![maxim integrated logo]

Keywords: op amp filters, Sallen-Key filters, high pass filter, opamps, single op amp

TUTORIAL 738

Minimizing Component-Variation Sensitivity in Single Op Amp Filters

Jul 22, 2002

Abstract: By delving a little more deeply into the Sallen-Key filter's component sensitivities, we discover a "sweet spot" in the mathematical analysis that allows the design of simple, single op amp filters whose performance rivals that of more complicated filters. This approach has been useful in extending the utility of Sallen-Key filters.

Dozens of filter topologies have been devised over the years, each with its own advantages. Yet engineers often rely on a small but popular subgroup of those topologies for which "cookbook" design methods are available. They choose simpler, single-op-amp filters for the less complicated lower-order designs. But when the associated cookbook approaches fail in developing a well-behaved complex filter, engineers generally turn to more complicated topologies. However, a deeper analysis of a common single-op-amp topology (the Sallen-Key filter) can lead to some interesting results if you dig below the level of cookbook formulas.

A typical filter-design process has three main stages. First, you establish the frequency range to be passed, the allowable passband ripple, etc., as determined by the system requirements. Next, you determine a transfer function (mathematical filter description) that meets the requirements, usually by choosing one of the standard types: Butterworth, Chebyshev, Bessel, Elliptical, etc. This step in the process is beyond the scope of the present article, but detailed information can be obtained from the references listed at the end.

The final step is to design and implement a circuit that provides the desired transfer function. The designer typically follows a strategy similar to the following, which is quite simple when applied to the cookbook topologies:

- Factor the transfer function into second-order parts.
- Choose a circuit topology (or topologies) that allows each second-order function to be synthesized independently.
- Design each stage as an independent second-order filter.
- Connect the resulting second-order filters in series.

A limitation common to the cookbook approaches can arise at this point. Single-op-amp filters often

exhibit great sensitivity to variations in their passive-component values, and filters with high Qs (i.e., most higher-order filters) are particularly sensitive. This sensitivity would not be a problem with perfect passive components, but actual components are available only in a limited number of standard values. Cookbook calculations can call for a 10.095k resistor, yet the nearest value actually available might be 10.0k.

Real-component values also vary from unit to unit and in response to variations in temperature and other environmental factors. The filter's sensitivity to these component variations can cause it to deviate significantly from its desired frequency response. In many cases, the designer then turns to a more complicated filter topology.

The Sallen-Key Topology

The Sallen-Key filter (Reference 1) is among the most common single-op-amp filters. Its low-pass version (**Figure 1**) has these design equations:

$$\frac{V_{out}}{V_{in}} = \frac{K}{S^2(R_1 R_3 C_2 C_4) + S(R_1 C_2 + R_3 C_4 + R_1 C_4 - K R_1 C_2) + 1} \tag{1a}$$

$$\omega_n = \frac{1}{\sqrt{R_1 R_3 C_2 C_4}} \tag{1b}$$

$$\frac{1}{Q} = \sqrt{\frac{R_3 C_4}{R_1 C_2}} + \sqrt{\frac{R_1 C_4}{R_3 C_2}} + (1 - K)\sqrt{\frac{R_1 C_2}{R_3 C_4}} \tag{1c}$$

where ω_n is the natural frequency, Q is the "quality factor" (a measure of the peaking that occurs near the natural frequency), and $K = 1 + R_B/R_A$ is the DC gain.
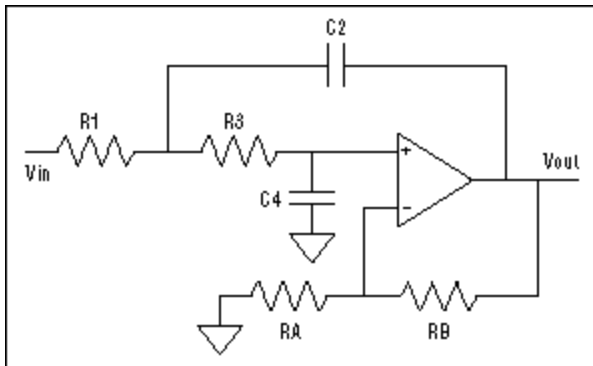

Figure 1.

Filter Sensitivity to Component Variations

Natural-frequency and Q sensitivities are useful for evaluating transfer-function stability. For the Sallen-Key filter (given in Reference 2 on page 159), these sensitivities are as follows:

$$S^Q_{R1} = -S^Q_{R3} = -\frac{1}{2} + Q\sqrt{\frac{R_3 C_4}{R_1 C_2}} \tag{2a}$$

$$S^Q_{C2} = -S^Q_{C4} = -\frac{1}{2} + Q\left(\sqrt{\frac{R_1 C_4}{R_3 C_2}} + \sqrt{\frac{R_3 C_4}{R_1 C_2}}\right) \tag{2b}$$

$$S^Q_K = QK\sqrt{\frac{R_3 C_2}{R_1 C_4}} \tag{2c}$$

$$S^Q_{RA} = -S^Q_{RB} = -Q(K-1)\sqrt{\frac{R_1 C_2}{R_3 C_4}} = -Q\frac{R_B}{R_A}\sqrt{\frac{R_1 C_2}{R_3 C_4}} \tag{2d}$$

$$S^{\omega n}_{R1} = S^{\omega n}_{R3} = S^{\omega n}_{C2} = S^{\omega n}_{C4} = -\frac{1}{2} \tag{2e}$$

$$S^{\omega n}_K = S^{\omega n}_{Ra} = S^{\omega n}_{Rb} = 0 \tag{2f}$$

Sensitivity is represented by 'S'. Its superscript is the circuit characteristic whose sensitivity is being evaluated, and its subscript is the circuit element whose effect on that characteristic is being evaluated. Thus, the first sensitivity equation (2a) shows the sensitivity of Q to variations in R_1 or R_3.

S is the power to which a component variation is raised to calculate the corresponding variation in a circuit characteristic. You may have noticed, for example, that the power for all the natural-frequency sensitivities is either -1/2 or 0. When S = -1/2 and the component varies by a factor of 'A', the natural frequency varies by $A^{-0.5}$ (i.e., $1/\sqrt{A}$). Thus, the new natural frequency will be the original frequency divided by $\sqrt{A}$. When S = 0, the frequency does not change, because $A^0 = 1$.

