2%	54 dB	0.03 dB	3.21
Kaiser	variabel	$-20\log_{10}(\delta)$	17.372δ	$(A-7.95)/14.36$

11.5.1 Eigenschaften des Fenster-Spektrums des Kaiser-Fensters

Im Kapitel 10.1.3 wurde der Tradeoff zwischen der Mainlobe-Width $\Delta\omega_w$ (Frequenzauflösung) und der Sidelobe-Suppression R (frequency leakage) diskutiert welche jeweils fensterabhängig waren. Da das Kaiser-Fenster jedoch eine variable Form hat, variiert auch der Faktor c und die Sidelobe-Suppression R je nach dem, welches α für das Kaiser-Fenster gewählt wird.

$$\boxed{\Delta\omega_W = c \cdot \frac{2\pi}{L-1}} \qquad \boxed{\Delta f_W = c \cdot \frac{f_s}{L-1} \quad \Leftrightarrow \quad L-1 = c \cdot \frac{f_s}{\Delta f_W}} \qquad \text{mit} \qquad \boxed{c = \frac{6(R+12)}{155}}$$

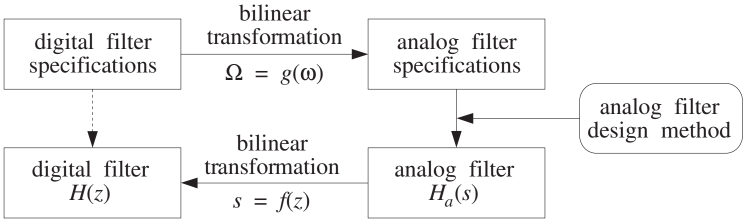
$$\Rightarrow \quad \boxed{\alpha = \begin{cases} 0.12438(R+6.3), & 60 < R < 120 \\ 0.76609(R-13.26)^{0.4} + 0.09834(R-13.26), & 13.26 < R < 60 \\ 0, & R < 13.26 \end{cases}}$$

Vergleich der Parameterwerte mit den anderen Fenstern

Fenster	R	c
Rechteck	13 dB	1
Hamming	40 dB	2
Kaiser	variabel	$6(R+12)/155$

12 IIR Digital Filter Design **S.563**

Ein einfacher aber indirekter Weg IIR-Filter zu designen ist mittels der Bilinear-Transformation und den Design-Methoden von analogen Filtern. Dazu sind folgende schritte notwendig:



- Die Spezifikationen des digitalen Frequenzganges mittels der Bilinear-Transformation in Spezifikationen des entsprechenden analogen Tiefpasses umwandeln.

digitaler Tiefpass	$\Rightarrow$	analoger Tiefpass:	$\boxed{\Omega = g_{TP}(\omega) = \tan\left(\frac{\omega}{2}\right)}$
digitaler Hochpass	$\Rightarrow$	analoger Tiefpass:	$\boxed{\Omega = g_{HP}(\omega) = -\cot\left(\frac{\omega}{2}\right)}$
digitaler Bandpass	$\Rightarrow$	analoger Tiefpass:	$\boxed{\Omega = g_{BP}(\omega) = \frac{c-\cos(\omega)}{\sin(\omega)}}$
digitale Bandsperre	$\Rightarrow$	analoger Tiefpass:	$\boxed{\Omega = g_{BS}(\omega) = \frac{\sin(\omega)}{\cos(\omega)-c}}$

- Den analogen Tiefpass $H_a(s)$ mit analogen Filter-Design-Methoden designen.
- Den analogen Tiefpass $H_a(s)$ mit der Bilinear-Transformation in das gewünschte digitale Filter $H(z)$ zurückmappen.

analoger Tiefpass	$\Rightarrow$	digitaler Tiefpass:	$\boxed{s = f_{TP}(z) = \frac{1-z^{-1}}{1+z^{-1}}}$
analoger Tiefpass	$\Rightarrow$	digitaler Hochpass:	$\boxed{s = f_{HP}(z) = \frac{1+z^{-1}}{1-z^{-1}}}$
analoger Tiefpass	$\Rightarrow$	digitaler Bandpass:	$\boxed{s = f_{BP}(z) = \frac{1-2cz^{-1}+z^{-2}}{1-z^{-2}}}$
analoger Tiefpass	$\Rightarrow$	digitale Bandsperre:	$\boxed{s = f_{BS}(z) = \frac{1-z^{-2}}{1-2cz^{-1}+z^{-2}}}$

- Damit gelten folgende Beziehungen zwischen dem analogen und digitalen Filter

$$\boxed{H(z) = H_a(s)\Big|_{s=f(z)} = H_a(f(z))} \qquad \boxed{H(\omega) = H_a(s)\Big|_{s=j\Omega} = H_a(\Omega)\Big|_{\Omega=g(\omega)} = H_a(g(\omega))}$$

12.1 Eigenschaften der Bilinear-Transformation

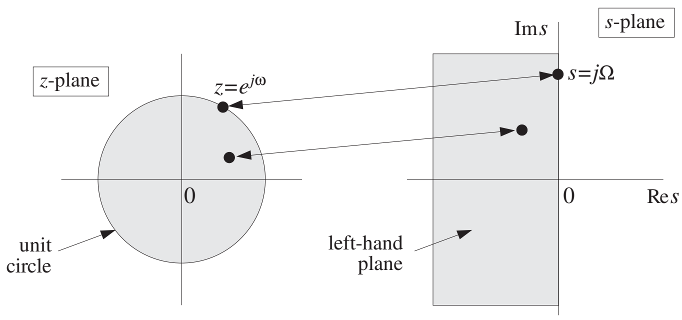
Die Bilinear-Transformation mappt ...

- die linke s-Halbebene auf das innere des Einheitskreises der z-Ebene (wenn $|c| \leq 1$).

$$\boxed{\text{stabiles analoges Filter} \overset{|c|\leq 1}{\Leftrightarrow} \text{stabiles digitales Filter}}$$

- die analoge Frequenzachse $s = j\Omega$ auf die digitale Frequenzachse $z = e^{j\omega}$.

$$\boxed{\omega = \pm\pi \quad \widehat{=} \quad \Omega = \pm\infty}$$



12.2 Tiefpass-Filter und Hochpass-Filter erster Ordnung

- Ein digitales Filter erster Ordnung hat die parametrische Form $\boxed{H(z) = \dfrac{b_0 + b_1\, z^{-1}}{1 + a_1\, z^{-1}}}$

- Von diesem muss die erlaubte Dämpfung A_c (in dB) bei der Grenzfrequenz ω_c spezifiziert werden.

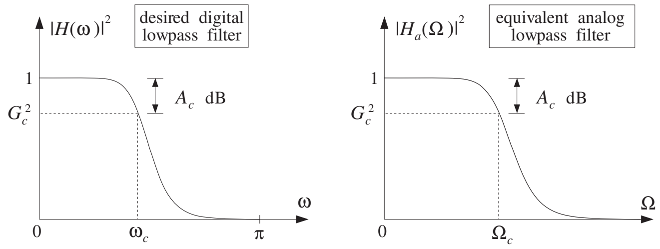
  $\boxed{\big|H(\omega_c)\big|^2 = G_c^2 = 10^{-A_c/10}}$ $\boxed{G_c = 10^{-A_c/20}}$ $\boxed{A_c = -10\,\log_{10}(G_c^2) = -20\,\log_{10}(G_c)}$

  Oft $\boxed{A_c = 3\,\text{dB} \quad\Rightarrow\quad G_c^2 = 1/2}$

- Die entsprechende analoge Grenzfrequenz ist $\boxed{\Omega_c = g_{LP}(\omega_c) = \tan\left(\dfrac{\omega_c}{2}\right) = \tan\left(\dfrac{\pi f_c}{f_s}\right)}$

Analoges Tiefpass-Filter

$$\boxed{H_a(s) = \frac{\alpha}{s+\alpha}}$$



- Parameter α

$$\big|H_a(\Omega_c)\big|^2 = \frac{\alpha^2}{|j\Omega_c + \alpha|^2} = \frac{\alpha^2}{\Omega_c^2 + \alpha^2} = G_c^2$$

$$\Rightarrow \quad \boxed{\alpha = \frac{G_c}{\sqrt{1-G_c^2}}\;\Omega_c = \underbrace{\frac{G_c}{\sqrt{1-G_c^2}}}_{1 \text{ if } G_c^2=1/2}\tan\left(\frac{\omega_c}{2}\right)}$$

Digitales Tiefpass-Filter

- Übertragungsfunktion $H(z)$

$$H(z) = H_a(s)\Big|_{s=f_{TP}(z)} = \frac{\alpha\,(1+z^{-1})}{1-z^{-1}+\alpha\,(1+z^{-1})}$$

$$\Rightarrow \quad \boxed{H(z) = b\,\frac{1+z^{-1}}{1-a\,z^{-1}}}$$

$$\text{mit} \quad \boxed{a = \frac{1-\alpha}{1+\alpha}, \quad b = \frac{\alpha}{1+\alpha} = \frac{1-a}{2}}$$

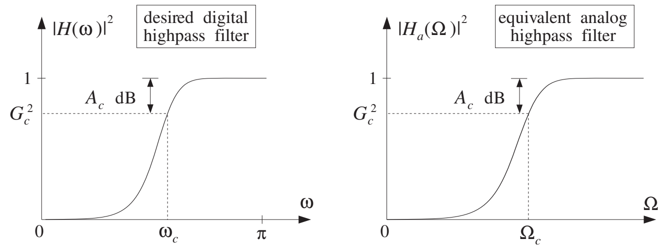
- Frequenzgang $H(\omega)$

$$H(\omega) = H_a(\Omega)\Big|_{\Omega=g_{TP}(\omega)} = \frac{\alpha}{\alpha + j\tan(\omega/2)}$$

$$\Rightarrow \quad \boxed{\big|H(\omega)\big|^2 = \frac{\alpha^2}{\alpha^2 + \tan^2(\omega/2)}}$$

Analoges Hochpass-Filter

$$\boxed{H_a(s) = \frac{s}{s+\alpha}}$$



- Parameter α

$$\big|H_a(\Omega_c)\big|^2 = \frac{\Omega_c^2}{|j\Omega_c + \alpha|^2} = \frac{\Omega_c^2}{\Omega_c^2 + \alpha^2} = G_c^2$$

$$\Rightarrow \quad \boxed{\alpha = \frac{\sqrt{1-G_c^2}}{G_c}\;\Omega_c = \underbrace{\frac{\sqrt{1-G_c^2}}{G_c}}_{1 \text{ if } G_c^2=1/2}\tan\left(\frac{\omega_c}{2}\right)}$$

Digitales Hochpass-Filter

- Übertragungsfunktion $H(z)$

$$H(z) = H_a(s)\Big|_{s=f_{TP}(z)} = \frac{1-z^{-1}}{1-z^{-1}+\alpha\,(1+z^{-1})}$$

$$\Rightarrow \quad \boxed{H(z) = b\,\frac{1-z^{-1}}{1-a\,z^{-1}}}$$

$$\text{mit} \quad \boxed{a = \frac{1-\alpha}{1+\alpha}, \quad b = \frac{1}{1+\alpha} = \frac{1+a}{2}}$$

- Frequenzgang $H(\omega)$

$$H(\omega) = H_a(\Omega)\Big|_{\Omega=g_{TP}(\omega)} = \frac{j\tan(\omega/2)}{\alpha + j\tan(\omega/2)}$$

$$\Rightarrow \quad \boxed{\big|H(\omega)\big|^2 = \frac{\tan^2(\omega/2)}{\alpha^2 + \tan^2(\omega/2)}}$$

- Wenn des Hochpass- und das Tiefpass-Filter die selbe 3 dB-Grenzfrequenz ω_c haben, sind sie komplementäre Filter.

