Module IN3031 / INM378 Digital Signal Processing and Audio Programming

Tillman Weyde

t.e.weyde@city.ac.uk

Filter Banks
Multi-Resolution Analysis
(separate slides)
Deep Learning and DSP

Real Time Digital Filters

- Common filters are linear time-invariant (LTI) systems, i.e. they do not change their behaviour depending on time or amplitude
  - FIR finite impulse response filters implement convolution
  - IIR infinite impulse response filters add recursion
- Filters for Real Time processing are **causal**
  - They use only past values

Uses of Digital Filters

- Digital filters are used for
  - anti-aliasing (before downsampling)
  - equalisation (removing frequency imbalances of microphones, room acoustics etc)
  - user sound modification (adjust to personal taste)
  - sound analysis (select the frequency range to analyse)
  - sound synthesis (shape the timbre of a synthetic sound)

Types of Digital Filters

- There are four common types of filters:
  - low pass (anti-aliasing, synthesis)
  - high pass (remove unwanted noise, protect speakers)
  - band pass (sound analysis)
  - band stop (removing unwanted noise, e.g. from power supply)
- Other types of filters:
  - comb filters (usually the result of short delays)
  - all pass filters (modify only the phase)

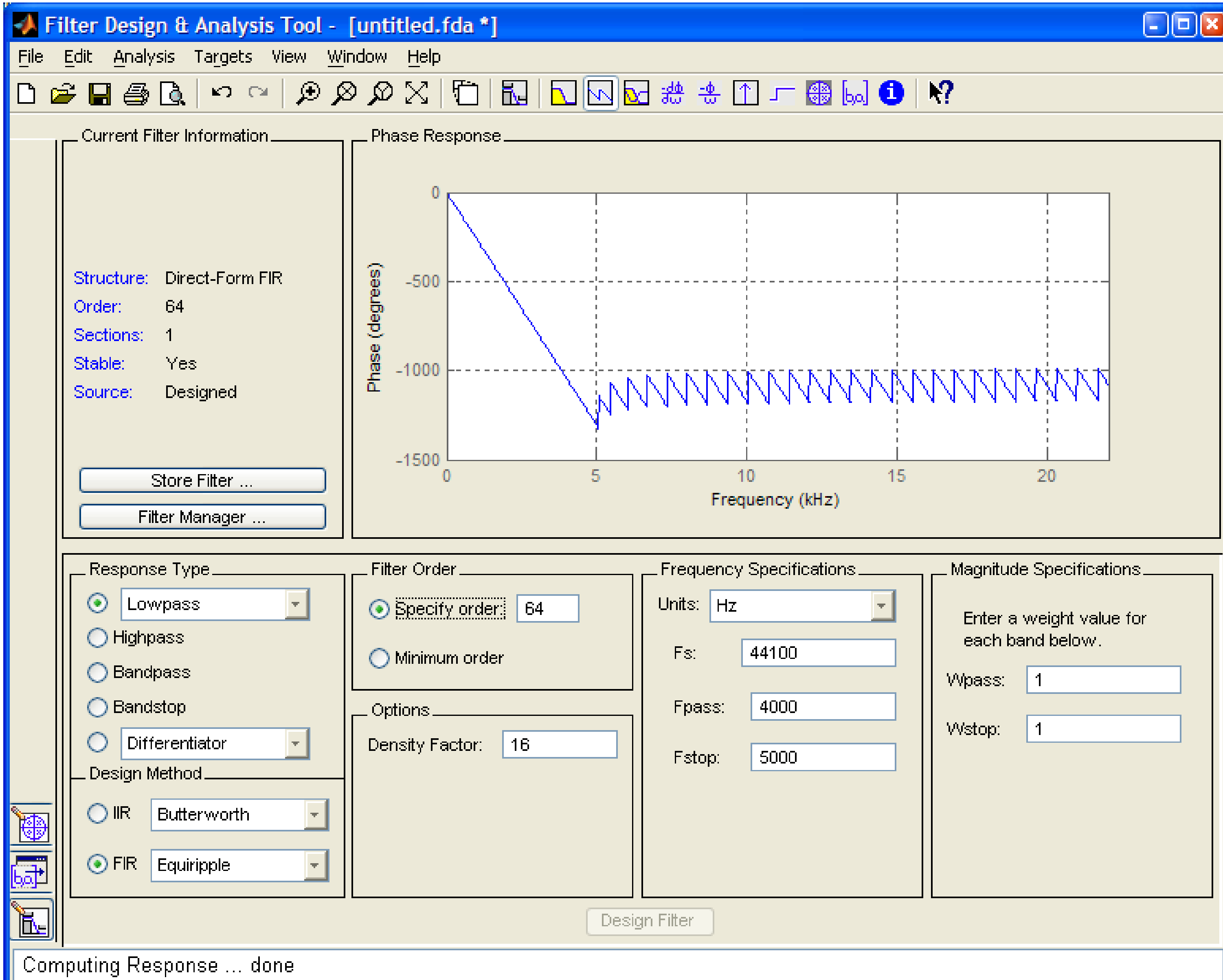
Properties of Digital Filters

- Filter architecture (FIR or IIR)
- Filter order (#sample delays = #coefficients-1)
- Filter coefficients (the values defining the filter operation)

resulting from these

- Frequency response
- Impulse response (sometimes step response)
- Time behaviour (phase response, group delay)

Phase Response



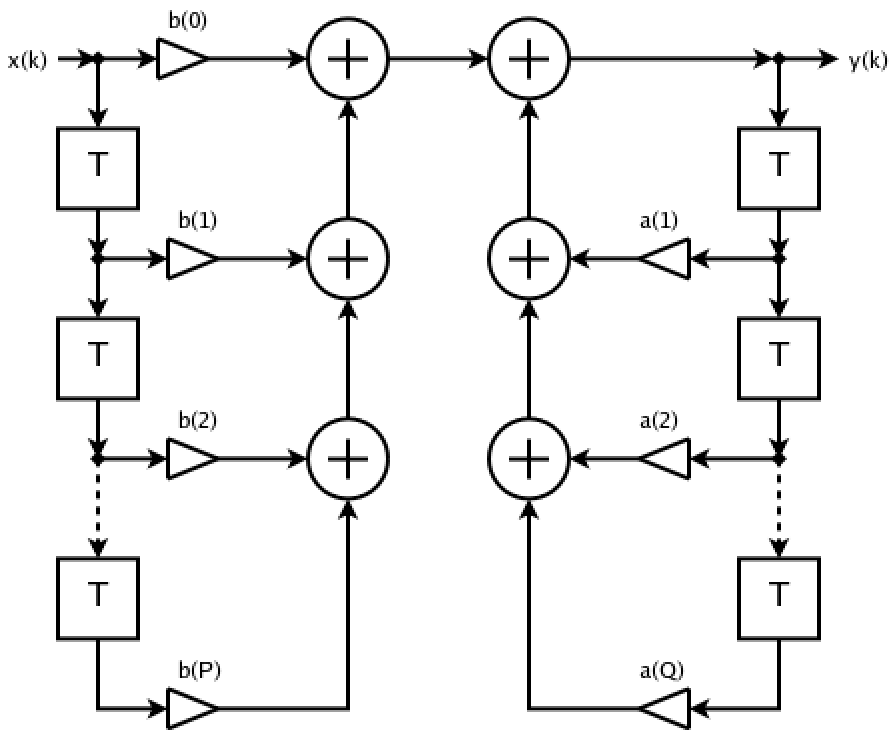
IIR Filter Design

IIR:

- Approach: Define numerical methods for finding appropriate filter coefficients (mathematically demanding).
- Pro:
  - IIR filters can be very efficient.
- Cons:
  - Uncontrolled phase behaviour
  - May be unstable
  - Quantisation noise may multiply through recursion

