

# Formelsammlung DSVB

Mario Felder  
Michi Fallegger

25. Januar 2015



# Inhaltsverzeichnis

<b>1</b>	<b>Digitale Signale im Zeitbereich</b>	<b>5</b>
1.1	Signal Analyse . . . . .	5
1.1.1	Sampling . . . . .	5
1.1.2	Standard digital Signale . . . . .	5
1.1.3	Statistische Signalparameter . . . . .	7
1.1.4	Messverhältnisse und dB . . . . .	7
1.2	Signal Operationen . . . . .	8
1.2.1	Korrelation . . . . .	8
1.2.2	Faltung . . . . .	9
<b>2</b>	<b>Analog-Digital &amp; Digital-Analog Wandlung</b>	<b>11</b>
2.1	Schritte der A/D- und D/A-Wandlung . . . . .	11
2.2	Abtasten und Aliasing . . . . .	12
2.2.1	Abtasten von Tiefpass Signalen . . . . .	12
2.2.2	Aliasing . . . . .	13
2.2.3	Abtasten von Bandpass Signalen . . . . .	14
2.3	Quantisierung von Signalen . . . . .	15
2.3.1	Uniforme Quantisierung . . . . .	15
2.3.2	Quantisierungsrauschen . . . . .	15
<b>3</b>	<b>Digitale Signale im Frequenzbereich</b>	<b>17</b>
3.1	Von Fourier Transformation zur DFT . . . . .	17
3.1.1	Übergang zu diskreter Zeit . . . . .	17
3.1.2	Übergang zu endlichem Messintervall . . . . .	17
3.1.3	Intuitiv . . . . .	18
3.2	Eigenschaften der DFT . . . . .	19

# INHALTSVERZEICHNIS

---

3.2.1	Gültigkeitsbereich der DFT . . . . .	20
3.3	Anwendungs Aspekte . . . . .	21
3.3.1	Zero-Padding . . . . .	21
3.3.2	Fensterfunktion . . . . .	21
3.4	Short-Time DFT . . . . .	22
3.5	Fast Fourier Transformation (FFT) . . . . .	23
3.5.1	Twiddle Faktoren . . . . .	23
3.5.2	Radix-2 FFT . . . . .	24
3.6	Goertzel-Algorithmus . . . . .	25
<b>4</b>	<b>Digtiales Filterdesign</b>	<b>27</b>
4.1	FIR Filter . . . . .	29
4.1.1	Symetrische FIR-Filter . . . . .	29
4.2	IIR Filter . . . . .	31
<b>5</b>	<b>Multirate und Filterbänke</b>	<b>33</b>
5.1	Decimation . . . . .	33
5.2	Interpolation . . . . .	35
5.3	Polyphasen Filter . . . . .	37
5.4	Abtastrate Realisierung . . . . .	37
5.5	Quadratur Spiegel Filter . . . . .	38
5.6	DFT Filterbank . . . . .	39
<b>6</b>	<b>Zufallssignale</b>	<b>43</b>
6.1	Autokorrelation und Spektrum . . . . .	43
6.2	Spectral Shaping in LTI Systems . . . . .	44
6.3	Lineare Modelle für Stochastische Prozesse . . . . .	45
6.3.1	Moving average model . . . . .	45
6.3.2	Autoregressive model . . . . .	46
6.3.3	ARMA model . . . . .	46
<b>7</b>	<b>Optimale Lineare Filter</b>	<b>47</b>
7.1	Wiener Filter . . . . .	47
<b>8</b>	<b>Adaptive Filter</b>	<b>49</b>
8.1	Linear Predictive Coding . . . . .	49
8.2	LMS Algorithmus . . . . .	51
<b>9</b>	<b>Matlab</b>	<b>53</b>

---

# Kapitel 1

## Digitale Signale im Zeitbereich

### 1.1 Signal Analyse

#### 1.1.1 Sampling

Die Sample-Frequenz  $f_S$  ist durch die Sample-Periode  $T_S$  gegeben:

$$f_S = \frac{1}{T_S}$$

Aus dem Signal  $x(t)$  wird durch die Abtastung:

$$x(n \cdot T_S) = x[n]$$

Das Signal  $x[n]$  ist kausal wenn:

$$x[n] = 0 \quad \text{für alle } n < 0$$

#### 1.1.2 Standard digital Signale

Einheitsimpuls oder Diracimpuls:

$$\delta = \begin{cases} 0 & : n \neq 0 \\ 1 & : n = 0 \end{cases}$$

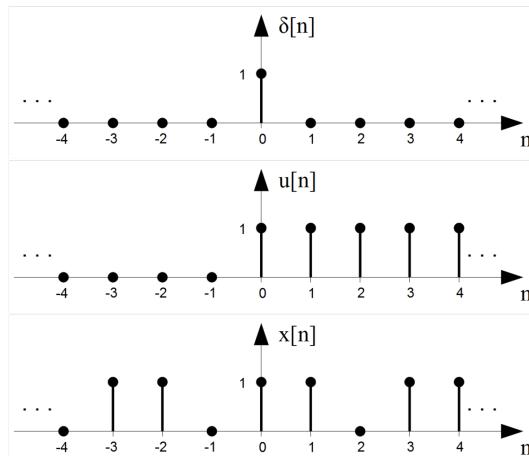
Einheitsschritt:

$$\delta = \begin{cases} 0 & : n < 0 \\ 1 & : n \geq 0 \end{cases}$$

Ein periodisches Signal ist beschrieben durch:

$$x[n] = x[n + T_0/T_S] \quad \text{mit } T_0/T_S = k$$

Ein periodisches Signal muss eine Periodendauer  $T_0$  haben, welche ein ganzzahliges Vielfaches der Sample-Periode  $T_S$  ist.



Komplexe harmonische Sequenz:

$$x[n] = \hat{X} \cdot e^{j2\pi f_0 n T_S}$$

Real- und Imaginäranteil der komplexen Harmonischen:

$$Re\{\} = \hat{X} \cdot \cos 2\pi f_0 n T_S$$

$$Im\{\} = \hat{X} \cdot \sin 2\pi f_0 n T_S$$

### 1.1.3 Statistische Signalparameter

Observations Intervall  $T$ :

$$T = N \cdot T_S$$

Der Mittelwert (**mean value**)  $\mu_x$  repräsentiert den DC-Anteil des Signals  $x[n]$ :

$$\mu_x = \frac{1}{N} \sum_{i=0}^{N-1} x[i]$$

Der quadratische Mittelwert (**quadratic mean value**)  $\rho_x^2$  korrespondiert zur durchschnittlichen Leistung des Signal  $x[n]$  mit DC-Anteil:

$$\rho_x^2 = \frac{1}{N} \sum_{i=0}^{N-1} x[i]^2$$

Die **Varianz**  $\sigma_x^2$  repräsentiert die durchschnittliche AC-Leistung des Signals  $x[n]$ :

$$\sigma_x^2 = \frac{1}{N} \sum_{i=0}^{N-1} (x[i] - \mu_x)^2$$

### 1.1.4 Messverhältnisse und dB

Leistungsverhältnis (**power ratio**):

$$A_{dB} = 10 \cdot \log_{10} \left( \frac{P_1}{P_2} \right)$$

Signal-to-noise ratio (**SNR**):

$$SNR_{dB} = 10 \cdot \log_{10} \left( \frac{P_{signal}}{P_{noise}} \right)$$

Linear $a/b$	Power Power	ratio [dB] Voltage
1/1000	-30	-60
1/100	-20	-40
1/10	-10	-20
1/2	$\approx -3$	$\approx -6$
1	0	0
2/1	$\approx -3$	$\approx 6$
10/1	10	20
100/1	20	40

## 1.2 Signal Operationen

### 1.2.1 Korrelation

Die statische Korrelation (**static correlation**) drückt die Ähnlichkeit zweier Signale  $x[n]$  und  $y[n]$  der selben Länge  $N$  aus:

$$R = \frac{1}{N} \sum_{i=0}^{N-1} x[i]y[i]$$

Je ähnlicher sich die Signale sind, desto grösser der Wert  $R$ .

Die lineare Korrelationsfunktion (**linear correlation function**):

$$r_{xy}[n] = \sum_{i=-\infty}^{\infty} x[i]y[i+n]$$

Die Korrelation ist nicht kommutativ,  $r_{xy}[n] \neq r_{yx}[n]$ .

Resultierende Länge von  $r_{xy}$ :

$$N_{xy} = N_x + N_y - 1$$

Matlabbefehl für Kreuz- und Autokorrelation: **xcorr**

---

### 1.2.2 Faltung

Die Faltung (**convolution**) ist definiert als:

$$z[n] = \sum_{i=-\infty}^{\infty} x[i]y[-i+n]$$

Die Faltung ist kommutativ:

$$z[n] = x[n] * y[n] = y[n] * x[n]$$

Resultierende Länge für  $z[n]$ :

$$N_z = N_x + N_y - 1$$

Bei der zyklischen Faltung (**circular convolution**) müssen die beiden Signale die selbe Länge  $N$  haben, oder mit Zero-Padding auf die selbe Länge gebracht werden:

$$z[n] = x[n] \circledast_N y[n] = y[n] \circledast_N x[n]$$

Lösung durch Matrix:

$$\begin{bmatrix} y[N] & y[N-1] & \dots & y[0] \\ y[0] & y[N] & \ddots & y[1] \\ \ddots & \ddots & \ddots & \ddots \\ y[N-1] & y[N-2] & \dots & y[N] \end{bmatrix} \cdot \begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N] \end{bmatrix} = \begin{bmatrix} z[0] \\ z[1] \\ \vdots \\ z[N] \end{bmatrix}$$

Matlabbefehl für Faltung: **conv**

Matlabbefehl für zyklische Faltung: **convmtx**



# Kapitel 2

## Analog-Digital & Digital-Analog Wandlung

### 2.1 Schritte der A/D- und D/A-Wandlung

**Sample:** Kontinuierliche Signalwerte werden mit der Samplefrequenz  $f_S$  aufgezeichnet. Dies erzeugt eine Sequenz von diskreten Signalwerten.

