HOGESCHOOL ROTTERDAM

# Training Digital Signal Processing

ELETDS02

Filters

brojz@hr.nl

muiko@hr.nl

Original author: Johan Peltenburg

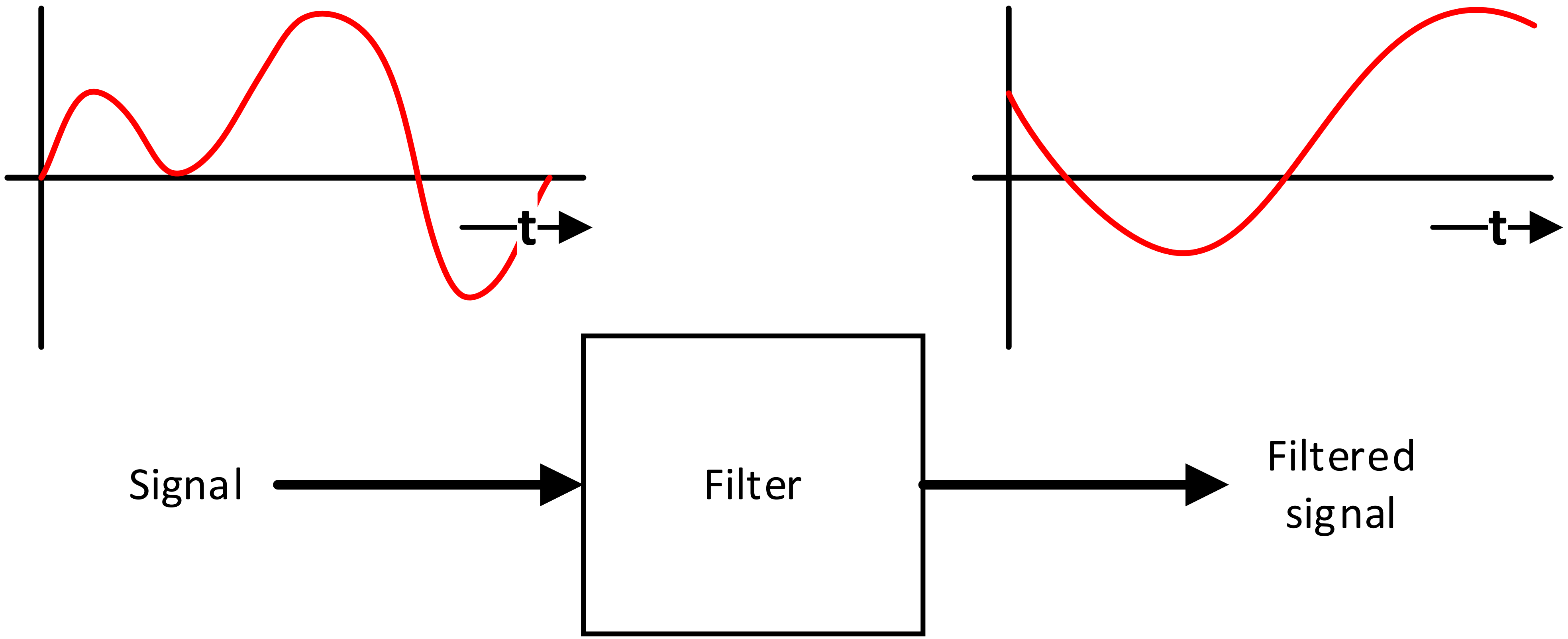
# Last week

- **Signals** in real life are **continuous** and **analog**.
- Need to sample and quantify them to be able to process them digitally.
- They become **discrete-time digital signals**.
- Signals can be represented as the sum of sines/cosines with certain **frequencies** and **amplitudes**.
- Many problems are specified or solved in the **Fourier frequency domain**.
- We can switch between time and frequency domain with the **Fourier Transform**.