IIR Filter Diagram

$$y[n] = -\sum_{i=1}^{k} a_i \, y[n-i] + \sum_{i=0}^{k} b_i \, x[n-i]$$

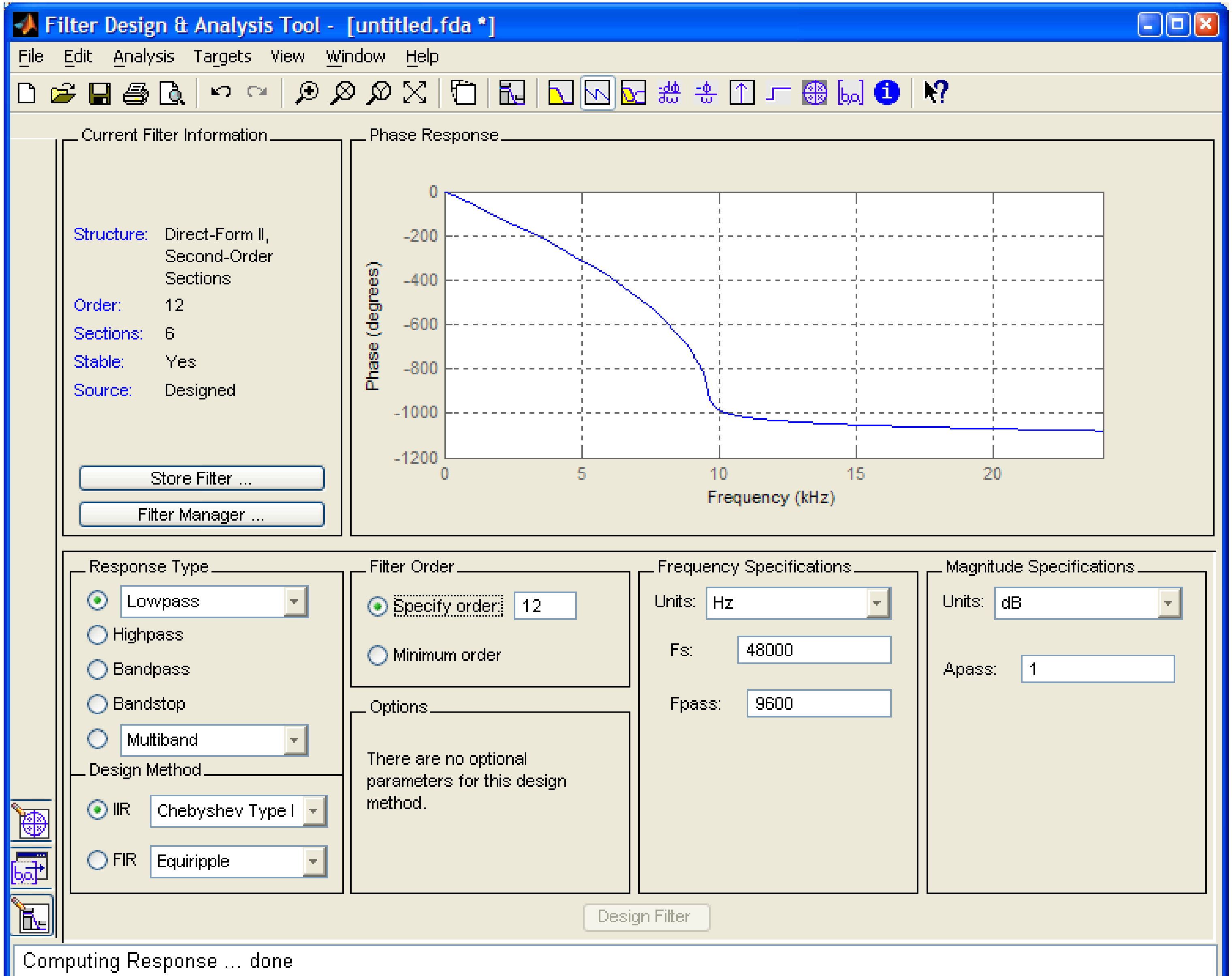


T represents a delay by one sample

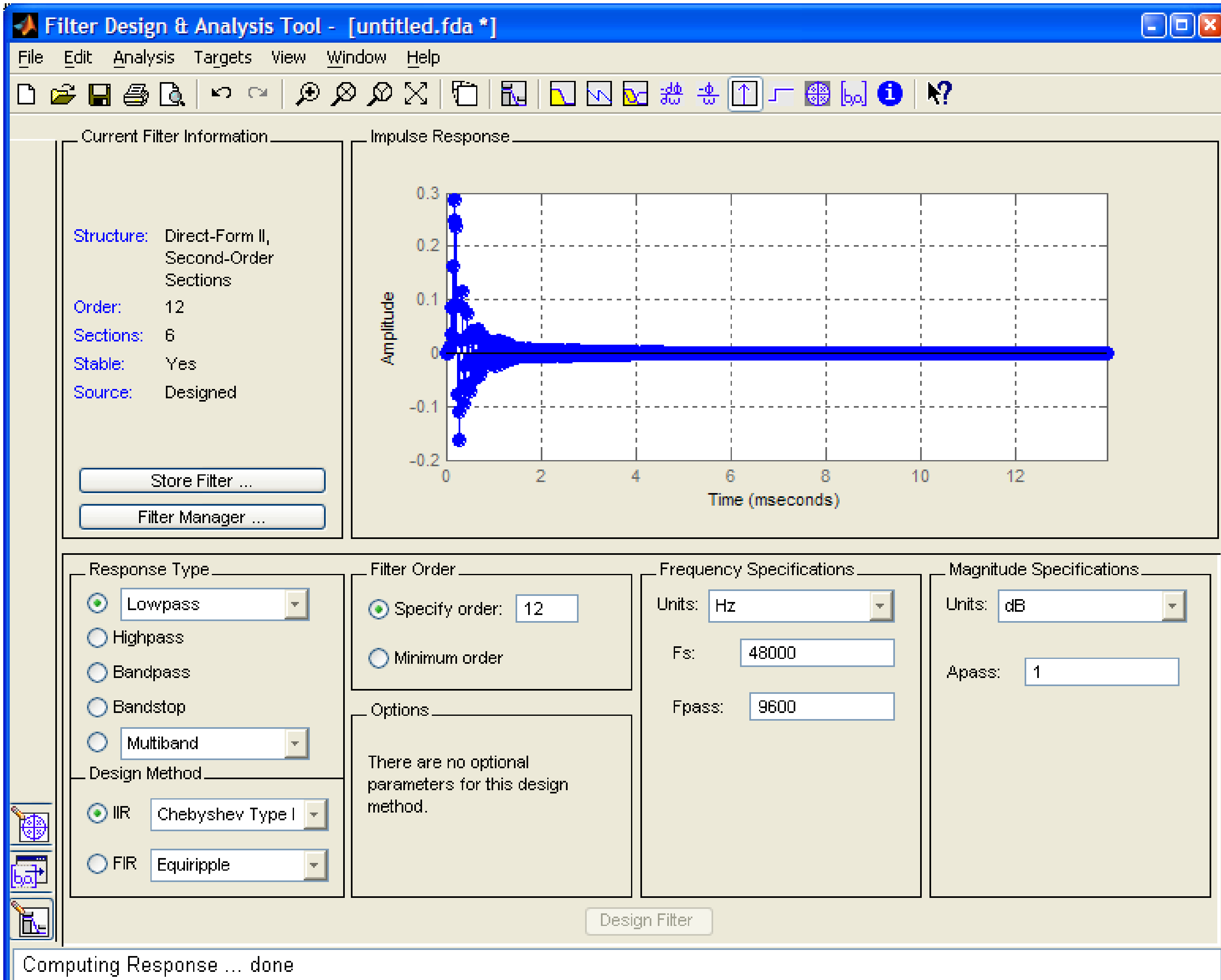
The Impulse Response of an IIR Filter

- An IIR filter g
  $g(x[n]) = - a_1 g(x[n-1]) - a_2 g(x[n-2]) - \ldots - a_k g(x[n-k])$
  $+ b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] + \ldots + b_k x[n-k]$
- The impulse response of g has to be computed recursively and may be infinitely long
- IIR filters
  - allow very effective filtering with few coefficients
  - may oscillate by themselves
  - frequency response is hard to compute

Phase Response



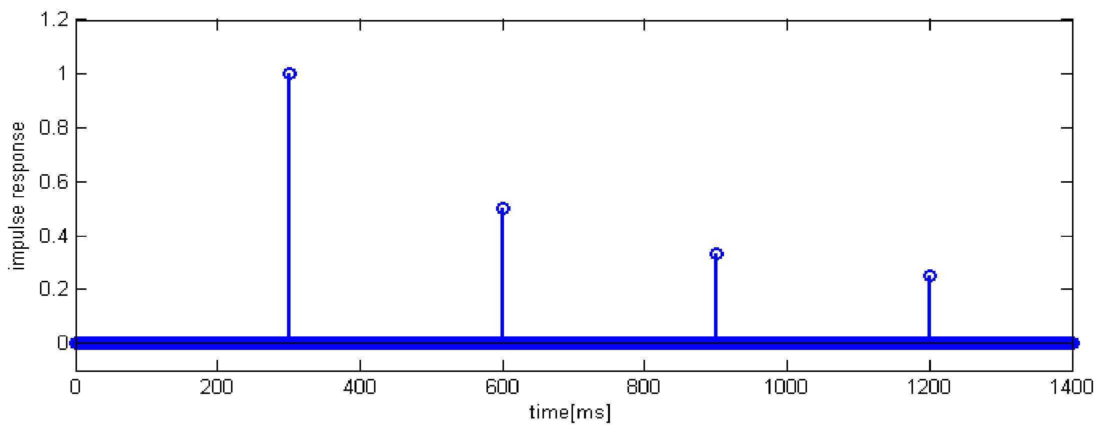
Impulse Response



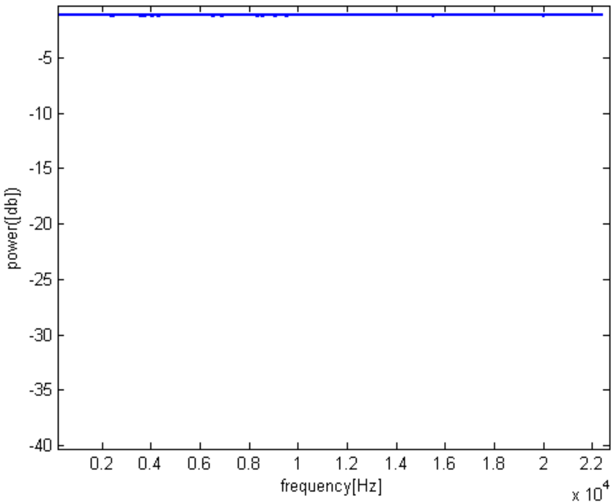
Impulse Responses and Effects

Echo

- Impulse response: Few filter coefficients **span over seconds**
- Frequency response is **flat**



