Digitale Signalverarbeitung auf FPGAs

Kap. 2 – Zeitdiskrete Signale und Systeme im Frequenzbereich

2016

Dr. Christian Münker

Fragen, Anmerkungen, Bugs bitte an mail@chipmuenk.de - viel Erfolg und Spaß!

Überblick Kapitel 2

- $H(z)$, $|H(z)|$, $H(e^{j\Omega})$ und $H(\Omega)$ ...
- Schreibformen von $H(z)$
- Pol / Nullstellendiagramm
- Kausalität
- Matlab / Python – Darstellung von $H(z)$
- Einfache Filter

Digitale Signalverarbeitung auf FPGAs

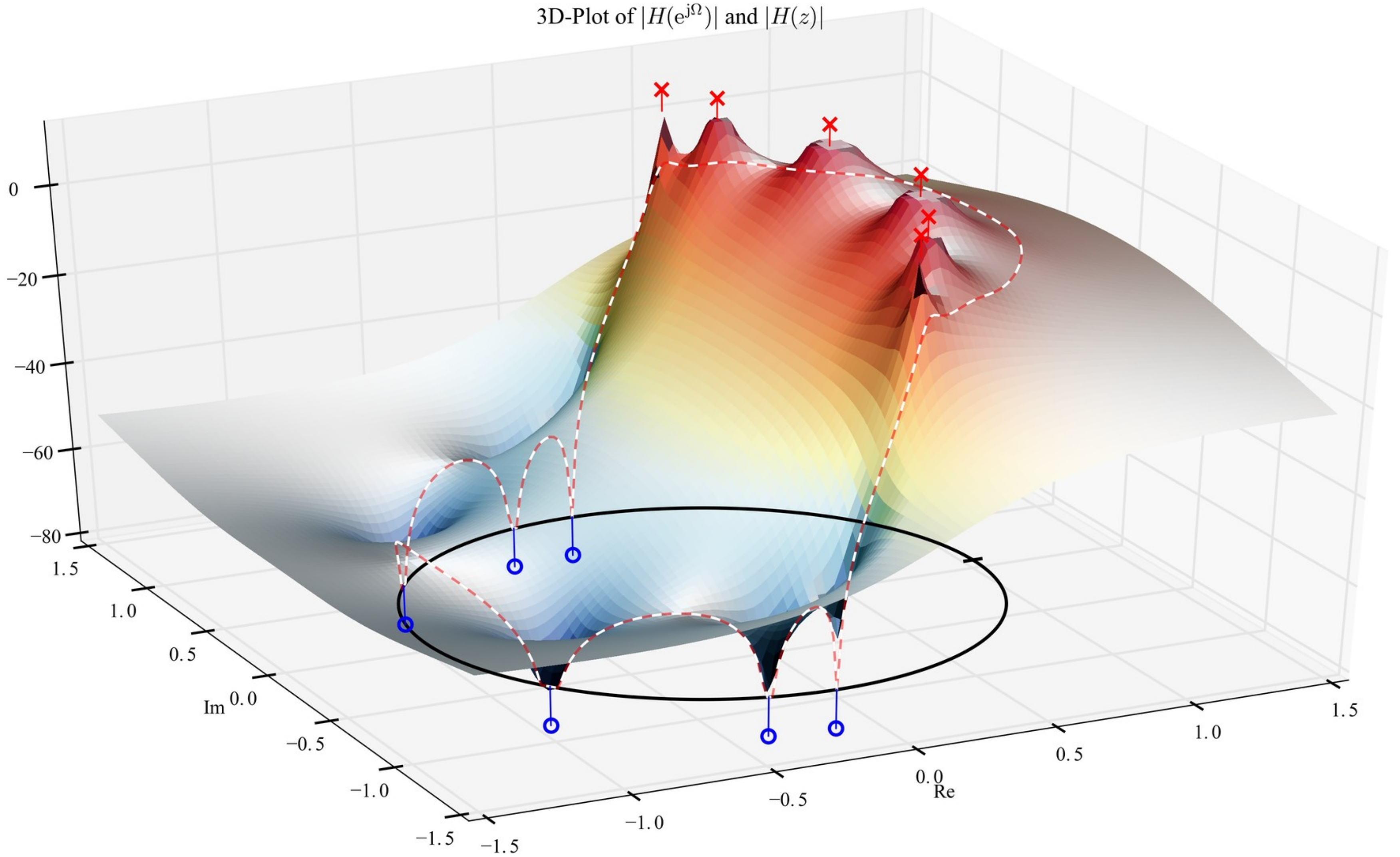
Kap. 2 Zeitdiskrete Signale und LTI-Systeme im Frequenzbereich

Teil 1 $H(z)$, $|H(z)|$, $H(e^{j\Omega})$ und $H(\Omega)$ ...

2016

Dr. Christian Münker

$H(z)$, $|H(z)|$, $H(\mathrm{e}^{\mathrm{j}\Omega})$ und $H(\Omega)$ ...



Zeitdiskrete Systeme im Frequenzbereich

Zeitdiskrete Signale und LTI-Systeme lassen sich auf verschiedene Arten darstellen (vgl. zeitkontinuierliche Signale und LTI-Systeme):

Zeitbereich: Folge + Impulsantwort $h[k]$ $\rightarrow$ **Kapitel 1**

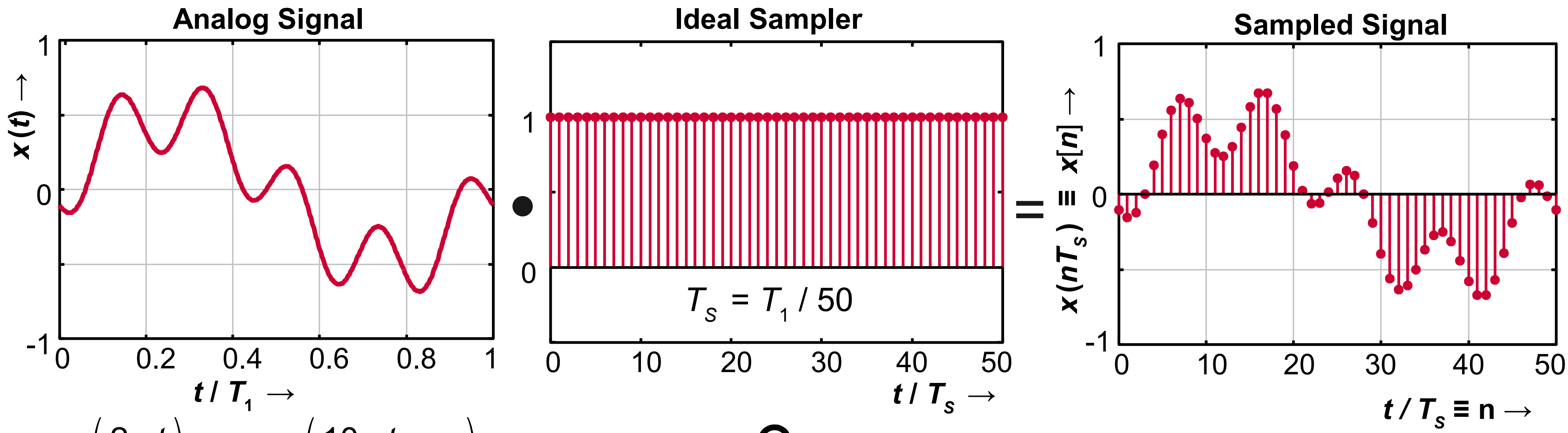
$\Updownarrow$

Komplexe *z*-Ebene: Systemfunktion $H(z)$, $|H(z)|$ $\rightarrow$ **Kapitel 1 + 2**

$\Updownarrow$

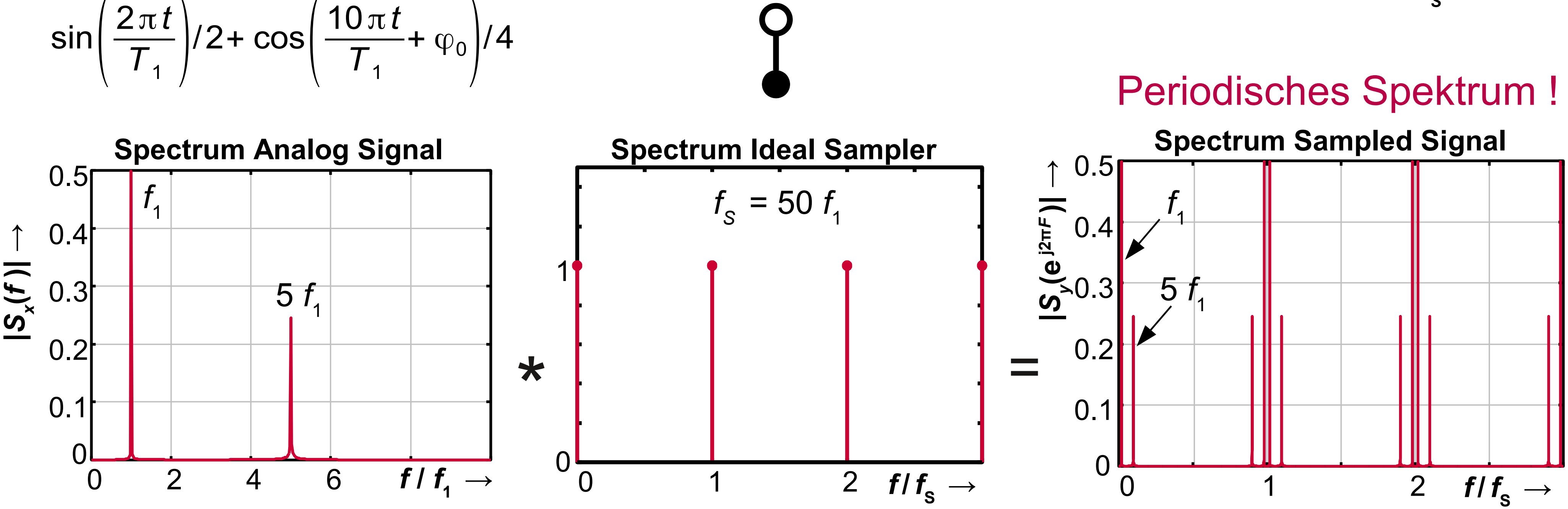
Frequenzbereich: Komplexer Frequenzgang $\underline{H}(e^{j\Omega})$ $\rightarrow$ **Kapitel 2**

Warum $\underline{H}(e^{j\Omega})$ und nicht $\underline{H}(\Omega)$?



$$\sin\left(\frac{2\pi t}{T_1}\right)/2+\cos\left(\frac{10\pi t}{T_1}+\varphi_0\right)/4$$

Periodisches Spektrum !



Frequenzdarstellung - Einheiten

Beispiel: f = 1 kHz (Perioden / s), f_S = 20 kHz (Samples / s)

Winkelfrequenz $\boldsymbol{\omega} = 2\pi f$ = 6,28 k rad / s

Normierte Frequenz $\boldsymbol{F} = f / f_S$ = 0,05 Perioden / Sample

Norm. Winkelfrequenz $\boldsymbol{\Omega} = 2\pi\ f / f_S$ = 0,314 rad / Sample

= 0,1 π rad / Sample

Achtung!

- Matlab bezieht normierte Frequenzen auf die Nyquistfrequenz $f_S/2$!
- Ω und ω werden in der Literatur nicht einheitlich verwendet!

Frequenzgang in der z-Ebene

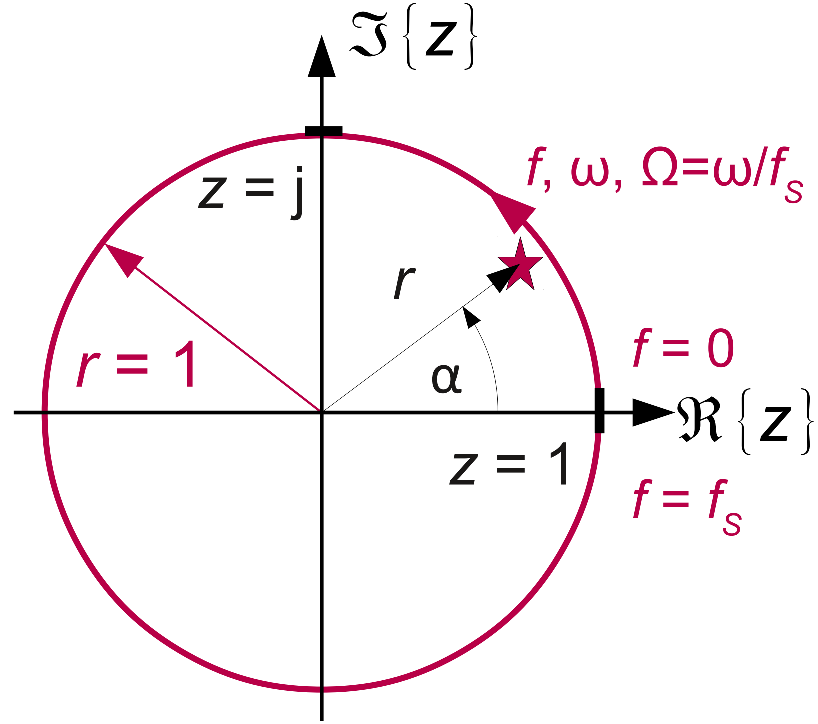
Beliebiger Punkt ★ in z-Ebene:

$$z = e^{sT_S} = e^{(\sigma + j\omega)T_S} = e^{\sigma T_S} \mathbf{e}^{\mathbf{j\omega T_S}} = r\,e^{j\alpha}$$
$$= x + jy = r(\cos\alpha + j\sin\alpha)$$

Für $\sigma = 0$, $r = 1$ Frequenzachse entlang des Einheitskreises (EK):

$$\Rightarrow \quad z = e^{j\omega T_S} = e^{j2\pi f / f_S}$$

$$\Rightarrow \quad H(f) = H\left(z = e^{j2\pi f/f_S} = e^{j\omega/f_S}\right)$$



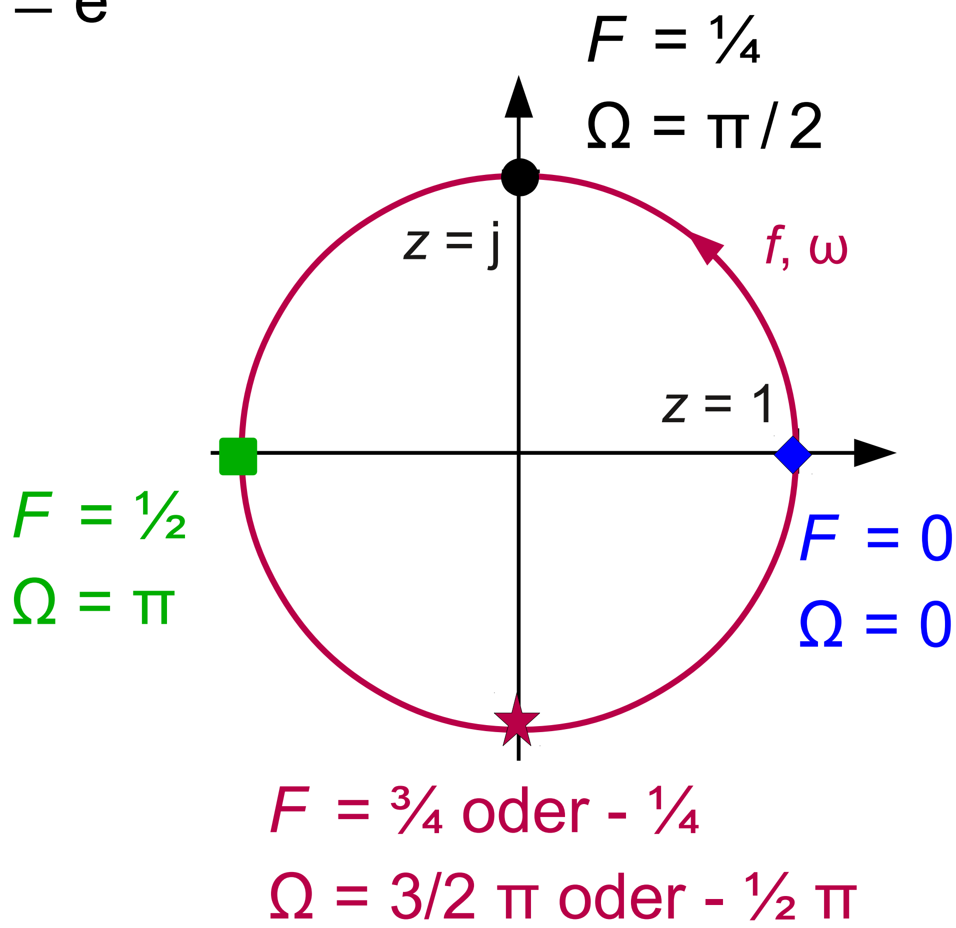
$$z = e^{j2\pi f/f_S} = e^{j2\pi(f + k f_S)/f_S} \quad \Rightarrow H(f) = H(f + k f_S);\; k = \ldots, -1, 0, 1, \ldots$$

Zeitdiskrete Spektren bzw. Frequenzgänge werden entlang des Einheitskreises abgelesen und sind periodisch mit f_S !

