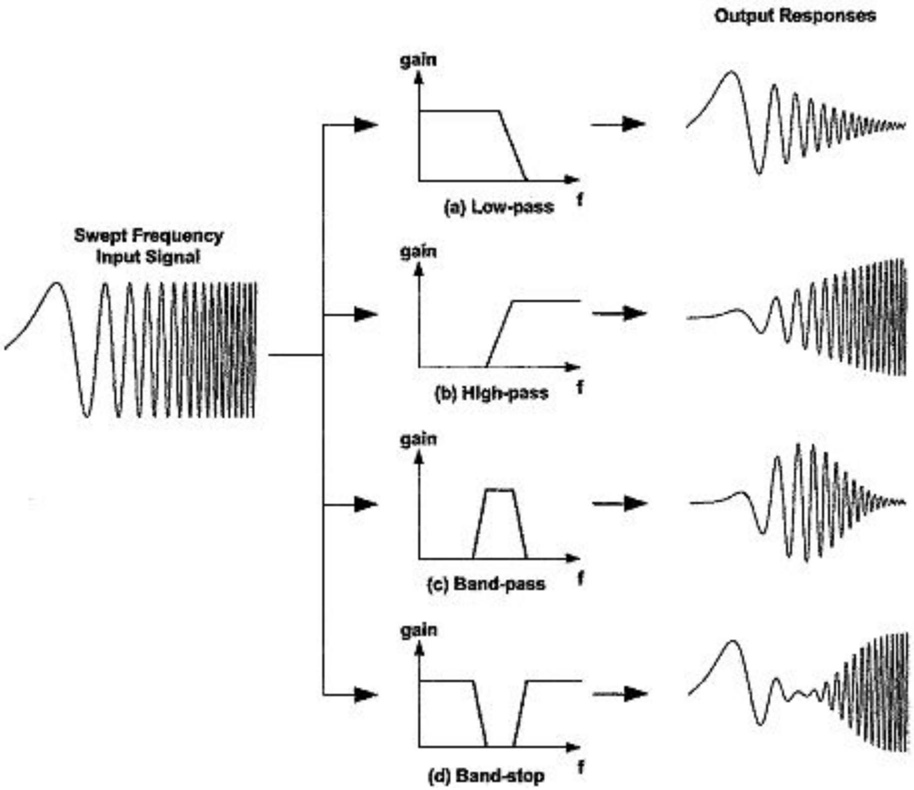
Filters

Analog Filters

- Often desired to “shape” signal with filters
- Characterized by their steady-state transfer function

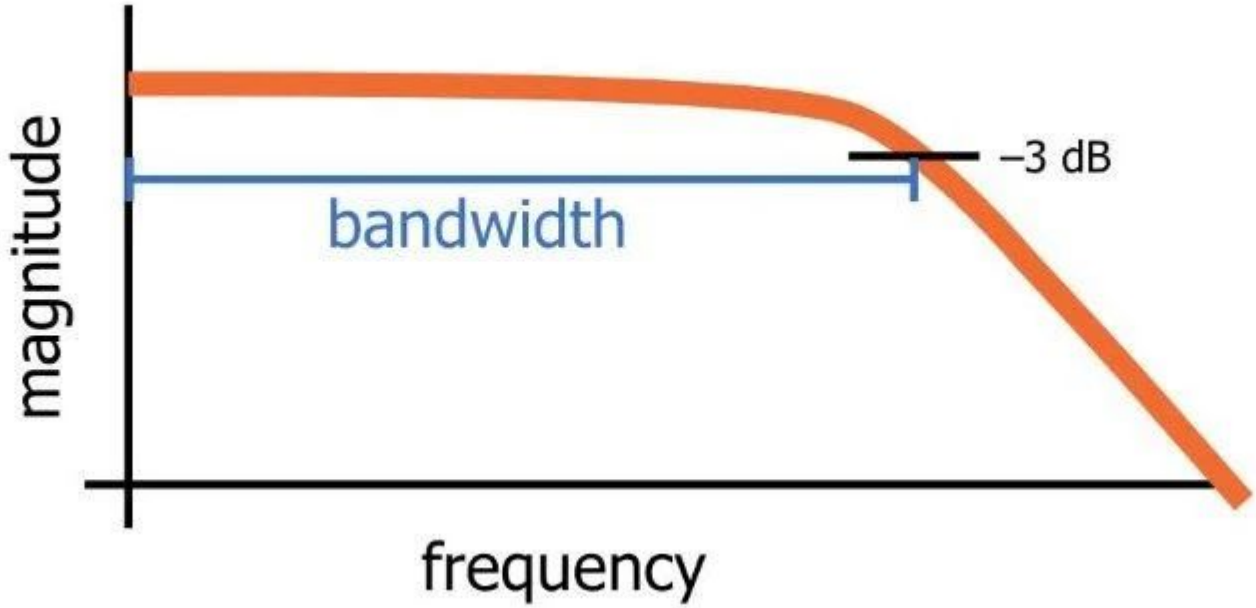
$$H(\omega) = \frac{V_{output}(\omega)}{V_{input}(\omega)}$$

- Frequency dependent



https://www.allaboutcircuits.com/technical-articles/low-pass-filter-tutorial-basics-passive-RC-filter/