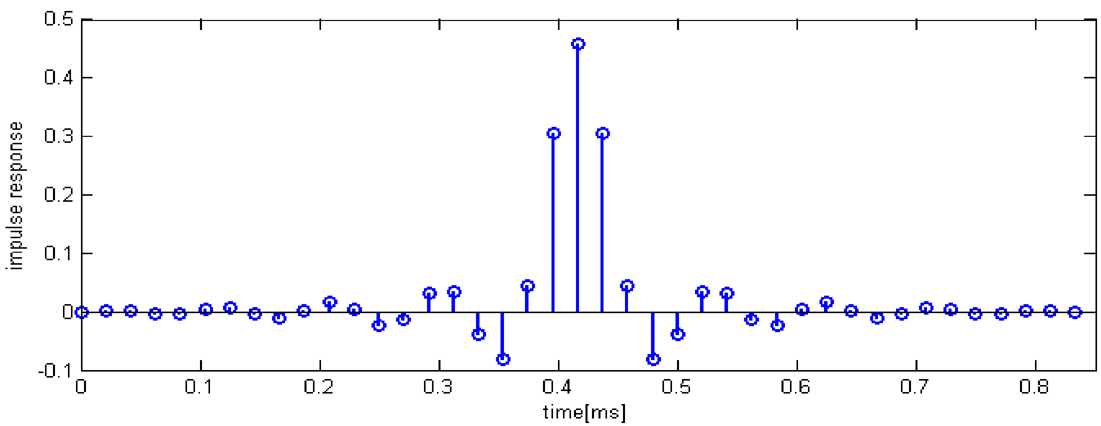
Impulse Response



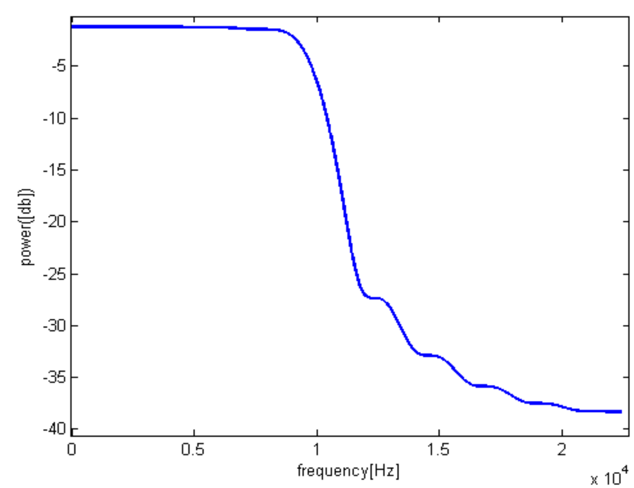
Frequency Domain

Low-Pass

- Impulse response: Many filter coefficients in the **first few milliseconds**
- **Approximates a rectangular window** in frequency domain



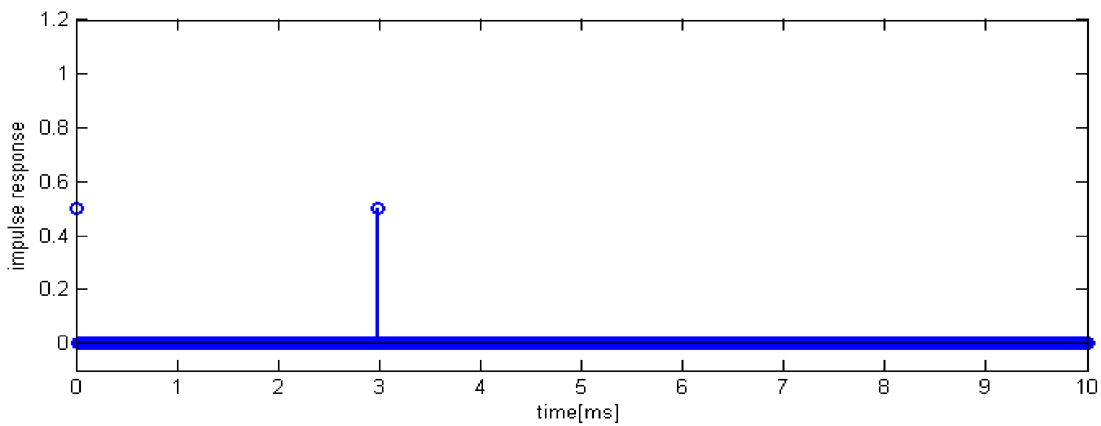
Impulse Response



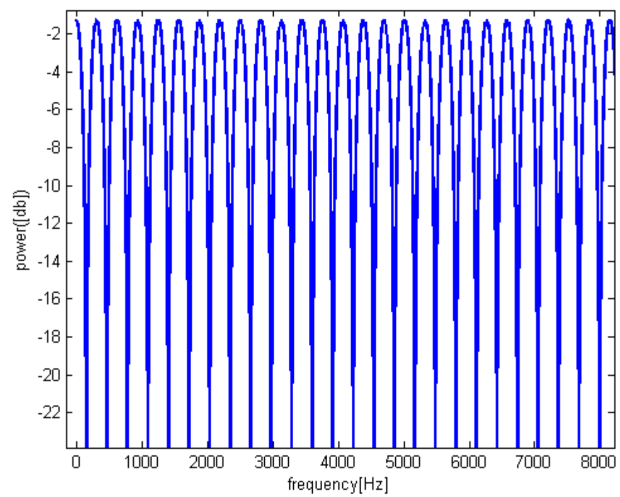
Frequency Domain

Flanger as Filter

- Impulse response: 2 filter coefficients in the **first few milliseconds (identity + delay)**
- **Comb filter shape** in frequency domain
- IR changes over time