**Quantize:** Die diskreten Signalwerte werden einer bestimmten Anzahl Quantisierungsleveln zugeordnet.

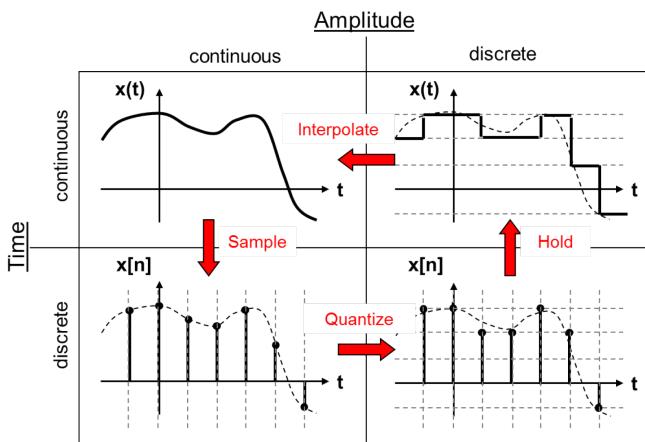
**Code:** Die quantisierten Abtastwerte können verwendet werden, um die erhaltene Pulsfolge zu modulieren (Pulse Code Modulation PCM). Meistens wird die Signalverarbeitung direkt mit den quantisierten Abtastwerten vorgenommen, so dass diese ohne Modulation gespeichert werden. Dazu wird die Repräsentierung der Quantisierungsleveln benötigt.

Die Digital-Analog Wandlung enthält folgende Schritte:

**Decode:** Die digitalen Werte werden in einer für die Digital-Analog Wandlung repräsentativer Form benötigt.

**Hold:** Das diskrete Signal muss über die Sampleperiode  $T_S$  konstant gehalten werden, ein Treppen ähnlicher Output entsteht.

**Interpolate:** Das kontinuierliche Treppensignal wird durch Mittelwerte von Tiefpass-Filtersn geglättet.



## 2.2 Abtasten und Aliasing

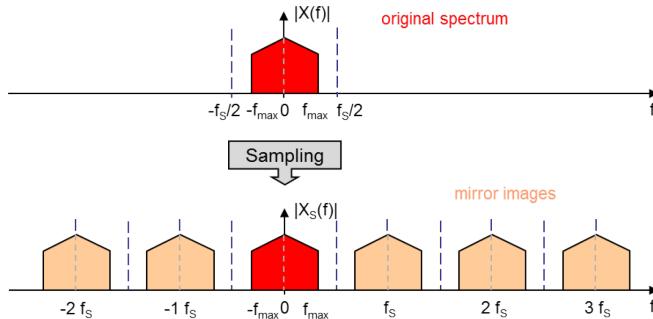
### 2.2.1 Abtasten von Tiefpass Signalen

Mathematisch wird die Abtastung des Signals  $x(t)$  durch eine Multiplikation mit Dirac-Impulsen der Periode  $T_S$  dargestellt:

$$x_s(t) = \sum_{n=-\infty}^{\infty} x(t) \cdot \delta(t - nT_S)$$

Das Frequenzspektrum des abgetasteten Singals ist:

$$X_S(f) = \frac{1}{T_S} \sum_{k=-\infty}^{\infty} X(f - kf_S)$$



Um das Signal rekonstruieren zu können, müssen die Spiegelfrequenzen von  $X(f)$  mit einem Tiefpassfilter unterdrückt werden. Das geht nur, wenn die grösste Frequenz  $f_{max}$  des Signals kleiner als die halbe Abtastfrequenz  $f_S/2$  ist.

$$f_S > 2 \cdot f_{max}$$

### 2.2.2 Aliasing

Aliasing entsteht, wenn das Abtasttheorem verletzt wird. Die Frequenzanteile über  $f_S/2$  werden in das Basisband gespiegelt und überlagern mit den gewünschten Frequenzen. Das analoge Signal kann nicht rekonstruiert werden.

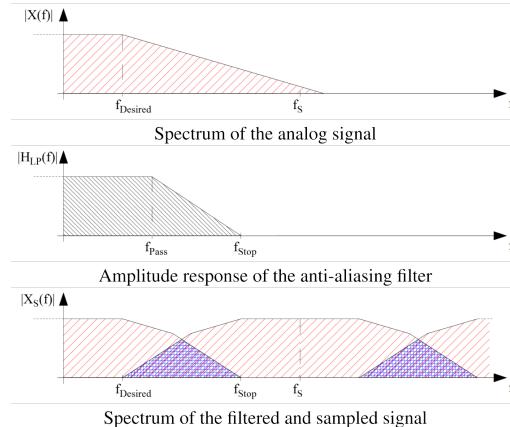
Dem kann mit einer Tiefpassfilterung entgegengewirkt werden. Der Filter hat folgende Spezifikationen:

$$f_{pass} \geq f_{desired}$$

$$f_{stop} \leq f_S - f_{desired}$$

## KAPITEL 2. ANALOG-DIGITAL & DIGITAL-ANALOG WANDLUNG

---



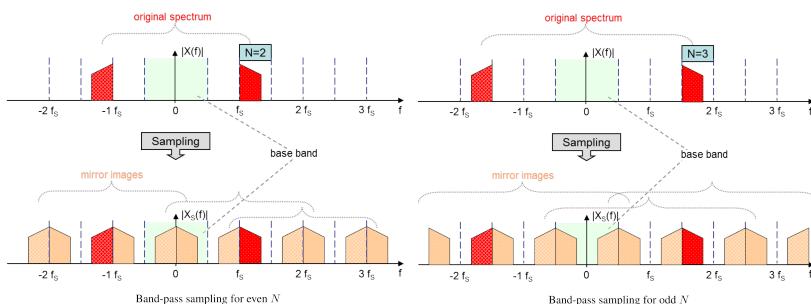
### 2.2.3 Abtasten von Bandpass Signalen

Das Abtasttheorem wird folgendermassen angepasst (für  $N \geq 1$ ):

$$\frac{2 \cdot f_{min}}{N} \geq f_S \geq \frac{2 \cdot f_{max}}{N + 1}$$

Für ungerade  $N$  erscheint das originale Spektrum invertiert im Basisband. Die originale Struktur des Spektrum wird zurückgewonnen durch invertierung jedes zweiten Samples im Zeitbereich:

$$\tilde{x}[n] = (-1)^n \cdot x[n]$$



## 2.3 Quantisierung von Signalen

### 2.3.1 Uniforme Quantisierung

Der Quantisierungsschritt (**quantization step**)  $\Delta$  ist gegeben durch die Auflösung mit  $W$  bits und der dynamischen Reichweite  $R$  des abgetasteten Signals  $x[n]$ :

$$\Delta = \frac{R}{2^W}$$

Durch das Abbilden eines amplitudenkontinuierlichen Signals auf eine endliche Anzahl von Rekonstruktionsleveln können zweier Fehler entstehen:

**Clipping:** Werte von  $x[n]$  ausserhalb des Bereichs  $R$  werden mit dem maximum bzw. minimum Rekonstruktionslevel dargestellt.

**Quantization error  $\epsilon$ :** Dieser Fehler tritt immer auf und kann nicht verhindert werden. Die Grösse des Fehlers ist gegeben durch die Quantisierungsgrösse  $\Delta$ .

Für den mid-tread quantizer (runden zum nächsten Wert):

$$-\Delta/2 < \epsilon \leq \Delta/2$$

Für den mid-rise quantizer:

$$-\Delta < \epsilon \leq 0$$

### 2.3.2 Quantisierungsrauschen

Der Quantisierungsfehler zeigt sich als Rauschen überlagert zum quantisierten Signal:

$$\epsilon[n] = x_q[n] - x[n]$$

Die Leistung  $P_\epsilon$  des Quantisierungsrauschens (**quantization noise**) ist:

$$P_\epsilon = \sigma_\epsilon^2 = \int_{-\infty}^{\infty} (\epsilon - \sigma_\epsilon)^2 \cdot p(\epsilon) d\epsilon = \frac{\Delta^2}{12}$$

## KAPITEL 2. ANALOG-DIGITAL & DIGITAL-ANALOG WANDLUNG

---

wobei  $p(\epsilon)$  die Wahrscheinlichkeitsdichte ist. Angenommen die Werte von  $\epsilon[n]$  sind statistisch unkorreliert und uniform verteilt über den Intervall  $(-\Delta/2, \Delta/2]$  ist die Wahrscheinlichkeitsdichte für einen mid-tread quantizer:

$$p(\epsilon) = 1/\Delta$$

Der Erwartungswert des Quantisierungsfehlers ist somit:

$$\mu(\epsilon) = \int_{-\infty}^{\infty} \epsilon \cdot p(\epsilon) d\epsilon = \frac{1}{2\Delta} \epsilon^2 \Big|_{-\Delta/2}^{\Delta/2} = 0$$



Das SNR ist:

$$SNR = \frac{P_x}{P_\epsilon} = 2^{2W} \cdot \frac{12P_x}{R^2}$$

$$\begin{aligned} SNR_{dB} &= 10 \cdot \left( \log_{10} 2^{2W} + \log_{10} \frac{12P_x}{R^2} \right) \\ &\approx 6W + 10 \cdot \log_{10} \frac{12P_x}{R^2} \end{aligned}$$

Bei einem harmonischen Inputsignal  $x[n]$  ergibt sich folgendes SNR für eine uniforme Quantisierung:

$$SNR_{db} = 6W + 1.76 \approx 6W$$

# Kapitel 3

## Digitale Signale im Frequenzbereich

### 3.1 Von Fourier Transformation zur DFT

#### 3.1.1 Übergang zu diskreter Zeit

Die zeitdiskrete Fourier Transformation (DTFT) kann berechnet werden mit:

$$X(\Omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\Omega n}$$

mit der normalisierten Winkelfrequenz:

$$\Omega = 2\pi f T_S = 2\pi \frac{f}{f_S}$$

Die DTFT produziert ein  $2\pi$ -periodisches, kontinuierliches Spektrum.

