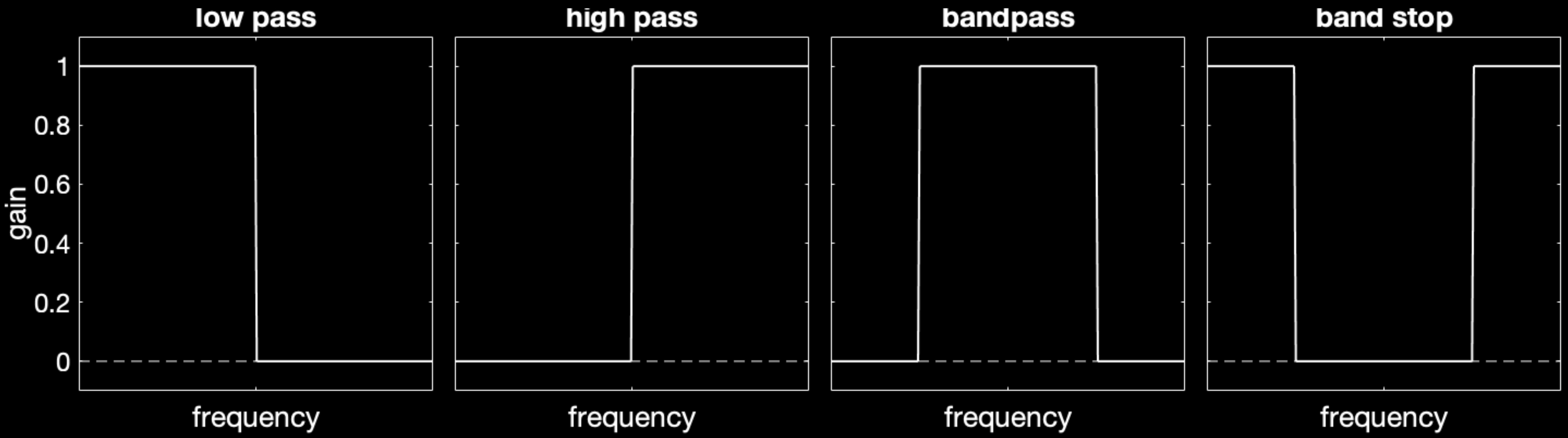
Filters

- Any time-varying voltage can be expressed as a superposition of sinusoids of different frequencies
- A *filter* removes undesired frequencies
  - Noise (e.g. 60 Hz, high-frequency hiss)
  - Irrelevant frequencies (e.g. < 20 Hz or > 20 kHz for audio) to prevent aliasing
- Types:



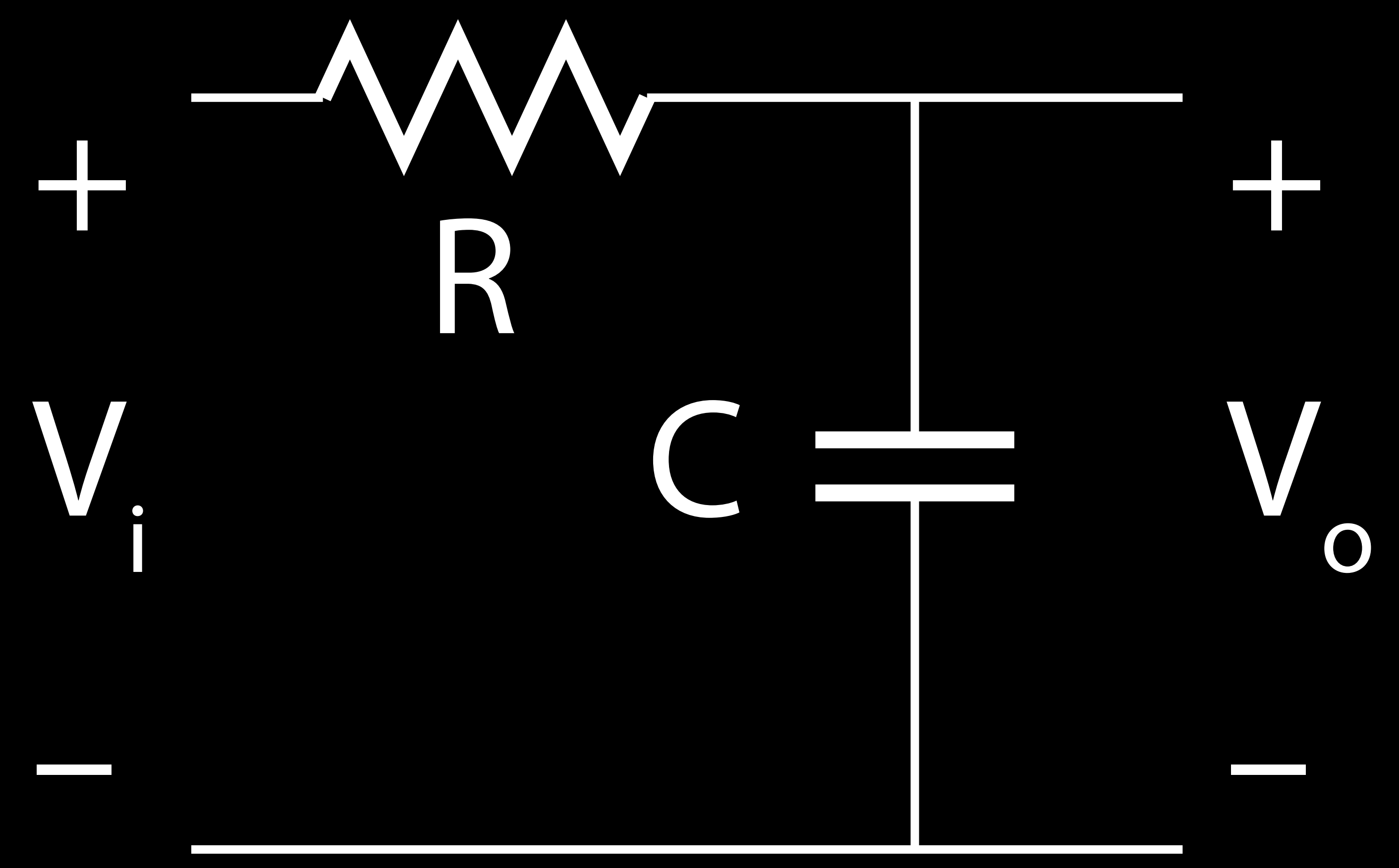
The simplest low-pass filter

- First-order Butterworth, aka RC
- Voltage law: $RC\frac{\partial V_o}{\partial t} + V_0 = V_i$
- If $V_i = A \sin \omega t$, the long-term output is

$$V_o = \frac{A}{\sqrt{1 + (RC\omega)^2}} \sin\left(\omega t - \arctan RC\omega\right)$$