Schreibweise: $\boldsymbol{H(z = e^{j2\pi f T_S}) = H(e^{j2\pi F}) = H(e^{j\Omega})}$

Normierte Frequenzen auf dem EK

$$r = 1 \;\Rightarrow\; z = e^{j\omega T_S} = e^{j\Omega} = e^{j2\pi f/f_S} = e^{j2\pi F}$$



Normierte Frequenz:

$F = f\,/\,f_S = 0 \ldots 1$ (oder ± 0,5)

Normierte Kreisfrequenz:

$\Omega = \omega\,/\,f_S = 0 \ldots 2\pi$ (oder $\pm\,\pi$)

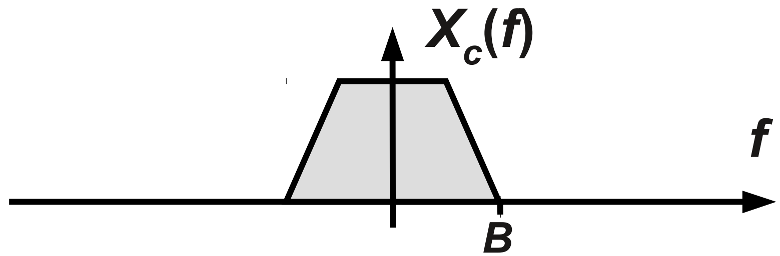
Wichtige Punkte auf dem EK:

- ◆ $z = 1 \;\rightarrow\; f = 0$ oder f_S
- ● $z = j \;\rightarrow\; f = f_S\,/\,4$
- ■ $z = -1 \;\rightarrow\; f = f_S\,/\,2$
- ☆ $z = -j \;\rightarrow\; f = \frac{3}{4}\,f_S$ oder $-f_S\,/\,4$

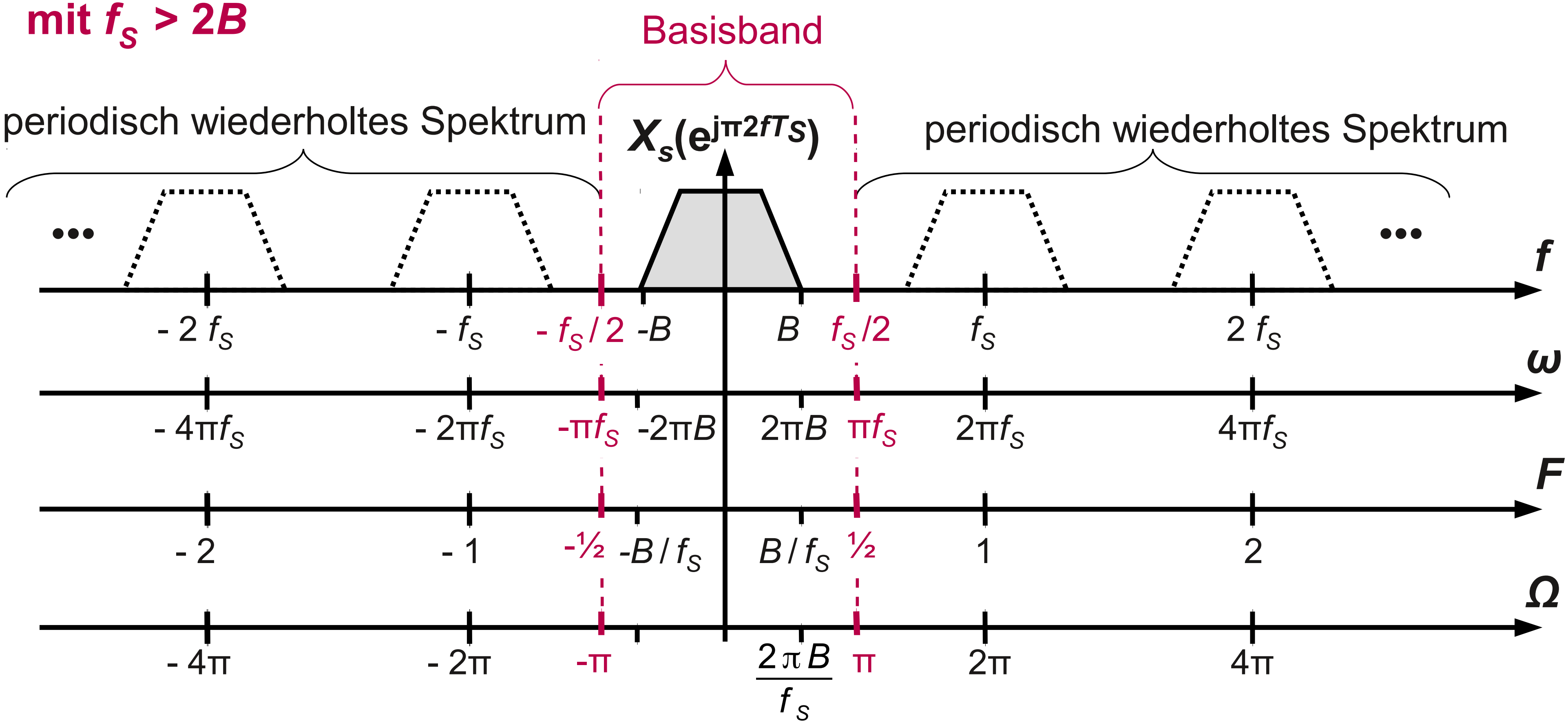
Achtung: manchmal auch Normierung auf $f_S\,/\,2$ (Matlab) !

Frequenzdarstellung periodischer Spektren

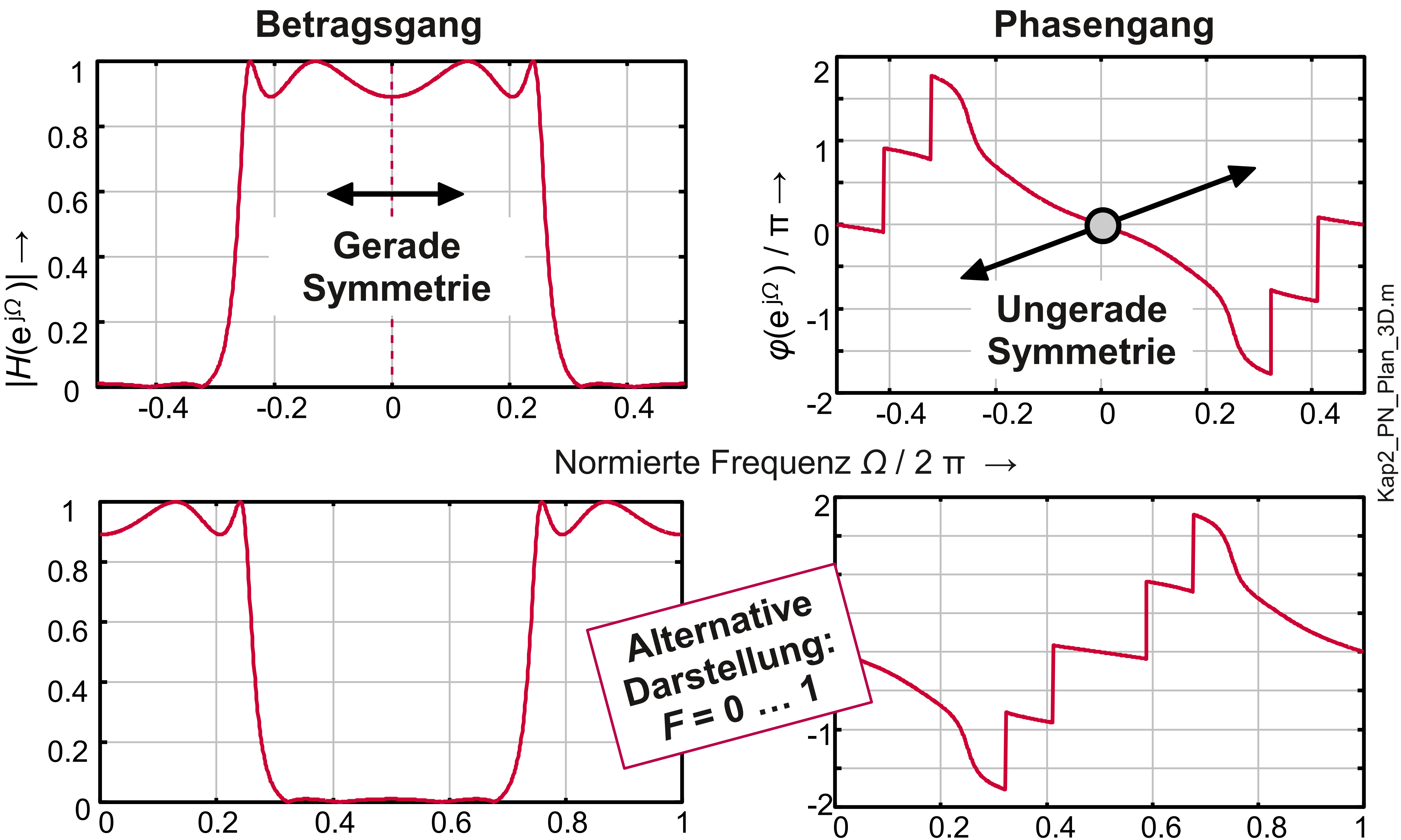
Zeitkontinuierliches Eingangs-signal mit Bandbreite *B*



Abgetastetes Signal mit f_S > 2*B*

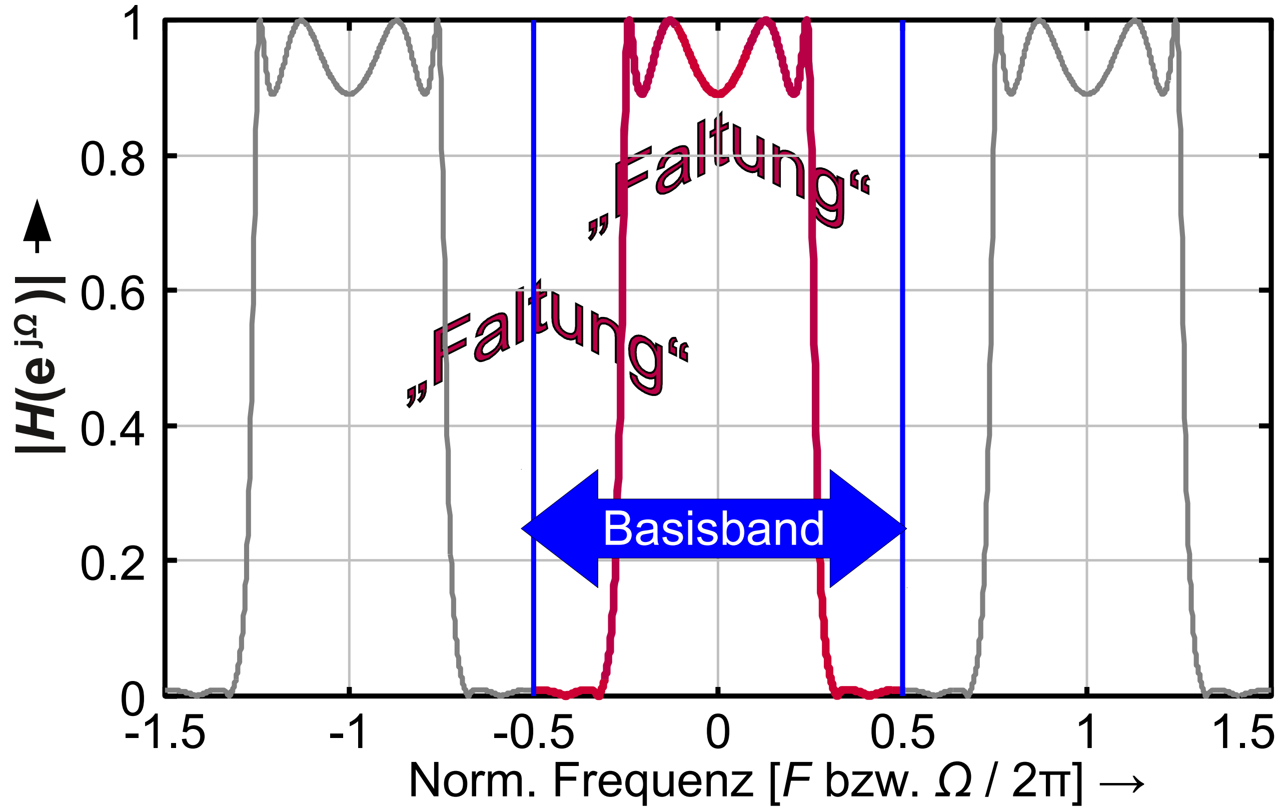


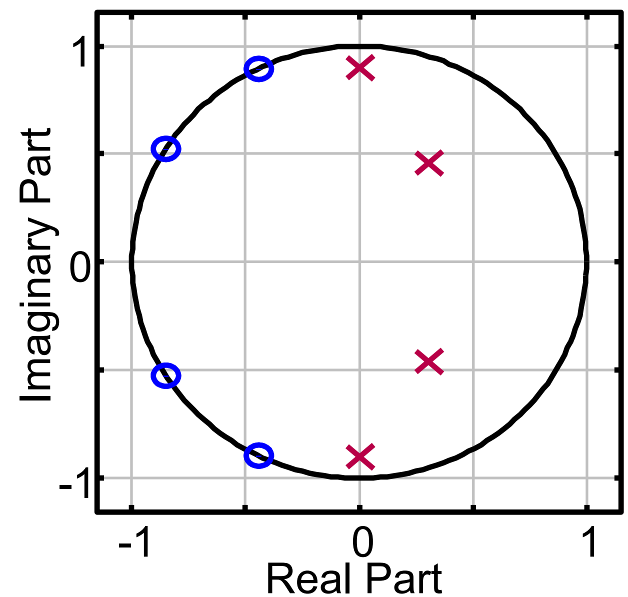
Spektren **reellwertiger** Signale und Systeme (1)



Spektren **reellwertiger** Signale und Systeme (2)