# FILTERS

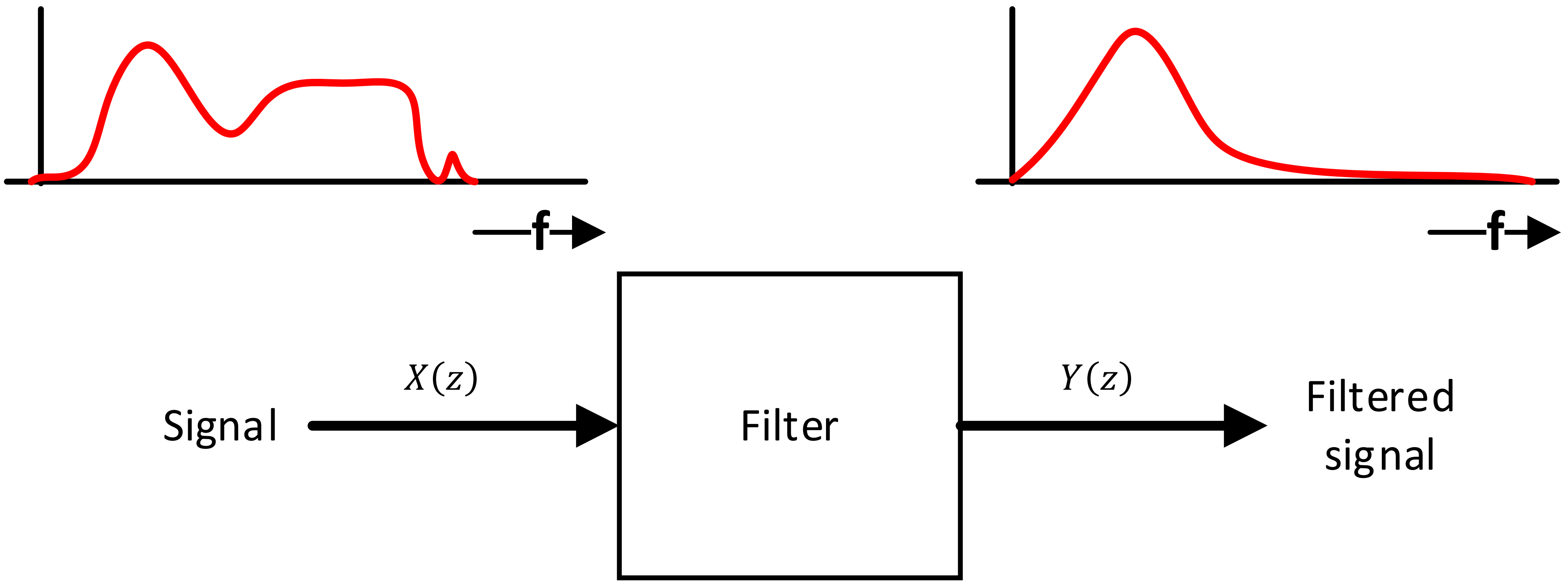
# Filters (1)

- Typical filters remove certain frequencies from a signal.



# Filters (2)

- Typical filters remove certain frequencies from a signal.