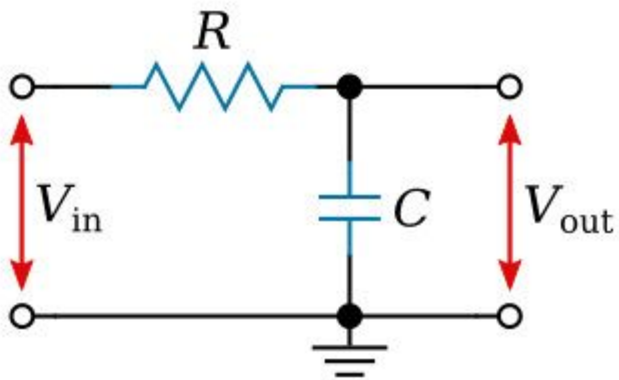
-3 dB



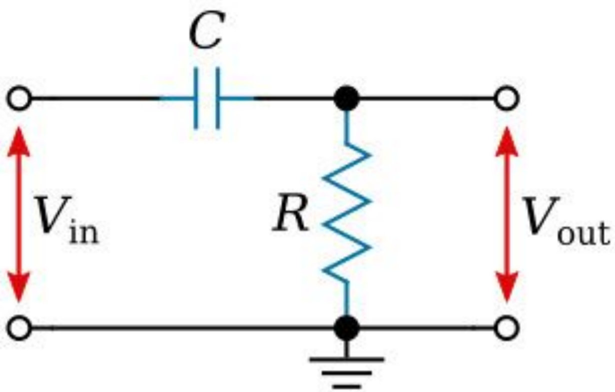
- “Cutoff frequency” usually refers to the –3 dB frequency
- The power decreases by 50% at this point
- Since $P \propto V^2$, voltage decreases by $1/\sqrt{2}$

RC Filters

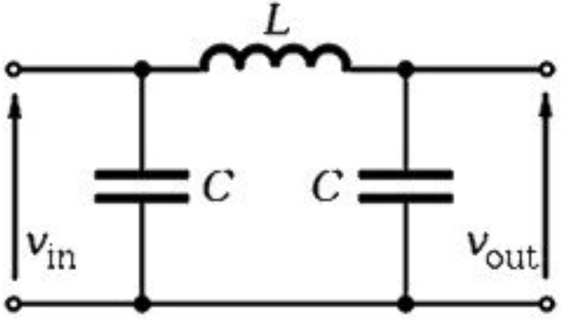
RC Low Pass Filter



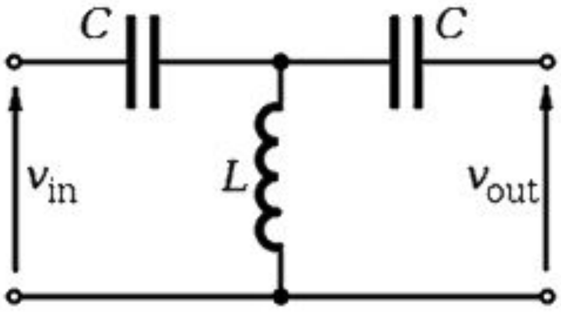
RC High Pass Filter



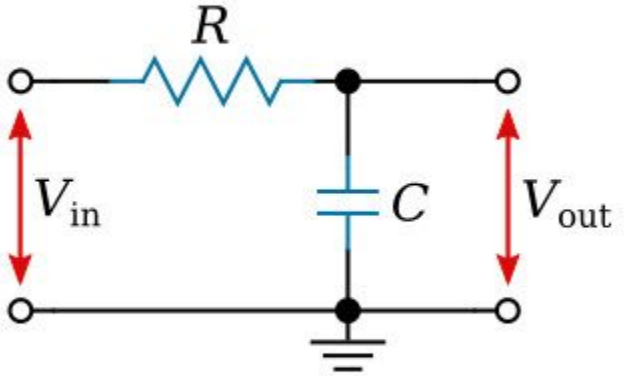
Pi Filter



T Filter



RC Filter -- Transfer function & cutoff frequency



- This circuit is a voltage divider

$$V_{\text{out}} = \frac{Z_C}{Z_R + Z_C} V_{\text{in}}$$

- With $Z_C = \frac{1}{j\omega C}$ and $Z_R = R$

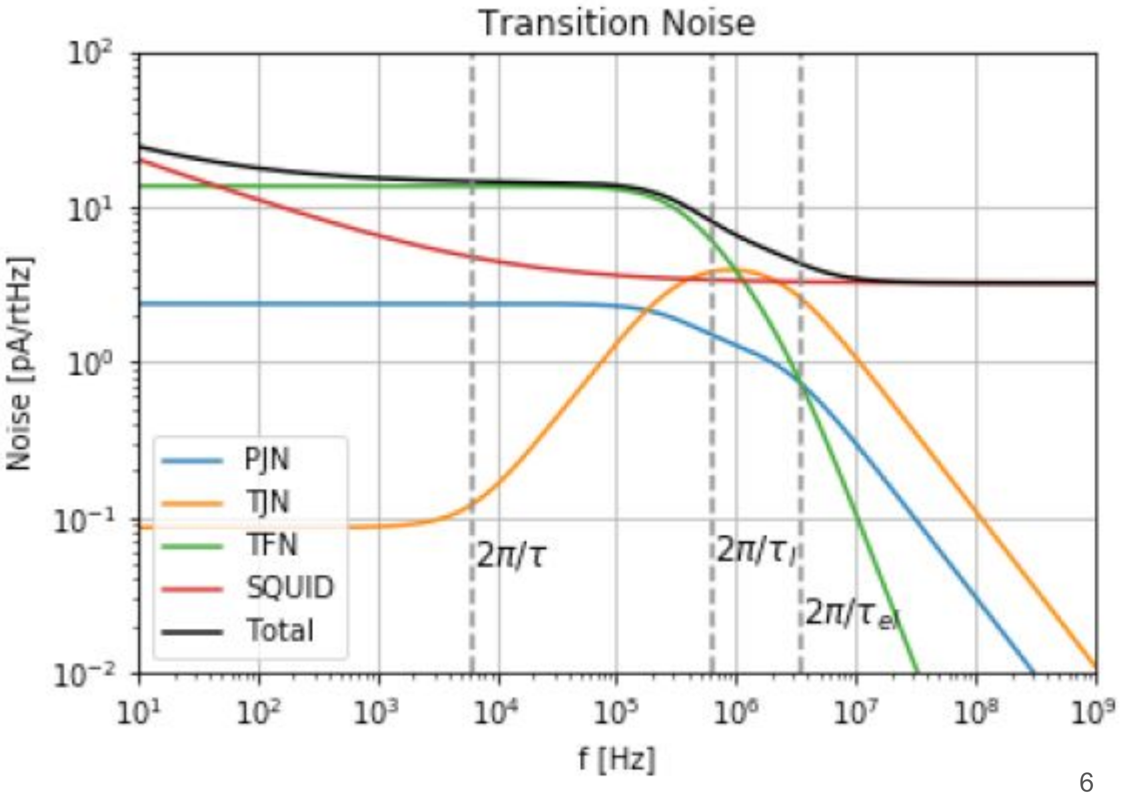
$$V_{\text{out}} = \frac{1/j\omega C}{R + 1/j\omega C} V_{\text{in}}$$

