Second Order Filters Overview

- What's different about second order filters
- Resonance
- Standard forms
- Frequency response and Bode plots
- Sallen-Key filters
- General transfer function synthesis

Prerequisites and New Knowledge

Prerequisite knowledge

- Ability to perform Laplace transform circuit analysis
- Ability to solve for the transfer function of a circuit
- Ability to generate and interpret Bode plots

New knowledge

- Ability to design second-order filters
- Knowledge of the advantages of transfer functions with complex poles and zeros
- Understanding of resonance
- Familiarity with second-order filter properties

Motivation

$$Y(s) = \left(\frac{\sum_{k=0}^{M} b_k s^k}{\sum_{k=0}^{N} a_k s^k} \right) X(s) = H(s)X(s)$$

- Second order filters are filters with a denominator polynomial that is of second order
- Usually the roots (poles) are complex

$$p_{1,2} = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2}$$

- In many applications, second order filters may be sufficient
- Are also the building blocks for higher-order filters

Standard Form

$$H(s) = \frac{N(s)}{\frac{s^2}{\omega_n^2} + \frac{s}{Q\omega_n} + 1}$$

- Second order filters can be designed as lowpass, highpass, bandpass, or notch filters
- All four types can be expressed in **standard form** shown above
- $N(s)$ is a polynomial in s of degree $m \leq 2$
  - If $N(s) = k$, the system is a lowpass filter with a DC gain of k
  - If $N(s) = k\frac{s^2}{\omega_n^2}$, the system is a highpass filter with a high frequency gain of k
  - If $N(s) = k\frac{s}{Q\omega_n}$, the system is a bandpass filter with a maximum gain of k
  - If $N(s) = k\left(1 - \frac{s^2}{\omega_n^2}\right)$, the system is a notch filter with a gain of k

Unity Gain Lowpass

A unity-gain lowpass second-order transfer function is of the form

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{1}{1 + 2\zeta\frac{s}{\omega_n} + \left(\frac{s}{\omega_n}\right)^2}$$

- ω_n is called the **undamped natural frequency**
- ζ (zeta) is called the **damping ratio**
- The poles are $p_{1,2} = (-\zeta \pm \sqrt{\zeta^2 - 1})\,\omega_n$
- If $\zeta \geq 1$, the poles are real
- If $0 < \zeta < 1$, the poles are complex
- If $\zeta = 0$, the poles are imaginary: $p_{1,2} = \pm j\omega_n$
- If $\zeta < 0$, the poles are in the right half plane ($\mathrm{Re}\{p\} > 0$) and the system is unstable

Unity Gain Lowpass Continued

The transfer function $H(s)$ can also be expressed in the following form

$$H(s) = \frac{1}{1 + 2\zeta\frac{s}{\omega_n} + \left(\frac{s}{\omega_n}\right)^2} = \frac{1}{1 + \frac{s}{Q\omega_n} + \left(\frac{s}{\omega_n}\right)^2}$$

where

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

The meaning of Q, the **Quality factor**, will become clear in the following slides.

$$H(j\omega) = \frac{1}{1 - \left(\frac{\omega}{\omega_n}\right)^2 + \frac{j\omega}{Q\omega_n}}$$

Unity Gain Lowpass Magnitude Response

$$
\begin{aligned}
20\log_{10}|H(j\omega)| &= 20\log_{10}\left|1-\left(\frac{\omega}{\omega_n}\right)^2+\frac{j\omega}{Q\omega_n}\right|^{-1} \\
&= -20\log_{10}\sqrt{\left(1-\frac{\omega^2}{\omega_n^2}\right)^2+\left(\frac{\omega}{Q\omega_n}\right)^2}
\end{aligned}
$$

For $\omega \ll \omega_n$,

$$
20\log_{10}|H(j\omega)| \approx -20\log_{10}|1| = 0 \text{ dB}
$$

For $\omega \gg \omega_n$,

$$
20\log_{10}|H(j\omega)| \approx -20\log_{10}\frac{\omega^2}{\omega_n^2} = -40\log_{10}\frac{\omega}{\omega_n} \text{ dB}
$$

At these extremes, the behavior is identical to two real poles.

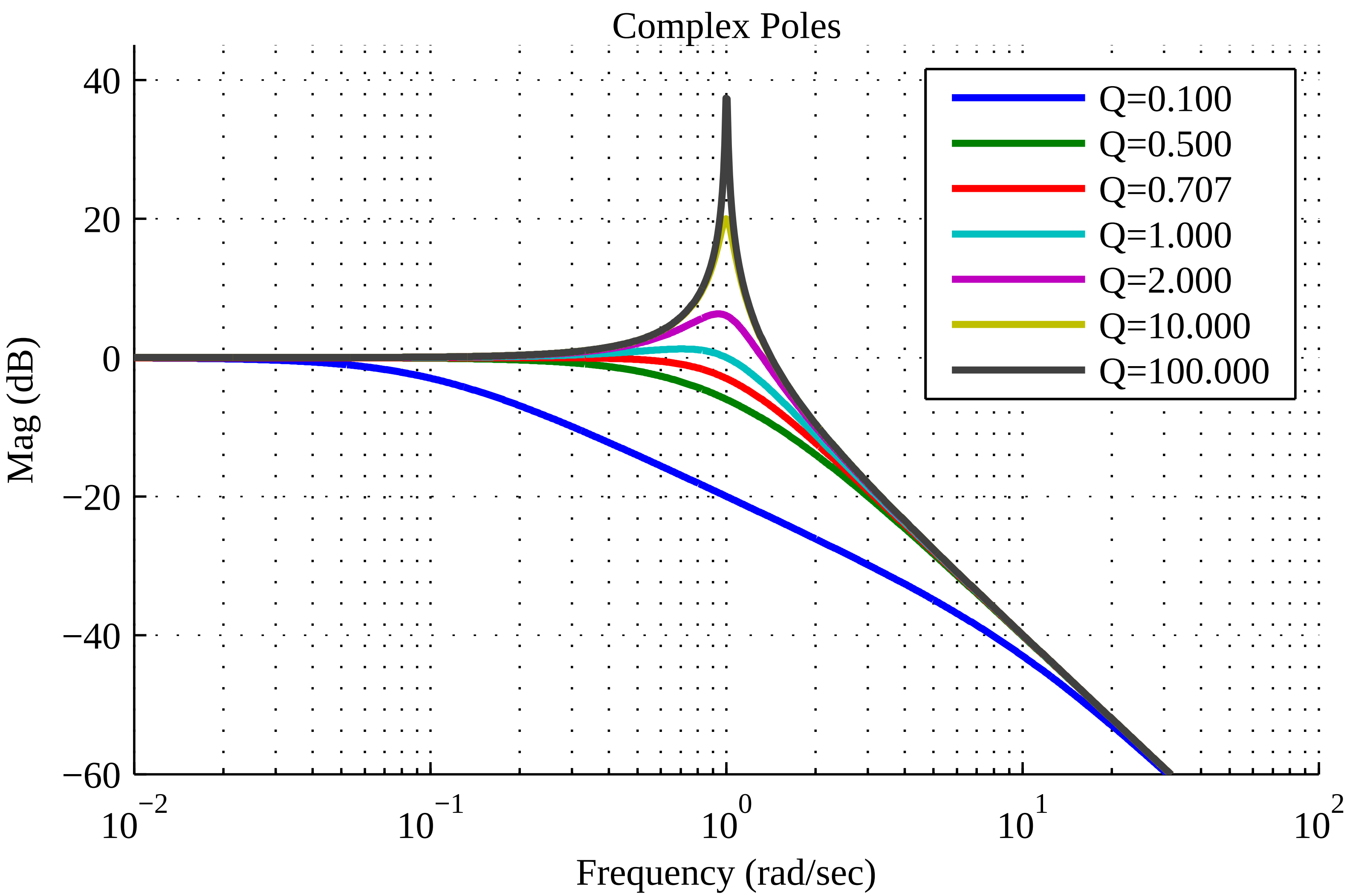
Lowpass Magnitude Continued

$$20\log_{10}|H(j\omega)| = -20\log_{10}\sqrt{\left(1-\frac{\omega^2}{\omega_n^2}\right)^2 + \left(\frac{\omega}{Q\omega_n}\right)^2}$$

