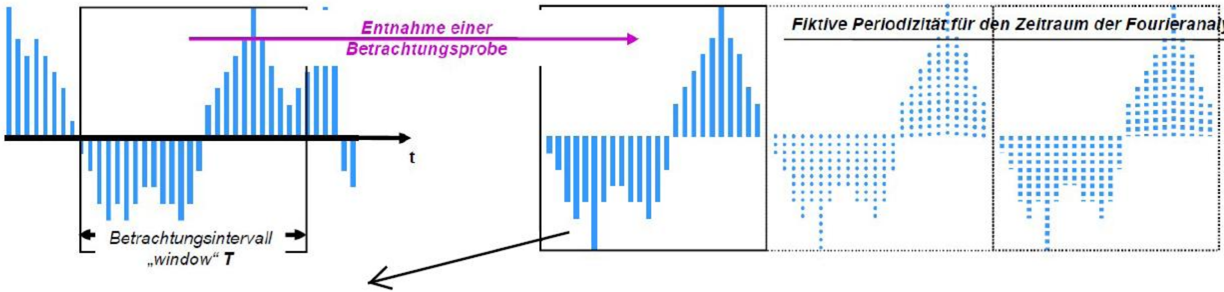
TECHNISCHE FREQUENZANALYSE von Signalen:

Jeder **periodische Signalvorgang** ***f(t)*** mit der Periode ***T*** kann als ***Summe von harmonischen Vorgängen*** interpretiert werden!

⇒ ***Herstellung von Pseudo-Periodizität:***

- Bei der technischen Frequenzanalyse von Signalen untersucht man ein einlaufendes (o. gespeichertes) Signal in äquidistanten Zeitabschnitten, dem sogenannten BETRACHTUNGSINTERVALL ***T*** *(engl. meist „window")*.



- Für die FREQUENZANALYSE dieses Zeitabschnittes akzeptiert man die Annahme, dass sich das „eingefangene" Signal immer wieder wiederholen könnte, d.h. ***das Window T wird zur PERIODE T erklärt.***
- Innerhalb dieses „Window" kann man nun (gemäß Fourier) alle enthaltenen Frequenzanteile analysieren.
- Durch das Ansetzen / Überlappen des nächsten Zeit-Betrachtungsintervalls T_2 an das zurückliegende und wiederholen der Analyse erhält man insgesamt die kontinuierliche Frequenzanalyse über einen längeren Signalzeitraum.

⇒ ***Länge des Betrachtungsintervalls T ↔ f_{min}-Grenze der Analyse:***

Je größer / länger das Window T, desto niedrigere Frequenzanteile können darin exakt analysiert werden!

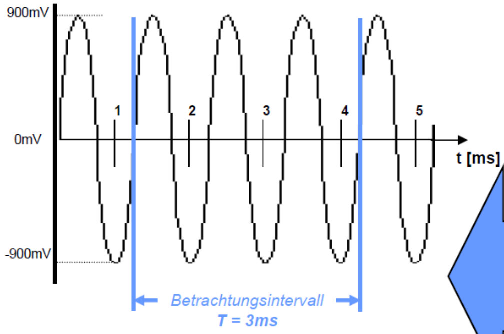
Beispiel AUDIO:

Signalfrequenzbereich: **20 Hz** – 20KHz   Minimales Betrachtungsintervall: T = $^1/_{20}$ Hz = 0,05 Sekunden = 50 ms (entspricht ca. 2048 Abtastwerten mit f_s = 48 KHz)

Einfaches Beispiel zur Technischen Frequenzanalyse:

Bsp.: Analoges Sinussignal

$f(t)$ = 900 mV * sin (2π1kHz * t)



$c_i = 1/T \times \int_{-T/2}^{+T/2} f(t) \times e^{-j\,(i*2\pi*t/T)}\,dt$

Fourier – Analyse:

$c_0 = 0$, $c_1 = 0$, $c_2 = 0$, $c_3 = 900mV$

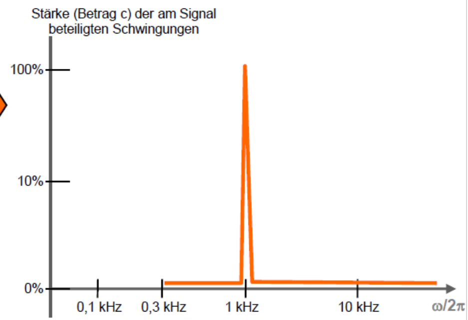
alle anderen $c_i = 0$,

untere Analysegrenze wg. T = 3ms: **333Hz**

Inverse Fourier-Transformation

$f(t) = c_0 + \sum_{i=-\infty}^{+\infty} c \times e^{-j\,(i*2\pi*t/T)}$

Analoge Fourier-Funktion $|F(\omega)|$:



Erkenntnisse:

→ Komplizierte Zeitsignale ergeben oft sehr einfache Frequenzfunktionen!

