design ideas

Edited by Bill Travis

Circuit adds programmability to sensor amplifier

Chuck Wojslaw, Catalyst Semiconductor, Sunnyvale, CA

The pressure-sensor amplifier circuit of **Figure 1** offers a number of advantages over the traditional approach using the classic three-op-amp instrumentation amplifier. The circuit can operate from a single supply and uses only two op amps and 1% resistors. If the reference voltage, V_{REF}, is 0V, the transducer gain for the circuit is

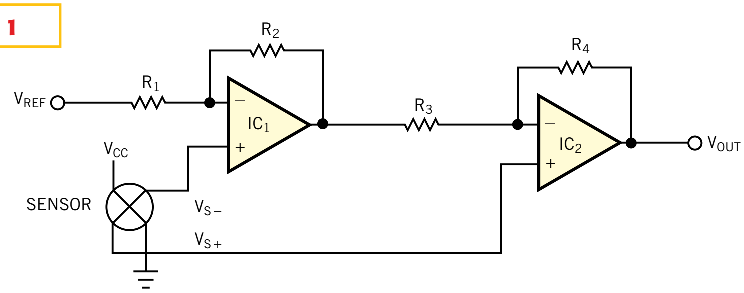
$$V_{OUT} = V_{S+}\left(\frac{R_3+R_4}{R_3}\right) - V_{S-}\left(\frac{R_1+R_2}{R_1}\right)\left(\frac{R_4}{R_3}\right).$$

To ensure equal gain for the two ground-referenced voltages making up the differential sensor voltage, you must impose the following condition:

$$\frac{R_1+R_2}{R_1} = \frac{R_3+R_4}{R_4}.$$

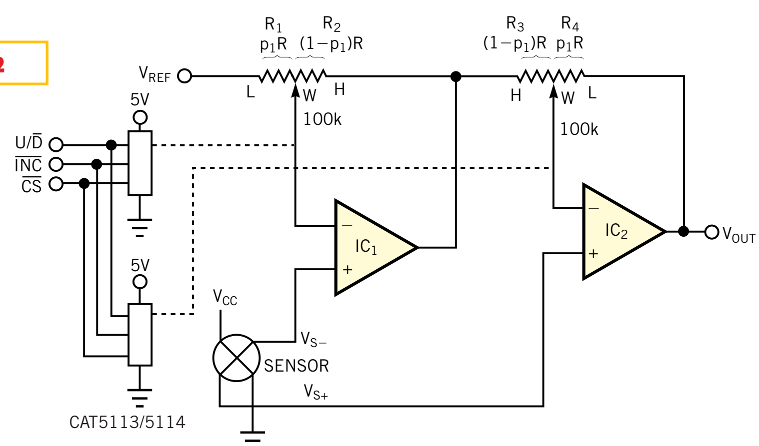
This restriction complicates the design and makes it difficult to add variability to the circuit. The circuit in **Figure 2** has the same advantageous topology as the original circuit, but it adds variability and programmability to the gain, simplifies the design expressions, and reduces the component count. Two ganged Catalyst digitally programmable potentiometers replace resistors R_1 through R_4. The potentiometers' wiper-to-low and wiper-to-high resistances establish the voltage gains of IC_1 and IC_2. Because the potentiometers are identical, the end-to-end resistances are the same, thus meeting the $(R_1+R_2)=(R_3+R_4)$ requirement. If you reverse the wiper-to-low and wiper-to-high resistances for IC_1 and IC_2, R_1 and R_4 are equal. If you mathematically model the wiper-to-low and wiper-to-high resistances as $(1-p_1)R_{POT}$ and $(p_1)R_{POT}$, the gain expression becomes

Figure 1



This circuit is a good sensor amplifier but suffers from a lack of variability.

Figure 2



An improved circuit features programmability and fewer components.