# Transfer function (1)

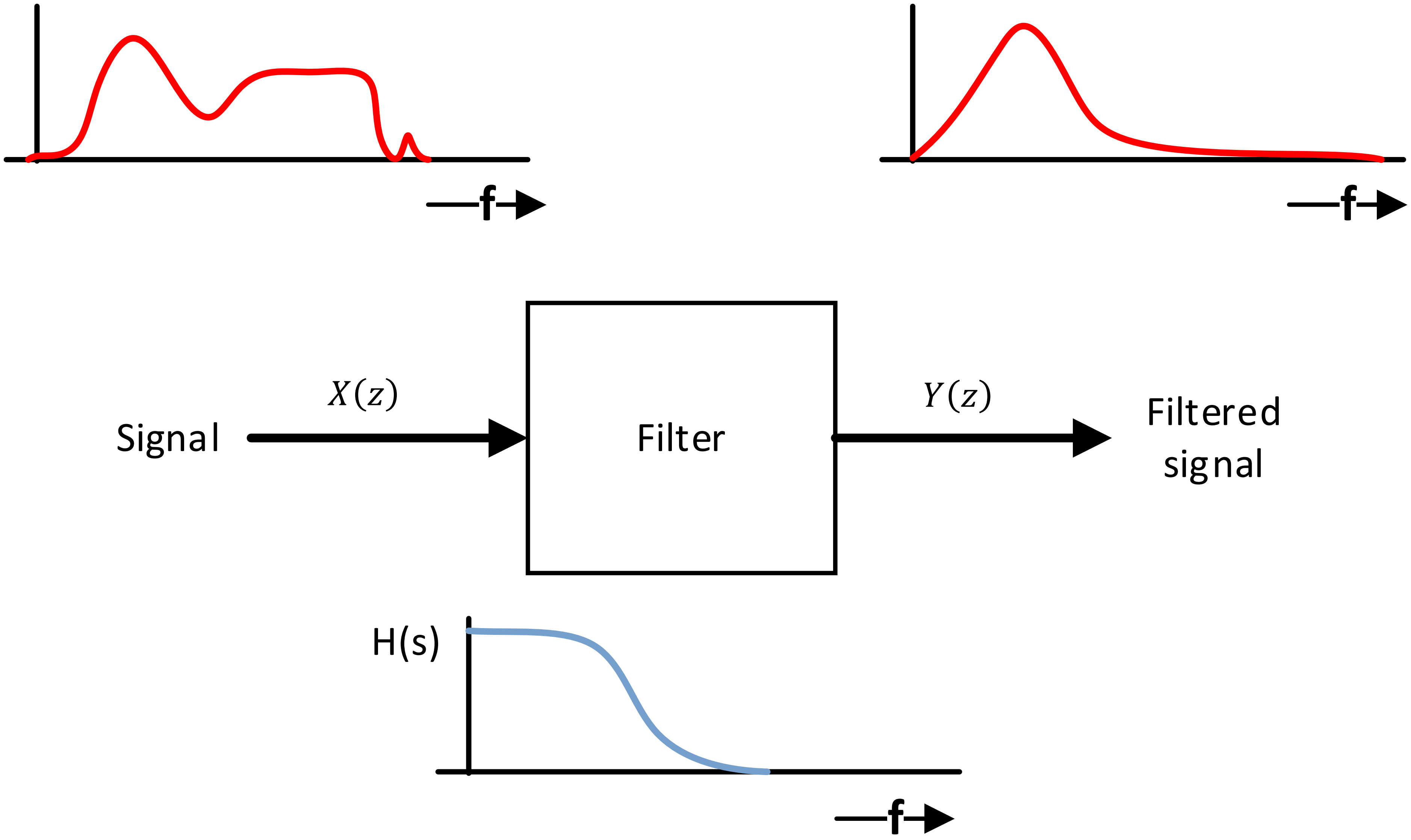
- Filters are often specified in the frequency domain.
- Filters can be described as a **transfer function**.
- The transfer function describes the relation between the input and output of the filter.
- Depending on what type of input signals (continuous or discrete) we have, it is denoted:

$$H(s) = \frac{Y(s)}{X(s)} \qquad H(z) = \frac{Y(z)}{X(z)}$$

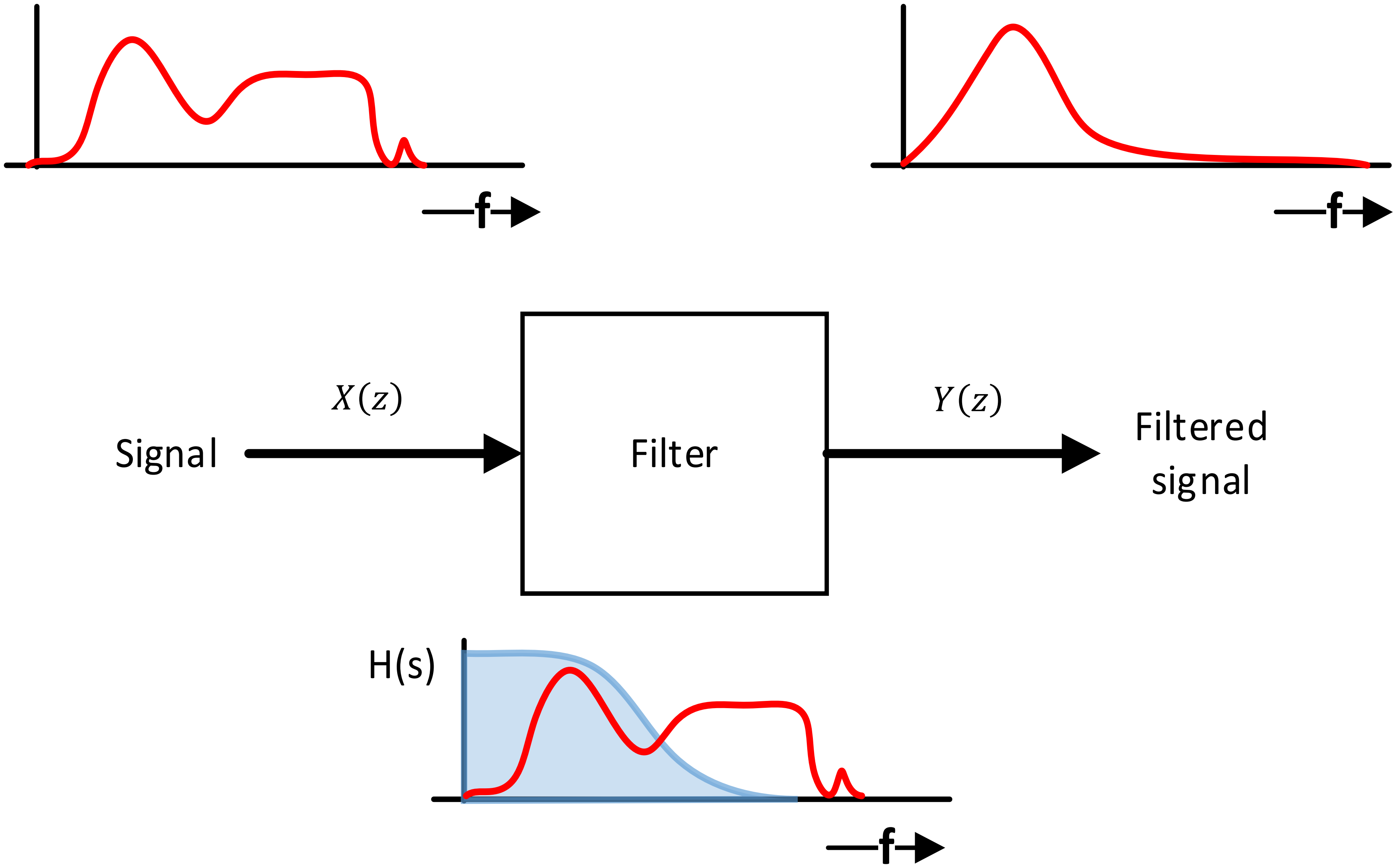
$$s = j2\pi f \qquad z = e^{j2\pi f}$$

- We can visualize this function.

# Transfer function (2)



# Transfer function (3)



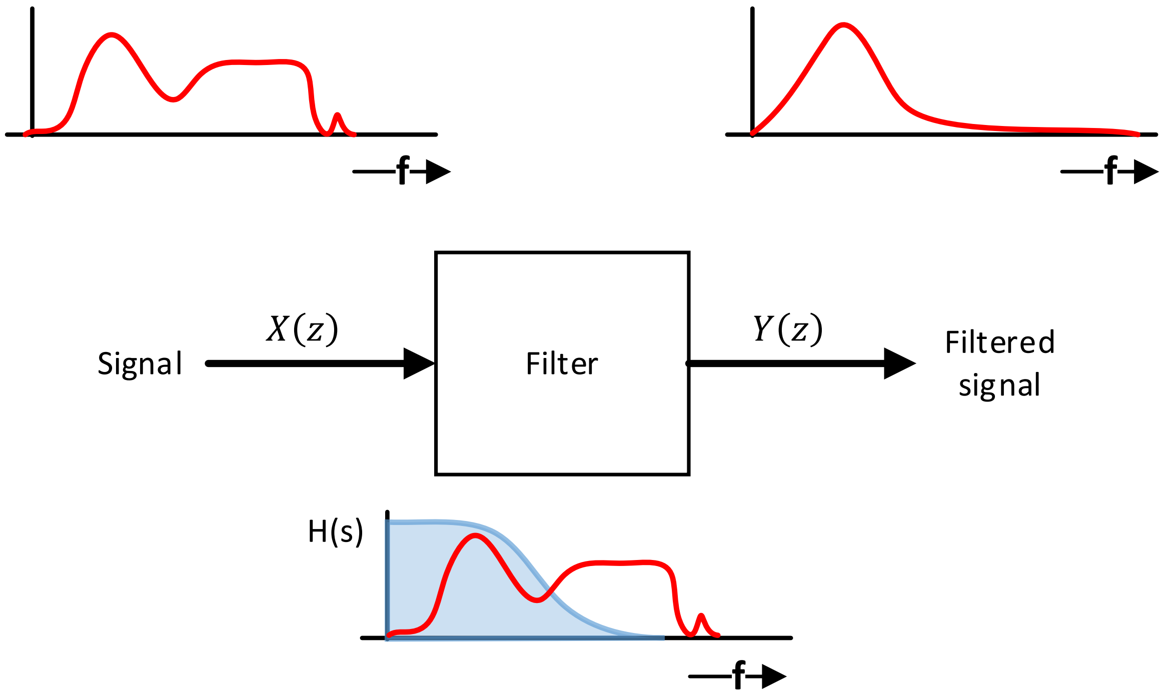
# Filters (3)