reduced amplitude — still a sin! — phase shift

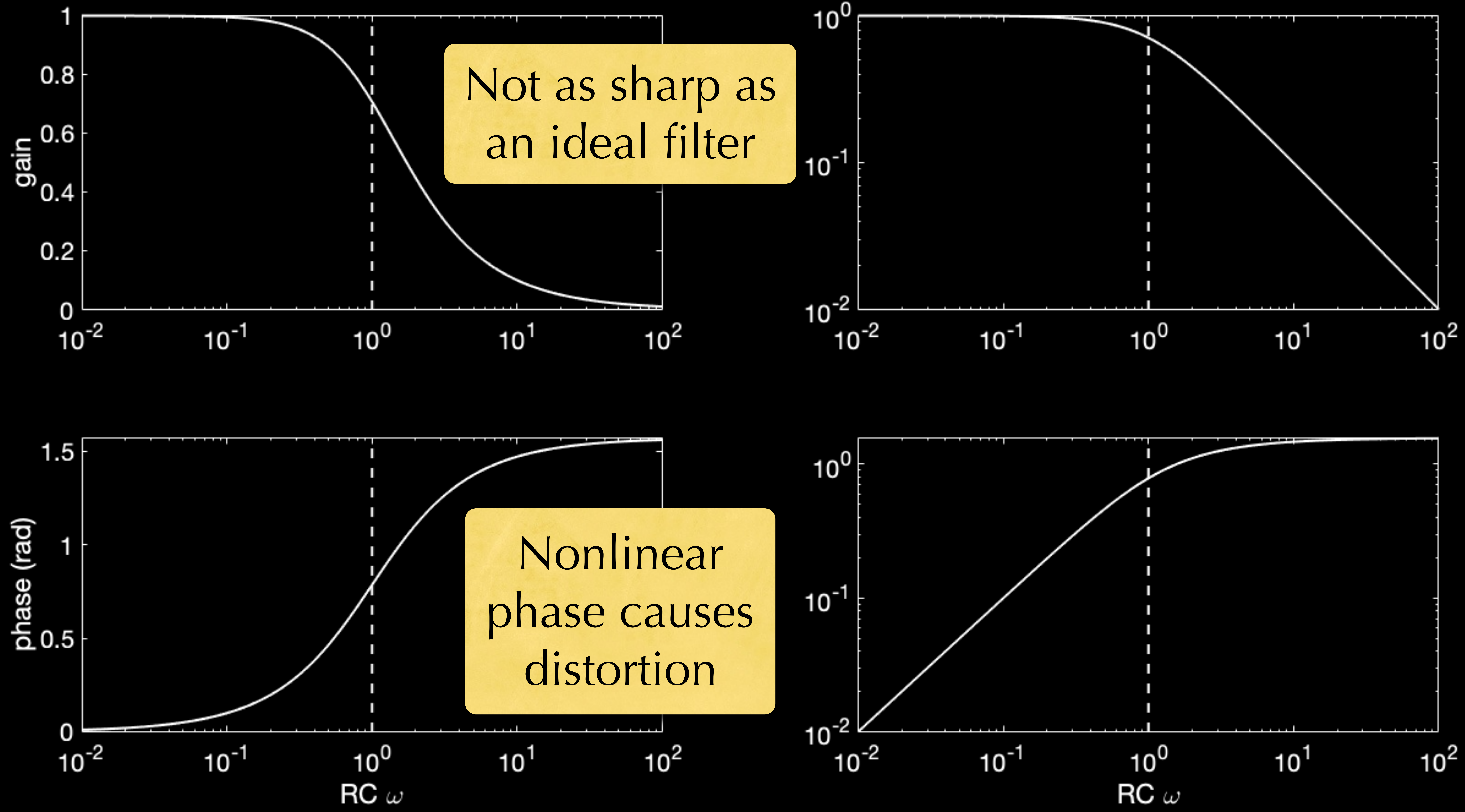
- Amplitude of V_o at frequency $\omega = 0$? A
- Amplitude of V_o at frequency $\omega = \infty$? 0
- Cutoff frequency? $\frac{1}{RC}$