#### 3.1.2 Übergang zu endlichem Messintervall

Wenn nur über  $N$  Abtastpunkte eine Fourier Analyse gemacht werden soll, ist die kleinste Frequenz, welche aufgenommen werden kann:

$$f_1 = \frac{1}{T} = \frac{1}{N \cdot T_S} = \frac{f_S}{N}$$

Die Diskrete Fourier Transformation (DFT) kann geschrieben werden als:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi n \frac{k}{N}} \quad k = 0, 1, 2, \dots, N-1$$

Die DFT ist beschränkt zu einer maximalen Frequenz von:

$$(N-1) \frac{f_S}{N}$$

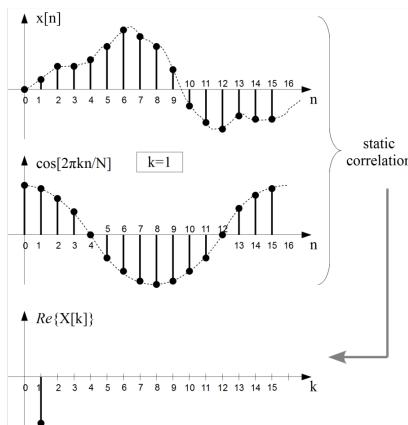
Die inverse diskrete Fourier Transformation (IDFT) ist gegeben durch:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j2\pi n \frac{k}{N}} \quad n = 0, 1, 2, \dots, N-1$$

		Signal	
		periodic	aperiodic
Time	continuous	Fourier Series Discrete Line Spectrum $c_n$	Fourier Transform Continuous Frequency Spectrum $X(f)$
	discrete	Discrete Fourier Transform (DFT) Discrete periodic Line spectrum $X[k]$	Discrete-Time Fourier Transform (DTFT) Continuous periodic Frequency Spectrum $X[\Omega]$

### 3.1.3 Intuitiv

$$X[k] = \underbrace{\sum_{n=0}^{N-1} x[n] \cos(2\pi n \frac{k}{N})}_{\Re\{X[k]\}} + j \cdot \underbrace{\sum_{n=0}^{N-1} x[n] (-1) \sin(2\pi n \frac{k}{N})}_{\Im\{X[k]\}}$$



## 3.2 Eingenschaften der DFT

### Periodizität

Das Spektrum der DFT ist  $f_S$ -periodisch

$$X[k] = X[k + N]$$

Dementsprechend ist IDFT periodisch mit  $T = NT_S$

$$x[n] = x[n + N]$$

### Symmetrie

Die DFT eines realwertigen Signals ist symmetrisch um den Punkt  $k = N/2$

$$X[N/2 + m] = X^*[N/2 - m]$$

### Zeit/Frequenz Verschiebung

Verschiebung einer periodischen Zeitsequenz um  $n_0$  hat einen linearen Phasen Offset bei allen Spektralwerten zur Folge

$$x[n + n_0] \quad \circ \bullet \quad e^{j2\pi n_0 \frac{k}{N}} \cdot X[k]$$

Die Multiplikation mit einem komplexen Exponent hat eine konstante Frequenzverschiebung zur Folge

$$e^{j2\pi k_0 \frac{n}{N}} \cdot x[n] \quad \circ \text{---} \bullet \quad X[k - k_0]$$

### Modulation

Konsequenz der Frequenzverschiebung ist die Modulation

$$\cos\left(2\pi k_0 \frac{n}{N}\right) \cdot x[n] \quad \circ \text{---} \bullet \quad \frac{1}{2} (X[k + k_0] + X[k - k_0])$$

### Parseval Theorem

Zwischen Signalwerten  $x[k]$  und Fourier Koeffizienten  $X[k]$  besteht folgende Beziehung:

$$\frac{1}{N} \sum_{n=0}^{N-1} x[n]^2 = \sum_{k=0}^{N-1} \left| \frac{X[k]}{N} \right|^2$$

### Zusammenhang von Faltung und Multiplikation

Punktweise Multiplikation zweier DFT Spektren  $X[k]$  und  $Y[k]$  im Frequenzbereich entspricht der zyklischen Faltung von  $x[n]$  und  $y[n]$  im Zeitbereich

$$x[n] \circledast_N y[n] \quad \circ \text{---} \bullet \quad X[k] \cdot Y[k]$$

#### 3.2.1 Gültigkeitsbereich der DFT

$$X[k] = X[\Omega]|_{\Omega=2\pi \frac{k}{N}}$$

$x[n]$  periodisch:

Der Messintervall  $NT_S$  ist ein ganzzahliger Vielfache der Periodendauer von  $x[n]$ .

$x[n]$  aperiodisch:

Alle Werte von  $x[n]$  ausserhalb dem Bereich  $0 \leq n < N$  sind Null.

## 3.3 Anwendungs Aspekte

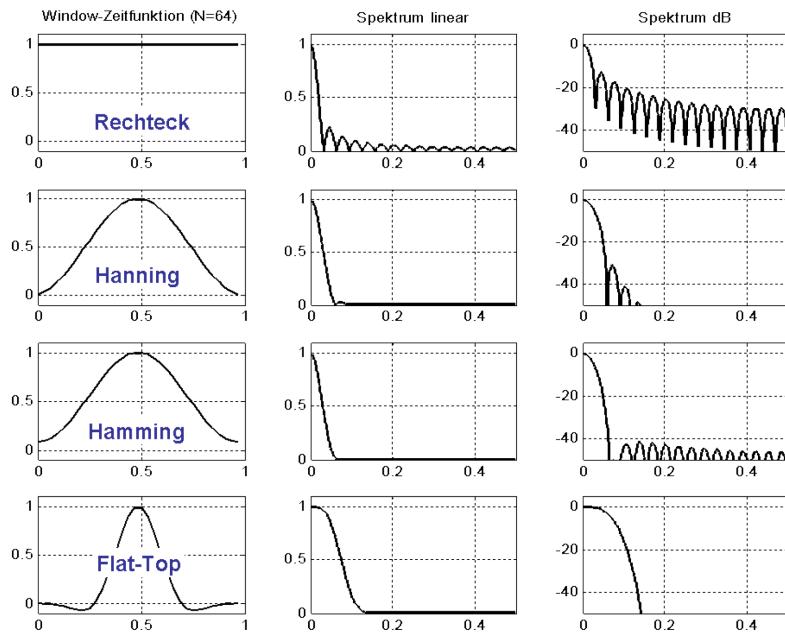
### 3.3.1 Zero-Padding

Um eine bessere Interpolation zwischen den  $N$  Frequenzpunkten in der DFT zu bekommen, kann das Signal im Zeitbereich mit Nullen aufgefüllt werden. Das Spektrum  $X(\Omega)$  wird nicht verändert, allerdings stehen zusätzliche Abtastpunkte auf der Achse der normalisierten Winkelgeschwindigkeit  $\Omega$  zur Verfügung.

### 3.3.2 Fensterfunktion

Die DFT wendet per Definition ein Rechteckfenster an, um die  $N$  Samples auszuschneiden. Es können andere Fensterfunktionen auf das Signal  $x[n]$  angewendet werden, allerdings gelten folgende Punkte:

- Je schmäler die Hauptkeule im Spektrum des Fensters, desto höher ist die spektrale Auflösung für  $X[k]$
- Je stärker die Dämpfung der Nebenkeulen im Spektrum des Fensters, desto besser die Unterdrückung der Leackage in  $X[k]$



### 3.4 Short-Time DFT

Wenn die Entwicklung des Frequenzspektrums über die Zeit interessiert, werden kontinuierliche Berechnungen des Spektrums von kurzen Signalsektoren durchgeführt. Da diese kurzen Sektoren meistens nicht ein ganzzahliges Vielfache der Periode ist, werden Fensterfunktionen angewendet.

Bei der Länge  $N$  des Fensters ergibt sich der Kompromiss von:

- grosse spektrale Auflösung ( $N$  gross)
- grosse zeitliche Auflösung ( $N$  klein)

Um beides zu erreichen, können die DFT-Fenster überlappt werden um maximal  $N - 1$ .

## 3.5 Fast Fourier Transformation (FFT)

Geschwindigkeitsvergleich FFT-DFT:

$$\text{speedup\_factor}_{\text{FFT}} = \frac{8N - 2}{5 \cdot \log_2 N} \approx 1.5 \frac{N}{\log_2 N}$$

Dabei ist  $N = 2^L$  mit einem ganzzahligen Wert für  $L$ .