Reference 2 covers sensitivity in much greater detail, and it includes the derivations of many of the sensitivity equations listed above. References 3 and 4 also have good treatments of this important subject. These equations are so complicated that it often is quite difficult to choose the six passive components for a desired natural frequency and Q, while simultaneously achieving low-Q sensitivities, unless you play around with the equations and notice some interesting things that happen when the gain (K) is set to 1.

K = 1 Simplifies the Sallen-Key Filter

We can simplify the Sallen-Key filter equations a great deal by choosing K = 1 as the DC gain value. In this case, the equation for Q (equation 1c) simplifies to

$$\frac{1}{Q} = \sqrt{\frac{R_3 C_4}{R_1 C_2}} + \sqrt{\frac{R_1 C_4}{R_3 C_2}} \tag{3}$$

We also find that the sensitivity of Q to RA and RB goes to zero for K = 1 (equation 2d). This is no surprise. Setting K = 1 configures the op amp as a voltage follower by connecting its output directly to

the inverting input, which eliminates RA and RB.

Substituting the simplified equation for Q (equation 3) into the sensitivity equations 2a and 2c simplifies those equations:

$$S_{R1}^{Q} = -S_{R3}^{Q} = -\frac{1}{2} + \frac{R_3}{(R_3 + R_1)} \tag{4a}$$

$$S_{C2}^{Q} = -S_{C4}^{Q} = \frac{1}{2} \tag{4b}$$

The sensitivity of Q to the resistors can be simplified further by setting the resistor values equal. When R1 and R3 are exactly equal, the sensitivity is zero. But the values of actual resistors are never truly equal. As they deviate from their nominal values, the sensitivity becomes non-zero but remains very small. Resistors with 5% tolerance, for example, cause a worst-case sensitivity of

$$-\frac{1}{2} + \frac{1.05}{(1.05 + 0.95)} = 0.025 \qquad \text{(substitution in equ. 4a)}$$

Thus, 5% resistor variations produce a 0.12% Q variation, which is negligible in comparison to the other sensitivities. Even if the resistor values are not equal, this sensitivity is between -1/2 and +1/2. A closer look at the Q equation shows other reasons for setting R1 = R3:

$$\frac{1}{Q} = \left(n + \frac{1}{n}\right)\sqrt{\frac{C_4}{C_2}} \qquad where \qquad n = \frac{R_1}{R_3} \tag{5}$$

When n = 1 (i.e., when $R_3 = R_1$), the quantity (n+1/n) has a minimum value of 2. Thus, the filter capacitors will be equal-valued when Q = 1/2 and the resistors are set equal to each other. For all Qs higher than 1/2 (the most common case by far), C_2 must be larger than C_4. If the resistors are not equal, the ratio of C_2 to C_4 must be made larger. To minimize this spread in capacitor values, the resistor values should therefore be equal.

For equal resistor values, the equation for Q simplifies to

$$\frac{1}{Q} = 2\sqrt{\frac{C_4}{C_2}} \tag{6}$$

Rearranging this equation to get C2 in terms of C4,

$$C_2 = 4Q^2 C_4 \tag{7}$$

We can substitute this expression into the equation for ω_n (equation 1b) and solve for C4:

$$C_4 = \frac{1}{2\omega_n RQ} = \frac{1}{4\pi F_n RQ} \qquad where \qquad R1 = R3 = R \tag{8}$$

Substituting this result into the equation for C2 (equation 7), we get

$$C_2 = \frac{2Q}{\omega_n R} = \frac{Q}{\pi F_n R} \qquad (9)$$

The Simplified Design Process

Setting the Sallen-Key filter gain to unity and setting R1 = R3 enables the design of low-sensitivity single-op-amp filters by solving two simple equations. The simplified design process is as follows:
- Select an appropriate resistor value.
- Use equations 8 and 9 for solving the capacitor values.
- If C2 is too large, start over with a larger resistor value.
- If C4 is too small, start over with a smaller resistor value.
- If C4 is too small *and* C2 is too large, you have reached the limit for this filter.
- Pick standard values closest to the calculated values.

Two examples illustrate this method and the benefits derived from its use.

Comparing the New Approach to a Cookbook Approach

The first example comes from work done by the author several years ago. To minimize circuit variations in production, he redesigned a circuit (initially created with cookbook techniques) to realize a third-order Butterworth low-pass filter with a -3dB frequency of 4.8kHz. The redesign eliminated a trim pot and its associated need for tweaking.

This filter required a second-order stage with Q = 1 and a natural frequency of 4.8kHz. It was implemented initially with the Sallen-Key topology and the design approach of Reference 2, pages 156 to 157, which sets the resistor values equal (R1 = R3 = R) and capacitor values equal (C2 = C4 = C). Choosing C = 0.001µF resulted in gain (K) = 2 and R = 33.2K. The Q sensitivities for this circuit are as follows:

$$s_{R1}^Q = -s_{R3}^Q = \frac{1}{2} \qquad s_{C2}^Q = -s_{C4}^Q = 1.5 \qquad s_{RA}^Q = -s_{RB}^Q = 1 \qquad (10)$$

The filter was redesigned with our new method using the same resistor value (33.2kΩ). Equations 8 and 9 result in C2 = 2000pF and C4 = 500pf. The sensitivities are as follows:

$$s_{C2}^Q = -s_{C4}^Q = \frac{1}{2} \qquad and \qquad s_{R1}^Q = -s_{R3}^Q = 0 \qquad (11)$$

