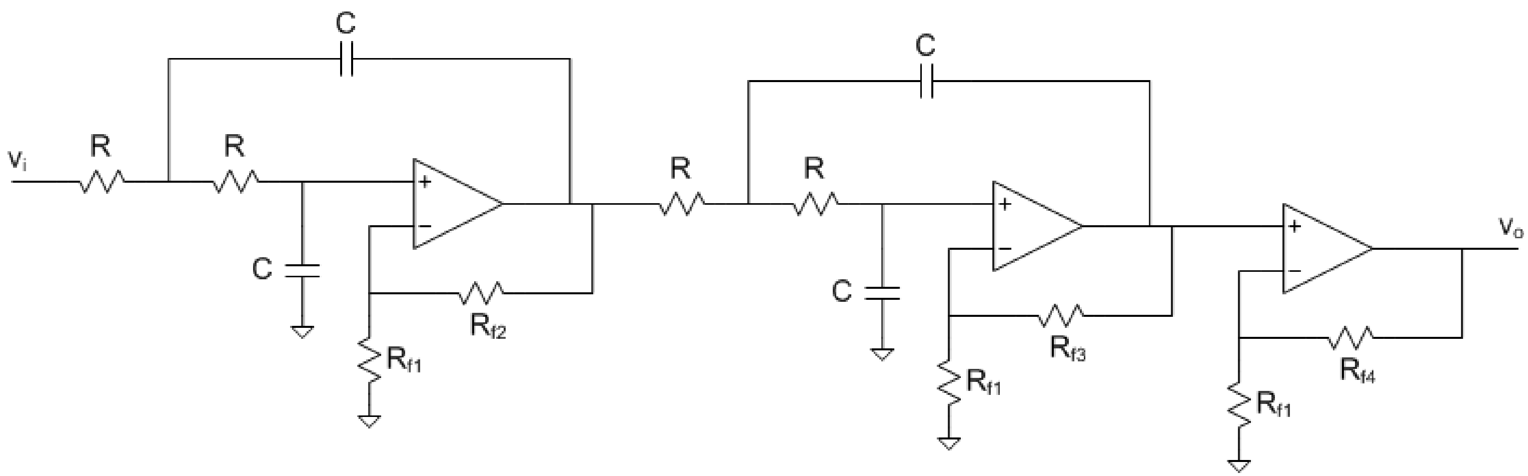
SECTION 7: ACTIVE FILTERS

ENGR 203 – Electrical Fundamentals III

Introduction

- In ENGR 202 we studied different types of first- and second-order ***passive*** filters
  - *Passive*, because they contain only passive components:
    - Resistors, capacitors, and inductors
- Can also construct filters using opamps
  - ***Active filters***



Introduction

- Active filters have advantages over passive filters:
  - Can build ***high-Q filters without inductors***
  - ***Low output impedance***
  - Easily ***adjustable***: f_c, Q
  - Can provide ***gain*** ($> 0\ \text{dB}$)
- Before getting into the design of active filters, we will look at two fundamental filter building blocks:
  - Opamp integrators
  - Opamp differentiators

4

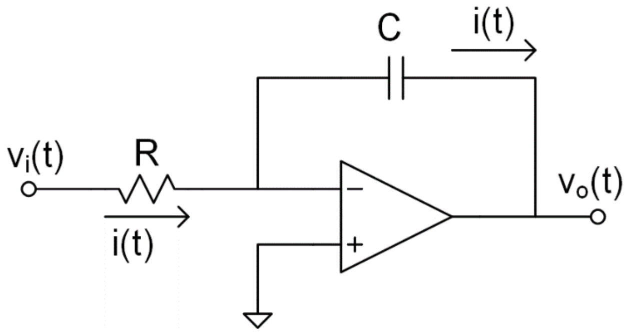
Opamp Integrators

Integrators and Differentiators

- Opamp circuits can perform many different ***mathematical operations***
  - ***Operational amplifiers***
- Multiplication
  - Inverting and non-inverting amplifiers
- Addition and subtraction
  - Summing and difference amplifiers
- Can also perform ***integration*** and ***differentiation***
  - Feedback controllers
  - Building block of active filters

Opamp Integrator – Time Domain

- Analyze the opamp integrator in the time domain



- Virtual ground at inverting input, so

$$i(t) = \frac{v_i(t)}{R}$$

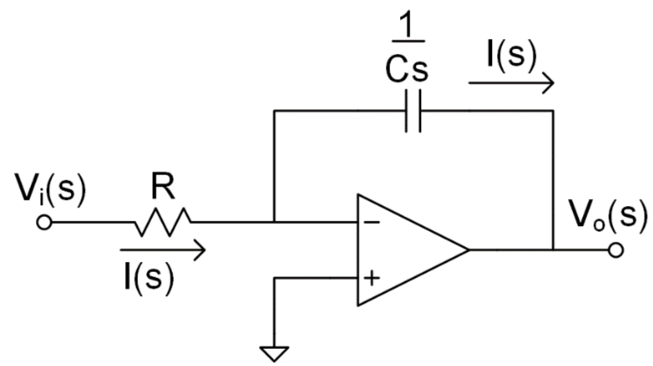
- Capacitor ***integrates*** input current to give output voltage

$$v_o(t) = -\frac{1}{C}\int_0^t i(\tau)d\tau \; v_o(t) = -\frac{1}{RC}\int_0^t v_i(\tau)d\tau$$

- Output is the (scaled and inverted) integral of the input

Opamp Integrator – Laplace Domain

- Analyze the opamp integrator in the Laplace domain



- Again, a virtual ground at inverting input, so

$$I(s) = \frac{V_i(s)}{R}$$

- Output voltage:

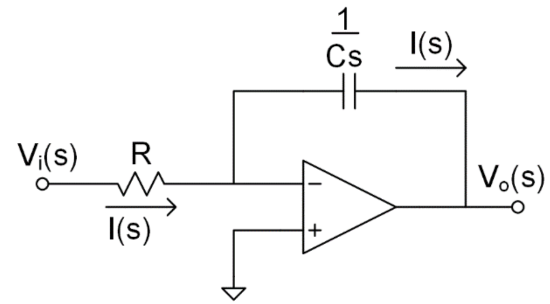
$$V_o(s) = -I(s)\frac{1}{Cs} = -\frac{V_i(s)}{RCs} = -\frac{1}{s}\cdot\frac{V_i(s)}{RC}$$

- Recall that multiplication by $1/s$ in the Laplace domain corresponds to integration in the time domain

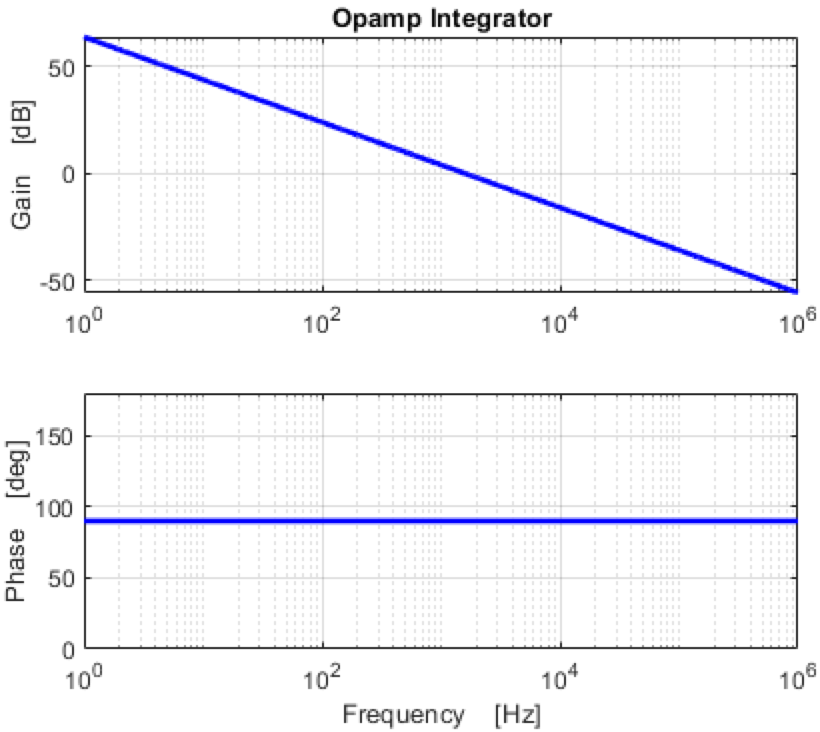
Opamp Integrator – Frequency Response

- Transfer function:

$$G(s) = -\frac{1}{RCs}$$



- Single pole at $s = 0$
  - Gain: constant slope of -20 dB/dec
    - Infinite DC gain
  - Phase: -90° from integrator pole + 180° from inversion yields constant +90°



Ideal Integrator - Problem

- Laplace domain step response of the ideal integrator

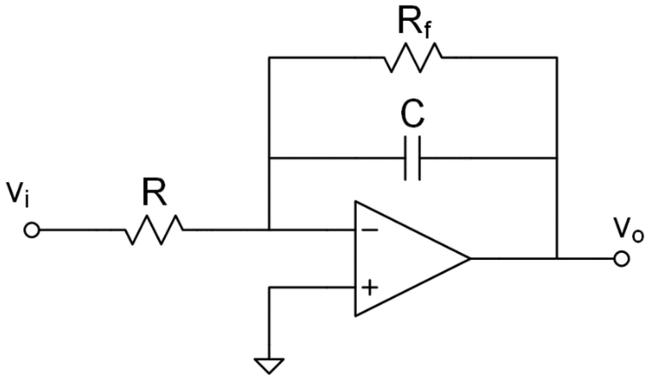
$$V_o(s) = \frac{1}{s} \cdot G(s) = -\frac{1}{RC} \cdot \frac{1}{s^2}$$

- Inverse transforming to the time domain

$$v_o(t) = -\frac{1}{RC} \cdot t$$

- Output increases linearly with time
- Opamp will quickly saturate in response *any* DC input component
  - Infinite DC gain
- Not a practical circuit
  - Inputs will always have some non-zero offset
  - Real (non-ideal) opamps have non-zero offset voltages and input bias currents

Practical Opamp Integrator



- Problem with ideal integrator is infinite DC gain
  - No DC feedback
  - Open-loop at DC
- Add a feedback resistor in parallel with the capacitor
  - Now there is a feedback path for DC signals
  - DC gain limited to R_f/R
- Behaves as an inverting opamp at low frequencies
- Still behaves as an integrator at high frequencies
- A ***practical*** or ***lossy*** integrator circuit

Opamp Integrator – Frequency Response

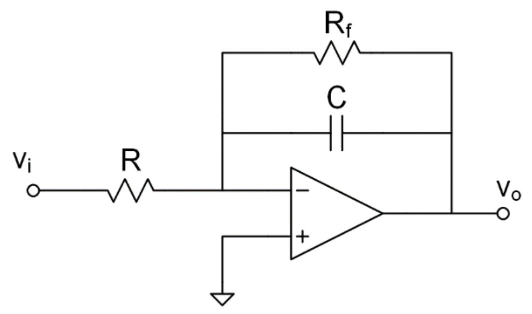
- Transfer function:

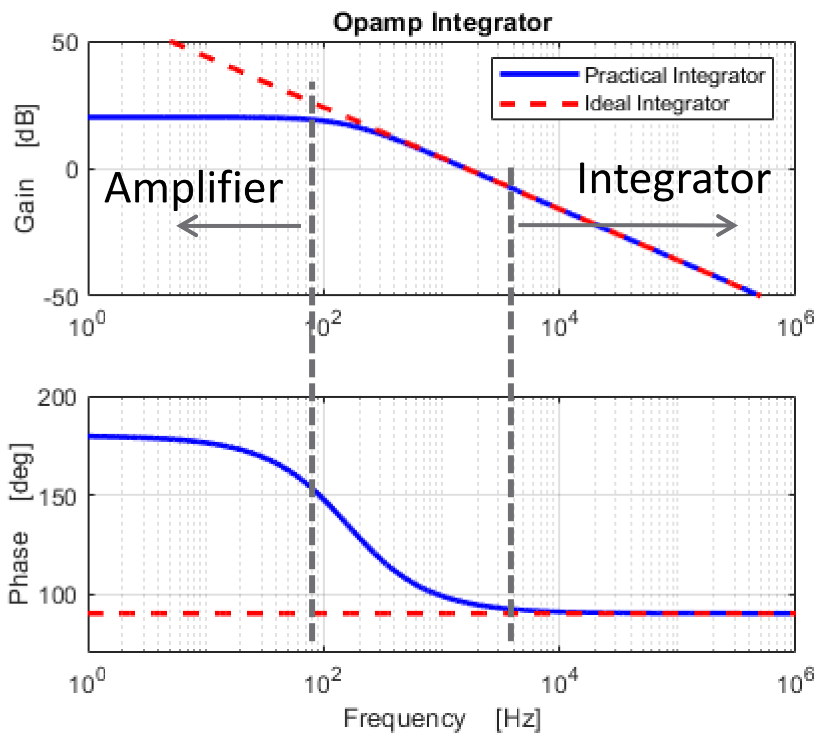
$$G(s) = -\frac{\frac{\frac{R_f}{Cs}}{R_f + \frac{1}{Cs}}}{R} = -\frac{R_f}{R}\frac{1}{R_f Cs + 1}$$

- Pole (corner frequency) set by the feedback network:

$$\omega_c = \frac{1}{R_f C}$$

- For $\omega \gg \omega_c$, still behaves like an integrator
  - Gain: rolls off at -20 dB/dec
  - Phase: ~90°

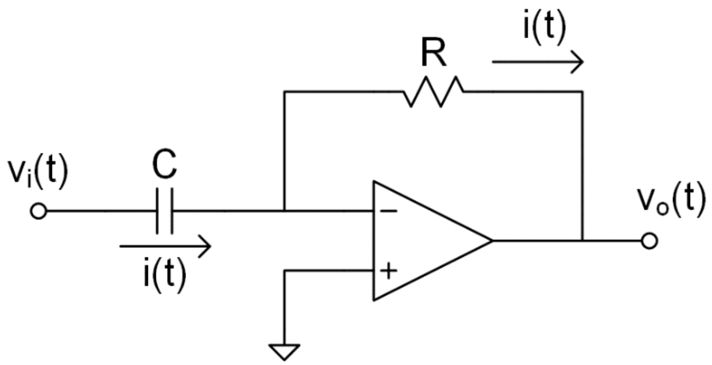




Opamp Differentiators

Opamp Differentiator – Time Domain

- Analyze the opamp differentiator in the time domain



- Virtual ground at inverting input, so

$$i(t) = C\frac{dv_i}{dt}$$

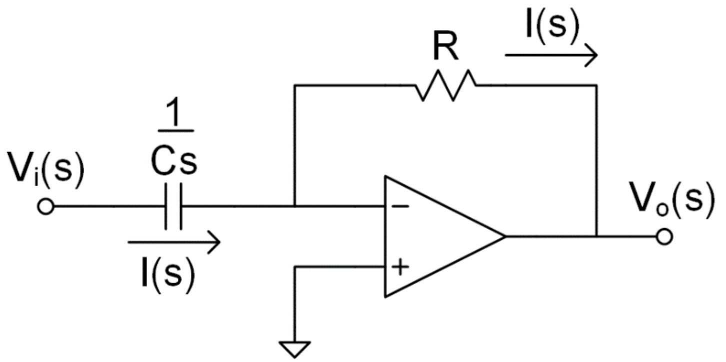
- Ohm's law gives the output voltage

$$v_o(t) = -Ri(t) = -RC\frac{dv_i}{dt}$$

- Output is the (scaled and inverted) derivative of the input

Opamp Differentiator – Laplace Domain

- Analyze the differentiator in the Laplace domain
- Again, a virtual ground at inverting input, so



$$I(s) = Cs \cdot V_i(s)$$

- Output voltage:

$$V_o(s) = -RI(s) = -RCsV_i(s) = -s \cdot RCV_i(s)$$

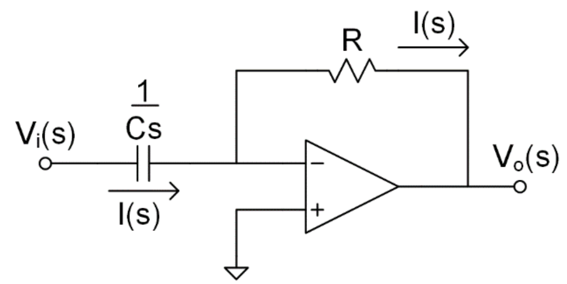
- Recall that multiplication by s in the Laplace domain corresponds to differentiation in the time domain

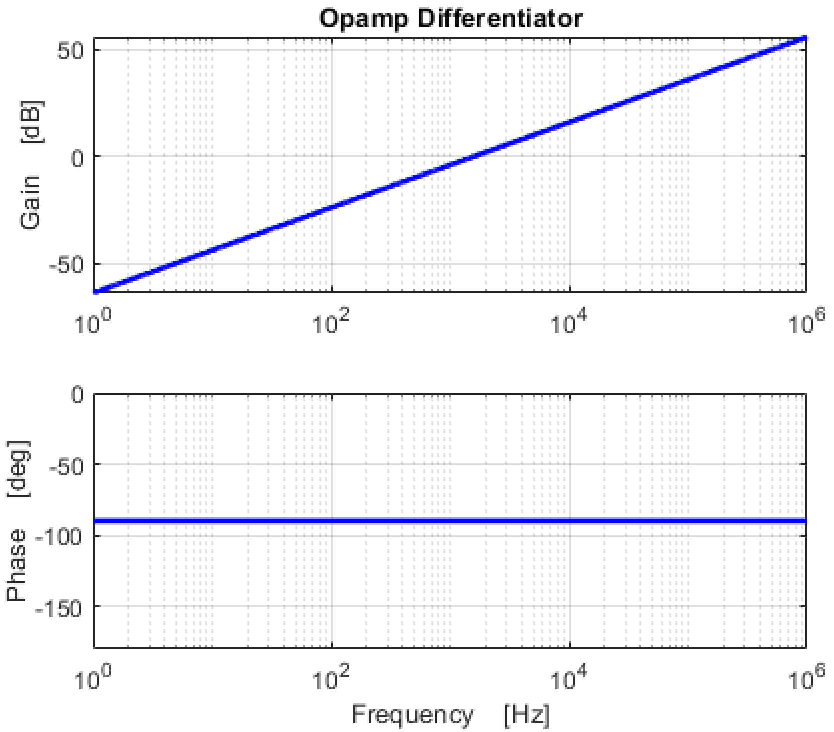
Opamp Differentiator – Frequency Response

- Transfer function:

$$G(s) = -RCs$$

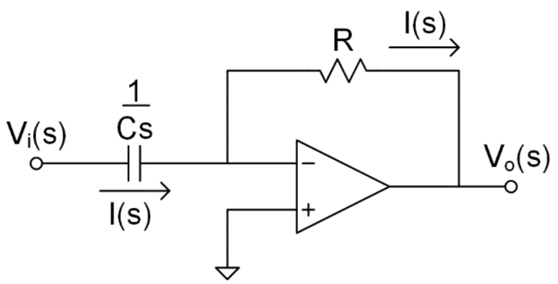
- Single zero at $s = 0$
  - Gain: constant slope of +20 dB/dec
    - Very large high-frequency gain
  - Phase: +90° from zero at the origin + 180° from inversion yields constant +270° = -90°

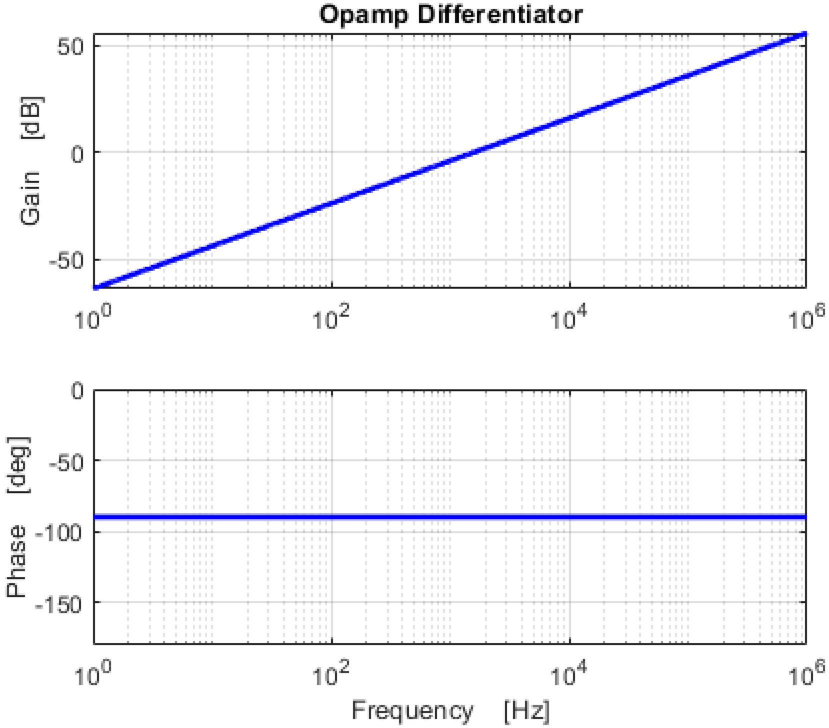




Ideal Differentiator - Problem

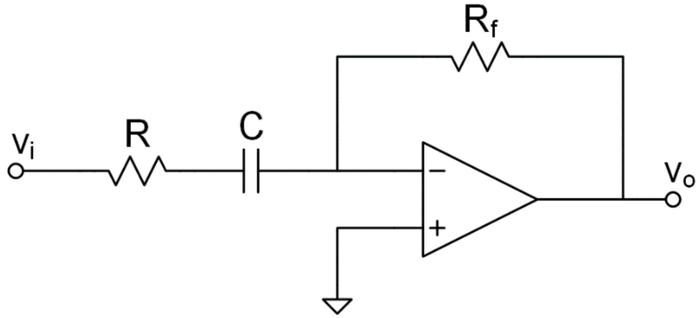
- Gain continues to increase with frequency
- High-frequency gain is very large
  - Any input signal will include some noise
- Better to limit the gain above some upper frequency





Practical Opamp Differentiator

- Problem with ideal differentiator:
  - Low input impedance at high frequency
  - Excessive high-frequency input current

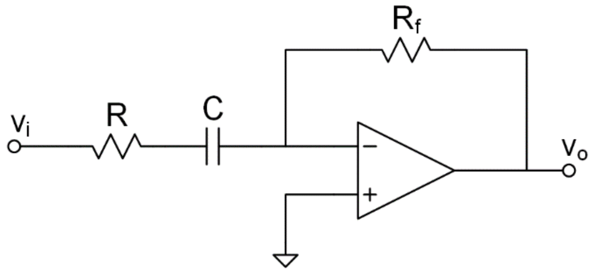


- Add a resistor in series with the input capacitor
  - High-frequency gain limited to R_f/R
- Still behaves as a differentiator at low frequencies
- Behaves as an inverting opamp at high frequencies
- A ***practical*** or ***lossy*** differentiator circuit