Impulse Response

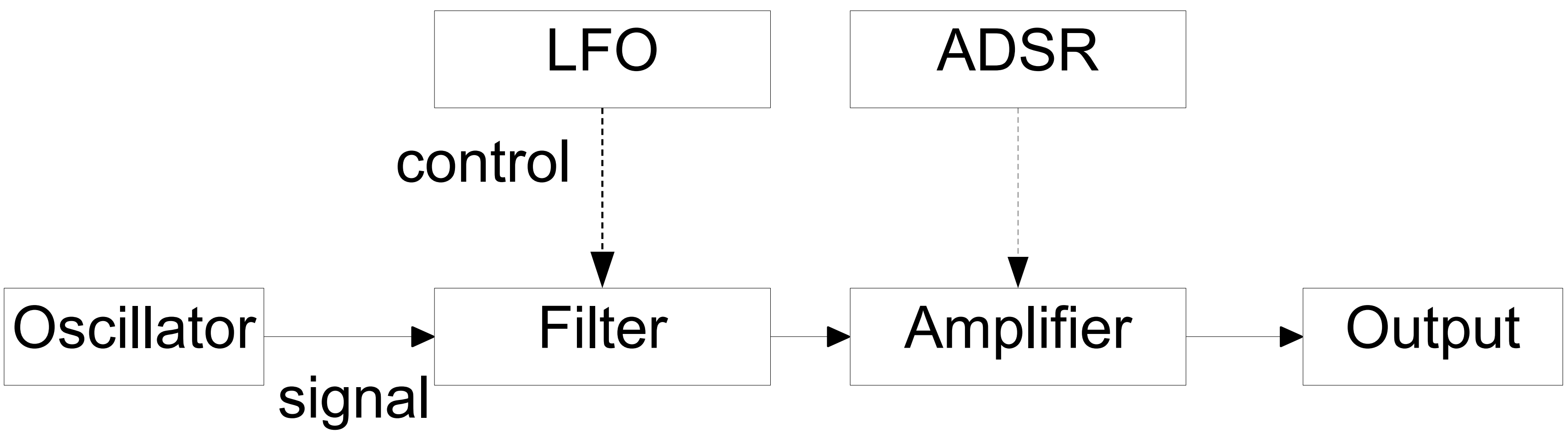


Frequency Domain

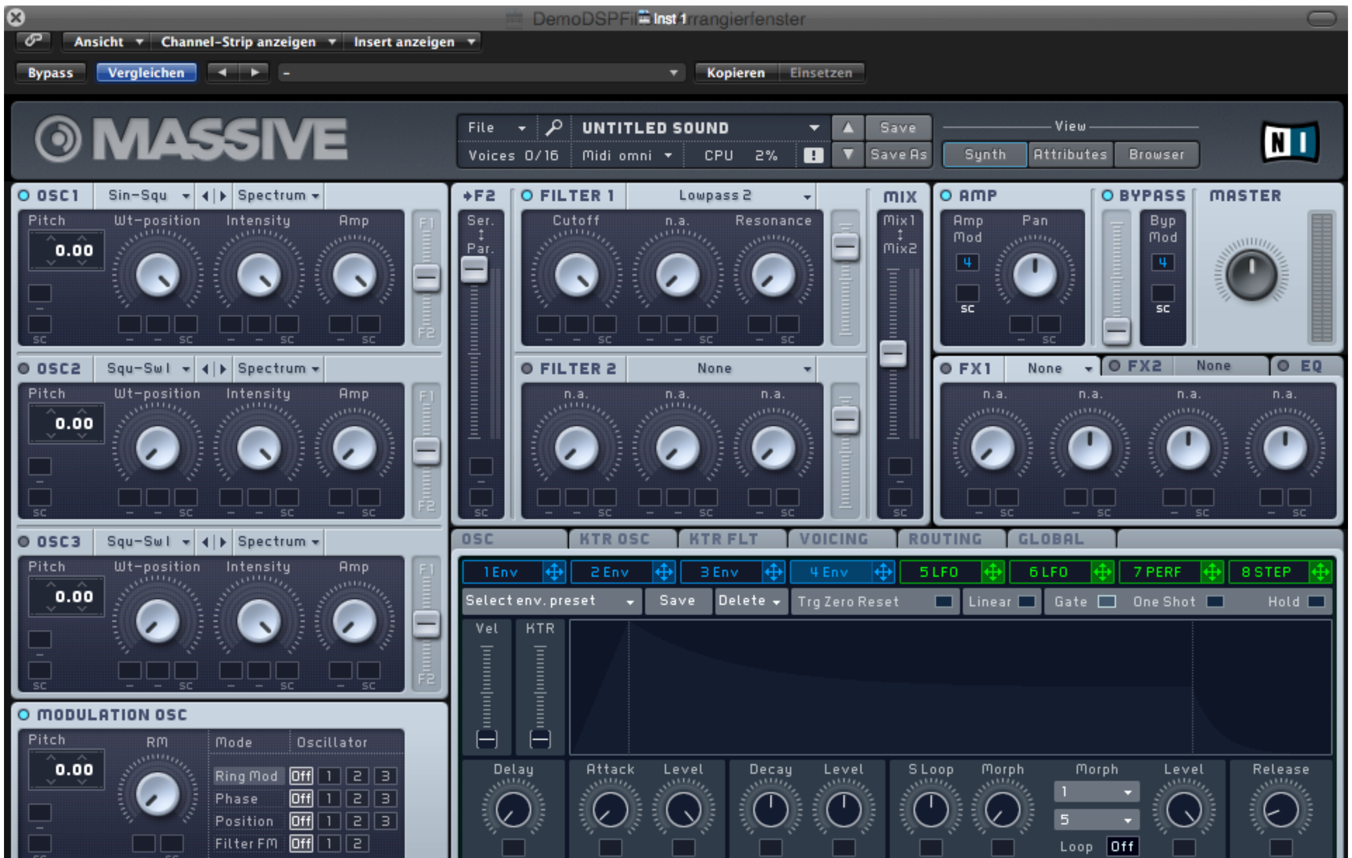
Using Filters for Subtractive Synthesis

Subtractive Sound Synthesis

- Most common form of analogue synthesis
- Generates a sound, filters and amplifies (attenuates) it
- Exemplary set-up:

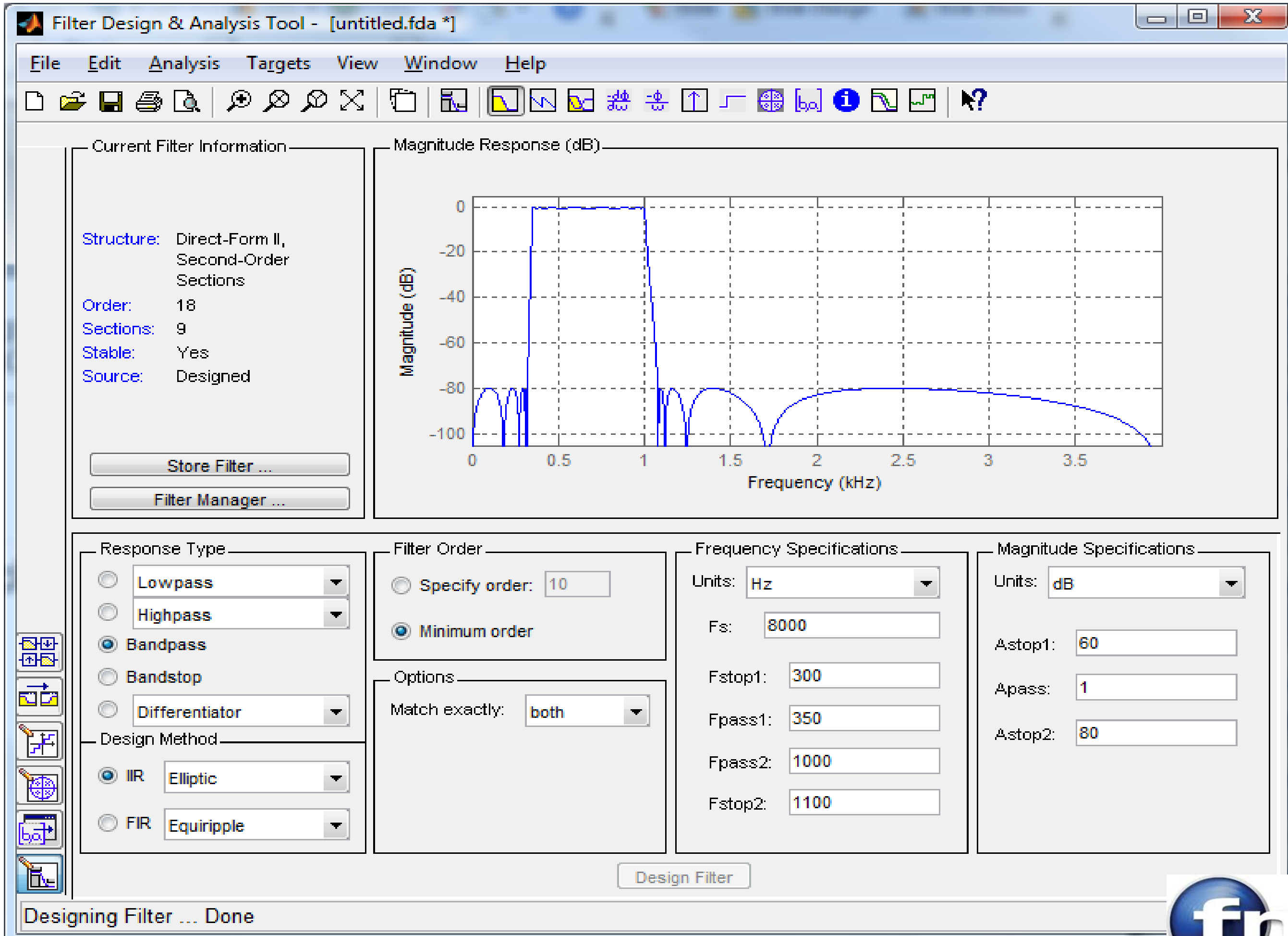


“Virtual” Synthesizer

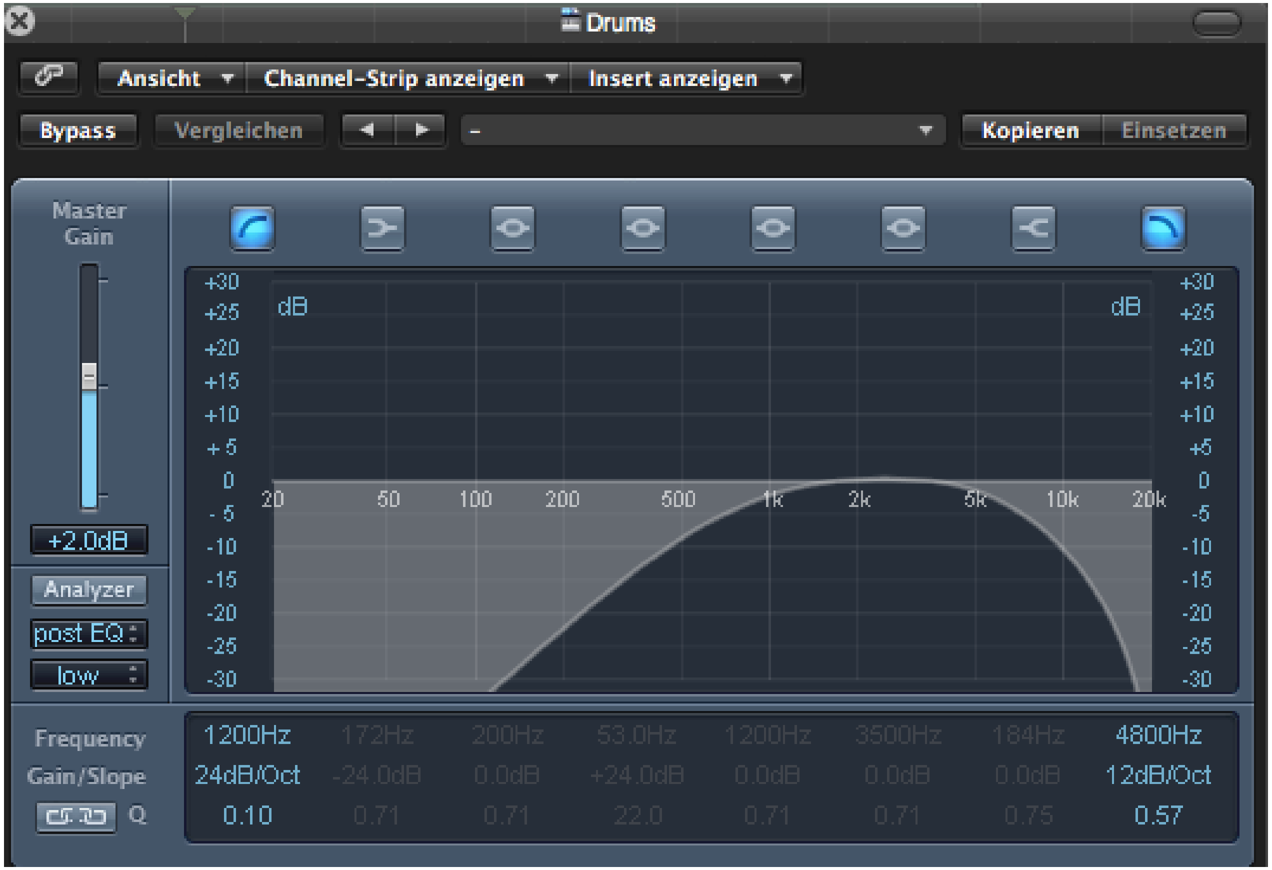


Filter Banks

Band Pass Filter



Band Pass in Music Production

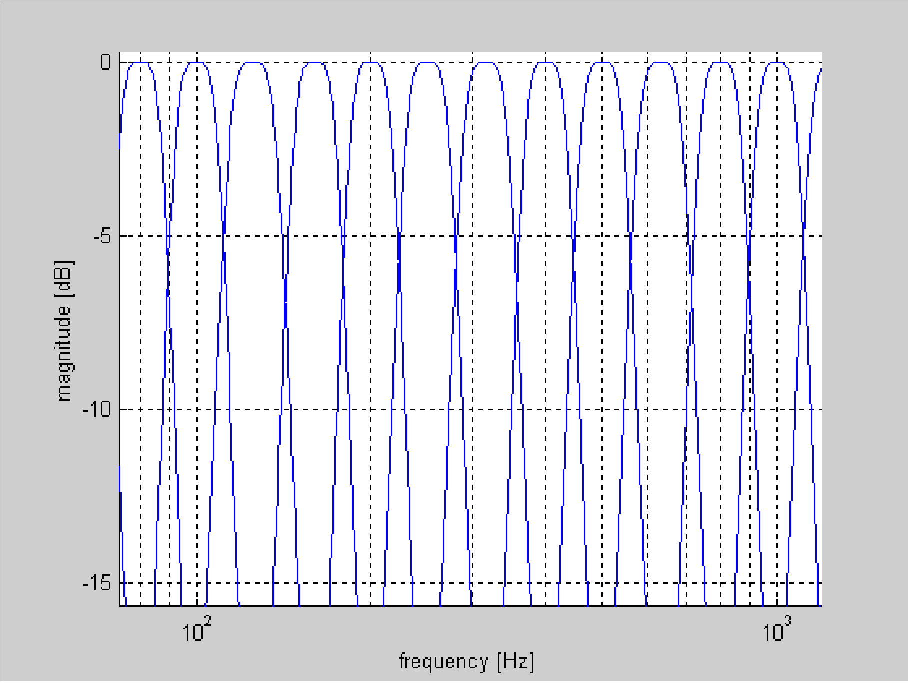


Filter banks

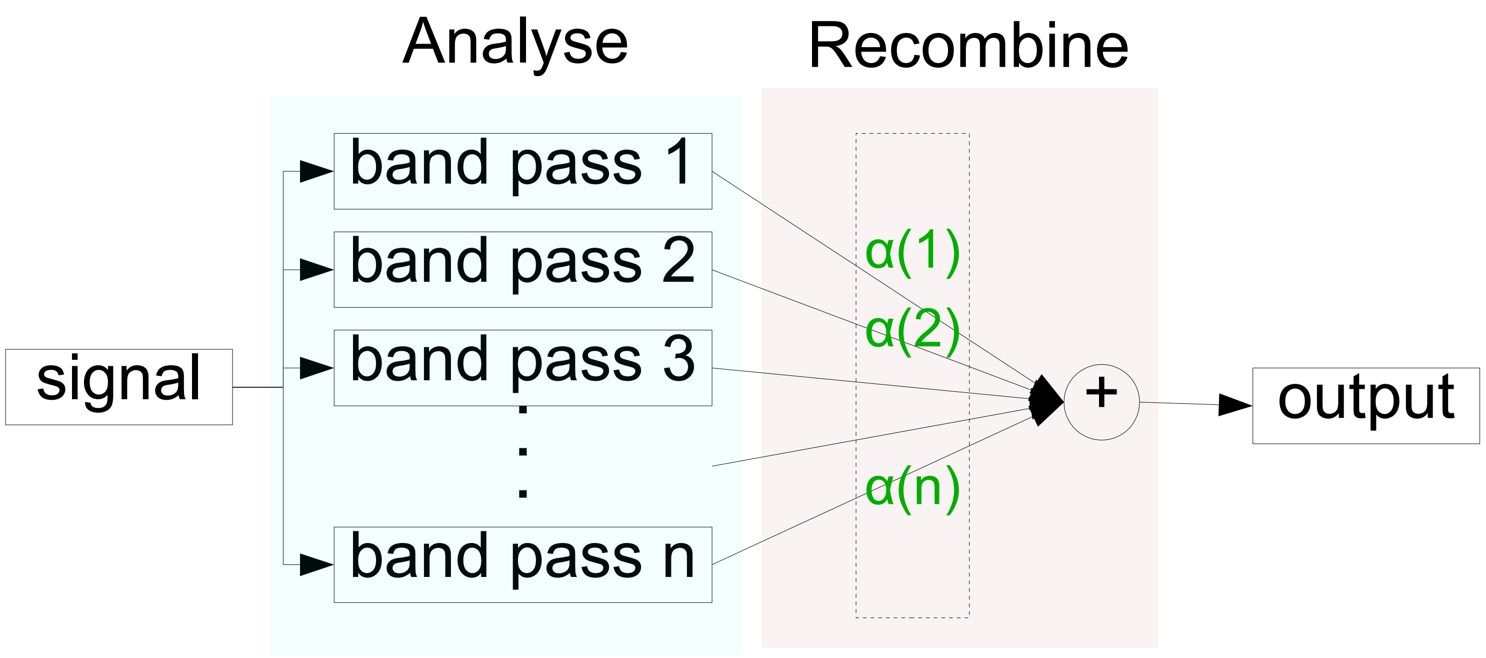
- Multiple band pass filters
  - Logarithmically spread over the spectrum
  - Overlapping up to half the pass band
- Analyse distinct frequency bands
  - Critical bands in hearing
  - Real time alternative to Fourier analysis
- Resynthesis used in Vocoders

Filter Bank Pass-Bands

- Multiple band pass filters
  - Cover the spectrum
  - Magnitude responses add up to 1 (approx)