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

$$V_{OUT} = \frac{V_{S+} - V_{S-}}{1 - p_1}, \text{ FOR } 0 \le p_1 \le 1.$$

The factor p_1 models the relative position of the wiper as it moves from one end ($p_1=0$) of the potentiometer to the other end ($p_1=1$). The number of values p_1 can assume is a function of the number of taps on the potentiometer. The sensor gain is inversely proportional to $(1-p_1)$, is pseudo-logarithmic, and varies from less than unity to 31 for a 32-tap potentiometer, such as the CAT5114, and to 99 for a 100-tap potentiometer, such as the CAT5113. The voltage, V_{REF}, represents a programmable offset voltage for the signal-conditioning circuit. You can easily implement the reference voltage by using a digitally programmable potentiometer configured as a programmable voltage divider.

Is this the best Design Idea in this issue? Select at www.ednmag.com.

Add CAD functions to Microsoft Office

Alexander Bell, Infosoft International, Rego Park, NY

Microsoft Word has excellent drawing capabilities. You could use it effectively to perform some CAD tasks, such as schematic entry. Word is by far the most popular text processor on the market, and it would seem desirable for technical writers to be able to create a single integrated document file combining both text and graphics in an editable format. Of course, many third-party drawing or CAD packages are available, but most of them cost a significant sum. Also, the use of third-party software components could raise licensing issues. Moreover, in a shared development situation, you could face a document-interchangeability problem. Some of your coworkers may lose the ability to perform schematic editing if the third-party component is not installed on their computers. This Design Idea provides a great deal of flexibility, because anyone who has Word can create and modify drawings using standard Word features.

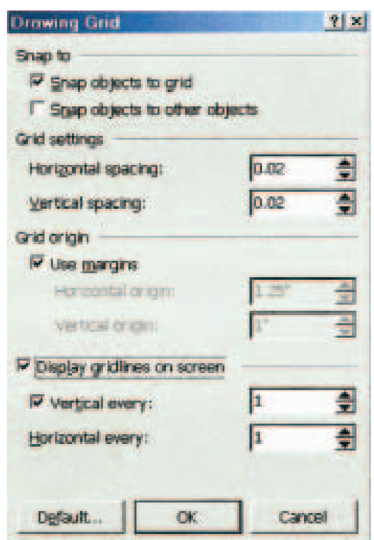


Figure 1 To display this dialogue box, click on the "Draw" button in the Drawing Toolbar, shown in Figure 2.

You can add some "convenience features" that can ease the schematic-entry process within Word. The first step is to create an Electronic Components Symbol Library and add it to the Clip Gallery. You can use those clips with Word or any other Microsoft Office application. These applications support drawing basically in the same way, with their built-in AutoShapes and Clip Art collection. In addition, you can use Clip Gallery with other graphics editors that support object linking and embedding. You can even use Clip Gallery as a stand-alone program. (For more information, refer to Microsoft Help for the Clip Gallery and related topics.) The task of creating the components library comprises several steps.

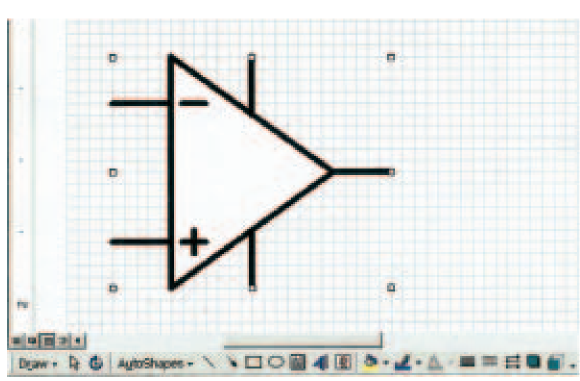


Figure 2 Create the component symbol by using Drawing Toolbar objects and then group them.