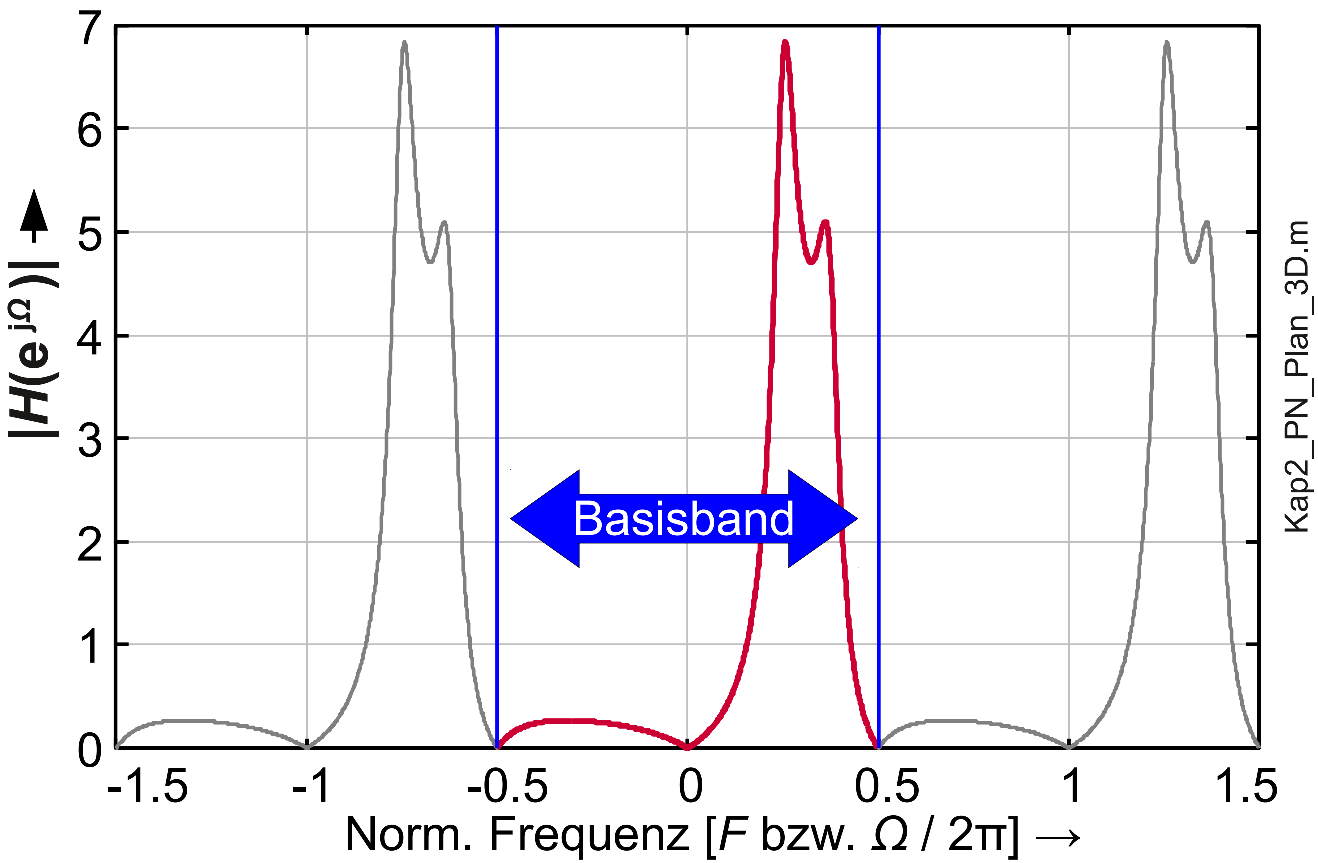
- Wiederholspektren bei $k f_S$ – *immer* bei zeitdiskreten Systemen
- Symmetrien um $k f_S / 2$ - *„Faltung“* - nur bei reellwertigen Systemen!
- Reellwertig ↔ P / N symm. zur *x*-Achse (konjugiert komplex)

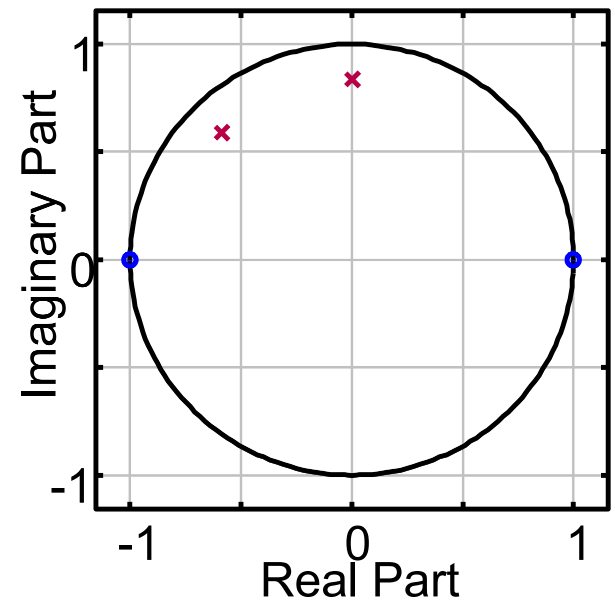




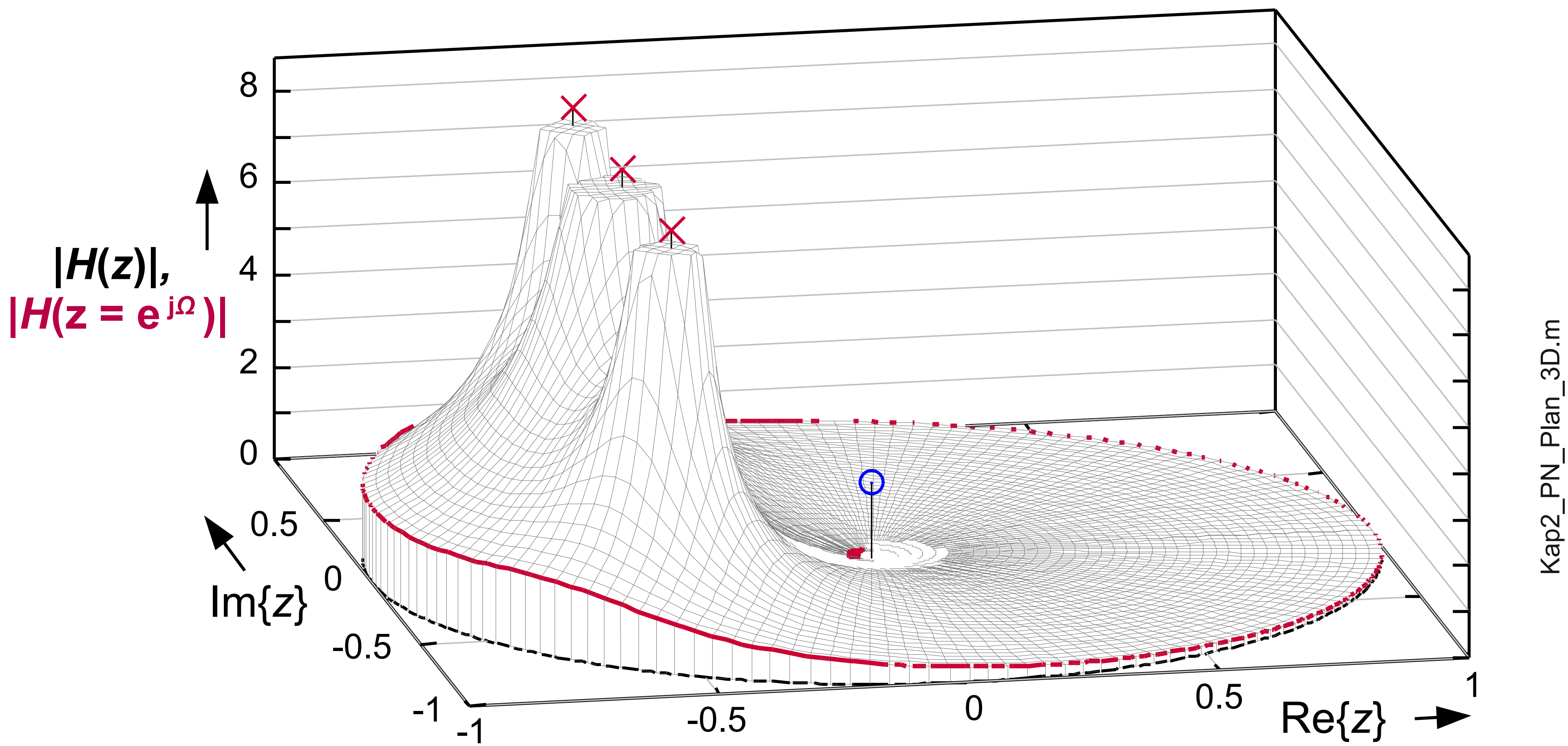
Spektren **komplexwertiger** Systeme

- Keine Symmetrien, nur Wiederholung bei *k*f_S - *keine „Faltung"!*
- In dieser Vorlesung fast ausschließlich reellwertige Systeme



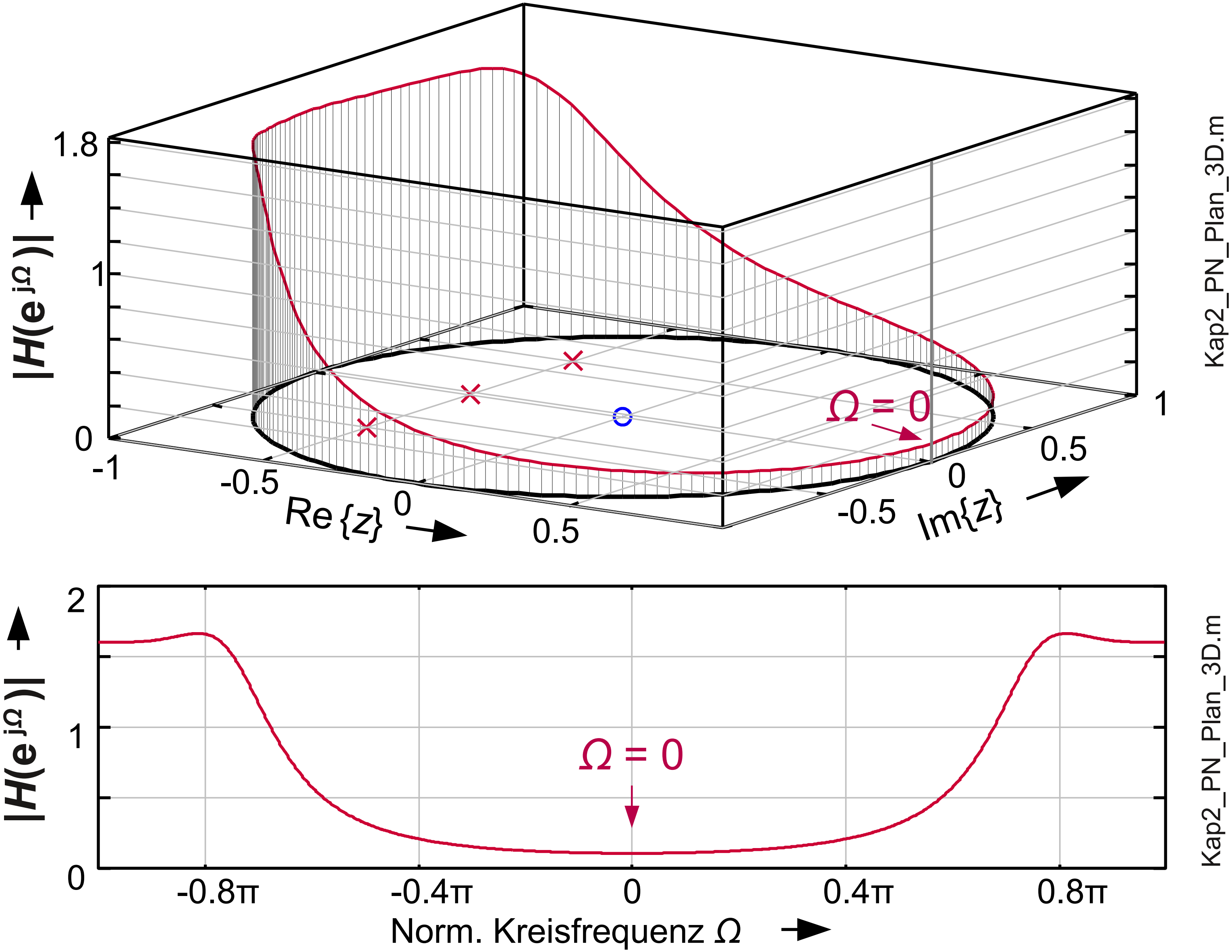


3D-Frequenzgang in der *z*-Ebene (1)



Frequenzgang **|*H*(e $^{j\Omega}$)|** wird entlang des Einheitskreises abgelesen

3D-Frequenzgang in der z-Ebene (2)



Digitale Signalverarbeitung auf FPGAs

Kap. 2 Zeitdiskrete Signale und LTI-Systeme im Frequenzbereich

Teil 2 *Schreibformen für die Systemfunktion H(z)*

2016

Dr. Christian Münker

Schreibformen für Systemfunktion $H(z)$

$H(z)$ kann durch mathematische Umformungen u.a. in die folgenden Formen gebracht werden:

Polynomform mit negativen Exponenten ↔ DZGL, SFG in Direktform

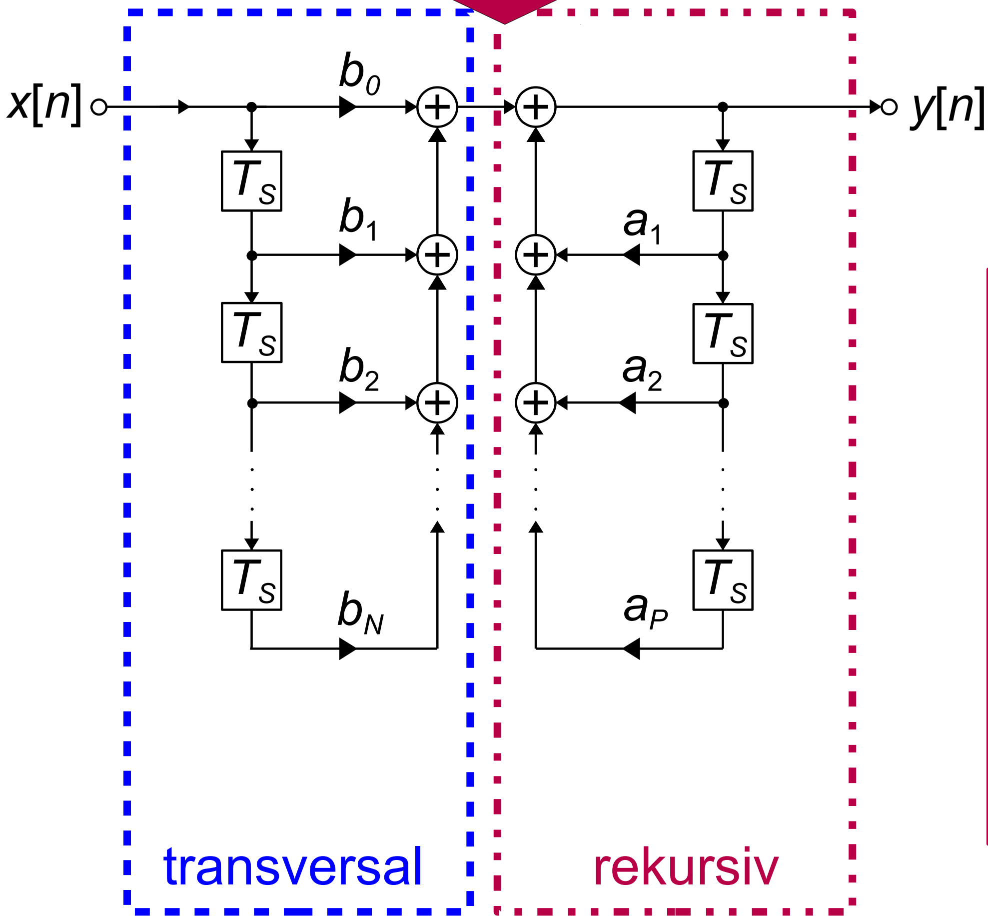
Polynomform mit positiven Exponenten durch Erweitern mit z^N ↔ Produktform, Matlab / Python-Repräsentation

Produktform durch Faktorisierung → Bestimmung von Pol/Nullstellen

Summenform durch Polynomdivision → Summe aus Termen erster und zweiter Ordnung (hier nicht behandelt)

Polynomform (neg. Exponenten) von *H*(*z*)

$$y[n] = \sum_{k=0}^{N} b_k x[n-k] + \sum_{i=1}^{P} a_i y[n-i] \quad \Leftrightarrow \quad Y(z) = \sum_{k=0}^{N} b_k X(z) z^{-k} + \sum_{i=1}^{P} a_i Y(z) z^{-i}$$



$$\Rightarrow H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{N} b_k z^{-k}}{1 - \sum_{i=1}^{P} a_i z^{-i}}$$

Differenzengleichung

⇕

Übertragungsfunktion *H*(*z*)

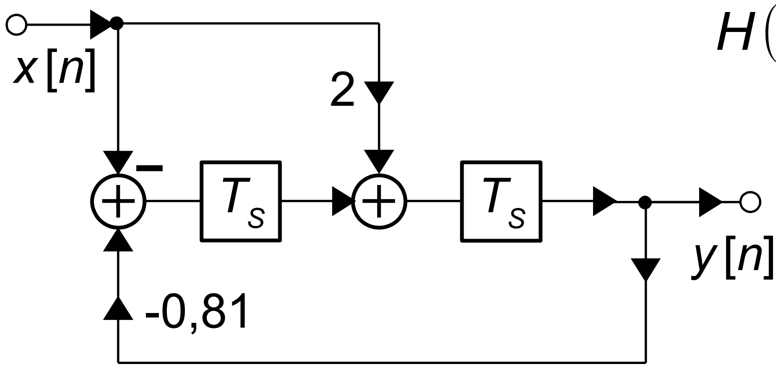
⇕

Direkte Hardware-Konstruktion

Erweitern von Nenner und Zähler mit $z^P\, z^N$ führt zu Polynomform mit *positiven Exponenten*:

$$\Rightarrow H(z) = \frac{\sum_{k=0}^{N} b_k z^{-k}}{1 - \sum_{i=1}^{P} a_i z^{-i}} = \frac{\sum_{k=0}^{N} b_k z^{N-k}}{z^P - \sum_{i=1}^{P} a_i z^{P-i}} \frac{z^P}{z^N}$$

Diese Form kann leichter in die Produktform zerlegt werden und definiert die Koeffizienten für die Matlab / Python-Repräsentation:



$$H(z) = \frac{2z^{-1} - z^{-2}}{1 + 0{,}81\, z^{-2}} = \frac{2z - 1}{z^2 + 0{,}81} = \frac{2(z - 0{,}5)}{(z - 0{,}9\,\mathrm{j})(z + 0{,}9\,\mathrm{j})}$$

ML / Python: `b = [2, -1]`

`a = [1, 0, 0.81]`

Produkt- oder Pol/Nullstellenform von $H(z)$

$$H(z) = \frac{\sum\limits_{k=0}^{N} b_k z^{N-k}}{z^P - \sum\limits_{i=1}^{P} a_i z^{P-i}} \frac{z^P}{z^N} = b_0 \frac{\prod\limits_{k=1}^{N} (z - \underline{z}_{0,k})}{\prod\limits_{i=1}^{P} (z - \underline{z}_{\infty,i})} z^{P-N}$$

$H(z)$ ist eindeutig definiert durch Koeffizienten a_k, b_k oder durch Nullstellen $z_{0,k}$ / Polstellen $z_{\infty,i}$ und Faktor b_0 !

