Lecture 7, ATIK

Continuous-time filters 2
Discrete-time filters

What did we do last time?

Switched capacitor circuits with nonideal effects in mind

- What should we look out for?
- What is the impact on system performance, like filters.

Continuous-time filters

- The way forward, and the background to generate filters.
- OTA/Gm-C and active-RC

Revisited today

What will we do today?

Continuous-time filters

- Some slides from last lecture
- Scaling
- Sensitivity
- Noise

Discrete-time filters

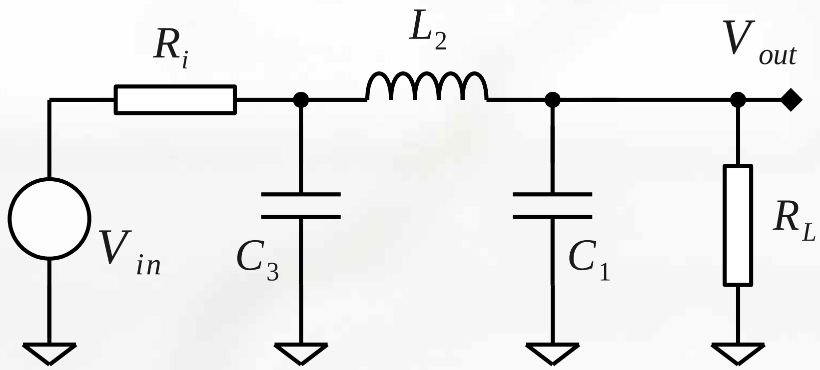
- Simulation of the continuous-time filters

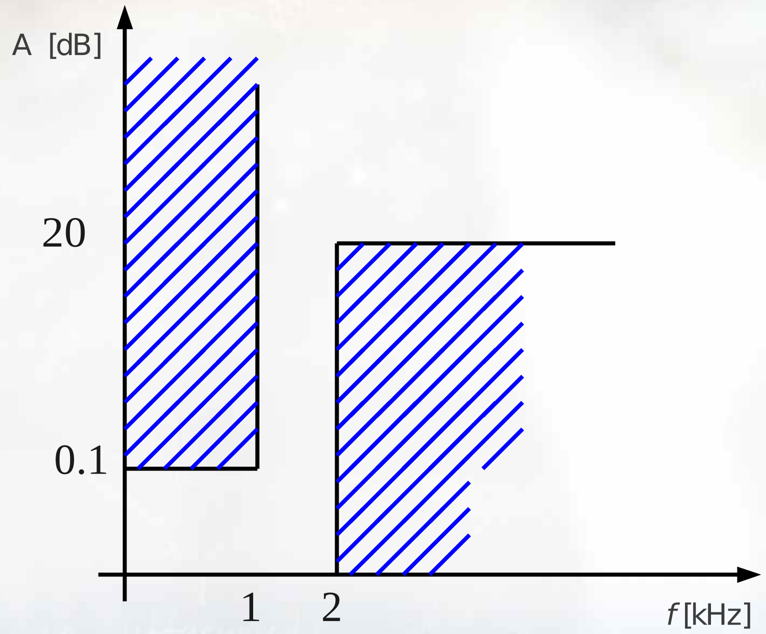
- Discrete-time accumulators
- LDI transform
- Bilinear transform

Ladder networks

Use an analog reference filter, values are given by table or MATLAB-ish.

Typically doubly-resistive terminated for maximum power transfer.



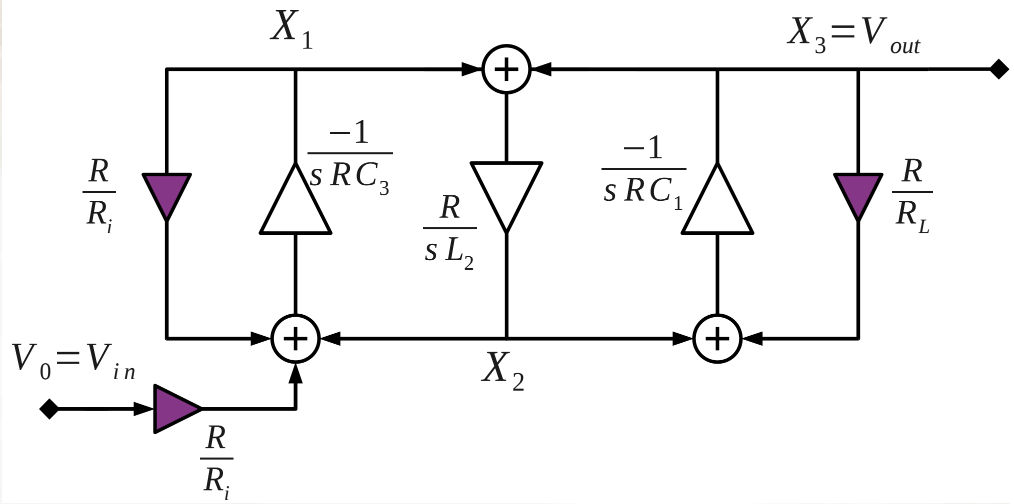


What is the DC gain?

State-space representation

Note the voltages and currents through the ladder network.

Introduce a helping "R" to create voltages



Input current

$$I_0 = \frac{V_0 - V_1}{R_i} \Rightarrow R \cdot I_0 = \frac{R}{R_i} \cdot (V_0 - V_1)$$

Inductor current

$$I_2 = \frac{V_1 - V_3}{sL_2} \Rightarrow R \cdot I_2 = \frac{R}{sL_2} \cdot (V_1 - V_3)$$

