Edited by Bill Travis

Make noise with a PIC

Peter Guettler, APS Software Engineering GmbH, Cologne, Germany

Building a stable noise generator for audio-frequency purposes requires only a few components. The circuit in **Figure 1** relies on linear-feedback shift registers and some simple software. An eight-pin Microchip (www.microchip.com) PIC12C508 controller (IC_2) with a short program generates pseudorandom noise at its output pin, GP0. A single controller is sufficient for simple applications. To obtain Gaussian-distributed noise, you can use a number of identical PIC controllers in parallel in a true realization of the central-limit theorem. (The central-limit theorem states that the sum of an infinite number of noise sources has Gaussian distribution, regardless of the individual noise distribution of each generator.) Using an infinite number of noise generators is impractical, but 10 to 16 are sufficient in most cases. And, because the smallest member of the PIC family is an inexpensive chip with low current consumption, the circuit is easy to realize.

All noise generators run the same program (**Listing 1** on the Web version of this Design Idea at www.edn.com). Perfectionists might program each PIC with an individual initial value for the shift register, but, because all controllers run uncorrelated with their own internal oscillators and start out of reset at different times, this measure is unnecessary. Op amp IC_{1A} sums and level-shifts the noise signals. Summing resistors R_1 and R_2 must have a value of 10 kΩ times the number of noise generators you use. The output signal of IC_{1A} feeds a −3-dB/octave filter to obtain pink noise. Buffer IC_{1B} decouples the filter and provides low output impedance. The signal amplitude is approximately 400 mV p-p with a flat spectral distribution of 20 Hz to 20 kHz. Closing S_1 or applying a low level at pin GP4 immediately stops all noise generators and freezes the prevailing signal amplitude. You can download the PIC software from the Web version of this Design Idea at www.edn.com.□

Publish your Design Idea in *EDN*. See the What's Up section at www.edn.com.

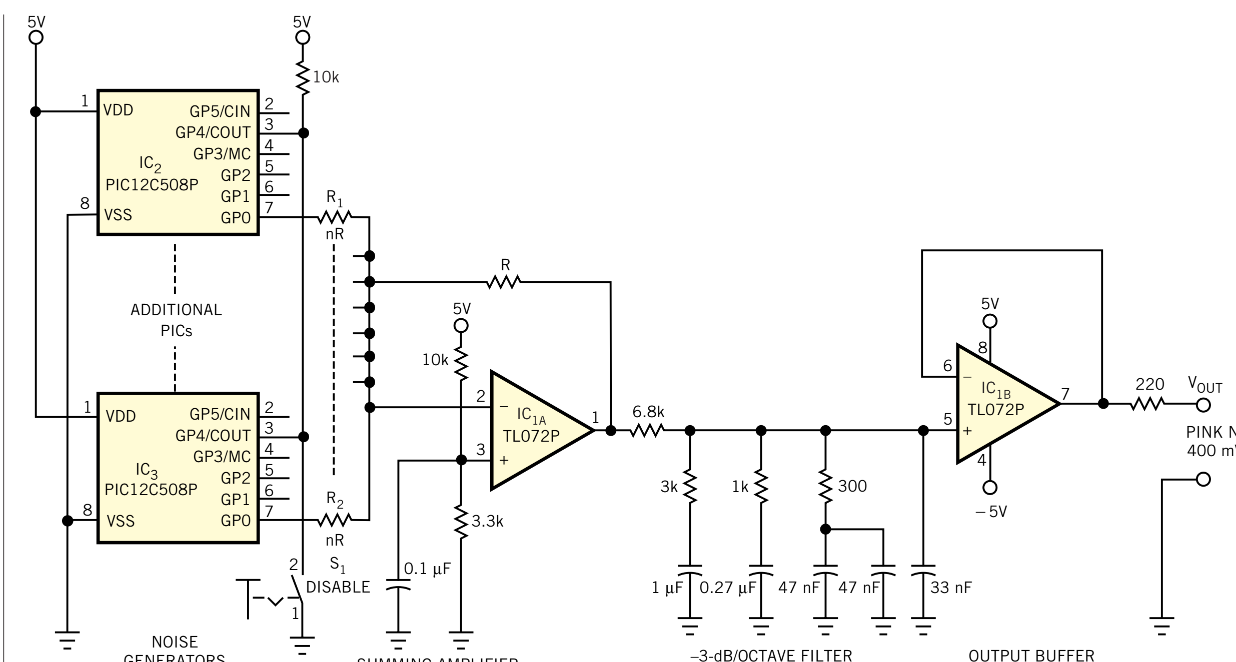


Figure 1 This simple circuit generates Gaussian-distributed noise for audio applications.