DFT

$$[N^2]_{\text{cpl.Mul.}} + [N^2 - N]_{\text{cpl.Add.}}$$

FFT

$$\left[ \frac{N}{2} \cdot \log_2 N \right]_{\text{cpl.Mul.}} + [N \cdot \log_2 N]_{\text{cpl.Add.}}$$

Die Frequenzauflösung ist gegeben durch

$$\Delta f = \frac{f_s}{N}$$

### 3.5.1 Twiddle Faktoren

Definition:

$$W_N = e^{-j \frac{2\pi}{N}}$$

#### Periodizität

$W_N$  kann nur für  $N$  verschiedene Zahlen ausgewertet werden. Der Twiddle Faktor ist somit  $N$ -periodisch

$$W_N^{k+N} = W_N^k$$

#### Symmetrie

Abgesehen vom Vorzeichen nimmt  $W_N$  nur  $N/2$  verschiedene Werte an.

$$W_N^{k+N/2} = -W_N^k$$

### 3.5.2 Radix-2 FFT

Die DFT wird mit dem Twiddle Faktor umgeschrieben:

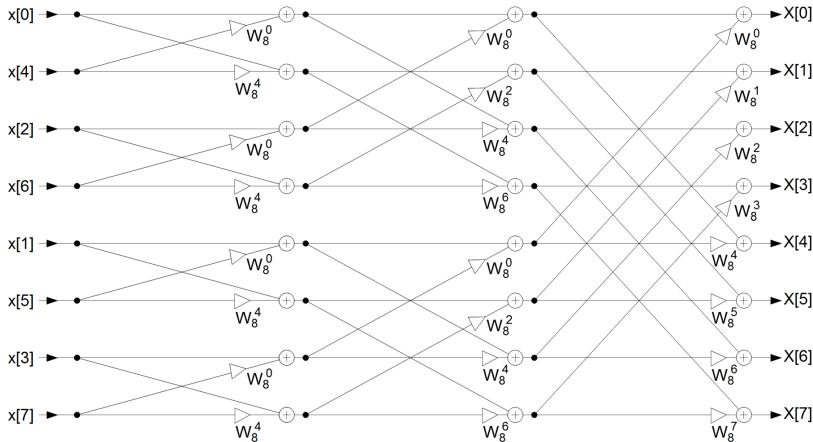
$$\begin{aligned} X[k] &= \sum_{n=0}^{N-1} x[n] W_N^{nk} \\ &= \underbrace{\sum_{n \text{ even}} x[n] W_N^{nk}} + \underbrace{\sum_{n \text{ odd}} x[n] W_N^{nk}} \end{aligned}$$

Es können zwei neue Sequenzen  $x_1[n]$  und  $x_2[n]$  mit geraden und ungeraden  $n$  von der Länge  $N/2$  erzeugt werden.

$$X[k] = \underbrace{\sum_{n=0}^{N/2-1} x_1[n] W_{N/2}^{nk} + W_N^k}_{x_1[\tilde{k}]} \cdot \underbrace{\sum_{n=0}^{N/2-1} x_2[n] W_{N/2}^{nk}}_{x_2[\tilde{k}]}$$

wobei

$$\tilde{k} = k \mod N/2$$



Benötigter Speicher der FFT bei in-place:  $2N$  (+ $N/2$  für Twiddle Faktoren)  
Index bit-reversed in Matlab: `bitrevorder`

## 3.6 Goertzel-Algorithmus

Definition:

$$X[k] = y_k[n]|_{n=N} = \sum_{i=0}^{N-1} x[i] W_N^{-k(N-i)}$$

Differenzengleichung:

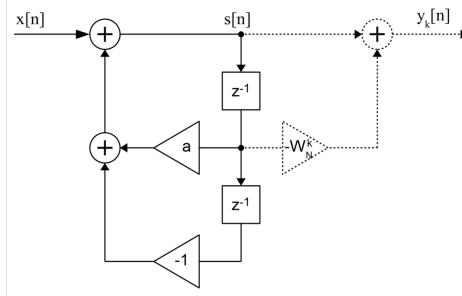
$$s[n] = x[n] + a \cdot s[n-1] - s[n-2]$$

$$y_k[n] = s[n] - W_N^k \cdot s[n-1]$$

mit

$$a = 2 \cos 2\pi \frac{k}{N}$$

$s[n]$  muss für alle Zeitpunkte  $n = 0, 1, \dots, N$  berechnet werden,  $y_k[n]$  nur für  $n = N$ .



Der Goertzel-Algorithmus gibt die DFT spectral component bei der Frequenz

$$f_k = k \frac{f_s}{N}$$

Mit dem Parseval Theorem kann der Leistungsgehalt  $P_k$  eines realwertigen Signals  $x[n]$  bei einer Frequenz von  $f_k$  ermittelt werden:

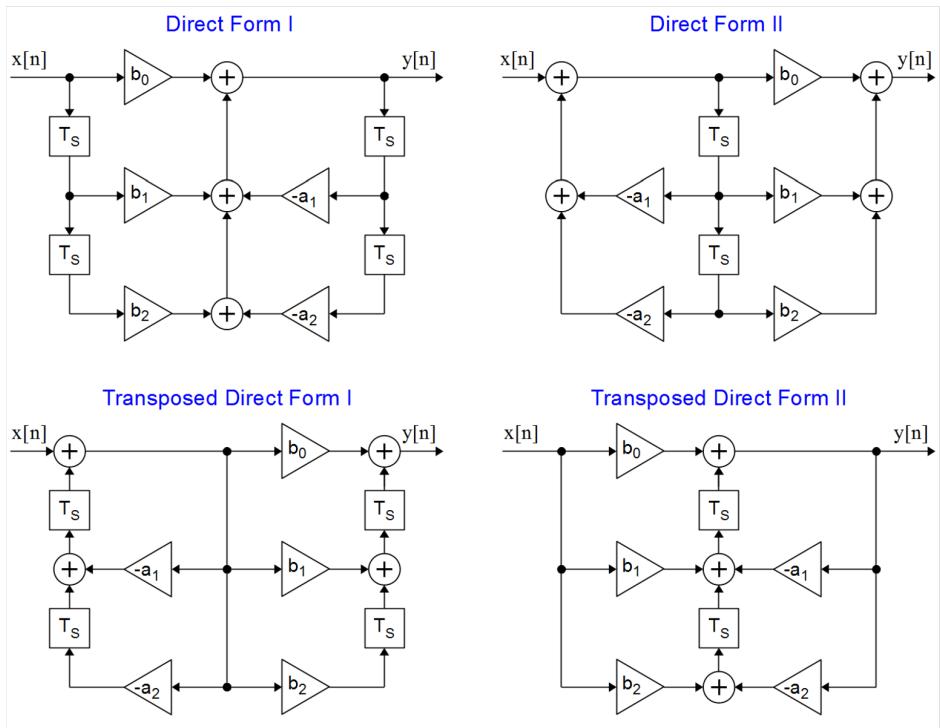
$$P_k = 2 \left| \frac{X[k]}{N} \right|^2 = \frac{2}{N^2} (\Re\{X[k]\}^2 + \Im\{X[k]\}^2)$$





# Kapitel 4

## Digtiales Filterdesign



## 4.1 FIR Filter

FIR (finite impulse response) Filter der Ordnung  $N$  hat folgende Übertragungsfunktion:

$$H(z) = b_0 + b_1 z^{-1} + \dots + b_N z^{-N}$$

Die Impulsantwort ist  $N + 1$  Zeitschritte lang und entspricht den Koeffizienten von  $H(z)$ :

$$h[n] = \{b_0, b_1, \dots, b_N, 0, 0, \dots\}$$

Stabilität:

Da alle Pole bei  $z = 0$  liegen, sind FIR-Filter per Definition stabil.

Lineare Phase:

Mit einem FIR-Filter ist es einfach möglich, eine lineare Phasenübertragung zu realisieren.

Implementation

Die Realisierung von FIR-Filter in HW oder SW ist straightforward und unkritisch.

### 4.1.1 Symmetrische FIR-Filter

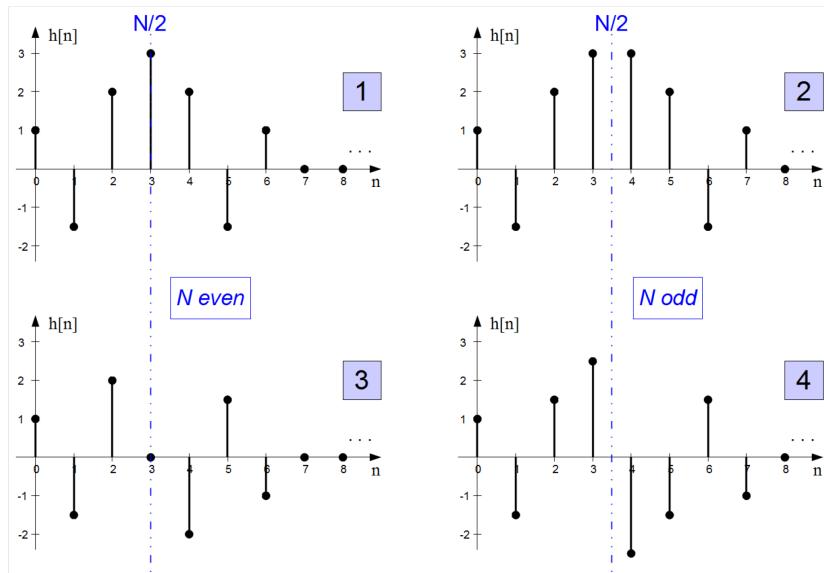
Ein FIR-Filter ist symmetrisch wenn

$$b_i = \pm b_{N-i} \quad i = 0, 1, \dots, N$$

Wenn die Gleichung mit "+erfüllt ist, spricht man von einem symmetrischen FIR-Filter, wenn sie mit "erfüllt ist, spricht man von einem asymmetrischen.

