Digital Signal Processing

Zusammenfassung

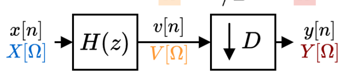
Joel von Rotz / Quelldateien

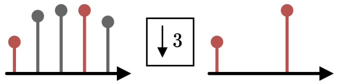
Table of Contents

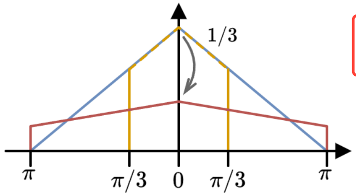
Sampling Rate Conversion

Decimation · · · · · · · · · · · · · Reducing sampling rate by an **Integer Factor** D

Ω — $/D$ → Ω







Sampling Theorem $f_{\max} < \frac{f_s}{2}$
Don't forget!

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

For $\Omega\in[-\pi,\pi]$

Decimated Frequency: $\boxed{F_Y=F_X/D \;\leftrightarrow\; \Omega_Y=\Omega_{X,V}/D}$

Ideally filtered

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

General Formula

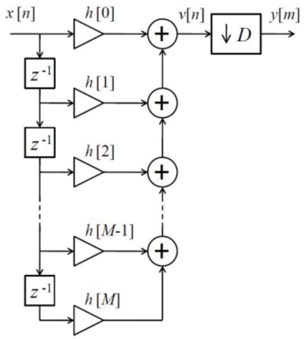
$$Y(z)=\frac{1}{D}\sum_{d=0}^{D-1}V\left(\frac{\Omega}{D}-2\pi\cdot\frac{d}{D}\right)$$

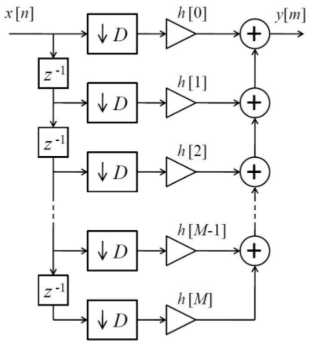
Direct Implementation

FIR Filter of order M produces full signal $v[n]$ + downsampler discards $D-1$ samples afterwards → **inefficient!**

Efficient Implementation

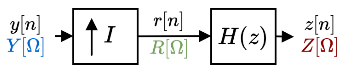
Downsampling beforehand allows the multiplier to operate at the reduced sampling rate → **much better!**

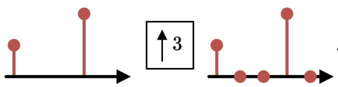


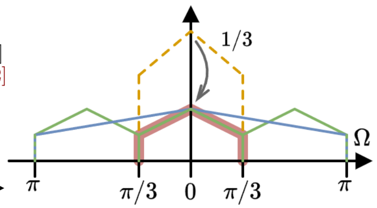


Interpolation · · · · · · · · · · · · · Increase sampling rate by an **Integer Factor** I

Ω — $\times I$ → Ω







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

Interpolated Frequency: $\boxed{F_{Z,R}=I\cdot F_Y \;\leftrightarrow\; \Omega_{Z,R}=I\cdot\Omega_Y}$

Interpolation Formula

$$R(\Omega)=\sum_{n=-\infty}^{\infty}r[n]e^{-j\Omega\cdot n}=\sum_{n=-\infty}^{\infty}y[m]e^{-j\Omega\cdot I\cdot m}=Y(I\Omega)$$

Low Pass Filter

For $\Omega\in[-\pi,\pi]$

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

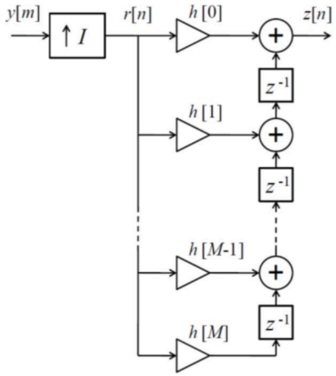
Lowpass-filter uses Ω and **NOT** Ω !

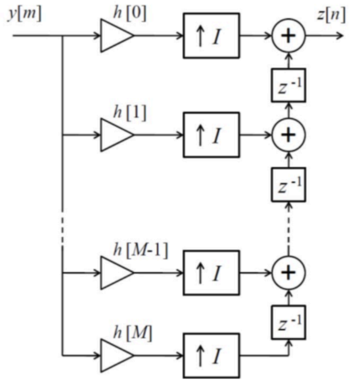
Direct Implementation

FIR or IIR Filter ; $I-1$ out of I $r[n]$ samples are zero → **inefficient!**

Efficient Implementation

Upsampling after filtering → multiplier operates at **reduced** sampling rate (F_Y) → **much better!**





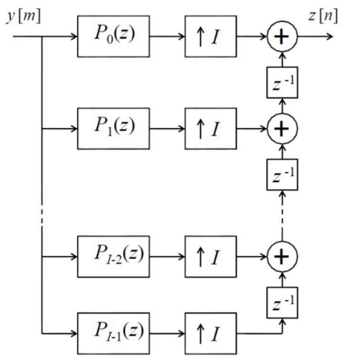
Polyphase Filter Structure · · · · · · · · · · · · · Efficient filter implementation