Filtering a signal is the same as **multiplying the spectrum of the signal with the transfer function** in the frequency domain.

Obvious when you look at the formula:

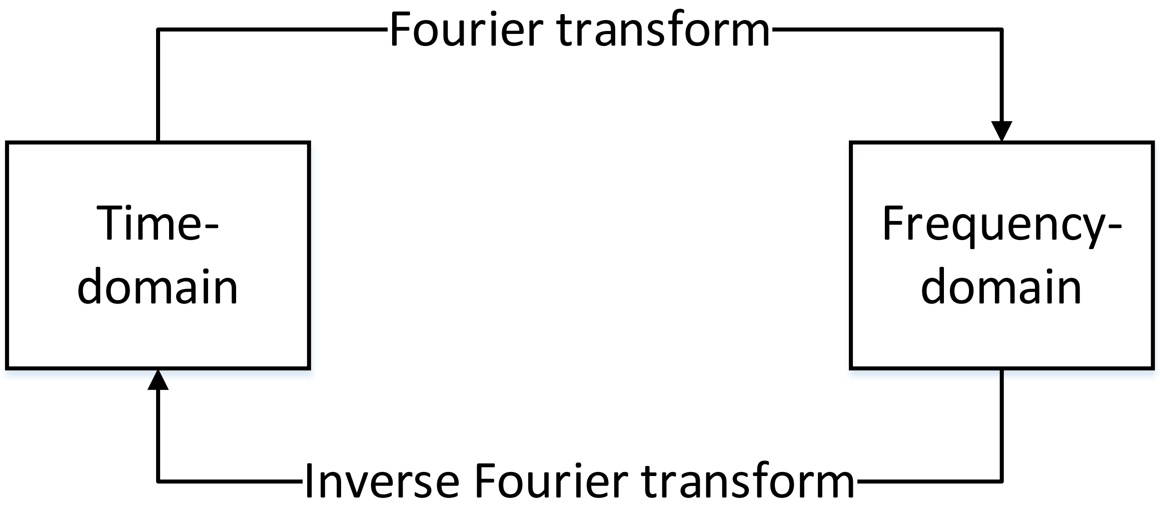
$$H(z) = \frac{Y(z)}{X(z)}$$

$$Y(z) = H(z) \cdot X(z)$$



# Time-domain filter variant

- To design a filter, we specify the transfer function in the frequency domain.
- We use the IDTFT (Inverse Discrete Time Fourier Transform) to go back to the discrete-time domain.




- Technical note: this only works for non-recursive filters.

# Convolution

- Multiplication in the frequency domain is the same as **convolution** in the time-domain.
- Convolution for 1 output sample can be seen as the inner product of two vectors:

**Inner product operator**

$3 \times 1$

$1 \times 3$

$$\vec{a} \cdot \vec{b} = \vec{a}^T \vec{b} = [1 \quad 2 \quad 3] \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}$$

$1 \times 1$

$$= 1 \cdot 4 + 2 \cdot 5 + 3 \cdot 6 = 32$$

**Scalar**

# FIR Filter

- Formula of a filter in the discrete-time domain:

Convolution.

Vector with coefficients $b_0$ to $b_N$.