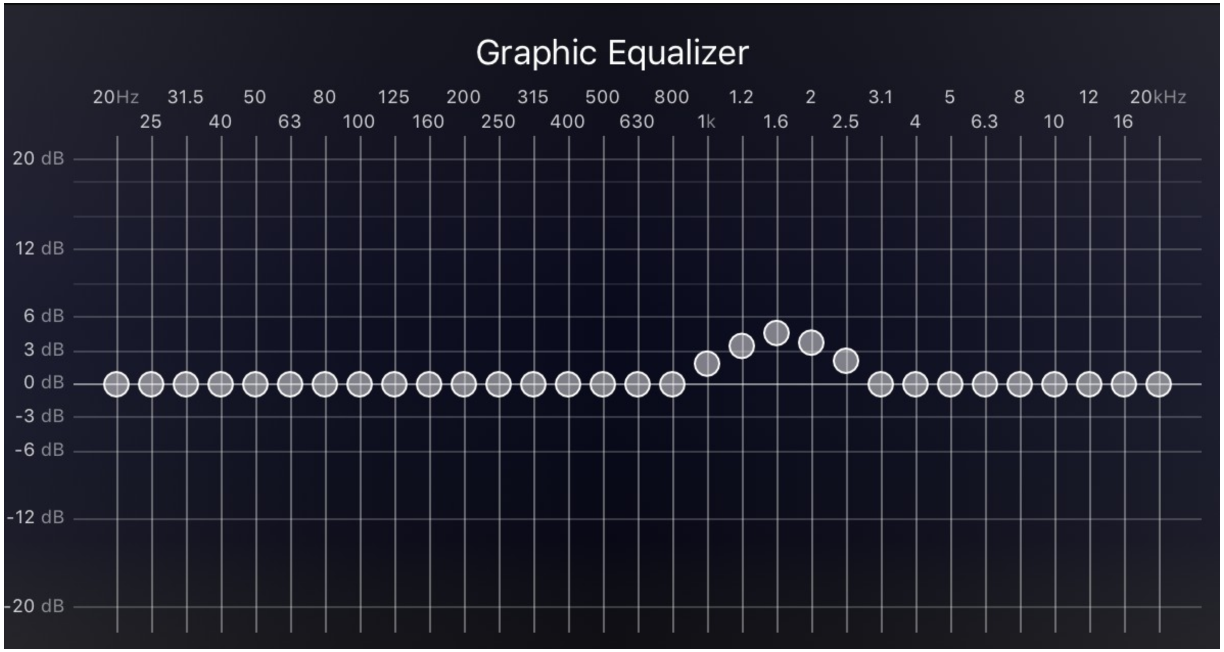


Equalisation with Filterbank



Filter Bank EQ

- Typically bands spaced in octaves or thirds



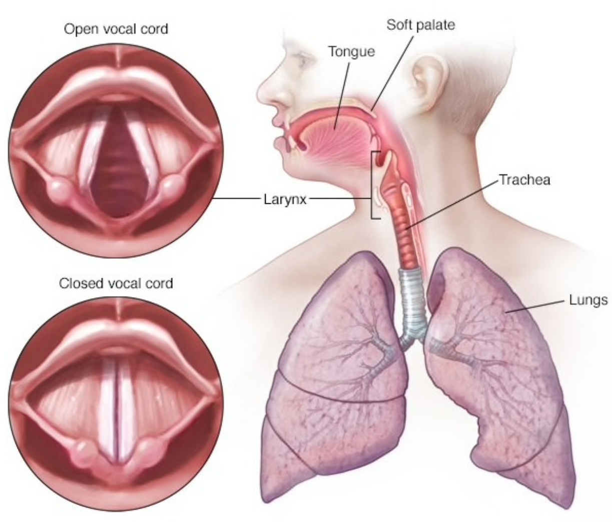
Filter Bank Vocoder

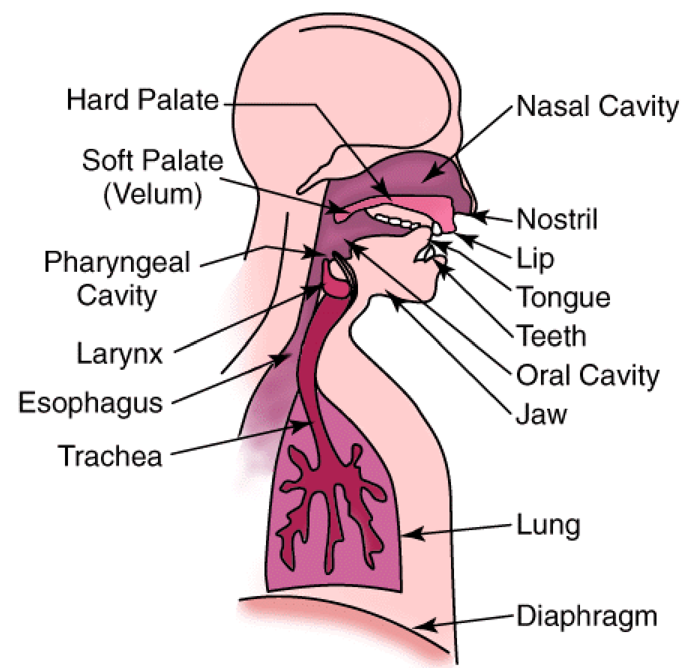
Vocoder: voice coder

- Idea
  - Extract the spectrum from speech
  - Reduce spectrum
  - Transfer the spectrum to another signal (carrier)
- Originally developed for communication

Filter Bank Vocoder

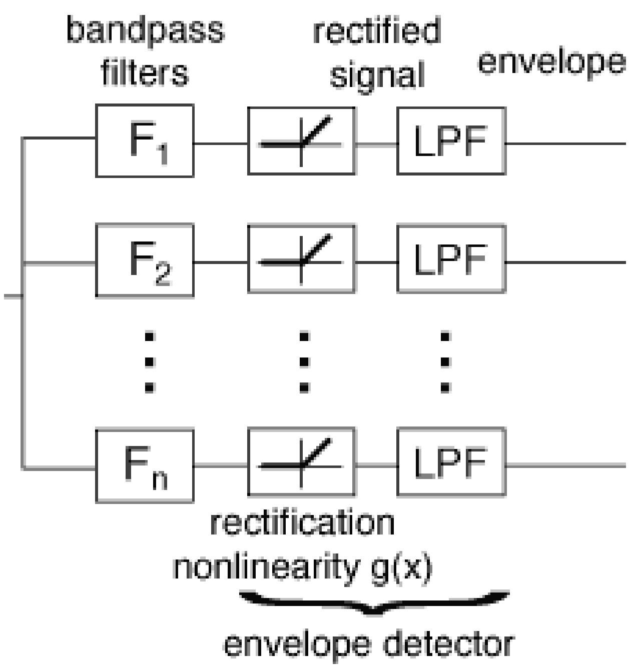
- Why is this a good idea?
  - human voice can be modelled as
    - sound source (vocal chords)
    - and a filter (vocal tract)





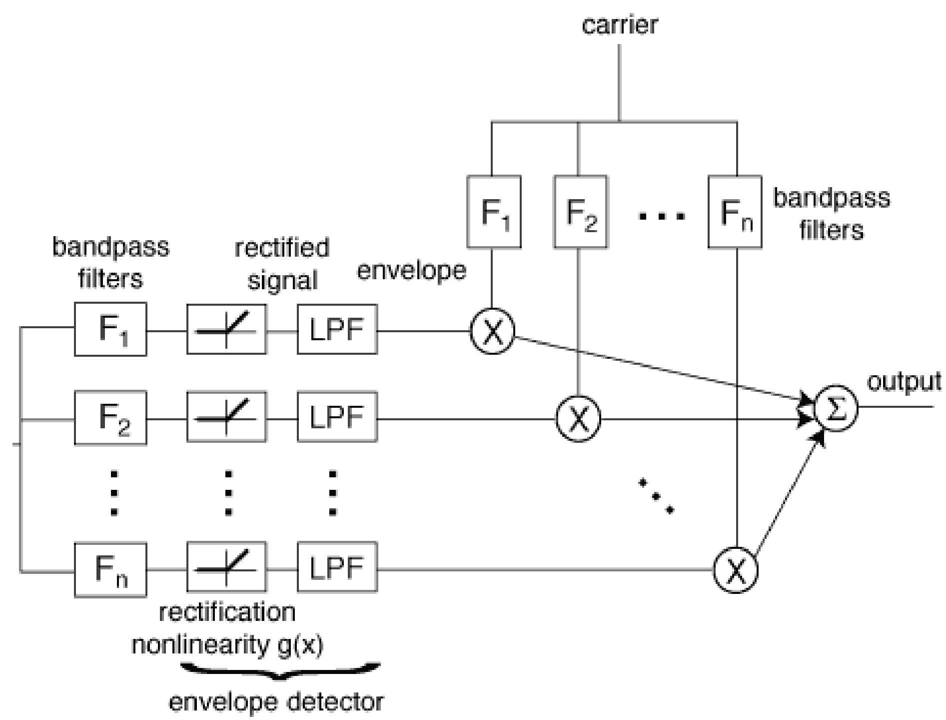
Filter Bank Analysis

- Real-time continuous spectral analysis
- Operation
  - Split the signal into frequency bands
  - Track the amplitude envelope (rectify and low-pass filter)