  $\boxed{H_{TP}(z) + H_{HP}(z) = 1}$ und $\boxed{\alpha = \Omega_c = \tan\left(\dfrac{\omega_c}{2}\right)}$

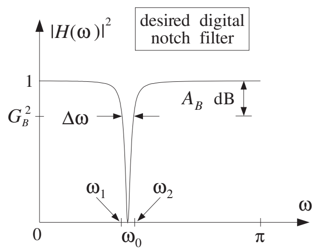
12.3 Notch-Filter und Peak-Filter zweiter Ordnung

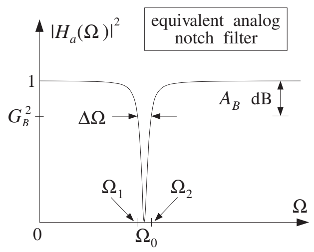
- Ein digitales Filter zweiter Ordnung hat die parametrische Form $\boxed{H(z) = \dfrac{b_0 + b_1\, z^{-1} + b_2\, z^{-2}}{1 + a_1\, z^{-1} + a_2\, z^{-2}}}$
- Die Spezifikations-Parameter eines Notch- bzw. Peak-Filters sind:
  - Mittenfrequenz ω_0
  - Bandbreite $\boxed{\Delta\omega = \omega_2 - \omega_1}$
  - Dämpfung G_B^2 bei den Grenzfrequenzen ω_1 und ω_2 $\boxed{G_B^2 = \left|H(\omega_1)\right|^2 = \left|H(\omega_2)\right|^2}$
- Die entsprechenden analogen Parameter-Spezifikationen sind:

$$\boxed{\Omega_0 = \tan\left(\frac{\omega_0}{2}\right)} \qquad \boxed{\Delta\Omega = (1+\Omega_0^2)\,\tan\left(\frac{\Delta\omega}{2}\right)} \qquad \boxed{\Omega_0^2 = \Omega_1 \cdot \Omega_2}$$

Analoges Notch-Filter

$$\boxed{H_a(s) = \frac{s^2 + \Omega_0^2}{s^2 + \alpha s + \Omega_0^2}}$$





- Parameter α

$$\left|H_a(\Omega_{1,2})\right|^2 = \frac{(\Omega_{1,2}^2 - \Omega_0^2)^2}{(\Omega_{1,2}^2 - \Omega_0^2)^2 + \alpha^2\,\Omega_{1,2}^2} = G_B^2$$

$$\Rightarrow \quad \boxed{\alpha = \underbrace{\frac{\sqrt{1-G_B^2}}{G_B}}_{1 \text{ if } G_B^2 = 1/2}\ \underbrace{(1+\Omega_0^2)\,\tan\left(\frac{\Delta\omega}{2}\right)}_{\Delta\Omega}}$$

Digitales Notch-Filter

- Übertragungsfunktion $H(z)$

$$H(z) = H_a(s)\Big|_{s=f_{TP}(z)}$$

$$\Rightarrow \quad \boxed{H(z) = b\,\frac{1 - 2\cos(\omega_0)z^{-1} + z^{-2}}{1 - 2b\cos(\omega_0)\,z^{-1} + (2b-1)\,z^{-2}}}$$

$$\text{mit} \quad \boxed{b = \frac{1}{1 + \underbrace{\frac{\sqrt{1-G_B^2}}{G_B}\ \tan\left(\frac{\Delta\omega}{2}\right)}_{\beta}}}$$

- Frequenzgang $H(\omega)$

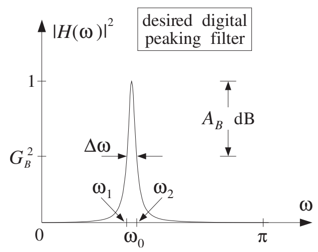
$$H(\omega) = H_a(\Omega)\Big|_{\Omega=g_{TP}(\omega)}$$

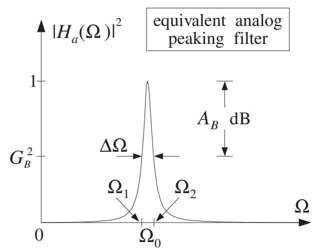
$$= \frac{-\tan^2(\frac{\omega}{2}) + \Omega_0^2}{-\tan^2(\frac{\omega}{2}) + j\alpha\tan(\frac{\omega}{2}) + \Omega_0^2}$$

$$\Rightarrow \quad \boxed{\left|H(\omega)\right|^2 = \frac{(\tan^2(\frac{\omega}{2}) - \Omega_0^2)^2}{(\tan^2(\frac{\omega}{2}) - \Omega_0^2)^2 + \alpha^2\,\tan^2(\frac{\omega}{2})}}$$

Analoges Peak-Filter

$$\boxed{H_a(s) = \frac{\alpha s}{s^2 + \alpha s + \Omega_0^2}}$$





- Parameter α

$$\left|H_a(\Omega_{1,2})\right|^2 = \frac{\alpha^2\,\Omega_{1,2}^2}{(\Omega_{1,2}^2 - \Omega_0^2)^2 + \alpha^2\,\Omega_{1,2}^2} = G_B^2$$

$$\Rightarrow \quad \boxed{\alpha = \underbrace{\frac{G_B}{\sqrt{1-G_B^2}}}_{1 \text{ if } G_B^2 = 1/2}\ \underbrace{(1+\Omega_0^2)\,\tan\left(\frac{\Delta\omega}{2}\right)}_{\Delta\Omega}}$$

Digitales Peak-Filter

- Übertragungsfunktion $H(z)$

$$H(z) = H_a(s)\Big|_{s=f_{TP}(z)}$$

$$\Rightarrow \quad \boxed{H(z) = \frac{(1-b)\,(1-z^{-2})}{1 - 2b\cos(\omega_0)\,z^{-1} + (2b-1)\,z^{-2}}}$$

$$\text{mit} \quad \boxed{b = \frac{1}{1 + \underbrace{\frac{G_B}{\sqrt{1-G_B^2}}\ \tan\left(\frac{\Delta\omega}{2}\right)}_{\beta}}}$$

- Frequenzgang $H(\omega)$

$$H(\omega) = H_a(\Omega)\Big|_{\Omega=g_{TP}(\omega)}$$

$$= \frac{j\alpha\tan(\frac{\omega}{2})}{-\tan^2(\frac{\omega}{2}) + j\alpha\tan(\frac{\omega}{2}) + \Omega_0^2}$$

$$\Rightarrow \quad \boxed{\left|H(\omega)\right|^2 = \frac{\alpha^2\,\tan^2(\frac{\omega}{2})}{(\tan^2(\frac{\omega}{2}) - \Omega_0^2)^2 + \alpha^2\,\tan^2(\frac{\omega}{2})}}$$

- Das Notch-Filter hat eine Doppelte Nullstelle bei ω_0 $\boxed{z_{NS} = e^{\pm j\omega_0}}$
  und das Peak-Filter hat die Nullstellen Nullstellen bei DC und bei der halben Abtast-frequenz $\boxed{z_{NS} = \pm 1}$
- Wenn des Notch- und das Peak-Filter die selbe 3 dB-Mittenfrequenz ω_0 und die selbe Bandbreite $\Delta\omega$ haben, sind sie komplementäre Filter. $\boxed{H_{notch}(z) + H_{peak}(z) = 1}$

12.4 Filter höherer Ordnung

- Für das Design eines Filters höherer Ordnung müssen folgende Spezifikationen bekannt sein:
  - Filtertyp
  - Durchlassbereich f_{pass}
  - Sperrbereich f_{stop}
  - Passband-Dämpfung A_{pass}
  - Stoppband-Dämpfung A_{stop}
- Die Pass- und Stoppband-Dämpfung können in relativen (dB) oder in Absoluten Werten gegeben sein.

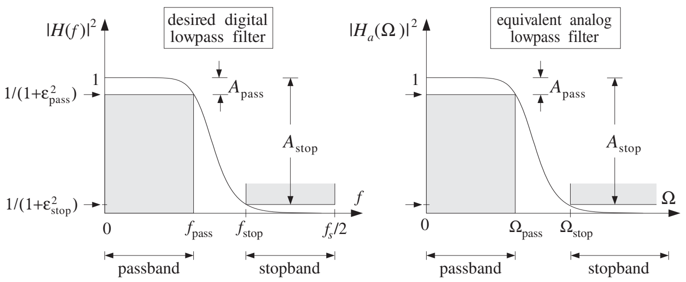
$$\boxed{\begin{aligned} \varepsilon_{pass} &= \sqrt{10^{A_{pass}/10} - 1} \\ \varepsilon_{stop} &= \sqrt{10^{A_{stop}/10} - 1} \end{aligned}} \quad \Leftrightarrow \quad \boxed{\begin{aligned} A_{pass} &= 10\log_{10}(1 + \varepsilon_{pass}^2) \\ A_{stop} &= 10\log_{10}(1 + \varepsilon_{stop}^2) \end{aligned}}$$

- Umrechnung der digitalen Filter-Spezifikationen in die Spezifikationen des analogen Tiefpasses mittels Bilinear-Transformationen.