Figure 2 shows the schematics used to simulate these circuits, and **Figure 3** the results of SPICE simulations. These frequency response plots are the result of a Monte Carlo run of 100 different "builds" for each filter, using 1%-tolerance resistors and 5% capacitors. For each "build," the SPICE simulator randomly varies the component values within their specified tolerances. Note that for all frequencies in the passband (especially those near the natural frequency), the new filter has substantially less variation than the old one.
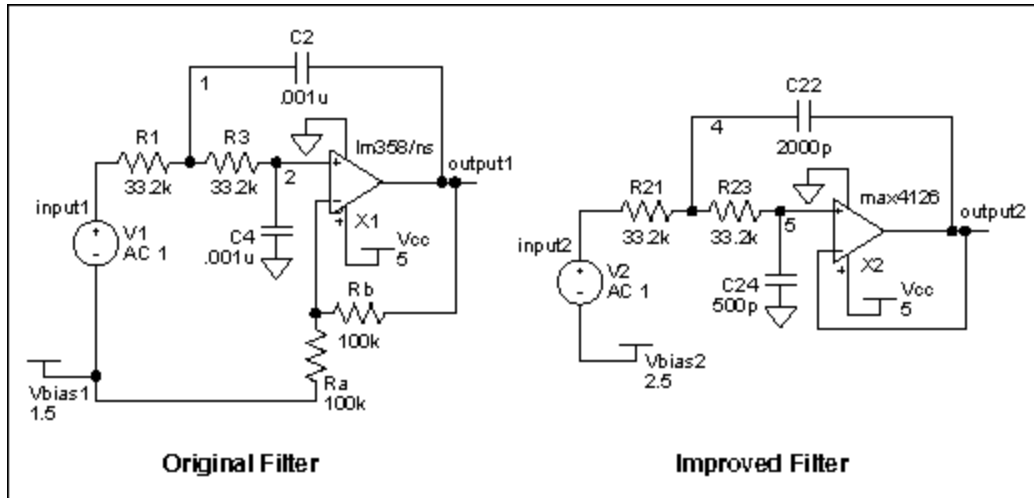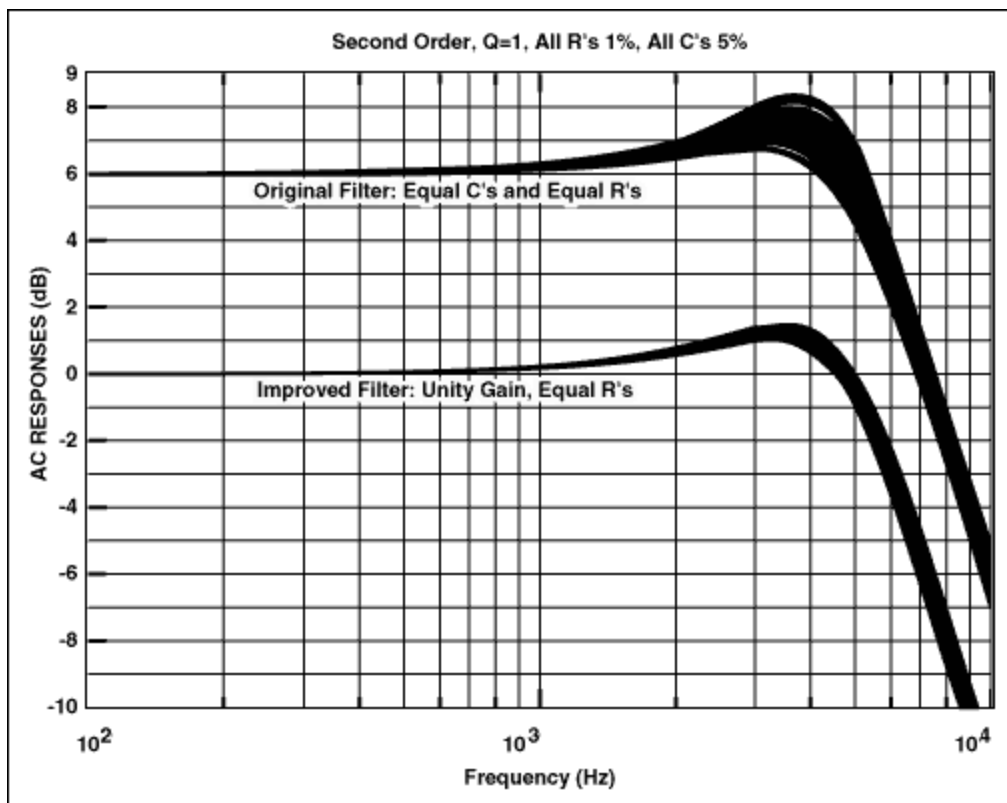
Figure 2.

Figure 3.

It is also important to note that these simulation results are not the only steps needed to prove circuit operation; you should also build and test the circuit. Once the SPICE simulator indicates AC performance equal to that of the actual circuit with nominal component values, you can use the Monte Carlo function common in SPICE simulators to evaluate how the circuit response changes with variation in the components.

Cascaded Stages Implement Higher-Order Filters

The unity-gain Sallen-Key approach has two disadvantages. It cannot provide gain, and for high-Q filters its capacitor ratio may be too large to allow realization of the filter. Existing amplifier stages can often provide the needed gain; but, if not, the worst-case solution is to add a single-op-amp gain stage.

High-order filters often require at least one stage with a very high Q. This stage can be implemented with a more complicated topology, whereas the other stages are implemented with low-sensitivity Sallen-Key circuits. Even with Q limitations, the Sallen-Key topology can execute the high-order filters traditionally implemented with multi-op-amp topologies. The following example shows a new procedure for designing such a filter, demonstrating a dramatic improvement in performance over the older methods. Nominal specifications are as follows:

- Seventh-order Chebyshev
- 0.05dB ripple
- 8kHz -3dB frequency
- Gain = 10

An actual derivation of the transfer function from these objective specifications is beyond the scope of this article, but the references cover that subject in detail. The transfer function has three complex-pole pairs and one simple pole:

Fn	Q
7.834kHz	5.5662
6.560kHz	1.6636
4.492kHz	0.7882
3.162kHz	simple

The schematics for two versions of this filter are shown in **Figure 4**. One was designed with the cookbook approach and the other with our new method. A single op amp in the last stage of each circuit provides the required gain and the seventh pole.
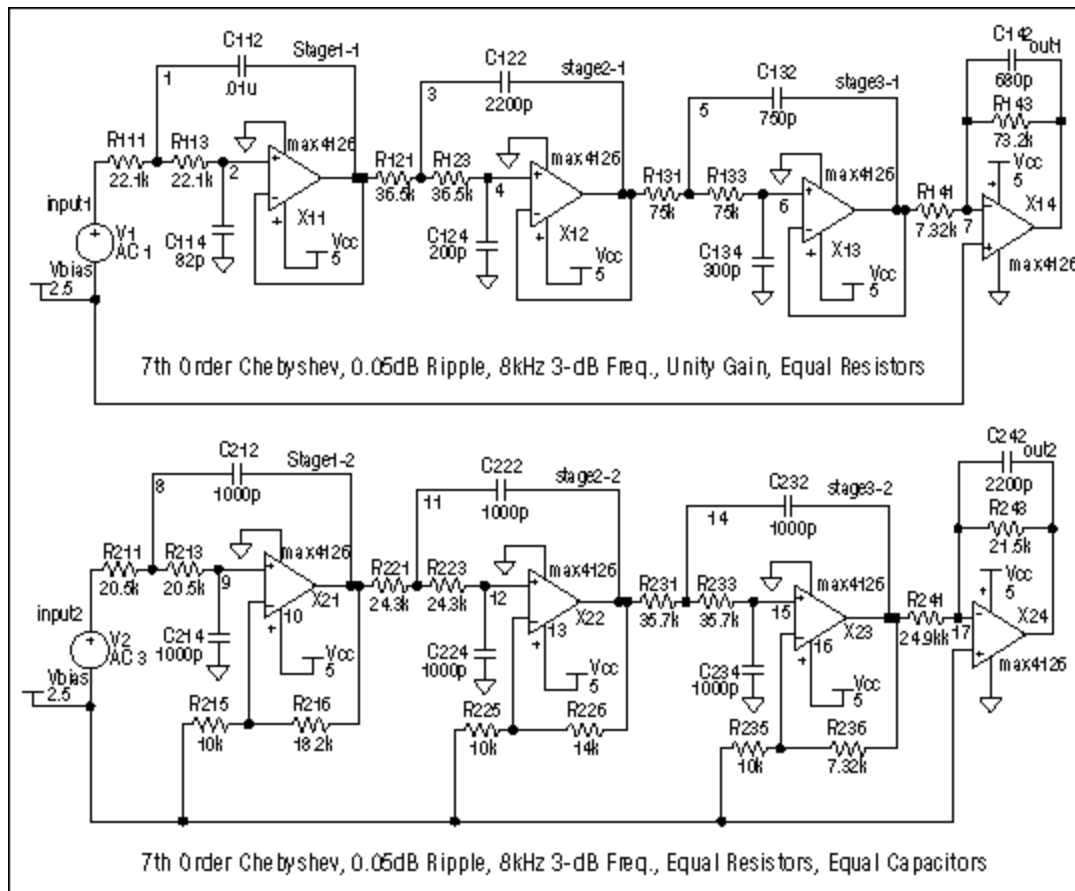
Figure 4.

Figure 5 shows the results of a Monte Carlo analysis with 1% resistors and 5% capacitors. The results are offset on the graph for visual clarity. The cookbook version has a variation of approximately 28dB near the natural frequency, rendering that design useless. In contrast, the unity-gain/equal-resistor version has a gain variation near resonance of only 4dB.
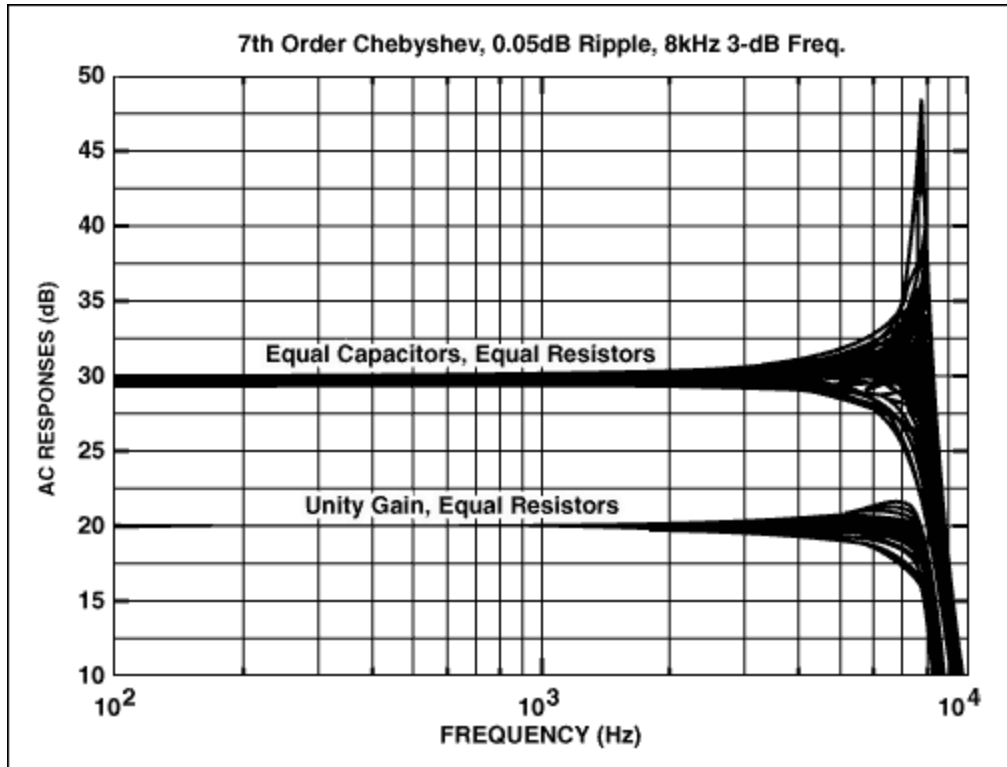
7th Order Chebyshev, 0.05dB Ripple, 8kHz 3-dB Freq.

Equal Capacitors, Equal Resistors

Unity Gain, Equal Resistors

Figure 5.