For $\omega = \omega_n$,

$$20\log_{10}|H(j\omega_n)| = -20\log_{10}\frac{1}{Q} = 20\log_{10} Q = Q_{\mathrm{dB}}$$

Lowpass Magnitude



Matlab Code

```
function [] = BodeComplexPoles();

N  = 1000;
wn = 1;
Q  = [0.1 0.5 sqrt(1/2) 1 2 10 100];

w = logspace(-2,2,N);

nQ = length(Q);
mag = [];
phs = [];

legendLabels = cell(nQ,1);
for c1 = 1:nQ
    sys = tf([wn^2],[1 wn/Q(c1) wn^2]);
    [m, p] = bode(sys,w);
    mag = [mag reshape(m,N,1)];
    phs = [phs reshape(p,N,1)];
    legendLabels{c1} = sprintf('Q=%5.3f',Q(c1));
end

figure(1);
FigureSet(1,'Slides');
h = semilogx(w,20*log10(mag));
set(h,'LineWidth',1.5);
grid on;
xlabel('Frequency (rad/sec)');
```

```
ylabel('Mag (dB)');
title('Complex Poles');
xlim([w(1) w(end)]);
ylim([-60 45]);
box off;
AxisSet(8);
legend(legendLabels);
print('BodeMagComplexPoles','-depsc');

figure(2);
FigureSet(2,'Slides');
h = semilogx(w,phs);
set(h,'LineWidth',1.5);
grid on;
xlabel('Frequency (rad/sec)');
ylabel('Phase (deg)');
title('Complex Poles');
xlim([w(1) w(end)]);
ylim([-190 10]);
box off;
AxisSet(8);
legend(legendLabels);

print('BodePhaseComplexPoles','-depsc');
```

Lowpass Phase

$$\angle H(j\omega) \quad = \quad \angle \frac{1}{1 - \left(\frac{\omega}{\omega_n}\right)^2 + \frac{j\omega}{Q\omega_n}}$$

For $\omega \ll \omega_n$,

$$\angle H(j\omega) \approx \angle 1 = 0^\circ$$

For $\omega \gg \omega_n$,

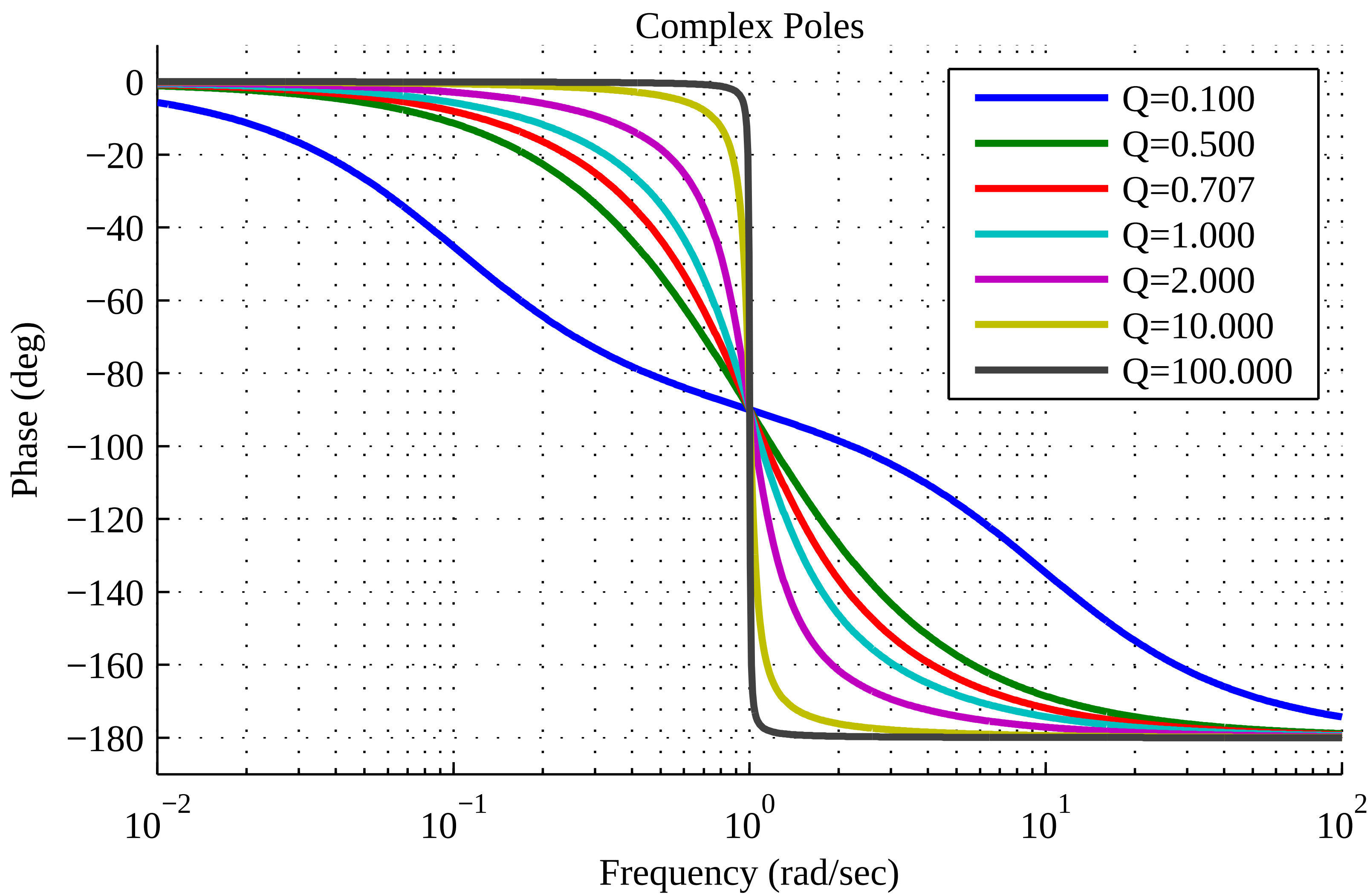
$$\angle H(j\omega) \approx \angle \frac{1}{-\frac{\omega}{Q\omega_n}} = \angle - \frac{Q\omega_n}{\omega} = \angle -1 = -180^\circ$$

For $\omega = \omega_n$,

$$\angle H(j\omega) \approx \angle \frac{1}{Qj} = \angle \frac{1}{j} = \angle -j = -90^\circ$$

At the extremes, the behavior is identical to two real poles. At other values of ω near ω_n, the behavior is more complicated.