Tiefpass

$$\boxed{\Omega_{pass} = \tan\left(\frac{\omega_{pass}}{2}\right) = \tan\left(\frac{\pi f_{pass}}{f_s}\right)}$$

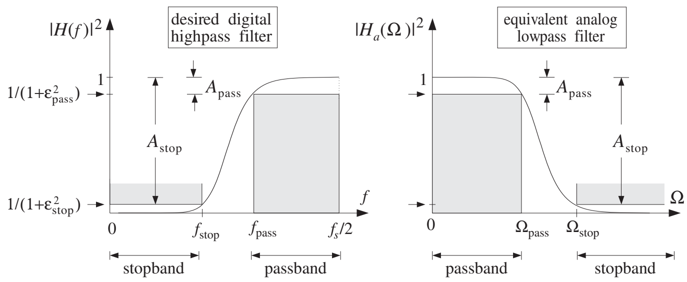
$$\boxed{\Omega_{stop} = \tan\left(\frac{\omega_{stop}}{2}\right) = \tan\left(\frac{\pi f_{stop}}{f_s}\right)}$$



Hochpass

$$\boxed{\Omega_{pass} = -\cot\left(\frac{\omega_{pass}}{2}\right) = -\cot\left(\frac{\pi f_{pass}}{f_s}\right)}$$

$$\boxed{\Omega_{stop} = -\cot\left(\frac{\omega_{stop}}{2}\right) = -\cot\left(\frac{\pi f_{stop}}{f_s}\right)}$$



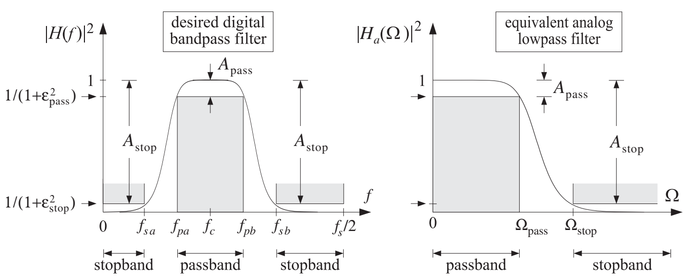
Bandpass

$$\boxed{c = \cos(\omega_c)} \quad \Rightarrow \quad \boxed{c = \frac{\sin(\omega_{pa} + \omega_{pb})}{\sin(\omega_{pa}) + \sin(\omega_{pb})}}$$

$$\boxed{\Omega_{pass} = \left|\frac{c - \cos(\omega_{pb})}{\sin(\omega_{pb})}\right|}$$

$$\boxed{\Omega_{stop} = \min\{|\Omega_{sa}|, |\Omega_{sb}|\}}$$

$$\text{mit} \quad \boxed{\Omega_{sa} = \frac{c - \cos(\omega_{sa})}{\sin(\omega_{sa})}} \quad \boxed{\Omega_{sb} = \frac{c - \cos(\omega_{sb})}{\sin(\omega_{sb})}}$$



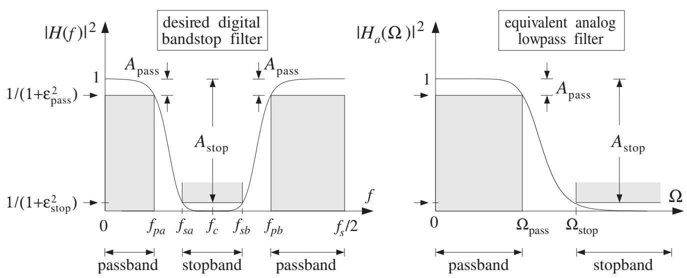
Bandsperre

$$\boxed{c = \cos(\omega_c)} \quad \Rightarrow \quad \boxed{c = \frac{\sin(\omega_{pa} + \omega_{pb})}{\sin(\omega_{pa}) + \sin(\omega_{pb})}}$$

$$\boxed{\Omega_{pass} = \left|\frac{\sin(\omega_{pb})}{\cos(\omega_{pb}) - c}\right|}$$

$$\boxed{\Omega_{stop} = \min\{|\Omega_{sa}|, |\Omega_{sb}|\}}$$

$$\text{mit} \quad \boxed{\Omega_{sa} = \frac{\sin(\omega_{sa})}{\cos(\omega_{sa}) - c}} \quad \boxed{\Omega_{sb} = \frac{\sin(\omega_{sb})}{\cos(\omega_{sb}) - c}}$$



12.4.1 Butterworth-Filter

- Das analoge Tiefpass kann mit einer beliebigen Filter approximiert werden. Eine sehr weit verbreitete und simple Approximation ist das Butterworth-Filter.

Butterworth-Filter: $\boxed{\left|H(\Omega)\right|^2 = \dfrac{1}{1+\left(\dfrac{\Omega}{\Omega_0}\right)^{2N}}}$ $\quad N$: Filterordnung, Ω_0 : 3 dB Grenzfrequenz

Dämpfung in dB: $\boxed{A(\Omega) = -10\log_{10}\left|H(\Omega)\right|^2 = 10\log_{10}\left[1+\left(\dfrac{\Omega}{\Omega_0}\right)^{2N}\right]}$

- Vom Butterworth-Filter ist die Ordnung N zu berechnen.

$$\boxed{N = \lceil N_{exakt} \rceil} \qquad \boxed{N_{exakt} = \frac{\ln(\varepsilon_{stop}/\varepsilon_{pass})}{\ln(\Omega_{stop}/\Omega_{pass})} = \frac{\ln(e)}{\ln(w)}} \qquad \boxed{e = \frac{\varepsilon_{stop}}{\varepsilon_{pass}} = \sqrt{\frac{10^{A_{stop}/10}-1}{10^{A_{pass}/10}-1}}, \quad w = \frac{\Omega_{stop}}{\Omega_{pass}}}$$

- Aufgrund der aufgerundeten Ordnung N ist das Filter etwas "zu gut", d.h. dass entweder die Stoppband-Dämpfung oder die Passband-Dämpfung "zu gut" ist, je nach dem wie die exakte 3 dB Grenzfrequenz Ω_0 bestimmt wird.

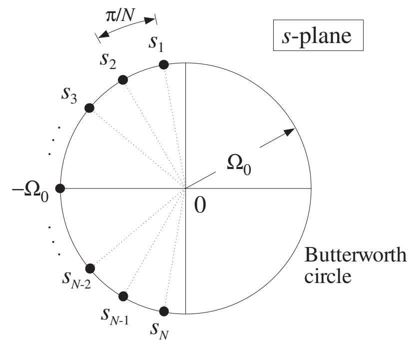
$A(\Omega_{stop}) > A_{stop}$: $\boxed{\Omega_0 = \dfrac{\Omega_{pass}}{(10^{A_{pass}/10}-1)^{1/2N}} = \dfrac{\Omega_{pass}}{\varepsilon_{pass}^{1/N}}}$

$A(\Omega_{pass}) < A_{pass}$: $\boxed{\Omega_0 = \dfrac{\Omega_{stop}}{(10^{A_{stop}/10}-1)^{1/2N}} = \dfrac{\Omega_{stop}}{\varepsilon_{stop}^{1/N}}}$

- Die Übertragungsfunktion $H(s)$ kann über eine spektrale Faktorisierung ermittelt werden. Daraus geht hervor, dass alle Polstellen s_i (Nullstellen des Nenner-Polynoms) auf einem Halbkreis in der linken s-Halbebene liegen müssen, damit das Filter stabil ist.

$$\boxed{s_i = \Omega_0\, e^{j\theta_i}, \qquad \theta_i = \frac{\pi}{2N}(N-1+2i)} \qquad i = 1,2,...,N$$

$$\Rightarrow \qquad \boxed{H_a(s) = \prod_{i=1}^{N} \frac{1}{(1-\frac{1}{\Omega_0}\,e^{-j\theta_i}\, s)}}$$



- Die Polstellen kommen immer in konjugiert-komplexen-Paaren vor und können daher zu Second-Order-Sections zusammengefasst werden

$$\boxed{H_a(s) = H_0(s)\, H_1(s)\, H_2(s) \ldots H_K(s)}$$

$$\boxed{\begin{aligned} H_0(s) &= \begin{cases} 1, & N = 2K \\ \dfrac{1}{1+s/\Omega_0}, & N = 2K+1 \end{cases} \\ H_i(s) &= \frac{1}{1-2\dfrac{s}{\Omega_0}\cos(\theta_i)+\dfrac{s^2}{\Omega_0^2}} \qquad i = 1,2,...,K \end{aligned}}$$

- Der analoge Tiefpass $H_a(s)$ muss nun mit der entsprechenden Bilinear-Transformation wieder in das gewünschte digitale Filter $H(z)$ zurückgemappt werden.
  - Tiefpass $\overset{s=f_{TP}(z)}{\Longrightarrow}$ Tiefpass
  - Tiefpass $\overset{s=f_{BP}(z)}{\Longrightarrow}$ Bandpass
  - Tiefpass $\overset{s=f_{HP}(z)}{\Longrightarrow}$ Hochpass
  - Tiefpass $\overset{s=f_{BS}(z)}{\Longrightarrow}$ Bandsperre

Tiefpass

Übertragungsfunktion $H(z)$:

$$H_i(z) = H_i(s)\Big|_{s=f_{TP}(z)}$$

$$\Rightarrow \boxed{H_0(z) = \frac{G_0(1+z^{-1})}{1+a_{01}\,z^{-1}}}$$

mit $\boxed{G_0 = \frac{\Omega_0}{\Omega_0+1}}$ und $\boxed{a_{01} = \frac{\Omega_0-1}{\Omega_0+1}}$

$$\Rightarrow \boxed{H_i(z) = \frac{G_i(1+z^{-1})^2}{1+a_{i1}\,z^{-1}+a_{i2}\,z^{-2}}}$$

mit $\boxed{G_i = \frac{\Omega_0^2}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i1} = \frac{2(\Omega_0^2-1)}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i2} = \frac{1+2\,\Omega_0\cos(\theta_i)+\Omega_0^2}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

$$\Rightarrow \boxed{H(z) = H_0(z)\,H_1(z)\,H_2(z)\ldots H_K(z)}$$

Frequenzgang $H(\omega)$:

$$H(\omega) = H_a(\Omega)\Big|_{\Omega=g_{TP}(\omega)}$$

$$\Rightarrow \boxed{\left|H(\omega)\right|^2 = \frac{1}{1+(\tan(\frac{\omega}{2})/\Omega_0)^{2N}}}$$

3 dB Grenzfrequenz ω_0:

$\boxed{\omega_0 = 2\,\arctan(\Omega_0)}$ $\boxed{f_0 = \frac{f_s}{\pi}\,\arctan(\Omega_0)}$

Hochpass

Übertragungsfunktion $H(z)$:

$$H_i(z) = H_i(s)\Big|_{s=f_{HP}(z)}$$

$$\Rightarrow \boxed{H_0(z) = \frac{G_0(1-z^{-1})}{1+a_{01}\,z^{-1}}}$$

mit $\boxed{G_0 = \frac{\Omega_0}{\Omega_0+1}}$ und $\boxed{a_{01} = -\frac{\Omega_0-1}{\Omega_0+1}}$

$$\Rightarrow \boxed{H_i(z) = \frac{G_i(1-z^{-1})^2}{1+a_{i1}\,z^{-1}+a_{i2}\,z^{-2}}}$$

mit $\boxed{G_i = \frac{\Omega_0^2}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i1} = -\frac{2(\Omega_0^2-1)}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i2} = \frac{1+2\,\Omega_0\cos(\theta_i)+\Omega_0^2}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

$$\Rightarrow \boxed{H(z) = H_0(z)\,H_1(z)\,H_2(z)\ldots H_K(z)}$$

Frequenzgang $H(\omega)$:

$$H(\omega) = H_a(\Omega)\Big|_{\Omega=g_{HP}(\omega)}$$

$$\Rightarrow \boxed{\left|H(\omega)\right|^2 = \frac{1}{1+(\cot(\frac{\omega}{2})/\Omega_0)^{2N}}}$$

3 dB Grenzfrequenz ω_0:

$\boxed{\omega_0 = 2\,\arctan\left(\frac{1}{\Omega_0}\right)}$ $\boxed{f_0 = \frac{f_s}{\pi}\,\arctan\left(\frac{1}{\Omega_0}\right)}$