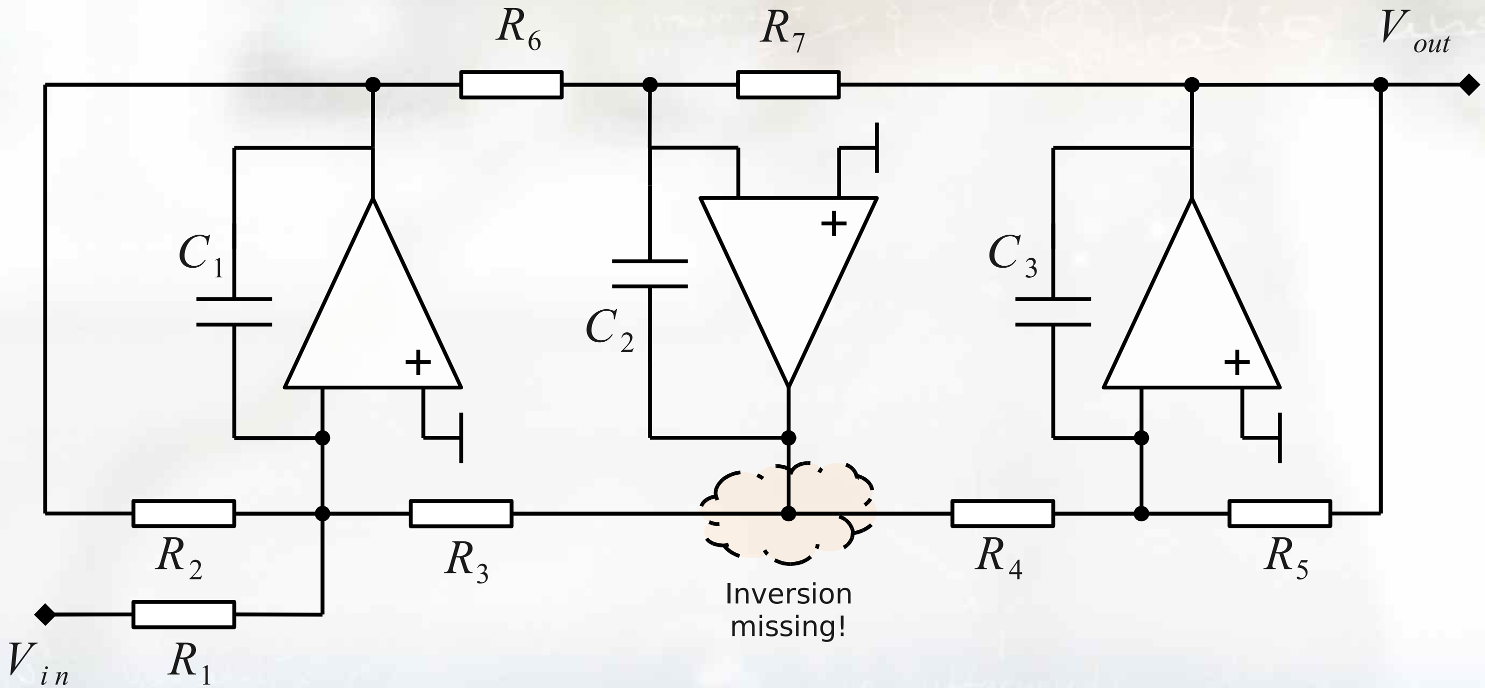
Kirchhoff

$R \cdot I_0 = V_1 \cdot sC_3 + R \cdot I_2$, etc.

Combine all equations into a flow graph

Active-RC implementation

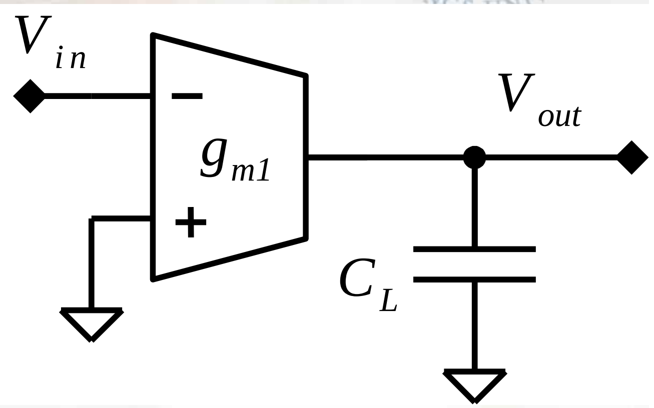
Replace integrators with corresponding active components



Identify the paths and set the values accordingly

Transconductance-C filters

(Operational) transconductance amplifier with capacitive load



$$V_{out}(s)\cdot sC_L = I_{out}(s) = -g_{m1}\cdot V_{in}(s) \Rightarrow \frac{V_{out}(s)}{V_{in}(s)} = \frac{-g_{m1}}{sC_L}$$

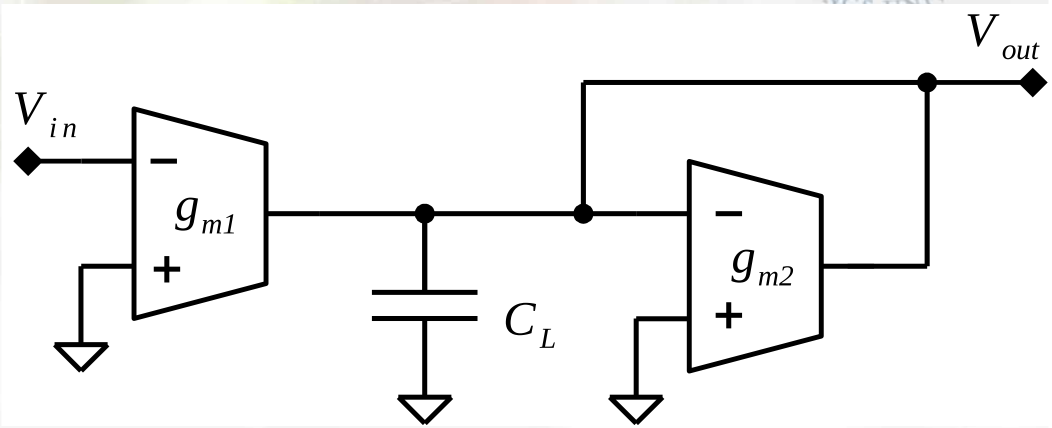
How do we sum several inputs for the Gm-C?

How do we create a gain circuit with Gm-C?

Transconductance-C filters, cont'd

A first-order pole

Requires two transconductors!