Unity Gain Lowpass Phase



Lowpass Comments

$$H(j\omega) = \frac{1}{1 + \left(\frac{j\omega}{Q\omega_n}\right) + \left(\frac{j\omega}{\omega_n}\right)^2}$$

- The second-order response attenuates twice as fast as a first-order response (40 dB/decade)
- Generally better than the cascade of two first-order filters
- Offers additional degree of freedom (Q), which is the gain near $\omega = \omega_n$
- Q may range from 0.5 to 100
- Is usually near $Q = 1$

Lowpass Maximum Magnitude

What is the frequency at which $|H(j\omega)|$ is maximized?

$$H(j\omega) = \frac{1}{1 + \left(\frac{j\omega}{Q\omega_n}\right) + \left(\frac{j\omega}{\omega_n}\right)^2} \qquad |H(j\omega)| = \frac{1}{\sqrt{\left(1 - \frac{\omega^2}{\omega_n^2}\right)^2 + \left(\frac{\omega}{Q\omega_n}\right)^2}}$$

- For high values of Q, the maximum of $|H(j\omega)| > 1$
- This is called **peaking**
- The largest Q before the onset of peaking is $Q = \frac{1}{\sqrt{2}} \approx 0.707$
  - Said to be **maximally flat**
  - Also called a **Butterworth response**
  - In this case, $|H(j\omega_n)| = -3$ dB and ω_n is the cutoff frequency

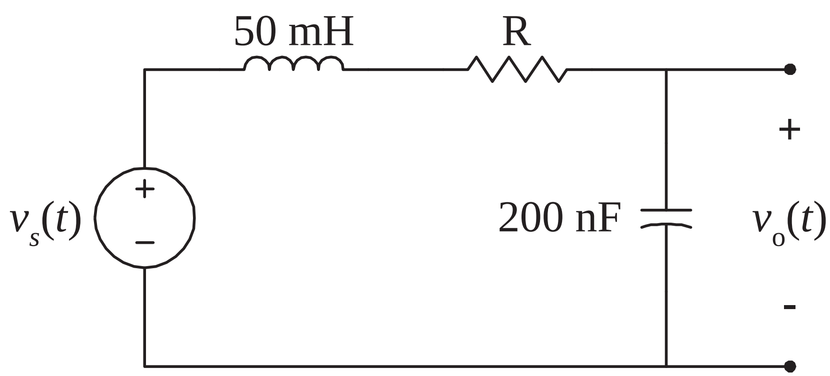
Lowpass Maximum Continued

If $Q > 0.707$, the maximum magnitude and frequency are as follows:

$$\omega_r = \omega_n \sqrt{1 - \frac{1}{2Q^2}} \qquad |H(j\omega_r)| = \frac{Q}{1 - \frac{1}{4Q^2}}$$

- ω_r is called the **resonant frequency** or the **damped natural frequency**
- As $Q \rightarrow \infty$, $\omega_r \rightarrow \omega_n$
- For sufficiently large Q (say $Q > 5$)
  - $\omega_r \approx \omega_n$
  - $|H(j\omega_r)| \approx Q$
- Peaked responses are useful in the synthesis of high-order filters

Example 1: Passive Lowpass



Generate the bode plot for the circuit shown above.

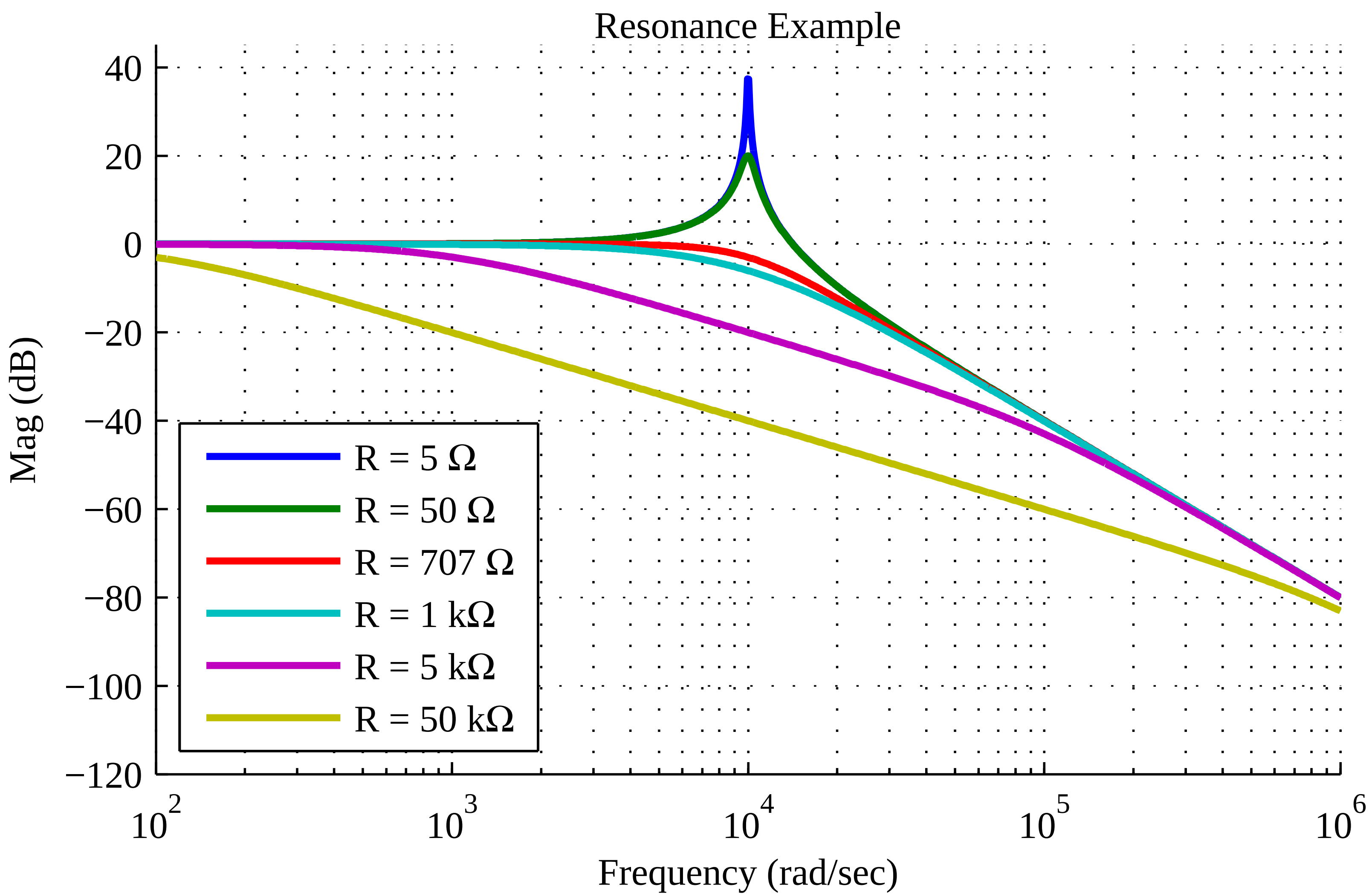
$$H(s) = \frac{\frac{1}{LC}}{s^2 + \frac{R}{L}s + \frac{1}{LC}} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{\omega_n^2}{s^2 + \frac{\omega_n}{Q}s + \omega_n^2}$$