→ Die Transformation vom Zeit- in den Frequenzbereich bietet somit Vorteile bei der

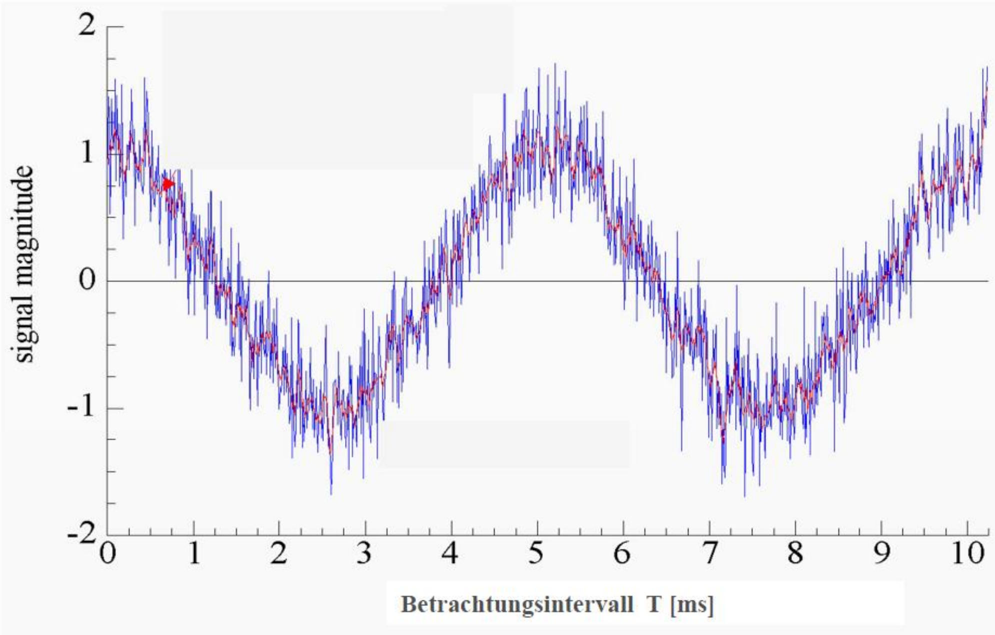
- ***Signalanalyse*** (man erkennt schnell, welche Frequenzanteile die MASSGEBLICHEN im Signal sind!)

  und der

- ***Signalverarbeitung*** (gezielte Manipulation von Fourierkoeffizienten führt zu gezielten *Frequenzmanipulationen* am Signal!)

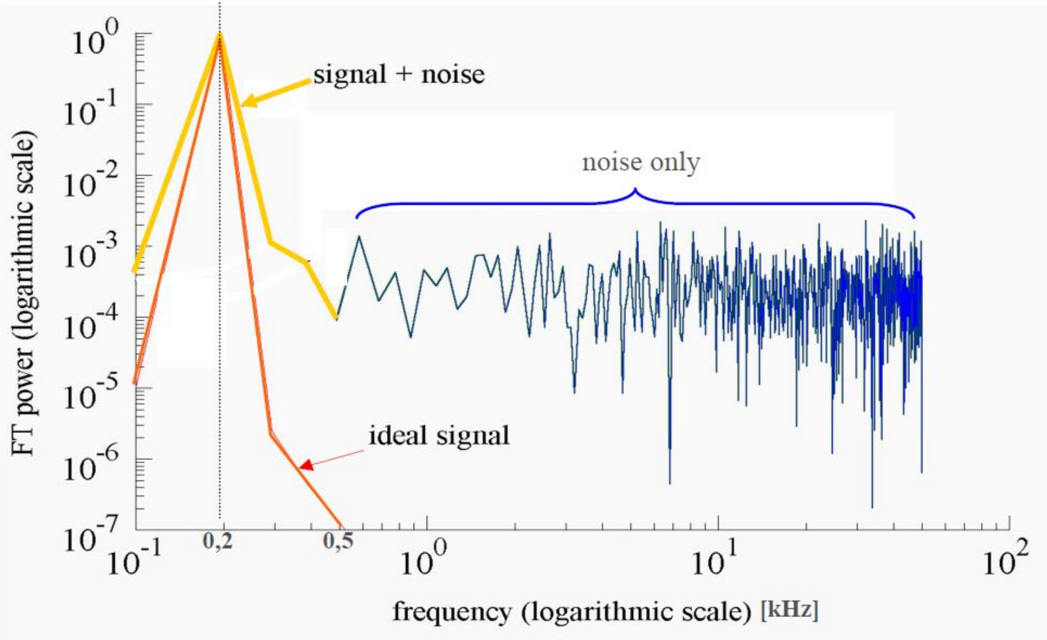
Reales Beispiel: Audio-Signalprobe von 10ms

Welche Signale sind hier zusammengemischt?