First, open a new file and then adjust the grid-line settings. You can reduce "Horizontal Spacing" and "Vertical Spacing" to 0.02 in. for good resolution. Make the gridlines visible by checking the box, "Display gridlines on screen" (**Figure 1**). To make the Drawing Grid dialogue box pop up, use the Drawing Toolbar button, labeled "Draw" in **Figure 2**. Set the scale to its maximum: 500%. Save the file as a Template by using the "Save As" option and choosing "Save as Type: Document Template" with extension .dot (for example, MyCad.dot) in the Word Templates default folder and then close the file. Now, when you open the new file for drawing, use this template as follows: Go to the "File" menu item, click the "New" subitem, and then choose the "MyCad" template. Note that the grid lines are not printable objects; they are visible only on the screen. You could type the text, draw the schematic, and then turn the grid lines off in the final document.

The next step is to create the symbol and add it to the Library. Use Drawing Toolbar objects via the corresponding buttons (**Figure 2**). After you have finished the drawing, select

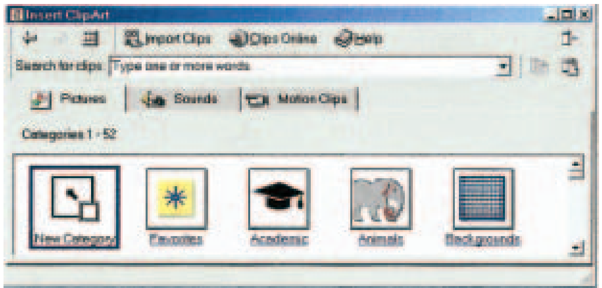


Figure 3 You can use the Clip Gallery with many Microsoft Office applications and even in stand-alone mode.

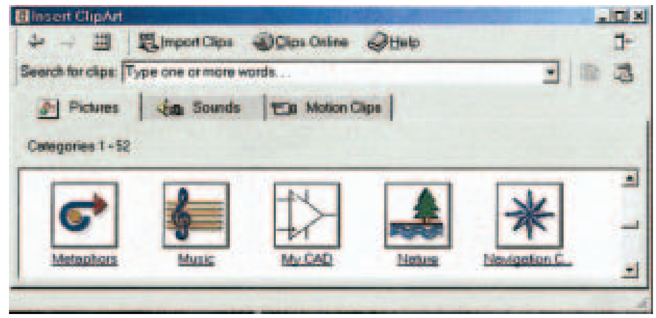


Figure 4 Add the new category, "MyCad," which serves as your Components Library.

all of its constituent parts by pressing the Shift key and holding it down while making the selection and then right-click the mouse on the pop-up menu, select the "Grouping" item, and click "Group." Now, Word treats the created component symbol as Clip or Auto Shape Object, which you can move, resize, and format as a whole image. To edit a part of the symbol image, you should first ungroup it, make the necessary changes, and then group it again. To add symbols to the Library, you should first create a new category in the Clip Gallery, which serves as a "container" for your graphic images and then copy and paste the symbols into the Gallery. The procedure is as follows:

1. Open the Clip Gallery by clicking on the icon "Insert Clip Art" located at the Drawing toolbar.
2. Click on the "New Category" icon (**Figure 3**) and type the name (for example, MyCad) for your Library when prompted, then click "OK."
3. Check to see whether the new "MyCad" category icon appears in the Clip Gallery window (**Figure 4**) and then click on the icon to open it.
4. Select the component you want to add to the Library and copy it to the clipboard (the shortcut for the "Copy" command is Ctrl-C).
5. Paste the component to the "MyCad" Category (the shortcut for the "Paste" command is Ctrl-V). Type the name for the symbol when prompted and then click "OK."
6. Repeat steps 2 to 5 for any components you want to add.

Figure 5 shows a variety of symbols in the MyCad Category. The usage of the Components Library is straightforward and similar to the usage of built-in Clips. To insert the symbol image into you document, you could use the Clip Gallery's standard "Import Clips" feature or the "Drag-and-Drop" option, depending on your personal preference. The Clip Gallery has a Help utility you can use to obtain more information. You could further improve ease of use by adding macros to automate the most common tasks, such as resizing, rotating, or flipping objects or adding labels, for example. You can assign the macros to shortcuts (certain key combinations). However, only qualified users, who have extensive experience dealing with VBA/macros, should attempt these operations. Also, you must be aware of the security issues associated with the use of macros in the applications:

- Some macros can perform potentially dangerous and harmful actions, and some of them may contain viruses. Use the macros at your sole risk without warranty of any kind.
- When using the Clip Gallery in a shared-development environment, and also with regard to network-installation or distribution issues, refer to the licensing/legal information, "Legal restrictions for using clips provided in the Clip Gallery" in the in the Help menu and the Microsoft Office End-User Legal Agreement, because certain restrictions could apply.