Circuit provides linear resistance-to-time conversion

S Kaliyugavaradan and D Arul Raj, Anna University, Chennai, India

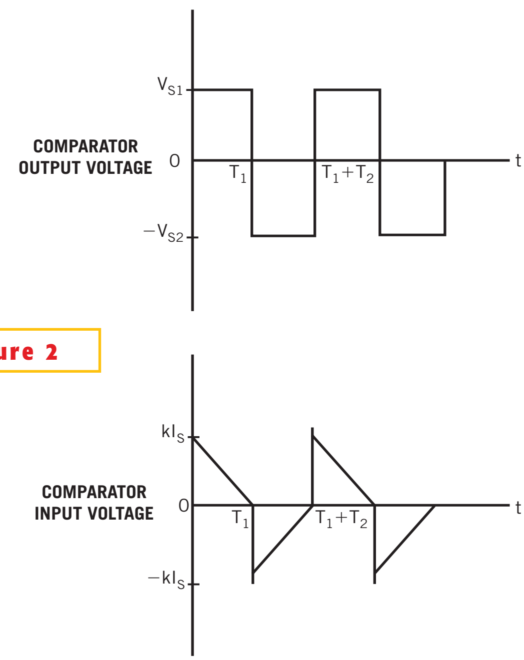
RESISTANCE-BASED transducers, such as strain gauges and piezoresistive devices, find common use in the measurement of several physical parameters. For applications in which digital processors or microcontrollers serve for data acquisition and signal processing, the transducer's response must assume a form suitable for conversion to digital format. It is desirable to convert the resistance change of such sensors into a proportional frequency or a time interval so that you can easily obtain an output in digital form, using a counter/timer. The circuit of **Figure1** linearly converts the sensor resistance, R_S, into a proportional time period. The circuit is essentially a relaxation oscillator, comprising a current source, a bridge amplifier, a comparator, and a flip-flop. The current, I_S, divides in the paths of R_1 and R_2 as if the two resistors were connected in parallel. Assuming ideal op amps, the circuit functions as an oscillator when R_X (R_4+R_S) is greater than R_1R_3/R_2.

The circuit produces waveforms at the input and output of the comparator, IC_2 (**Figure 2**). T_1 and T_2 are the time intervals for which the comparator's output assumes levels V_{S1} and $-V_{S2}$, respectively. The output voltage from IC_2, with its levels changed via a zener-diode circuit, serves as clock input to a D flip-flop. From the 7474 flip-flop, you obtain a square-wave output that is high and low alternately for a time period $T=4C(R_2R_X-R_1R_3)/R_1$. This equation indicates that the circuit converts a change in sensor resistance into a proportional time period ΔT with sensitivity $\Delta T/\Delta R_S=4C(R_2/R_1)$. The following salient features of **Figure 1** merit mention:

- The sensor is grounded; you can easily vary the conversion sensitivity by varying either R_1 or R_2.
- You can adjust the offset value, T_0 (about which changes in T occur because of a change in the sensor's resistance), by changing either R_3 or R_4 without affecting the conversion sensitivity.
- The offset voltages of the op amps alter T_1 and T_2 in opposite ways, such that their overall effect on $T(T_1+T_2)$ is not appreciable.
- Thanks to the current source, the output is largely insensitive to noise voltages in the line of the current source and to changes that occur in V_{S1} and V_{S2}.