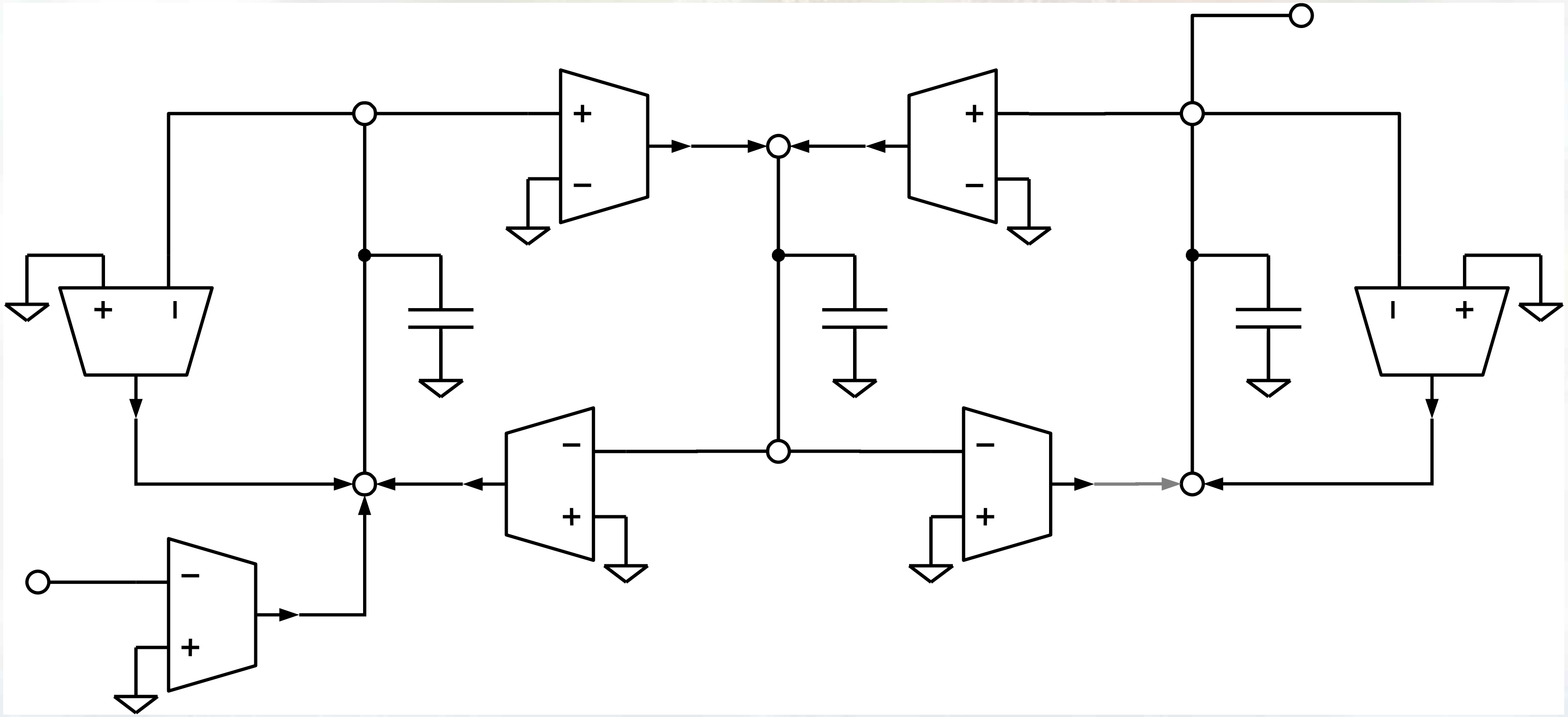


$$-g_{m1}\cdot V_{in}(s)-V_{out}(s)\cdot sC_L-g_{m2}\cdot V_{out}(s)=0$$

$$\frac{V_{out}(s)}{V_{in}(s)}=\frac{-g_{m1}}{g_{m2}+s\cdot C_L}=\frac{-g_{m1}/g_{m2}}{1+\frac{s}{g_{m2}/C_L}}$$

Transconductance-C filters, cont'd

Leapfrog: replacement of active-RC integrators with Gm-C elements.



Requires more active elements, but easier to tune

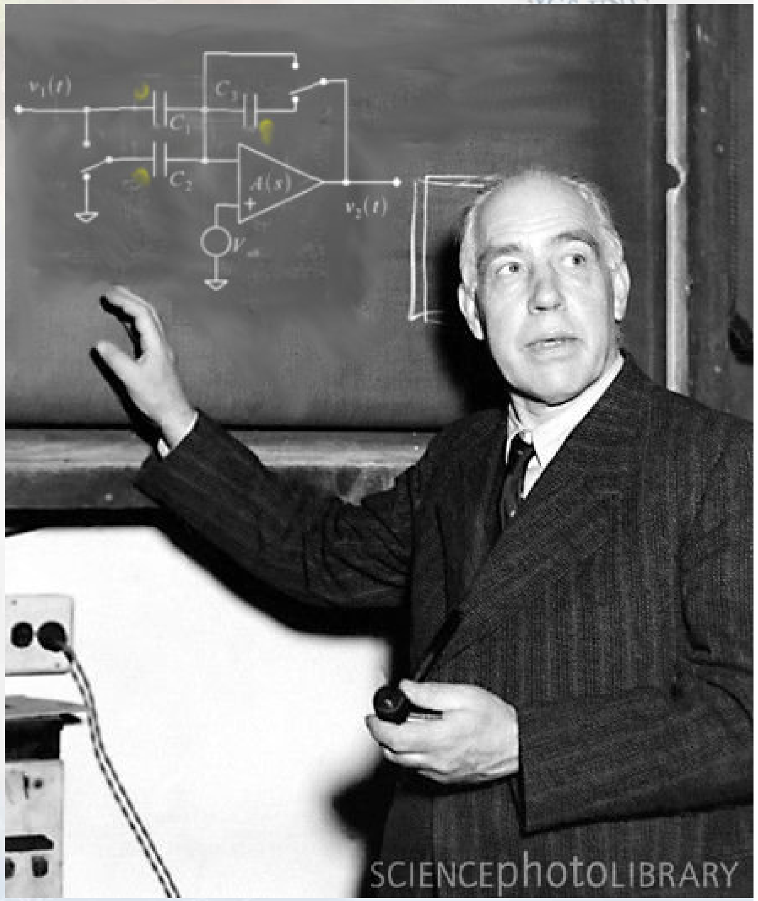
Continuous-time filters, scaling

For optimal performance, we need to scale our filters

- Minimize distortion, i.e., minimize your swing
- Maximize SNR, i.e., maximize your swing

What norm to use in order to scale?

- Different nodes might peak at different frequencies!