Frequenzanalyse der Audio-Signalprobe von 10ms

Ergebnis der Fourier-Analyse!



Wie geht das?

Transformation von *digitalen* Abtastwerten in den Frequenzbereich

1. Abtastheorem (Nyquist) und A/D-Wandlung

Welches ist die höchste Frequenz F_{max}, die **in** einem realen, mit f_{sample} digitalisierten, Signal analysiert werden kann?

$F_{max} = f_{sample} / 2 \Rightarrow$ natürliches Abbruchkriterium der Frequenzanalyse

2. Fourier: aus *N, äquidistanten & diskreten Zeitwerten* berechnen sich zwangsläufig *L äquidistante & diskrete Frequenzwerte*

$$X(\Omega_l) = 2/N \sum_{k=0}^{N-1} x_k \times e^{-j(k * l*2\pi/N)} \quad \text{mit} \quad e^{-j(k*l*2\pi/N)} = \cos(k*l*2\pi/N) - j \sin(k*l*2\pi/N)$$

3. Genauigkeit der diskreten Fourier-Analyse

Frequenzpunkte Ω_l an denen sich diskrete, äquidistante Fourier-Koeffizienten X ergeben:

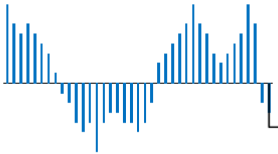
$\Omega_l = l * 2\pi/(N*T) = l * \omega_{min}$ mit $l = 0,1,2,3, \ldots N-1$

Spektrale Auflösung der diskreten Fourier-Transformation: $\omega_{min} = 2\pi/T_{window} = (2\pi)* f_{min}$ *[engl. „bin"]*

Je größer das Window T, desto kleiner $f_{min} = 1/T$, desto genauer die Frequenzanalyse!

DFT schnell im Kopf „durchführen“

Input:



Analyse:
Aus welchen Einzel-Schwingungen setzt sich das Signal zusammen ?

DFT-Analyse

$$X(\Omega_l) = 2/N \sum_{k=0}^{N/2} x_k \times e^{-j(k*l*2\pi/N)}$$

Ergebnis:	Amplitude	Frequenz	Phase
Schwingung1:	X_0	$f_{0\,=\,0Hz}$	φ_0
Schwingung2:	X_1	f_{min}	φ_1
Schwingung3:	X_2	f_2	φ_2
Schwingung4:	X_3	f_3	φ_3
Schwingung5:	X_4	f_4	φ_4
Schwingung6:	X_5	f_5	φ_5
....			
Schwingung N:	X_n	f_n	φ_n

„Fourier-Koeffizienten“

Bsp.: Wave-Datei

f_s=48kHz
Analysefenster T=24ms

→ Anzahl der Abtastwerte im Analysewindow:
N = 24ms * 48kHz
= 1152 Samples im Analysewindow

⇓

f_{max}= f_s/2 = 24kHz

f_{min}= 1/T = 1/24ms = 41,67Hz („bin“)

⇓

Koeffizienten können nur bei folgenden Frequenzen auftreten: f_l = l x f_{min} [$\Omega_l = 2\pi \cdot l \cdot f_{min}$]

In unserem Beispiel also nur bei : 0Hz / 41,67Hz / 83,3Hz / 125Hz / ....

Nach f_{max}: N/2 = 576 Fourierkoeffizienten kann die Analyse abgebrochen werden, da f_{max} erreicht.

Beispiel für eine ***Diskrete Fourier-Transformation*** (DFT)

Eine wave-Datei besteht aus 8 Abtastwerten $\boldsymbol{x_k}$, die mit einer Abtastfrequenz von 32 kHz und Q = 6bit erzeugt wurden. Welche Schwingungen sind in diesem Audiosignal enthalten ?

$x_0 =$	20_{dez}	$(010100)_{bin}$
$x_1 =$	0_{dez}	$(000000)_{bin}$
$x_2 =$	-20_{dez}	$(110100)_{bin}$
$x_3 =$	0_{dez}	$(000000)_{bin}$
$x_4 =$	20_{dez}	$(010100)_{bin}$
$x_5 =$	0_{dez}	$(000000)_{bin}$
$x_6 =$	-20_{dez}	$(110100)_{bin}$
$x_7 =$	0_{dez}	$(000000)_{bin}$

Fourier – Analyse:

Schritt 1: Dauer T der Wave-Datei: $T = 8 / f_{sample} = 8 / 32\text{kHz} = 0{,}25$ ms

Schritt 2: Spektrale Auflösung der diskreten Fourier-Transformation („bin"):
$f_{sample}/N = 32000\text{Hz} / 8 = \mathbf{4\ kHz = 1/T}$