Sallen-Key vs. Biquad Filters

It is interesting to compare this seventh-order Sallen-Key circuit with a multi-op-amp implementation of the same transfer function. The biquad is a very common three-op-amp filter that provides low sensitivity and simple design equations. Its schematic is shown in **Figure 6**. Using the techniques described in Reference 2, the sensitivities are as follows:

$$S^Q_{R1} = 1 \tag{12a}$$

$$S^Q_{R2} = S^Q_{R3} = S^Q_{C2} = -S^Q_{C1} = -\frac{1}{2} \tag{12b}$$

$$S^Q_{R4} = -S^Q_{R} = 0 \tag{12c}$$

$$S^Q_{R2} = S^Q_{R3} = S^Q_{C1} = S^Q_{C2} = -\frac{1}{2} \tag{12d}$$

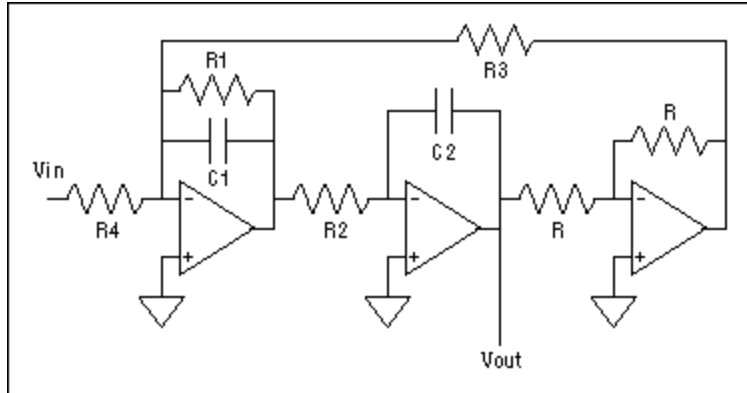$$S^{\omega_n}_{R1} = S^{\omega_n}_{R4} = S^{\omega_n}_{R} = 0 \tag{12e}$$

Figure 6.

The schematic for a biquad implementation of our specification is shown in **Figure 7**. Again, a single-op-amp stage at the end executes the gain and the seventh pole. **Figure 8** compares the frequency responses for Monte Carlo analyses of the unity-gain Sallen-Key and biquad implementations (results are again offset for clarity). With equivalent components, there is no significant difference in performance between these two filters. In fact, the Sallen-Key's low-frequency gain variation is slightly smaller than that of the biquad.
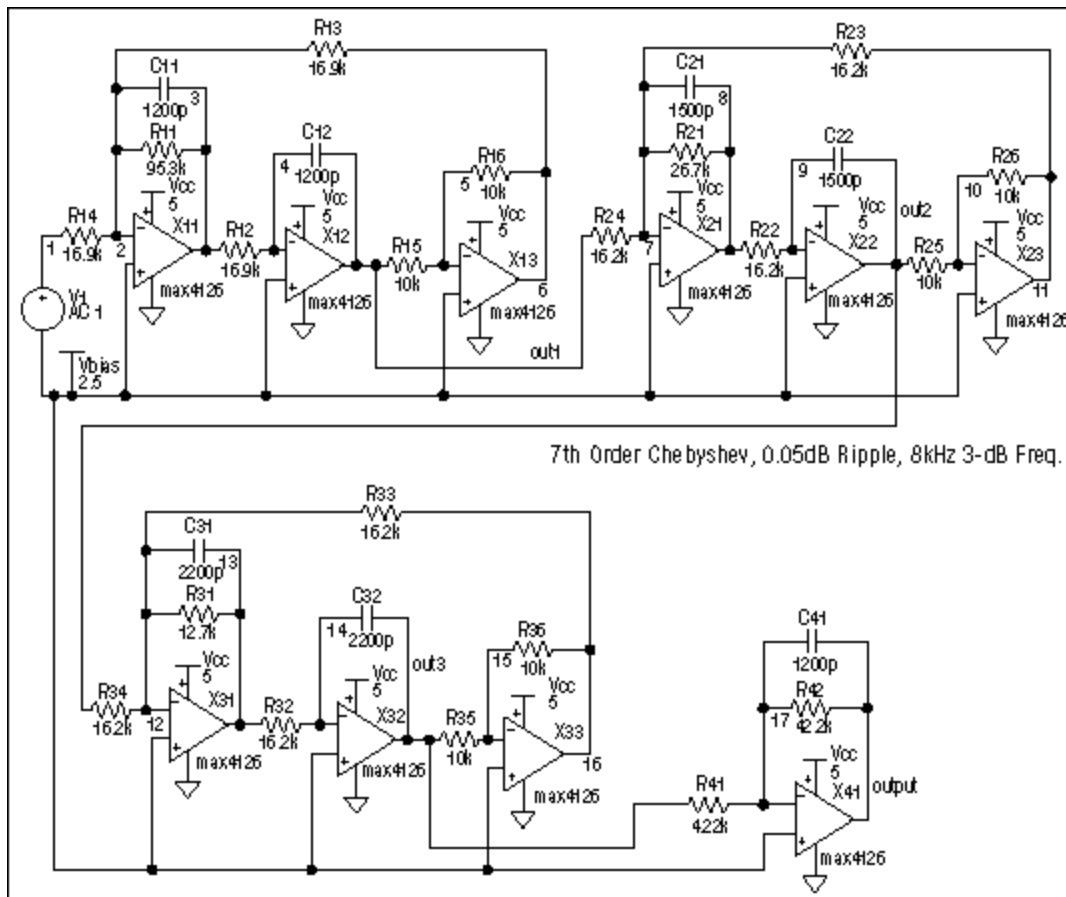