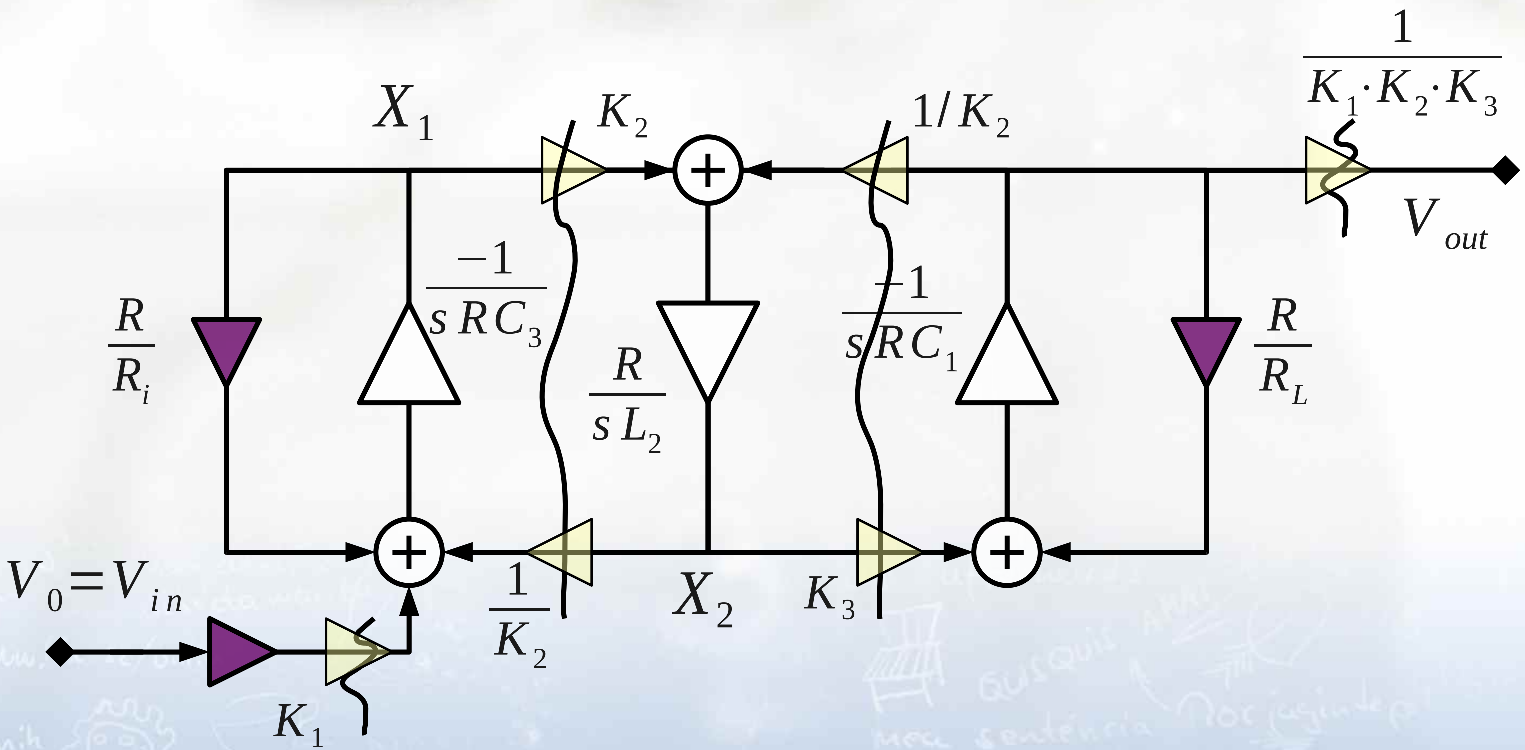


Continuous-time filters, scaling example

1) Originate from flow graph and introduce scaling coefficients.

2) All inputs scaled with K and all outputs scaled with 1/K

3) Take them "one-by-one" from input towards the output



Continuous-time filters, sensitivity

General

The transfer function is a function of all values, A_0, R, C, etc.

We can define the sensitivity as

$$S_i^H = \frac{q_i}{H} \cdot \frac{\partial H}{\partial q_i}$$

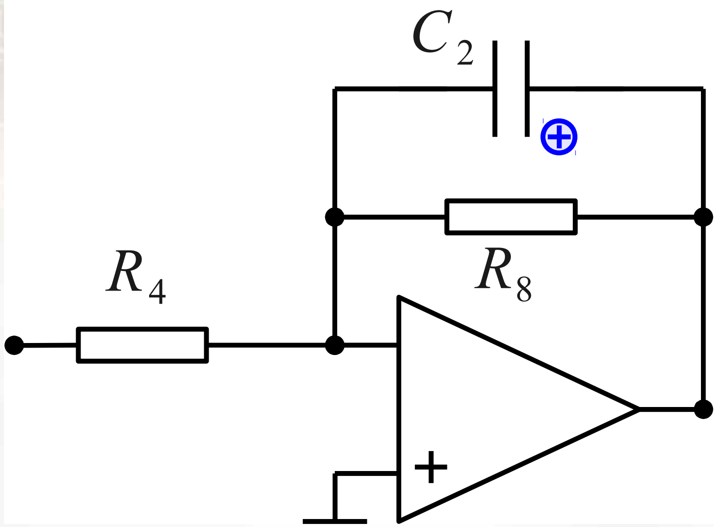
Leapfrog filters are less sensitive

than second-order sections.

Continuous-time filters, sensitivity example

First-order pole with active RC

$$H(s) = \frac{-R_8/R_4}{1 + \dfrac{s}{1/C_2 R_8}}$$



Variation as function of RC

$$S_C^H = \frac{C_2}{H} \cdot \frac{\partial H}{\partial C_2} = \frac{C_2}{H} \cdot H \cdot \frac{-1}{1 + \dfrac{s}{1/C_2 R_8}} \cdot s R_8$$

such that

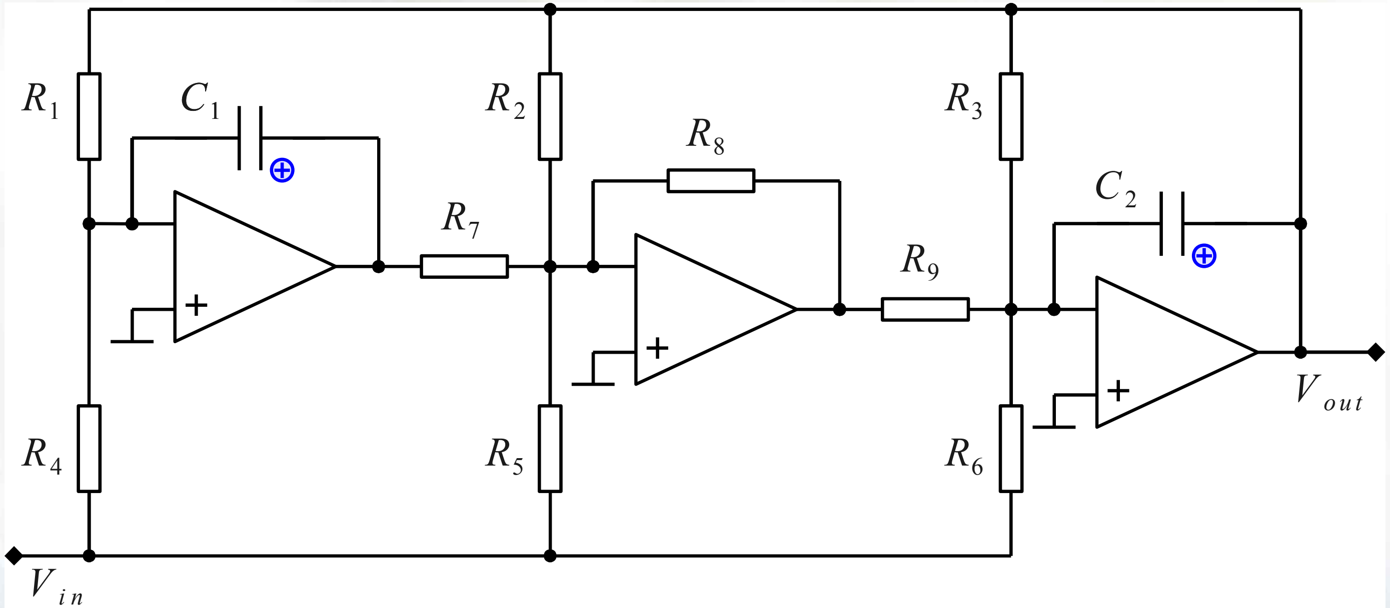
$$S_C^H = \frac{-s C_2 R_8}{1 + s C_2 R_8}$$

What does this imply?

Continuous-time filters, noise

All OP amp noise can be reverted back to their inputs

- Scaled with the transfer functions to the output
- Use the superfunction



All resistor noise is converted to output too

Going to discrete-time domain

A general transfer function is given by

$$H(z) = \frac{Y(z)}{X(z)} = \frac{a_0 + a_1 z + a_2 z^2 + \ldots}{b_0 + b_1 z + b_2 z^2 + \ldots}$$

Identify integrators in the expression and rewrite

$$A(z) = \frac{\alpha}{z-1}$$

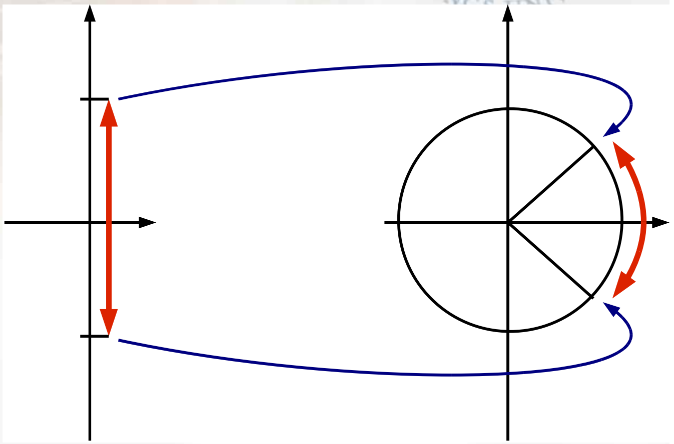
(Notice that the approach is a bit different to "digital" IIR/FIR filter design)

For filter purposes we will originate from the continuous-time filter specification

eventually determines our a,b values.