Practical Opamp Differentiator

- Transfer function:

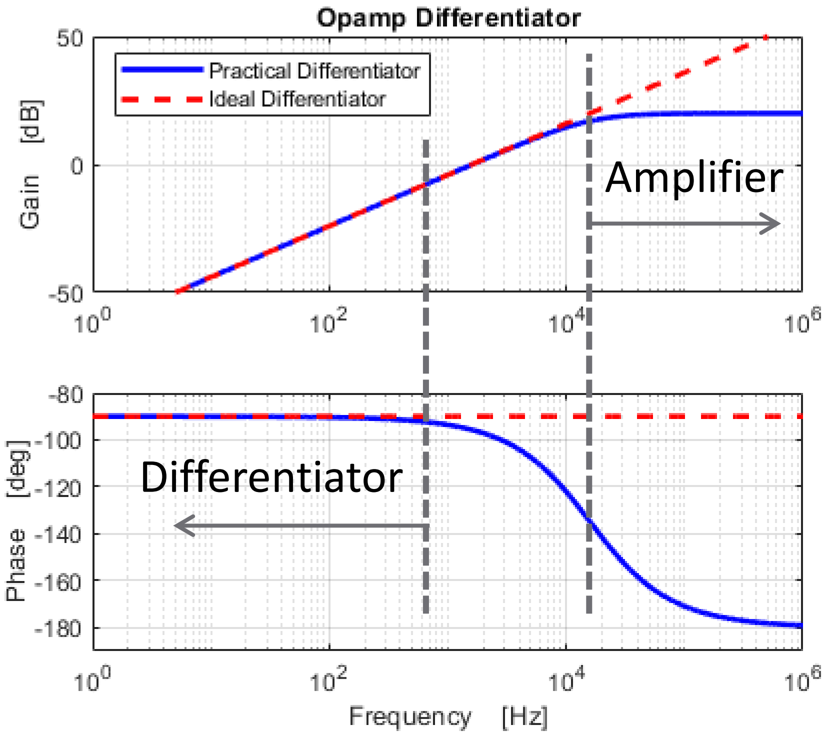
$$G(s) = -\frac{R_f}{R + \frac{1}{Cs}} = -\frac{R_f Cs}{RCs + 1}$$



- Pole (corner frequency) set by the input network:

$$\omega_c = \frac{1}{RC}$$

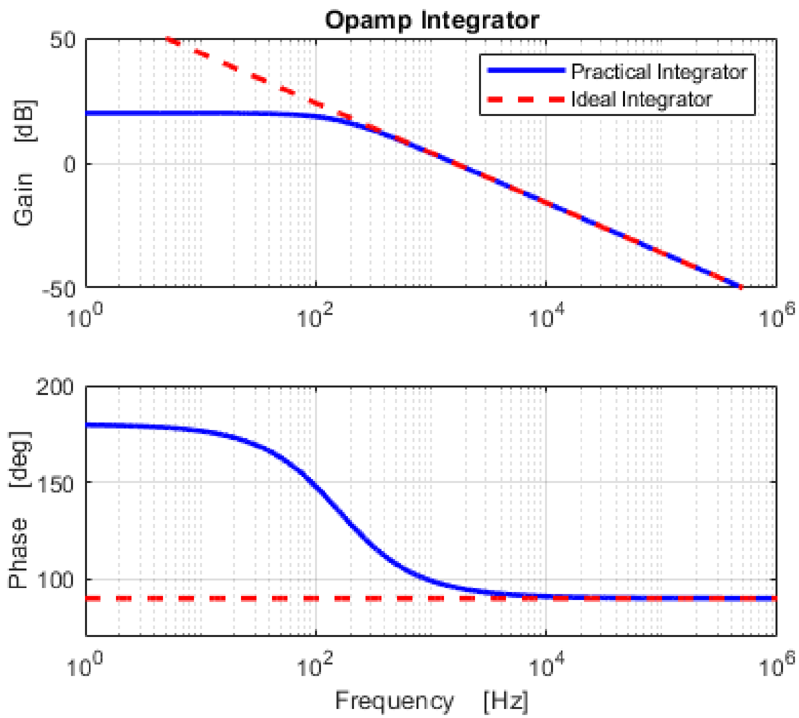
- For $\omega \ll \omega_c$, still behaves like a differentiator
  - Gain: increases at +20 dB/dec
  - Phase: ~-90°

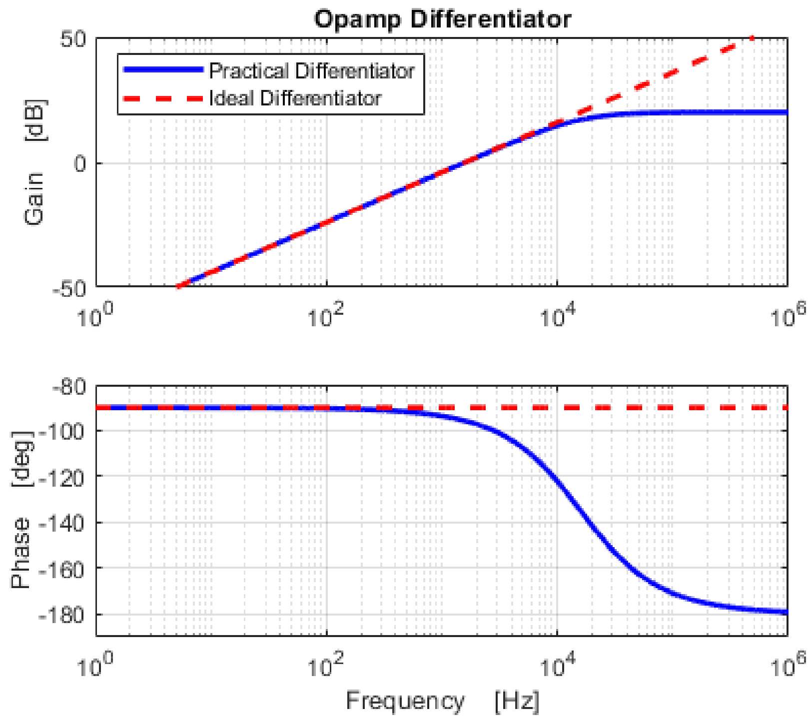


First-Order Opamp Active Filters

First-Order Active Filters

- Practical integrator and differentiator circuits
  - Additional resistors fix problems with ideal circuits
  - First-order low pass and high pass filters



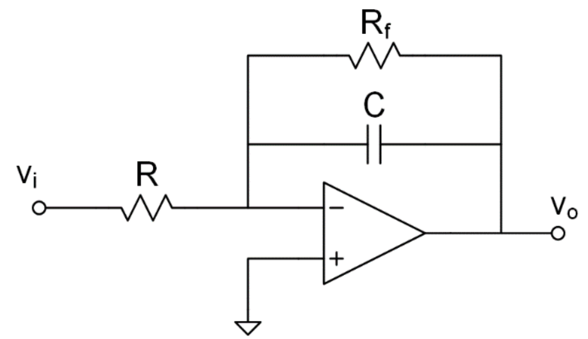


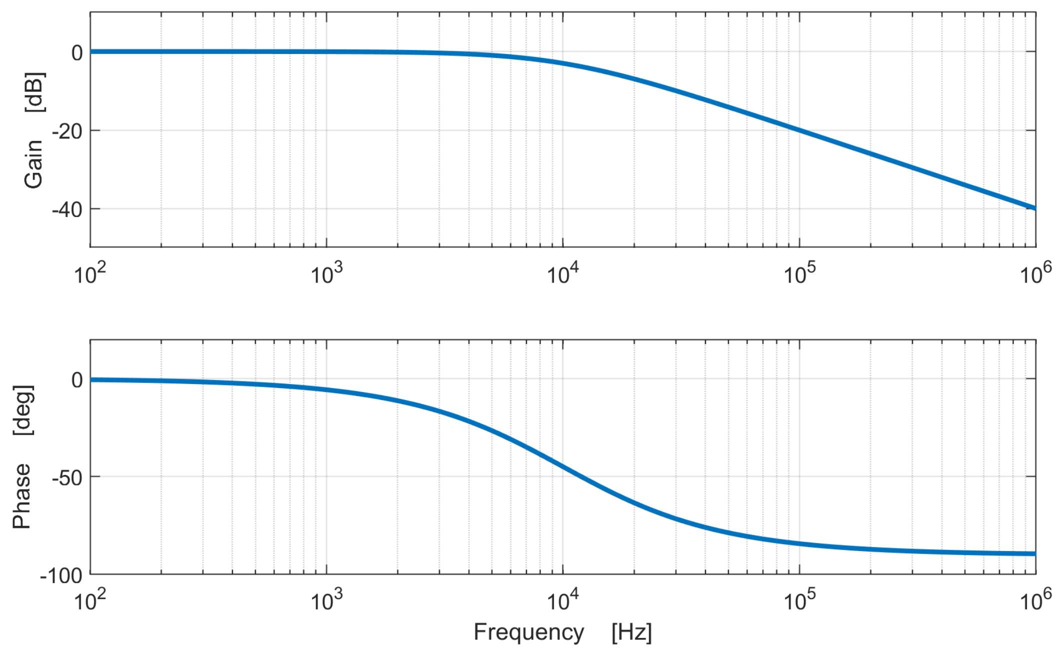
First-Order Low Pass Filter

- Transfer function

$$G(s) = -\frac{R_f}{R}\frac{1}{(R_f C s + 1)}$$

$$G(s) = -\frac{R_f}{R}\frac{\frac{1}{R_f C}}{\left(s + \frac{1}{R_f C}\right)}$$





- Corner frequency

$$f_c = \frac{1}{2\pi R_f C}$$

- Pass-band gain

$$A_v = -\frac{R_f}{R}$$

First-Order High Pass Filter

- Transfer function

$$G(s) = -\frac{R_f C s}{RCs + 1}$$

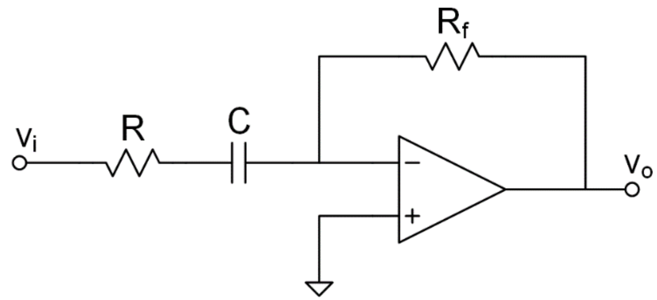
$$G(s) = -\frac{R_f}{R}\frac{s}{\left(s + \frac{1}{RC}\right)}$$