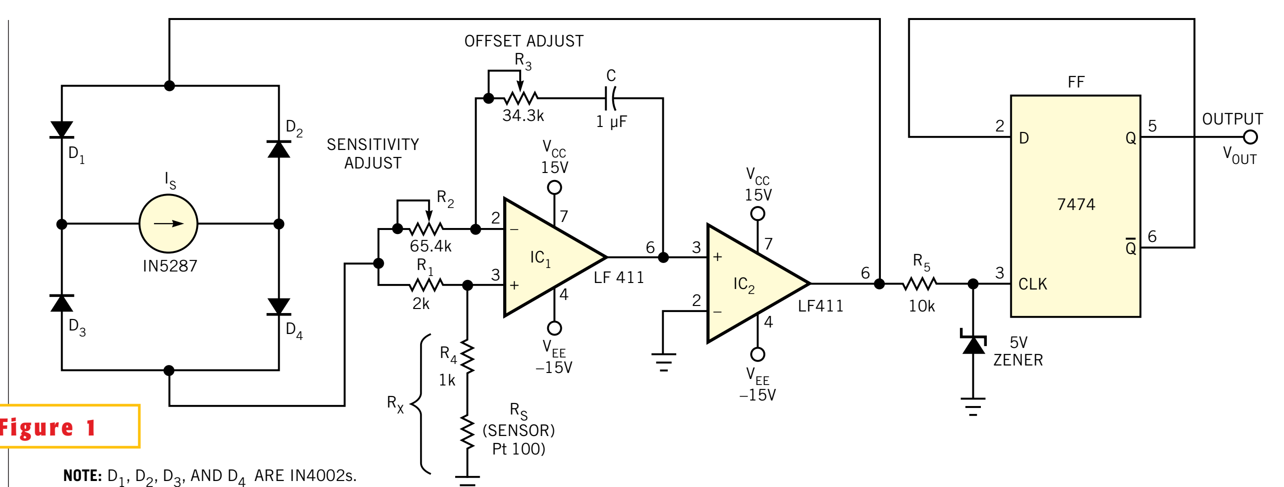
Consider the example of converting a

Figure 2



These waveforms represent the input and output of comparator IC_2.

Figure 1



This simple circuit converts a resistance reading to a time period.

Pt-100 (platinum RTD) sensor in the range of 119.4 to 138.51Ω, which corresponds to a temperature range of 50 to 100°C, into time periods of 10 to 12.5 msec. The design is simple. Because the current through the sensor is a fraction of I_S, I_S should be low enough to keep the self-heating error to an acceptably low level. This design uses an IN5287 current regulator; it provides an I_S of approximately 0.33 mA and has a dynamic impedance better than 1.35 MΩ. For a better current source, you could use a circuit based on a voltage-regulator IC. In the next step, with suitable and practical fixed values for R_1 and C, you adjust R_2 until you obtain the needed sensitivity: 130.82 μsec/Ω. Following this step, with a fixed R_4, you adjust R_3 to obtain the offset required in the output (T). **Figure 1** shows the values of components for this example. The resistors all have 1% tolerance and 0.25W rating, and C is a polycarbonate capacitor.□

PC-configurable RLC resonator yields single-output filter

Saurav Gupta, New Delhi, India

This Design Idea presents a versatile filter circuit for low-power-consumption instrumentation that you can program from your PC using the parallel port. The circuit uses analog switches and latches instead of digital potentiometers for the digital control (**figures 1** and **2**). By running simple software code on the PC, you can configure a single robust design to work as a lowpass, highpass, or bandpass filter, and you can also select the desired center frequency, ω_0 (**Listing 1**). Unlike a similarly controllable design (**Reference 1**), this design is a single-output-at-a-time filter. Many power-sensitive systems do not require simultaneous-filter functions.