Example 1:Passive Lowpass Continued

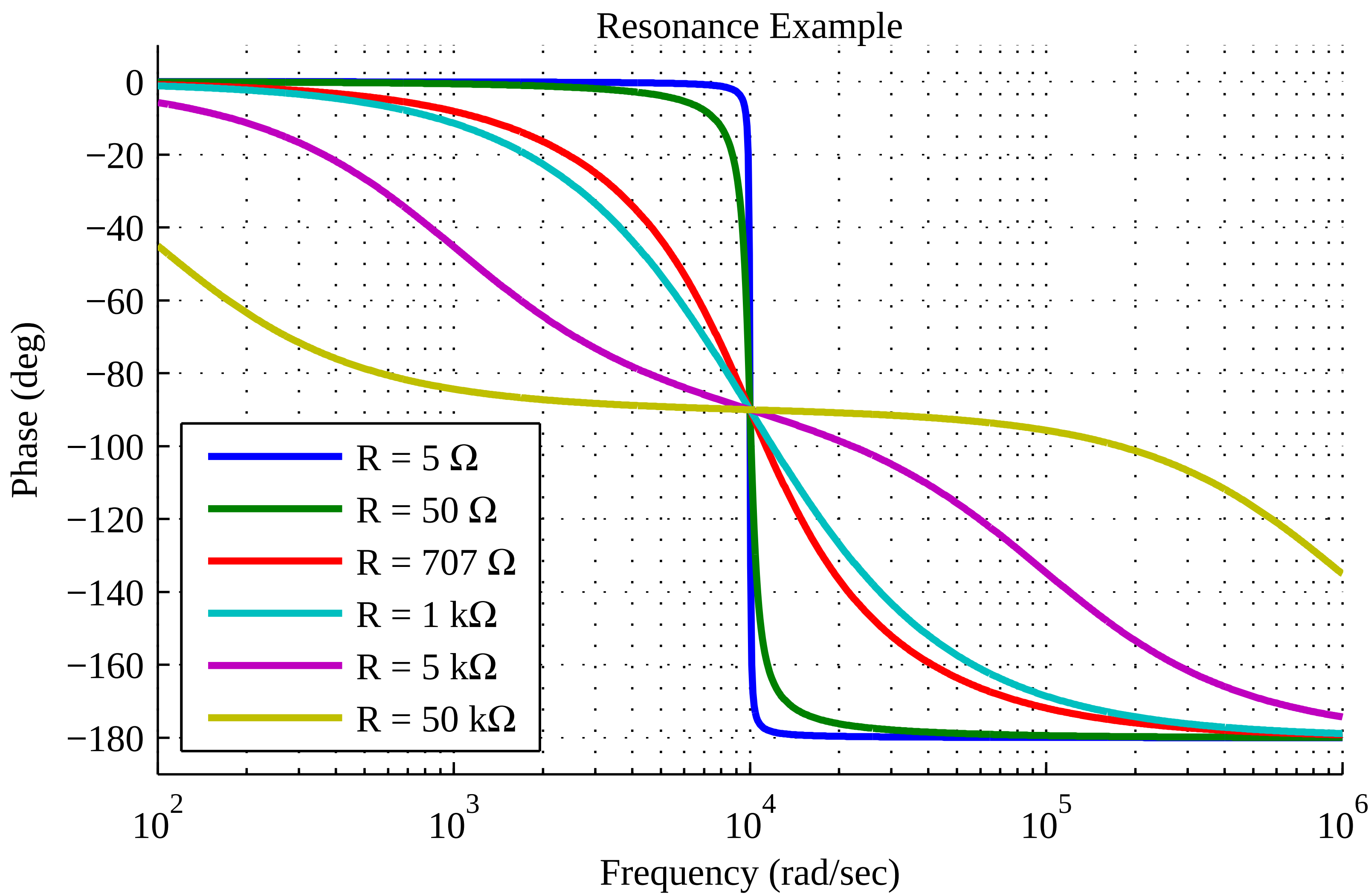
$$\begin{aligned}
\omega_n &= \sqrt{\frac{1}{LC}} = 10 \text{ k rad/s} \\
\zeta &= \frac{R}{2L}\sqrt{LC} = \frac{R}{2}\sqrt{C}L = R \times 0.001 \\
Q &= \frac{1}{\sqrt{LC}}\frac{L}{R} = \sqrt{\frac{L}{C}}\frac{1}{R} = \frac{500}{R}
\end{aligned}$$

$R = 5\ \Omega$	$\zeta = 0.005$	$Q = 100$	Very Light Damping
$R = 50\ \Omega$	$\zeta = 0.05$	$Q = 10$	Light Damping
$R = 707\ \Omega$	$\zeta = 1.41$	$Q = 0.707$	Strong Damping
$R = 1\ \text{k}\Omega$	$\zeta = 1$	$Q = 0.5$	Critical Damping
$R = 5\ \text{k}\Omega$	$\zeta = 5$	$Q = 0.1$	Over Damping