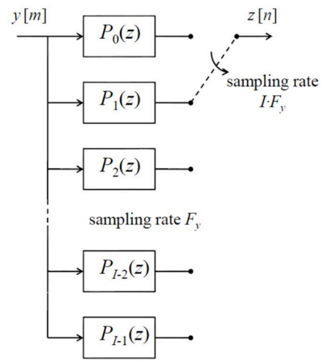
TODO Relearn

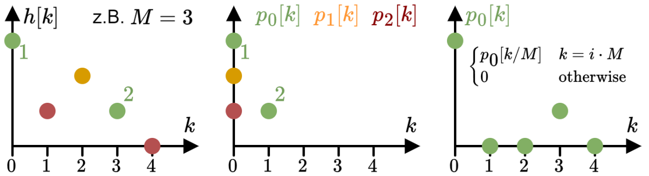
Split filter into M **downsampled** variants of the impulse resonse $h[k]$. Every variant $p_{i[k]}$ holds only every M-th coefficient („sum" of variants = $h[k]$)

$$p_i[k]=h[kM+i],\quad i=0,1,...,M-1 \quad\mid\quad H(z)=\sum_{i=0}^{M-1}z^{-1}P_i(z^M)$$

$$P_{i(z)}=\sum_{k=-\infty}^{\infty}p_i[k]z^{-k}=\sum_{k=-\infty}^{\infty}h[kM+i]z^{-k},\quad i=0,1,...,M-1$$

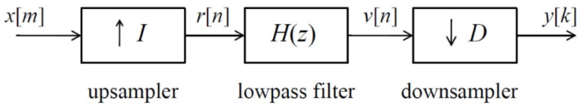






TODO include fourier transform equation by hand!

Sampling Rate Conversion



$$F_X \Rightarrow F_R = I \cdot F_X \Rightarrow F_V = F_R \Rightarrow F_Y = I/D \cdot F_X = 1/D \cdot F_V$$

Decimation $\Rightarrow$ loss of information

$$F_H(\Omega) = \min\left(\frac{\pi}{I}, \frac{\pi}{D}\right)$$

Interpolation $\Rightarrow$ higher intermediate sampling rate

$$\frac{I}{D} = \underbrace{I \cdot \left(\frac{1}{D}\right)}_{H_I \to I \to D \to H_D} = \underbrace{\left(\frac{1}{D}\right) \cdot I}_{I \to H \to D}$$

TODO FIR-order = FIR-length

Filter Banks

TODO

Quadrature Mirror Filters

TODO

DFT Filter Banks

TODO

Random Signals

TODO

Autocorrelation and Spectrum

TODO

Spectral Shaping

TODO

Linear Models for Stochastic Processes

TODO

Spectral Density Estimation

TODO

Wiener Filters

TODO

Unconstrained Wiener Filters

TODO

The Principle of Orthogonality

TODO

Kalman Filter

TODO

Linear Predictive Coding

TODO

LMS Algorithm

TODO

The LMS Algorithm

TODO

Acoustic Echo Cancellation

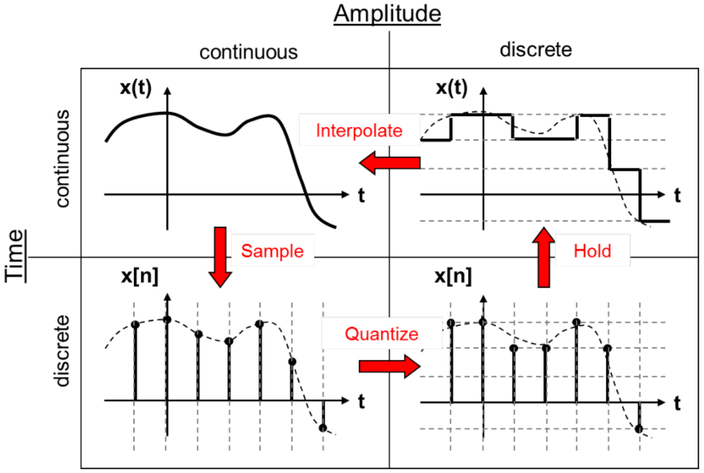
TODO

Hello World

Hello World

Digital Signal Processing (DSP)

DSP concerned in the application of the following methods: ① **Signal Generation**, ② **Signal Analysis**, ③ **Signal Composition**, ④ **Signal Selection**



Pros (3 P's): **P**rogrammability, **P**arametrizability, Re-**P**eatability

Cons: additional effort for ADC & DAC, No processing of broadband HF, electromagnetic disturbance

Signal Analysis

Sampling an Analog Signal

$$f_S = \frac{1}{T_S} \qquad x(n \cdot T_S) = x[n]$$

Other Functions

causal: $x[n] = 0$ for $n < 0$

T_S: Always known!

unit impulse

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

step impulse

$$u[n] = \begin{cases} 0 : n < 0 \\ 1 : n \geq 0 \end{cases}$$

periodic symbols ($k = T_0/T_S$)

$$x[n] = x[n + T_0/T_S]$$

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

$$= \hat{X}\left(C(\_) + j \cdot S(\_)\right)$$

expected/mean value

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

$(\text{mean value})^2$ / avg DC power

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

variance/avg AC power

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

Acoustic signals: corresponds to audible power content

power ratio

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

Signal-to-Noise-ratio

$$\text{SNR} = 10 \cdot \log_{10}\left(\frac{P_S}{P_N}\right)$$

$$= 20 \cdot \log_{10}\left(\frac{U_S}{U_N}\right)$$

power ratio

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

static correlation ($x = y \Rightarrow \uparrow R$)

$\Rightarrow$ yields new signal, quantifying the similarty of x and y

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

linear correlation

$$r_{xy}[n] = \frac{1}{N}\sum_{i=0}^{N-1} x[i]y[i+n]$$

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

$$r_{xy}[n] \neq r_{yx}[n]$$

(linear) convolution

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

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

circular convolution

$$N_X = N_Y$$

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

Example:

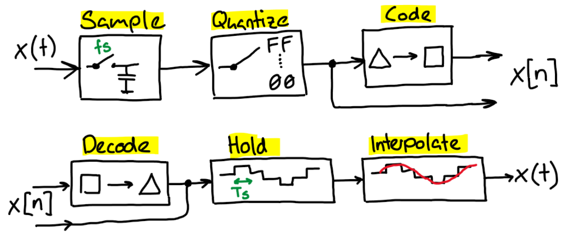
$$x = \underbrace{\{0.5, 1\}}_{=a+0.5} \quad \& \quad y = \underbrace{\{1, -1\}}_{=-b+1}$$

$$\begin{pmatrix} 1 & 0 & -1 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0.5 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0.5 \\ -1 \end{pmatrix}$$

$$z[n] = \{0.5, 0.5, -1\}$$

$$x \cdot y \mid_{b=a} = -a^2 + 0.5a + 0.5$$

A/D & D/A Conversion



Code/Decode z.B. DFT & IDFT ; *Interpolate* z.B. Tiefpass-Filter

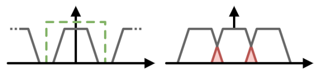
Sampling & Aliasing

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

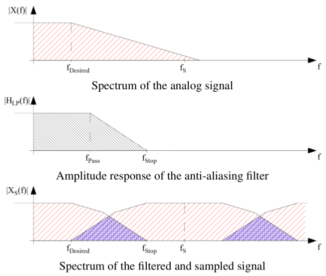
Sampling Theorem! $\left[f_{\max} < \frac{f_S}{2}\right]$

⚠ Aliasing

Bei $f_{\max} > f_S/2$ ensteht Aliasing.

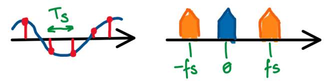


Wenn Theorem nicht möglich ist.



$$f_{\text{Pass}} \geq f_{\text{Desired}} \qquad f_{\text{Stop}} \geq f_S - f_{\text{Desired}}$$

⚠ Sampling



period. Spektrum mit f_s-vielfachen *Spiegelbilder*. Mit spektraler Verschiebung

$$x(t)e^{j2\pi f_0 t} \bullet\!\!-\!\!\circ\ X(f - f_0)$$

ergibt

$$X_s(f) = \frac{1}{T_S}\sum_{k=-\infty}^{\infty} X(f - k \cdot f_S)$$

Sampling of Band-Pass Signals

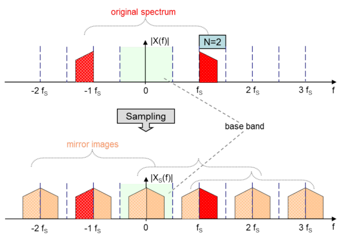
generalized sampling theorem:

$$-\frac{N+1}{2}f_S \leq f \leq -\frac{N}{2}f_S \quad \text{and}$$

$$\frac{N}{2}f_S \leq f \leq \frac{N+1}{2}f_S$$

Wird generalisierten Theorem eingehalten, kann Signal rekonstruiert werden. Zum prüfen, ob eine Sampling Frequenz für ein Band-Pass Signal gültig ist:

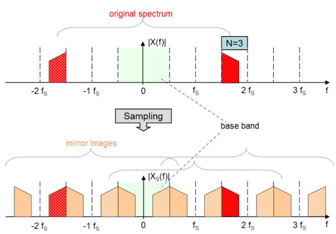
$$2 \cdot \frac{f_{\min}}{N} \geq f_S \geq 2 \cdot \frac{f_{\max}}{N+1}$$



Odd N: Verschiebung mit Kosinus $f_S/2$

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

$$(-1)^n = \cos(\pi \cdot n) = \cos(2\pi f_S/2 \cdot n \cdot T_S)$$



Digital Signals in Frequency Domain

Fourier Transformation to DFT

Discrete-Time Fourier Transform (DTFT)

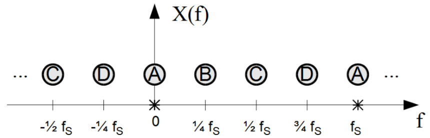
Transition to Discrete Time

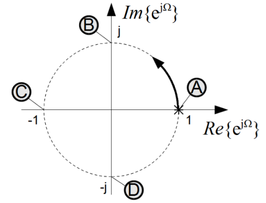
$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi f \cdot t}dt \longrightarrow X_S(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi f \cdot n \cdot T_S}dt$$

$$\Omega = 2\pi f T_s = 2\pi \frac{f}{f_S} \Longrightarrow \boxed{X(\Omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\Omega n}}$$

$\Longrightarrow$ $X(\Omega)$: **Discrete-Time Fourier Transform (DTFT)**

Ω: normalized angular frequency





Transition to Finite Measurement Level

Fourier has ∞ long measurement time $\rightarrow$ Confine to N sample points, which leads to a discrete frequency range.

Discrete frequency range:

$$0, \frac{f_S}{N}, 2\frac{f_S}{N}, ..., (N-1)\frac{f_S}{N}$$

Measurement Interval: $T = N \cdot T_S$

ⓘ Lowest capturable frequency

(With exception of any DC component)

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

Discrete Fourier Transform (DFT)

$$\boxed{X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j2\pi n\frac{k}{N}}}$$

with $k = 0, 1, 2, ..., N-1$

Inv. Discrete Fourier Transform (IDFT)

synthesis equation: $x[n]$ is periodic at $T_S \cdot N$

$$\boxed{x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k] \cdot e^{j2\pi n\frac{k}{N}}}$$

with $n = 0, 1, 2, ..., N-1$

Either DFT or IDFT require the normalization factor $1/N$ to re-obtain the original signal. $\Rightarrow$ IDFT has the normalization factor above.