Bandpass

Übertragungsfunktion $H(z)$:

$$H_i(z) = H_i(s)\Big|_{s=f_{BP}(z)}$$

$$\Rightarrow \boxed{H_0(z) = \frac{G_0(1-z^{-2})}{1+a_{01}\,z^{-1}+a_{02}\,z^{-2}}}$$

mit $\boxed{G_0 = \frac{\Omega_0}{1+\Omega_0}}$ und $\boxed{a_{01} = -\frac{2c}{1+\Omega_0}}$

und $\boxed{a_{02} = \frac{1-\Omega_0}{1+\Omega_0}}$

$$\Rightarrow \boxed{H_i(z)=\frac{G_i(1-z^{-2})^2}{1+a_{i1}z^{-1}+a_{i2}z^{-2}+a_{i3}z^{-3}+a_{i4}z^{-4}}}$$

mit $\boxed{G_i = \frac{\Omega_0^2}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i1} = \frac{4c\,(\Omega_0\cos(\theta_i)-1)}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i2} = \frac{2\,(2c^2+1-\Omega_0^2)}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i3} = -\frac{4c\,(\Omega_0\cos(\theta_i)+1)}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i4} = \frac{1+2\,\Omega_0\cos(\theta_i)+\Omega_0^2}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

$$\Rightarrow \boxed{H(z) = H_0(z)\,H_1(z)\,H_2(z)\ldots H_K(z)}$$

Frequenzgang $H(\omega)$:

$$H(\omega) = H_a(\Omega)\Big|_{\Omega=g_{BP}(\omega)}$$

$$\Rightarrow \boxed{\big|H(\omega)\big|^2 = \frac{1}{1+\left(\frac{(c-\cos(\omega))/\sin(\omega)}{\Omega_0}\right)^{2N}}}$$

3 dB Grenzfrequenzen ω_{0a} und ω_{0b}:

$$\boxed{\tan\left(\frac{\omega_{0a}}{2}\right)=\tan\left(\frac{\pi f_{0a}}{f_s}\right)=\frac{\sqrt{\Omega_0^2+1-c^2}-\Omega_0}{1+c}}$$

$$\boxed{\tan\left(\frac{\omega_{0b}}{2}\right)=\tan\left(\frac{\pi f_{0b}}{f_s}\right)=\frac{\sqrt{\Omega_0^2+1-c^2}+\Omega_0}{1+c}}$$

Bandsperre

Übertragungsfunktion $H(z)$:

$$H_i(z) = H_i(s)\Big|_{s=f_{BS}(z)}$$

$$\Rightarrow \boxed{H_0(z) = \frac{G_0(1-2c\,z^{-1}+z^{-2})}{1+a_{01}\,z^{-1}+a_{02}\,z^{-2}}}$$

mit $\boxed{G_0 = \frac{\Omega_0}{1+\Omega_0}}$ und $\boxed{a_{01} = -\frac{2c\,\Omega_0}{1+\Omega_0}}$

und $\boxed{a_{02} = -\frac{1-\Omega_0}{1+\Omega_0}}$

$$\Rightarrow \boxed{H_i(z)=\frac{G_i(1-2c\,z^{-1}+z^{-2})^2}{1+a_{i1}z^{-1}+a_{i2}z^{-2}+a_{i3}z^{-3}+a_{i4}z^{-4}}}$$

mit $\boxed{G_i = \frac{\Omega_0^2}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i1} = \frac{4c\,\Omega_0\,(\cos(\theta_i)-\Omega_0)}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i2} = \frac{2\,(2c^2\,\Omega_0^2+\Omega_0^2-1)}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i3} = -\frac{4c\,\Omega_0\,(\cos(\theta_i)+\Omega_0)}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

und $\boxed{a_{i4} = \frac{1+2\,\Omega_0\cos(\theta_i)+\Omega_0^2}{1-2\,\Omega_0\cos(\theta_i)+\Omega_0^2}}$

$$\Rightarrow \boxed{H(z) = H_0(z)\,H_1(z)\,H_2(z)\ldots H_K(z)}$$

Frequenzgang $H(\omega)$:

$$H(\omega) = H_a(\Omega)\Big|_{\Omega=g_{BS}(\omega)}$$

$$\Rightarrow \boxed{\big|H(\omega)\big|^2 = \frac{1}{1+\left(\frac{\sin(\omega)/(\cos(\omega)-c)}{\Omega_0}\right)^{2N}}}$$

3 dB Grenzfrequenzen ω_{0a} und ω_{0b}:

$$\boxed{\tan\left(\frac{\omega_{0a}}{2}\right)=\tan\left(\frac{\pi f_{0a}}{f_s}\right)=\frac{\sqrt{1+\Omega_0^2\,(1-c^2)}-1}{\Omega_0(1+c)}}$$

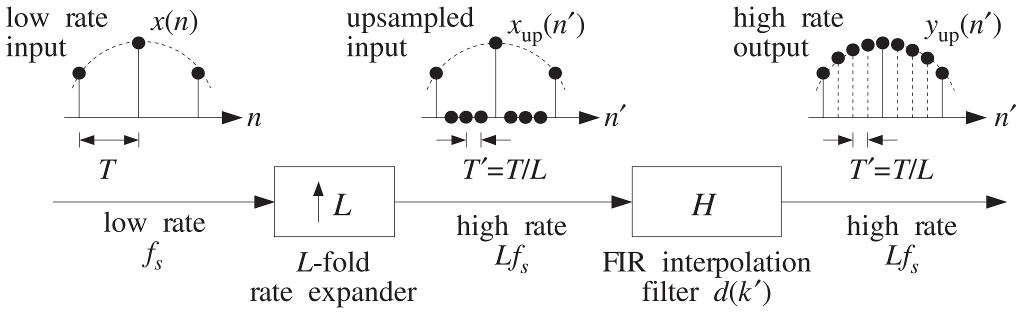
$$\boxed{\tan\left(\frac{\omega_{0b}}{2}\right)=\tan\left(\frac{\pi f_{0b}}{f_s}\right)=\frac{\sqrt{1+\Omega_0^2\,(1-c^2)}+1}{\Omega_0(1+c)}}$$

13 Interpolation, Dezimierung und Oversampling S.632

13.1 Interpolation und Oversampling

Ziel:
Ein Upsampling mit anschliessender Interpolation wird oft gemacht, um die Anforderungen an das analoge Anti-Image-Filter (D/A) zu verringern. Stattdessen wird jedoch ein digitales Interpolations-Filter mit hohen Anforderungen gebraucht.

13.1.1 Schritte im Zeitbereich



- Digitales Signal $x(n)$ um den Faktor L upsamplen, wobei alle neuen Samples auf Null gesetzt werden.

$$\boxed{x_{up}(n') = \begin{cases} x(n), & \text{wenn } n' = n\,L \\ 0, & \text{sonst} \end{cases}} \quad \text{mit} \quad \boxed{n' = n\,L + i} \quad i = 0, 1, 2, ..., L-1$$

- Das Upgesampletes Signal $x_{up}(n')$ wird durch ein FIR-Interpolations-Filter gelassen, welches die neuen Samples auf den richtigen Wert setzt.

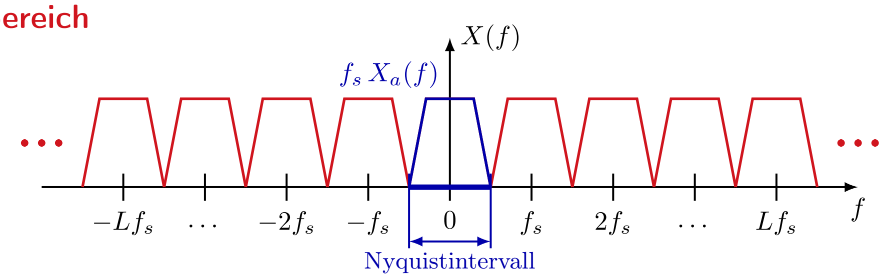
$$\boxed{y_{up}(n') = \sum_{k'} h(k')\,x_{up}(n'-k')}$$

- Es resultiert ein Signal $y_{up}(n')$ welches eine höhere Taktrate hat.

$$\boxed{f_s' = L \cdot f_s \quad \Leftrightarrow \quad T' = \frac{T}{L}}$$

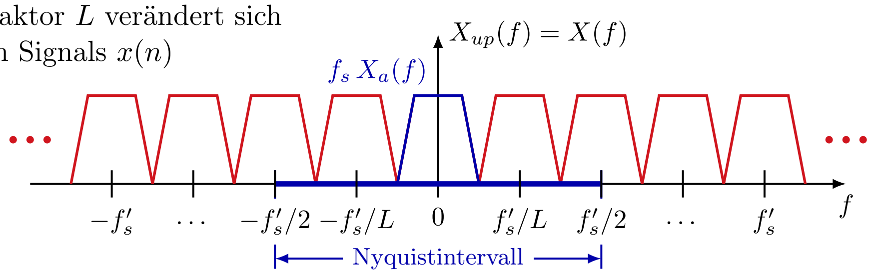
13.1.2 Schritte im Frequenzbereich

- Spektrum $X(f)$ des langsam abgetasteten Signal $x(n)$.



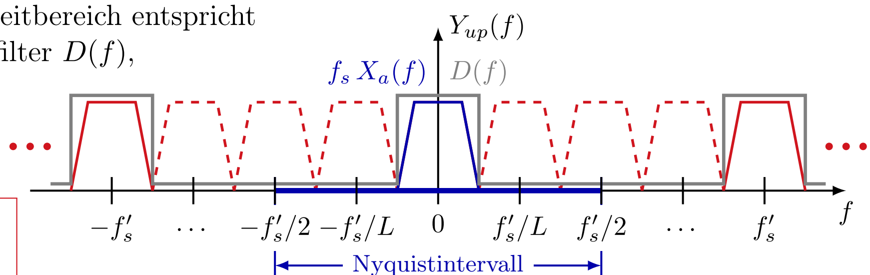
- Durch das Upsamplen um den Faktor L verändert sich das Spektrum $X(f)$ des digitalen Signals $x(n)$ nicht, da die neuen Samples alle Null sind. Es ändert nur die Samplingfrequenz $f_s \to f_s'$

$\Rightarrow$ $\boxed{X_{up}(f) = X(f)}$



- Eine perfekte Interpolation im Zeitbereich entspricht einem idealen digitalen Tiefpassfilter $D(f)$, welches die spektralen Kopien des Original-Spektrums im Nyquistband auslöscht.

$\Rightarrow$ Das analoge Anti-Image-Filter (Rekonstruktionsfilter) muss keine hohen Anforderungen bezüglich Steilheit erfüllen.