You can download several sample macros from the Web version of this article at www.ednmag.com. You can use them to Add Label to the selected graphic component, to resize the component or to rotate it 45° clockwise. You can store macros in the Macro Module that you add to the default Normal.dot template, thus affecting all opened documents. Alternatively, you can store the macros in the Template Macro Module within the MyCad.dot file. In this case, the macros are available only for opened documents based on the MyCad template file. For more information on creating and storing template files, refer to the Microsoft online Help features.

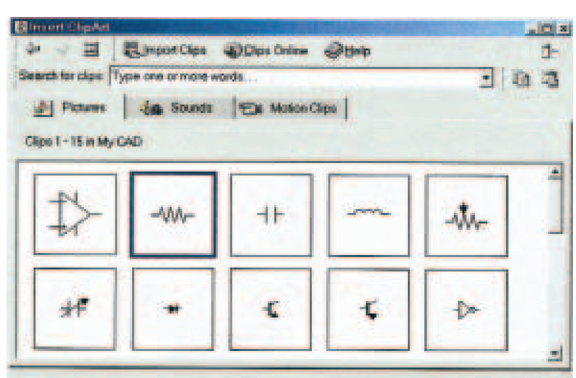


Figure 5 After adding the components to the Clip Gallery, you can use them in the same way you'd use built-in Clips.

Is this the best Design Idea in this issue? Select at www.ednmag.com.

Single IC forms pseudorandom-noise source

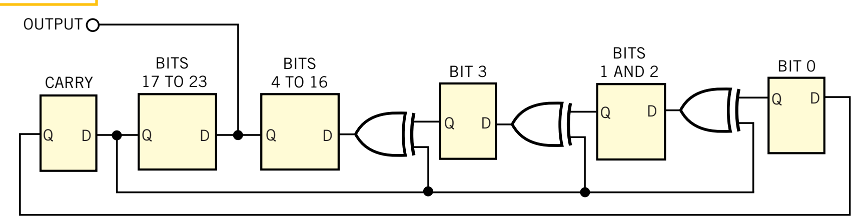
Steve Ploss, Veridian, Dayton, OH

Trying to find a single IC noise source can be frustrating. National Semiconductor once made such a noise source for audio applications, but it's now hard to find. This situation leaves the designer with several choices, most of which involve using several ICs. You can take the analog approach of using a lot of gain to amplify diode-avalanche noise, or you can design a linear-feedback-shift-register pseudorandom-noise source using multiple CMOS ICs. Alternatively, you could use a single-chip microcontroller, such as Microchip's 12C508. The classic approach to generating a pseudorandom-noise sequence is to model the linear-feedback-shift register in software. The method involves testing and XORing together multiple bits to provide the single-bit feedback into the shift register. One problem with this approach is that it doesn't yield the highest possible bit rate. The approach in **Figure 1** is more optimum for using the lowest number of machine cycles per loop.

The approach is similar to the one that CRC bit-shift algorithms take. These algorithms test only a single bit but XOR multiple bits in parallel. A microcontroller quickly accomplishes the task. The bit-test and XOR operations require only three instruction cycles to complete. **Listing 1** shows the assembly code for the complete noise source. The entire loop takes only 10 instruction cycles. You can download **Listing 1** from the Web version of this Design Idea at www.ednmag.com. The circuit in **Figure 2** produces an output rate of 100 kHz for the 12C508 with its internal 4-MHz oscillator. The shift register's length is effectively 25 bits, three bytes plus a carry bit. For a maximal-length sequence such as this, the pattern does not repeat itself for 5 minutes and 35 sec. If you desire higher bit rates, you can simply use a 20-MHz part with a crystal to obtain an output rate of 500 kHz. Another trick for increasing the rate is to use a part with an 8-bit output port and use that port in place of the Hibyte register. Then, the commands "movf Hibyte,W" and "movwf GPIO" are unnecessary. The result is a loop time of eight instruction cycles and a 25% increase in output rate. Be sure to configure the port pins as all outputs.