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

- periodicity in time $\rightarrow$ discrete line spectra in frequency (Fourier & DFT)
- sampling in time $\rightarrow$ periodic in frequency (DFT, DTFT)

DFT Intuitive

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j2\pi n \frac{k}{N}}$$
$$= \sum_{n=0}^{N-1} x[n] \cos\left(-2\pi n \frac{k}{N}\right) + j \cdot \sum_{n=0}^{N-1} x[n] \sin\left(-2\pi n \frac{k}{N}\right)$$
$$= \underbrace{\sum_{n=0}^{N-1} x[n] \cos\left(2\pi n \frac{k}{N}\right)}_{\mathrm{Re}(X[k])} + j \cdot \underbrace{\sum_{n=0}^{N-1} x[n](-1) \sin\left(2\pi n \frac{k}{N}\right)}_{\mathrm{Re}(X[k])}$$

Static Correlation

Every DFT coefficient $X[k]$ is equal to the *static* correlation between the signal $x[n]$ and discrete sine and cosine functions of frequency kf_S/N.

Meaning: the DFT indicates how similar the signal is to harmonic oscillations with frequency k

Properties of the DFT

Important Properties

Periodicity DFT works with discrete time signal samples, the spectrum is f_S periodic.

$$\mathrm{DFT}: X[k] = X[k+N] \qquad \mathrm{IDFT}: x[n] = x[n+N] \text{ with } T = NT_S$$

Symmetry DFT of a real-valued signal is symmetric around the point $k = N/2$

$$X\left[\frac{N}{2} + m\right] = X^*\left[\frac{N}{2} - m\right]$$

Time/Frequency Shifting Shifting a periodic time sequence corresponds to a linear phase offset to all spectral values

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

The inverse is also true → mult. complex exp. in time leads to frequency shift

$$e^{j2\pi \cdot n_0 \frac{k}{N}} \cdot x[n] \quad \circ\!\!-\!\!\bullet \quad X[k - k_0]$$

Modulation Direct consequence of frequency shift → modulation property

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

Parseval Theorem left side equals to energy of signal → right side has use for SNR (separate noise frequency from signal frequency)

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

Correspondence of Conv. and Multi. fast conv. → $\mathrm{IDFT}(\mathrm{DFT}(x[n] \cdot \mathrm{DFT}(y[n])))$

$$x[n] \circledast_N y[n] \quad \circ\!\!-\!\!\bullet \quad X[k] \cdot Y[k] \quad (k = 0, 1, \ldots N-1)$$

Range of Validity of the DFT

aperiodic $x[n]$ all signal values $x[n]$ are zero outside the range $0 \le n \le N$. DFT samples the DTFT at discrete points of normalized angular frequency:

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

IF NOT (range outside $\neq 0$) → DFT = approximation of DTFT → solution: *windowing*

periodic $x[n]$ measurement interval $N \cdot T_S$ is an integer multiple of the period duration of $x[n]$

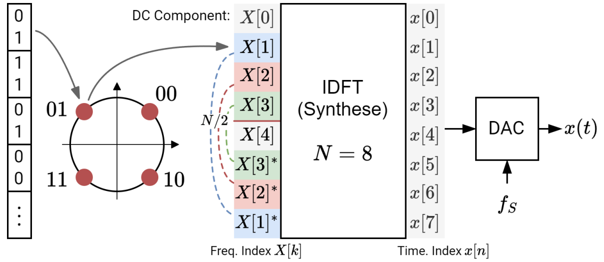
OFDM Principle

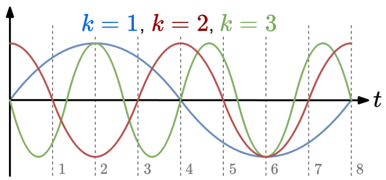
Orthogonal Frequency Division Multiplexing

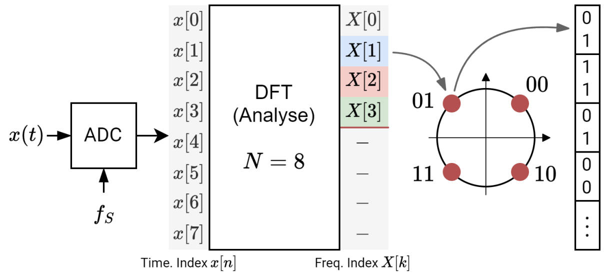
Bits are spread across different frequencies.

(1) bits are converted to phase (QPSK)