- Transfer function: $H(\omega) = \frac{V_{\text{out}}(\omega)}{V_{\text{in}}(\omega)} = \frac{1}{1 + j\omega RC}$

- Cutoff radial frequency: $|H(\omega)| = 1/\sqrt{2} \rightarrow \omega = \frac{1}{RC}$
- Cutoff frequency: $f = \frac{1}{2\pi RC}$

Why filtering?

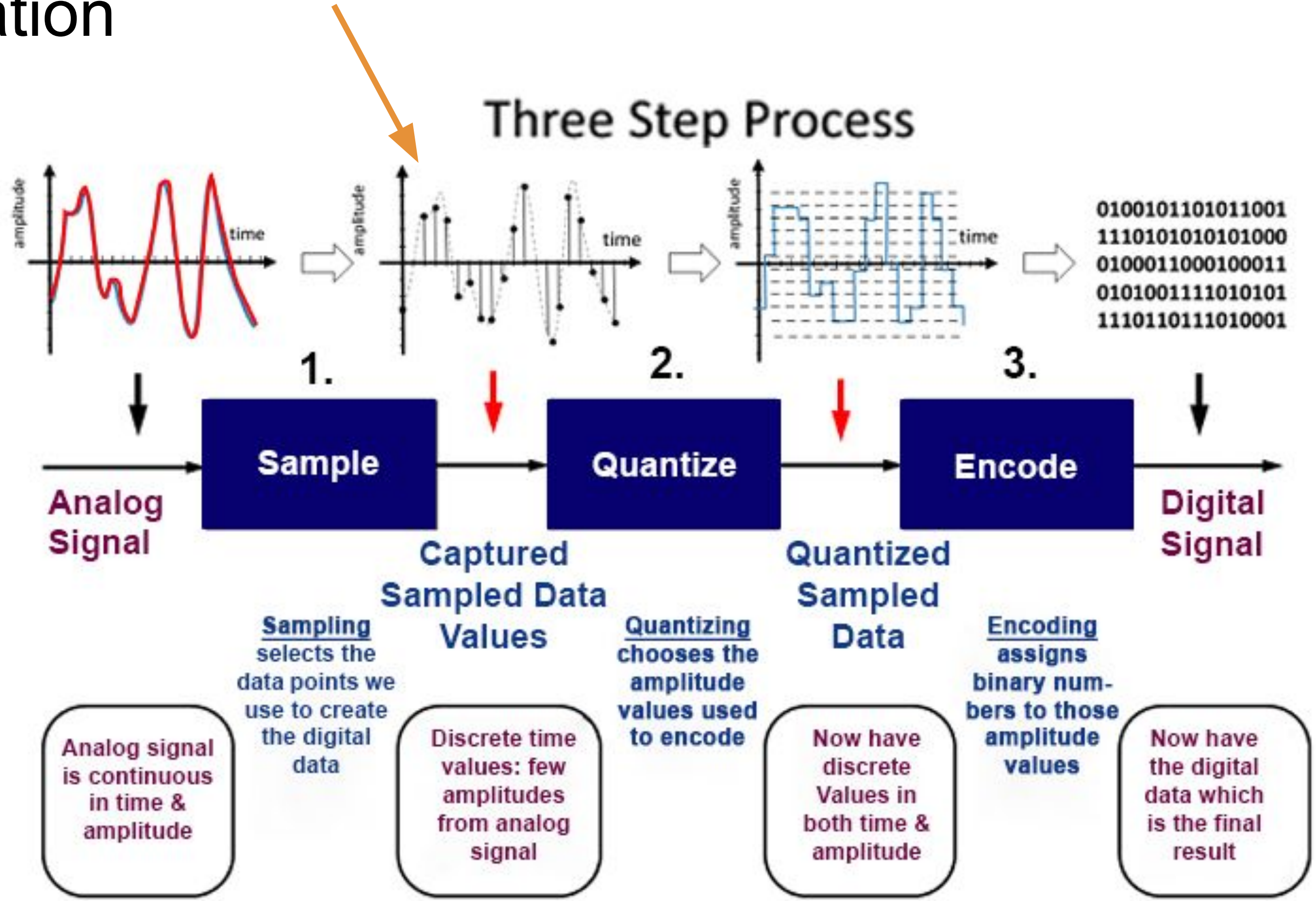
- Often beneficial to keep data in high signal/noise region, and reject low signal/noise region
- Or blocking a stream of noise polluting the signal
- And "Antialiasing"