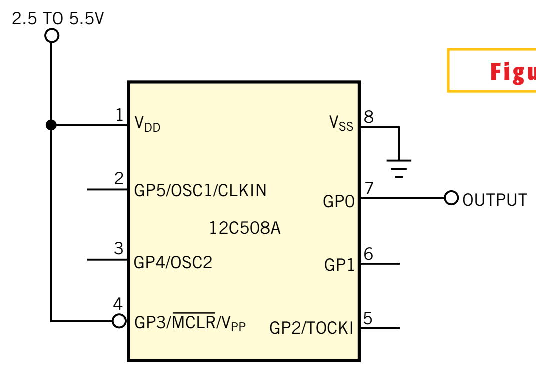
Is this the best Design Idea in this issue? Select at www.ednmag.com.

Figure 1



The shift register's length in this pseudorandom-noise generator is effectively 25 bits.

Figure 2



The connections for the 12C508 microcontroller are exceedingly simple.

LISTING 1—12C508 ASSEMBLY CODE FOR PSEUDORANDOM-NOISE GENERATOR

```
;
;*************************************************************************
;    Pseudorandom noise generator
;*************************************************************************
;     12/13/01 Stephen J. Ploss
;
;

        LIST    p=12C508A

#include "P12C508A.INC"

        __CONFIG _CP_OFF & _WDT_OFF & _IntRC_OSC & _MCLRE_OFF

        cblock    0x07                ;Start of GP Regs

        Lobyte                        ;Shift Registers
        Midbyte                       ;       "
        Hibyte                        ;       "

        endc

start   movlw     0xC0                ;No Wake on pin change,
                                      ;No pull ups, ignore timer

        option
        movlw     0xFE                ;Only GP0 as output
        tris      GPIO

        ;Start the PN Generator
main    movlw     0xFF                ;Initialize shift reg's
        movwf     Lobyte
        movwf     Midbyte
        movwf     Hibyte

loop    rlf       Lobyte,F            ;Perform shift
        rlf       Midbyte,F
        rlf       Hibyte,F
        movlw     0x1A                ;Polynomial
        btfsc     STATUS,C            ;Test Carry
        xorwf     Lobyte,F            ;XOR if set

        movf      Hibyte,W            ;Get Hibyte
        movwf     GPIO                ;Send LSB to output
        goto      loop

        END
```

LED dimmer uses only two lines

Jerry Wasinger, WallyWare Inc, Norcross, GA

THE CIRCUIT IN **Figure 1** provides 32 steps of brightness control—from 0 to 100%—for a backlight or instrument panel, using just two general-purpose-microprocessor signals. In addition, the circuit requires little board space, because it uses only three SOT-23s and one µMax package. Although **Figure 1** shows the circuit driving white LEDs, the load could also be a dc motor or an incandescent lamp. The basis of the circuit is a modified Schmitt-trigger relaxation oscillator (**Figure 2**). The output of IC_1 is high when

$$\tau_H = CkR_P \ln\left[\frac{V_{CC}-V_D-V_N}{V_{CC}-V_D-V_P}\right],$$

where k is the wiper position, V_{CC} is the supply voltage, V_D is the diode voltage, and V_N and V_P are the threshold voltages of the Schmitt trigger. The output of IC_1 is low when