Nutzen:

- **Stabilitätsanalyse:** Alle Pole innerhalb des EK ($|z_{\infty,i}| < 1$)?
- **Frequenzgang:** Geometrisches Abschätzen des Einflusses von P / N auf Punkte auf EK (nächster Abschnitt)
- **Systementwurf** im Frequenzbereich durch Platzieren von P / N

Digitale Signalverarbeitung auf FPGAs

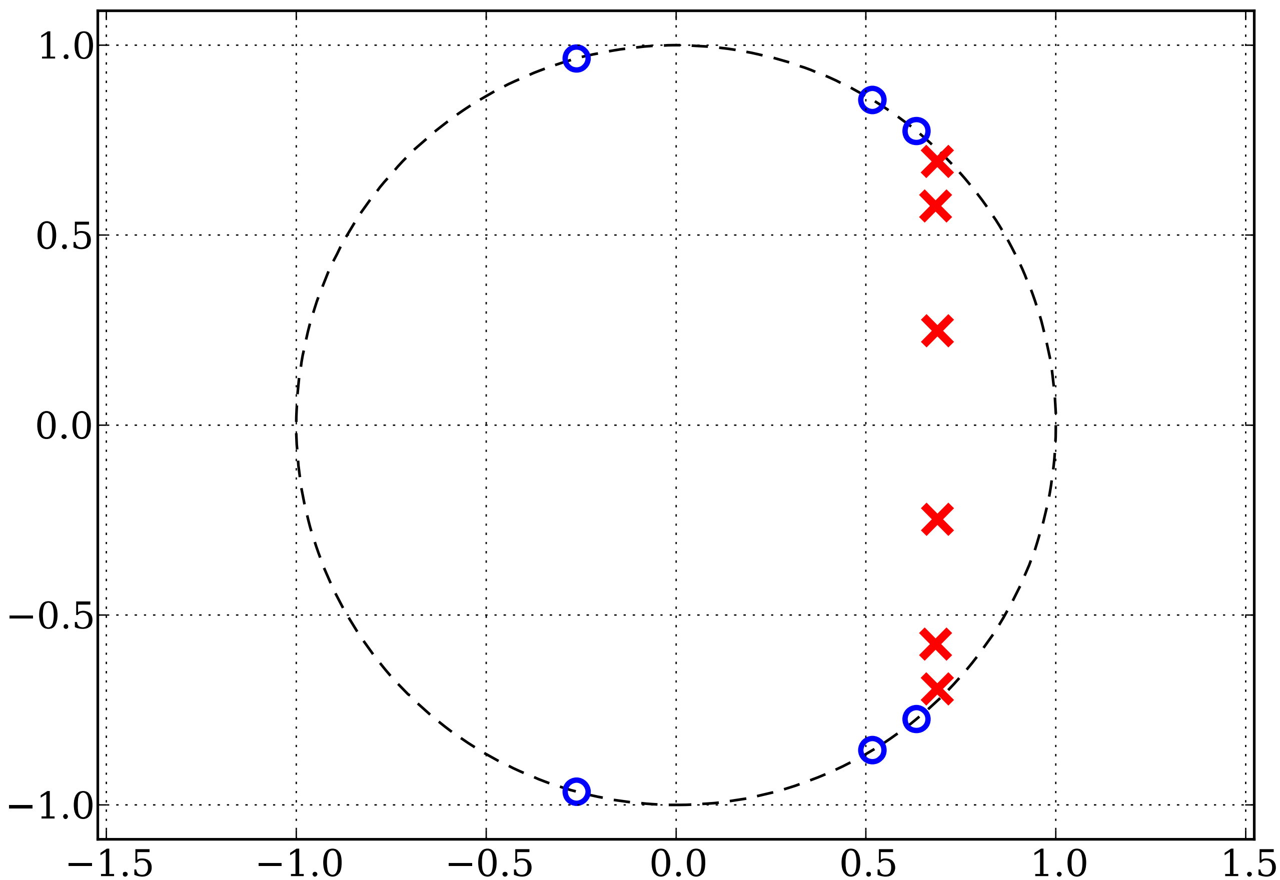
Kap. 2 Zeitdiskrete Signale und LTI-Systeme im Frequenzbereich

Teil 3 *Pole und Nullstellen*

2016

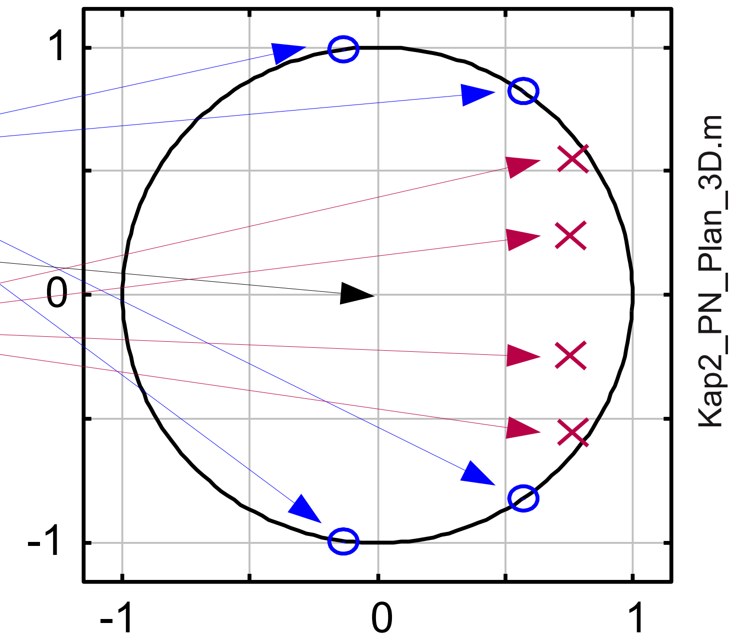
Dr. Christian Münker

Pole und Nullstellen



Pol/Nullstellendiagramm

$$H(z) = \frac{\sum_{k=0}^{N} b_k z^{N-k}}{z^P - \sum_{i=1}^{P} a_i z^{P-i}} \frac{z^P}{z^N} = b_0 \frac{\prod_{k=1}^{N}(z - \underline{z}_{0,k})}{\prod_{i=1}^{P}(z - \underline{z}_{\infty,i})} z^{P-N}$$



(hier: $P = N$)

- Grafische Darstellung von Pol- und Nullstellen gibt viele Informationen auf einen Blick
- Stabilität, Reellwertigkeit, Frequenzcharakteristik, Kausalität, ...

Einfluss von Polen / Nullstellen auf Frequenzgang

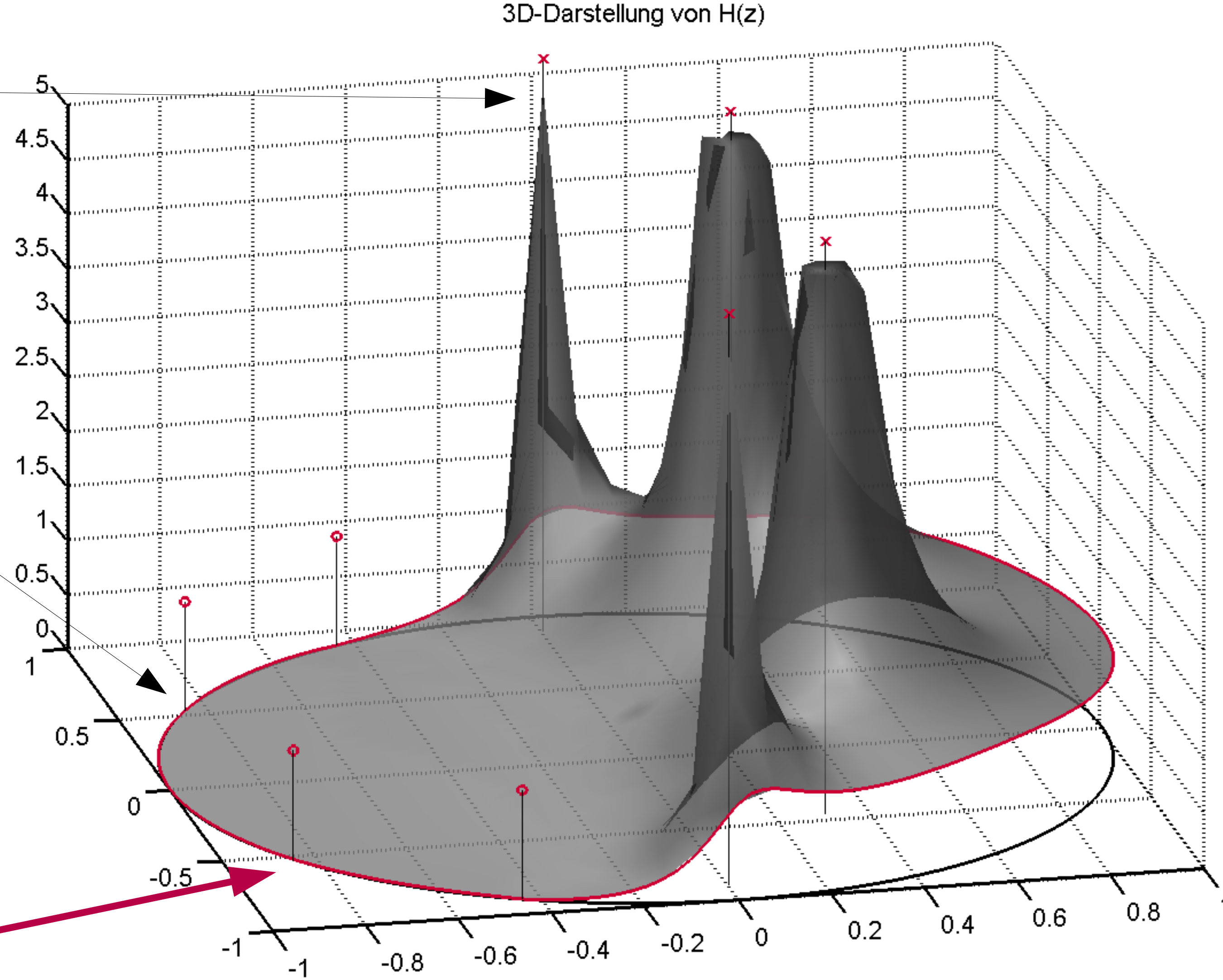
Polstellen = Zeltstangen

\+

Nullstellen = Heringe (Befestigung)

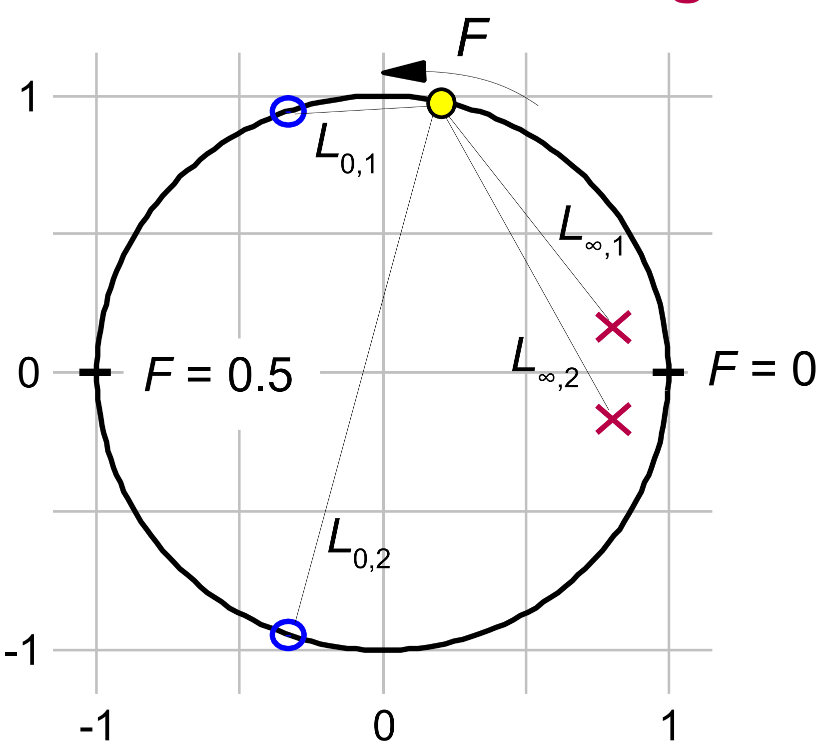
=

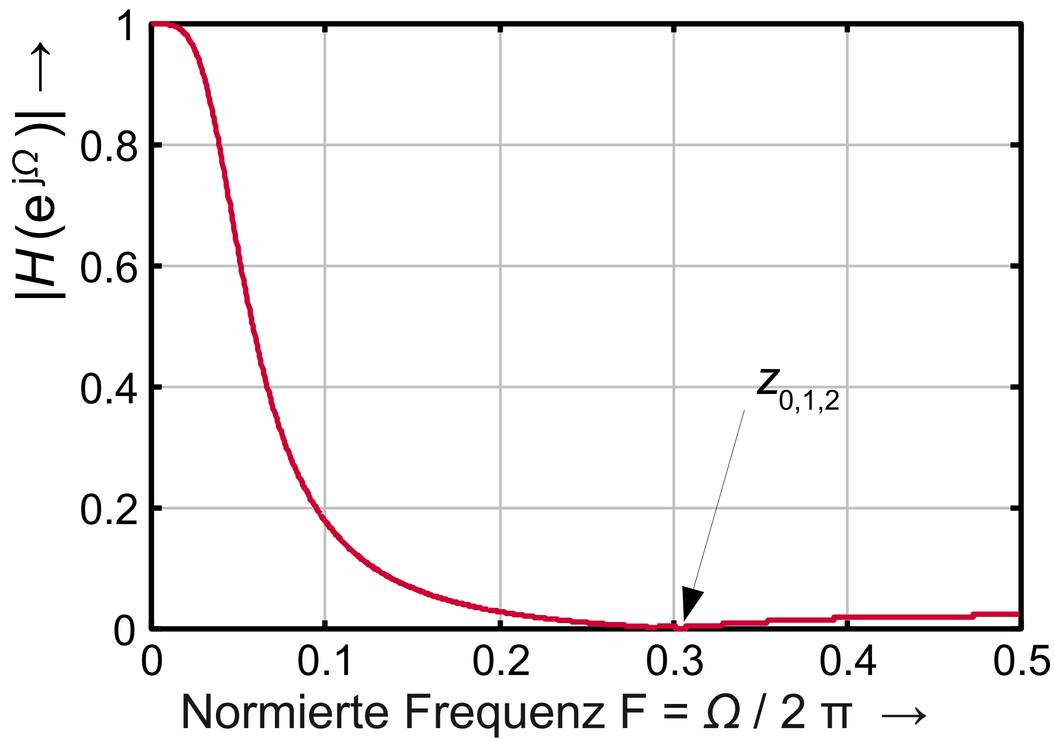
Frequenzgang entlang EK



$$H(z) = b_0 \frac{\prod\limits_{k=1}^{N}\left(z-\underline{z}_{0,k}\right)}{\prod\limits_{i=1}^{P}\left(z-\underline{z}_{\infty,i}\right)} z^{P-N} \Rightarrow \left|H\left(z = e^{j\Omega}\right)\right| = b_0 \frac{\prod\limits_{k=1}^{N}\overbrace{\left|e^{j\Omega}-\underline{z}_{0,k}\right|}^{L_{0,i}}}{\prod\limits_{i=1}^{P}\underbrace{\left|e^{j\Omega}-\underline{z}_{\infty,i}\right|}_{L_{\infty,i}}} \underbrace{\left|e^{j\Omega(P-N)}\right|}_{=1}$$