Filter Bank Synthesis

- Operation
  - Use a carrier signal (broadband)
  - Split into frequency bands
  - Amplify per band with voice envelope



Vocoder Use

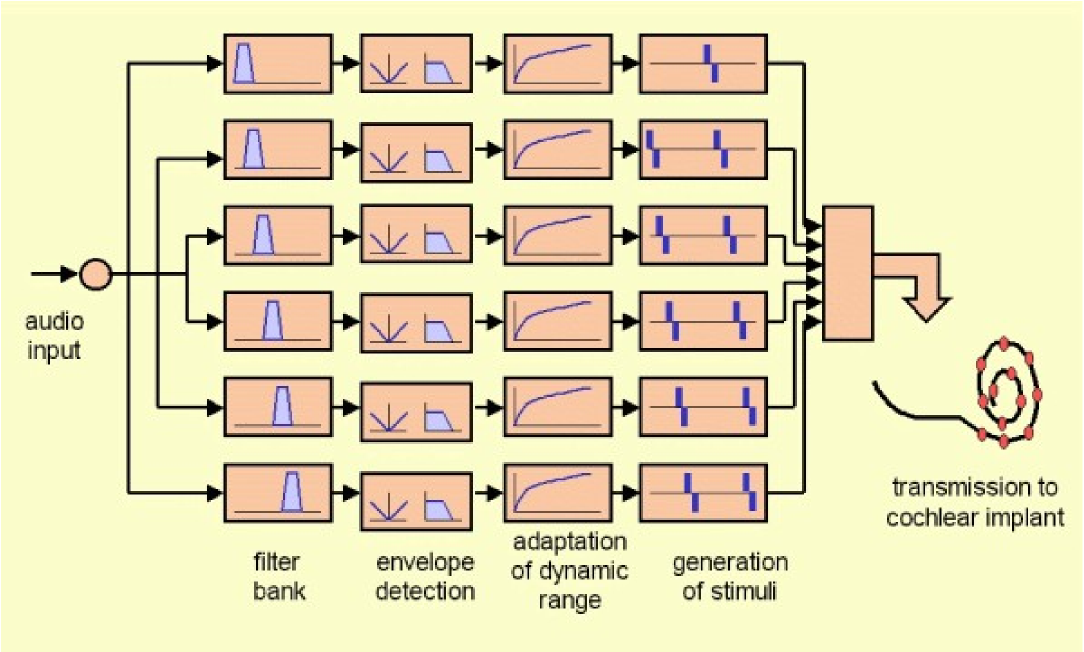
- Films
  Computer and robot voices

- Music examples
  Electronic: e.g. Kraftwerk
  Pop: e.g. Earth Wind and Fire, Electric Light Orchestra

- A 'talkbox' is a acoustical version of the vocoder Films
  Computer and robot voices

Cochlear Implants

- Similar architecture to vocoders
- Need to generate neural compatible stimuli at the end

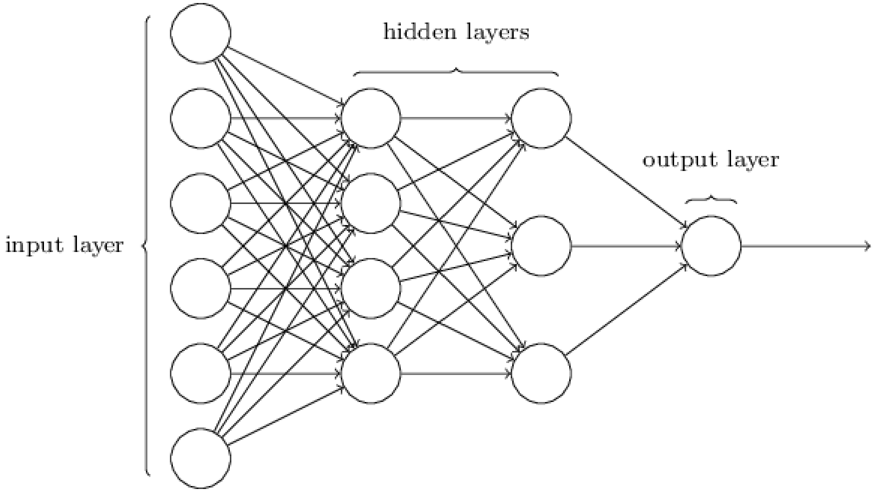


Neural Networks in DSP

- **Universal regression** models
  - **Map** continuous values to continuous values
  - Can be operate in **high dimensions**
- **Inspired** by **biological** neural systems (brains)
  - Consist of many neurons
  - (Almost) every neuron has a **non-linearity**
  - Trained via **gradient descent**
  - Can have many layers → **Deep Learning**

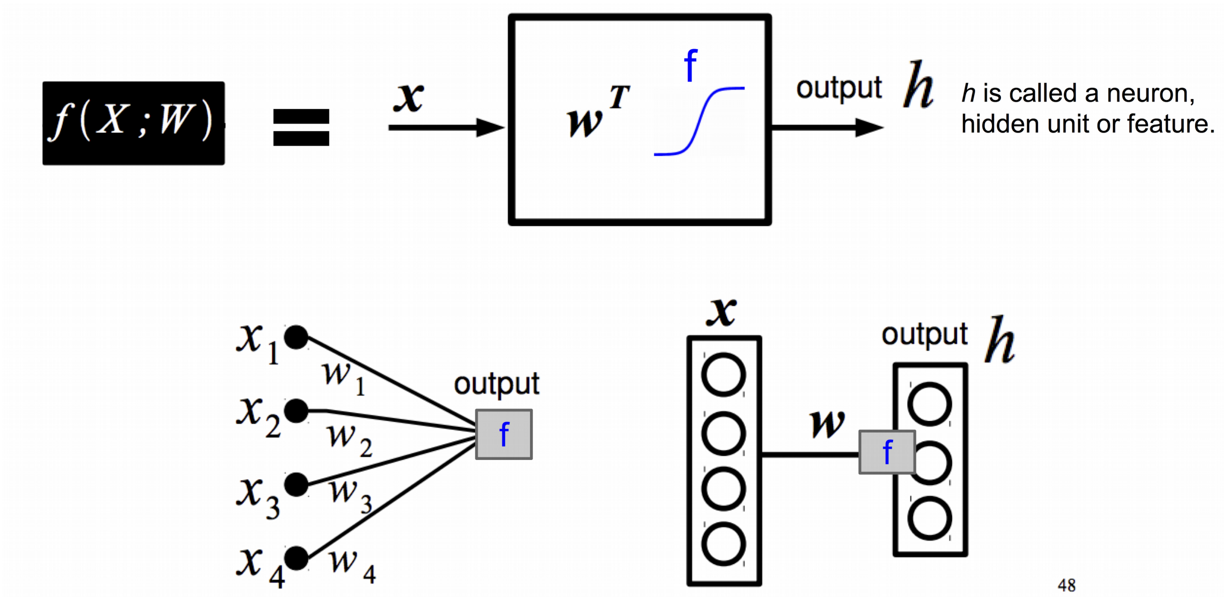
Feed-Forward Neural Network

- Several layers of **neurons**
- **No recurrent** connections
- Hidden layers needed for flexibility (universal approximation)