$$\tau_L = C(1-k)R_P \ln\left[\frac{V_D-V_P}{V_D-V_N}\right].$$

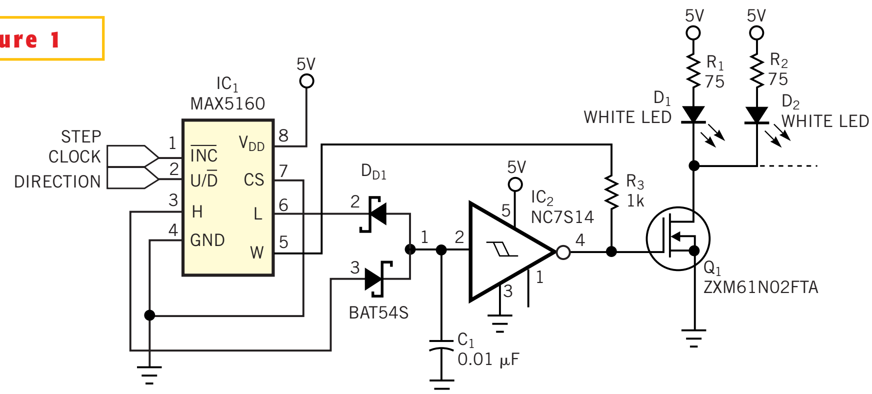
The period and duty cycle are $T=\tau_L+\tau_H$, and $DC=100\tau_H/T$, where DC is duty cycle. Initial inspection of the foregoing equations may make the substitutions to solve for the duty cycle appear to be a tedious, algebraic exercise. But, when you substitute V_{CC}, V_D, V_N, and V_P into the two logarithmic terms, their results are close in value. So, you can simplify the expanded solution to the duty-cycle equation to

$$DC = 100\left[\frac{kR_P}{(1-k)R_P+kR_P}\right] =$$

$$100\frac{kR_P}{R_P} = 100k.$$

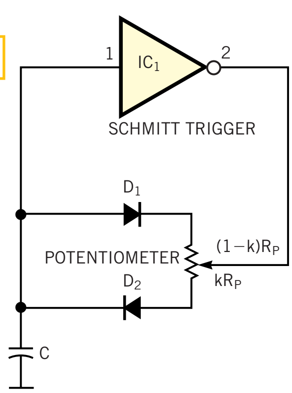
Based on the above considerations, you can see that the duty cycle of the circuit is not only linear, but also independent of the selected components. The component selection affects only the center frequency of oscillation. An approximate equation for the operating frequency is $f=1.75/(CR_P)$. This circuit uses a MAX5160 digital potentiometer that has a full-scale resistance of 200 kΩ in combination with a 10-nF capacitor. This combination results in an operating frequency of approximately 875 Hz. Note that slight mismatches of the two logarithmic terms cause the frequency to vary slightly as you adjust the duty cycle. **Figure 3** shows a plot of duty cycle versus the potentiometer's tap position; **Figure 4** shows the variation in the oscillation frequency with tap position.

Figure 1



By controlling the duty cycle of a Schmitt trigger, you obtain 32 steps of brightness in an LED display.

Figure 2



The MAX5160 digital potentiometer controls the duty cycle of the Schmitt trigger.

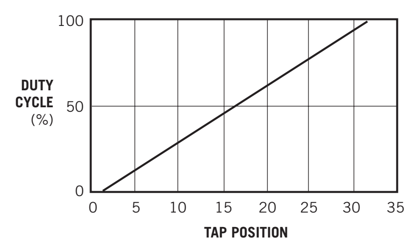


Figure 3 The duty cycle of the Schmitt trigger is a linear function of the potentiometer's tap position.

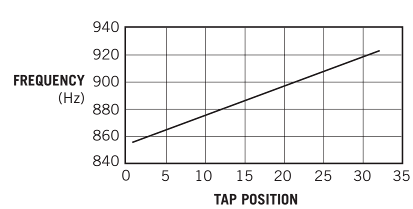


Figure 4 Frequency also varies with the tap position, but with no effect on performance.