Digitization

https://www.geeksforgeeks.org/sampling-in-digital-communication/

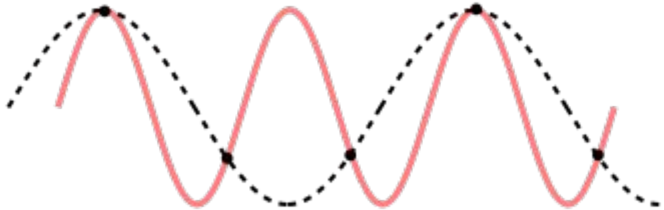
Clock



https://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem

Nyquist–Shannon sampling theorem

- Sampling frequency (f) used when digitizing an analog signal
- Max frequency component can be resolved is f/2
    - → Nyquist frequency
- Frequency components above f/2 are “aliased” below f/2
    - Eg. if f=1000 Hz, f/2=500 Hz
        - 510 Hz sine wave looks identical to 490 Hz sine wave

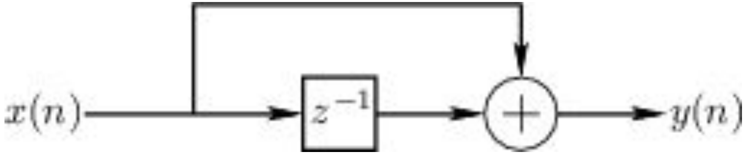


- **“Anti-aliasing” is mandatory before digitization**
    - Can be achieved with a low-pass filter

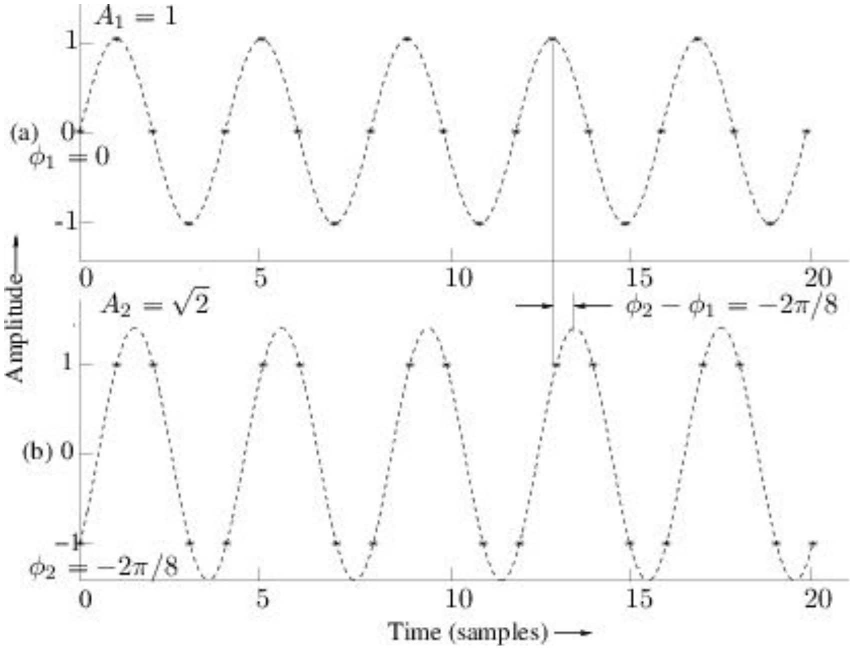
https://www.dsprelated.com/freebooks/filters/Simplest_Lowpass_Filter.html

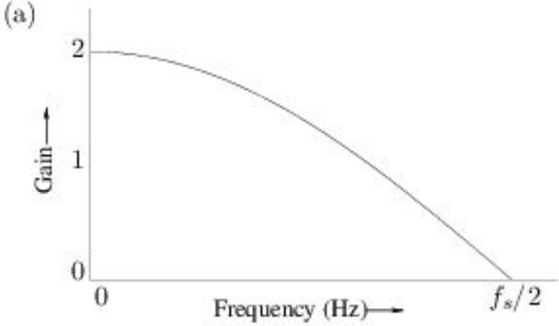
Simplest example of digital filter

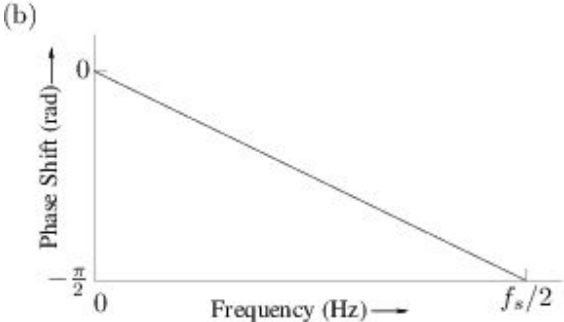
- $y(n) = x(n) + x(n-1)$
- z^{-1} means “delay a sample”



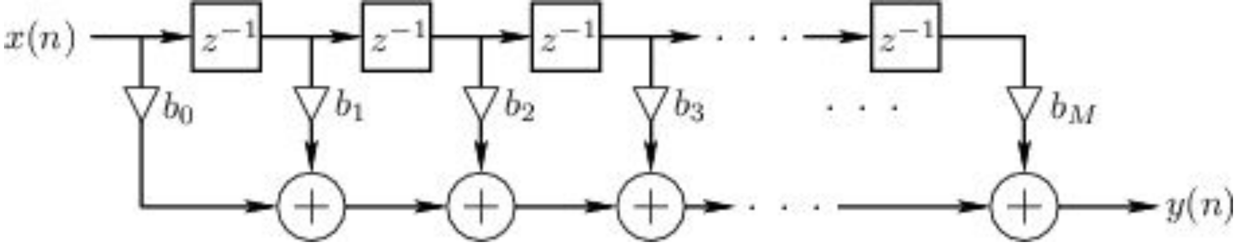
- Rolling average
- Can analyze its response
- Low pass filter of some sort
- With a phase delay
- Can be implemented as Finite Impulse Response (FIR) filter