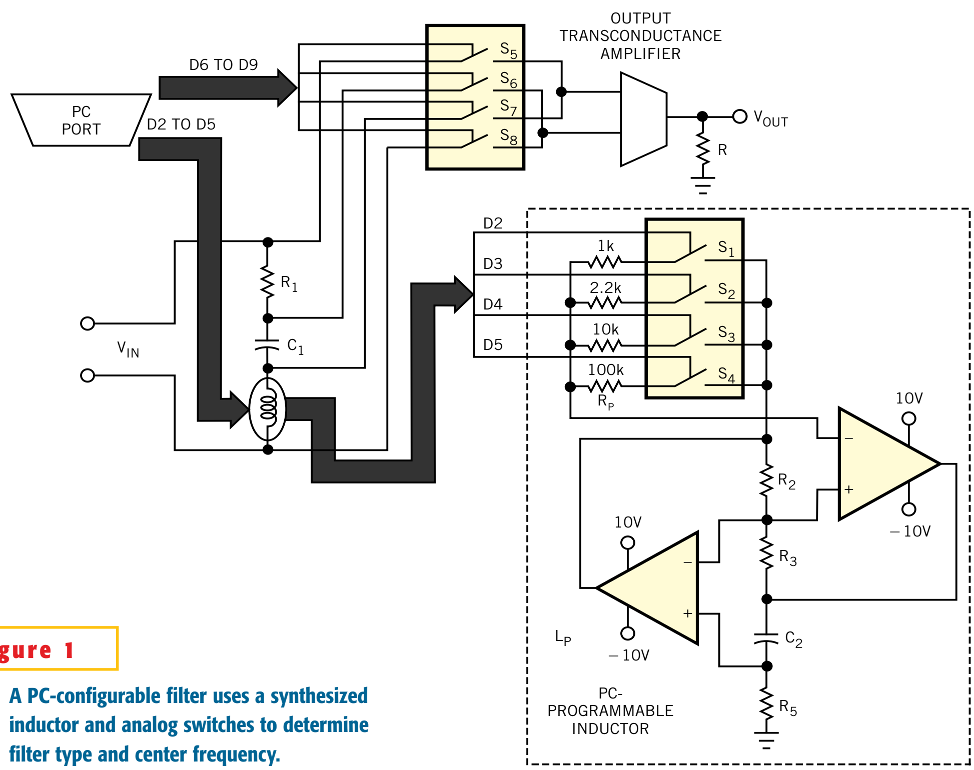


Figure 1

A PC-configurable filter uses a synthesized inductor and analog switches to determine filter type and center frequency.

The design exploits the fact that a series RLC resonator can provide various filter functions with its elements. Because the design is based on an RLC section, it is trivial to convert the design into a PC-controlled resonator. In **Figure 1**, the inductor, L_P, is implemented as a PC-controlled synthesized inductor. The value of the inductor is $L_P=C_2R_PR_3R_5/R_2$. Here, R_P can assume any of 15 possible values, depending upon the state of switches S_1 through S_4 (determined by PC-port data bits D2 through D5). The expression for the frequency is $\omega_0=(R_2/C_1R_PR_3R_5)^{1/2}$. You can thus effectively select 15 frequency values. (This design uses 12 values of practical interest.) Data bits D6 through D9 from the PC's parallel port set the state of analog switches S_5 through S_8. The state of the switches determines the type of filter.

Figure 3 shows the software-generated display for the circuit. This design uses a

TABLE 1–REPRESENTATIVE PORT SETTINGS AND FILTER PARAMETERS

Filter type/center frequency	Port setting								Hex output from PC
	D9	D8	D7	D6	D5	D4	D3	D2	
Lowpass/9.93 kHz	0	0	1	1	0	1	0	0	X34
Highpass/22.9 kHz	1	0	1	0	0	1	1	0	XA6
Bandpass/3.16 kHz	0	1	0	1	1	0	0	0	X58
Bandpass/37.3 kHz	0	1	0	1	0	1	1	1	X57