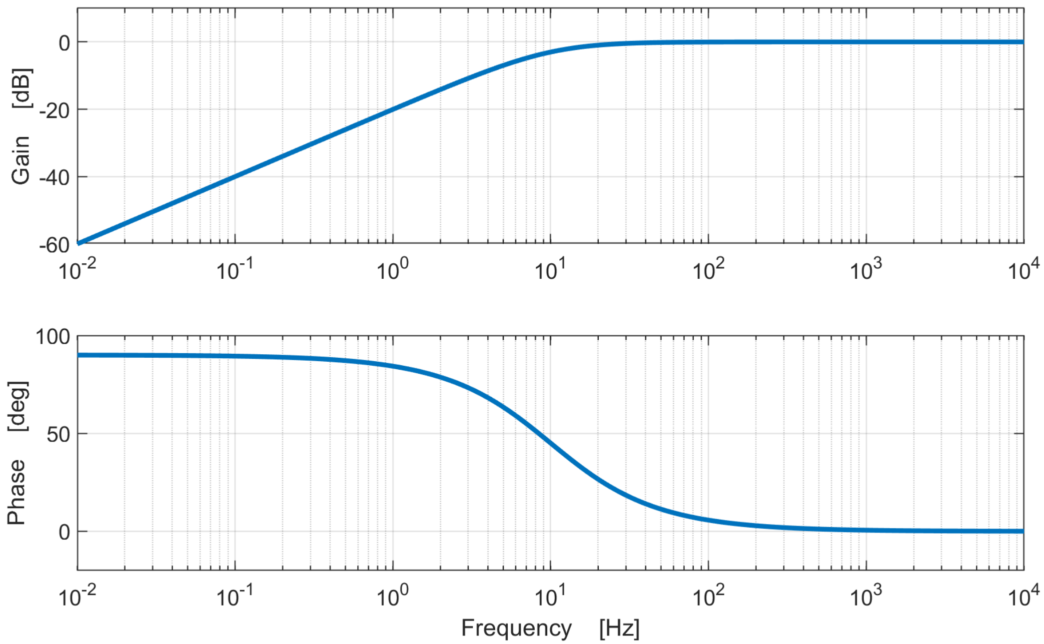
- Corner frequency

$$f_c = \frac{1}{2\pi RC}$$

- Pass-band gain

$$A_v = -\frac{R_f}{R}$$





23

Higher Order Active Filters

Higher-Order Active Filters

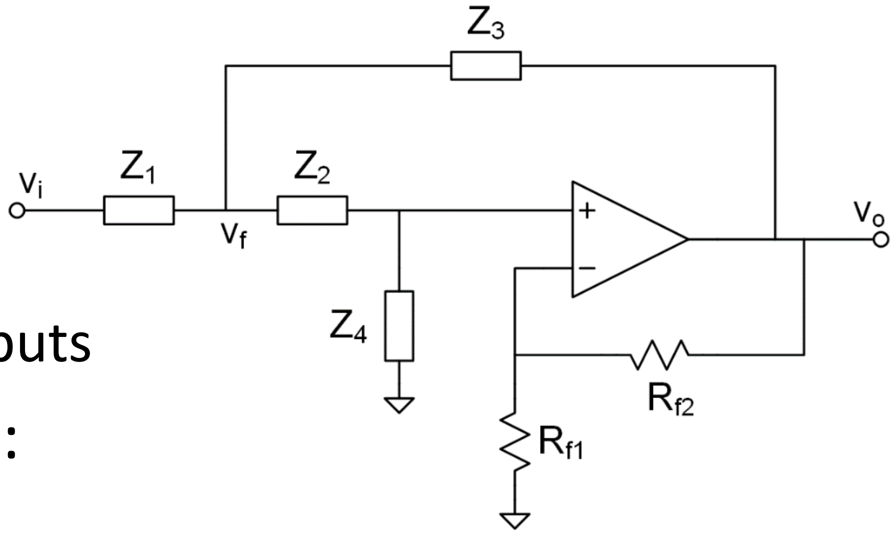
- Higher order active filters can be constructed by:
  - Cascading first-order active filters
  - Using second-order active filter stages
  - Cascading second- and first-order stages
- Create higher order band pass/stop filters similarly:
  - Cascade first-order high/low pass filters
  - Use and/or cascade second-order band pass/stop stages
- Many different second-order active filter topologies
  - We'll look at the ***Sallen-Key circuit***

Sallen-Key Filter – Generalized Form

- Sallen-Key filter topology
  - Low pass and high pass filters
  - Band-pass, and notch filters with slight modifications
- We'll look first at the filter in its most generalized form, then consider the specific low pass and high pass filter forms
- Type of filter depends on the location of components – resistors and capacitors

Sallen-Key Filter – Generalized Form

- Transfer function
  - Nodal analysis
    - KCL at V^+ and V_f
    - Virtual short at opamp inputs
  - After *a lot* of ugly algebra:



$$G(s) = \frac{1}{\beta \frac{Z_1 Z_2}{Z_3 Z_4} + \beta \frac{Z_2}{Z_4} + \beta \frac{Z_1}{Z_4} + (\beta - 1)\frac{Z_1}{Z_3} + \beta}$$

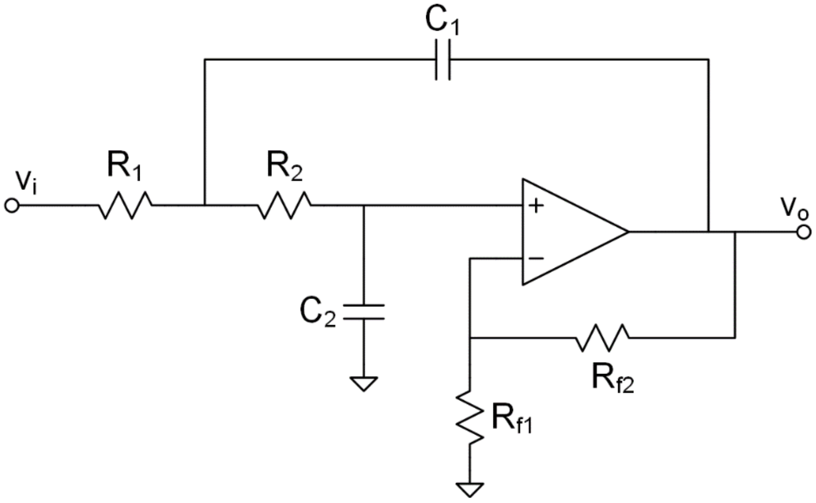
where β is the feedback path gain

$$\beta = \frac{R_{f1}}{R_{f1} + R_{f2}}$$

Sallen-Key Low Pass Filter

Sallen-Key Second-Order Low Pass Filter

- Z_1 and Z_2 are resistors
- Z_3 and Z_4 are capacitors



- Transfer function

$$G(s) = \frac{1}{\beta R_1 R_2 C_1 C_2 s^2 + \beta R_2 C_2 s + \beta R_1 C_2 s + (\beta - 1) R_1 C_1 s + \beta}$$

$$G(s) = \frac{\frac{1}{\beta R_1 R_2 C_1 C_2}}{s^2 + \left(\frac{1}{R_1 C_1} + \frac{1}{R_2 C_1} + \frac{(\beta - 1)}{\beta R_2 C_2}\right) s + \frac{1}{R_1 R_2 C_1 C_2}}$$

Sallen-Key Low Pass Filter

- Generalized second-order low pass transfer function:

$$G(s) = K \cdot \frac{\omega_0^2}{s^2 + \frac{\omega_0}{Q}s + \omega_0^2}$$

- Equating coefficients with the Sallen-Key transfer function gives
  - ***Resonant frequency***:

$$\omega_0 = \frac{1}{\sqrt{R_1 R_2 C_1 C_2}}$$

  - ***Quality factor***:

$$Q = \frac{\sqrt{R_1 R_2 C_1 C_2}}{R_2 C_2 + R_1 C_1 + \left(\frac{\beta - 1}{\beta}\right) R_1 C_1}$$

  - ***DC gain***:

$$K = \frac{1}{\beta} = \frac{R_{f1} + R_{f2}}{R_{f1}}$$

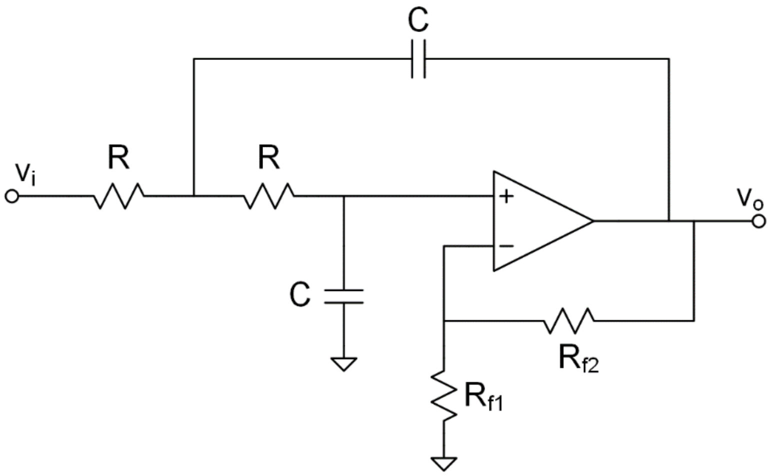
Sallen-Key Low Pass Filter

- ω_0, Q, and gain all set by appropriate component selection, but
  - There are more degrees of freedom than we need
  - Transfer function is a bit more complicated than we'd like

- Simplify by setting component values equal

- Transfer function becomes

$$G(s) = \frac{\frac{1}{\beta(RC)^2}}{s^2 + \left(\frac{3 - \frac{1}{\beta}}{RC}\right)s + \frac{1}{(RC)^2}}$$



  - Where now

$$\omega_0 = \frac{1}{RC} \quad \text{and} \quad Q = \frac{1}{3 - \frac{1}{\beta}}$$

Sallen-Key Low Pass Filter

$$G(s) = \frac{\frac{1}{\beta(RC)^2}}{s^2 + \left(\frac{3 - \frac{1}{\beta}}{RC}\right)s + \frac{1}{(RC)^2}}$$

- We can also write the transfer function in terms of DC gain, K

$$G(s) = \frac{\frac{K}{(RC)^2}}{s^2 + \left(\frac{3 - K}{RC}\right)s + \frac{1}{(RC)^2}}$$

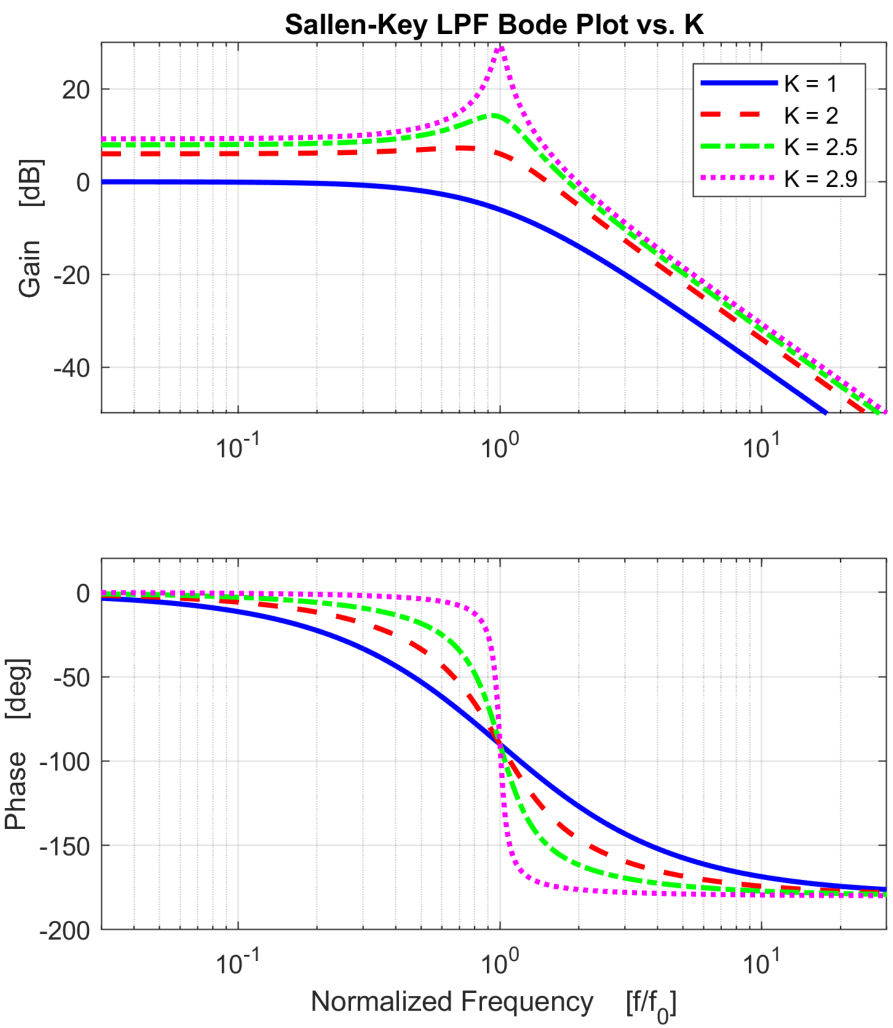
- ω_0 and Q in terms of K:

$$\omega_0 = \frac{1}{RC} \quad \text{and} \quad Q = \frac{1}{3-K}$$

- ***The filter's DC gain is dependent on the filter's Q and vice versa***
  - For independent control of DC gain, cascade an additional gain stage