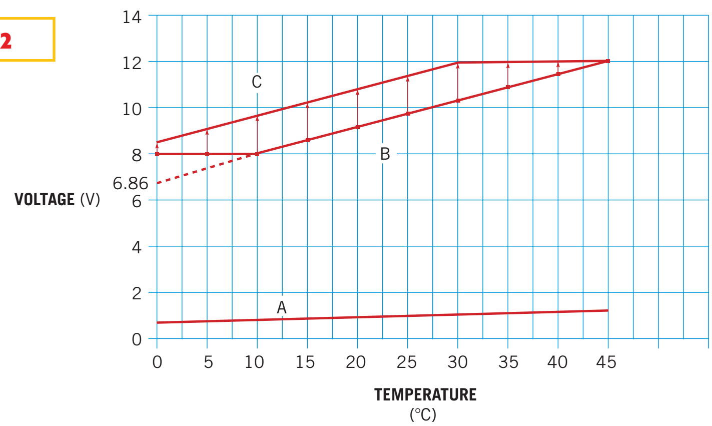
Is this the best Design Idea in this issue? Select at www.ednmag.com.

Circuit generates fan-speed control

Jim Christensen, Maxim Integrated Products, Sunnyvale, CA

Fan noise is becoming a significant issue as electronic equipment increasingly enters the office and the home. Noise is proportional to fan speed, and the airflow—hence, fan speed—necessary for cooling is less at low ambient temperatures. Because ambient temperature is lower than the upper design point most of the time, a fan can run slower, making it easier on the ears. Fan-control circuits range from simple switches that boost the speed from low to high, to digital, proportional speed-control designs. High/low-speed switches are inexpensive, but the sound of sudden speed changes can be annoying. Digitally controlled fans perform well, but the circuitry is costly, and the system must include a serial bus. As an alternative, consider a low-cost, self-contained analog circuit for fan-speed control (**Figure 1**). You can easily adjust the circuit for any desired linear relationship between the fan voltage and temperature (**Figure 2**, curves B and C). We plotted measured data points against the desired voltage in **Figure 2**.