Example 1: Lowpass Magnitude Response



Example 1: Lowpass Phase Response



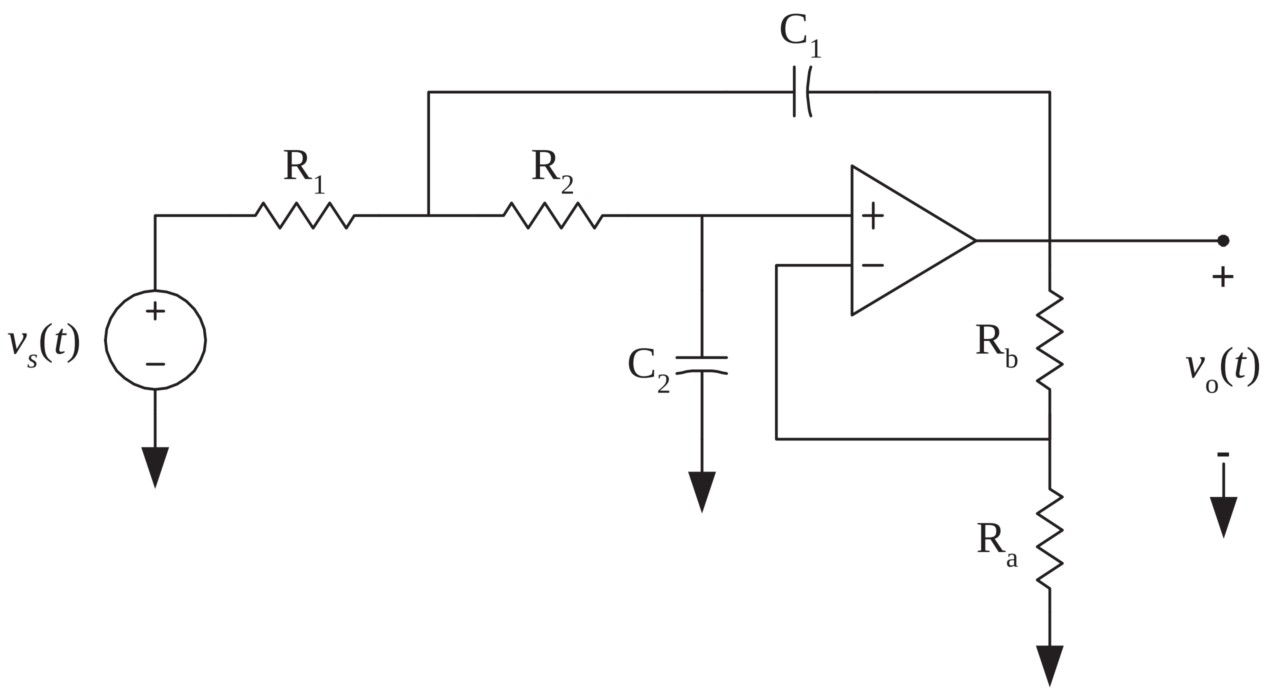
Practical Second-Order Filter Tradeoffs

- There are many standard designs for second-order filters
- A library of first and second-order filters is all you need to synthesize any transfer function
- The ideal transfer functions are identical
- Engineering design tradeoffs are practical
  - Cost (number of op amps)
  - Complexity
  - Sensitivity to component value variation
  - Ease of tuning
  - Ability to vary cutoff frequencies and/or gains

Practical Second-Order Filters

- Popular practical second-order filters include
  - KRC/Sallen-key filters
  - Multiple-feedback filters: more than one feedback path
  - State-variable filters
  - Biquad/Tow-Thomas filters filters

Example 2: Lowpass Sallen-Key Filter



Example 2: Questions

- Is the input impedance finite?
- What is the output impedance?
- How many capacitors are in the circuit?
- What is the DC gain?
- What is the "high frequency" gain?
- What is the probable order of the transfer function?
- Solve for the transfer function

Example 2: Work Space

Example 2: Work Space

Example 2: Tuning

$$|H(0)| = \frac{R_a + R_b}{R_a} = k$$

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

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

- 6 circuit parameters (four resistors, two capacitors)
- 3 constraints
  - DC gain, k
  - Natural frequency
  - Q
- Prudent to pick $K = 1$, since gain is often added at in the first and/or last stages

Example 2: Tuning Tips

$$|H(0)| = \frac{R_A + R_B}{R_A}$$

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

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

1. If you select $R_1 = R_2 = R$ and $C_1 = C_2 = C$, the equations simplify considerably
   - Watch out for coarse tolerances on components
2. Helps guard against op amp limitations if you select $k = 1$
   - Gain is often used at end or beginning of filter stages
3. Best to start of with two capacitances in a ratio $\frac{C_1}{C_2} \geq 4Q^2$

Example 3: Design

Design a Sallen-Key lowpass filter with a $Q = 1/\sqrt{2}$, **corner frequency** of 1000 Hz, and DC gain of 1.

- Redraw the circuit with a DC gain of 1
- Arbitrarily pick $C_2 = 1$ nF, a common capacitor value that is cheaply and readily available

Hint: Recall that the solution to $ax^2 + bx + c = 0$ is $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

Example 3: DC Gain Solution

$$|H(0)| = \frac{R_A + R_B}{R_A} = 1 + \frac{R_B}{R_A}$$

Pick $R_B = 0$ and $R_A = \infty$

Example 3: Solution

Define $R_1 = mR_2$ and $C_1 = nC_2$

$$\begin{aligned}
\omega_n &= \frac{1}{\sqrt{R_1C_1R_2C_2}} \\
&= \frac{1}{R_2C_2\sqrt{nm}} \\
Q &= \frac{1}{\sqrt{\frac{R_2C_2}{R_1C_1}} + \sqrt{\frac{R_1C_2}{R_2C_1}} + (1-K)\sqrt{\frac{R_1C_1}{R_2C_2}}} \\
&= \frac{1}{\sqrt{\frac{1}{mn}} + \sqrt{\frac{m}{n}}}\frac{\sqrt{mn}}{\sqrt{mn}} \\
&= \frac{\sqrt{mn}}{1+m}
\end{aligned}$$