(2) the result → IDFT (3) $x[n] \to x(t)$ via DAC



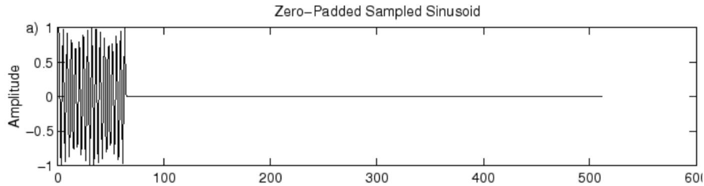


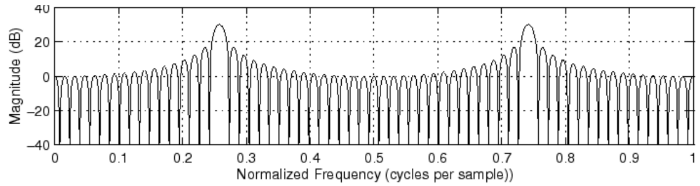


Practical Application Aspects of the DFT

DFT and Zero-padding

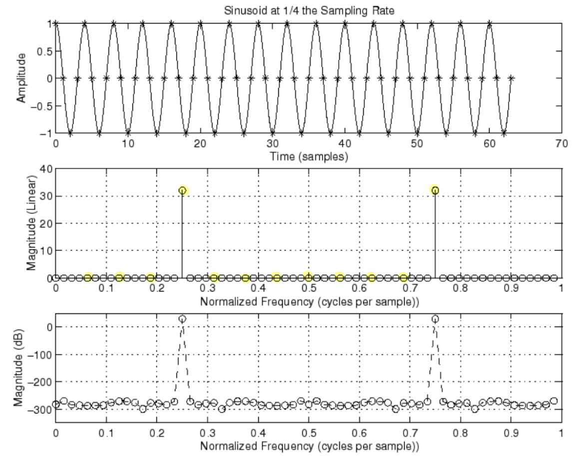
- Extending signal(t) with zeros → better interpolation (thinner frequency *bins*)
- does not modify DTFT $X[\Omega]$, but provides additional sample points along Ω
- Rectangular window of length N → convolution of $X[k]$ with $\sin(x)/x$ (lobes)
- Important lobe-structure characteristics
  - **Width of the main lobe** (example: ≈ 0.03 cycles per sample)
  - **Attenuation of the first side lobe** relative to main lobe ($\approx 12\text{dB}$)



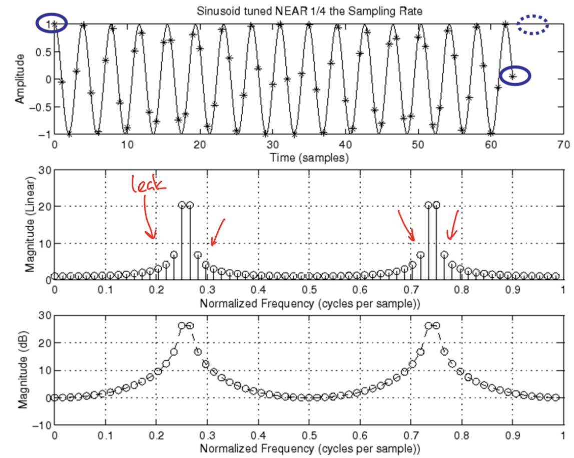


Choice of Measurement Interval & Leakage Effect

Example: $N = 64, f_0 = f_S/4, T = N \cdot T_S = 16 \cdot T_0$



Example: $f_0 = f_S/4 + f_S/128$ → measurement interval no integer multiple of the period duration: **Leakage effect**:



DFT and Windowing

- DFT applies rectangular window N samples
- Applying the *Blackman Window* and afterwards appending zeros
  - Reduces virtual periodic continuation of the signal „outside" of signal, thus reducing the leakage effect.

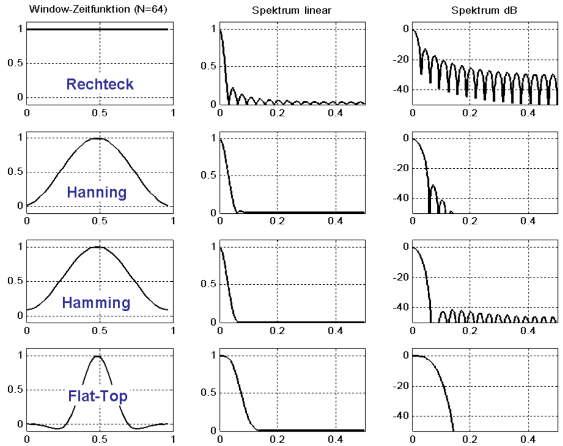
Choice of Windowing Function

Choice Compromise

Choice of Window function leads to a compromise between the attenuation of leakage and spectral resolution in the spectrum $X[k]$

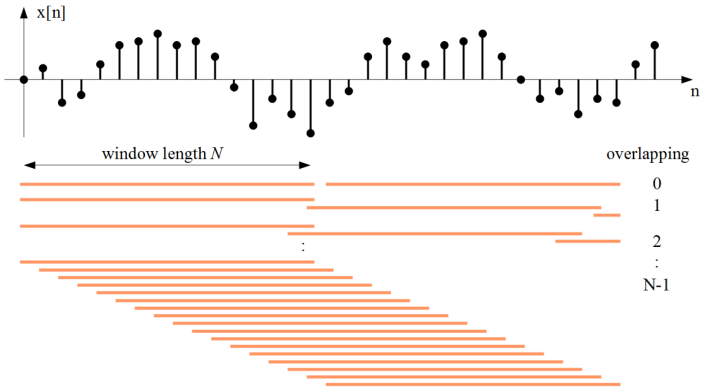
- **Narrow main lobe**: higher the spectral resolution for $X[k]$
- **Higher the side lobe attenuation**: better suppression of leakage in $X[k]$

$\Longrightarrow$ **Ideal**: DC-function $\rightarrow$ indefinitely small main lobe, no side lobes ($N \rightarrow \infty$)



Short-Time DFT

- continuous evaluation of the frequency spectrum of short signal sections
  - Allows the observation of frequency spectrum over time
  - **BUT** more computation required $\rightarrow$ solution: FFT



Fast Fourier Transformation (FFT)

Fast DFT

Complexity of the FFT

Divide-and-Conquer principle

- **Divided** get either N sample values (**decimation-in-time**) or N spectral values (**decimation-in-frequency**)
  - Split values recursively into r sub-sequences (r: radix) $\rightarrow$ radix-2 algo often used

$$N = 2^L \text{ where } L \text{ is some integer}$$

- N almost always a power of two

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

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

assuming $T_{\text{compute,Add}} = T_{\text{compute,Mul}}$: $\quad$ speedup factor$_{\text{FFT}} = \dfrac{8N-2}{5 \cdot \log_2(N)} \approx 1.5 \dfrac{N}{\log_2(N)}$

Properties of the Twiddle Factors

In order to reduce the computational effort we introduce the **twiddle factor** $W_N = e^{-j2\frac{\pi}{N}}$ and can write the DFT new:

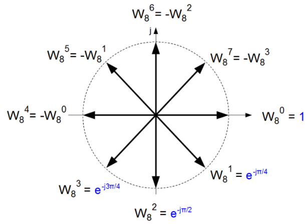
$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot W_N^{n \cdot k}, \quad k = \{0, 1, 2, ..., N-1\}$$

Periodicity W_N^k can evaluate to N different numbers only

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

Symmetry Apart from sign, every W_N^k takes on only $N/2$ different values within each period.