Sallen-Key Low Pass Filter

- Note dependence of Q and K
  - Both set by feedback path gain
- Q and gain are independent of ω_0
  - ω_0 set by capacitors and resistors at the input
- Second-order
  - Gain roll-off: -40 dB/dec

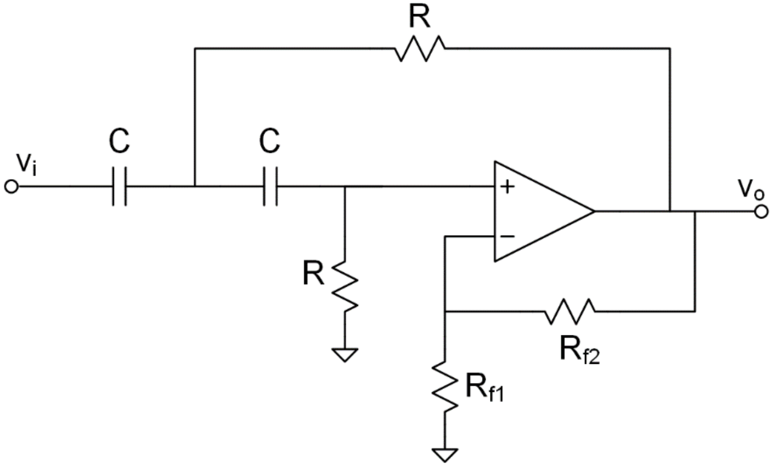


Sallen-Key High Pass Filter

Sallen-Key High Pass Filter

- Here, we will jump straight to the simplified circuit with equal-valued components
- Location of resistors and capacitors swapped relative to low pass filter
- High pass transfer function

$$G(s) = \frac{\frac{1}{\beta}s^2}{s^2 + \left(\frac{3-\frac{1}{\beta}}{RC}\right)s + \frac{1}{(RC)^2}}$$



  - Again,

$$\omega_0 = \frac{1}{RC} \quad \text{and} \quad Q = \frac{1}{3-\frac{1}{\beta}}$$

Sallen-Key High Pass Filter

- As with the low pass filter, we can write the transfer function in terms of gain, K
  - K still represents passband gain, but now it is the high-frequency gain, not the DC gain

$$G(s) = \frac{Ks^2}{s^2 + \left(\frac{3-K}{RC}\right)s + \frac{1}{(RC)^2}}$$

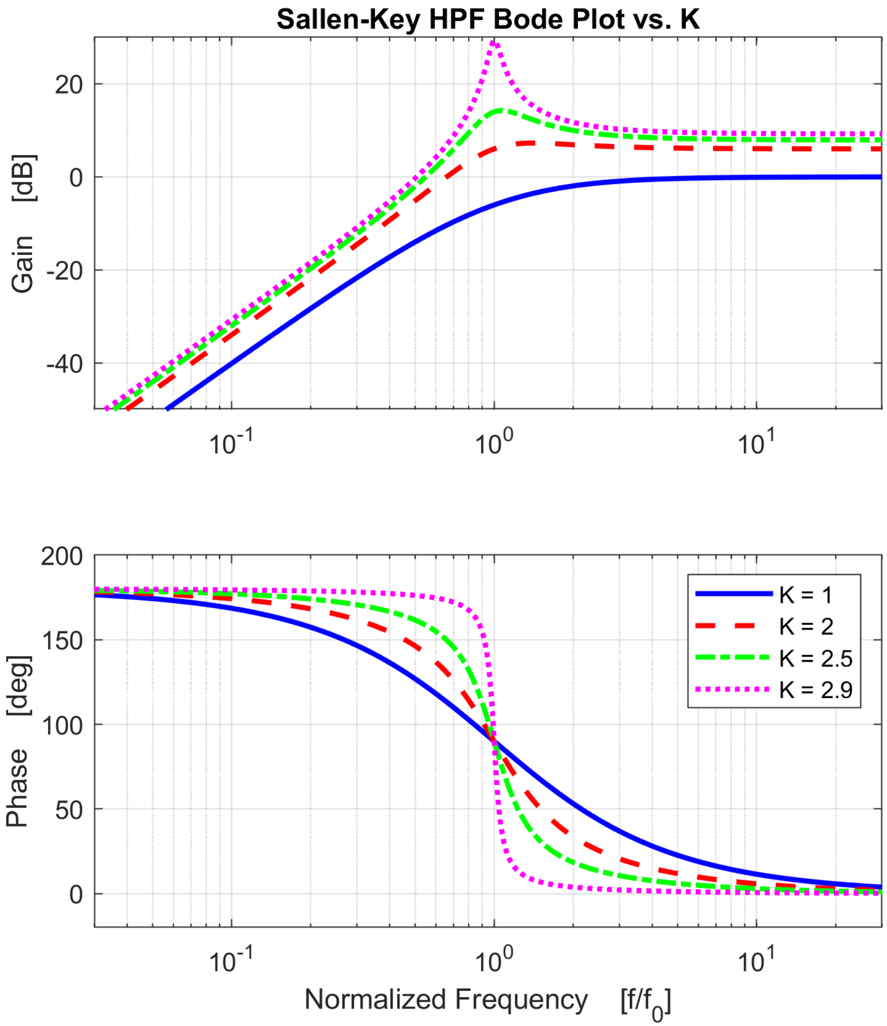
- ω_0 and Q are the same as for the low pass filter:

$$\omega_0 = \frac{1}{RC} \quad \text{and} \quad Q = \frac{1}{3-K}$$

- Same dependence between passband gain, resonant frequency, and Q

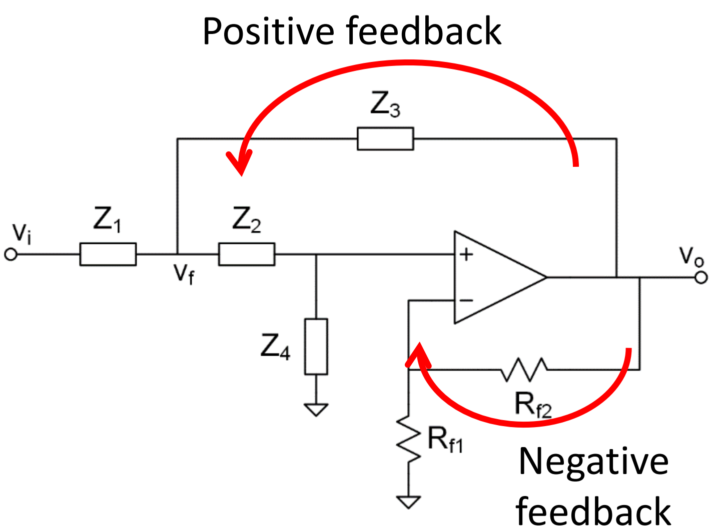
Sallen-Key High Pass Filter

- Note dependence of Q and K
  - Both set by feedback path gain
- Q and gain are independent of ω_0
  - ω_0 set by capacitors and resistors at the input
- Second-order
  - Gain roll-off: -40 dB/dec



Sallen-Key Filter – Stability

- Sallen-Key filter has ***two feedback paths***:
  - ***Negative*** feedback
    - Generally ***stabilizing***
  - ***Positive*** feedback
    - Generally ***destabilizing***

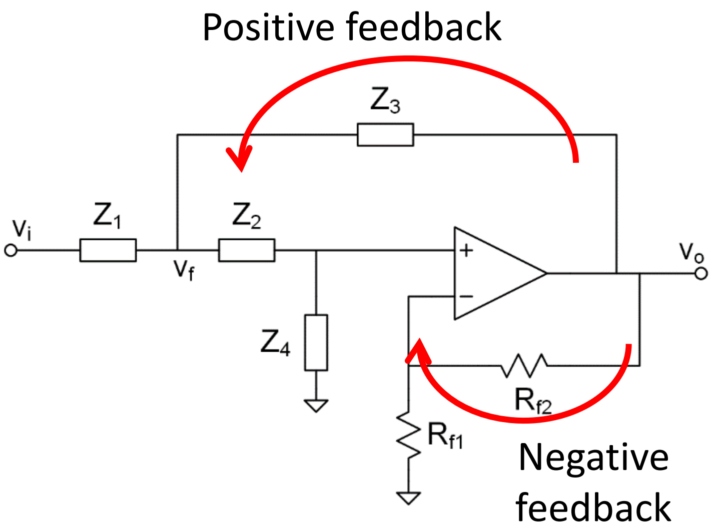


- Relative amount of negative and positive feedback determines stability
  - ***Net negative feedback: circuit is stable***
    - Behaves as a linear filter/amplifier
  - ***Net positive feedback: circuit is unstable***
    - Will oscillate or saturate

Sallen-Key Filter – Stability

- Overall net feedback must remain negative
  - But, we can vary just how negative by varying β
- Varying β allows us to vary Q:

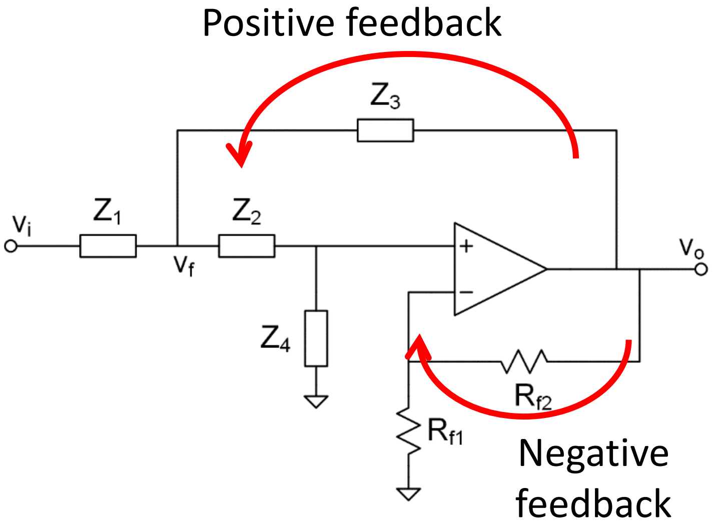
$$Q = \frac{1}{3 - \frac{1}{\beta}} = \frac{1}{3 - K}$$