13.1.3 Tiefpass - Interpolationsfilter

- Das ideale Interpolationsfilter ist ein idealer Tiefpass mit

Frequenzgang:

$$\boxed{D(\omega') = \begin{cases} L, & -\pi/L \le \omega' \le \pi/L \\ 0, & \text{sonst} \end{cases}}$$

Grenzfrequenz:

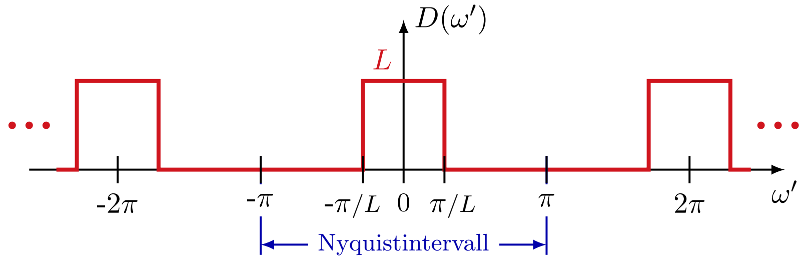
$$\boxed{f_c = \frac{f_s}{2} = \frac{f_s'}{2\,L}} \qquad \boxed{\omega_c' = \frac{2\pi f_c}{f_s'} = \frac{\pi}{L}}$$

Samplingfrequenz:

$$\boxed{f_s' = L\,f_s}$$

Impulsantwort:

$$\boxed{d(k') = \frac{1}{2\pi}\int_{-\pi}^{\pi} D(\omega')\, e^{j\omega' k'}\, d\omega' = \frac{\sin(\pi k'/L)}{\pi k'/L}}$$



- Durch das abschneiden der Impulsantwort $d(k')$ des Tiefpasses ein bei $\pm ML$ ergibt sich ein akausales FIR-Filter. Dieses FIR-Filter läuft auf der hohen Abtastfrequenz f_s'.

Akausales FIR-Filter: $$\boxed{y_{up}(n') = \sum_{k'=-ML}^{ML} d(k')\, x_{up}(n'-k')}$$

- Der Tiefpass kann kausal gemacht werden, indem er um ML Samples verzögert wird und eventuell noch mit einem Fenster $w(n')$ Multipliziert wird.

Kausales FIR-Filter: $$\boxed{h(n') = w(n')\, d(n'-ML)} \qquad n' = 0, 1, ..., N-1$$

$$\Rightarrow \quad \boxed{y_{up}(n') = \sum_{k'=0}^{N-1} w(k')\, d(k'-ML)\, x_{up}(n'-k')} \qquad \boxed{N = 2ML+1}$$

- Da das Upgesamplete Signal $x_{up}(n')$ sehr viele Nullen enthält, werden bei der Direktform sehr viele Multiplikationen mit Null durchgeführt. Wird das Filter jedoch in der Polyphasen-Form realisiert, so reduziert sich die Anzahl Multiplikationen drastisch.

Direktform-Filter: $\boxed{2ML \text{ - Multiplikationen pro Sample}}$

Polyphasen-From-Filter: $\boxed{2M \text{ - Multiplikationen pro Sample}}$

- Das Polyphasen-Form-Filter enthält (L-1)-Subfilter $d_i(k)$, welche auf der tiefen Abtastfrequenz f_s laufen.

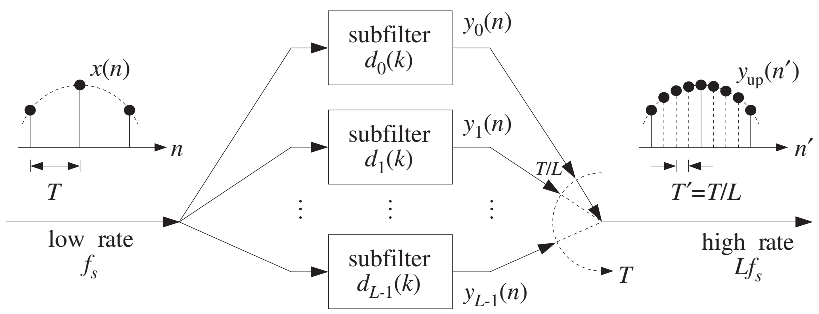
Subfilter-Koeffizienten: $$\boxed{d_i(k) = d(kL+i) = \frac{\sin(\pi(k+i/L))}{\pi(k+i/L)}} \qquad \begin{array}{l} -M \le k \le M-1 \\ i = 0, 1, ..., L-1 \end{array}$$

Die akausalen Ausgänge $y_i(n)$ der Subfilter sind damit:

Akausales Polyphasen-Subfilter: $$\boxed{y_i(n) = \sum_{k=-M}^{M-1} d_i(k)\, x(n-k)} \qquad i = 0, 1, ..., L-1$$

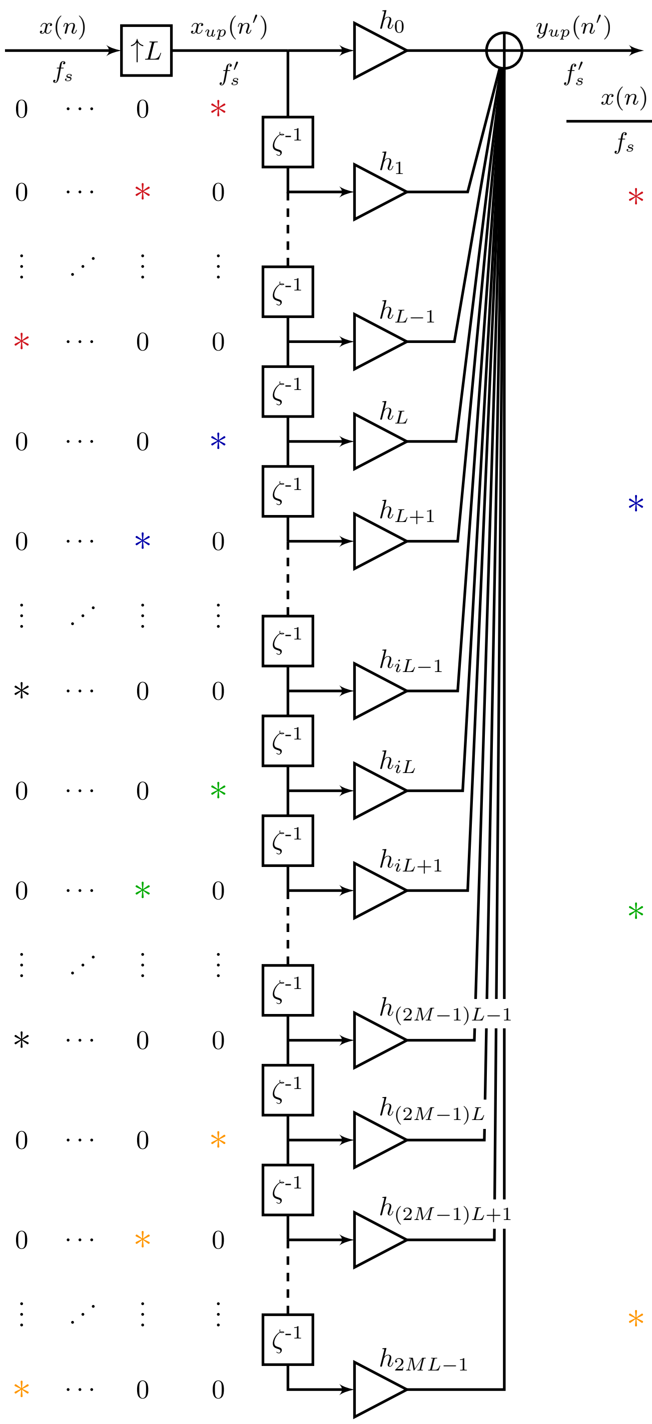
Diese Ausgänge $y_i(n)$ werden jeweils um $i\,T/L = i\,T'$ Samples verzögert.

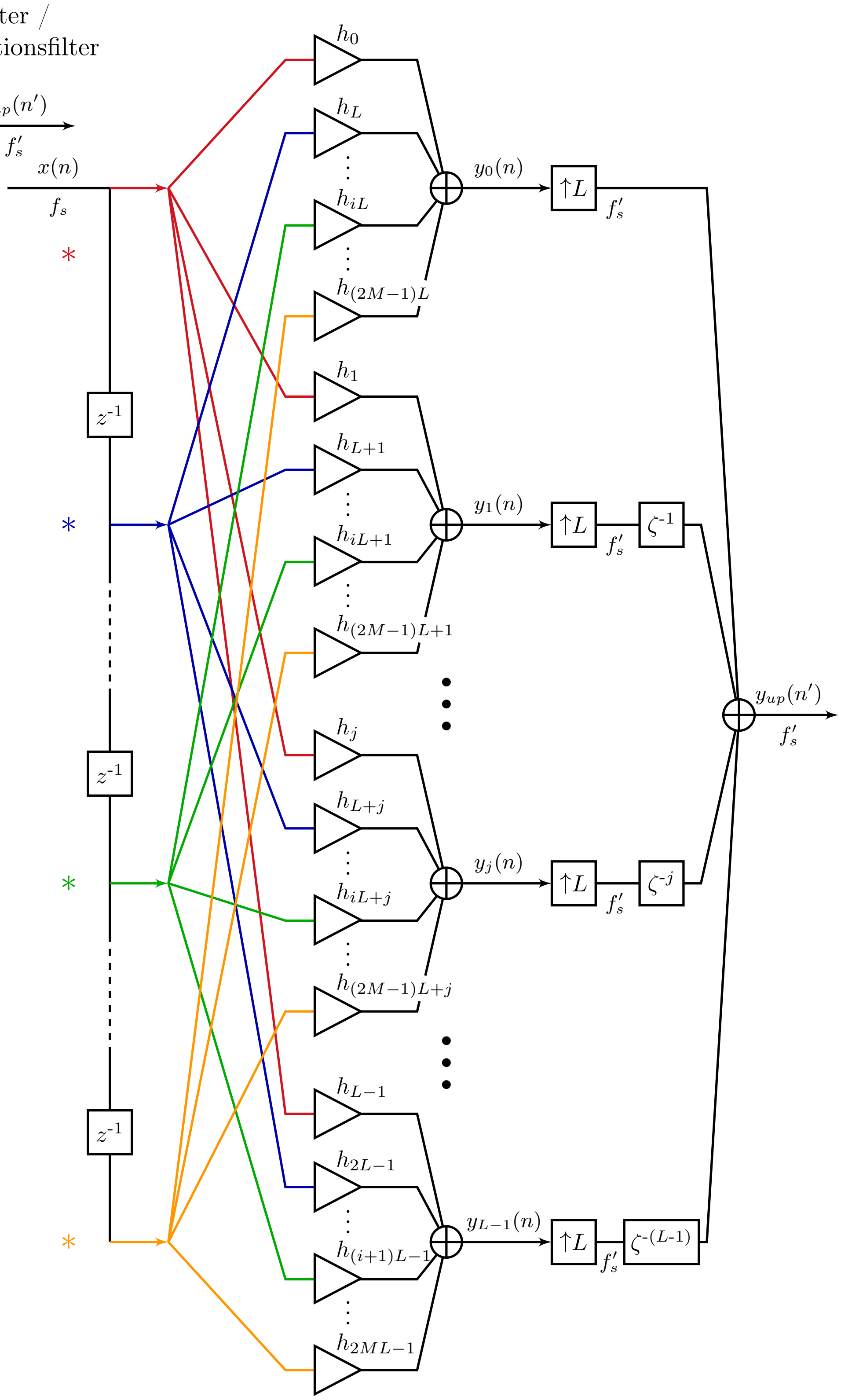
Verzögerte Ausgänge: $$\boxed{y_i(n) = y_{up}(nL+i)} \quad \Rightarrow \quad \boxed{y_{up}(n') = \sum_{i=0}^{L-1} y_{up}(nL+i) = \sum_{i=0}^{L-1} y_i(n)}$$



Der Filter-Koeffizient $d(ML)$ wird von keinem Subfilter $d_i(k)$ berechnet.
Dies ist jedoch nicht weiter tragisch, da $d(ML) = 0$ ist.