Schritt 3: Frequenzpunkte $f_l = l * 4\text{kHz}$ $\quad l = 0,1,2,3, \dots N-1$

Schritt 4: sinnvoller Abbruch der Reihenentwicklung für $\Omega_{max} \approx 2\pi * f_{sample} /2$
$l * 4000\text{Hz} \approx 16\text{ kHz} \rightarrow \mathbf{l = 4}$ mit $\Omega_4 = 2\pi * 16$ kHz

Schritt 5: Berechnung der Frequenzkoeffizienten (s.a. Lösungsblatt im Intranet)

$$X(l=0) = x_0 \times 1 + x_1 \times 1 + x_2 \times 1 + \dots = 0 \rightarrow |X(0)| = 0$$

$$X(l=1) = x_0 \times e^{-j(0*2\pi/8)} + x_1 \times e^{-j(1*2\pi/8)} + x_2 \times e^{-j(2*2\pi/8)} + x_3 \times e^{-j(3*2\pi/8)} + \dots = 0 \rightarrow |X(1)| = 0$$

$$X(l=2) = x_0 \times e^{-j(0*4\pi/8)} + x_1 \times e^{-j(1*4\pi/8)} + x_2 \times e^{-j(2*4\pi/8)} + x_2 \times e^{-j(3*4\pi/8)} + \dots = 80 - j\,0 \rightarrow |X(2)| = 80$$

$$X(l=3) = x_0 \times e^{-j(0*6\pi/8)} + x_1 \times e^{-j(1*6\pi/8)} + x_2 \times e^{-j(2*6\pi/8)} + x_2 \times e^{-j(3*6\pi/8)} + \dots = 0 \rightarrow |X(3)| = 0$$

$$X(l=4) = x_0 \times e^{-j(0*8\pi/8)} + x_1 \times e^{-j(1*8\pi/8)} + x_2 \times e^{-j(2*8\pi/8)} + x_2 \times e^{-j(3*8\pi/8)} + \dots = 0 \rightarrow |X(4)| = 0$$

$X(l=5) \rightarrow$ Abbruch

<u>Reale Algorithmen zur Frequenzanalyse</u>

1. *DFT mit Zero-Padding*

- Aufgrund des Abtasttheorems können aus N-Zeitabtastwerten nur **N/2** gültige Frequenzpunkte berechnet werden, d.h. man erhält de facto eine reduzierte spektrale Auflösung. Für eine <u>feine</u> Frequenzanalyse manchmal ZU WENIG.

- Mit einem einfachen Trick kann man aber aus N-Abtastwerten sehr wohl N-Frequenzkoeffizienten berechnen:
  Man erweitert die Zeitprobe virtuell um ca. N/4-Abtastwerte links und rechts, mit dem Wert „0". Die DFT läuft nun also über ein window mit 1,5 * N Abtastwerte. Unter Einhaltung der Nyquist-Regel ergeben sich somit nach Reihenabbruch $\approx$ N Frequenzkoeffizienten

---

2. Beschleunigte Variante: *Fast Fourier-Transformation (FFT)* http://www.nauticom.net/www/jdtaft/papers.htm

- *Danielson und Lanczos* zeigten 1942, dass eine Diskrete Fourier-Transformation der Länge N mit N = 2^X / x = ganzzahlig, positiv
  - als Summe zweier DFTs der Länge N/2,
  - als Summe dreier DFTs der Länge N/3 …. usw.

  $\rightarrow$ ergo als ***Summe von N DFTs der Länge N/N = 1 angesehen werden kann !***

- Damit ist in Rechenprogrammen eine - gegenüber der Standdard-DFT verkürzte - rekursive Lösungsmöglichkeit realisierbar, um die jeweiligen Frequenzkoeffizienten noch schneller zu berechnen => Butterfly-Algorithmus

---

3. Beschleunigte und Auflösungs-verbesserte Variante: *Diskrete Cosinus-Transformation (DCT)*

- Mathematischer Trick, um aus **N** Zeitabtastwerten sehr schnell **N** Frequenzkoeffizienten zu erhalten:
  Man entnimmt dem Signalverlauf eine Probe von N Abtastwerten und verdoppelt diese durch Spiegelung an der Amplitudenachse. Dann Fast-Fourier-Analyse (FFT) der 2N - Abtastwerte, wobei nur die Cosinus-Koeffizienten von Relevanz sind, da das Signal nun spiegelsymmetrisch zur Y-Achse ist. Im Ergebnis der Analyse erhält man sehr schnell 2 N Frequenzkoeffizienten von denen mindestens N unterhalb der Nyquistgrenze liegen (= verbesserte Frequenzauflösung geg. FFT + genauso schnell)

Prinzipielle Funktionsweise von digitalen Filtern
(am Beispiel Tiefpass)