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



MCU only requires $\frac{N}{2} \cdot 2$ (Re & Im) space.

Radix-2 decimation-in-time FFT

Splitting the twiddle-factor DFT up into odd and even yields two new sequences of length $N/2$:

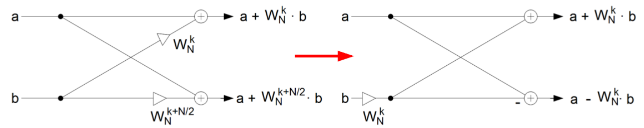
$$X[k] = \underbrace{\sum_{n=0}^{N/2-1} x_1[n] W_N^{2nk}}_{x_1 \rightarrow n \text{ even}} + \underbrace{\sum_{n=0}^{N/2-1} x_2[n] W_N^{(2n+1)k}}_{x_2 \rightarrow n \text{ odd}}$$

$$\text{introducing } W_N^2 = W_{N/2}: \quad = \underbrace{\sum_{n=0}^{N/2-1} x_1[n] W_{N/2}^{nk}}_{X_1[\tilde{k}]} + W_N^k \cdot \underbrace{\sum_{n=0}^{N/2-1} x_2[n] W_{N/2}^{nk}}_{X_2[\tilde{k}]}$$

$\Rightarrow X_1[\tilde{k}], X_2[\tilde{k}]$: $N/2$-point DFT $\longrightarrow \tilde{k} = k \bmod N/2$ (limit k-range to meaningful $N/2$)

Recursively applying the splitting procedure leads to $\frac{N}{2}$ 2-point DFTs:

$$X[k] = \sum_{n=0}^{0} x_1[n] W_2^{nk} + W_2^k \cdot \sum_{n=0}^{0} x_2[n] W_2^{nk} = \underline{x_1[0] + W_2^k \cdot x_2[0]}, \qquad k = \{0, 1\}$$

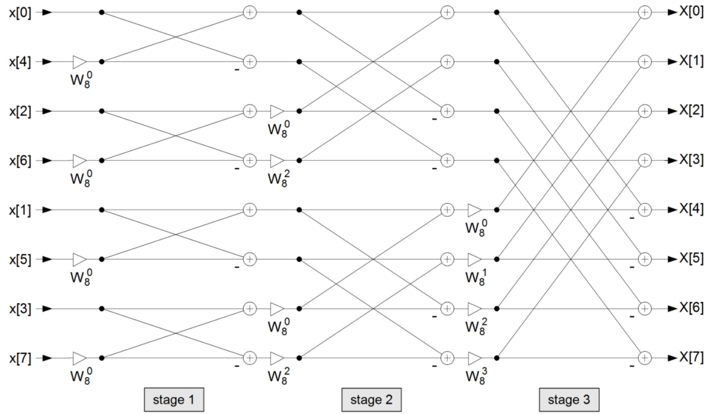


- Butterfly structure requires $\log_2(N)$ processing stages ($N = 8 \rightarrow 3$ stages)

Efficient FFT Implementation

1. As soon as the butterfly operation has been performed, input pair can be re-used to store the calculated output-pair, thus performing the entire FFT **in-place**.
2. Order of input values is **bit-reversed**: 0 (`000`), 4 (`001` $\rightarrow$ `100`), 2 (`010`), 6 (`110`), 1, 5, 3, 7.

Matlab command `bitrevorder` for bit-reversed order



The Goertzel Algorithm

Goertzel is used, if only an individual $X[k]$ of all N spectral components is required:

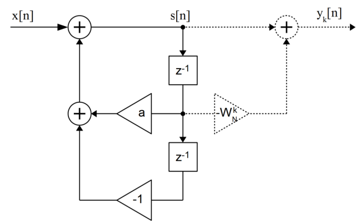
$$s[n] = x[n] + a \cdot s[n-1] - s[n-2]$$

$$y_{k[n]} = s[n] - W_N^k \cdot s[n-1]$$

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

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

$$a = 2 \cdot \cos\left(2\pi\frac{k}{N}\right), \quad W_N^k = e^{-j \cdot 2\pi k/N}$$



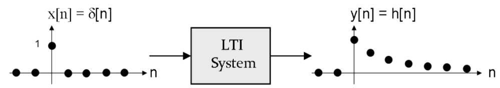
Digital LTI Systems

- **Definition** of LTI Systems
  - **Linearity**: $\quad y[n] = k_1 \cdot S\{x_1\} + k_2 \cdot S\{x_2\} = S\{k_1 \cdot x_1 + k_2 \cdot x_2\}$
  - **Time-Invariance**: $\quad x[n] \rightarrow y[n] \quad \Longrightarrow \quad x[n-d] \rightarrow y[n-d]$
- Allowed **Operations**
  - Multiplication of a signal with a constant: $\quad x[n] \cdot a$
  - Addition of two signals: $\quad x[n] + y[n]$
  - Time delay of a signal by $k \cdot T_s$: $\quad x[n - k \cdot T_S]$