- Direktform FIR-Interpolationsfilter / Polyphasen-Form FIR-Interpolationsfilter





- Im Frequenzbereich kann sehr schön gezeigt werden, dass das Ausgangssignal $y_{up}(n')$ die Summe der Subfilter-Ausgänge $y_i(n)$, welche jeweils um $iT/L = iT'$ verzögert wurden, ist.

$$\boxed{T \rightarrow z^{-1} \quad \cup \quad T' \rightarrow \zeta^{-1}} \quad \Rightarrow \quad \boxed{z = \zeta^L, \quad \Leftrightarrow \quad \zeta = z^{1/L}}$$

$$\Rightarrow \quad \boxed{Y_{up}(\zeta) = \sum_{i=0}^{L-1} \zeta^{-i}\, Y_i(\zeta^L) = \sum_{i=0}^{L-1} z^{-i/L}\, Y_i(z)}$$

- Diese Beziehung gilt auch für das Tiefpass - Interpolationsfilter $d(k')$ und die subgesampelten Subfilter $d_i(k)$.

$$\Rightarrow \quad \boxed{D(\zeta) = \sum_{i=0}^{L-1} \zeta^{-i}\, D_i(\zeta^L) = \sum_{i=0}^{L-1} z^{-i/L}\, D_i(z)}$$

- Das Polyphasen-Tiefpass-Subfilter kann kausal gemacht werden, indem es um M Samples verzögert wird und eventuell noch mit einem subgesampelten Fenster $w_i(n)$ Multipliziert wird.

  Kausales Polyphasen-Subfilter:

$$\boxed{h_i(n) = w_i(n)\, d_i(n-M) = w(nL+i)\, d(nL+i-ML)} \qquad \begin{aligned} n &= 0,1,...,2M-1 \\ i &= 0,1,...,L-1 \end{aligned}$$

$$\Rightarrow \quad \boxed{y_i(n) = \sum_{k=0}^{2M-1} h_i(k)\, x(n-k) = \sum_{k=0}^{2M-1} w(kL+i)\, d(kL+i-ML)\, x(n-k)} \qquad i = 0,1,...,L-1$$

- Dies führt führt im Frequenzbereich zu folgenden Beziehungen (ohne Fenster $w_i(n)$):

$$\boxed{H_i(z) = z^{-M}\, D_i(z)} \qquad \boxed{Y_i(z) = H_i(z)\, X(z) = z^{-M}\, D_i(z)\, X(z)}$$

13.1.4 Herleitung des Idealen Interpolationsfilter

- Im Frequenzbereich gelten folgende Beziehungen:

Analoges Signal:	$x_a(t)$	$\circ\!\!-\!\!\bullet$	$X_a(f)$
langsam abgetastetes Signal:	$x(n) = x_a(nT)$	$\circ\!\!-\!\!\bullet$	$\boxed{X(f) = \frac{1}{T}\sum_{m=-\infty}^{\infty} X_a(f - m f_s)}$
Upgesampletes Signal:	$x_{up}(n)$	$\circ\!\!-\!\!\bullet$	$\boxed{X_{up}(f) = X(f) = \frac{1}{T}\sum_{m=-\infty}^{\infty} X_a(f - m f_s)}$
schnell abgetastetes Signal:	$x'(n') = x_a(n'T')$	$\circ\!\!-\!\!\bullet$	$\boxed{X'(f) = \frac{1}{T'}\sum_{m'=-\infty}^{\infty} X_a(f - m' f_s')}$

- Daraus ergibt sich für den Frequenzgang und die z-Transformation:

  Frequenzgang: $\boxed{X_{up}(\omega') = X(\omega) = X(\omega' L)}$ mit $\boxed{\omega = \frac{2\pi f}{f_s}, \quad \omega' = \frac{2\pi f}{f_s'} \quad \Rightarrow \quad \omega = \omega' L}$

  z-Transformation: $\boxed{X_{up}(\zeta) = X(z) = X(\zeta^L)}$ mit $\boxed{z = e^{j\omega}, \quad \zeta = e^{j\omega'} \quad \Rightarrow \quad z = \zeta^L}$

- Das Ziel des Interpolationsfilter $D(f)$ ist:

$$\boxed{X'(f) \overset{!}{=} Y_{up}(f) = D(f)\, X_{up}(f) = D(f)\, X(f)} \quad \Rightarrow \quad \boxed{X'(f) = D(f)\, X(f)}$$

- Daraus folgt für das Interpolationsfilter $D(f)$:

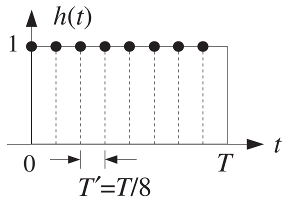
$$\begin{aligned} \overbrace{\frac{1}{T'}\sum_{m'=-\infty}^{\infty} X_a(f - m' f_s')}^{X'(f)} &= D(f)\, \overbrace{\frac{1}{T}\sum_{m=-\infty}^{\infty} X_a(f - m f_s)}^{X(f)} \\ \frac{1}{T'} X_a(f) + \text{replicas} &= \underbrace{D(f)\, \frac{1}{T} X_a(f)}_{\text{Passband}} + \underbrace{D(f)\, \frac{1}{T}\sum_{m=1}^{L-1} X_a(f - m f_s)\, +\text{replicas}}_{\text{Stoppband}} \end{aligned}$$

$$\Rightarrow \quad \boxed{D(f) = \begin{cases} T/T' = L, & -f_s/2 \le f \le f_s/2 \\ 0, & \text{sonst} \end{cases}}$$

13.1.5 Einfache Interpolationsfilter

Hold-Interpolator

Impulsantwort: $\boxed{h(t) = \begin{cases} 1, & 0 \leq t < T \\ 0, & \text{sonst} \end{cases}}$



Digitale Impulsantwort:

$$\boxed{d(k') = h(k'T') = \begin{cases} 1, & 0 \leq k' \leq L-1 \\ 0, & \text{sonst} \end{cases}}$$

Polyphasen-Subfilter-Koeffizienten:

$\boxed{d_i(k) = d(kL+i)}$ $\quad i = 0, 1, ..., L-1$

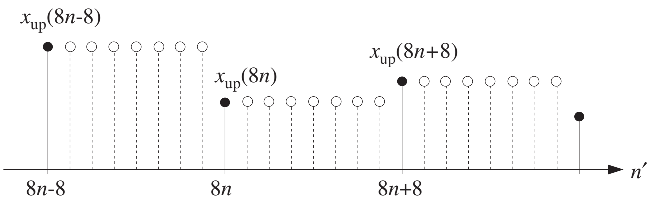
$$\Rightarrow \boxed{\begin{aligned} d_i(k) &= d_i(0)\,\delta(k) \\ &= d(i)\,\delta(k) \\ &= \delta(k) \end{aligned}}$$

Polyphasen-Subfilter-Ausgänge:

$$\boxed{y_i(n) = \sum_k d_i(k)\,x(n-k) = x(n)}$$

$i = 0, 1, ..., L-1$

$$\Rightarrow \boxed{y_{up}(nL+i) = y_i(n)}$$



Übertragungsfunktion:

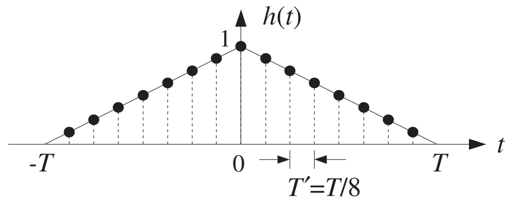
$$\boxed{D(\zeta) = \sum_{i=0}^{L-1} \zeta^{-i}\, \underbrace{D_i(\zeta^L)}_{1} = \frac{1-\zeta^{-L}}{1-\zeta^{-1}}}$$

Frequenzgang:

$$\boxed{D(f) = \frac{\sin(\pi f/f_s)}{\sin(\pi f/(Lf_s))}\, e^{-j\pi(L-1)f/(Lf_s)}}$$

Linear-Interpolator

Impulsantwort: $\boxed{h(t) = \begin{cases} 1-|t|/T, & |t| \leq T \\ 0, & \text{sonst} \end{cases}}$



Digitale Impulsantwort:

$$\boxed{d(k') = h(k'T') = \begin{cases} 1-|k'|/L, & |k'| \leq L-1 \\ 0, & \text{sonst} \end{cases}}$$

Polyphasen-Subfilter-Koeffizienten:

$\boxed{d_i(k) = d(kL+i)}$ $\quad i = 0, 1, ..., L-1$

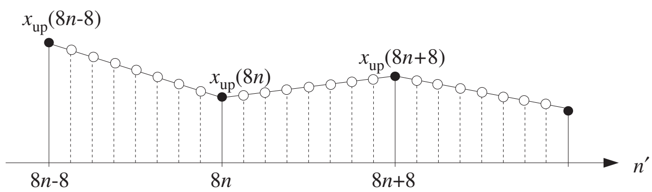
$$\Rightarrow \boxed{\begin{aligned} d_i(k) &= d_i(0)\,\delta(k) + d_i(-1)\,\delta(k+1) \\ &= d(i)\,\delta(k) + d(-L+i)\,\delta(k+1) \\ &= \left(1-\frac{i}{L}\right)\,\delta(k) + \frac{i}{L}\,\delta(k+1) \end{aligned}}$$

Polyphasen-Subfilter-Ausgänge:

$$\boxed{\begin{aligned} y_i(n) &= \sum_k d_i(k)\,x(n-k) \\ &= \left(1-\frac{i}{L}\right)\,x(n) + \frac{i}{L}\,x(n+1) \end{aligned}}$$

$i = 0, 1, ..., L-1$

$$\Rightarrow \boxed{y_{up}(nL+i) = y_i(n)}$$

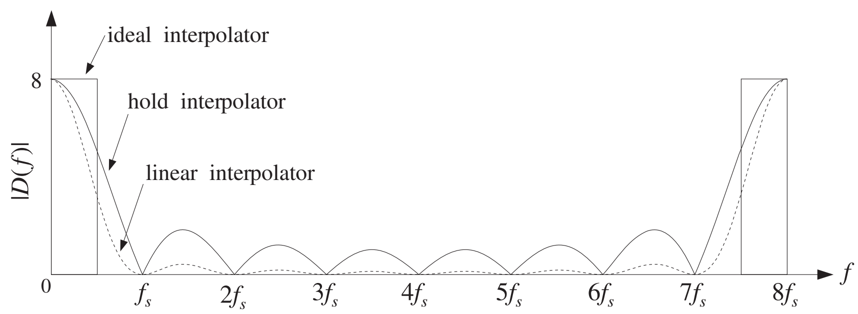


Übertragungsfunktion

$$\boxed{D(\zeta) = \sum_{i=0}^{L-1} \zeta^{-i}\, \underbrace{D_i(\zeta^L)}_{1+i/L(\zeta^L-1)} = \frac{1}{L}\left(\frac{1-\zeta^{-L}}{1-\zeta^{-1}}\right)^2 \zeta^{L-1}}$$