$$G = \frac{1}{\sqrt{1 + (RC\omega)^2}}$$

$$\phi = \arctan RC\omega$$

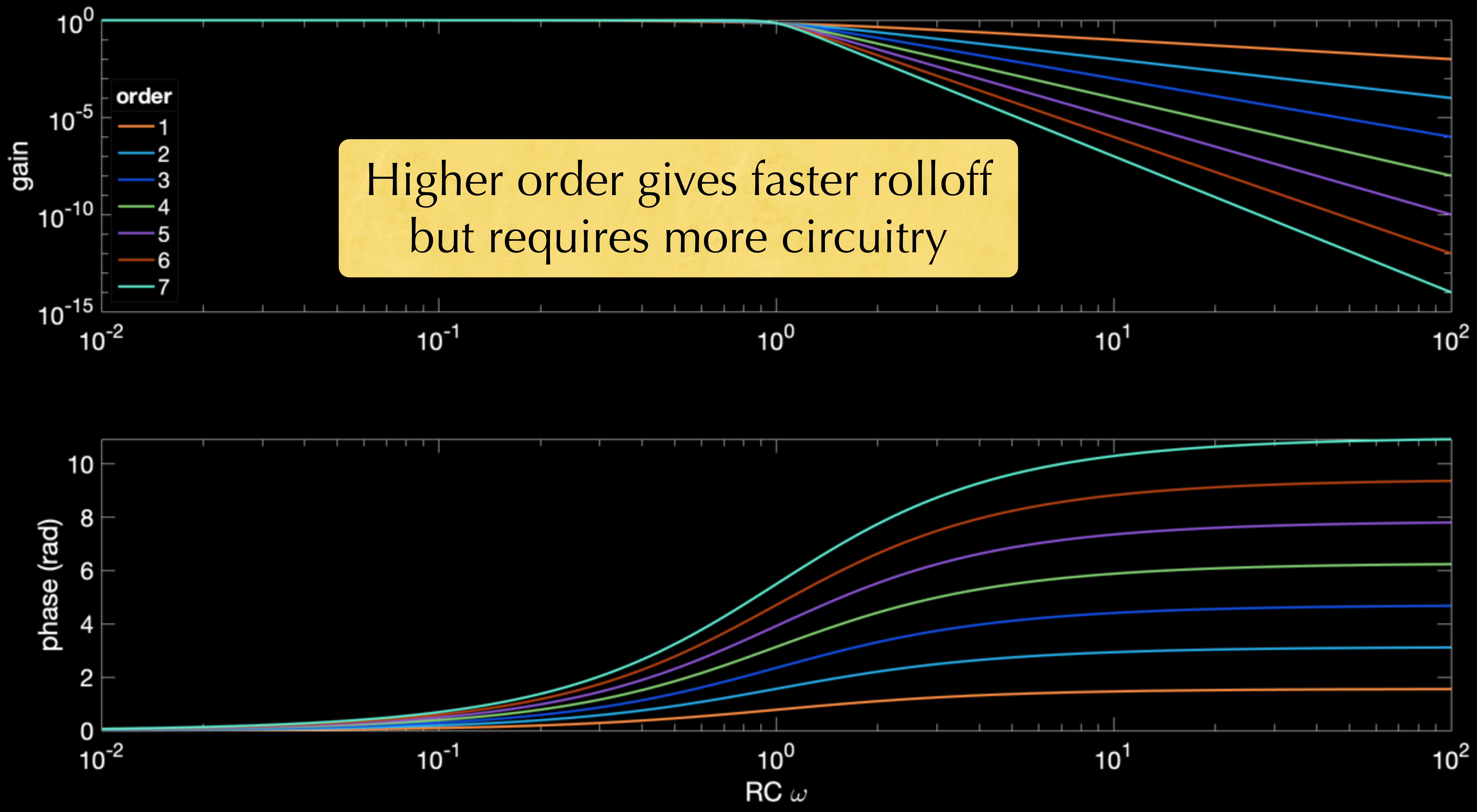
First-order low-pass Butterworth filter

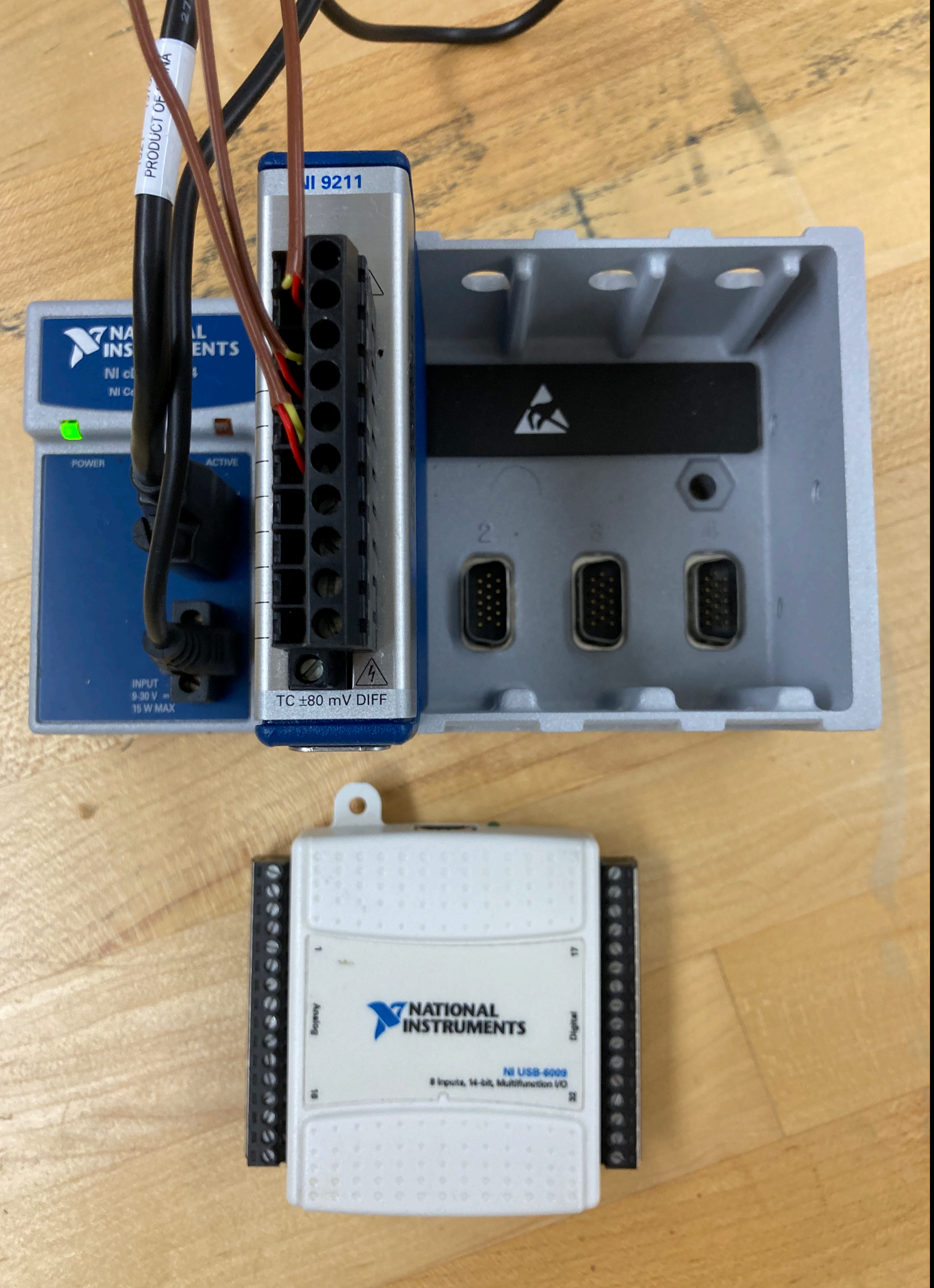


Not as sharp as an ideal filter

Nonlinear phase causes distortion

Low-pass Butterworth filters





Computerized data acquisition

- Often connect via USB; sometimes internal
- Wires from transducers connect to screw terminals
- Range often -5 V to 5 V or 0 V to 10 V
- Special ports for thermocouples
- Analog inputs, digital input/output, occasionally analog outputs
- Varying channel count (1 to ~30), bit depth (8, 12, 16), sampling rates (~1 to ~10^6 S/s)
- Software interface (LabView, Matlab, …)
- National Instruments, Measurement Computing, Keyence, Arduino, …

Writing a Methods section

Explain what you did so that an engineer could repeat your work.

- Post the question(s) being addressed
- State the data you used to answer this question, explaining why this data is appropriate to answer it
- Explain the experimental setup and how you made it
- Explain process of data collection
- Explain how you analyzed the data