7th Order Chebyshev, 0.05dB Ripple, 8kHz 3-dB Freq.
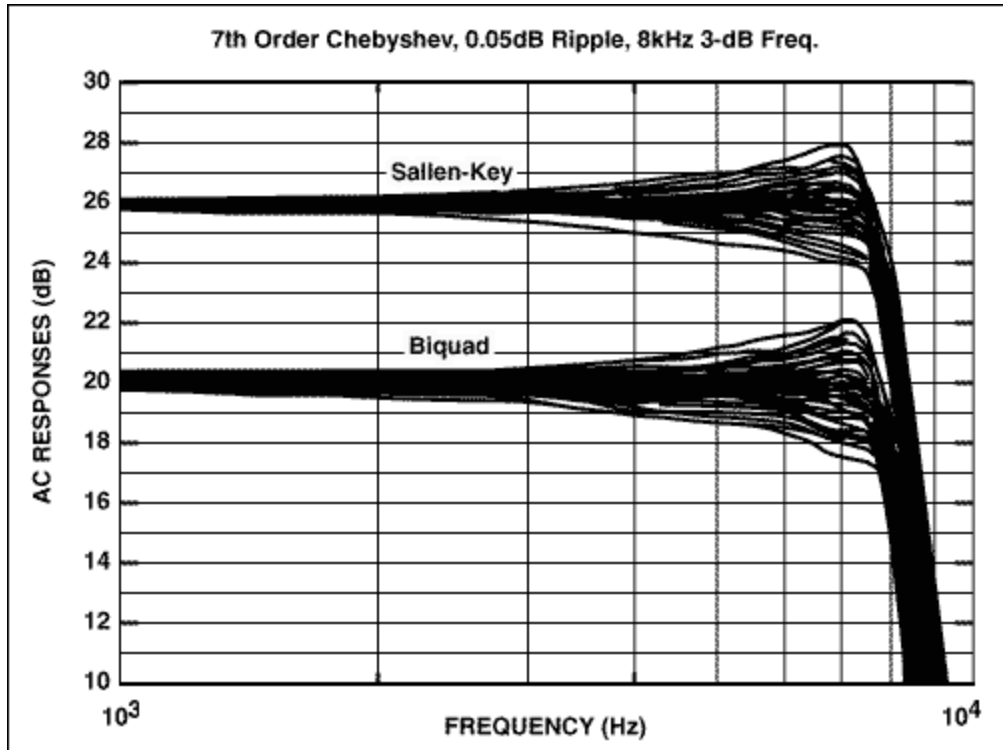
Figure 7.

Figure 8.

The following table lists the component count, passband variation, and spread of capacitor values for three implementations of the seventh-order filter just discussed:

Circuit	Passband Variation	Op Amps	Resistors	Capacitors	Capacitor Spread
Unity Gain S-K	4dB	4	8	7	Narrow to wide
Equal R/C S-K	28dB	4	14	7	Narrow
Biquad	4dB	10	20	7	Narrow

Extending These Techniques to High-Pass Filters

We can also design low-sensitivity high-pass filters using these techniques. The equivalent Sallen-Key high-pass filter is shown in **Figure 9**.
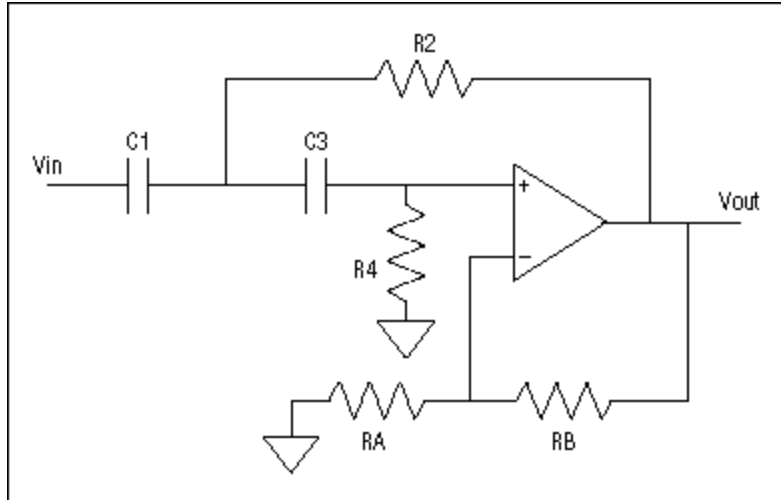
Figure 9.

Its design equations and sensitivities are as follows:

$$\frac{V_{out}}{V_{in}} = \frac{S^2 R_2 R_4 C_1 C_3 \left(1 + \frac{R_b}{R_a}\right)}{S^2(R_2 R_4 C_1 C_3) + S(R_4 C_3 + R_2 C_1 + R_2 C_3 - K R_4 C_3) + 1} \qquad (13a)$$

$$\omega_n = \frac{1}{\sqrt{R_2 R_4 C_1 C_3}} \qquad (13b)$$

$$\frac{1}{Q} = \sqrt{\frac{R_2 C_1}{R_4 C_3}} + \sqrt{\frac{R_2 C_3}{R_4 C_1}} + (1 - K)\sqrt{\frac{R_4 C_3}{R_2 C_1}} \qquad (13c)$$

$$S_{C3}^{Q} = - \ S_{C1}^{Q} = -\frac{1}{2} + Q \sqrt{\frac{R_2 C_1}{R_4 C_3}} \qquad (13d)$$

$$S_{R4}^{Q} = - S_{R2}^{Q} = -\frac{1}{2} + Q \left(\sqrt{\frac{R_2 C_1}{R_4 C_3}} + \sqrt{\frac{R_2 C_3}{R_4 C_1}}\right) \qquad (13e)$$

$$S_{RA}^{Q} = - S_{RB}^{Q} = - Q(K\text{-}1)\sqrt{\frac{R_4 C_3}{R_2 C_1}} = - Q \frac{R_B}{R_A}\sqrt{\frac{R_4 C_3}{R_2 C_1}} \qquad (13f)$$

$$S_{R2}^{\omega_n} = S_{R4}^{\omega_n} = S_{C1}^{\omega_n} = S_{C3}^{\omega_n} = -\frac{1}{2} \qquad (13g)$$

$$S_{K}^{\omega_n} = S_{Ra}^{\omega_n} = S_{Rb}^{\omega_n} = 0 \qquad (13h)$$

As for the low-pass case, we can simplify by setting K = 1 and (in this example) setting the capacitor values equal. The equations then simplify to:

$$\frac{1}{Q} = 2\sqrt{\frac{R_2}{R_4}}$$ (14a)

$$S_{C3}^Q = -S_{C1}^Q = 0$$ (14b)

$$S_{R4}^Q = -S_{R2}^Q = \frac{1}{2}$$ (14c)

The result is two simple equations for the resistors, in which C1 = C2 = C:

$$R_2 = \frac{1}{2\omega_n C Q} = \frac{1}{4\pi F_n C Q}$$ (15a)

$$R_4 = \frac{2Q}{\omega_n C} = \frac{Q}{\pi F_n C}$$ (15b)

Thus, the high-pass design procedure is very similar to that of the low-pass case:
- Select an appropriate value for C.
- Calculate the resistor values using equations 15a and 15b.
- If R4 is too large, start over with a larger C value.
- If R2 is too small, start over with a smaller C value.
- If R2 is too small *and* R4 is too large, then you've reached the limit for this type of filter.
- Pick standard values closest to the calculated values.