Frequenzgang

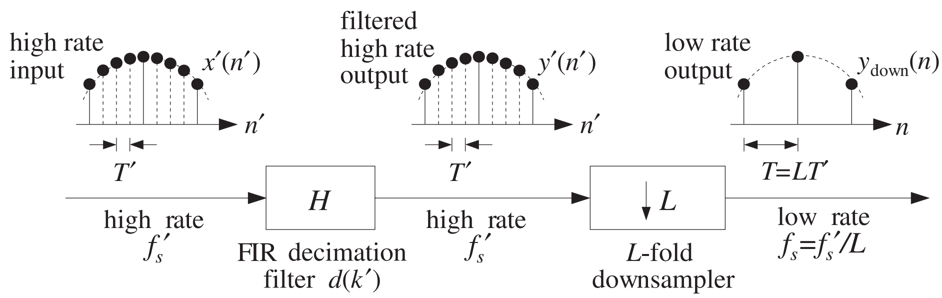
$$\boxed{D(f) = \frac{1}{L}\left|\frac{\sin(\pi f/f_s)}{\sin(\pi f/(Lf_s))}\right|^2}$$



13.2 Dezimierung und Downsampling

Ziel:
Eine Dezimierung mit anschliessendem Downsampling wird oft verwendet, um die Anforderungen an das analoge Anti-Aliasing-Filter (A/D) zu verringern. Stattdessen muss jedoch schneller Abgetastet werden, als eigentlich notwendig wäre und es wird ein digitales Dezimierungs-Filter mit hohen Anforderungen gebraucht.

13.2.1 Schritte im Zeitbereich



- Das schnell abgetastete Signal $x'(n')$ mit einem digitalen Dezimierungsfilter auf das Nyquistband der tiefen Abtastfrequenz $[-f_s/2, f_s/2]$ begrenzen.

$$\boxed{y'(n') = \sum_{k'} h(k')\, x'(n'-k')}$$

- Das bandbegrenzte Signal $y'(n')$ um den Faktor L downsamplen, d.h. nur jedes L-te Sample behalten.

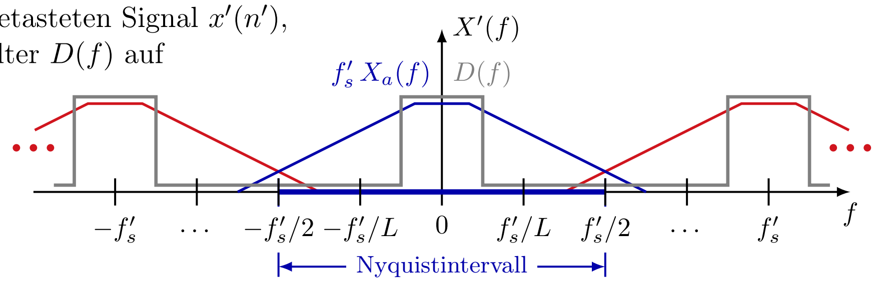
$$\boxed{y_{down}(n) = y'(n')\big|_{n'=nL} = y'(nL)}$$

- Es resultiert ein Signal $y_{down}(n)$ welches eine tiefere Taktrate hat.

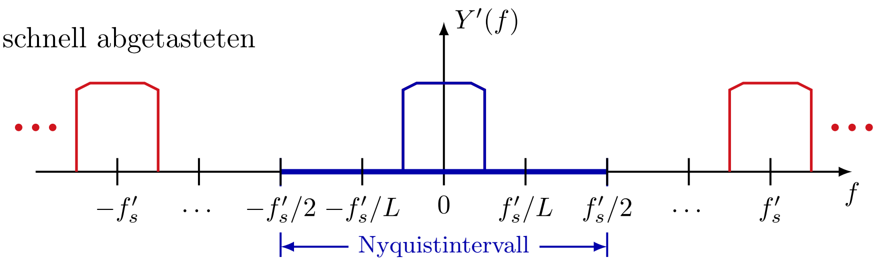
$$\boxed{f_s = \frac{f_s'}{L} \quad \Leftrightarrow \quad T = T' \cdot L}$$

13.2.2 Schritte im Frequenzbereich

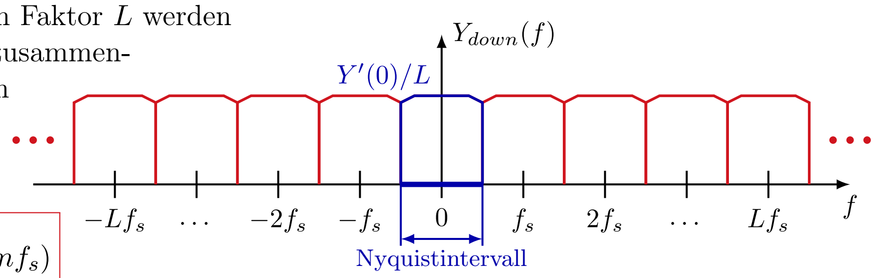
- Spektrum $X'(f)$ des schnell abgetasteten Signal $x'(n')$, welches mit dem Dezimierungsfilter $D(f)$ auf das Nyquistband der tiefen Abtastfrequenz $[-f_s/2, f_s/2]$ begrenzt wird.



- Spektrum $Y'(f)$ des gefilterten, schnell abgetasteten Ausgangssignal $y'(n')$



- Durch das Downsamplen um den Faktor L werden die Teil-Spektrum $Y'(f)$ näher zusammengeschoben bzw. L-mal periodisch widerhohlt.



$$\Rightarrow \quad \boxed{Y_{down}(f) = \frac{1}{L}\sum_{m=0}^{L-1} Y'(f - m f_s)}$$

13.2.3 Tiefpass - Dezimierungsfilter

- Das ideale Dezimierungsfilter ist ein idealer Tiefpass und fast identisch mit dem Interpolationsfilter.

Frequenzgang:

$$\boxed{D(\omega') = \begin{cases} 1, & -\pi/L \le \omega' \le \pi/L \\ 0, & \text{sonst} \end{cases}}$$

Grenzfrequenz:

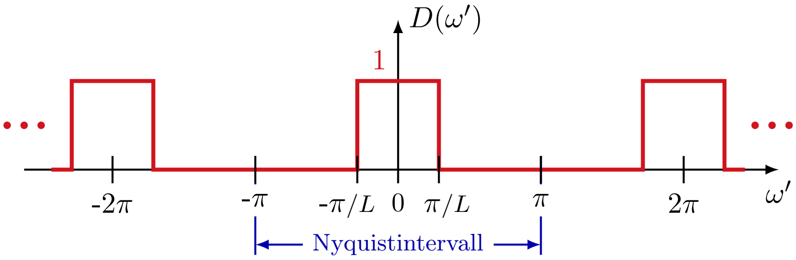
$$\boxed{f_c = \frac{f_s}{2} = \frac{f_s'}{2\,L}} \qquad \boxed{\omega_c' = \frac{2\pi f_c}{f_s'} = \frac{\pi}{L}}$$

Samplingfrequenz:

$$\boxed{f_s' = L\,f_s}$$

Impulsantwort:

$$\boxed{d(k') = \frac{1}{2\pi}\int_{-\pi}^{\pi} D(\omega')\,\mathrm{e}^{j\omega' k'}\,d\omega' = \frac{\sin(\pi k'/L)}{\pi k'}}$$



- Durch das abschneiden der Impulsantwort $d(k')$ des Tiefpasses ein bei $\pm ML$ ergibt sich ein akausales FIR-Filter. Dieses FIR-Filter läuft auf der hohen Abtastfrequenz f_s'.

Akausales FIR-Filter:

$$\boxed{y_{down}(n) = y'(nL) = \sum_{k'=-ML}^{ML} d(k')\,x'(nL-k')}$$

- Der Tiefpass kann kausal gemacht werden, indem er um ML Samples verzögert wird und eventuell noch mit einem Fenster $w(n')$ Multipliziert wird.

Kausales FIR-Filter: $\boxed{h(n') = w(n')\,d(n'-ML)}$ $\quad n' = 0, 1, ..., N-1$

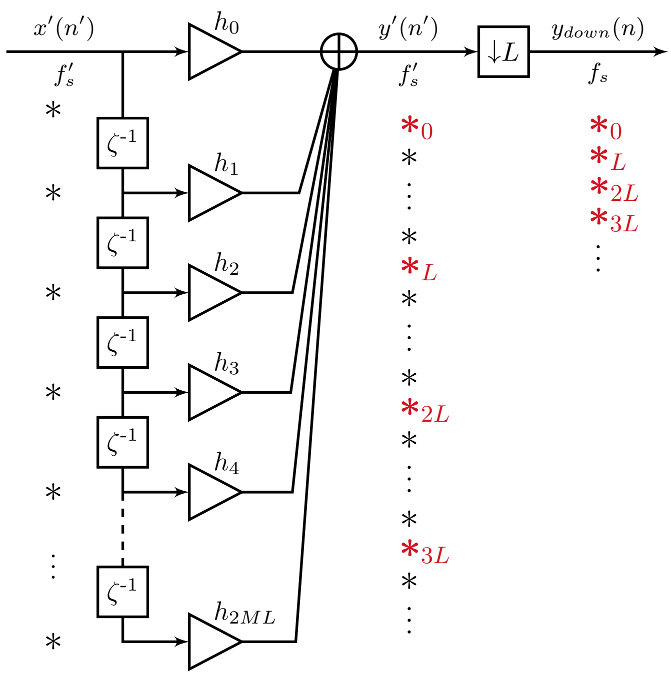
$$\Rightarrow \quad \boxed{y_{down}(n) = y'(nL) = \sum_{k'=0}^{N-1} w(k')\,d(k'-ML)\,x'(nL-k')} \qquad \boxed{N = 2ML+1}$$

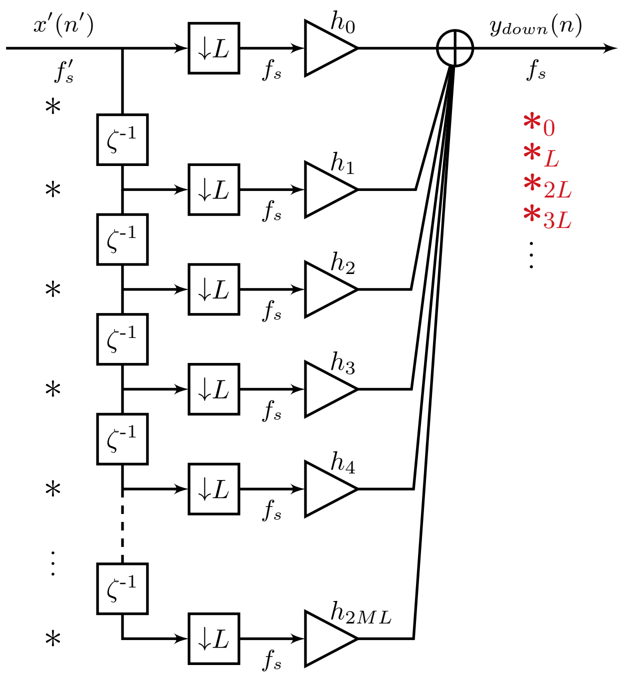
- Da nur jedes L-te Sample verwendet wird, ist es nicht sinnvoll jedes Sample der hohen Abtastfrequenz $y'(n')$ zu berechnen. Damit nur die Samples $y'(nL)$ berechnet werden, muss das Eingangssignal $x'(n')$ vor jedem Filterkoeffizienten downgesampelt werden, womit sich der Rechenaufwand stark reduziert.