Alle symmetrischen Filter haben eine lineare Phasenantwort im Pass-Band. Das heißt, sie haben ein konstantes Gruppen-Delay:

$$\tau_g = \frac{N}{2} \cdot T_S$$



Type	Symmetry	Order $N$	$ H(f=0) $	$ H(f=f_S/2) $	$H(\Omega)$
1 $h[n] = h[N-n]$ (symmetric)	even	any	any		$e^{-j\Omega \frac{N}{2}} \cdot H_{zp}(\Omega)$
2	odd	any	0		
3 $h[n] = -h[N-n]$	even	0	0		$e^{-j(\Omega \frac{N}{2} - \frac{\pi}{2})} \cdot H_{zp}(\Omega)$
4 (anti-symmetric)	odd	0	any		

## 4.2 IIR Filter

criteria	FIR	IIR
<b>stability</b>	always stable	not guaranteed, must be verified for actual implementation
<b>filter performance / filter order</b>	higher order for same amplitude response	lower filter order for same amplitude response
<b>phase response</b>	exactly linear possible	linear phase not possible, introduces phase distortion
<b>Arbitrary frequency response</b>	easy to realize	more difficult to realize
<b>design process</b>	optimum design only with CAD, manual design possible	basic filter characteristics can be designed manually
<b>design with analog filter model</b>	no direct conversion possible	standard design method
<b>finite word width effects</b>	relatively low	sensitive to quantization and scaling
<b>implementation</b>	non-critical in HW and SW, direct support in DSPs	more complicated, requires detailed analysis of word widths

Umformung vom p-Plan zum z-Plan

$$z = e^{pT_S}$$

Der IIR Filter kann als biquads implementiert werden. Dabei wird die Übertragungsfunktion in mehrere aufgeteilt, welche jeweils einen Konjunktions- Komplexen Pol haben:

$$H(z) = K \cdot \frac{(z - z_1)(z - z_1^*)}{(z - p_1)(z - p_1^*)} \cdot \dots \cdot \frac{(z - z_L)(z - z_L^*)}{(z - p_L)(z - p_L^*)}$$

Bei einer Fix-Point Implementierung mit  $W$  bits ( $b_k$ ) und  $F$  Nachkommastellen ist der dezimale Wert:

$$D_{ufix} = \sum_{k=0}^{W-1} b_k \cdot 2^{k-F}$$

Für Zahlen im Zweierkomplement:

$$D_{fix} = -b_{W-1} \cdot 2^{W-F-1} + \sum_{k=0}^{W-2} b_k \cdot 2^{k-F}$$

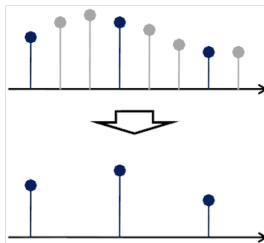


# Kapitel 5

## Multirate und Filterbänke

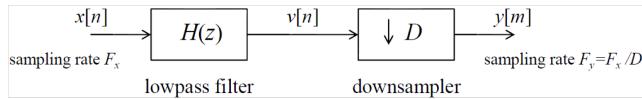
### 5.1 Decimation

Um ein Signal donwzusampeln, wird einfach nur jedes  $D$ te Sampel verwendet.



Dabei muss beachtet werden, dass das Abtasttheorem noch eingehalten wird. Das Signal muss zuerst Tiefpass gefiltert werden. Die Frequenzantwort des TP ist idealerweise:

$$H(\Omega) = \begin{cases} 1 & \text{if } \Omega \in [-\pi/D, \pi/D] \\ 0 & \text{otherwise} \end{cases}$$

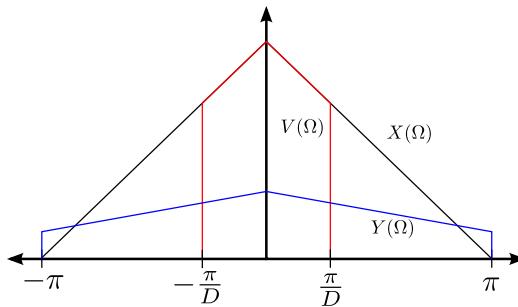


Das Resultat kann beschrieben werden als  $y[m] = v[mD]$ . Das Frequenzspektrum wird um den Faktor  $D$  gespreizt. Für einen idealen TP-Filter gilt:

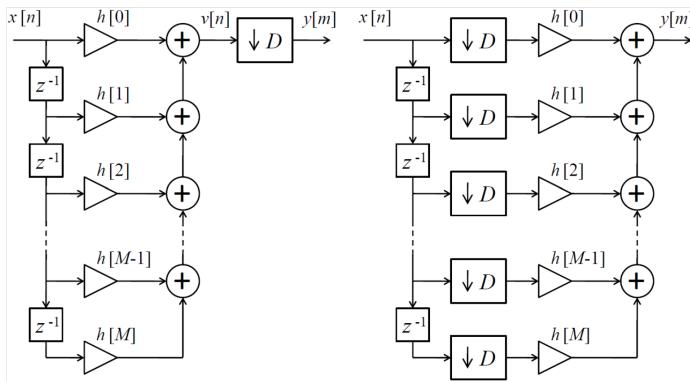
$$Y(\Omega) = \frac{1}{D} V(\Omega/D)$$

Allgemein:

$$Y(\Omega) = \frac{1}{D} \sum_{d=0}^{D-1} V(\Omega/D - 2\pi d/D)$$

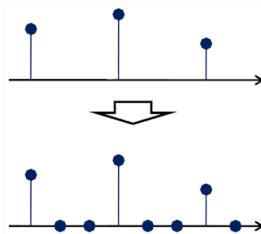


Als TP-Filter kann ein FIR Filter verwendet werden. Eine direkte Implementierung ist nicht effektiv, da  $D - 1$  vom Filter berechnete Werte vom Downsampler weggeworfen werden. Mit dem Downsampler vor dem Filter müssen weniger Berechnungen durchgeführt werden.



## 5.2 Interpolation

Bei einem Upsampler mit dem Faktor  $I$  werden zwischen zwei Samples jeweils  $I - 1$  Nullen eingefügt.



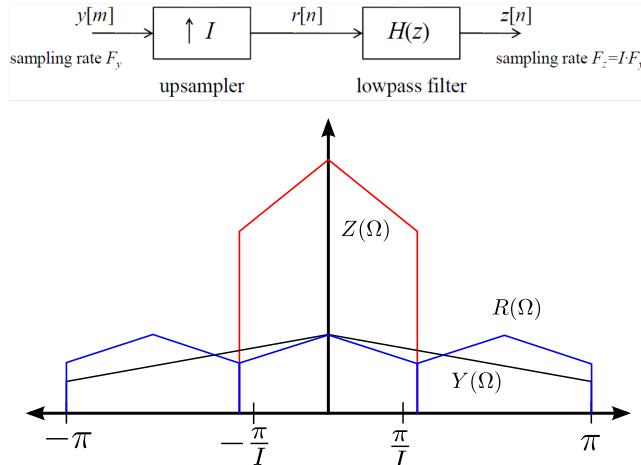
$$r[n] = \begin{cases} y[n/I] & \text{if } n \in [0, \pm I, \pm 2I, \dots] \\ 0 & \text{otherwise} \end{cases}$$

Die Abtastfrequenz  $F_z$  ist  $I$ -mal höher als die Abtastfrequenz  $F_y$  von  $y[m]$ . Das Spektrum des upgesampelten Signal ist:

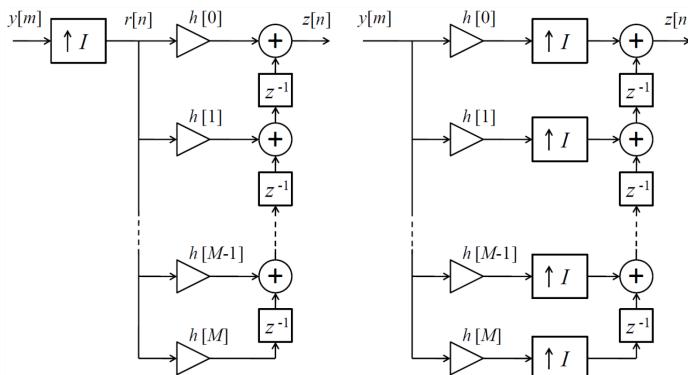
$$\begin{aligned} R(\Omega) &= \sum_{n=-\infty}^{\infty} r[n] e^{-j\Omega n} \\ &= \sum_{m=-\infty}^{\infty} r[m] e^{-j\Omega m} \\ &= Y(I\Omega) \end{aligned}$$

Damit das Frequenzspektrum von  $y[m]$  nicht periodisch mit der Periode von  $2\pi/I$  ist, ist eine Tiefpassfilterung nach dem Upsampling notwendig. Der ideale TP-Filter hat die Frequenzantwort:

$$H(\Omega) = \begin{cases} I & \text{if } \Omega \in [-\pi/I, \pi/I] \\ 0 & \text{otherwise} \end{cases}$$



Ein FIR oder IIR Filter kann als TP-Filter verwendet werden. Eine direkte Implementierung ist ineffektiv, da der Filter auch auf die Nullwerte angewandt wird.