Example 3: Solution

$$Q^2 = \frac{mn}{1 + 2m + m^2}$$

$$m^2 + 2m + 1 = \frac{mn}{Q^2}$$

$$\frac{1}{2}m^2 + m + \frac{1}{2} = \frac{mn}{2Q^2}$$

$$\frac{1}{2}m^2 + \left(1 - \frac{mn}{2Q^2}\right)m + \frac{1}{2} = 0$$

$$\beta \triangleq \frac{n}{2Q^2} - 1$$

$$\frac{1}{2}m^2 - \beta m + \frac{1}{2} = 0$$

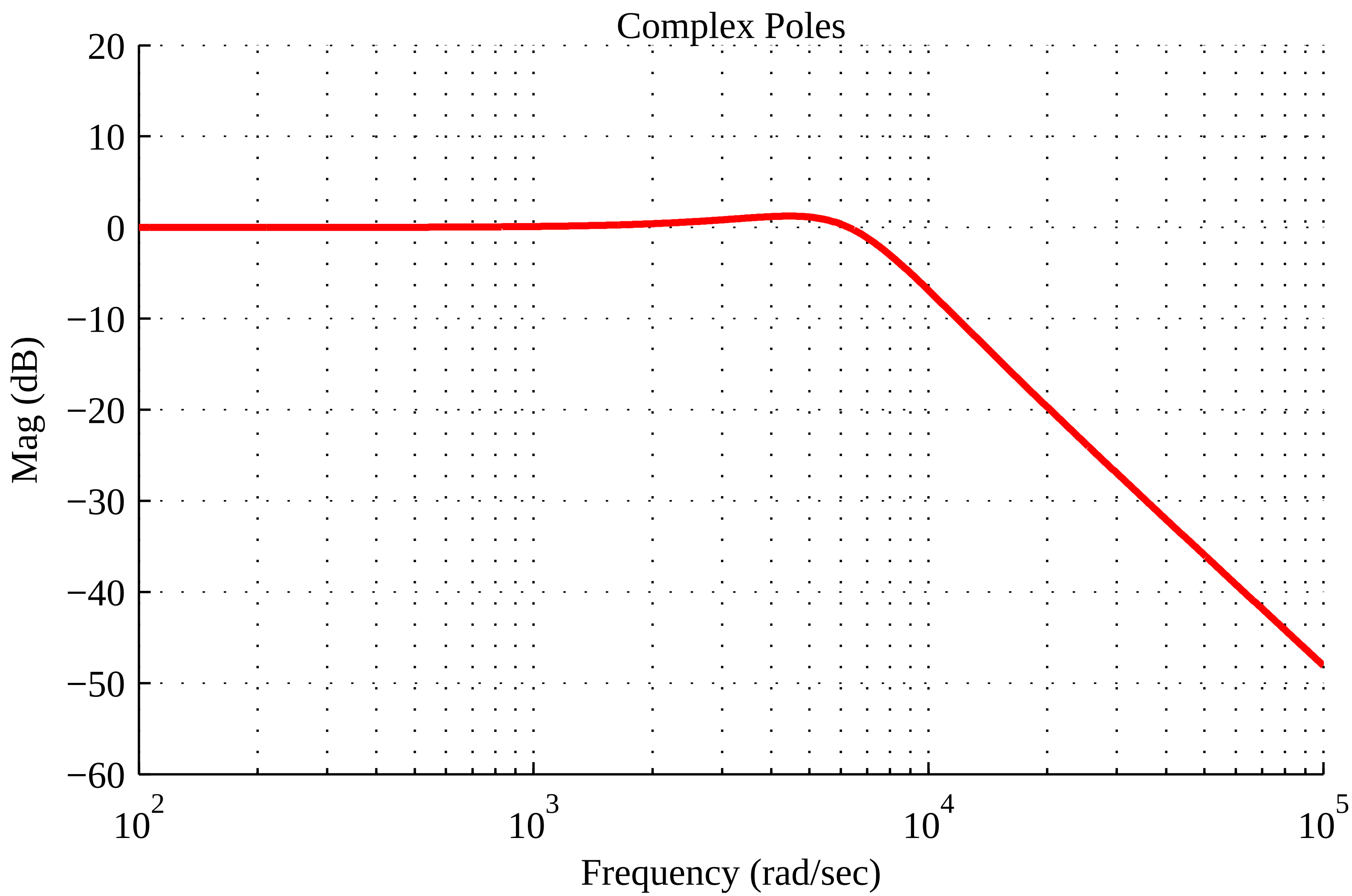
$$am^2 + bm + c = 0$$

$$m = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = \beta + \sqrt{\beta^2 - 1}$$

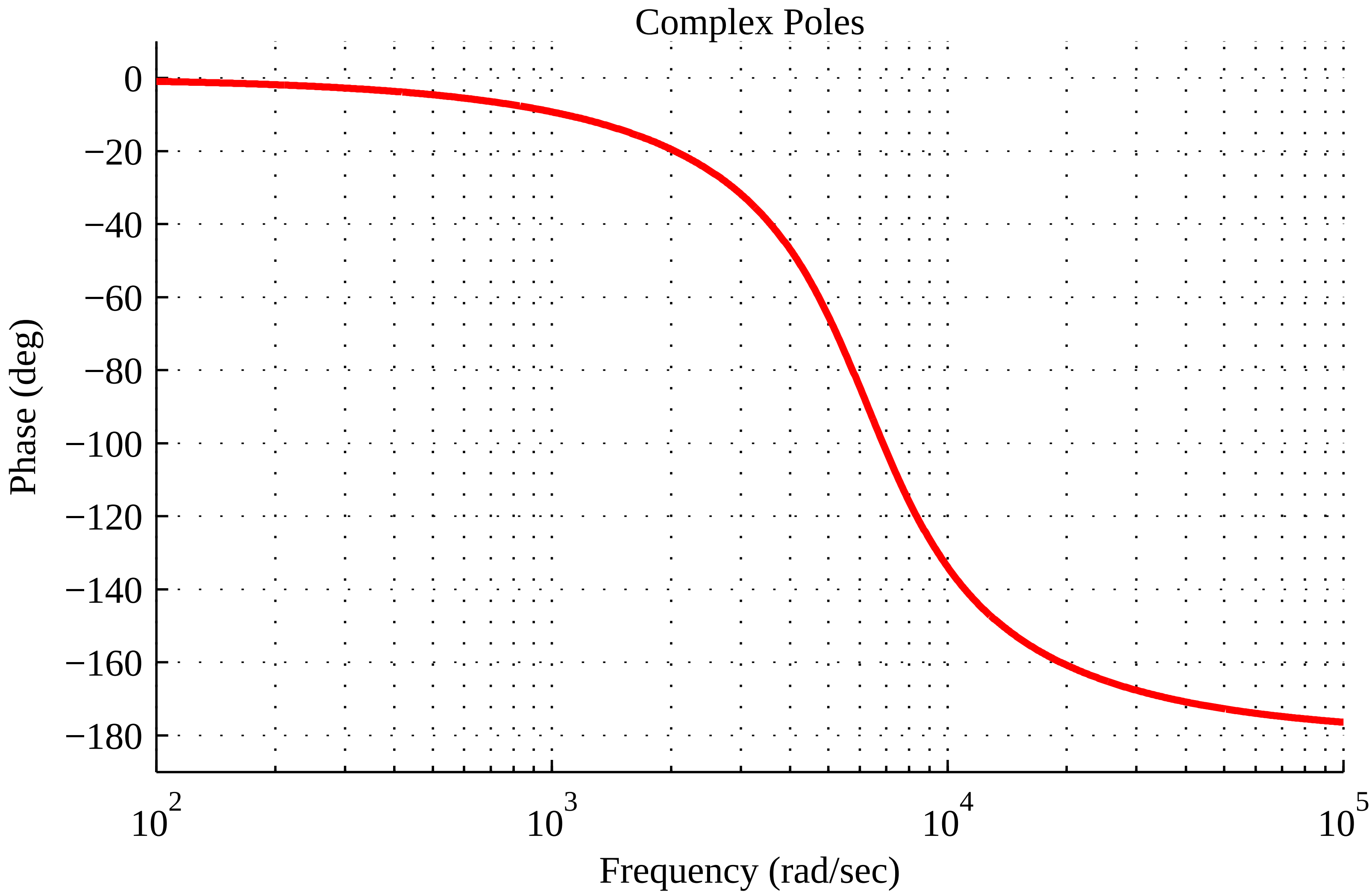
Example 3: Solution

1. Arbitrarily pick $C_2 = 10\,\text{nF}$, a common capacitor value that is cheaply and readily available
2. Select n such that $\frac{C_1}{C_2} \geq 4Q^2$, to ensure solution to quadratic equation for $m = \frac{R_1}{R_2}$ is real valued
   - Pick $n = 8$
   - Then $C_1 = 8Q^2 C_2 = 80\,\text{nF}$
3. Solve for $m = \beta + \sqrt{\beta^2 - 1}$, given $\beta = \frac{n}{2Q^2} - 1$
   - $m = 1$
4. Solve for $R_2 = \frac{1}{C_2 \omega_n \sqrt{nm}}$
   - $R_2 = 2.33\,\text{k}\Omega$
5. Solve for $R_1 = mR_2$, given m and R_2
   - $R_1 = 13.6\,\text{k}\Omega$