Normales-Filter: $\boxed{2ML^2 \text{ - Multiplikationen pro Sample}}$

Downgesampeltes-Filter: $\boxed{2ML \text{ - Multiplikationen pro Sample}}$

- Normales-Filter / Downgesampeltes-Filter





13.2.4 Einfaches Dezimierungsfilter

Ein sehr einfaches Dezimierungsfilter ist, L-schnelle Samples zu einem langsamen Sample zu mitteln.

Übertragungsfunktion:

$$\boxed{H(\zeta) = \frac{1}{L}\left[1 + \zeta^{-1} + ... + \zeta^{-(L-1)}\right] = \frac{1}{L}\frac{1-\zeta^{-L}}{1-\zeta^{-1}}}$$

Ausgangssignal:

$$\boxed{y_{down}(n) = \frac{x'(nL) + x'(nL-1) + ... + x'(nL-(L-1))}{L}}$$

13.3 Noise-Shaping Quantisierer

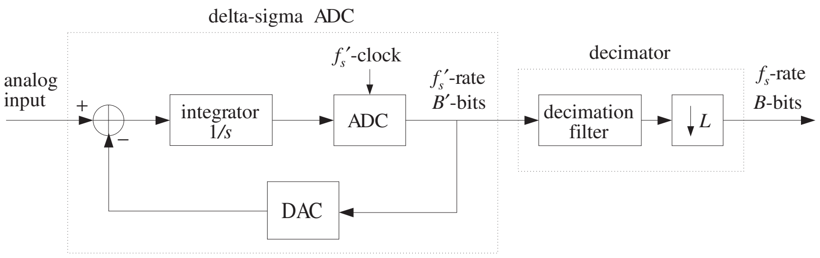
Die Idee von Noise-Shaping ist, das Quantisierungsrauschen umzuverteilen, sodass im Frequenzband $[-\frac{f_s}{2}, \frac{f_s}{2}]$ des analogen Eingangssignal $x_a(t)$ bzw. des dezimierten und downgesampelten Signals $y_{down}(n)$ möglichst wenig Rauschleistung vorhanden ist.

$\Rightarrow$ Rauschleistung in hohe Frequenzen verschieben, wo keine Signalleistung mehr vorhanden ist, damit das Rauschen anschliessend herausgefiltert werden kann.

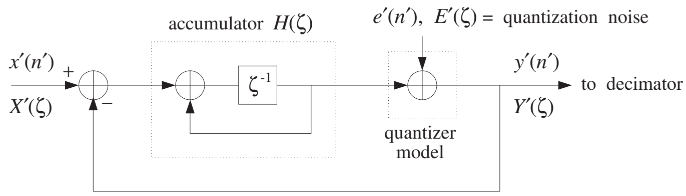
Dies wird bspw. beim Delta-Sigma-Wandler gemacht, indem ein Feedback-Pfad eingeführt wird.

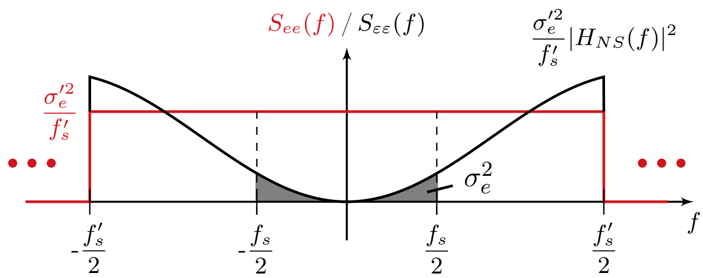
- Differenz Δ zwischen analogem Eingangssignal und dem digitalisierten Signal wird aufsummiert Σ
- Integrator (Summierer) kann ...
  ... langsamen Signalen gut folgen $\Rightarrow$ kleiner Quantisierungsfeher.
  ... schnellen Signalen schlecht folgen $\Rightarrow$ grosser Quantisierungsfeher.

  $\Rightarrow$ Integrator muss Frequenzen des analogen Signals $[-f_s/2 \leq f \leq f_s/2]$ gut folgen können!



Stochastisches, lineares diskretes Modell





- Übertragungsfunktion des Summierers: $\boxed{H(\zeta) = \frac{\zeta^{-1}}{1-\zeta^{-1}}}$
- Ausgangssignale des Delta-Sigma-Wandlers:

$$H(\zeta)(X'(\zeta) - Y'(\zeta)) + E'(\zeta) = Y'(\zeta) \quad \Rightarrow \quad Y'(\zeta) = \overbrace{\frac{H(\zeta)}{1+H(\zeta)}}^{\zeta^{-1}} X'(\zeta) + \overbrace{\frac{1}{1+H(\zeta)}}^{1-\zeta^{-1}} E'(\zeta)$$

$$\Rightarrow \quad \boxed{Y'(\zeta) = H_X(\zeta)\, X'(\zeta) + H_{NS}(\zeta)\, E'(\zeta)} \quad \text{mit} \quad \boxed{H_X(\zeta) = \zeta^{-1}, \qquad H_{NS}(\zeta) = 1 - \zeta^{-1}}$$

$$\Rightarrow \quad \boxed{y'(n') = x'(n'-1) + \varepsilon(n')} \quad \text{mit} \quad \boxed{\varepsilon(n') = e(n') - e(n'-1) \quad \Leftrightarrow \quad \mathcal{E}(\zeta) = (1-\zeta^{-1})E(\zeta)}$$

- Leistung des Quantisierungsrauschen nach dem Noise-Shaping:

$$\boxed{\sigma_e^2 = \frac{\sigma_e'^2}{f_s'} \int_{-f_s/2}^{f_s/2} |H_{NS}(f)|^2\, df} \qquad \boxed{|H_{NS}(f)|^2 = \left|2 \sin\left(\frac{\pi f}{f_s'}\right)\right|^{2p} \quad \text{für } -\frac{f_s'}{2} \leq f \leq \frac{f_s'}{2}} \qquad p: \text{ Ordnung des Noise-Shapers}$$

- Bitgewinn durch das Noise-Shaping (Herleitung siehe Kapitel 2.3)

$$\boxed{\Delta B = (p + 0.5)\log_2(L) - 0.5\log_2\left(\frac{\pi^{2p}}{2p+1}\right)}$$

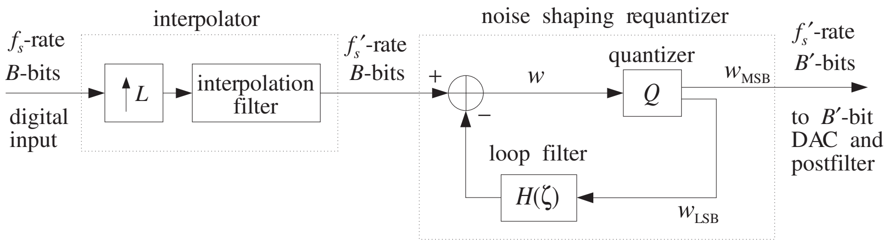
13.3.1 Noise-Shaping Requantisierer

Die Idee des Noise-Shaping kann auch bei Digital-Analog-Wandlern angewendet werden. Das Ziel hierbei ist mit weniger Bits $B' < B$ des DAC ein qualitativ gleichwertiges analoges Ausgangssignal $x_a(t)$ zu erzeugen.

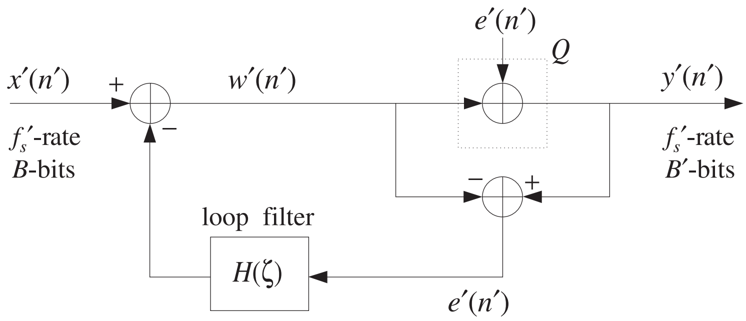
Dazu wird folgendes gemacht.

- Der Quantisierer teilt das B-bit-Wort lange in zwei Wörter auf:
  - w_{MSB}: B' - most-significant-Bits werden zum DAC weitergeleitet.
  - w_{LSB}: $(B - B')$ - least-significant-Bits (Rausch-Bits) werden durch das Loop-Filter geschickt.
- Die Rausch-Bits (least-significant-Bits) werden vom Eingangssignal abgezogen, wodurch sie Hochpass gefiltert werden.

  $\Rightarrow$ $\boxed{\text{Das Rauschen wird Hochpass gefiltert und dadurch im tieffrequenten Band reduziert.}}$



Stochastisches, lineares diskretes Modell



- Ausgangssignale des Noise-Shaping Requantisierers:

$$Y'(\zeta) = W'(\zeta) + E'(\zeta) \qquad \text{und} \qquad W'(\zeta) = X'(\zeta) - H(\zeta)\,E'(\zeta)$$

$$\Rightarrow \qquad Y'(\zeta) = X'(\zeta) + (1 - H(\zeta))\,E'(\zeta)$$

$$\Rightarrow \qquad \boxed{Y'(\zeta) = X'(\zeta) + H_{NS}(\zeta)\,E'(\zeta)} \qquad \text{mit} \qquad \boxed{H_{NS}(\zeta) = 1 - H(\zeta)}$$

- Noise-Shape-Filter in Abhängigkeit des Loop-Filters

  1.Ordnung: $\boxed{H(\zeta) = \zeta^{-1}}$ $\Rightarrow$ $\boxed{H_{NS}(\zeta) = 1 - \zeta^{-1}}$

  2.Ordnung: $\boxed{H(\zeta) = 2\zeta^{-1} - \zeta^{-2}}$ $\Rightarrow$ $\boxed{H_{NS}(\zeta) = (1 - \zeta^{-1})^2}$

- Bitgewinn durch das Noise-Shaping

$$\boxed{\Delta B = (p + 0.5)\,\log_2(L) - 0.5\,\log_2\left(\frac{\pi^{2p}}{2p+1}\right)}$$