$$y[k] = \sum_{n=0}^{N} b_n \cdot x[k-n] = \vec{b} \cdot \vec{x}[k]$$

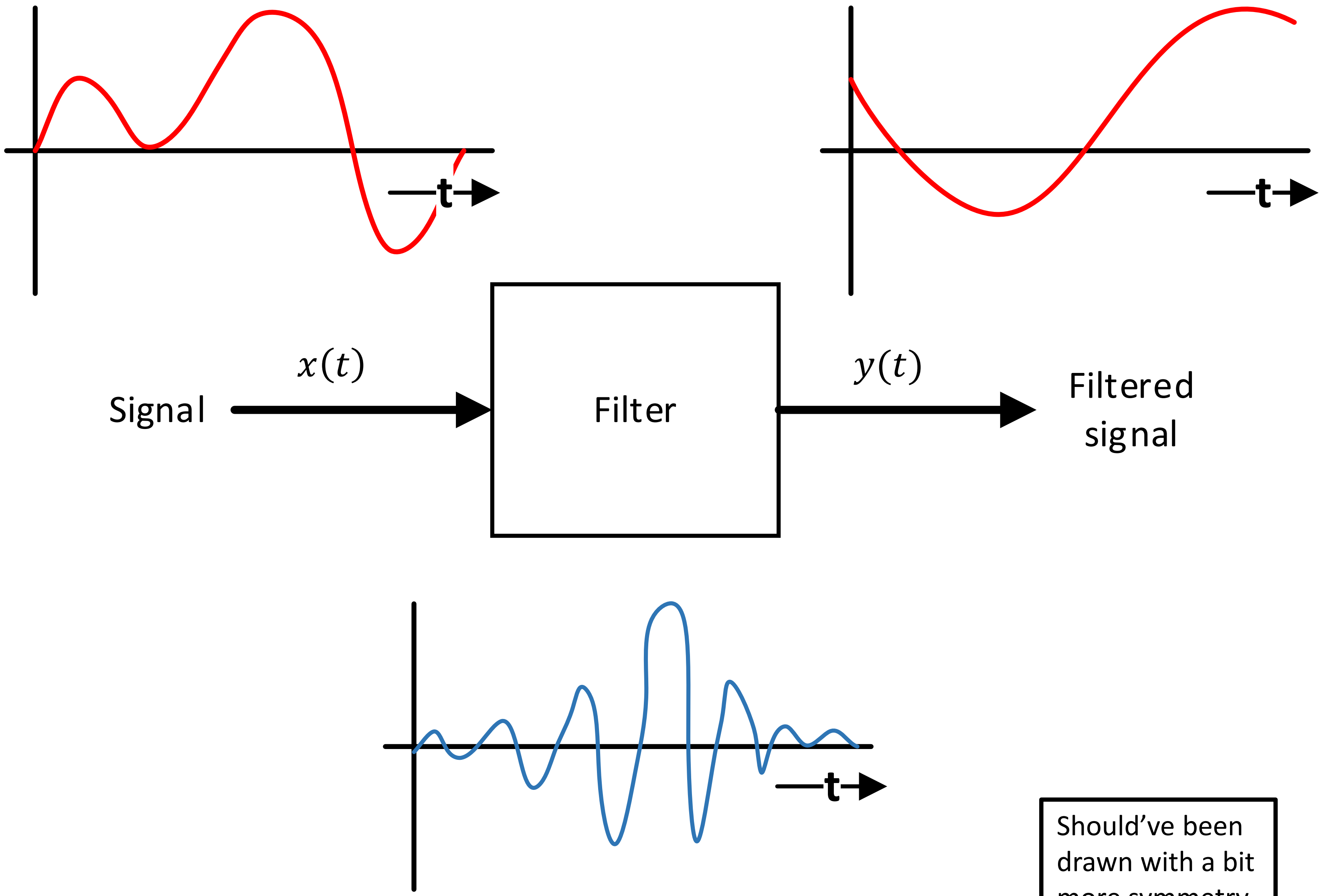
Vector with samples $x[k-N]$ to $x[k]$.

- We call this a **Finite Impulse Response Filter (FIR Filter)**
- $b_n$ are the filter coefficients calculated by the IDTFT.
- This is because if we let $x[k]$ be an impulse, the filter has a finite response due to a limited number of coefficients N (goes to zero after a while).
- FIR filters are of type LTI (linear time-invariant)