The lossless discrete integrator transform

Maps just part of the frequency range



$$s = s_0 \cdot \left[z^{0.5} - z^{-0.5} \right] = s_0 \cdot \frac{1 - z^{-1}}{z^{-0.5}}$$

with the scaling factor according to

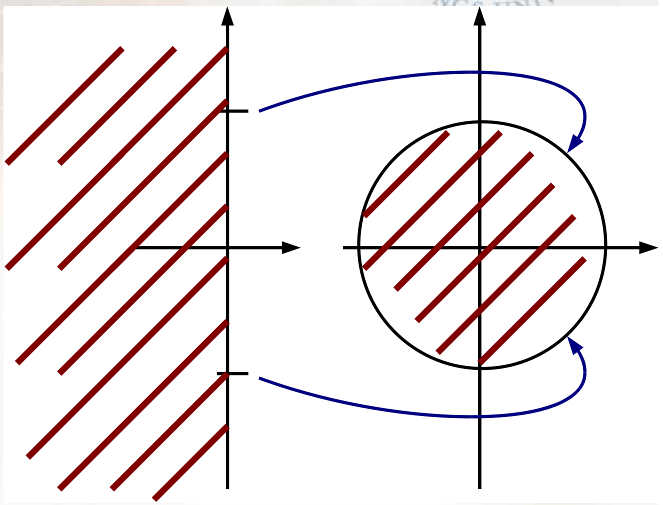
$$\omega_a = 2 s_0 \sin\left(\frac{\omega T}{2}\right)$$

which maps the analog frequencies with the discrete-time angles.

We have already decided to use high oversampling ratio, so the LDI transform is sufficiently OK.

The bilinear transform

Maps the "whole" analog LHP on the unit disk



$$s = s_0 \cdot \frac{z-1}{z+1}$$

with the scaling factor according to

$$\omega_a = s_0 \tan\left(\frac{\omega T}{2}\right)$$

which maps the analog frequencies with the discrete-time angles.

Bilinear transform is accurate,

but sensitivity requires us to select a low scaling factor.

Accumulators of different flavours

See the handouts for state-space realization and implementations

http://www.es.isy.liu.se/courses/ANIK/download/filterRef/ANTIK_0NNN_LN_leapfrogFiltersOH1_A.pdf

http://www.es.isy.liu.se/courses/ANIK/download/filterRef/ANTIK_0NNN_LN_leapfrogSynthesisExtra1_A.pdf

http://www.es.isy.liu.se/courses/ANIK/download/filterRef/ANTIK_0NNN_LN_leapfrogSynthesisExtra2_A.pdf

http://www.es.isy.liu.se/courses/ANIK/download/switcapRef/ANTIK_0NNN_LN_switcapHandout_B.pdf

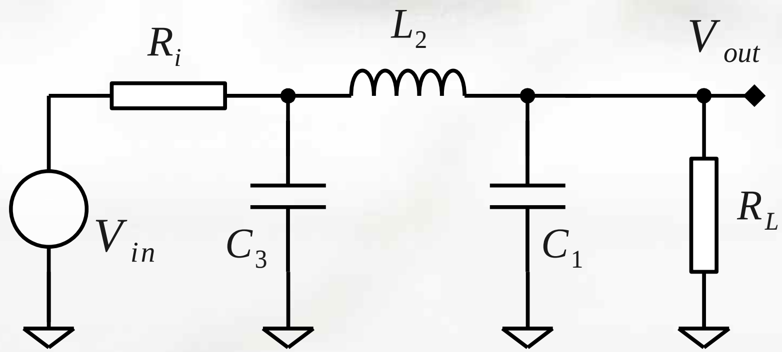
The steps forward

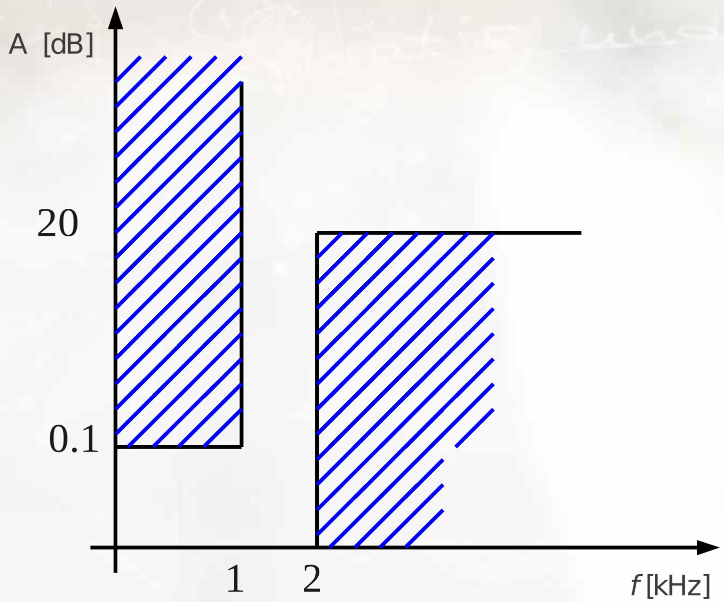
1) Analog reference filter
2) Frequency scaling of reference filter
3) Signal flow graph
4) Replacing integrators with discrete-time counterparts
5) Eliminate the z^(1/2)
6) Introduce SC network
7) Map the values
8) Scale

This you will do in the lab!

Step 1, Analog reference filter

Use a reference filter to get the order and component values





Optimum wrt. sensitivity (exercises in lessons)

Example: $C_1 = C_3 = 0.5$ and $L_2 = 1.3333$, $R_L = 1000$, $\omega_0 = 1000$

Step 2, Denormalization of reference filter

R0 is the termination load

$$R_n = \frac{R}{R_0}, \quad L_n = \frac{\omega_0 L}{R_0}, \quad C_n = \omega_0 C R_0$$