Example 3: Magnitude Response



Example 3: Phase Response



Example 3: MATLAB Code

```
function [] = SallenKeyLowpass();

RA = inf;
RB = 0;
C2 = 10e-9;
Q  = 1;
fn = 1000;

%==========================================================
% Preprocessing
%==========================================================
dCGain = 1 + RB/RA;
wn     = 2*pi*fn;
n      = 8*Q^2;
b      = n/(2*Q^2)-1;
m      = b + sqrt(b^2 -1);
C1     = n*C2;
R2     = 1/(wn*C2*sqrt(n*m));
R1     = m*R2;

fprintf('C1: %5.3f nF\n',C1*1e9);
fprintf('C2: %5.3f nF\n',C2*1e9);
fprintf('R1: %5.3f kOhms\n',R1*1e-3);
fprintf('R2: %5.3f kOhms\n',R2*1e-3);

N  = 1000;
functionName = sprintf('%s',mfilename);                    % Get the function name
```

```
w = logspace(2,5,N);

mag = [];
phs = [];

fileIdentifier = fopen([functionName '.tex'],'w');

sys = tf([dCGain],[1/(wn^2) 1/(wn*Q) 1]);
[m, p] = bode(sys,w);
bodeMagnitude = reshape(m,N,1);
bodePhase     = reshape(p,N,1);

figure(1);
FigureSet(1,'Slides');
h = semilogx(w,20*log10(bodeMagnitude),'r');
set(h,'LineWidth',1.5);
grid on;
xlabel('Frequency (rad/sec)');
ylabel('Mag (dB)');
title('Complex Poles');
xlim([w(1) w(end)]);
ylim([-60 20]);
box off;
AxisSet(8);
fileName = sprintf('%s-Magnitude',functionName);
print(fileName,'-depsc');
fprintf(fileIdentifier,'%%============================================================================\n'
fprintf(fileIdentifier,'\\newslide\n');
fprintf(fileIdentifier,'\\slideheading{Example \\arabic{exc}: Magnitude Response}\n');
fprintf(fileIdentifier,'%%============================================================================\n'
```

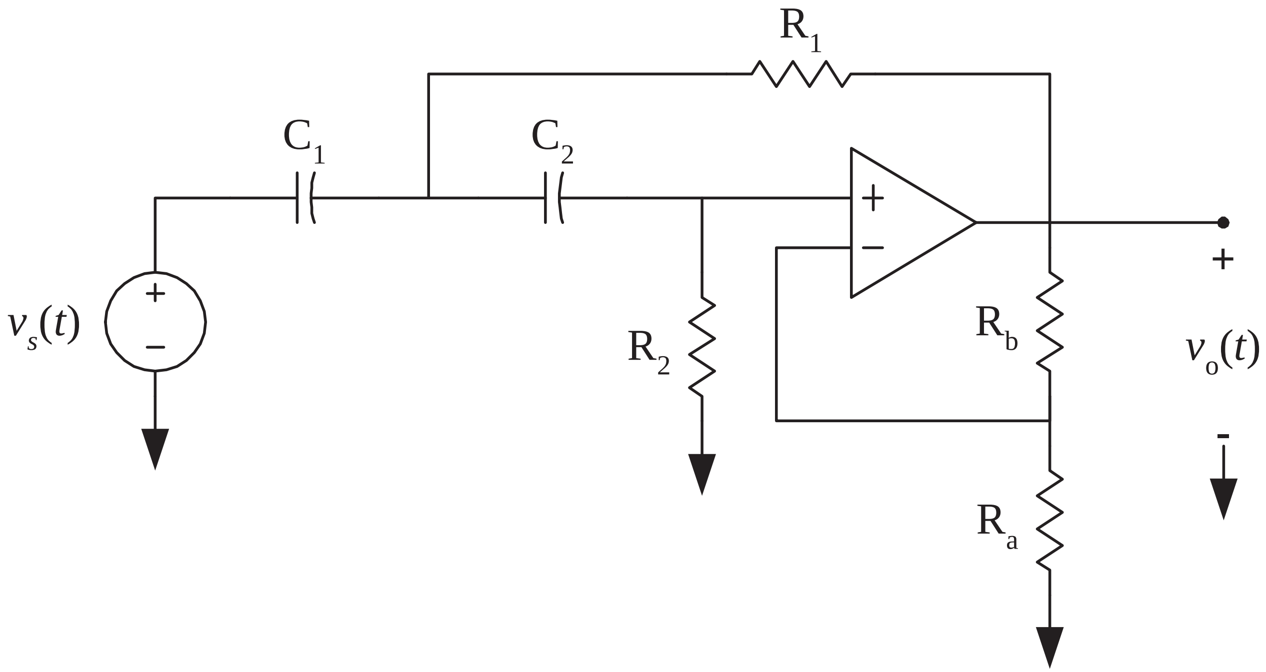
```
fprintf(fileIdentifier,'\\includegraphics[scale=1]{Matlab/%s}\n',fileName);

figure(2);
FigureSet(2,'Slides');
h = semilogx(w,bodePhase,'r');
set(h,'LineWidth',1.5);
grid on;
xlabel('Frequency (rad/sec)');
ylabel('Phase (deg)');
title('Complex Poles');
xlim([w(1) w(end)]);
ylim([-190 10]);
box off;
AxisSet(8);
fileName = sprintf('%s-Phase',functionName);
print(fileName,'-depsc');
fprintf(fileIdentifier,'%%==============================================================================\n'
fprintf(fileIdentifier,'\\newslide\n');
fprintf(fileIdentifier,'\\slideheading{Example \\arabic{exc}: Phase Response}\n');
fprintf(fileIdentifier,'%%==============================================================================\n'
fprintf(fileIdentifier,'\\includegraphics[scale=1]{Matlab/%s}\n',fileName);


fprintf(fileIdentifier,'%%==============================================================================\n'
fprintf(fileIdentifier,'\\newslide \n');
fprintf(fileIdentifier,'\\slideheading{Example \\arabic{exc}: MATLAB Code}\n');
fprintf(fileIdentifier,'%%==============================================================================\n'
fprintf(fileIdentifier,'\t \\matlabcode{Matlab/%s.m}\n',functionName);

fclose(fileIdentifier);
```

Example 4: Highpass Sallen-Key Filter



Example 4: Questions

- Is the input impedance finite?
- What is the output impedance?
- How many capacitors are in the circuit?
- What is the DC gain?
- What is the “high frequency” gain?
- What is the probable order of the transfer function?
- Solve for the transfer function

Example 4: Work Space

Example 4: Work Space

Example 4: Solution

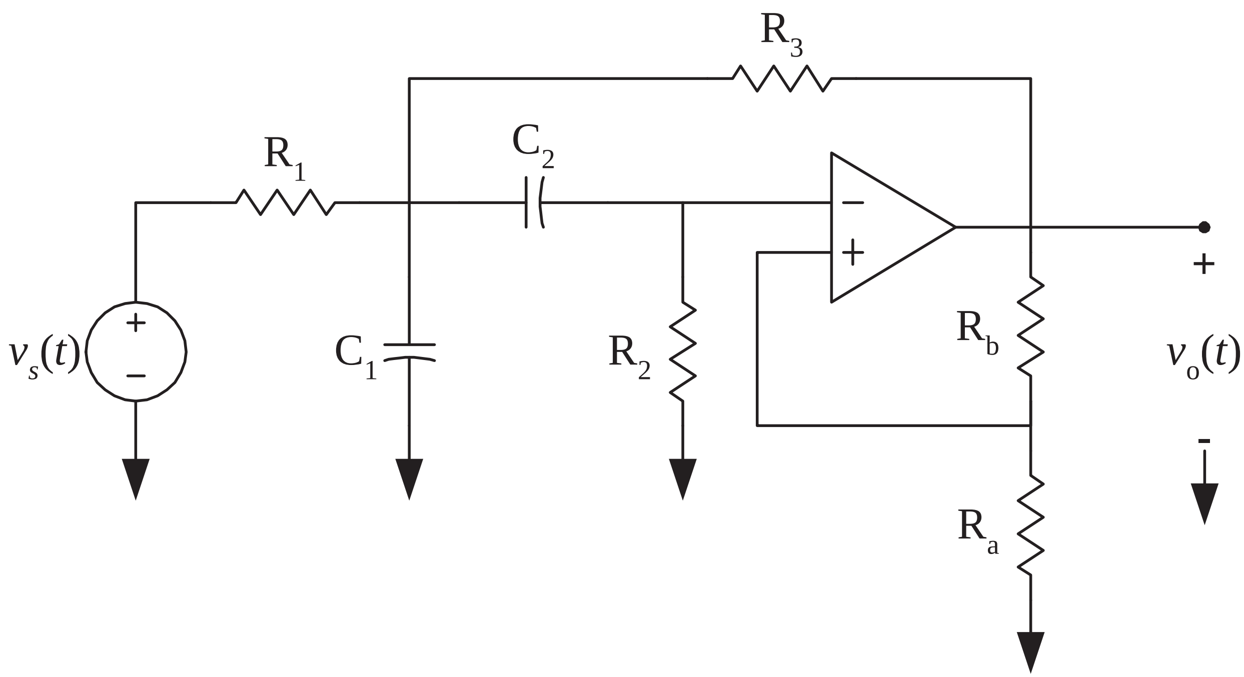
$$H(s) = K \frac{\left(\frac{s}{\omega_n}\right)^2}{1 + \frac{s}{Q\omega_n} + \left(\frac{s}{\omega_n}\right)^2}$$