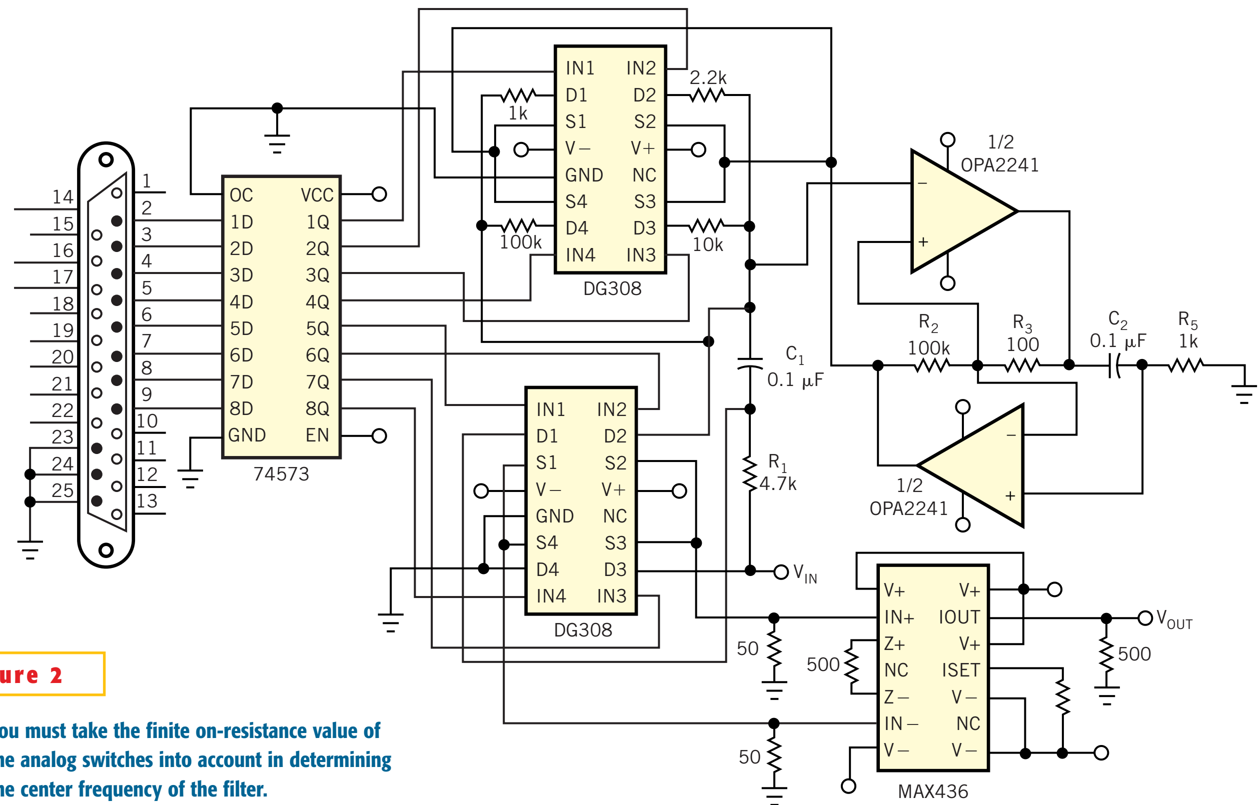


Figure 2

You must take the finite on-resistance value of the analog switches into account in determining the center frequency of the filter.

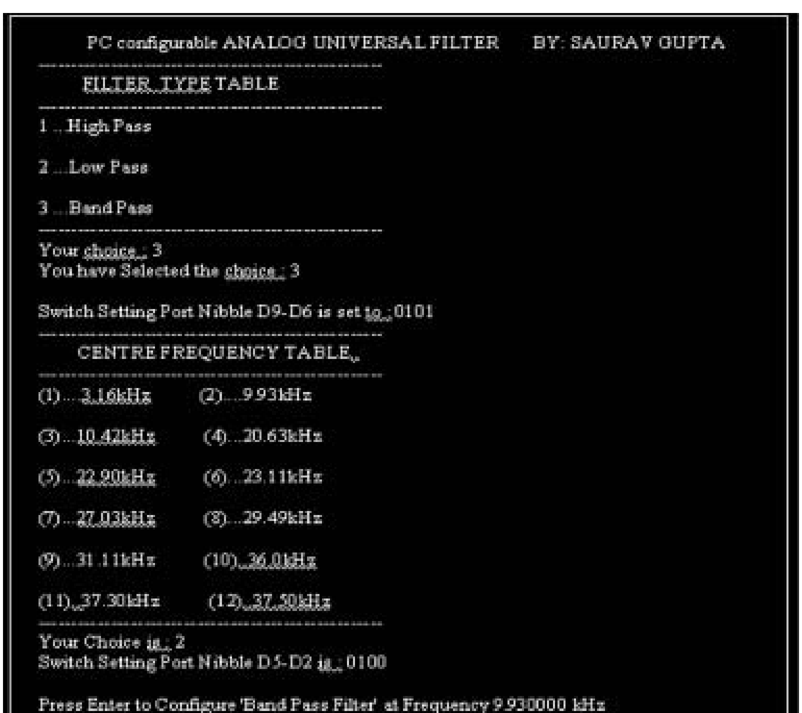


Figure 3

This user-friendly configuration screen allows you to determine filter type and frequency.

9.93-kHz bandpass filter for demonstration and testing. Increasing the number of analog switches can provide a wider range. Moreover, you could use additional switches for gain programmability. The 74573 latch provides the interface to the PC. **Table 1** shows the port/switch settings for a few frequency and filter-type selections. Note that the analog switches (DG308) have a finite operating on-resistance of approximately about 110Ω; you must take this resistance into account when you calculate the center frequency. For precision instrumentation, other switches are available with operating on-resistances as low as 30 to 50Ω. You can download **Listing 1** from the Web version of this Design Idea at www.edn.com.□

REFERENCE

1. Gupta, Saurav, and Tejinder Singh, "PC-based configurable filter uses no digital potentiometers," *EDN*, Jan 23, 2003, pg 76.