To illustrate this procedure, we can design a second-order high-pass filter stage with a Q of 1.0 and a natural frequency of 8.0kHz. First, choose C = 1200 pF. Next, set the resistor values using equations 15a and 15b: R_2 = 8.25kΩ and R_4 = 33.2kΩ. The frequency response is plotted along with that of the low-pass filter discussed earlier (**Figure 10**).
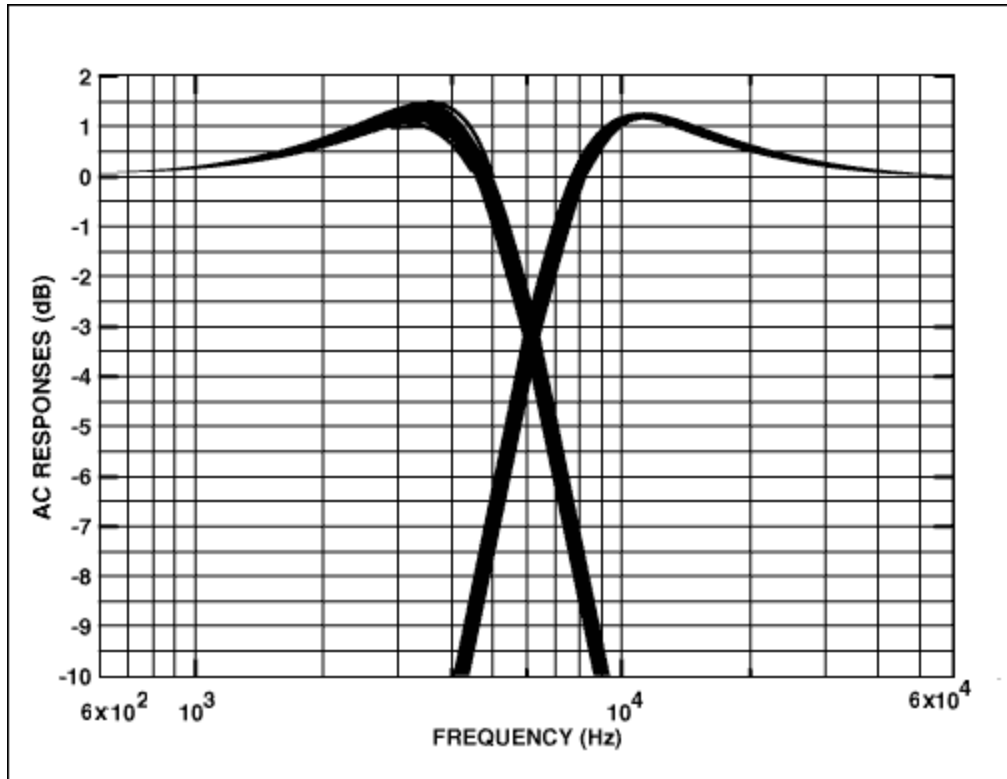
Figure 10.

The spread in frequency response for the high-pass filter is tighter than that of the low-pass filter, particularly around the peak. This occurs because we minimized sensitivity to the components with the widest variations: the capacitors. For the low-pass case, the Q sensitivity to resistors is minimized and the sensitivity to capacitors is 1/2. For the high-pass case, the sensitivity to resistors is 1/2 and the sensitivity to capacitors is minimized. We used 1% resistors and 5% capacitors, because low-tolerance resistors are more readily available than low-tolerance capacitors. If you choose 5% resistors, the two circuits exhibit a similar spread.

Conclusion

By delving a little more deeply into the Sallen-Key filter's component sensitivities, we found a "sweet spot" in the mathematical description that allows the design of simple, single-op-amp filters whose performance rivals that of more complicated filters. This approach has been useful in extending the utility of Sallen-Key filters to that of higher-order higher-Q filters, and it is likely that a similar investigation into other topologies would have similar results.

Appendix A

Be Careful with Your Math!

It is also instructive to show how the clever use of alternative mathematical representations can cause us to miss simplifications and lose sight of the physical relationships in a circuit. On page 158 of Reference 2 (the source for most of the mathematical analysis used here), the authors present a unity-gain low-pass Sallen-Key filter. They introduce the parameters $m = C4/C2$ and $n = R3/R1$ to simplify calculations and show the Sallen-Key Q sensitivity using these new parameters:

$$s^Q_{C2} = -s^Q_{C4} = -\frac{1}{2} + Q\left(\sqrt{\frac{m}{n}} + \sqrt{mn}\right)$$

$$s^Q_{R1} = -s^Q_{R3} = -\frac{1}{2} + Q\sqrt{mn}$$

By not replacing Q, m, and n with their equivalent expressions in terms of resistors and capacitors, we completely miss the fact that the sensitivity to capacitors is simply 1/2. The sensitivity to resistors appears to be a function of all the capacitors and resistors, but it is actually a function of the resistors alone.

The moral of this story is that we should always substitute the actual physical quantities back into our results to check for simplifications.

Appendix B

Always Question Your SPICE Simulator

Schematics based on SPICE simulations of second-order filters discussed in the body of the text are shown in Figure 2. The original filter, featuring an LM358 op amp, a 5V supply, and a 1.5V virtual ground established by Vbias1, is so arranged because the LM358 cannot reliably swing closer to the positive rail than about (VCC -1.5V). With a 10% tolerance on the 5V rail, we must assume that VCC can be as low as 4.5V, so a 1.5V bias centers the signal within the minimum available range.

The new filter is built around the MAX4126, a dual op amp in the µMax package (1/2 the footprint of a standard 8-pin SO). Its input common-mode range extends beyond the rails. The output can swing to within a few millivolts of the rails when unloaded and to within 200mV when loaded with 250 Ω. This symmetrical input and output voltage range allows us to maximize the circuit's dynamic range by setting the virtual ground at $V_{CC}/2$. The combination of nearly ideal input- and output-voltage ranges also simplifies operation with a 3V power supply.