### 5.3 Polyphasen Filter

Die um  $M$  downgesampelte Impulsantwort  $h[k]$  ist definiert als:

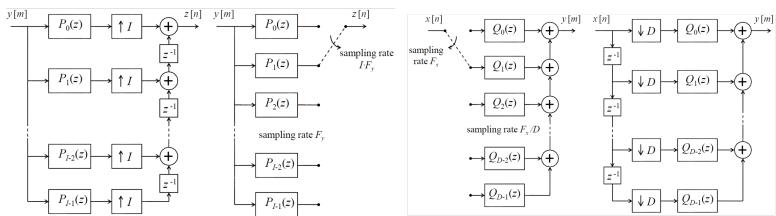
$$p_i[k] = h[kM + i] \quad i = 0, 1, \dots, M - 1$$

Dazugehörende z-Transformation:

$$P_i(z) = \sum_{k=-\infty}^{\infty} p_i[k]z^{-k} \quad i0, 1, \dots, M - 1$$

Somit

$$H(z) = \sum_{i=0}^{M-1} z^{-1} P_i(z^M)$$



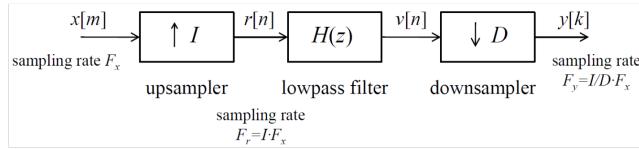
### 5.4 Abtastrate Realisierung

Wenn ein up- bzw. downsample Faktor gefordert ist, welcher nicht ganzzahlig ist, kann dieser mit

$$\frac{I}{D}$$

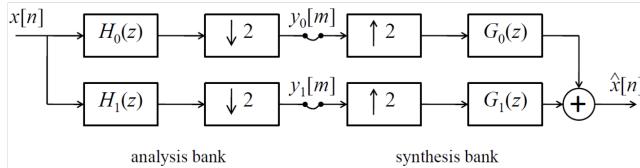
dargestellt werden.

Wenn zuerst der Downampler kommt, gehen Informationen verloren. Andersrum entsteht eine hohe Abtastfrequenz dazwischen. Es ist jedoch vorzuziehen, zuerst Upsemplen, danach Downsamplen. So kann der Tiefpass der Interpolation mit dem Tiefpass der Decimation kombiniert werden.



## 5.5 Quadratur Spiegel Filter

Um den Datenverlust beim Downsampling zu kompensieren, kann das Signal über zwei Kanäle übertragen werden. Der eine Kanal filtert das Signal mit einem TP  $H_0$ , der Andere mit einem HP  $H_1$ .



Die DTFT der zwei generierten Signalen  $y_0[m]$  und  $y_1[m]$  sind:

$$Y_{0/1} = \frac{1}{2} \left( H_{0/1} \left( \frac{\Omega}{2} \right) X \left( \frac{\Omega}{2} \right) + H_{0/1} \left( \frac{\Omega}{2} - \pi \right) X \left( \frac{\Omega}{2} - \pi \right) \right)$$

Das Spektrum des synthetisierten Signals  $\hat{x}[n]$  ist

$$\begin{aligned} \hat{X}(\Omega) &= \frac{1}{2} (H_0(\Omega)G_0(\Omega) + H_1(\Omega)G_1(\Omega)) \cdot X(\Omega) \\ &\quad + \underbrace{\frac{1}{2} (H_0(\Omega - \pi)G_0(\Omega) + H_1(\Omega - \pi)G_1(\Omega)) \cdot X(\Omega - \pi)}_{\text{alias term}} \end{aligned}$$

Um keinen Aliasterm zu haben müssen die Bedingungen  $G_0(\Omega) = H_1(\Omega - \pi)$  und  $G_1(\Omega) = -H_0(\Omega - \pi)$  erfüllt sein:

$$\begin{aligned} H_0(\Omega) &= H(\Omega) \\ H_1(\Omega) &= H(\Omega - \pi) \\ G_0(\Omega) &= H(\Omega) \\ G_1(\Omega) &= -H(\Omega - \pi) \end{aligned}$$

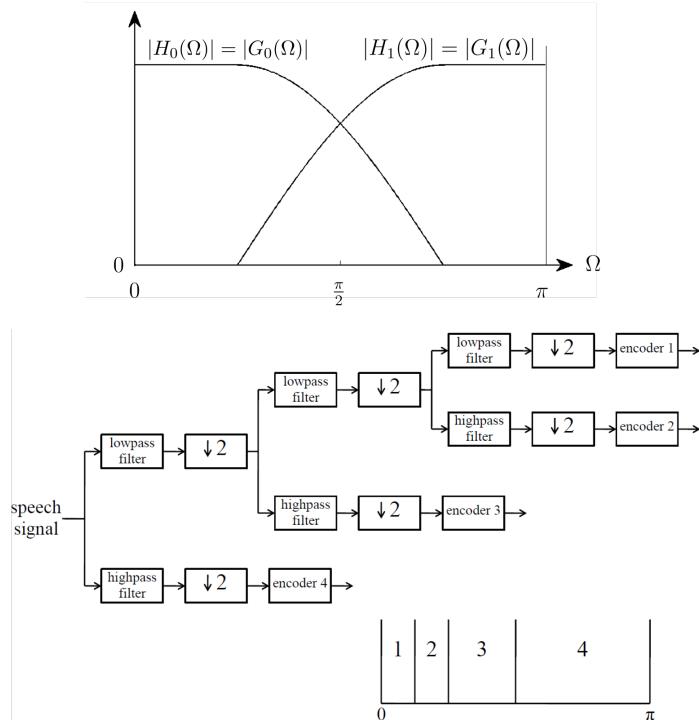
Weiter gilt

$$\hat{X}(\Omega) = T(\Omega)X(\Omega)$$

mit

$$T(\Omega) = \frac{1}{2}(H^2(\Omega) - H^2(\Omega - \pi))$$

Um perfekte Rekonstruktion zu haben, muss  $T(\Omega)$  eine Allpass Charakteristik mit linearer Phasenverschiebung aufzeigen. Dies kann am besten mit einem symmetrischen FIR Filter erreicht werden.

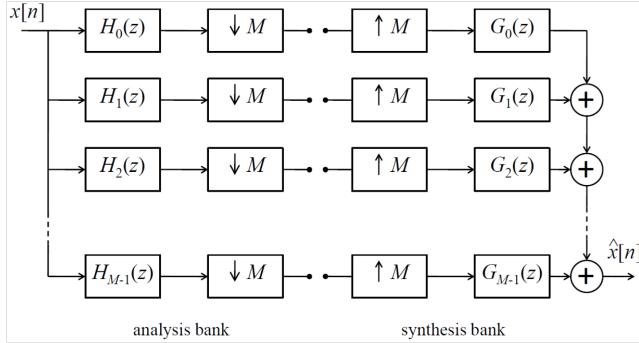


## 5.6 DFT Filterbank

**critical sampling:** Die Anzahl der Filter  $H_0(z), H_1(z), \dots, H_{M-1}$  entspricht dem Downsampling  $M$ .

**oversampling:** Anzahl Kanäle ist grösser als der Downsampling Faktor.

**undersampling:** Anzahl Kanäle ist kleiner als der Downsampling Faktor.



Prototype Filter  $H(z)$ :

$$H_l(z) = H(z \cdot e^{-j2\pi l/M}) \quad l = 0, 1, \dots, M-1$$

$$G_l(z) = G(z \cdot e^{-j2\pi l/M}) \quad l = 0, 1, \dots, M-1$$

Frequenzgang:

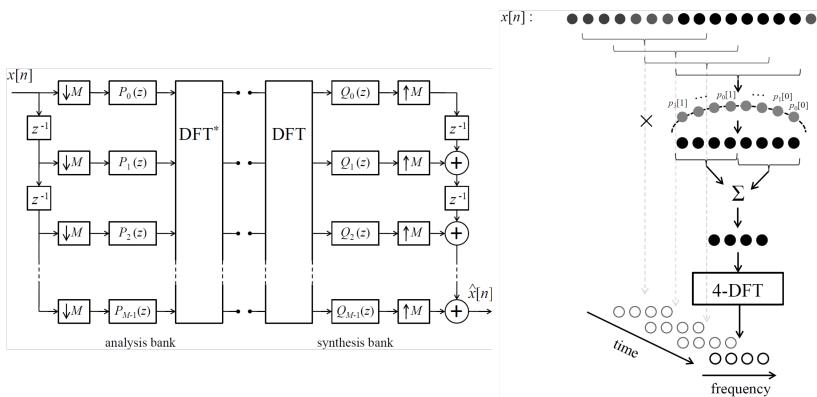
$$H_l(\Omega) = H(\Omega - 2\pi l/M)$$

Impulsantwort:

$$h[k] = \begin{cases} 1 & \text{if } k \in \{0, 1, \dots, M-1\} \\ 0 & \text{otherwise} \end{cases} \quad g[k] = \begin{cases} \frac{1}{M} & \text{if } k \in \{0, 1, \dots, M-1\} \\ 0 & \text{otherwise} \end{cases}$$

Mit polyphase

$$H_l(z) = \sum_{i=0}^{M-1} e^{j2\pi li/M} \cdot z^{-i} P_i(z^M) \quad l = 0, 1, \dots, M-1$$





# Kapitel 6

## Zufallssignale

### 6.1 Autokorrelation und Spektrum

Mittelwert eines Zufallssignals  $x[n]$  ist

$$m_x = E\{x[n]\}$$

Autokorrelation

$$\gamma_{xx}[m] = E\{x^*[n] \cdot x[n+m]\}$$

Wobei  $\gamma_{xx}[0]$  die Leistung  $P_x$  repräsentiert.