- As β increases:
  - Negative feedback increases
  - Overall feedback becomes more negative
  - Quality factor, Q, decreases
  - Damping ratio, ζ, increases
  - Pass band gain, K, decreases

Sallen-Key Filter – Stability

- As β decreases:
  - Negative feedback decreases
  - Overall feedback becomes less negative
  - Quality factor, Q, increases
  - Damping ratio, ζ, decreases
  - Pass band gain, K, increases



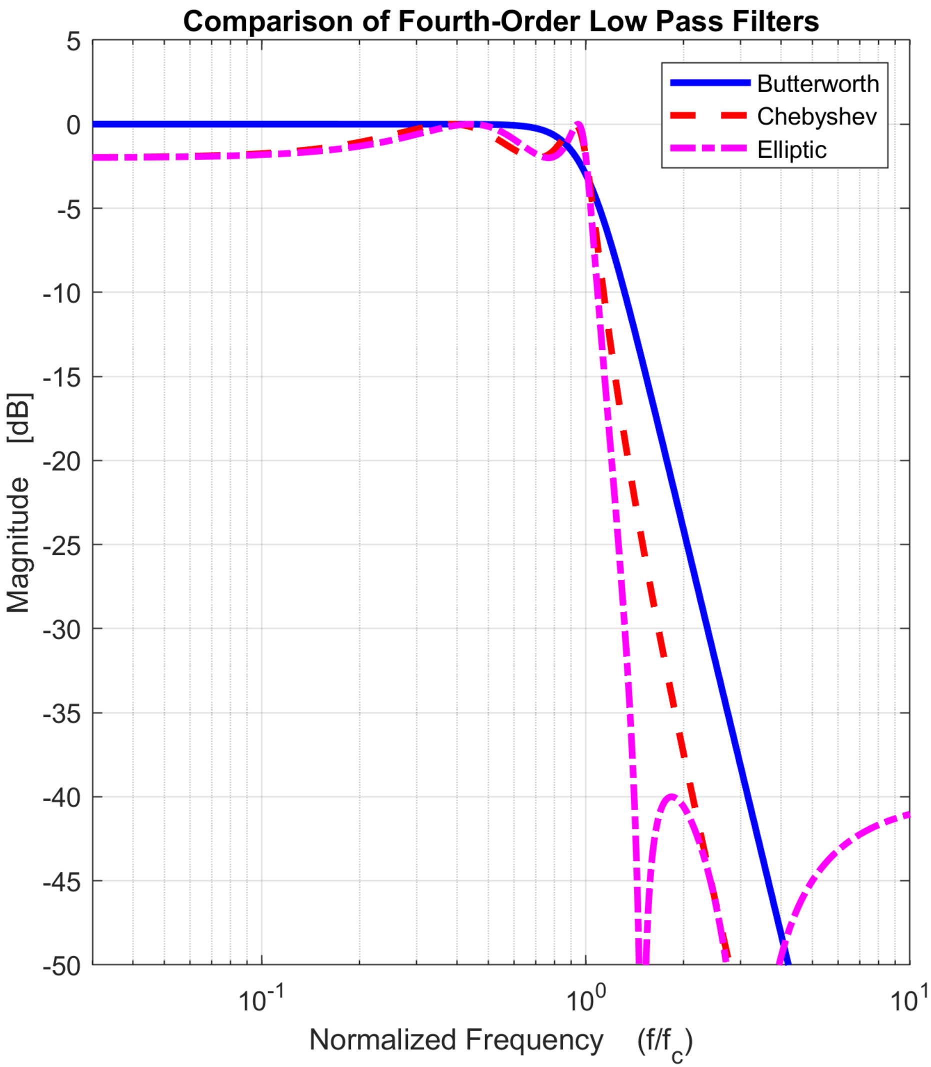
- There is an upper limit on K:
  - For $K = 3$, $Q = \infty$ and $\zeta = 0$
  - An un-damped circuit
  - Negative and positive feedback cancel
  - The border between stability and instability
  - ***For stability***: $K \leq 3$

Filter Families

Filter Families

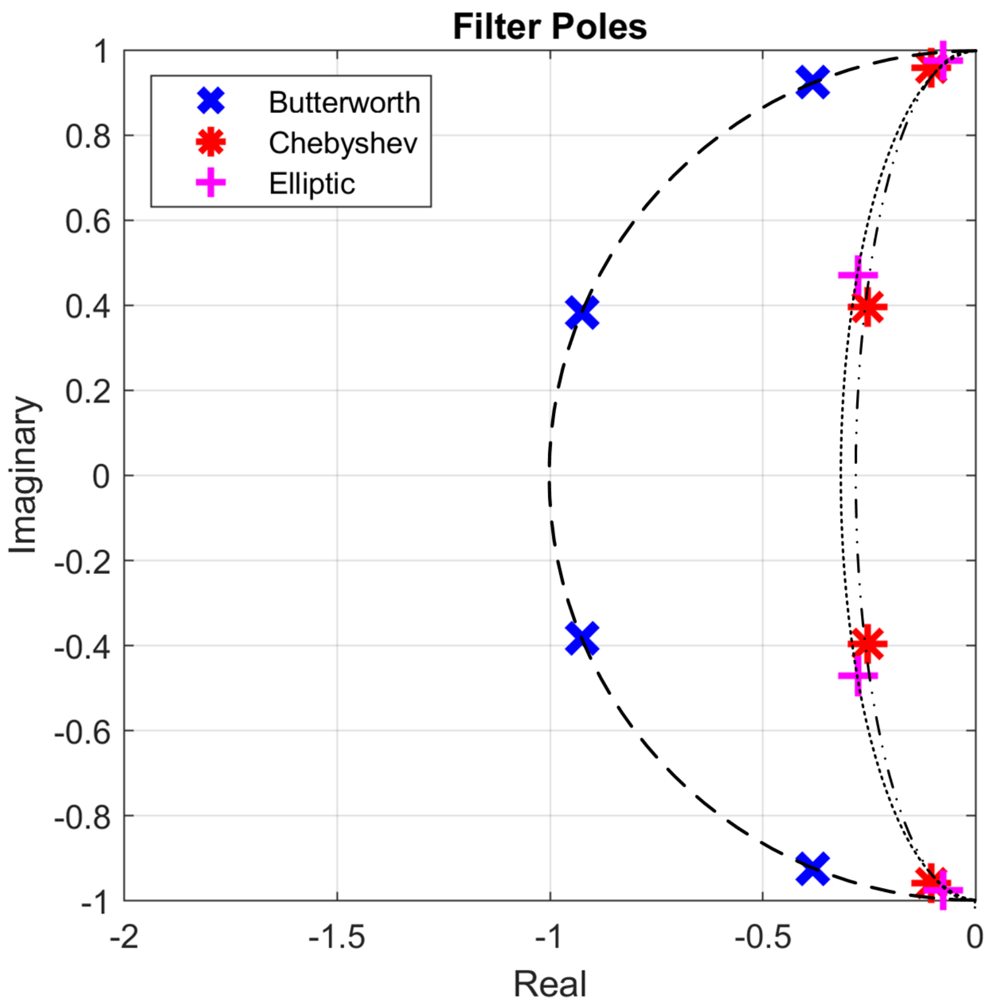
- Higher-order filters of all types can be designed with transfer functions that fit into one of several ***families of filters***
  - Butterworth
  - Chebyshev
  - Elliptic
  - Bessel
- Each filter family defined by the nature of its ***characteristic polynomial***
- Equivalently, each defined by ***pole locations***, e.g.,
  - Butterworth poles lie evenly spaced on a circle in the left half of the complex plane

Filter Families – Frequency Response



- Butterworth
  - Maximally-flat pass band
  - Slow roll off
- Chebyshev
  - Steeper roll off
  - Pass band ripple
- Elliptic
  - Very steep roll off
  - Pass band ripple
  - Stop band ripple
- As always, all about trade offs

Filter Families – System Poles

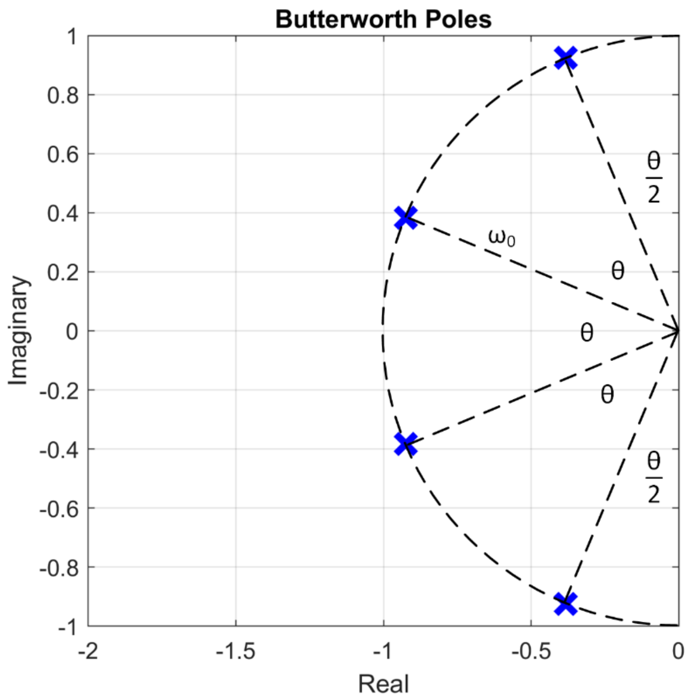


- ***Butterworth***
  - Poles lie on a semi-circle in the LHP
  - Equally spaced
  - Equal magnitude, ω_0
- ***Chebyshev/elliptic***
  - Poles lie on semi-ellipses in the LHP
  - Varying magnitudes

Butterworth Poles

- Butterworth poles:
  - Magnitude: ω_0
  - Order: N
  - Separation angles: $180°/N$
  - Poles for $k = 1 \ldots N$

$$s_k = \omega_0 \left[-\sin\left[\frac{\pi(2k-1)}{2N}\right] + j\cos\left[\frac{\pi(2k-1)}{2N}\right] \right]$$



- Each complex conjugate pair are the poles of a single second-order Sallen-Key stage
  - All with equal ω_0
  - Each with different ζ

45

Filter Synthesis

Filter Synthesis Procedure

1. Determine filter order, N, and cutoff frequency, ω_c
2. Determine ω_0 and Q or ζ for each stage by utilizing either
   a) Design tables, or
   b) MATLAB
3. For each stage, select R and C to yield the required ω_0

$$\omega_0 = \frac{1}{RC}$$

4. For each stage, select R_{f1} and R_{f2} to set gain, K, to provide the required Q

$$K = 3 - \frac{1}{Q}$$

Filter Design Tables

- ***Design tables*** exist for different filters of different orders from different filter families
  - Pole locations, ω_0, and Q given for each second- and first-order stage for a given filter order, N
    - Only second-order stages for even N
    - Second-order plus one first-order stage for odd N
  - Frequencies are normalized
    - Multiply ω_0 by the cutoff frequency, ω_c
    - Multiply σ and ω_d by ω_c

Butterworth Design Table

Order, N	Section	Poles		ω_0	Q	K
		σ	ω_d			
2	1	0.7071	0.7071	1.00	0.7071	1.5858
3	1	0.5000	0.8660	1.00	1.0000	1.0000
	2	1.0000	-	1.00	-	-
4	1	0.9239	0.3827	1.00	0.5412	1.1522
	2	0.3827	0.9239	1.00	1.3065	2.2346
5	1	0.8090	0.5878	1.00	0.6180	1.382
	2	0.3090	0.9511	1.00	1.6182	2.382
	3	1.0000	-	1.00	-	-
6	1	0.9659	0.2588	1.00	0.5176	1.0681
	2	0.7071	0.7071	1.00	0.7071	1.5858
	3	0.2588	0.9659	1.00	1.9319	2.4824