Geometrische Deutung:

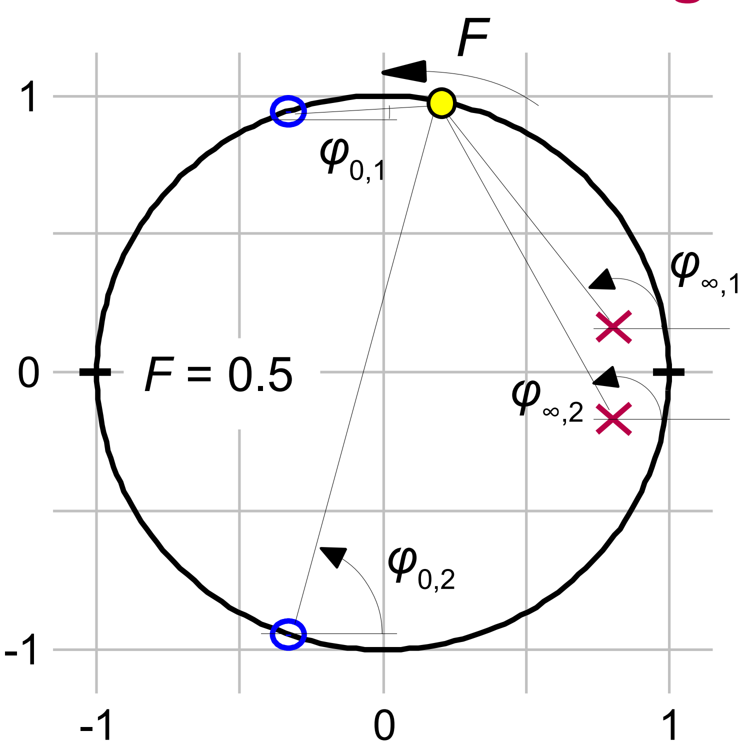


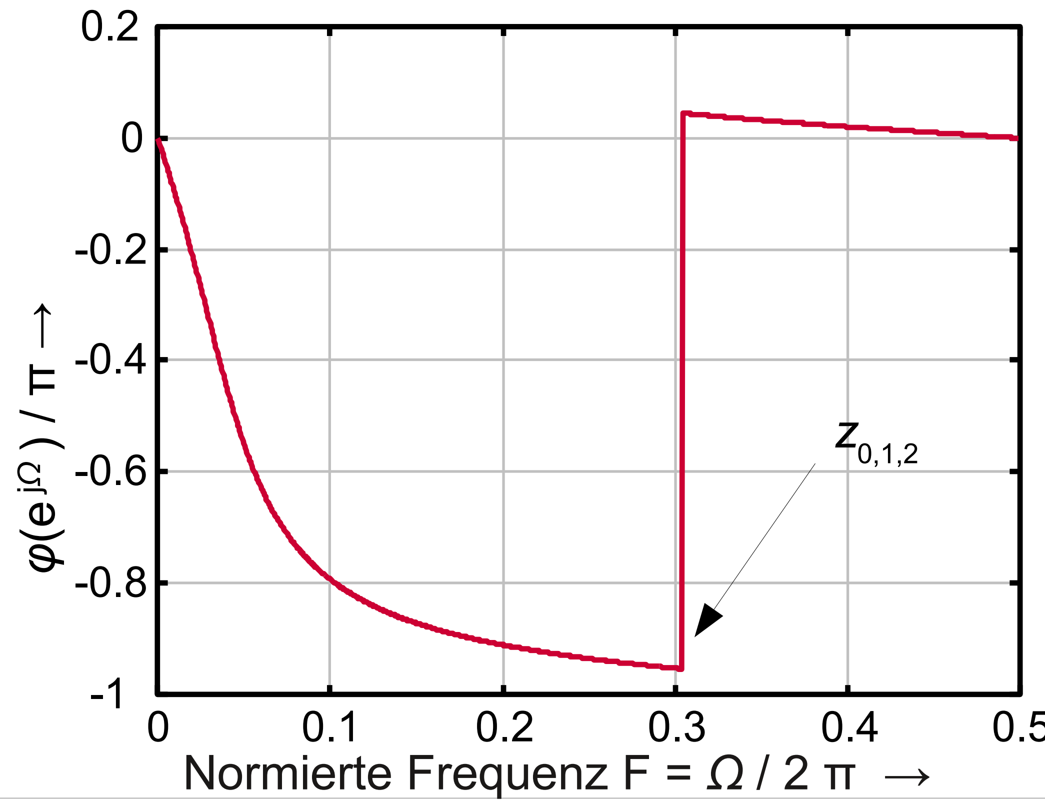


Phasengang aus P/N-Diagramm

$$H\left(e^{j\Omega}\right) = b_0 \frac{\prod_{k=1}^{N} e^{j\Omega} - \underline{z}_{0,k}}{\prod_{i=1}^{P} e^{j\Omega} - \underline{z}_{\infty,i}} e^{j\Omega(P-N)} \Rightarrow \varphi\left(e^{j\Omega}\right) = \sum_{k=1}^{N} \underbrace{\sphericalangle\left(e^{j\Omega} - \underline{z}_{0,k}\right)}_{\varphi_{0,k}} - \sum_{i=1}^{P} \underbrace{\sphericalangle\left(e^{j\Omega} - \underline{z}_{\infty,i}\right)}_{\varphi_{\infty,i}} + \underbrace{\Omega(P-N)}_{=\text{ lineare Phase}}$$

Geometrische Deutung:





Pol- und Nullstellen bei transversalen Filtern

Ausgangspunkt für Rechnungen / Simulationen: Zeitbereich

Impulsantwort / SFG eines tranversalen (FIR) - Filters mit *N* Verzögerungen:

$$h[n] = \sum_{k=0}^{N} b_k \delta[n-k] \Leftrightarrow H(z) = \sum_{k=0}^{N} b_k z^{-k}, \quad b_k \in \mathbb{C}$$

Übertragungsfunktion in Polynomform

$$\Leftrightarrow H(z) = z^{-N} \sum_{k=0}^{N} b_k z^{N-k}$$

Übertragungsfunktion in Polynomform mit positiven Exponenten → Einfachere Zerlegung in Produktform

$$\Leftrightarrow H(z) = b_0 \, z^{-N} \prod_{k=1}^{N} (z - \underline{z}_{0,k})$$

Übertragungsfunktion in Produktform

→ *N* Nullstellen

→ *N* Polstellen im Ursprung (z^{-N}) erkennbar

$H(z)$ **vollständig** definiert durch Koeff. b_k oder Nullstellen $z_{0,k}$ und Faktor b_0 !

Ausgangspunkt für Rechnungen / Simulationen: Zeitbereich

DZGL / SFG eines rekursiven (IIR) Filters mit N Verzögerungen im transversalen und P Verzögerungen im rekursiven Teil:

$$y[n] = \sum_{k=0}^{N} b_k x[n-k] + \sum_{i=1}^{P} a_i y[n-i] \Rightarrow H(z) = \frac{\sum_{k=0}^{N} b_k z^{-k}}{1 - \sum_{i=1}^{P} a_i z^{-i}} = b_0 \frac{\prod_{k=1}^{N} (z - \underline{z}_{0,k})}{\prod_{i=1}^{P} (z - \underline{z}_{\infty,i})} z^{P-N}$$

Übertragungsfunktion in Produktform:

→ N Nullstellen, P Polstellen, zusätzlich

→ (P - N) Nullstellen bzw.
(N - P) Polstellen im Ursprung (z^{P-N})

$H(z)$ **vollständig** definiert durch Null- und Polstellen $z_{0,k}$, $z_{\infty,i}$ und Faktor b_0 !

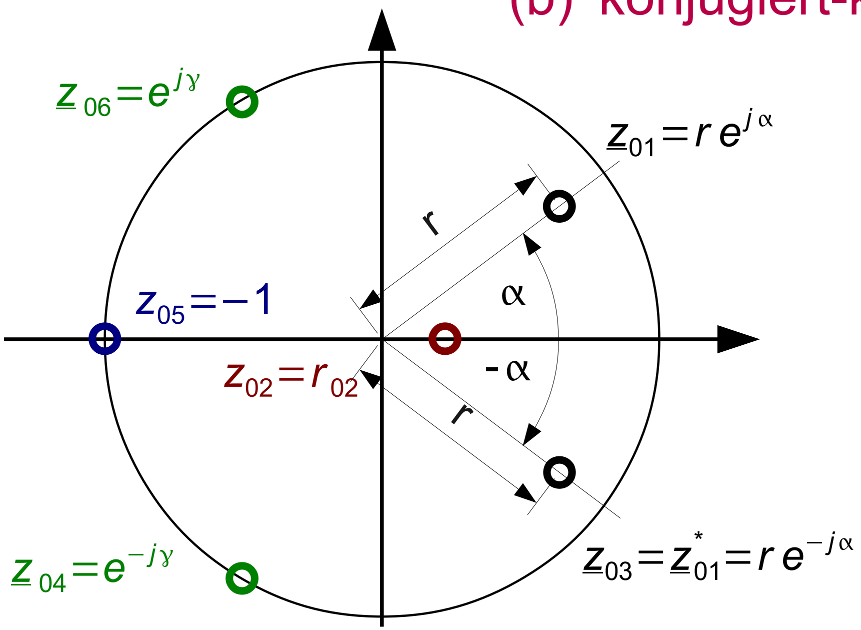
Filter mit reellwertigen Koeffizienten

Wo müssen Nullstellen z_{0K} der Übertragungsfunktion $H(z)$ liegen, damit ein Filter nur reellwertige Koeffizienten b_k hat (analog für Polstellen)?

$$h[n] = \sum_{k=0}^{N} b_k \delta[n-k] \;\Rightarrow\; H(z) = \sum_{k=0}^{N} b_k z^{-k} = \sum_{k=0}^{N} b_k \frac{z^{N-k}}{z^N} = b_0 z^{-N} \prod_{k=1}^{N} (z - \underline{z}_{0k}), \quad b_k \in \mathbb{R}$$

Erfüllbar für: (a) reellwertige Nullstellen (z_{02} und z_{05})

(b) konjugiert-komplexe Nullstellenpaare ($\underline{z}_{01} = \underline{z}^*_{03}$, $z_{04} = z^*_{06}$)

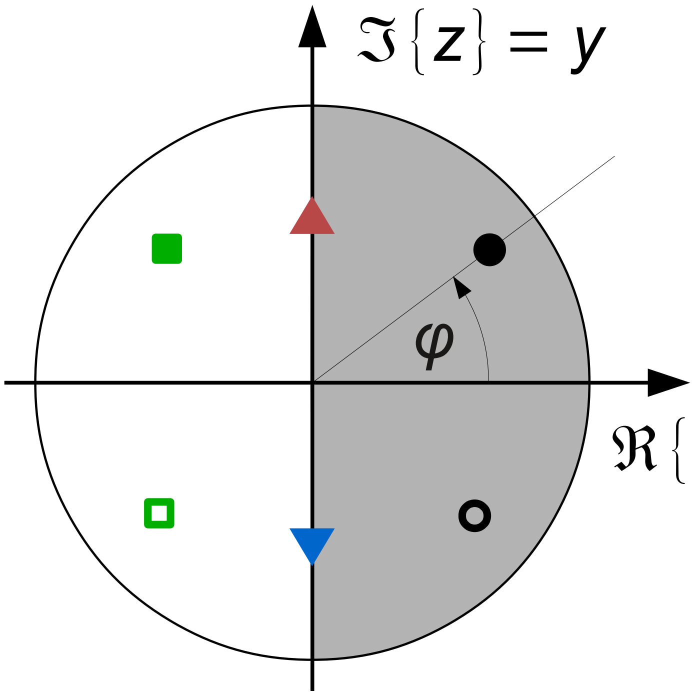


Produkt von zwei konjugiert komplexen Nullstellen liefert *reellwertigen* Beitrag!

$$(z - z_{0k})(z - z^*_{0k}) = z^2 - 2\,\Re\{z_{0k}\}\,z + |z_{0k}|^2$$

Gilt genauso für Polstellen(paare) bei rekursiven Systemen!

Atan2 und arctan - Funktion



$$\varphi = \arctan\frac{y}{x} = \arctan\frac{-y}{-x}$$

Nur eindeutig für x > 0 !

Neue Definition: $$\varphi = \mathrm{atan2}(y,x) = \begin{cases} \arctan\frac{y}{x} & \text{für } x>0 \\ \arctan\frac{y}{x}+\pi & \text{für } x<0,\ y\geq 0 \\ \arctan\frac{y}{x}-\pi & \text{für } x<0,\ y<0 \\ \frac{\pi}{2} & \text{für } x=0,\ y>0 \\ -\frac{\pi}{2} & \text{für } x=0,\ y<0 \end{cases}$$

Matlab: `atan2(y,x)` oder `angle(z)`

Python: `np.arctan2(y,x)`
oder `np.angle(z)`

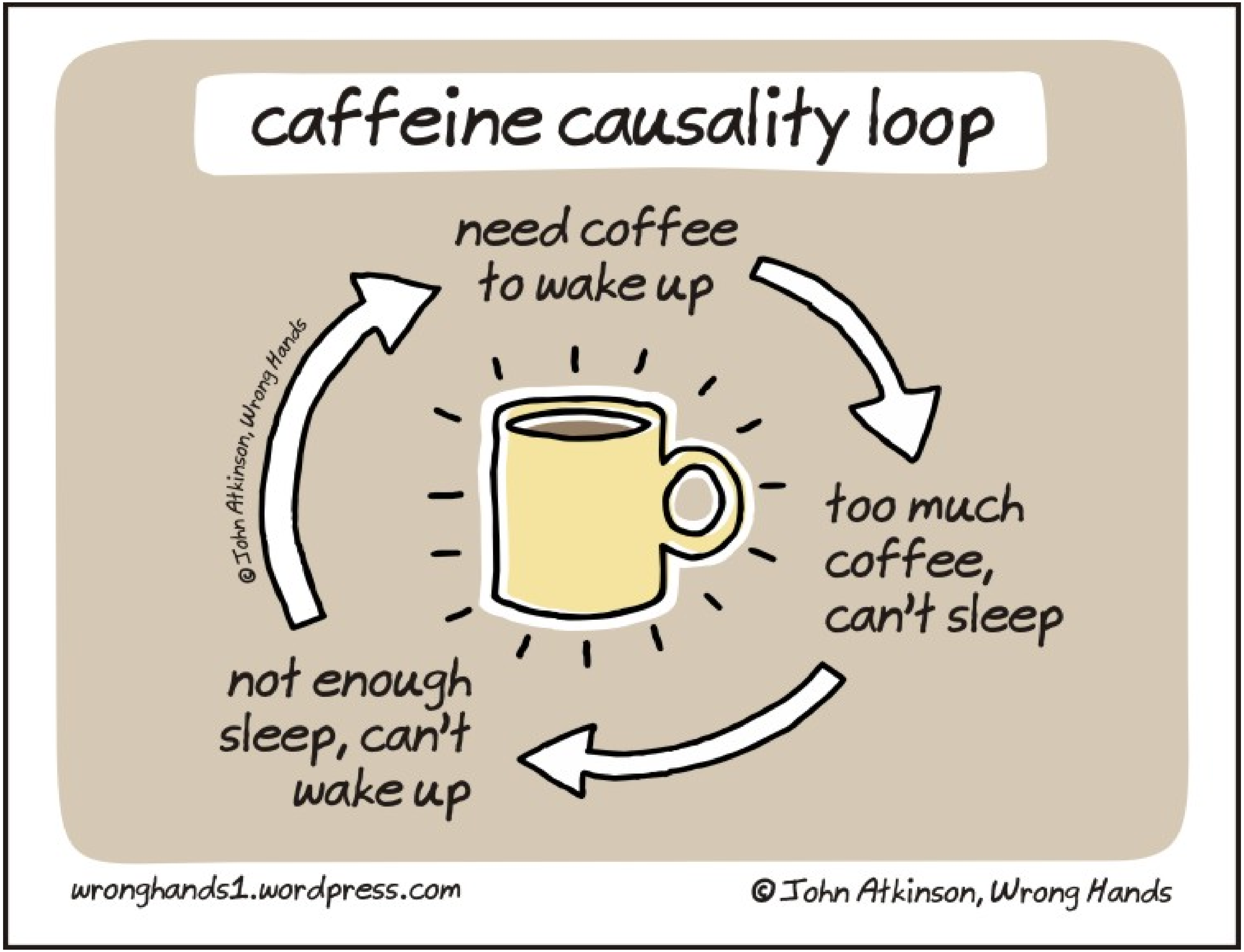
Digitale Signalverarbeitung auf FPGAs

Kap. 2 Zeitdiskrete Signale und LTI-Systeme im Frequenzbereich

Teil 4 *Kausalität*

2016

Dr. Christian Münker


caffeine causality loop
need coffee to wake up
too much coffee, can't sleep
not enough sleep, can't wake up
©John Atkinson, Wrong Hands
wronghands1.wordpress.com
©John Atkinson, Wrong Hands

Pol- und Nullstellen im Ursprung

$$H(z) = \frac{\sum_{k=0}^{N} b_k z^{-k}}{1 - \sum_{i=1}^{P} a_i z^{-i}} = b_0 \frac{\prod_{k=1}^{N} (z - \underline{z}_{0,k})}{\prod_{i=1}^{P} (z - \underline{z}_{\infty,i})} z^{P-N}$$

- Anzahl der Nullstellen außerhalb des Ursprungs ist Grad des Zählerpolynoms (*N*)
- Anzahl der Pole außerhalb des Ursprungs ist Grad des Nennerpolynoms (*P*)
- „Ordnung“ des Filters ist max[*N*,*P*]
  - falls $P > N$: es gibt P - N zusätzliche Nullstellen im Ursprung
  - falls $P < N$: es gibt N - P zusätzliche Pole im Ursprung (z.B. FIR: $P = 0$)

  Da P / N im Ursprung keinen Einfluss auf Betragsgang haben (warum?), werden sie im P / N-Diagramm häufig nicht eingetragen.
- Aber welchen Einfluss haben P / N im Ursprung ???