These differences in output-voltage range do not show up in the AC SPICE analyses we have run, nor are they apparent in a small-signal AC test of any real circuit, provided the op amp has sufficient bandwidth. For the same reason, these simulations give virtually identical results with most SPICE op-amp models. Only a large-signal-transient (time-domain) analysis can show the limitations on a signal range.

Good engineers always check their designs by running a transient (time-domain) analysis to ensure that they have modeled their circuits properly. They should also breadboard the circuit and take any simulation results with a grain of salt, particularly when the simulation encroaches upon signal-range limitations.

Figure 11 includes the results of two transient analyses run with TopSPICE. The first plot, which compares the MAX4126 and the LM358 output capabilities, is a simulation of the same circuit description used for AC analysis. Each output was loaded with a moderate 2kΩ load terminated at the virtual ground.
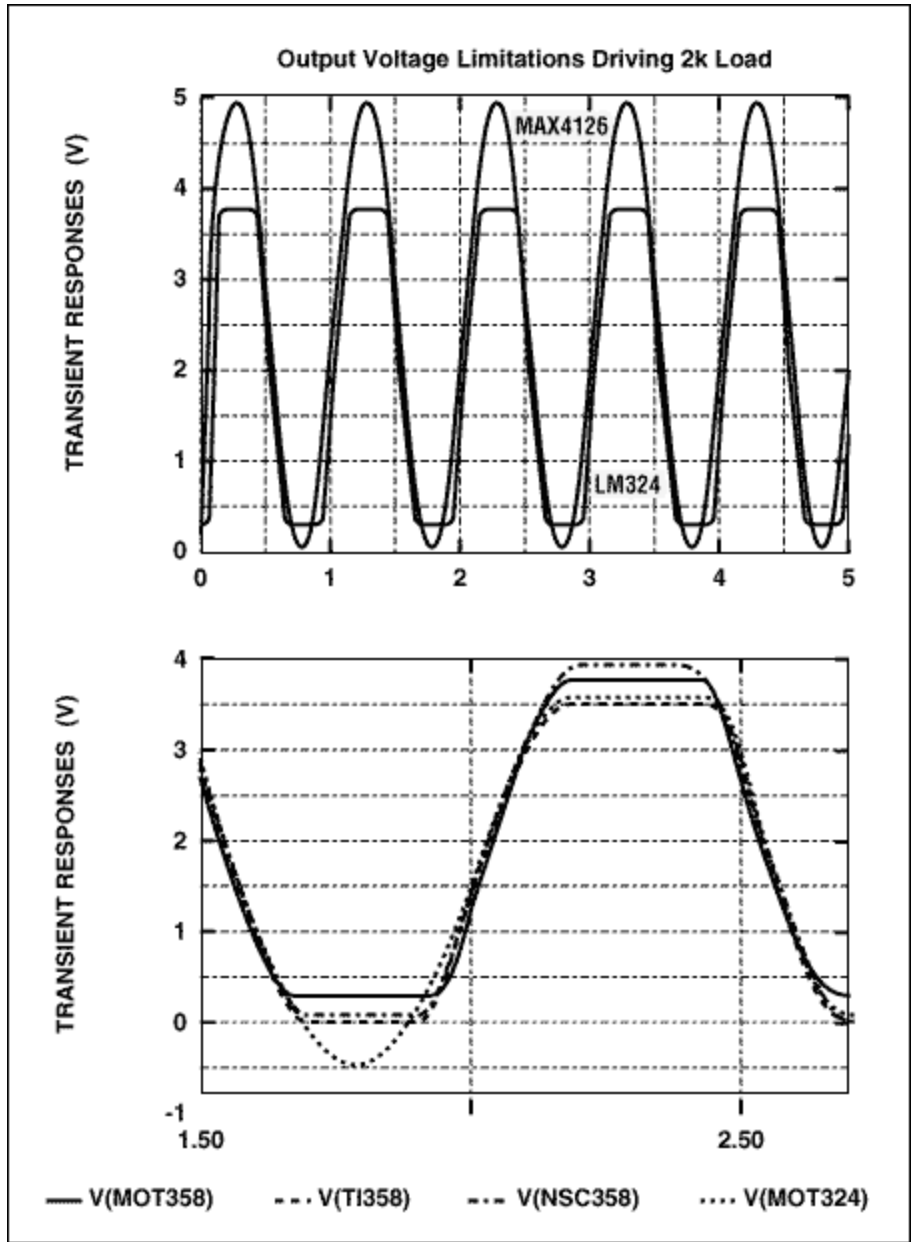
Output Voltage Limitations Driving 2k Load

Figure 11.

The second plot shows differences among four vendor-supplied versions of the LM358 and the LM324 (a quad version of the LM358). One model indicates that the LM358 will drive all the way to ground, another shows the output going to within about 100mV of ground, and a third shows the output reaching only about 350mV. Real-circuit testing with real devices indicates that this is the more accurate result. Finally, the last model shows the output going 0.5V below the negative rail!

References

1. Sallen, R.P. and Key, E.L., "A Practical Method of Designing Active Filters," *IRE Transactions on Circuit Theory,* vol. CT-2, pp.74 to 85, March 1955.
2. Huelsman, L.P. and Allen, P.E., *Introduction to the Theory and Design of Active Filters,* McGraw-Hill, New York, 1980.

3. Budak, Aram, *Passive and Active Network Analysis and Synthesis,* Houghton Mifflin Company, Boston, 1974.
4. Ghausi, M.S. and Laker, K.R., *Modern Filter Design: Active RC and Switched Capacitor,* Prentice-Hall, Englewood Cliffs, N.J., 1981.
5. TopSPICE mixed-mode circuit simulator, Rev

More Information

For Technical Support: http://www.maximintegrated.com/support
For Samples: http://www.maximintegrated.com/samples
Other Questions and Comments: http://www.maximintegrated.com/contact

Application Note 738: http://www.maximintegrated.com/an738
TUTORIAL 738, AN738, AN 738, APP738, Appnote738, Appnote 738
Copyright © by Maxim Integrated Products
Additional Legal Notices: http://www.maximintegrated.com/legal