Chebyshev Design Table – 0.5 dB ripple

Order, N	Section	Poles		ω_0	Q	K
		σ	ω_d			
2	1	0.71281	1.004	1.2313	0.8638	1.8422
3	1	0.3123	1.0219	1.0689	1.7062	2.4139
	2	0.6265	-	0.6265	-	-
4	1	0.4233	0.4210	0.5970	0.7051	1.5818
	2	0.1754	1.0163	1.0313	2.9406	2.6599
5	1	0.2931	0.6252	0.6905	1.1778	2.1510
	2	0.1120	1.0116	1.0177	4.5450	2.7800
	3	0.3623	-	0.3623	-	
6	1	0.2898	0.2702	0.3962	0.6836	1.5372
	2	0.2121	0.7382	0.7681	1.8104	2.4476
	3	0.0777	1.0085	1.0114	6.5128	2.8465

Chebyshev Design Table – 1.0 dB ripple

Order, N	Section	Poles		ω_0	Q	K
		σ	ω_d			
2	1	0.5489	0.8951	1.0500	0.9565	1.9545
3	1	0.2471	0.9660	0.9771	2.0177	2.5044
	2	0.4942	-	0.4942	-	-
4	1	0.3369	0.4073	0.5286	0.7846	1.7254
	2	0.1395	0.9834	0.9932	3.5590	2.7190
5	1	0.2342	0.6119	0.6552	1.3988	2.2851
	2	0.0895	0.9901	0.9941	5.5564	2.8200
	3	0.2895	-	0.2895	-	-
6	1	0.2321	0.2662	0.3531	0.7609	1.6857
	2	0.1699	0.7272	0.7468	2.1980	2.5450
	3	0.0622	0.9934	0.9954	8.0037	2.8751

Chebyshev Design Table – 3.0 dB ripple

Order, N	Section	Poles		ω_0	Q	K
		σ	ω_d			
2	1	0.3225	0.7772	0.8414	1.3047	202335
3	1	0.1493	0.9038	0.9161	3.0677	2.6740
	2	0.2986	-	0.2986	-	-
4	1	0.2056	0.3921	0.4427	1.0765	2.0711
	2	0.0852	0.9465	0.9503	5.5789	2.8208
5	1	0.1436	0.5970	0.6140	2.1375	2.5322
	2	0.0549	0.9659	0.9675	8.8178	2.8866
	3	0.1775	-	0.1775	-	-
6	1	0.1427	0.2616	0.2980	1.0443	2.0425
	2	0.1044	0.7148	0.7224	3.4581	2.7108
	3	0.0382	0.9764	0.9772	12.7800	2.9218

Filter synthesis in MATLAB

- MATLAB has built-in filter design functions, e.g.,
  - butter.m
  - cheby1.m
  - ellip.m

- Design procedure:
  1. Use functions to get transfer function coefficients for given filter specifications
  2. Create MATLAB transfer function object
  3. Determine filter poles, ω_0, and Q from transfer function – *place low-Q stages first*
  4. Determine component values from ω_0 and Q

Butterworth Filter – `butter(…)`


```
[b,a] = butter(N,wn,ftype,'s')
```

- Inputs:
  - `N`: filter order
  - `wn`: cutoff frequency [rad/sec]
  - `ftype`: filter type: `'low'`, `'bandpass'`, `'high'`, `'stop'` – optional – default: `'low'`
  - `'s'`: specifies *analog* filter
- Outputs:
  - `b`: coefficients of the transfer function's numerator polynomial
  - `a`: coefficients of the transfer function's denominator polynomial

Chebyshev Filter – cheby1(…)


```
[b,a] = cheby1(N,R,wn,ftype,'s')
```

- Inputs:
  - N: filter order
  - R: pass band ripple [dB]
  - wn: cutoff frequency [rad/sec]
  - ftype: filter type: 'low', 'bandpass', 'high', 'stop' – optional – default: 'low'
  - 's': specifies *analog* filter
- Outputs:
  - b: coefficients of the transfer function's numerator polynomial
  - a: coefficients of the transfer function's denominator polynomial

Elliptic Filter – `ellip(…)`


```
[b,a] = cheby1(N,Rp,Rs,wn,ftype,'s')
```

- Inputs:
  - `N`: filter order
  - `Rp`: pass band ripple [dB]
  - `Rs`: stop band attenuation [dB]
  - `wn`: cutoff frequency [rad/sec]
  - `ftype`: filter type: `'low'`, `'bandpass'`, `'high'`, `'stop'` – optional – default: `'low'`
  - `'s'`: specifies *analog* filter
- Outputs:
  - `b`: coefficients of the transfer function's numerator polynomial
  - `a`: coefficients of the transfer function's denominator polynomial

Transfer Function Model – `tf(…)`


```
sys = tf(b,a)
```

- `b`: vector of numerator polynomial coefficients
- `a`: vector of denominator polynomial coefficients
- `sys`: transfer function model object

- Transfer function is assumed to be of the form

$$G(s) = \frac{b_1 s^r + b_2 s^{r-1} + \cdots + b_r s + b_{r+1}}{a_1 s^n + a_2 s^{n-1} + \cdots + a_n s + a_{n+1}}$$

- Inputs to `tf(…)` are
  - `Num = [b1,b2,…,br+1];`
  - `Den = [a1,a2,…,an+1];`

Getting ω_0 and Q – `damp(…)`


```
[wn,zeta,p] = damp(sys)
```

- `sys`: transfer function system model object
- `wn`: vector of natural frequencies (magnitudes) of poles
- `zeta`: vector of damping ratios, ζ, of poles
- `p`: vector of poles

- Use `wn` values for ω_0 of each filter stage
- Calculate Q of each stage from ζ values

$$Q = \frac{1}{2\zeta}$$

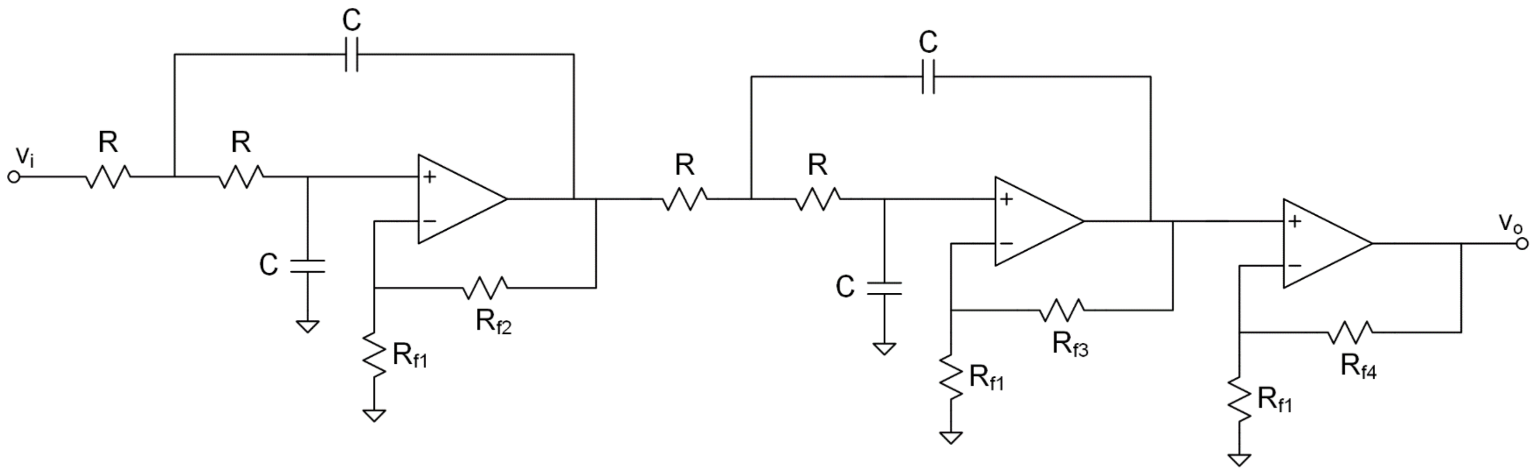
Filter Design Example

Sallen-Key Filter – Example

- Design a Butterworth (maximally-flat) low pass active filter to satisfy the following specifications:
  - **Corner frequency**: f_c = 1MHz
  - **Frequency response roll off** beyond f_c: 80dB/dec
  - **Pass band (DC) gain**: 12dB (4)
- Roll off spec of 80 dB/dec tells us we need a fourth-order filter – cascade two Sallen-Key stages
- Add a constant gain stage if necessary to meet gain specification

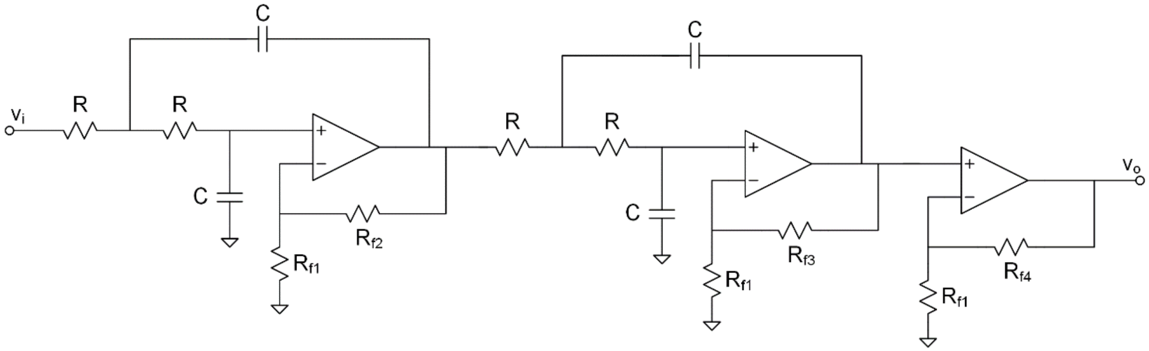
Sallen-Key Filter – Example

- Fourth-order filter
  - Cascade two second-order Sallen-Key stages
- Additional gain stage necessary to meet gain specification
  - Non-inverting opamp amplifier



- Note that the circuit in this example has been simplified by setting R_{f1} equal in each stage
  - Not necessarily the right choice

Sallen-Key Filter – Example



- Butterworth filter, so, for both stages,

$$\omega_0 = \omega_c = 2\pi \cdot f_c = 2\pi \cdot 1\ MHz$$

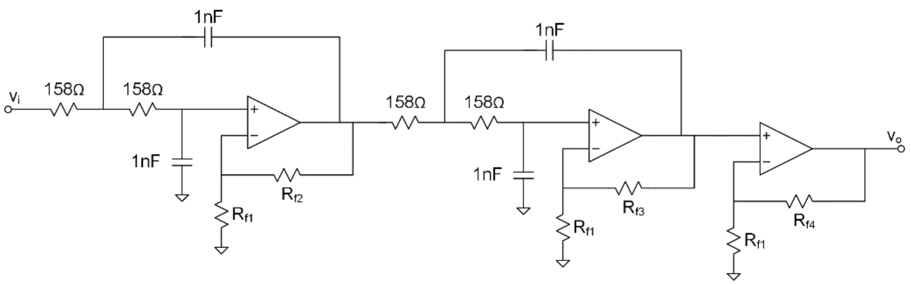
- Determine R and C for desired ω_c
  - Arbitrarily choose $C = 1\ nF$

$$R = \frac{1}{2\pi f_c C} = \frac{1}{2\pi \cdot 1\ MHz \cdot 1\ nF} = 159\ \Omega$$