Für echte Zufallssignale gilt

$$\gamma_{xx}[m] = \gamma_{xx}^*[-m]$$

Signale, welche den Mittelwert nicht bei Null haben, werden mit der Autokovarianz charakterisiert

$$c_{xx} = E\{(x[n] - m_x)^* \cdot (x[n+m] - m_x)\}$$

$$y_{xx} = c_{xx}[m] + m_x^2$$

Die Autokovarianz bei  $m = 0$  repräsentiert die Varianz

$$E\{|x[n] - m_x|^2\}$$

Spektrum durch die DTFT im Intervall  $[n_1, n_2]$

$$X_{[n_1, n_2]}(\Omega) = \sum_{n=n_1}^{n_2} x[n] \cdot e^{j\Omega n}$$

Leistungsdichte Spektrum (power density spectrum)

$$\Gamma_{xx}(\Omega) = \sum_{m=-\infty}^{\infty} \gamma_{xx}[m] \cdot e^{-j\Omega m}$$

Weisses Rauschen hat die Autokorrelation

$$\gamma_{ww}[m] = \begin{cases} \sigma_w^2 & \text{if } m = 0 \\ 0 & \text{if } m \neq 0 \end{cases}$$

Das Leistungsdichte Spektrum beim seissen Rauschen ist über  $\Omega$  konstant.

## 6.2 Spectral Shaping in LTI Systems

Mittelwert des Ausgangssignals  $y[n]$ :

$$m_y = H(0) \cdot m_x$$

mit

$$H(\Omega) = \sum_{k=-\infty}^{\infty} h[k] \cdot e^{j\Omega k}$$

Autokorrelation von  $y[n]$

$$\gamma_{yy}[m] = h^*[-i] * \gamma_{xx}[m] * h[k]$$

Frequenzbereich

$$\Gamma_{yy}(\Omega) = H^*(\Omega) \cdot \Gamma_{xx}(\Omega) \cdot H(\Omega) = |H(\Omega)|^2 \cdot \Gamma_{xx}(\Omega)$$

## 6.3 Lineare Modelle für Stochastische Prozesse

Weisses Rauschen

$$\Gamma_{ww}(z) = \sigma_w^2$$

Ausgang des LTI

$$\Gamma_{yy}(z) = \sigma_w^2 \cdot H(z) \cdot H(z^{-1})$$

Weisses Rauschen kann erzeugt werden, indem ein bekanntes Zufallsignal mit

$$\frac{1}{H(z)}$$

multipliziert wird. Ergibt für die Differenzengleichung

$$w[n] = \frac{1}{b_0} \left( y[n] + \sum_{k=1}^N a_k y[n-k] - \sum_{k=1}^M b_k w[n-k] \right)$$

### 6.3.1 Moving average model

$H(z)$  ist ein FIR Filter mit Ordnung  $M$  welcher ein weisses Rauschen  $w[n]$  in ein Zufallsignal transformiert:

$$y[n] = \sum_{k=0}^M b_k w[n-k]$$

Im spezial Fall  $b_0 = b_1 = \dots = b_M = (M+1)^{-1}$  repräsentiert jedes Sample des Filterausgangs der Durchschnitt des Eingangssignals in einem verschiebbaren Fenster der Länge  $M+1$ .

Berechnung der Koeffizienten bei gegebener Autokorrelation  $\gamma_{yy}$ :

$$\gamma_{yy}[m] = \begin{cases} \sigma_w^2 \sum_{k=m}^M b_k \cdot b_{k-m}^* & \text{if } 0 \leq m \leq M \\ 0 & \text{if } m > M \\ \gamma_{yy}^*[-m] & \text{if } m < 0 \end{cases}$$

### 6.3.2 Autoregressive model

Zufallssequenz ist durch einen All-pole Filter generiert:

$$y[n] = w[n] - \sum_{k=1}^N a_k y[n-k]$$

Hat keine Pole und ist immer stabil.

Berechnung der Koeffizienten bei gegebener Autokorrelation  $\gamma_{yy}$ :

$$\gamma_{yy}[m] = \begin{cases} -\sum_{k=1}^N a_k \cdot \gamma_{yy}[m-k] & \text{if } m > 0 \\ \sigma_0^2 - \sum_{k=1}^N a_k \cdot \gamma_{yy}[m-k] & \text{if } m = 0 \\ \gamma_{yy}^*[-m] & \text{if } m < 0 \end{cases}$$

Das Lösungssystem in Matrizenform:

$$\begin{bmatrix} \gamma_{yy}[0] & \gamma_{yy}[-1] & \dots & \gamma_{yy}[-N] \\ \gamma_{yy}[1] & \gamma_{yy}[0] & \dots & \gamma_{yy}[-N+1] \\ \vdots & \vdots & & \vdots \\ \gamma_{yy}[N] & \gamma_{yy}[N-1] & \dots & \gamma_{yy}[0] \end{bmatrix} \cdot \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} \sigma_w^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

### 6.3.3 ARMA model

Ist die Zusammensetzung des MA und AR Models. Dabei sind die Ordnungen der Polynomen  $B(z)$  und  $A(z)$  1 oder höher. Somit hat der Pol-Nullstellenplan Nullen und Pole. Dieser ist allerdings geeigneter als die Spezialfälle MA und AR, da weniger Koeffizienten für ein gutes Modell benötigt werden.

# Kapitel 7

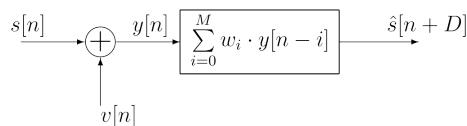
# Optimale Lineare Filter

## 7.1 Wiener Filter

Ist optimal im mittleren quadratischen Fehler.

Zurückgewonnenes Signal  $\hat{s}[n + D]$ .

- *Smoothing:* Bei  $D < 0$  ist der Sinn Rauschen vom Signal zu eliminieren und eine Verzögerung von  $-D$  in Kauf zu nehmen. Für jedes Sample erzeugt der Filter eine Schätzung auf der Basis von  $M + 1$  Sampels von  $y[n]$
- *Filtering:* Bei  $D = 0$  ist die Absicht das Signal  $s[n]$  zu regenerieren. Es werden keine Verzögerungen erzeugt, da die Abschätzung aufgrund des aktuellen und vergangenen Werten gemacht wird.
- *Prediction:* Bei  $D > 0$  will das Signal vorhergesagt werden.



Filterausgang:

$$\hat{s}[n + D] = \sum_{i=0}^M w_i \cdot y[n - i]$$

Fehler:

$$\hat{s}[n + D] - s[n + D]$$

Um die optimalen Parameter  $w_i$  zu finden muss der Gradient auf Null gesetzt werden

$$\sum_{i=0}^M \tilde{w}_i \cdot \gamma_{yy}[m - i] = \gamma_{sy}[m + D] \quad m = 0, 1, \dots, M$$

mit der Autokorrelation des Filter Eingangs:

$$\gamma_{yy} = E\{y[n] \cdot y^*[n - m]\}$$

und der Kreuzkorrelation

$$\gamma_{sy} = E\{s[n] \cdot y^*[n - m]\}$$

Winer-Hopf Gleichung

$$\mathbf{R}_{yy} = \begin{bmatrix} \gamma_{yy}[0] & \gamma_{yy}[-1] & \dots & \gamma_{yy}[-M] \\ \gamma_{yy}[1] & \gamma_{yy}[0] & \dots & \gamma_{yy}[1 - M] \\ \vdots & \vdots & & \vdots \\ \gamma_{yy}[M] & \gamma_{yy}[M - 1] & \dots & \gamma_{yy}[0] \end{bmatrix}$$

$$\mathbf{r}_{sy} = \begin{bmatrix} \gamma_{sy}[D] \\ \gamma_{sy}[D + 1] \\ \vdots \\ \gamma_{sy}[D + M] \end{bmatrix} \quad \tilde{\mathbf{w}} = \begin{bmatrix} \tilde{w}_0 \\ \tilde{w}_1 \\ \vdots \\ \tilde{w}_M \end{bmatrix}$$

Die optimalen Filterkoeffizienten sind gegeben Durch:

$$\tilde{\mathbf{w}} = \mathbf{R}_{yy}^{-1} \mathbf{r}_{sy}$$

# Kapitel 8

## Adaptive Filter

### 8.1 Linear Predictive Coding

Sprachsignal mit Ordnung  $P$  (all-pole filter):

$$H(z) = \frac{g}{1 - \sum_{k=1}^P a_k z^{-k}}$$

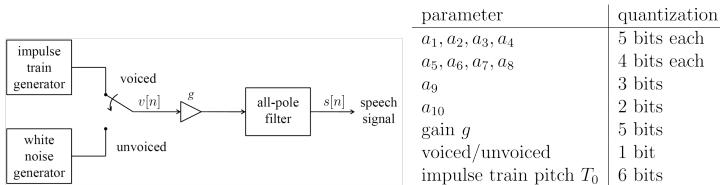
LPC-10e Algrithmus:

- 8000 Samples/s
- 180 Samples/Segment
- $P = 10$
- 54 bits pro Segment für Encoding
- Bitrate von 2400 bit/s

Entscheidet für jedes Segment, ob es voiced oder unvoiced ist. Konsonanten zählen als unvoiced, da das Signal einem Rauschen gleicht. Vokale erzeugen eine periodische Schwingung mit  $T_0$ .