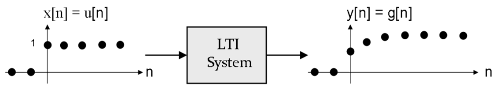
System Descriptions in the Time Domain

Impulse Response

System Identification, Measurement



$$y[n] = \sum_{i=-\infty}^{\infty} x[i] \cdot h[n-i] = x[n] * h[n] \rightarrow \text{Linear Convolution}$$



$$h[n] = g[n] - g[n-1] \iff g[n] = \sum_{k=0}^{n} h[k]$$

Difference Equation System Implementation (algorithm)

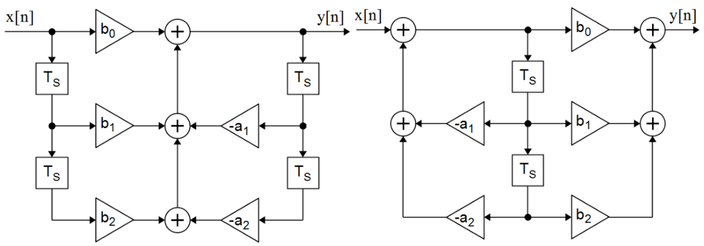
$$y[n] = \sum_{k=0}^{N} b_k \cdot x[n-k] - \sum_{k=1}^{M} a_k \cdot y[n-k]$$

System Order
System order is defined by $\max(N, M)$

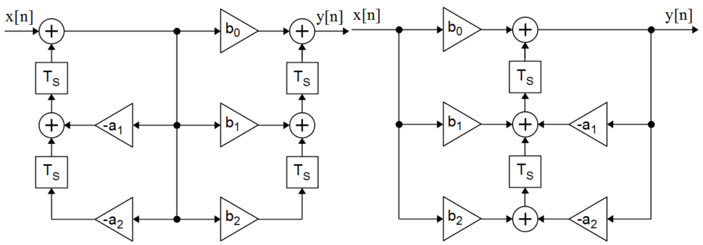
Recursive
A system is recursive, when $M \geq 1$.

Signal-Flow Diagram System Implementation (architecture)

Direct Form 1 Direct Form 2



Tranposed Direct Form 1 Transposed Direct Form 2



System Descriptions in the Frequency Domain

Transfer Function Coupling Analysis and Implementation

$$y[n] = \sum_{k=0}^{N} b_k \cdot x[n-k] - \sum_{k=1}^{M} a_k \cdot y[n-k]$$

$$y[n] = \sum_{k=0}^{N} b_k \; z^{-k} \cdot X(z) - \sum_{k=1}^{M} a_k \; z^{-k} \cdot Y(z)$$

z-Transfer-Function

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\prod_{n=0}^{N} b_n z^{-n}}{\prod_{m=0}^{M} a_m z^{-m}}$$

Pol/Zero-Plot Intuitive Analysis and Design

$$H(z) = K_0 \cdot \frac{(z-z_1)(z-z_2)...(z-z_N)}{(z-p_1)(z-p_2)...(z-p_M)} \cdot z^{M-N}$$

z_i zeros ; p_i poles
K_0 gain

$\boxed{(M > N) \wedge b_0 \neq 0} \to M - N$ additional zeros at $z = 0$ & only this case holds $K_0 = b_0$

$\boxed{N > M} \to N - M$ additional poles at $z = 0$

Causal LTI System is stable if $|p_i| < 1, i = 1, ..., M$ (all poles within the unit circle of z-plane)

Frequency Response System Identification, Analysis and Design

$$h[n] \circ\!\!-\!\!\bullet H[\Omega] \quad ; \quad H(\Omega) = |H(\Omega)| \cdot e^{j \cdot \varphi(H(\Omega))} \quad ; \quad |H(\Omega)|_{\text{dB}} = 20 \cdot \log_{10}(|H(\Omega)|)$$

$|H(\Omega)|$: amplitude response ; $\varphi(H(\Omega))$ phase response ;

- Frequency components in input are delayed differently, the output suffers from distortions $\to$ Therefore, linear phase response $\varphi(H(\Omega)) = -K \cdot \Omega$ is desirable, since only then all frequency components are delayed: **group delay**

$$\tau_g = -\frac{d\varphi(H(\Omega))}{d\Omega}$$

Any LTI system reacts to a sinusoidal input signal with a sinusoidal output signal of the same frequency:

$$x[n] = \cos(2\pi f_0 \cdot n \cdot T_S) \Longrightarrow y[n] = |H(\Omega_0)| \cdot \cos(2\pi f_0 \cdot n \cdot T_S + \varphi(H(\Omega_0)))$$

The phase and amplitude can be extracted through:

$$|Y(\Omega)| = |X(\Omega)| \cdot |H(\Omega)| \quad ; \quad \varphi(Y(\Omega)) = \varphi(X(\Omega)) + \varphi(H(\Omega))$$

Relation between frequency response and transfer function

- DTFT and z-transform relation $\Rightarrow z = r \cdot e^{j\Omega}$
- Frequency response = DTFT of impulse response $\Rightarrow H(\Omega) = H(z)|_{z=e^{j\Omega}}$
- To obtain frequency response by evaluating:

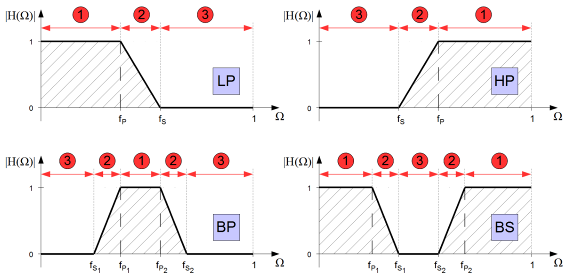
Amplitude response

$$|H(z)| = |K| \frac{\prod_{n=1}^{N} |z - z_n|}{\prod_{m=1}^{M} |z - p_M|} |z|^{M-N}$$

Phase response

$$\varphi(H(\Omega)) = \sum_{k=1}^{N} \varphi(z - z_k) - \sum_{k=1}^{M} \varphi(z - p_k) + \sum_{k=N+1}^{M} \varphi(z)$$

Design of Digital Filters



$a_k, b_k \longrightarrow$ filter coefficients $\qquad N, M \longrightarrow$ filter order

FIR Filter

Definition and Properties

FIR filter of order N has the transfer function & impulse response:

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

$$h[n] = \{b_0, b_1, ..., b_N, 0, 0, ...\} \qquad \text{size } N+1$$

with $N + 1$ coefficients.

- **Stability** per definition, all poles are at $z = 0$
- **Linear Phase**: easier to realize a linear-phase transfer characteristics (group delay)
- **Implementation**: easy implementation on HW and SW

Disadvantages: higher order requires more computational effort.
Other names: all-zero filter, transversal filter, moving-average filter

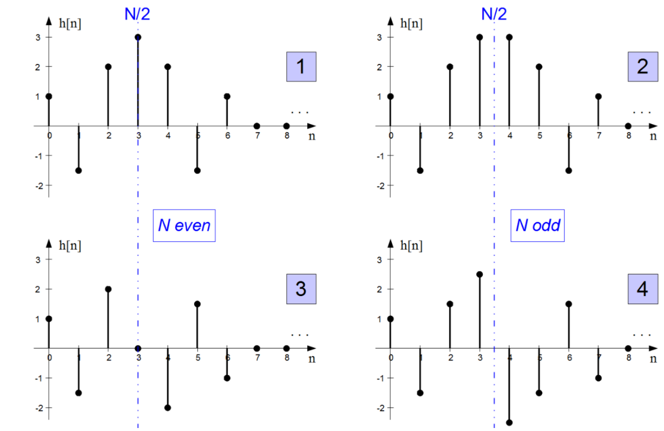
Symmetric FIR Filters

A FIR filter is symmetric when $\boxed{b_i = \pm b_{N-1}, \quad i = 0, 0, ..., N}$

- in case of $+$: (mirror-)symmetric
- in other cases: anti-symmetric

Lineare Phase Response for all symmetric FIR filters

1 All symmetric FIR filters feature a linear phase response within their **pass** band (group delay): $\tau_g = \frac{N}{2} \cdot T_S$



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

[1]: transfer function of symm. FIR are the product of a linear-phase term and some real-valued transfer function $H_{\text{zp}(\Omega)}$ (**zp**: zero-phase filter)
$\Longrightarrow$ anti-symmetric: constant 90° phase offset

Stop Band 180° Jump

In the stop band of a symm. FIR filter there can be 180°-phase-jumps. Such discontinuities in phase response occur at a pair of complex-conj. zeros at the unit circle. This are often tolerated in favor of sufficient attenuation in the stop band

Type	low-pass (LP)	high-pass (HP)	band-pass(BP)	band-stop (BS)
1	yes	yes	yes	yes
2	yes	–	yes	–
3	–	–	yes	–
4	–	yes	yes	–

Window Design Method

Matlab: `fir1`

The *Window Design Method* always yields low pass filters → other filters are done via sum and difference of low-pass filters at different cut-off frequencies.
Start of with a desired frequency response $H_{d(\Omega)}$ of an ideal TP-filter with cutoff at f_C:

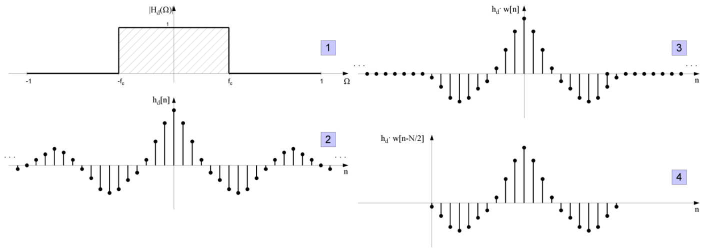
$$h_{d\mathrm{TP}}[n] = \frac{\sin(\Omega_C \cdot n)}{\pi \cdot n} \bullet\!\!-\!\!\circ \text{ rectangular signal in freq. domain}$$

- **Restrictions**: finite length ideal impulse response (corresponds to multiplication with □-window **/** convolution with sinc-function) → overshoot at edges of pass and stop bands
  - persists for $N \to \infty$ (only helps to reduce the width of the transition band)
  - Solution: use different windowing functions to smooth the overshoot at a cost of wider transition bands

Example High-Pass Filter

TP with cutoff at $f_S/2$ minus TP with cutoff at f_C:

$$\begin{aligned} h_{d\mathrm{HP}}[n] &= \frac{\sin(\pi \cdot n)}{\pi \cdot n} - \frac{\sin(\Omega_C \cdot n)}{\pi \cdot n} \\ &= \{..., -h_{d\mathrm{TP}}[-2], -h_{d\mathrm{TP}}[-1], 1 - h_{d\mathrm{TP}}[0], -h_{d\mathrm{TP}}[1], -h_{d\mathrm{TP}}[2], ...\} \end{aligned}$$



Steps of the window design method for FIR filters: Ideal low-pass frequency response (1), ideal low-pass impulse response (2), windowed (3) and shifted (4) practical low-pass impulse response.