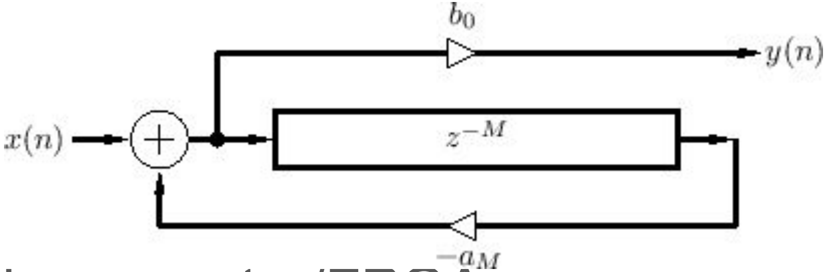




FIR



- Output depends on input only
- VS. Infinite Impulse Response (IIR) filter, output depends on output recursively.



- FIR can be applied in computer/FPGA
- Property of FIR defined by its filter “kernel”
- Implemented with **correlation** or **convolution**
  - Kernels are “flipped” between correlation and convolution implementations

Correlation vs convolution

Correlation

$$(f \star g)(t) \overset{\mathrm{def}}{=} \int_{-\infty}^{\infty} f^*(\tau) g(t+\tau)\, d\tau$$

Convolution

$$f(x) * g(x) = \int_{-\infty}^{\infty} f(\tau) g(x-\tau)\, d\tau$$

Convolution properties

$$f * g = g * f$$

$$f * (g * h) = (f * g) * h$$

$$f * \delta = f$$

$$(f * g)' = f' * g = f * g'$$

$$\overline{f * g} = \overline{f} * \overline{g}$$

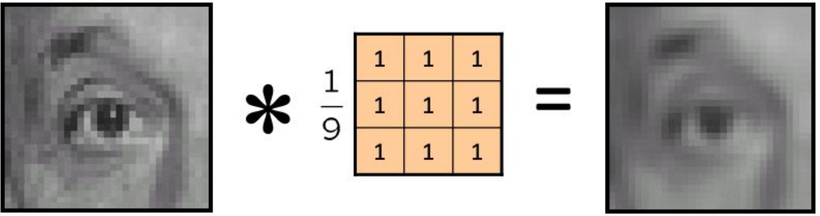
$$(f * g)' = f' * g = f * g'$$

$$\int_{\mathbf{R}^d} (f * g)(x)\, dx = \left(\int_{\mathbf{R}^d} f(x)\, dx\right) \left(\int_{\mathbf{R}^d} g(x)\, dx\right).$$

$$\mathcal{F}\{f * g\} = \mathcal{F}\{f\} \cdot \mathcal{F}\{g\}$$

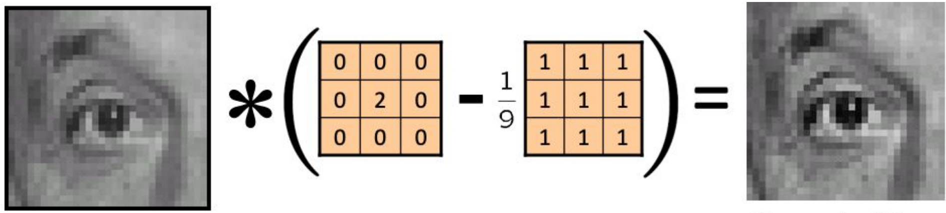
My favourite filter lecture → https://www.cs.cornell.edu/courses/cs6670/2011sp/lectures/lec02_filter.pdf

Digital filter example: image processing



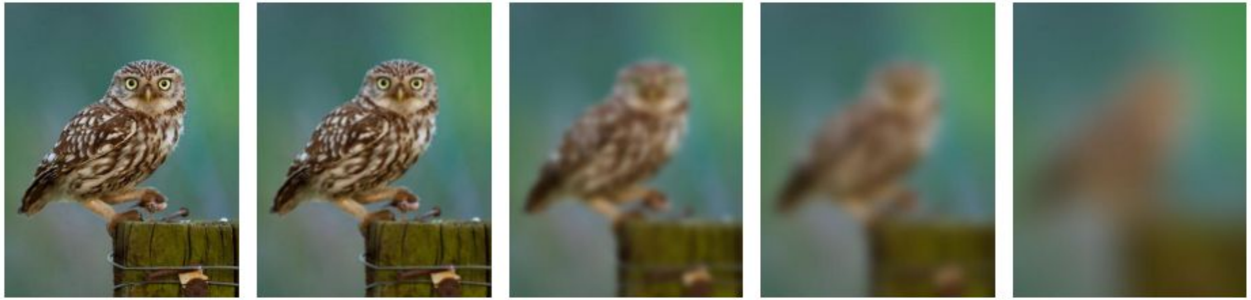
Original

Blur (with a mean filter)