Kausalität von $H(z)$ (1)

Entwurf ausgehend von der Differenzengleichung oder der Struktur ergibt immer ein ***kausales System*** (= keine Reaktion am Ausgang bevor Stimulus am Eingang angelegt wird).

Voraussetzung: Es werden nur Verzögerungen verwendet, z.B. $x[n-3]$

Das gleiche gilt, wenn man mit der Übertragungsfunktion $H(z)$ in Polynomform mit *negativen Koeffizienten* beginnt, z.B. z^{-3} (warum?)

Startet man mit der Polynomform mit positiven Koeffizienten oder durch Angabe der P/N, können ***akausale Systeme*** entstehen.

Kausalität lässt sich am Einfachsten im Zeitbereich (Differenzengleichung) beurteilen, $y[n]$ darf nicht abhängen von $x[n+k]$:

$$H(z)=\frac{Y(z)}{X(z)}=\frac{k\prod_{i=1}^{N}(z-\underline{z}_{0,i})}{\prod_{i=1}^{P}(z-\underline{z}_{\infty,i})}=\frac{\sum_{i=0}^{N}\tilde{b}_i z^{i}}{z^{P}+\sum_{i=1}^{P}a_i z^{i}}=\frac{z^N}{z^P}\frac{\sum_{i=0}^{N}\tilde{b}_{N-i} z^{-i}}{1+\sum_{i=1}^{P}a_i z^{-i}}=\frac{z^{N-P}\sum_{i=0}^{N}b_i z^{-i}}{1+\sum_{i=1}^{P}a_i z^{-i}};\; b_i=\tilde{b}_{N-i}$$

$$\Rightarrow Y(z)=z^{N-P}\sum_{i=0}^{N}b_i z^{-i}X(z)-\sum_{i=1}^{P}a_i z^{-i}Y(z)$$

$$y[n]=\boxed{\delta[n+N-P]}*\sum_{i=0}^{N}b_i\,x[n-i]-\sum_{i=1}^{P}a_i\,y[n-i]=\sum_{i=0}^{N}b_i\,\boxed{x[n+N-P-i]}-\sum_{i=1}^{P}a_i\,y[n-i]$$

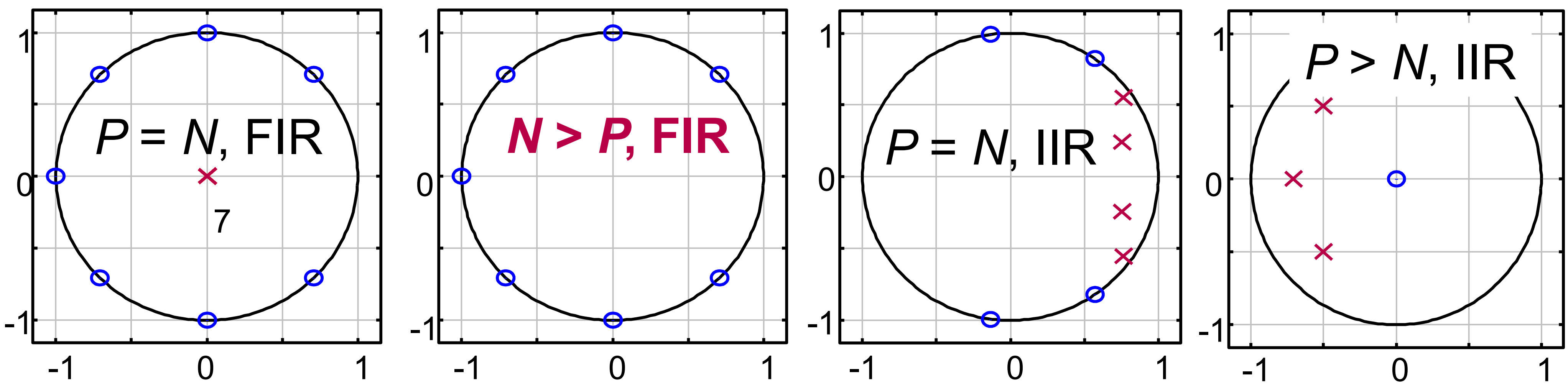
$\rightarrow$ ***N* ≤ *P***, ansonsten wird System akausal!

Kausalität von $H(z)$ (3)

$$H(z) = \frac{Y(z)}{X(z)} = k \frac{\prod_{i=1}^{N} (z - \underline{z}_{0,i})}{\prod_{i=1}^{P} (z - \underline{z}_{\infty,i})} = \frac{\sum_{i=0}^{N} \tilde{b}_i z^i}{z^P + \sum_{i=1}^{P} a_i z^i} = \frac{\sum_{i=0}^{N} b_i z^{-i}}{1 + \sum_{i=1}^{P} a_i z^{-i}} z^{N-P}; \quad b_i = \tilde{b}_{N-i}$$

- $P > N$: zusätzliche Verzögerung, $z^{-(N-P)}$
- $N > P$: nicht-kausales System, $z^{+(N-P)}$

Kap2_PN_Plan_3D.m



Digitale Signalverarbeitung auf FPGAs

Kap. 2 Zeitdiskrete Signale und LTI-Systeme im Frequenzbereich

Teil 5 *Darstellung von H(z) mit Python und Matlab*

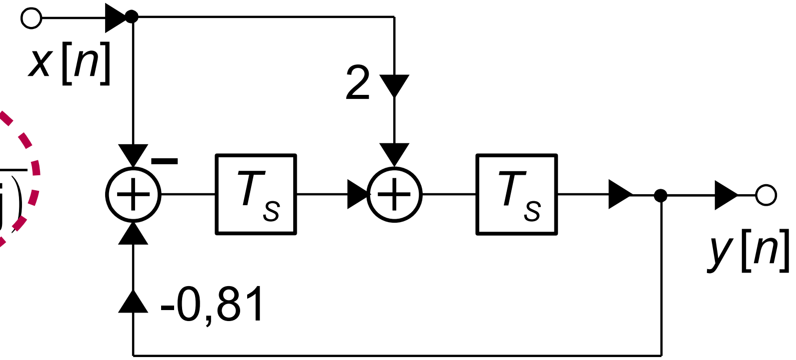
2016

Dr. Christian Münker

spyder

matplotlib

$$H(z) = \frac{2z^{-1} - z^{-2}}{1 + 0{,}81\,z^{-2}} = \frac{2z-1}{z^2 + 0{,}81} = \frac{2(z-0.5)}{(z-0.9\,j)(z+0.9\,j)}$$



Zähler- / Nennerkoeff.: `b = [2, -1]; a = [1, 0, 0.81];`

Nullstellen / Pole: `N = [0; 0.5]; P = [-0.9*j; 0.9*j];`

Umrechnung:

```
N = roots(b); P = roots(a);

b = k0 * poly(P); a = poly(N);
```

oder `[b,a] = zp2tf(z,p,k0);` **bzw.** `[z,p,k0]=tf2zp(b,a);`

Achtung: In Matlab sind Koeffizienten Zeilenvektoren, P/N Spaltenvektoren!

Python: np.roots(), np.poly(), scipy.signal.zpk2tf() und tf2zpk

Python / Matlab: Darstellung von *H*(*z*)

Pol/Nullstellen-Diagramm

Nullstellen / Pole: `zplane(N,P);`

Zähler- / Nennerkoeff.: `zplane(b,a);`

Je nachdem ob `zplane` mit Spalten- oder Zeilenvektoren aufgerufen wird, werden die Vektoren als P/N oder Koeffizienten interpretiert!

`P` bzw. `a` sind optional. Welche Werte haben `P` und `a` bei FIR-Filtern?

Frequenzgang

Kombiniert Betrag / Phase: `freqz(b,a);`

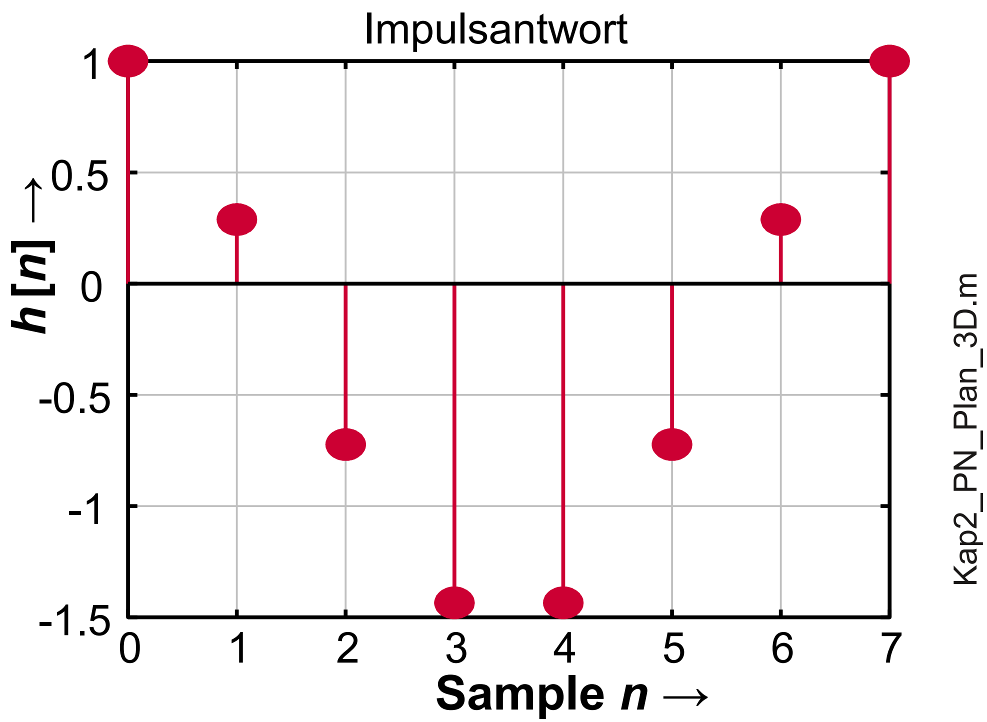
Detaillierte Plots: `[F,H] = freqz(b,a);`

Weitere Optionen sind die Anzahl der Frequenzpunkte und die Abtastfrequenz (Python: `scipy.signal.freqz()`).

Python / Matlab: Impulsantwort *h*[*n*]

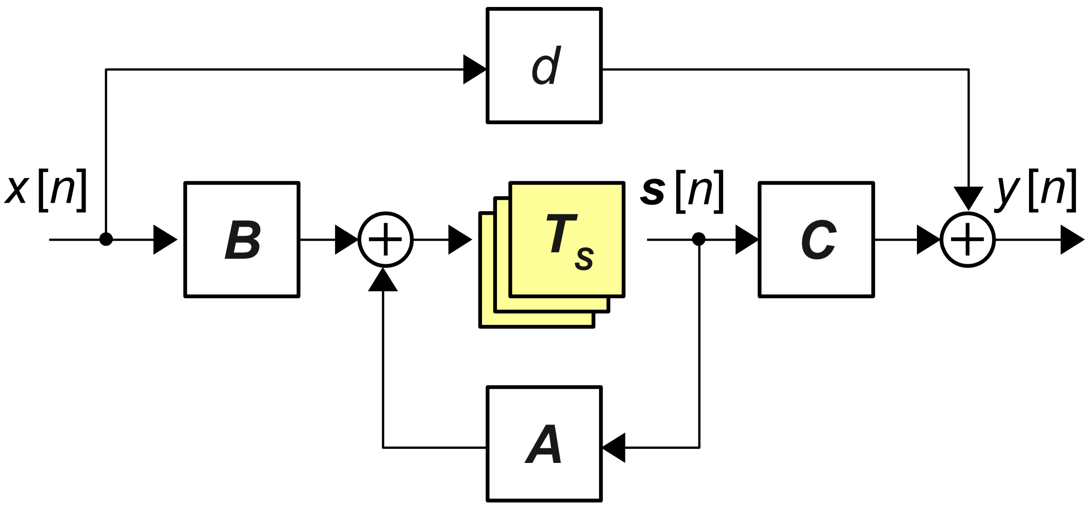
```
[himp,t]=impz(b,a,n,f_S);
stem(t, himp);
```

- Wenn P / N (Produktform) gegeben sind: Bestimmung der Koeffizienten **`b, a`** mit `poly()`
- Koeffizienten **`b`** entsprechen Impulsantwort bei FIR-Filter



`a, n, f_S` sind optional. `n` gibt die Anzahl der zu plottenden Impulse an, ansonsten versucht Matlab eine gute Darstellung zu finden. Durch Angabe von `f_S` wird die *t*-Achse mit absoluten Zeitangaben skaliert.