Example values gives

$$R = 1000$$

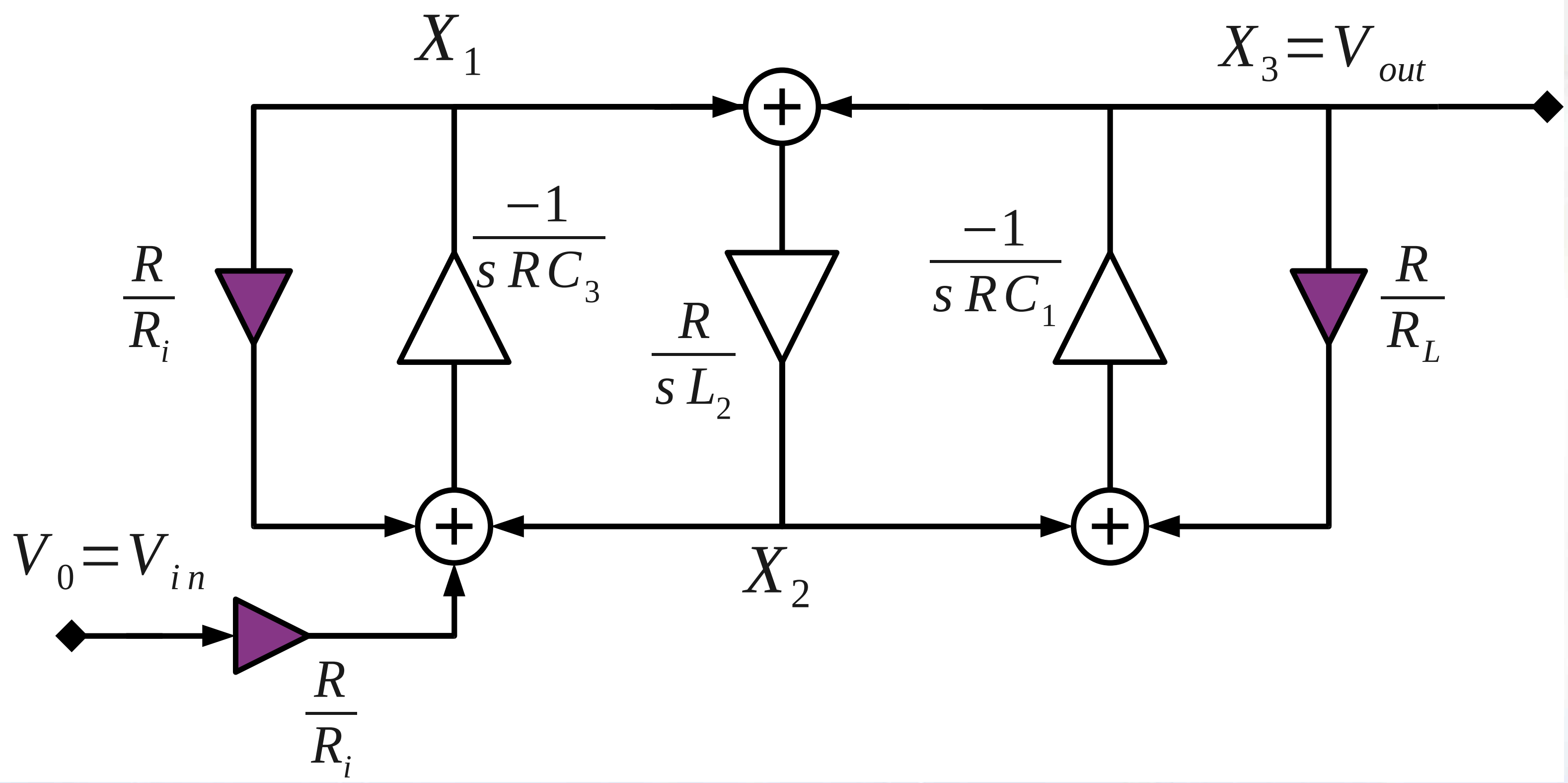
$$L = 1.3333$$

$$C_1 = C_3 = 0.5 \cdot 10^{-6}$$

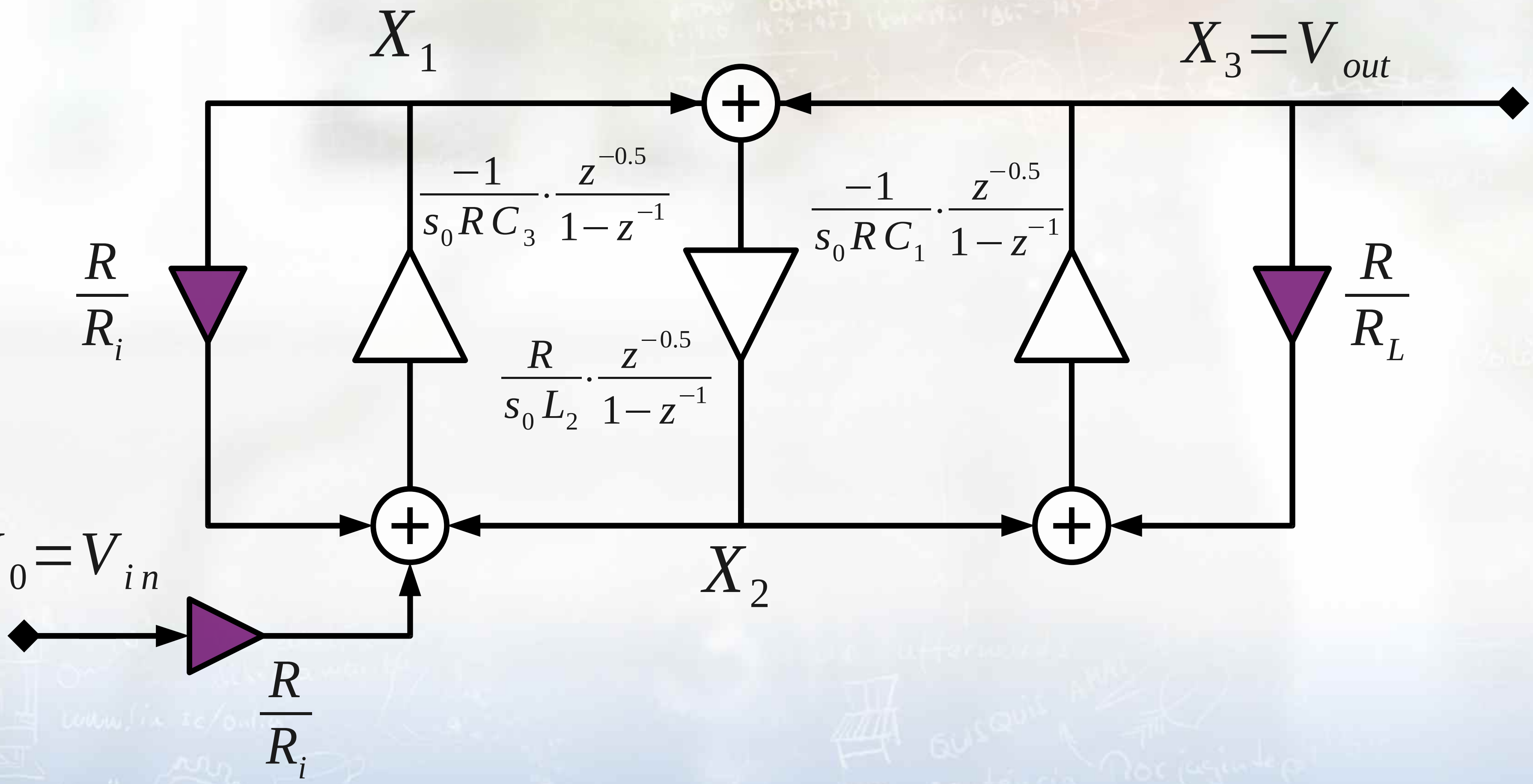
In the lab, a MATLAB script is given

Step 3, Signal flow graph

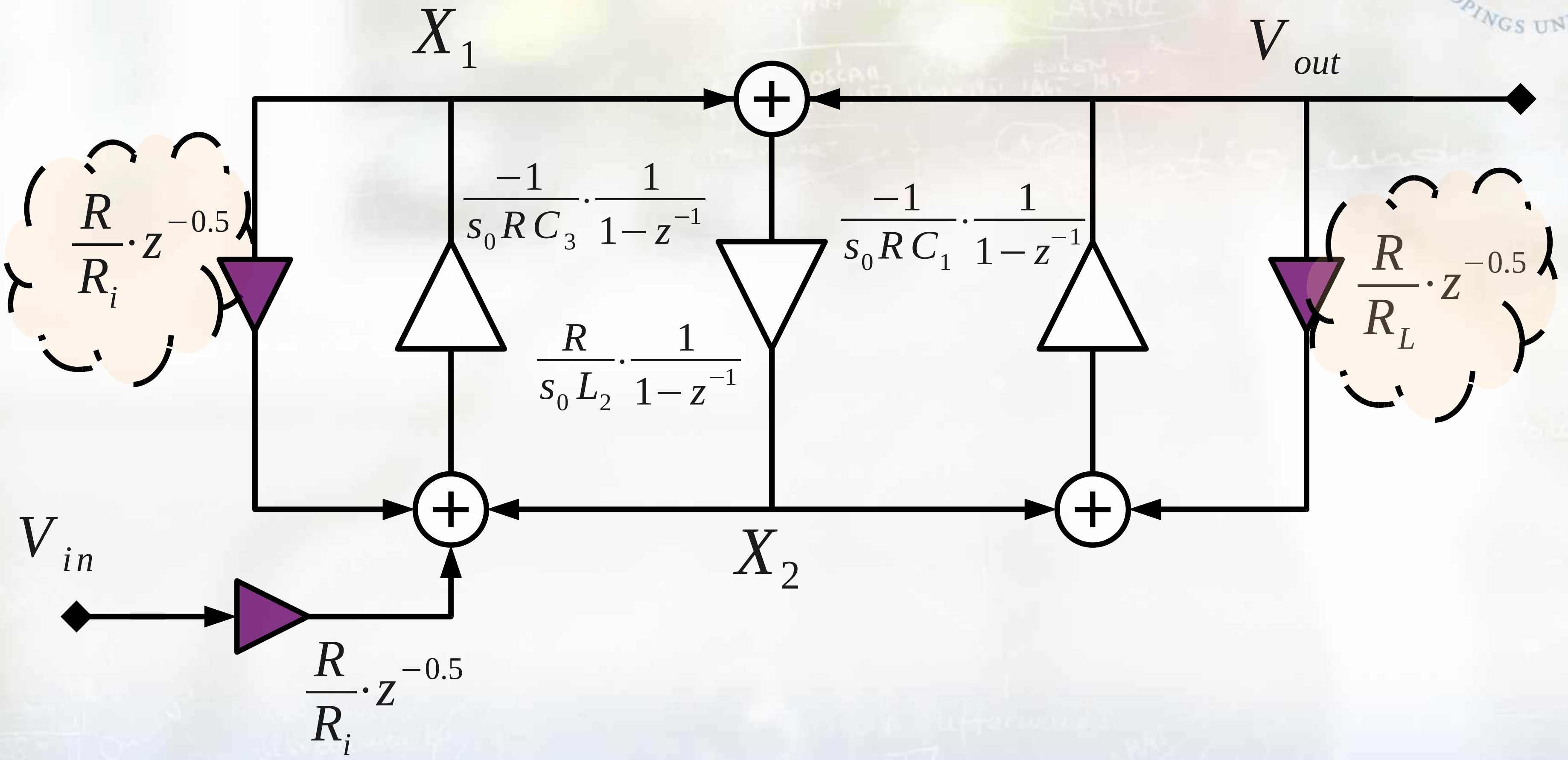
Example third-order leapfrog



Step 4, Replacing integrators with discrete-time counterparts (LDI)



Step 5, Eliminate the z^(1/2)



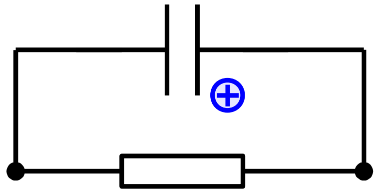
Step 5, Eliminate the z^(1/2), cont'd

Consider the product

$$R \cdot z^{-0.5} = \frac{R}{z^{0.5}} = \frac{R}{\cos\frac{\omega T}{2} + j\sin\frac{\omega T}{2}} = \frac{R}{\sqrt{1-\left(\frac{\omega_a}{2s_0}\right)^2} + j\frac{\omega_a}{2s_0}}$$

Compare with a capacitor in parallel with a resistor:

$$C = \frac{1}{2s_0 R}$$

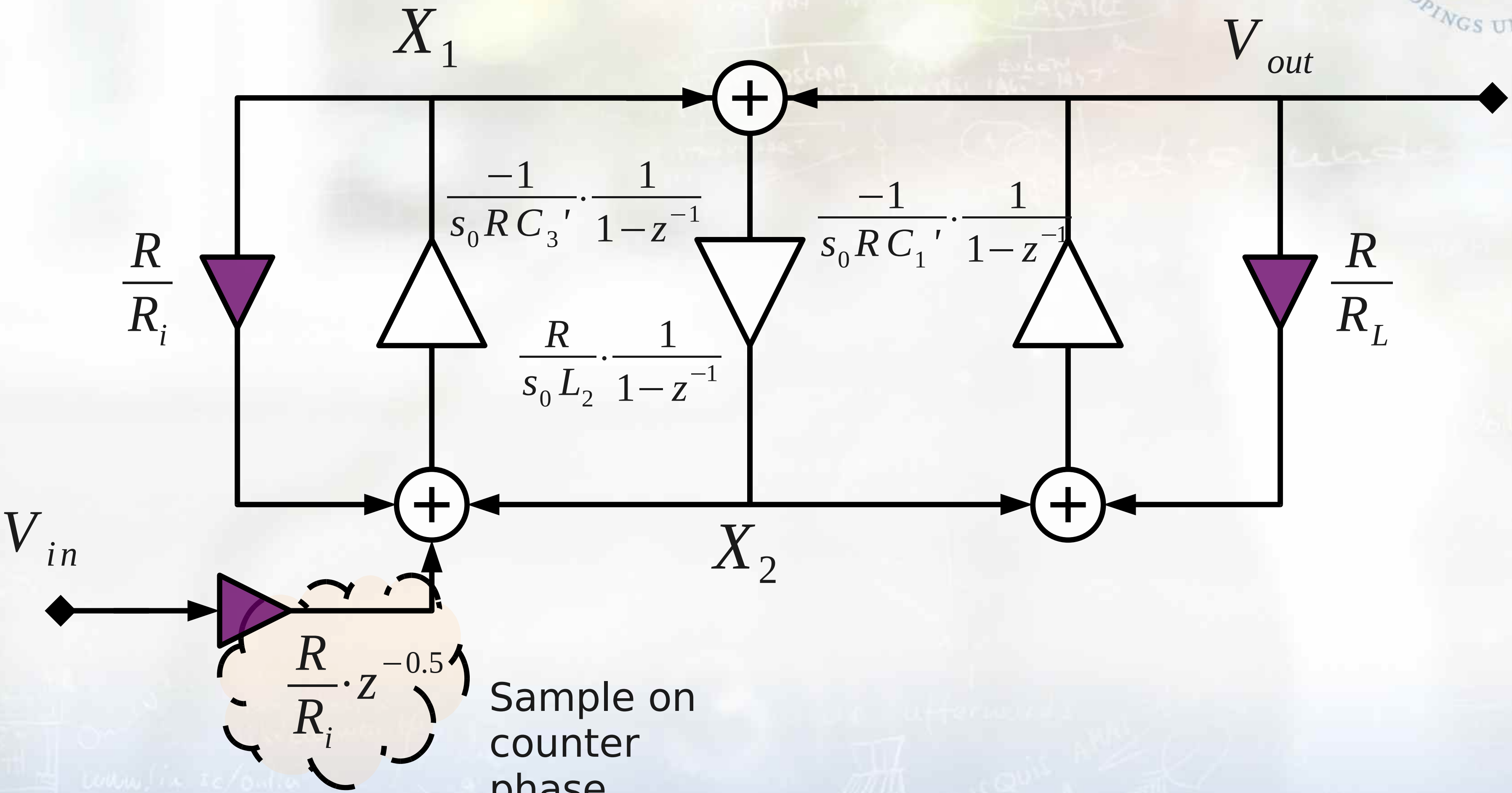


$$R' = \frac{R}{\sqrt{1-\left(\frac{\omega_a}{2s_0}\right)^2}}$$

Remember that originally this resistance was in parallel with capacitor and we can reduce the C value.

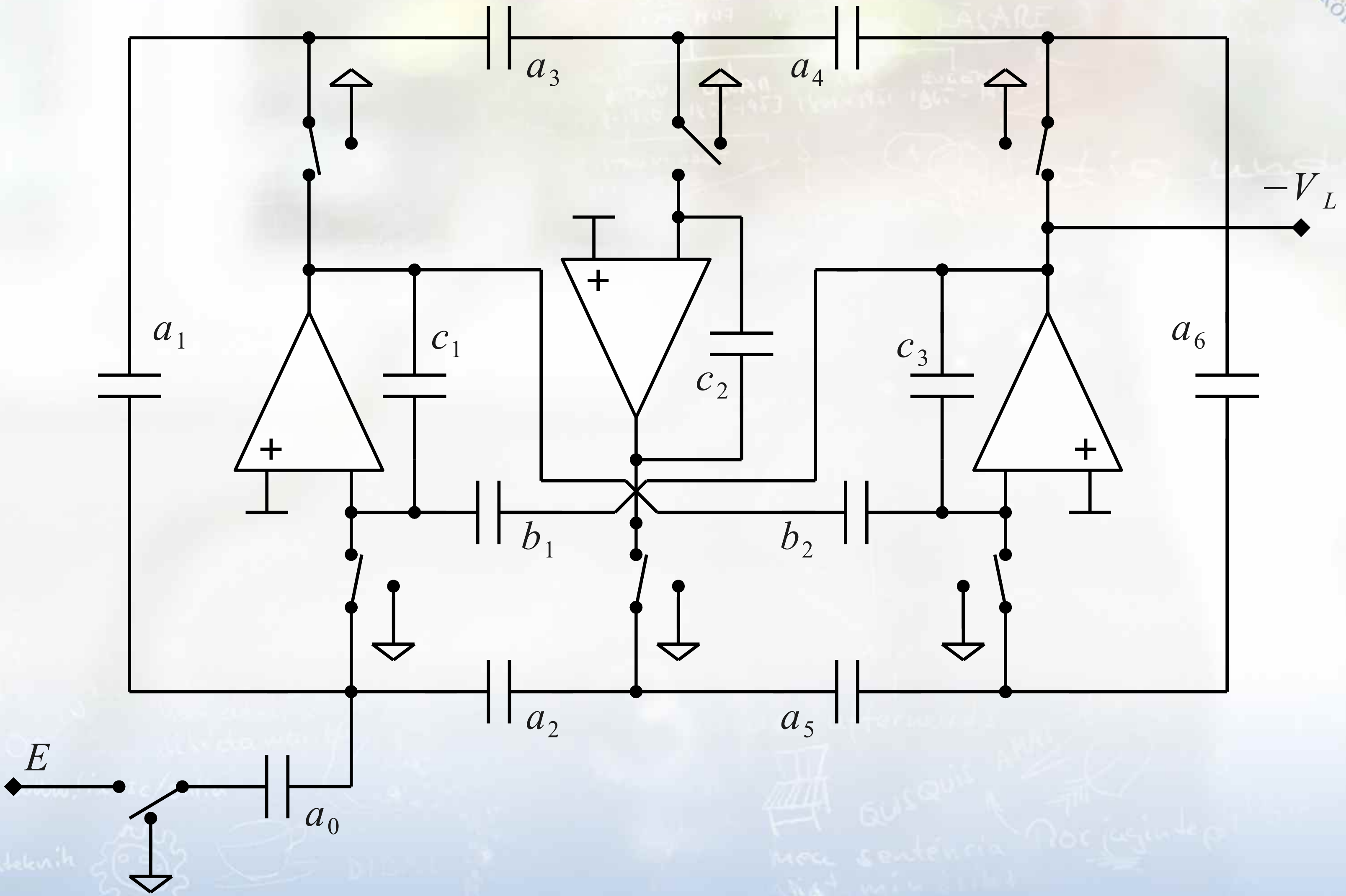
The change in resistance can be ignored since the value is much smaller than unity.

Step 5, Eliminate the z^(1/2), cont'd



Sample on counter phase.

Step 6, Introduce SC network



Step 7, Map the values

Consider the transfer functions from node to node

Example: V_{in} to X_1:

$$X_1 = \frac{1}{s_0 R_i C_3'} \text{ vs. } \frac{a_0}{c_1} \text{ vs. } \frac{a_1}{c_1}$$

etc.

More variables than equations and you need to fix values:

Typically choose all the opamp capacitors equal

All others: "as equal as possible"

Step 8, Scale

Follows exactly the same principles as fore continous-time filters.

Introduce scaling and make sure all entry points are scaled with K and all exit points are scaled with 1/K.

Left as exercise and laboratory work.

Remember that reference filter has a DC gain of 0.5!

Remember that gain is dependent on sample frequency!

Conclusions SC filters

Today computer-aided, of course.

- Since it is computer aided - choose the leapfrog filters for optimum sensitivity!
- Use fixed templates and plug in your values

Remember to use reasonable values

- i.e., in an integrated circuit, the sizes should be in pF region.
- This also implies high cut-off frequencies!

What did we do today?

Continuous-time filters

- Wrap-up and some more conclusions

Discrete-time filters

- Simulation of the continuous-time filters

- Discrete-time accumulators
- LDI transform
- Bilinear transform

What will we do next time?

Data converters

- Fundamentals

- DACs

Lecture 8, ATIK

Data converters 1

LiU EXPANDING REALITY