Original

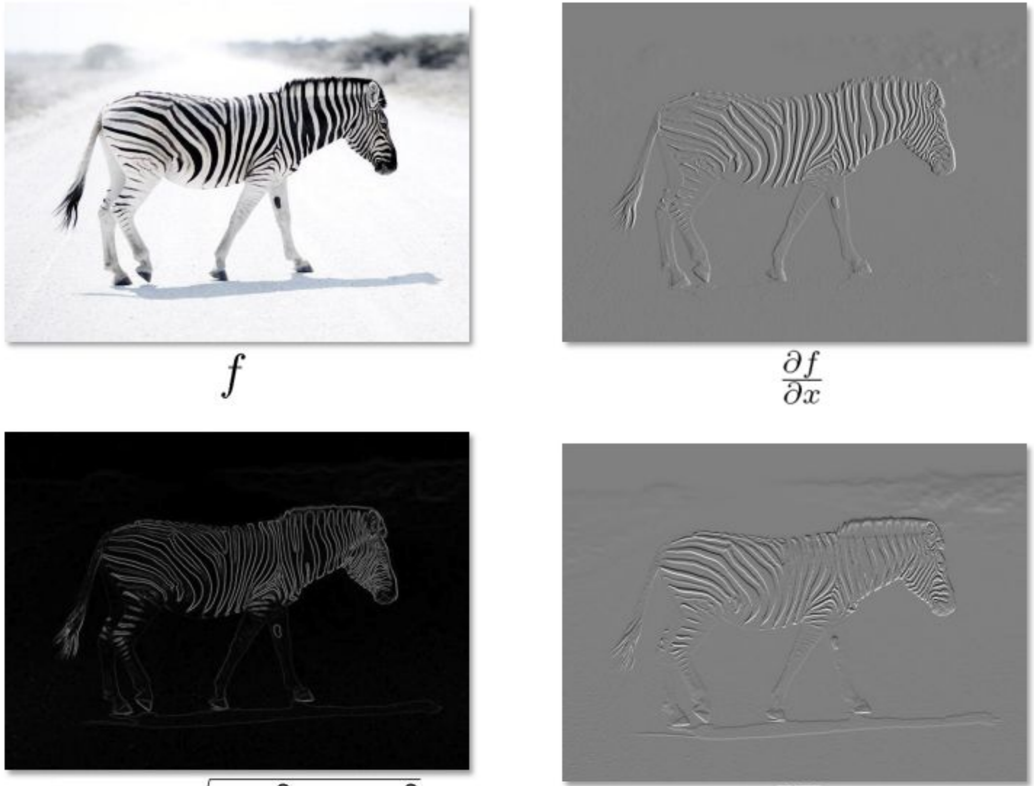
Sharpening filter
(accentuates edges)



Gaussian Filter

σ = 1 pixel σ = 5 pixels σ = 10 pixels σ = 30 pixels

Pulse triggering == Edge finding



f

$\frac{\partial f}{\partial x}$

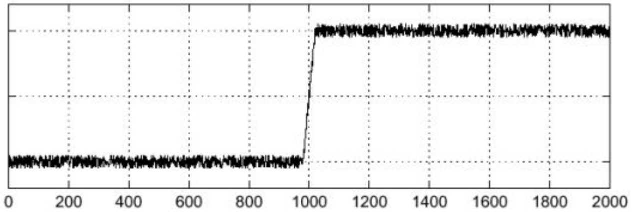
$\|\nabla f\| = \sqrt{(\frac{\partial f}{\partial x})^2 + (\frac{\partial f}{\partial y})^2}$

$\frac{\partial f}{\partial y}$

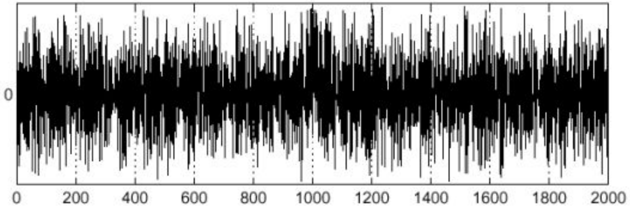
Noise?