$$K = 1 + \frac{R_B}{R_A}$$

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

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

Example 5: Bandpass Sallen-Key Filter



Example 5: Questions

- Is the input impedance finite?
- What is the output impedance?
- How many capacitors are in the circuit?
- What is the DC gain?
- What is the "high frequency" gain?
- What is the probable order of the transfer function?
- Solve for the transfer function

Example 5: Work Space

Example 5: Work Space

Transfer Function Synthesis

You can build almost any transfer function with integrators, adders, & subtractors

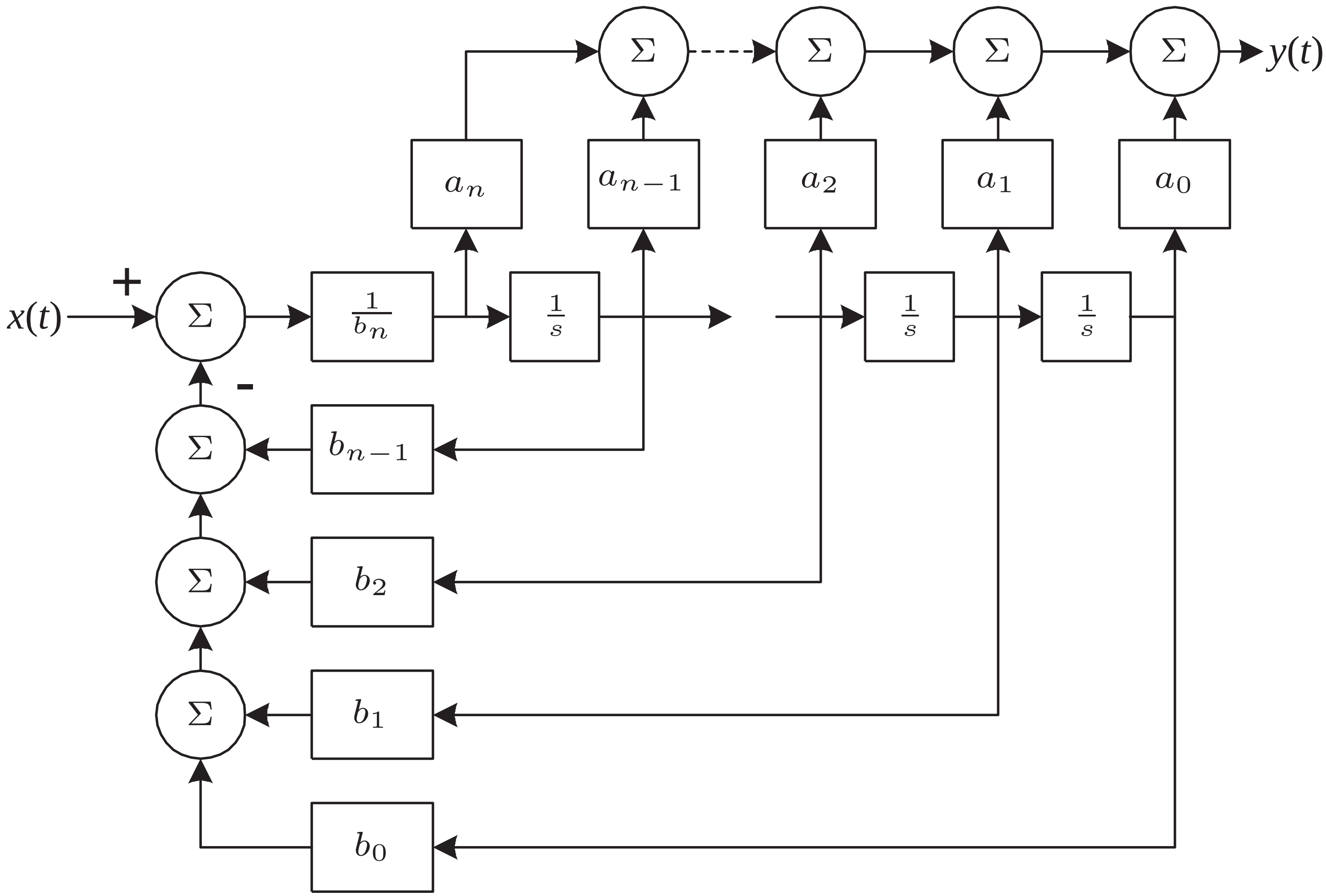
$$Y(s) = H(s)X(s) = \frac{N(s)}{D(s)}X(s) = \frac{X(s)}{D(s)}N(s) = U(s)N(s)$$

where

$$\begin{aligned}
U(s) &\triangleq \frac{1}{D(s)}X(s) \\
D(s) &\triangleq b_n s^n + b_{n-1}s^{n-1} + \cdots + b_1 s + b_0 \\
X(s) &= b_n s^n U(s) + b_{n-1}s^{n-1}U(s) + \cdots + b_1 sU(s) + b_0 U(s) \\
s^n U(s) &= \frac{1}{b_n}\left(X(s) - b_{n-1}s^{n-1}U(s) - \cdots - b_1 sU(s) - b_0 U(s)\right) \\
Y(s) &= U(s)N(s) \\
&= a_n s^n U(s) + a_{n-1}s^{n-1}U(s) + \cdots + a_1 sU(s) + a_0 U(s)
\end{aligned}$$

Transfer Function Synthesis Diagram

Transfer Function Synthesis Clean Diagram



Example 6: Transfer Function Synthesis

Draw the block diagram for the following transfer function:

$$H(s) = \frac{s^3 + 5s^2 + 2}{s^4 + 17s^3 + 82s^2 + 130s + 100}$$

Analog Filters Summary

- There are many types of filters
- Second-order filters can implement the four basic types
  - Second-order
  - Lowpass & Highpass
  - Bandpass & Bandstop
  - Notch
- For a given $H(s)$, there are many implementations & tradeoffs
- Second order filters are fundamentally different than first-order filters because they can have complex poles
  - Causes a resonance