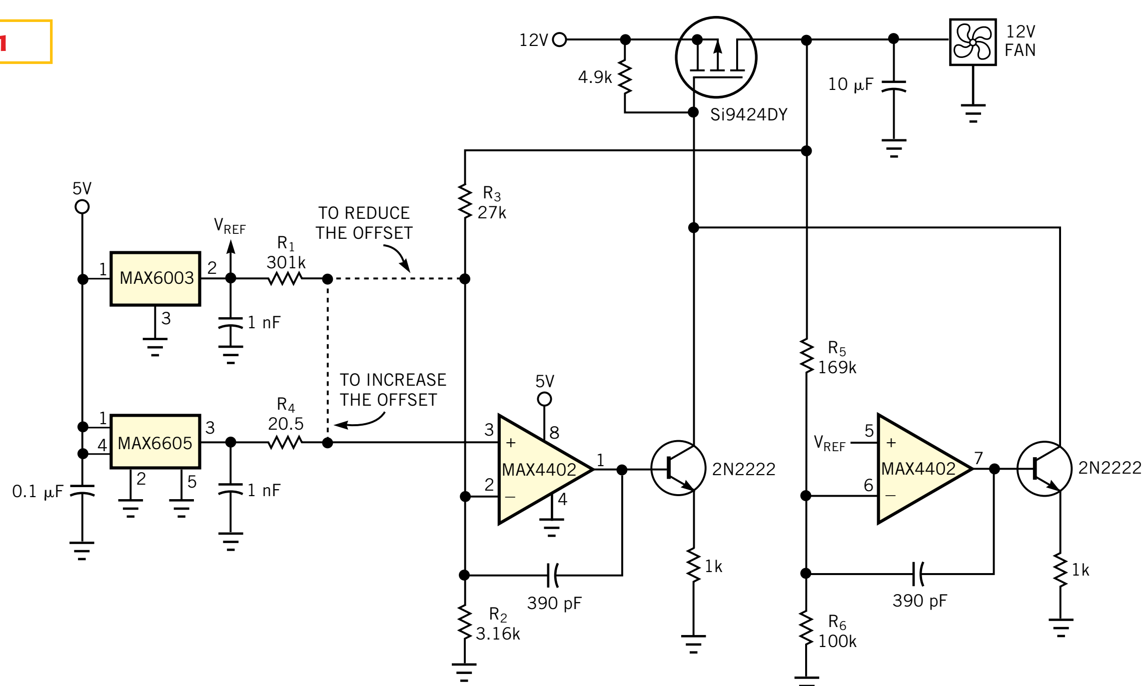
Figure 2



These curves illustrate voltage output versus temperature for the circuit in Figure 1.

Curve A in **Figure 2** represents the output of a MAX6605 analog temperature sensor versus temperature in degrees Celsius: V_{SENSOR}=0.0119 V/°C+0.744V. Curve B relates the fan voltage to tem-

Figure 1



This circuit delivers a continuous and linear fan-control voltage that is proportional to temperature.

perature and combines a minimum "floor" voltage of 8V with a sloping line: $V_{FAN}=0.114V/°C \times T+6.86V$, where T is the system temperature. The floor voltage ensures fan rotation at low temperatures. Above 10°C, the voltage increases with a slope of 0.114V/°C until it reaches full value at 45°C. Simply amplifying the MAX6605's output does not provide the 8V floor voltage. Moreover, the gain (9.58=0.114/0.0119) needed to obtain the fan-voltage slope is not the same gain (9.22=6.86/0.744) needed to obtain the y-intercept point.

To transform Curve A into Curve B, you must subtract a voltage offset from the temperature sensor's output and then multiply the result by a constant. The circuit in **Figure 1** performs this operation. You connect the dotted line labeled "To reduce the offset." One op amp creates the sloped line, and the second op amp creates the floor voltage. The op amps' outputs connect to transistors in such a way that the op amp demanding a higher output voltage dominates. The following equations allow you to determine resistor values:

For the condition in which $R_2 << R_1$, R_1=any reasonable value; $R_2=R_1(A_V V_{TEMP0}-V_{Y\text{-}INTB})/[(A_V-1)(V_{REF}-V_{TEMP0}+V_{Y\text{-}INTB}/A_V)]$; and $R_3=R_2(A_V-1)$, where $A_V=0.114/0.0119=9.58$=the ratio of the desired slope in volts per degrees Celsius to that of the sensor; $V_{TEMP0}=0.744V$=the temperature-sensor voltage at 0°C; $V_{Y\text{-}INTB}=6.86V$=the y-intercept indicated by the desired (extrapolated) temperature curve; and $V_{REF}=3V$=the reference voltage.

Thus, choosing $R_1=301$ kΩ lets you calculate $R_2=3.158$ kΩ and $R_3=27.09$ kΩ. The closest 1% values are 3.16 and 27 kΩ, respectively. The following equation lets you calculate the floor voltage: $R_5=R_6(V_{FLOOR}-V_{REF})/(V_{REF})$, where R_6 equals any reasonable value. $V_{FLOOR}=8V$, the desired minimum output voltage. Thus, choosing $R_6=100$ kΩ lets you calculate $R_5=169$ kΩ. In some cases, the required offset gain is greater than the required slope gain, so you must increase the temperature sensor's natural offset. For a desired temperature, Curve C, expressed as $V_{FAN}=(0.114\ V/°C)(T)+8.5V$, the gain (slope) of $A_V=9.58$ is the same as for Curve B, but the required offset gain is (8.5/0.744)=11.42 is greater. You therefore use the "To increase the offset" version of the circuit in **Figure 1**. The following equation applies in such cases: $R_4=R_1(V_{Y\text{-}INTC}/A_V-V_{TEMP0})/(V_{REF}-V_{INTC}/A_V)=20.41$ kΩ, where $V_{Y\text{-}INTC}=8.5V$ is the intersection of the desired temperature curve with the y-axis. For $R_1=301$ kΩ, the closest 1% value for R_4 is 20.5 kΩ.