A low pass filter

$f(x)$



$\frac{d}{dx}f(x)$



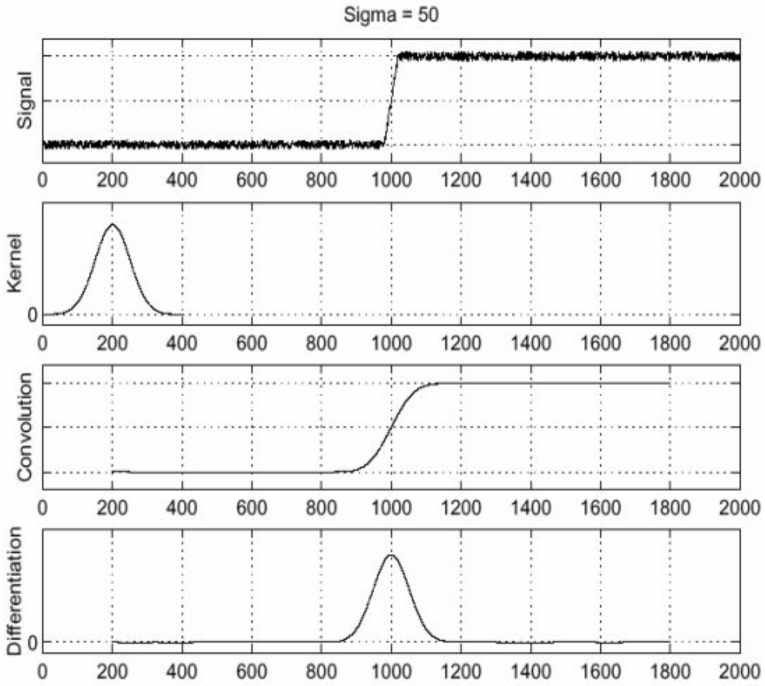
f

h

$f * h$

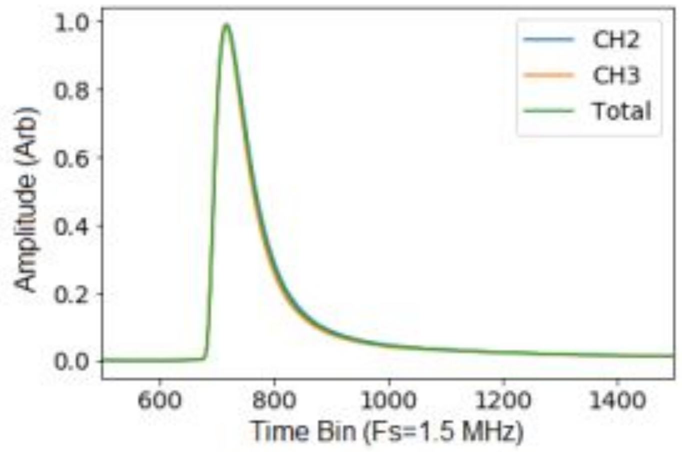
$$\frac{d}{dx}(f * h)$$

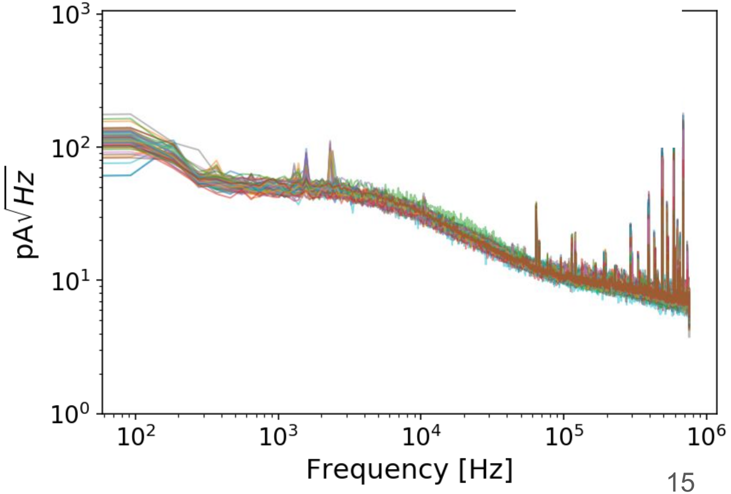
$$= f * \frac{d}{dx}h$$



Optimal Filter (Matched Filter)

- Assume a known pulse shape
- Assume a known noise spectrum
  - 60 Hz and harmonics from wall
  - 2 kHz from fluorescent light
  - 300 kHz from shaky power supply
- How to optimize filtering so to maintain the optimal signal-to-noise?
- Cannot do time domain pulse fit
  - Correlations from bin to bin





https://www.slac.stanford.edu/exp/cdms/ScienceResults/Theses/kurinsky.pdf

OF as frequency domain chi2 fit

For the chi-squared function

$$\chi^2 = \int_{-\infty}^{\infty} \frac{|v(f) - As(f)|^2}{J(f)} df$$

we find the goodness of fit by minimizing χ^2 with respect to A, as

$$0 = \frac{d\chi^2}{dA} = \frac{d}{dA} \int_{-\infty}^{\infty} \frac{v^*(f)v(f) - 2As^*(f)v(f) + A^2 s^*(f)s(f)}{J(f)} df$$

$$0 = 2\int_{-\infty}^{\infty} \frac{-s^*(f)v(f) + As^*(f)s(f)}{J(f)} df$$

$$\int_{-\infty}^{\infty} \frac{s^*(f)v(f)}{J(f)} df = A\int_{-\infty}^{\infty} \frac{s^*(f)s(f)}{J(f)} df$$

$$A = \frac{\int_{-\infty}^{\infty} \frac{s^*(f)v(f)}{J(f)} df}{\int_{-\infty}^{\infty} \frac{|s(f)|^2}{J(f)} df}$$

This suggests that the optimum filter for this signal has the form

$$\phi(f) = \frac{s^*(f)}{J(f)}$$

so that we can write the optimal estimate as

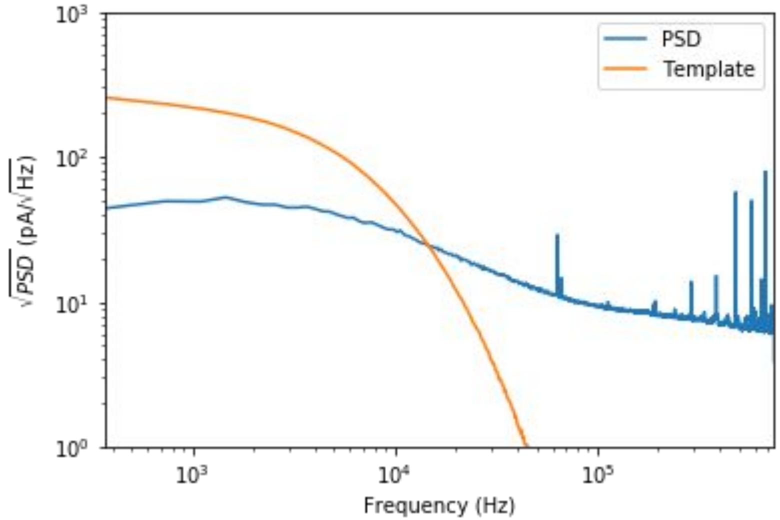
$$A = \frac{\int_{-\infty}^{\infty} \phi(f)v(f)df}{\int_{-\infty}^{\infty} \phi(f)s(f)df}$$

or simplifying further, we can renormalize the filter as

$$\phi'(f) = \frac{\frac{s^*(f)}{J(f)}}{\int_{-\infty}^{\infty} \phi(f)s(f)df}$$

to give the resulting simple estimator

$$A = \int_{-\infty}^{\infty} \phi'(f)v(f)df$$



$$\phi(f) = \frac{s^*(f)}{J(f)}$$

Energy and timing resolution

- The energy resolution can be evaluated by the standard deviation of amplitude on noise traces.