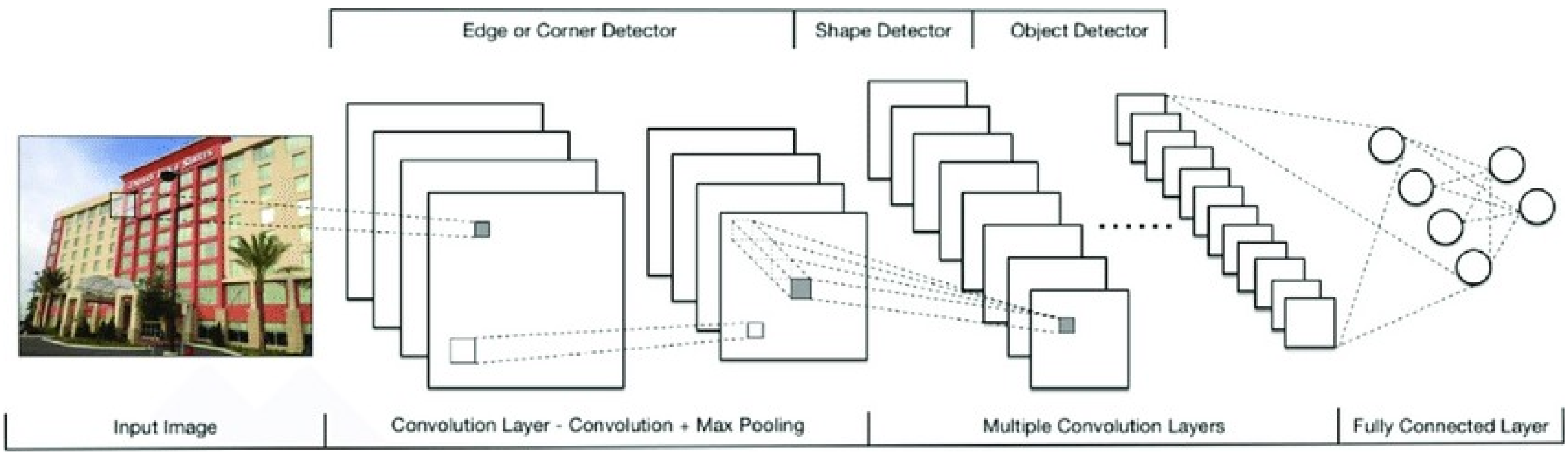
A Neural Network Layer

- Each **neuron** multiplies it's inputs with a weight vector
- Then the non-linearity (sigmoid or ReLu) is applied



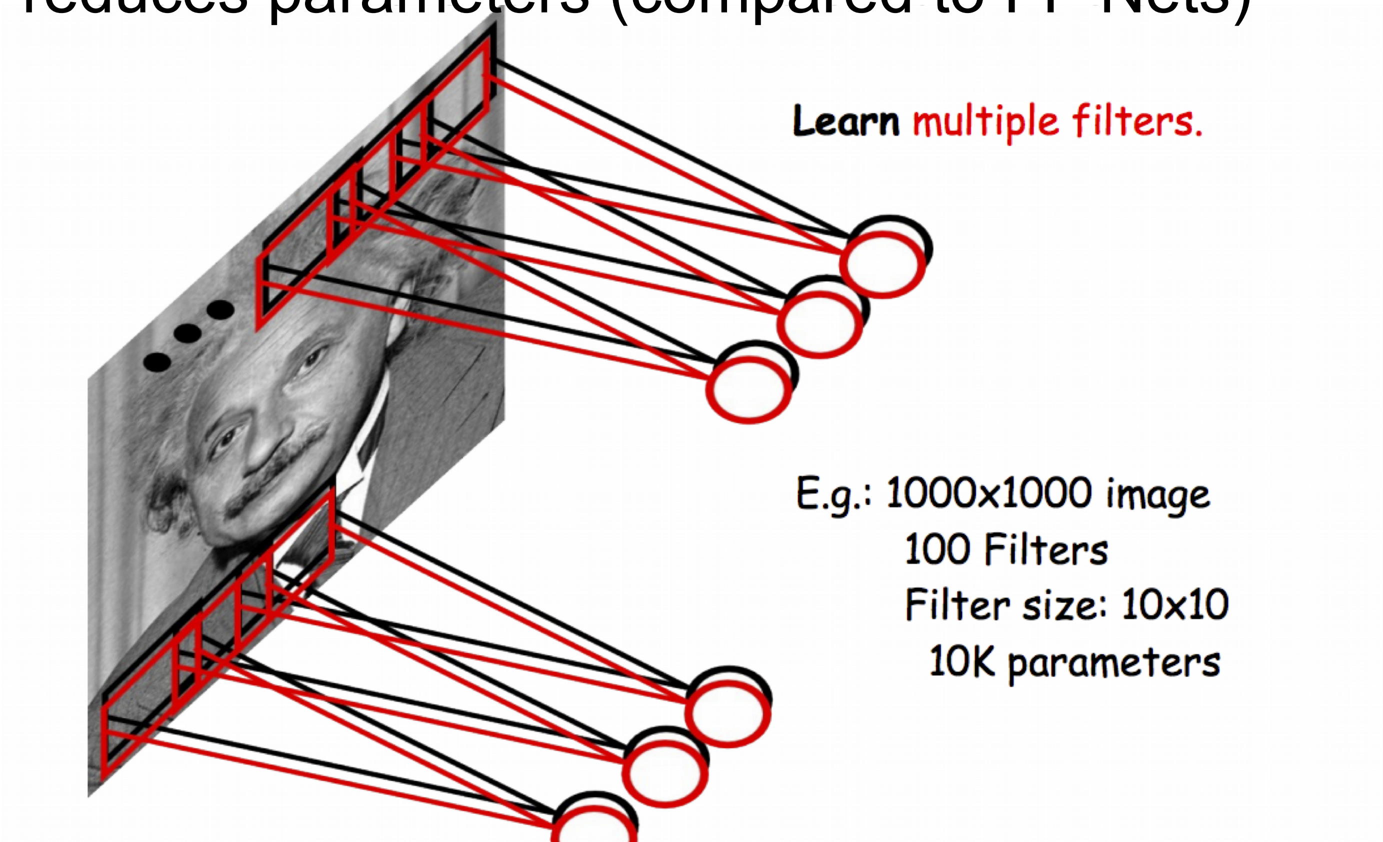
Convolutional NNs

- Feed-froward networks with hidden layers convolved (correlated) over input/previous layers
- Trained with backprop (gradient descent)
- Can learn to detect features in signals



ConvNets Efficiency

- Reusing the same weight vector across the input reduces parameters (compared to FF Nets)

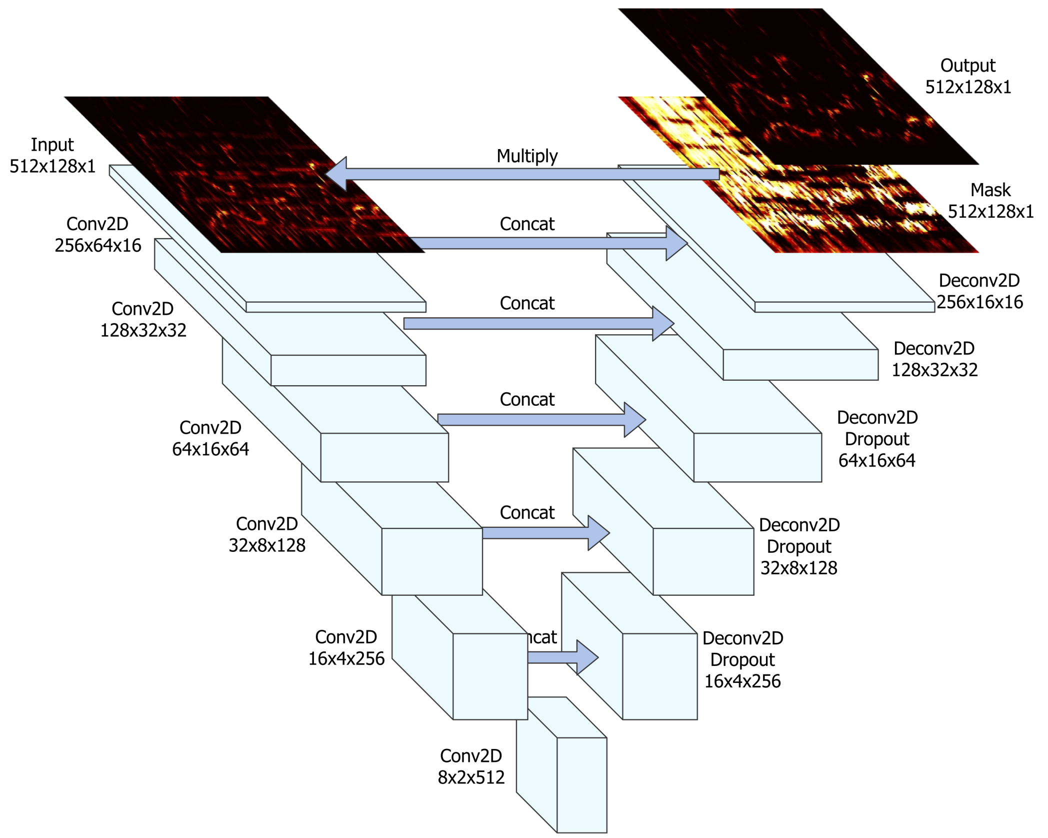


NNs for Audio

- Idea:
  - represent audio as 2-D image → spectrogram
  - use convnets like for images
- Applications:
  - Denoising, source separation

U-Nets for Voice Extraction

- Use a deep U-shaped Network for separating voice from accompaniment
- Jansson et al: Singing Voice Separation … ISMIR 2017



WaveNet

- Apply ConvNet in
  1-D to audio analysis and synthesis
- van den Oord et al: WaveNet, arXiv 2016