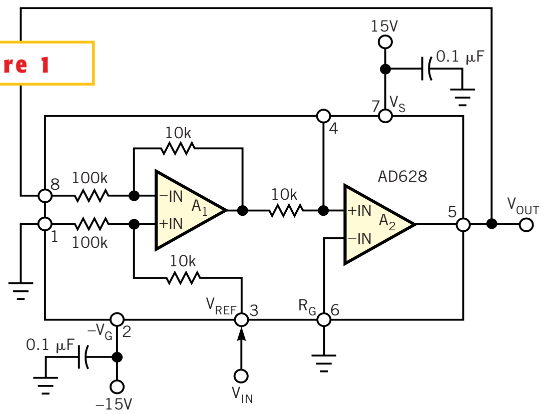
Single IC provides gains of 10 and −10

Moshe Gerstenhaber and Charles Kitchin, Analog Devices, Wilmington, MA

Real-world data-acquisition systems require amplifying weak signals to match the full-scale input range of an A/D converter. Unfortunately, when you configure them as gain blocks, most common amplifiers have both gain errors and offset drift. The typical two-resistor gain-setting arrangement found in many op-amp circuits has serious accuracy and drift limitations. With standard 1% resistors, the circuit gain can be off by as much as 2%. Also, the gain can vary with temperature, because each resistor drifts differently. You can use monolithic resistor networks for precise gain setting, but these components are expensive and consume valuable pc-board space. The circuits of **figures 1** and **2** offer improved performance and lower cost; they are also smaller. The single-μSOIC approach is the smallest available for this function, and the circuits require no external components. **Figure 1** shows an AD628 precision gain block connected to provide a voltage gain of 10. The gain block itself comprises two internal amplifiers: a gain-of-0.1 difference amplifier, A_1, followed by an uncommitted buffer amplifier, A_2. You can configure it to provide different gains by strapping or grounding the appropriate pins.

For a gain of 10, the input signal connects between the V_{REF} pin (Pin 3) and ground, instead of to the op amp's inputs. With the input tied to the V_{REF} pin, the

Figure 1



This circuit has a precise gain of 10 and uses no external components.

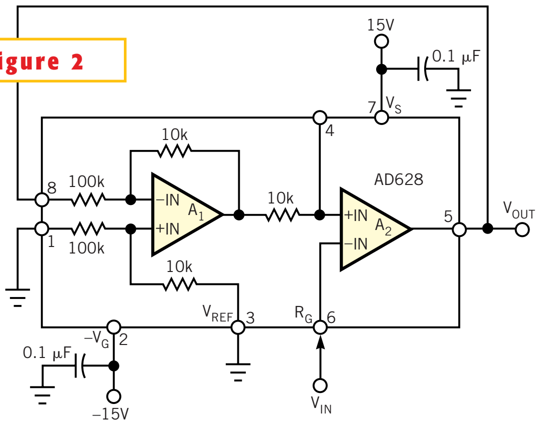
voltage at the noninverting input of A_1 equals V_{IN}(100 kΩ/110 kΩ), or V_{IN}(10/11). The inverting input of A_2 (Pin 6) is grounded; therefore, feedback from the output of A_2 forces the noninverting input of A_2 to be 0V. The output of A_1 must then also be at 0V. The voltage on the inverting input of A_1 must be equal to the voltage on the noninverting input of A_1, so both equal V_{IN}(10/11). Thus, the output voltage of A_2, V_{OUT}, equals

$$V_{OUT} = V_{IN} \times \frac{10}{11} \times \left(1 + \frac{100k}{10k}\right)$$
$$= V_{IN} \times \frac{10}{11} \times 11 = 10V_{IN},$$

providing a precise gain of 10 with no external components.

The companion circuit of **Figure 2** provides a gain of −10. This time, the input connects between the inverting input of A_2 (Pin 6) and ground. Operation is similar to that of **Figure 1**, but A_2 now inverts the input signal by 180°. With the V_{REF} pin grounded, the noninverting input of A_1 is at 0V, so feedback forces the inverting input of A_1 to 0V as well. Because A_1 operates at a gain of 0.1, the output of A_2 necessary to force the inverting input of A1 to 0V is $-10V_{IN}$. The two connections exhibit different input impedances. When you drive the V_{REF} input (Pin 3) for a gain of 10, the input impedance to ground is 110 kΩ; it is approximately 50 GΩ when you drive the noninverting input of A_2 (Pin 6) for a gain of −10. All resistors are internal to the gain block, so both accuracy and drift are excellent. These circuits have gain accuracy better than 0.1%, with a gain temperature coefficient lower than 5 ppm/°C. The −3-dB bandwidth is approximately 110 kHz with a 10-mV input and 95 kHz with a 100-mV input. Although ±15V supplies are appropriate, you may operate these circuits with dual supplies from ±2.25V to ±18V.□

Figure 2



This companion circuit to the one in Figure 1 delivers an accurate gain of −10.