HOGESCHOOL ROTTERDAM

# Filters (4)

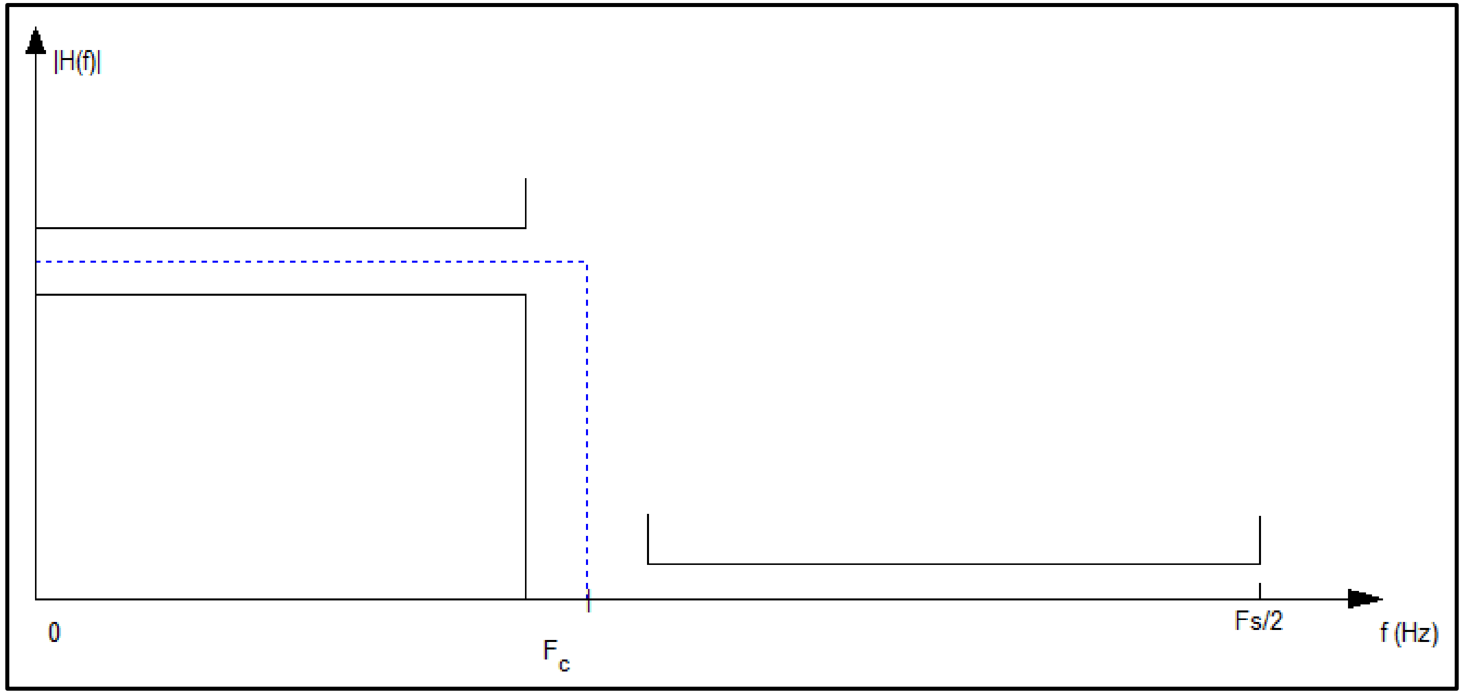



Should've been drawn with a bit more symmetry

# Common transfer functions of filters

- Low-pass
- High-pass
- Band-pass
- Band-stop
  - Notch filter

# Filter characteristics (1)




An ideal low-pass filter.

- Cut-off frequency: $F_c$
- Point where only half of the power is left in the signal at that frequency.
- Also called "-3 dB point".

- Filter magnitude is often specified in dB.

$$A_{dB} = 10 \cdot log_{10}( A)$$

$$A = 10^{\frac{A_{dB}}{10}}$$

$$A_{dB} = 10 \cdot \log_{10}(2) \approx 3.01$$

- Time-discrete filters can only handle frequencies up to half the sample rate: $\frac{F_s}{2}$
- This is due to the Shannon-Nyquist sampling theorem:
- To reconstruct a signal containing frequencies up to $B$ Hz we need to sample the signal with more than $2B$ Hz.

# Some filter terminology

- Pass band: Frequency range in which the filter lets signals pass.
- Stop band: Frequency range in which the filter suppresses signals.
- Roll-off: Steepness of the filter between the pass band and the stop band.
- Bode-plot: A graphical representation of the transfer function, including phase characteristics.

# Summary

- Filters remove certain frequencies from a signal.
- Filters have **a transfer function** often specified in the **frequency domain**.
- We can implement the filter in the discrete-time domain by using the **IDTFT**.
- (Discrete-time) filters have several characteristic such as response shape (LP, HP, BP, BS ), cut-off frequency and others.