State-Space Darstellung von $H(z)$



$$\mathbf{s}[n+1] = \mathbf{A}\mathbf{s}[n] + \mathbf{B}x[n]$$
$$y[n] = \mathbf{C}\mathbf{s}[n] + d\,x[n]$$

$$\mathbf{S}(z)\mathbf{z} = \mathbf{A}\mathbf{S}(z) + \mathbf{B}X(z)$$
$$Y(z) = \mathbf{C}\mathbf{S}(z) + d\,X(z)$$

- Beschreibung eines beliebigen LTI Netzwerks mit Hilfe seiner **Zustandsvariablen** ***s***[*n*] (= **state** variables, Register) und Differenzen/Differenzialgleichungen erster Ordnung
- Eingangssignal *x*[*n*], Ausgangssignal *y*[*n*] und die Zustandsvariablen ***s***[*n*] sind über die Vektoren / Matrizen ***A***, ***B***, ***C***, ***d*** miteinander verknüpft
- Hier: Single-Input, Single-Output (SISO), aber MIMO genauso möglich

Digitale Signalverarbeitung auf FPGAs

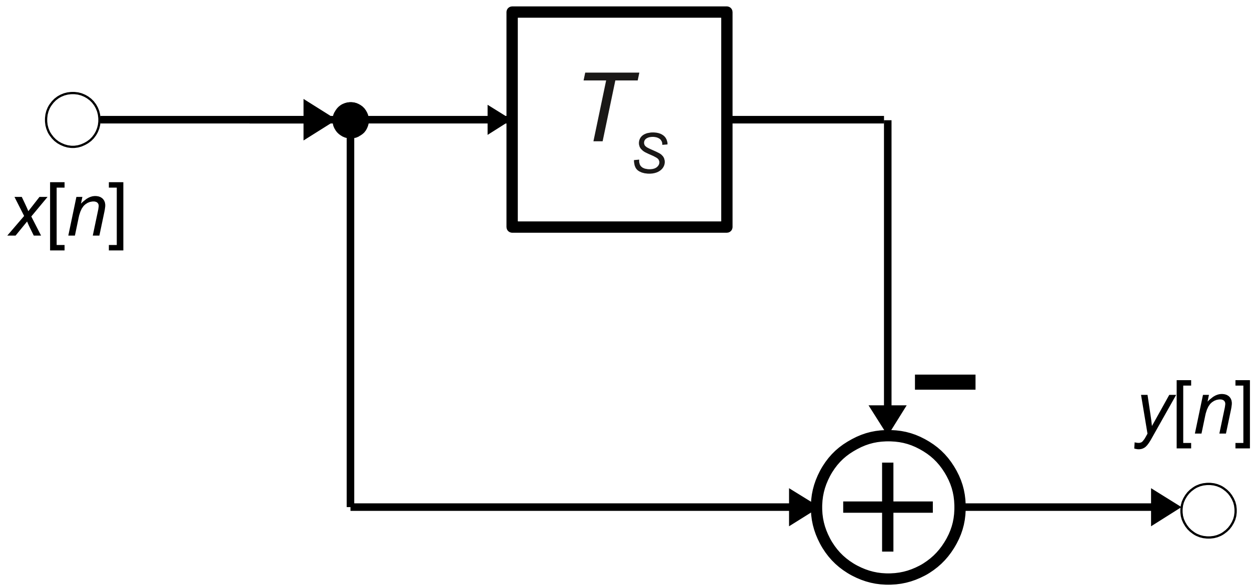
Kap. 2 Zeitdiskrete Signale und LTI-Systeme im Frequenzbereich

Teil 6 *Einfache Filter*

2016

Dr. Christian Münker

Einfache Filter



Wichtige Umformungen

Euler

$$e^{\pm j\Omega} = \cos\Omega \pm j\sin\Omega$$
$$e^{j\Omega} + e^{-j\Omega} = 2\cos\Omega$$
$$e^{j\Omega} - e^{-j\Omega} = 2j\sin\Omega$$

Endliche geometrische Reihe

$$H(z) = \sum_{n=0}^{N} z^{-n} = \frac{1-z^{-(N+1)}}{1-z^{-1}}$$

$$\Rightarrow \; H(e^{j\Omega}) = \sum_{n=0}^{N} e^{-jn\Omega} = \frac{1-e^{-j(N+1)\Omega}}{1-e^{-j\Omega}}$$

Zerlegung in „Spiegelpolynome“

$$H(z) = 1 + z^{-N} = z^{-N/2}\left(z^{N/2} + z^{-N/2}\right)$$

$$\Rightarrow \; H(e^{j\Omega}) = e^{-jN\Omega/2}\left(e^{jN\Omega/2} + e^{-jN\Omega/2}\right) = 2e^{-jN\Omega/2}\cos(N\Omega/2)$$

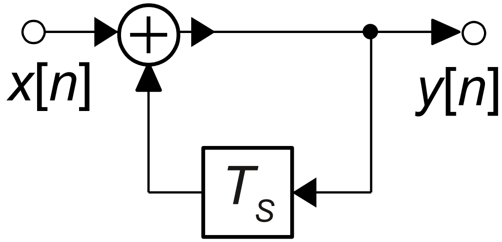
***N*-fache Nullstelle auf EK**

$$1 - z^N = 0 \;\Rightarrow\; z = \sqrt[N]{1} = \sqrt[N]{e^{j2k\pi}} \Rightarrow z = e^{j2k\pi/N} \quad \text{mit } k = \ldots, -1, 0, 1, \ldots$$

$$1 + z^N = 0 \;\Rightarrow\; z = \sqrt[N]{-1} = \sqrt[N]{e^{j\pi(2k+1)}} \Rightarrow z = e^{j\pi(2k+1)/N} \quad \text{mit } k = \ldots, -1, 0, 1, \ldots$$

$\Rightarrow$ N Nullstellen auf EK verteilt mit $\Delta\varphi = 2\pi/N$

Einfache Filter: Integrator

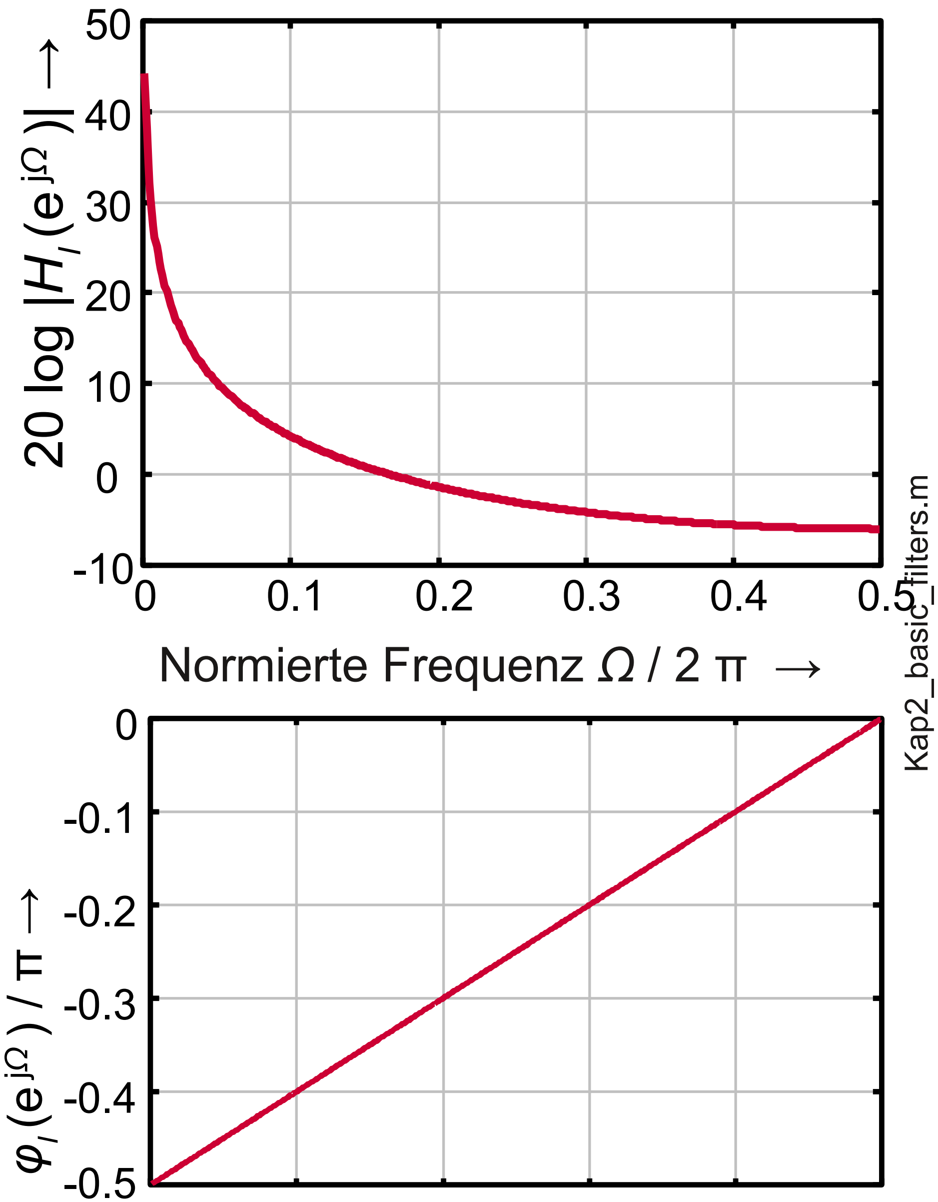


$$H_I(z) = \frac{1}{1 - z^{-1}} = \frac{z}{z-1}$$

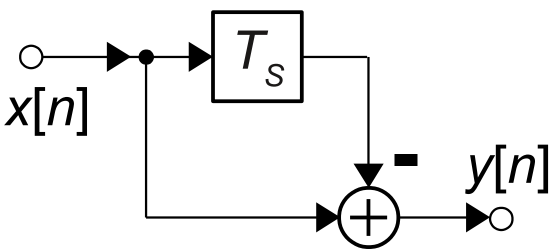
$$H_I(\mathrm{e}^{\mathrm{j}\Omega}) = \frac{1}{1 - \mathrm{e}^{-\mathrm{j}\Omega}} = \frac{1}{\mathrm{e}^{-\mathrm{j}\Omega/2}\left(\mathrm{e}^{+\mathrm{j}\Omega/2} - \mathrm{e}^{-\mathrm{j}\Omega/2}\right)}$$

$$= \frac{\mathrm{e}^{\mathrm{j}\Omega/2}}{2\,\mathrm{j}\sin(\Omega/2)} \approx \frac{1}{\mathrm{j}\Omega} \quad \text{für } \Omega \ll 1$$

- P/N-Diagramm?
- Stabilität?
- Lineare Phase??



Einfache Filter: Differenzierer



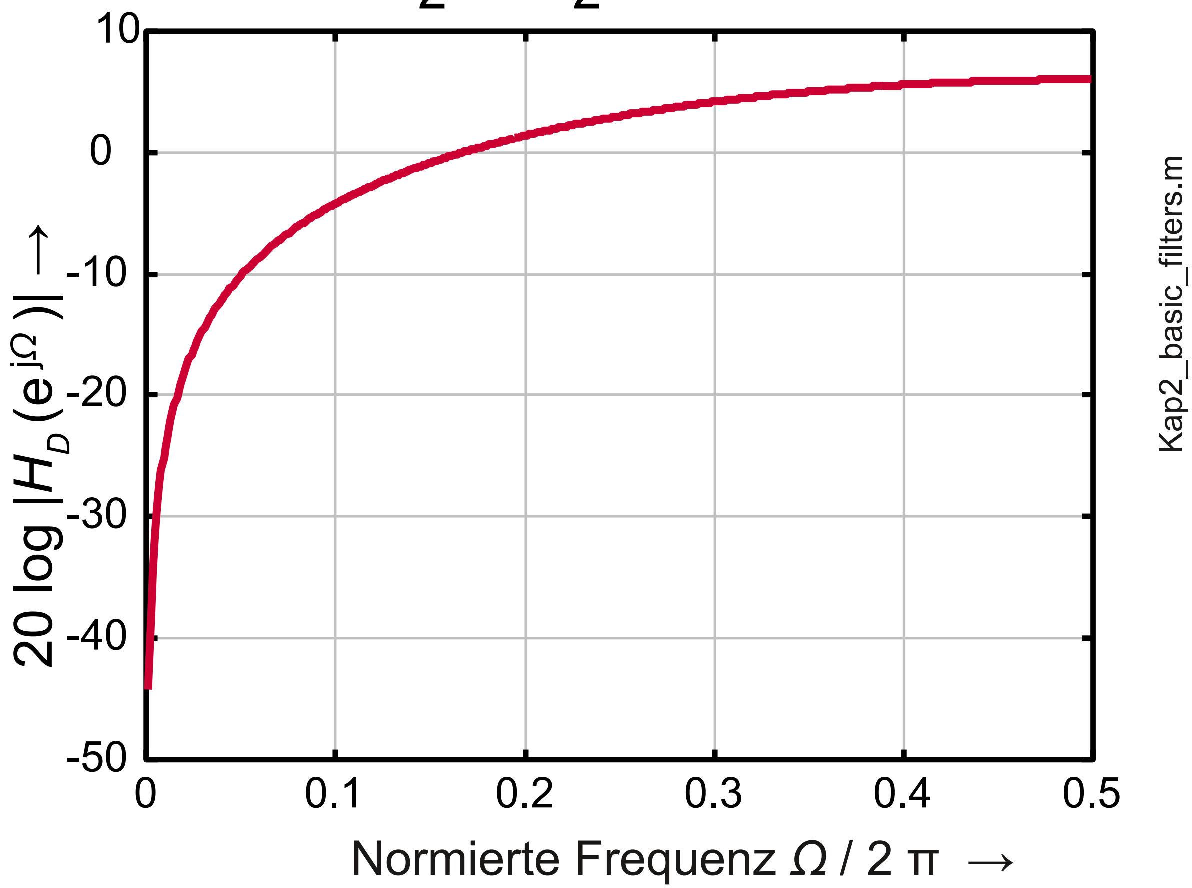
$$H_D(z) = 1 - z^{-1}$$

$$H_D(e^{j\Omega}) = 1 - e^{-j\Omega} = e^{-j\Omega/2}\left(e^{+j\Omega/2} - e^{-j\Omega/2}\right)$$

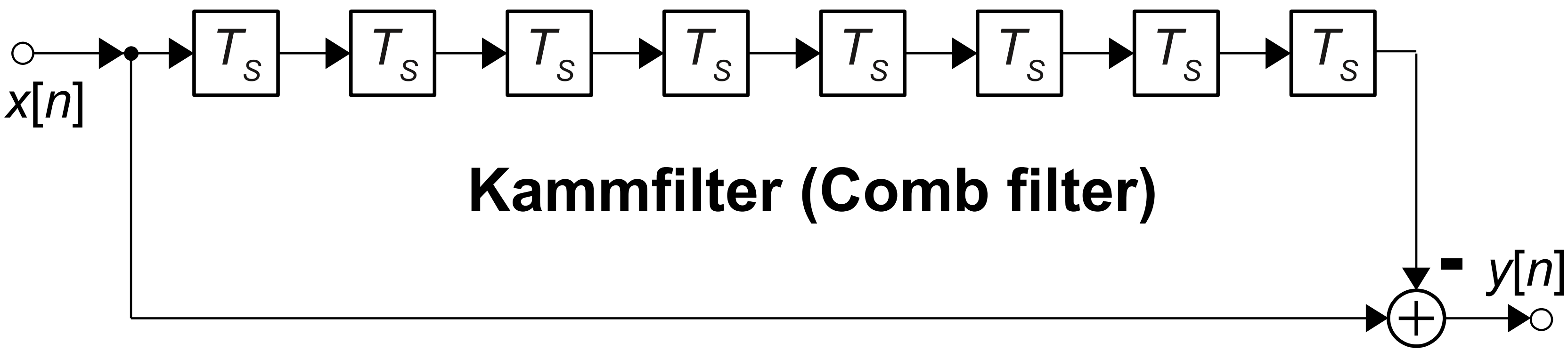
$$= 2\,j\,e^{-j\Omega/2}\sin(\Omega/2) \approx j\Omega \text{ für } \Omega \ll 1$$

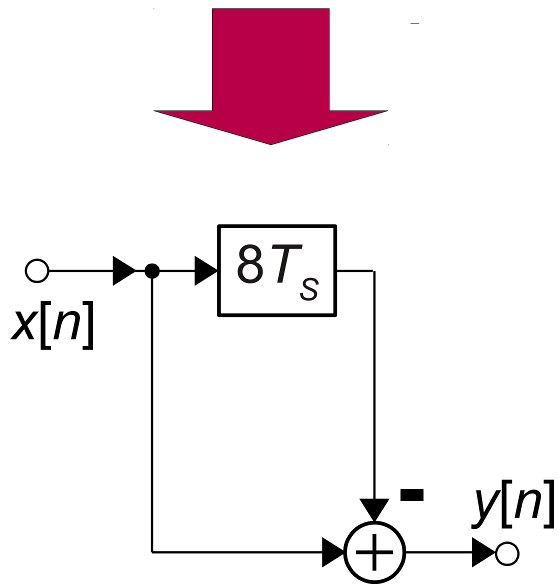
$$\varphi_D(\Omega) = \frac{\pi - \Omega}{2} - \frac{\pi}{2}\left(1 - \operatorname{sgn}\left(\sin(\Omega/2)\right)\right)$$

- $H_D(z = 1)$? $H_D(z = -1)$?
- P/N-Diagramm?
- Stabilität?
- Phasengang?