Differenzengleichung

$$s[n] = \sum_{k=1}^P a_k s[n-1] + g \cdot v[n]$$



LPC-10e vocoder: parameters and quantization.

Parameter  $a_n$  müssen so gefunden werden, dass

$$\hat{s}[n] = \sum_{k=1}^P a_k s_{in}[n-k]$$

möglichst nahe an  $s_{in}[n]$  kommt. So ergibt sich:

$$\mathbf{R}_{ss} = \begin{bmatrix} \gamma_{ss}[0] & \gamma_{ss}[1] & \dots & \gamma_{ss}[P-1] \\ \gamma_{ss}[1] & \gamma_{ss}[0] & \dots & \gamma_{ss}[P-2] \\ \vdots & \vdots & & \vdots \\ \gamma_{ss}[P-1] & \gamma_{ss}[P-2] & \dots & \gamma_{ss}[0] \end{bmatrix}$$

$$\mathbf{r}_{ss} = \begin{bmatrix} \gamma_{ss}[1] \\ \gamma_{ss}[2] \\ \vdots \\ \gamma_{ss}[P] \end{bmatrix}$$

und

$$\mathbf{a} = \mathbf{R}_{ss}^{-1} \mathbf{r}_{ss}$$

Die Werte der Autokorrelation können angenähert werden durch

$$\hat{\gamma}_{ss}[m] = \frac{1}{N-m} \sum_{n=1}^{N-m} s_{in}[n] s_{in}[n+m]$$

mit  $N$  für die Anzahl Werte im Segment.

Der Fehler ist

$$d[n] = s_{in}[n] - \hat{s}[n]$$

Die Verstärkung

$$g = \sqrt{\frac{1}{N} \sum_{n=1}^N d^2[n]}$$

Durchschnittliche Amplituden Differenz Funktion (average magnitude difference function):

$$\bar{\gamma}_d[m] = \frac{1}{N-m} \sum_{n=1}^{N-m} \left| \frac{d[n]}{g} - \frac{d[n+m]}{g} \right|$$

Wenn das minimum von  $\bar{\gamma}_d[m]$  unter einen Schwellwert fällt, wird das Segment als voice deklariert

$$T_0 = \arg \min_{m \in [20, 160]} \bar{\gamma}_d[m]$$

## 8.2 LMS Algorithmus

Passt seine Parameter stetig an.

Es wird der Eingangvektor und die Filterkoeffizienten benötigt

$$\mathbf{y}_n = \begin{bmatrix} y[n] \\ y[n-1] \\ \vdots \\ y[n-M] \end{bmatrix} \quad \mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_M \end{bmatrix}$$

Der Ausgang wird berechnet mit

$$\hat{s}[n] = \mathbf{w}^T \mathbf{y}_n$$

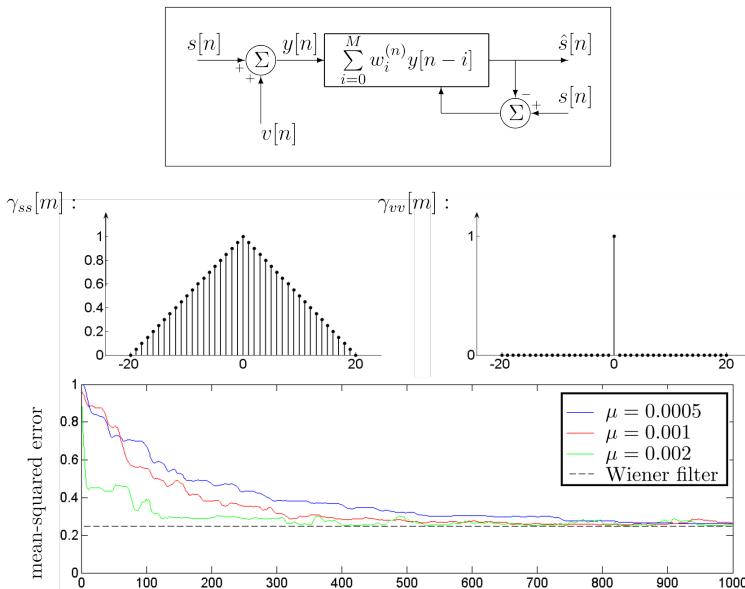
Der mittlere quadratische Fehler ist

$$\varepsilon_{MSE}(\mathbf{w}) = E\{(\hat{s}[n] - s[n])^2\}$$

Der Vektor  $\mathbf{w}$  wird mit Nullen initialisiert und berechnet sich aus

$$\mathbf{w}^{(n+1)} = \mathbf{w}^{(n)} + \underbrace{\mu \left( s[n] - \left( \mathbf{w}^{(n)} \right)^T \mathbf{y}_n \right) \cdot \mathbf{y}_n}_{=\mathbf{w}_\Delta^{(n)}}$$

$\mu$  ist die Schrittweite und gibt an, wie oft der Algorithmus angepasst wird.



# Kapitel 9

# Matlab

## Matrizengenerierung

Befehl	Beschreibung	Syntax
<code>zeros</code>	Matrix mit Nullen	<code>zeros(N),</code> <code>zeros(M,N)</code>
<code>ones</code>	Matrix mit Einheiten	<code>ones(N),</code> <code>ones(M,N)</code>
<code>rand</code>	Matrix mit gleichverteilten Zufallszahlen	<code>rand(N),</code> <code>rand(M,N)</code>
<code>randn</code>	Matrix mit normalverteilten Zufallszahlen	<code>randn(N),</code> <code>randn(M,N)</code>
<code>linspace</code>	Matrix mit linear steigenden Werten	<code>linspace(N),</code> <code>linspace(M,N)</code>
<code>logspace</code>	Matrix mit logarithmisch steigenden Werten	<code>logspace(N),</code> <code>logspace(M,N)</code>

### Elementar Funktionen

Befehl	Beschreibung	Syntax
<code>log</code>	natürlicher Logarithmus	
<code>log10/2</code>	Logarithmus zur Basis 10 bzw. 2	
<code>round</code>	Rundung zur nächsten ganzen Zahl	
<code>ceil</code>	Rundung Richtung $\infty$	<code>ceil(n.x) = n + 1</code>
<code>floor</code>	Rundung Richtung $-\infty$	<code>floor(n.x) = n</code>
<code>mean</code>	Arithmetische Mittel	

### Komplexe Zahlen

Befehl	Beschreibung	Syntax
<code>abs</code>	Betrag	
<code>angle</code>	Phase	
<code>real</code>	Realanteil	
<code>imag</code>	Imaginäranteil	
<code>conj</code>	conjungiert complex	
<code>polar</code>	Polarkoordinaten-Darstellung	

### Graphik

Befehl	Beschreibung	Syntax
<code>figure</code>	Grafik-Fenster	
<code>plot</code>	2-D Grafik	<code>plot(X1,Y1,S1,X2,Y2,S2,...)</code>
<code>subplot</code>	Unterteilung des Grafik-Fensters	<code>subplot(z,s,p)</code>
<code>stem</code>	Darstellung diskreter Werte	
<code>hold</code>	Halten der Grafik zum Übereinanderzeichnen	

---

## Transformation und Filterung

Befehl	Beschreibung	Syntax
<b>dftmtx</b>	Transformationsmatrix für DFT	<code>dftmtx(N)</code>
<b>fft/fft2</b>	schnelle DFT (FFT), 1-/2-dimensional	
<b>ifft/ifft2</b>	inverse FFT, 1-/2-dimensional	
<b>fftshift</b>	Verschiebung des Gleicheils in die Mitte des Spektrums	
<b>spectrum</b>	Leistungsdichte-Spektrum	
<b>filter</b>	Filterung	<code>filter(B,A,X)</code> $(H(z) = B(z)/A(z))$
<b>conv</b>	Faltung	
<b>fftfilt</b>	schnelle Faltung	
<b>freqz</b>	Frequenzgang	<code>freqz(B,A,N)</code>
<b>impz</b>	Impulsantwort	<code>impz(B,A,N)</code>
<b>zplane</b>	Pol/Nullstellenverteilung	<code>zplane(B,A)</code>

## Filterentwurf

Befehl	Beschreibung	Syntax
<b>fir1</b>	Hoch-, Tief-, Bandpass, Bandsperre	<code>fir1(N,Wn,'low')</code> <code>fir1(N,Wn,'high')</code> <code>Wn=[W1,W2]</code> <code>fir1(N,Wn,'bandpass')</code> <code>fir1(N,Wn,'stop')</code>
<b>fir2</b>	allgemeiner Frequenzverlauf	
<b>butter</b>	Butterworth IIR	<code>[B,A]=butter(N,Wn,'low')</code>
<b>cheby1</b>	Chebyshev I IIR	
<b>cheby2</b>	Chebyshev II IIR	
<b>ellip</b>	Elliptisch Cauer	

## Fensterung

Befehl	Beschreibung	Syntax
<b>rectwin</b>	Rechteckfenster	<code>rectwin(N)</code>
<b>bartlett</b>	Dreieck-Fenster	<code>bartlett(N)</code>
<b>hamming</b>	Hamming-Fenster	<code>hamming(N)</code>
<b>kaiser</b>	Kaiser-Fenster	<code>kaiser(N)</code>