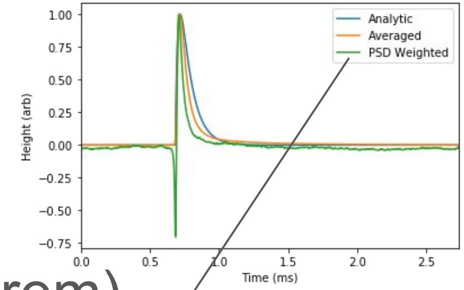
$$\sigma_A^2 = \overline{\langle A_{n_{filt}} \rangle^2} = \frac{\int_{-\infty}^{\infty} \overline{\left(\frac{S^*(f)n(f)}{J(f)}\right)^2} df}{\left(\int_{-\infty}^{\infty} \frac{S^*(f')S(f')}{J(f')} df'\right)^2} = \frac{1}{\left(\int_{-\infty}^{\infty} \frac{S^*(f')S(f')}{J(f')} df'\right)^2}$$

NEP

- Timing resolution

$$\sigma_{t_0}^2 = \overline{\langle t_0 \rangle^2} = \frac{1}{\left(\int_{-\infty}^{\infty} \frac{(a \cdot 2\pi f')^2 S^*(f')S(f')}{J(f')} df'\right)^2}$$

OF as a FIR filter



- Recall $\mathcal{F}\{f * g\} = \mathcal{F}\{f\} \cdot \mathcal{F}\{g\}$ (convolution theorem)

Sum over frequency bins

Sum over time bins

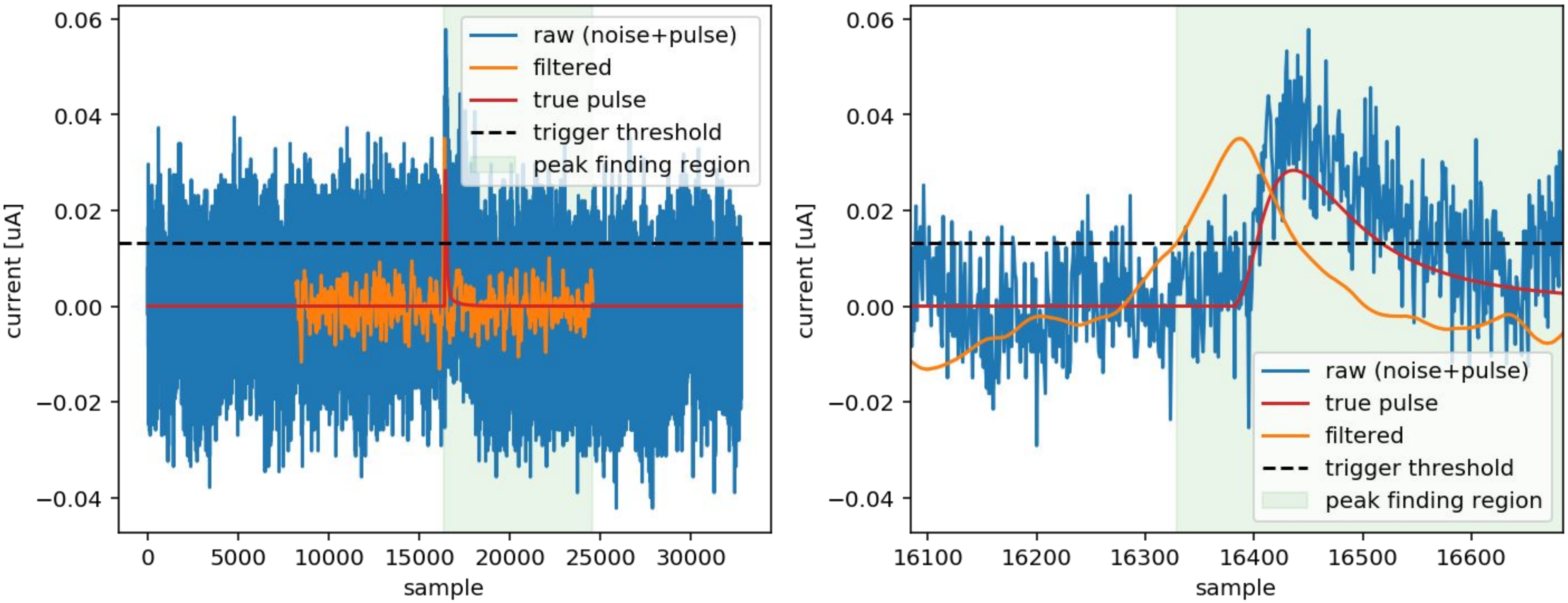
$$A = \sum_{m=0}^{N-1} \Phi(m) X(m) = \sum_{k=0}^{N-1} \phi(-k) x(k),$$

Frequency domain filter

Frequency domain data

FFT^{-1} of Φ

Time domain data


0.06
0.04
0.02
0.00
−0.02
−0.04
current [uA]
0
5000
10000
15000
20000
25000
30000
sample
raw (noise+pulse)
filtered
true pulse
trigger threshold
peak finding region
16100
16200
16300
16400
16500
16600
raw (noise+pulse)
true pulse
filtered
trigger threshold
peak finding region