Einfache Filter: Kammfilter mit N = 8 Delays (1)





$$H_C(z) = 1 - z^{-8}$$

Einfache Filter: Kammfilter mit N = 8 Delays (2)

$$H_C(z) = 1 - z^{-N} \quad \Rightarrow$$

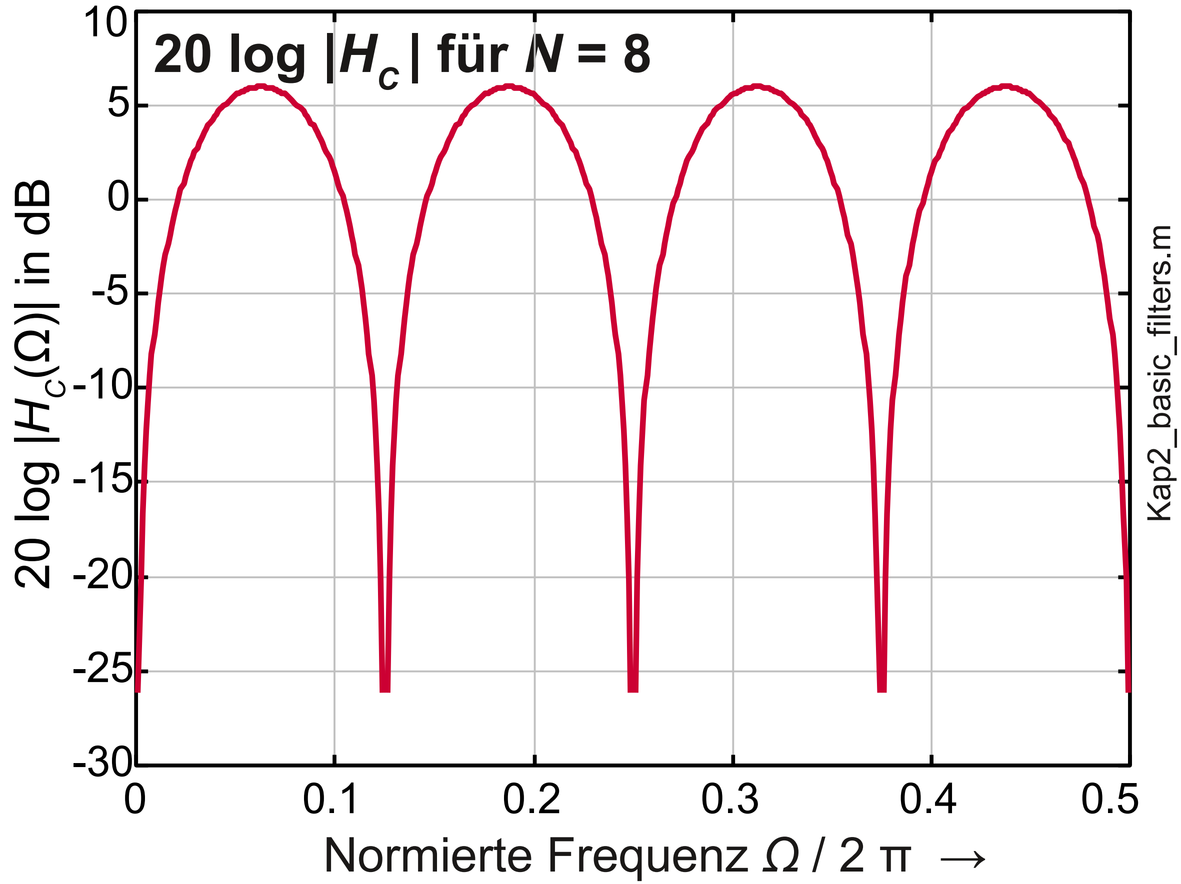
$$H_C(e^{j\Omega}) = 1 - e^{-jN\Omega}$$

$$= e^{-jN\Omega/2}\left(e^{+jN\Omega/2} - e^{-jN\Omega/2}\right)$$

$$= 2j\,e^{-jN\Omega/2}\sin(N\Omega/2)$$

$\Rightarrow$ ***N* Nullstellen** in $0 \leq \Omega < 2\pi$

- P/N-Diagramm?
- Stabilität?
- Phasengang?



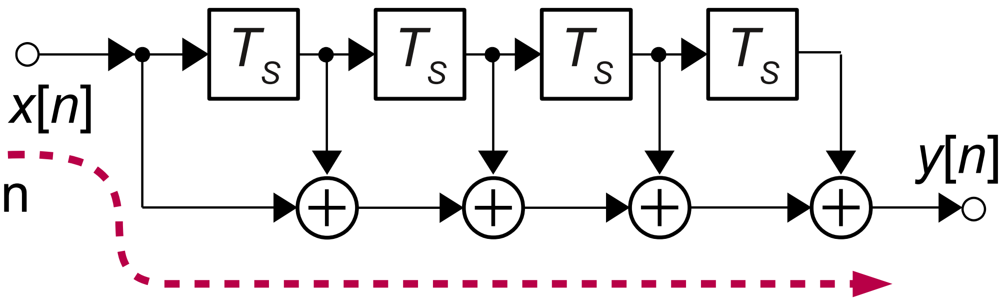
Anwendungen: Unterdrückung von periodischen Spektren (z.B. Fernsehtechnik), Teil des Cascaded Integrator Comb (CIC) Filters

Einfache Filter: Moving Average (MA) Filter

Ordnung *N* = 4:

N Delays,

N+1 Koeffizienten mit Wert 1



Hardware:

N Register und Addierer,

N Additionen / Sample

Kritischer Pfad:

N Addierer

Grundlegende Struktur für Digitale Signalverarbeitung:

- Gleiche Problemstellung bei rect-Puls und rect-Fenster !
- Multipliziererlose Filter sind wichtig vor allem für FPGA-Implementierungen !

MA-Filter (N = 4): Pol- und Nullstellen

$$\boldsymbol{h}[\boldsymbol{n}] = \{1;1;1;1;1\} = \sum_{m=0}^{4} \delta[n-m]$$

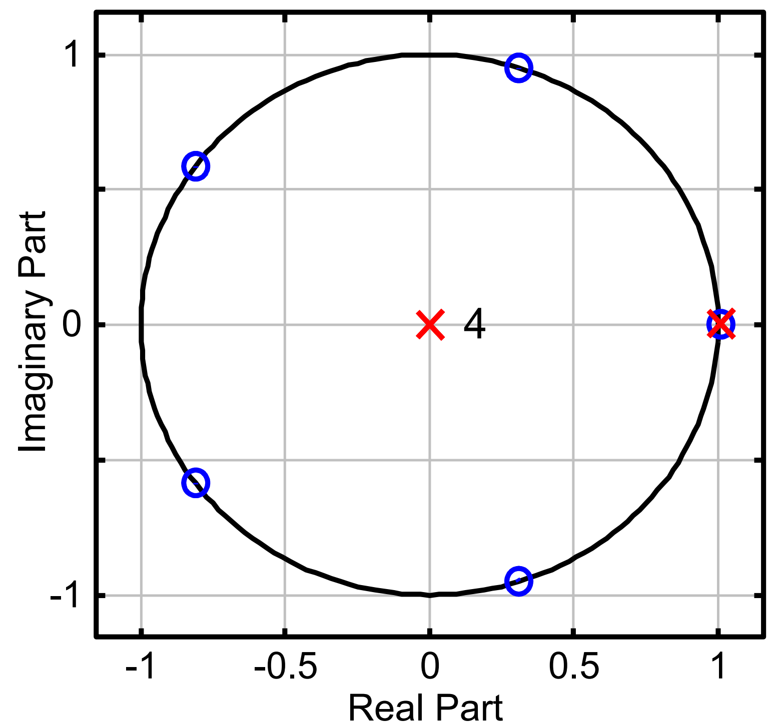
$$\Rightarrow \boldsymbol{H}(\boldsymbol{z}) = 1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} = \frac{1-z^{-5}}{1-z^{-1}} \text{ (mit endl. geom. Reihe) } = \frac{z^5-1}{(z-1)z^4}$$

Nullstellen: $z^5 - 1 = 0 \Rightarrow z = \sqrt[5]{1} = \sqrt[5]{e^{j2k\pi}} \Rightarrow z = e^{j2k\pi/5}$ mit $k = \dots, -1, 0, 1, \dots$

$\Rightarrow$ 5 Nullstellen auf EK verteilt mit $\Delta\varphi = 2\pi/5$

Polstellen: Vier Polstellen bei z = 0,
Eine Polstelle und eine Nullstelle
bei $z = 1$ ($f = 0$) **kompensieren sich**

$\rightarrow$ 4 wirksame Nullstellen!



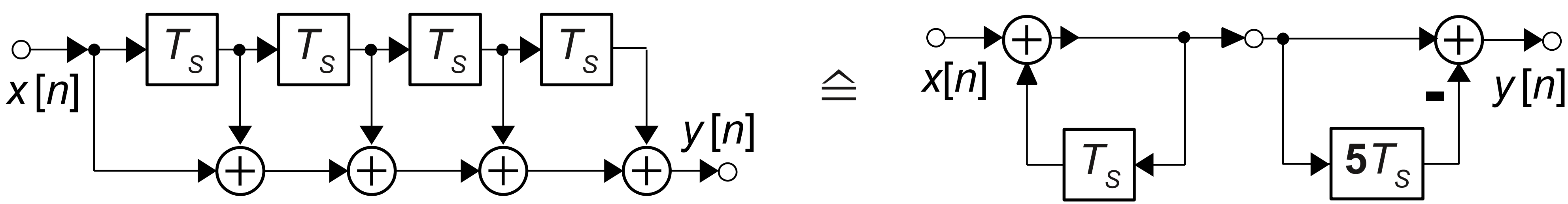
Matlab / Python

```
b = [1, 1, 1, 1, 1];
zplane(b);
```

Gedankenspiel ...

- MA-Filter kann auch als Kettenschaltung aus Kammfilter und Integrator realisiert werden
- Pol des Integrators würde Nullstelle bei $z = 1$ aufheben
- Theoretisch gleiche ÜF wie MA-Filter mit weniger Addierern
- Praktisch implementierbar? Reihenfolge Kammfilter / Integrator?

Siehe auch: Cascaced Integrator-Comb-Filter mit Überlauf-Arithmetik (Kap. 9)



MA-Filter (N = 4): Frequenzgang (1)

$$\boldsymbol{H\left(z = e^{j2\pi f T_s}\right)} \Rightarrow \begin{cases} H(f=0) & = & H(z=+1) = 5 \\ H(f=f_S/4) & = & H(z=j) = 1 \\ H(f=f_S/2) & = & H(z=-1) = 1 \end{cases}$$

$$= \sum_{n=0}^{4} e^{-jn\Omega} = e^{-j2\Omega} \sum_{n=-2}^{2} e^{-jn\Omega} = \underbrace{e^{-j2\Omega}}_{\text{Phase}} \underbrace{\left(1 + 2\cos(\Omega) + 2\cos(2\Omega)\right)}_{\text{Amplitude und Vorzeichen}}$$

Berechnung einzelner Werte von $H(f)$ aus Summe von cos-Funktionen leicht möglich, aber unpraktische Form für weitere Umformungen und Analysen.

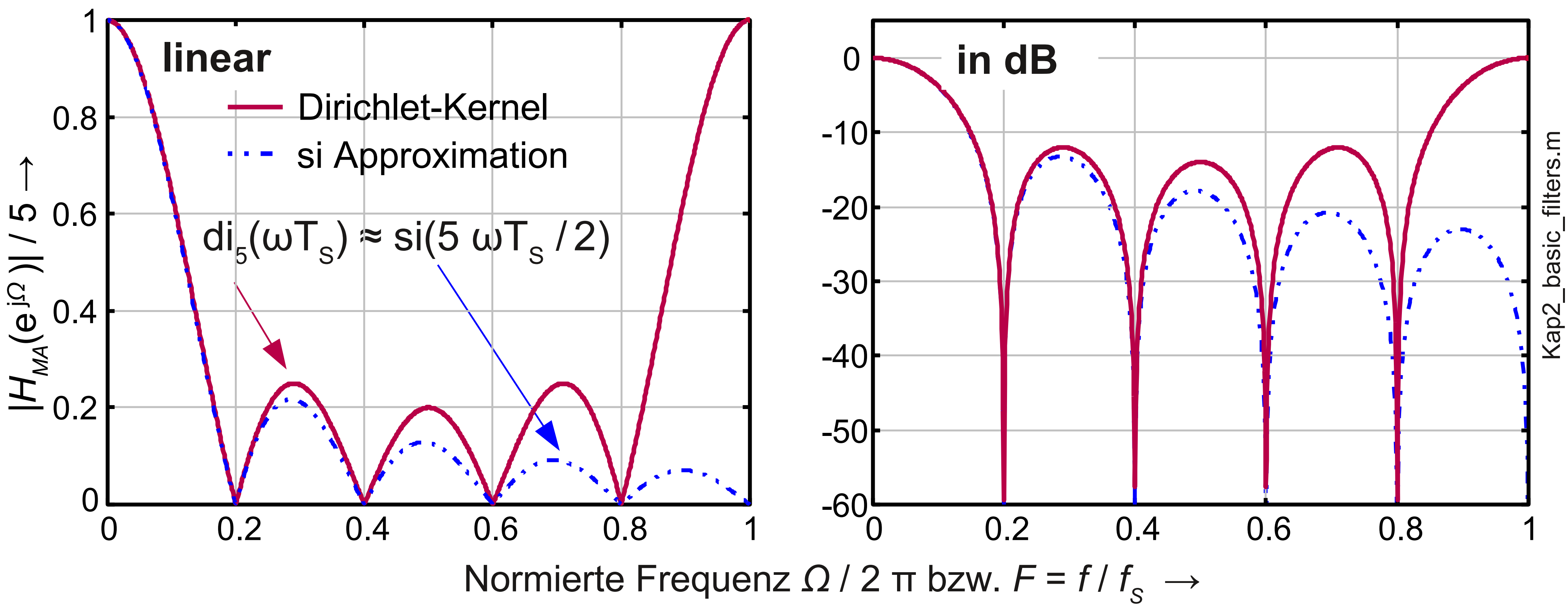
$$H(z) = \frac{1-z^{-5}}{1-z^{-1}} \Rightarrow H(z=e^{j\Omega}) = \frac{1-e^{-j5\Omega}}{1-e^{-j\Omega}} = \frac{e^{+j5\Omega/2}-e^{-j5\Omega/2}}{e^{+j\Omega/2}-e^{-j\Omega/2}} \cdot \frac{e^{-j5\Omega/2}}{e^{-j\Omega/2}}$$

$$= \underbrace{\frac{\sin(5\Omega/2)}{\sin(\Omega/2)}}_{\text{Amplitude}} \underbrace{e^{-j2\Omega}}_{\text{Phase}} = 5 \underbrace{\frac{\sin(5\Omega/2)}{5\sin(\Omega/2)}}_{:= \mathrm{di}_5(\Omega)} \underbrace{e^{-j2\Omega}}_{\text{Phase}}$$

Beispiel: MA Filter mit $N = 4$ Delays und $N + 1 = 5$ Koeffizienten

➔ $h[n]$ ist rect-Puls der Länge $N + 1 = 5$

➔ $H(z)$ hat $N = 4$ Nullstellen im Bereich $f = 0 \dots f_S$ ($F = 0 \dots 1$)



MA-Filter der Ordnung N / Dirichlet-Kernel

Spektrum eines MA-Filters / Rechteckpulses mit Länge $N+1$

$$h[n] = \sum_{m=0}^{N} \delta[n-m] \;\Rightarrow\; H(z) = \sum_{m=0}^{N} z^{-m}$$

$$\Rightarrow\; H(e^{j\Omega}) = \sum_{n=0}^{N} e^{-jn\Omega} = \frac{1-e^{-j(N+1)\Omega}}{1-e^{-j\Omega}} = \frac{e^{-j(N+1)\Omega/2}}{e^{-j\Omega/2}} \cdot \frac{e^{+j(N+1)\Omega/2} - e^{-j(N+1)\Omega/2}}{e^{+j\Omega/2} - e^{-j\Omega/2}}$$

$$= e^{-jN\Omega/2} \cdot \frac{\sin((N+1)\Omega/2)}{\sin(\Omega/2)} = \underbrace{e^{-jN\Omega/2}}_{\text{lineare Phase}} \cdot \underbrace{(N+1)}_{H(f=0)} \cdot \underbrace{\mathrm{di}_{N+1}(\Omega)}_{\text{Amplitudengang}}$$

mit **Dirichlet-Kernel $\mathrm{di}_K(x)$** („periodische si-Funktion“):

$$\mathrm{di}_K(x) := \frac{\sin(K\,x/2)}{K \sin(x/2)} \approx \mathrm{si}(K\,x/2) \text{ für } x/2 \ll 1$$

Achtung: Verschiedene Definitionen für di_K ; hier: Matlab $\texttt{diric}(x, K) \equiv \dfrac{\sin(K\,x/2)}{K \sin(x/2)}$

Berechnung von $\mathrm{di}_K(0)$ über Regel von L'Hospital:

$$\left.\frac{f(x)}{g(x)}\right|_{x \to x_0} = \left.\frac{f'(x)}{g'(x)}\right|_{x \to x_0} \Rightarrow \mathrm{di}_K(\Omega = 0) = \left.\frac{K/2 \cos(K\,\Omega/2)}{K/2 \cos(\Omega/2)}\right|_{\Omega \to 0} = 1$$