  - If using $\pm 1\%$ resistors, 158 Ω is a standard value

$$\boxed{R = 158\ \Omega \quad \text{and} \quad C = 1\ nF}$$

Sallen-Key Filter – Example



- To determine gain of each stage, consult the Butterworth design table

Order, N	Section	Poles		ω_0	Q
		σ	ω_d		
4	1	0.9239	0.3827	1.00	0.5412
	2	0.3827	0.9239	1.00	1.3065

- Calculate K for each stage from its Q

$$K_1 = 3 - \frac{1}{Q_1} = 3 - \frac{1}{0.5412} = 1.152$$

$$K_2 = 3 - \frac{1}{Q_2} = 3 - \frac{1}{1.3065} = 2.235$$

Sallen-Key Filter – Example

- Alternatively, use MATLAB to determine ω_0 and K values for each stage

```
%% filter specs
N = 4;
fc = 1e6;
wc = fc*(2*pi);

% design a butterworth filter
[num,den] = butter(N,wc,'s');
Gb = tf(num,den);

% get poles and corresponding magnitudes and damping
[w0, zeta, p] = damp(Gb);

Q = 1./(2*zeta);
K = 3 - 1./Q;

filt_params = table(Q, w0, K);

display(filt_params)
```

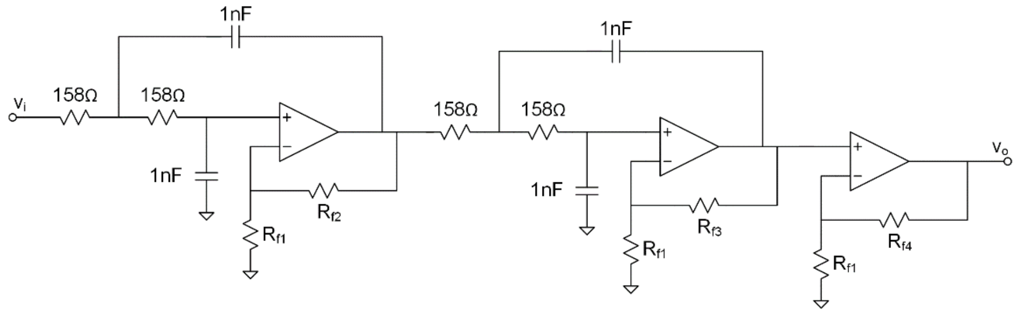
```
filt_params =

  4×3 table
```

Q	w0	K
0.5412	6.2832e+06	1.1522
0.5412	6.2832e+06	1.1522
1.3066	6.2832e+06	2.2346
1.3066	6.2832e+06	2.2346

```
>>
```

- Note that we would put the low-Q stage first

Sallen-Key Filter – Example



- Arbitrarily choose $R_{f1} = 5.11\ k\Omega$
- Calculate R_{f2} and R_{f3} to give the required K_1 and K_2

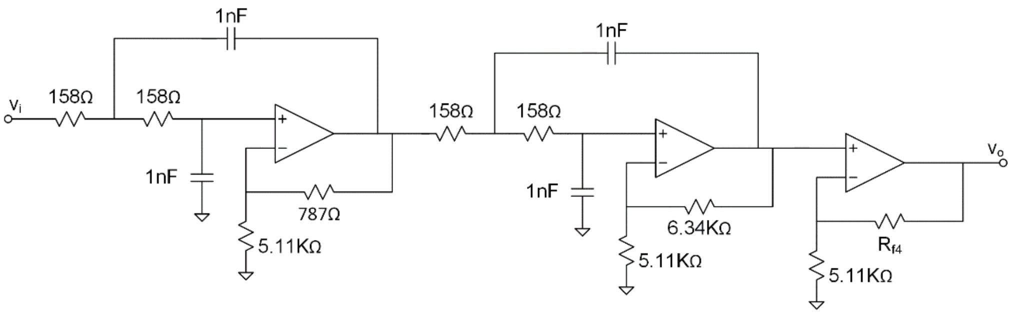
$$K_1 = \frac{R_{f1} + R_{f2}}{R_{f1}} \quad \rightarrow \quad R_{f2} = R_{f1}(K_1 - 1) = 5.11\ k\Omega \cdot 0.152 = 778\ \Omega$$

$$K_2 = \frac{R_{f1} + R_{f3}}{R_{f1}} \quad \rightarrow \quad R_{f3} = R_{f1}(K_2 - 1) = 5.11\ k\Omega \cdot 1.235 = 6.31\ k\Omega$$

- Again, assuming $\pm 1\%$ resistors, we choose the closest standard values:

$$R_{f2} = 787\ \Omega \quad \text{and} \quad R_{f3} = 6.34\ k\Omega$$

Sallen-Key Filter – Example



- Finally, set the gain of the third stage to satisfy the gain requirement
- Overall gain given by

$$K = K_1 K_2 K_3 = 4 \quad \rightarrow \quad K_3 = \frac{4}{K_1 K_2} = \frac{4}{1.152 \cdot 2.235} = 1.554$$

- Calculate R_{f4} to give the required K_3

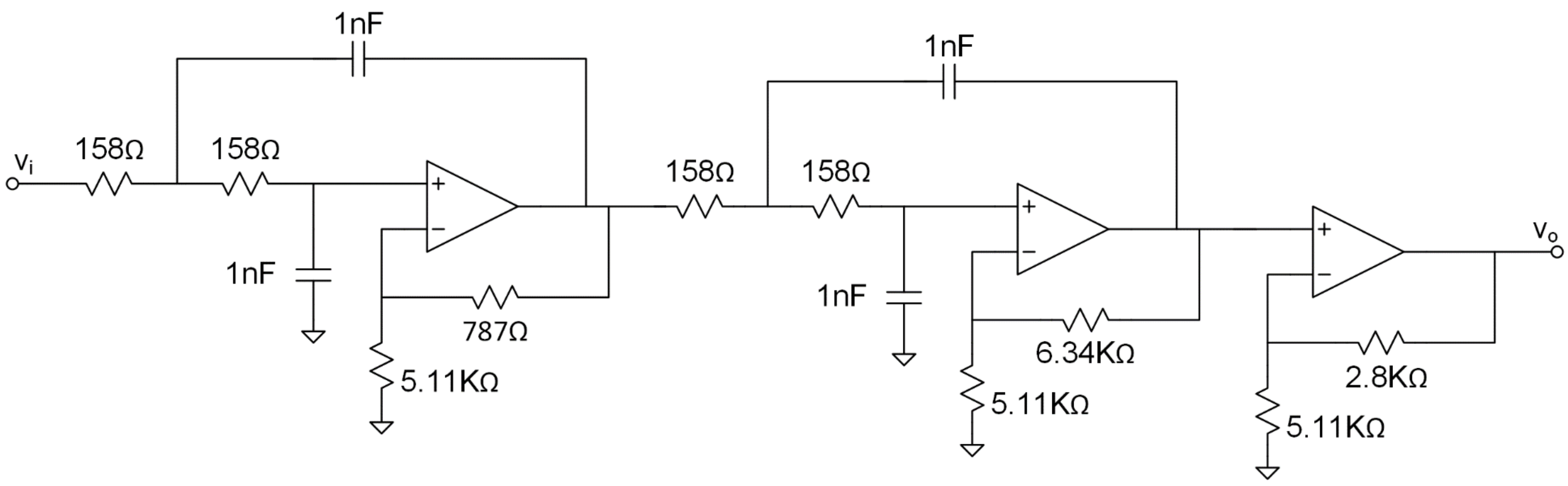
$$K_3 = \frac{R_{f1} + R_{f4}}{R_{f1}} \quad \rightarrow \quad R_{f4} = R_{f1}(K_3 - 1) = 5.11\ k\Omega \cdot 0.554 = 2.83\ k\Omega$$

- Again, assuming $\pm 1\%$ resistors, we choose the closest standard value:

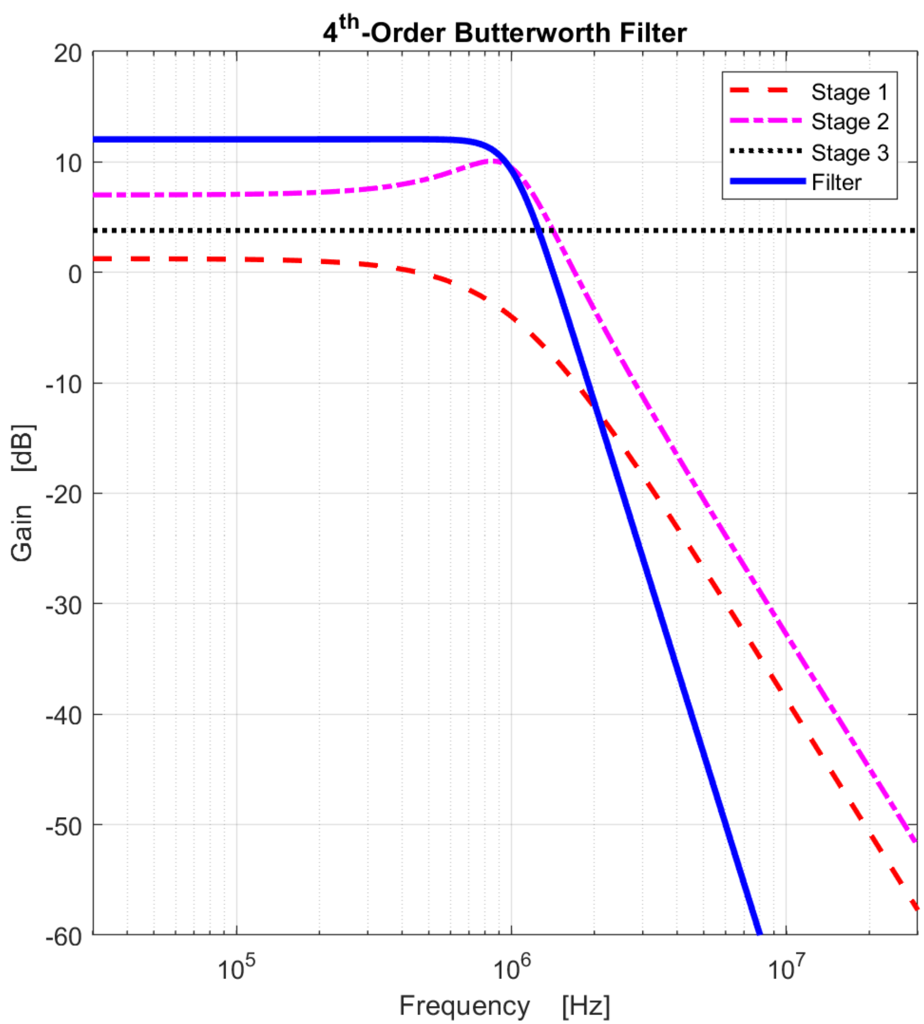
$$\boxed{R_{f4} = 2.8\ k\Omega}$$

Sallen-Key Filter – Example

The complete 4th-order Sallen-Key Butterworth low pass filter:



Sallen-Key Filter – Example



- DC gain: ~12 dB
- $f_c \approx 1\ MHz$
- Gain rolloff: -80 dB/dec
- Stage 1:
  - Low Q
  - Low gain
- Stage 2:
  - Higher Q
  - Higher gain
- Stage 3:
  - Constant gain