design ideas

Edited by Bill Travis

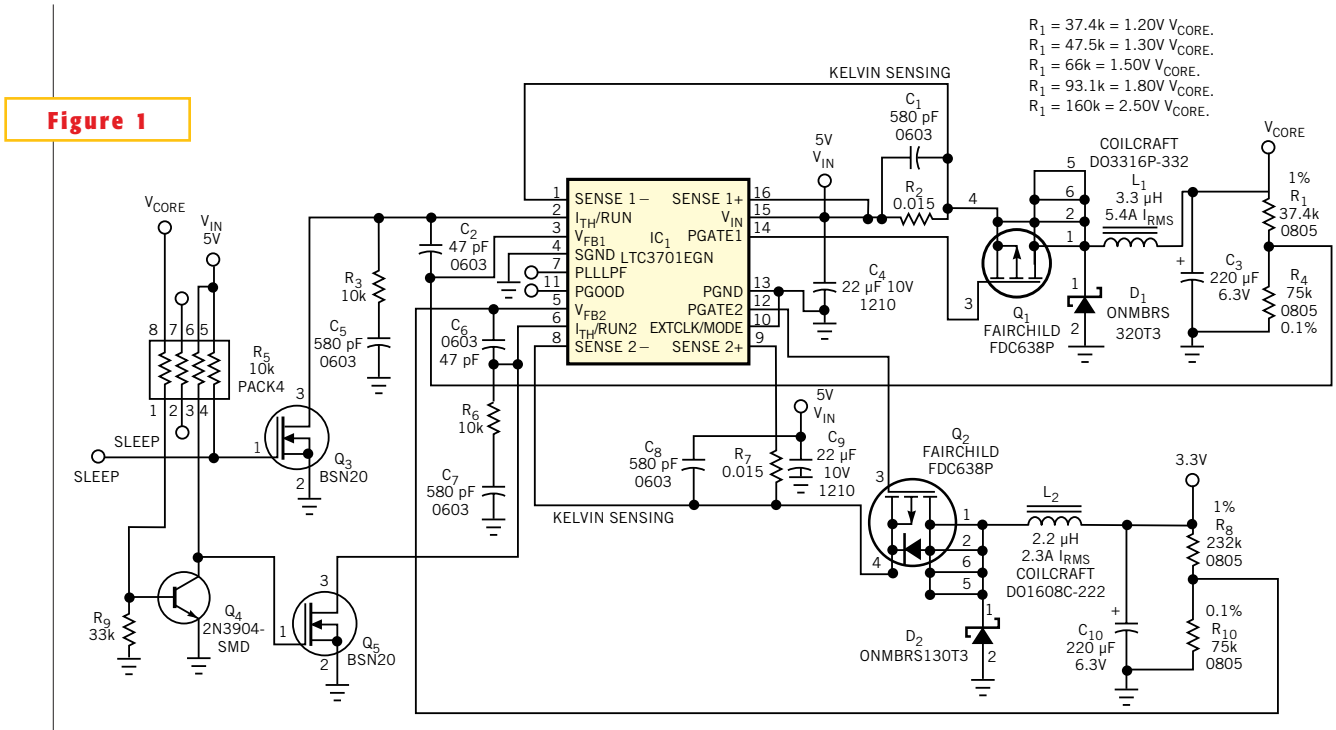
Scheme adds sequencing and shutdown control to regulator

Said Jackson, Equator Technologies Inc, Campbell, CA

Modern microprocessor- or FPGA-based circuits require separate and independent power-supply voltages for the core and the I/O circuits. Some devices require stringent control of the turn-on characteristics and sequencing of these multiple power supplies to avoid internal parasitic current flows and consequent latch-ups. Although regulators exist with specific soft-start and shutdown inputs, it may be more cost-effective to use regulators that do not inherently provide these features and to add these features with external discrete devices. This Design Idea shows how to use an inexpensive Linear Technology (www.linear.com) LTC3701 dual switching regulator to provide a sequenced, and standby-controlled, power supply for an Equator Technologies (www.equator.com) broadband-signal processor. You can also adjust the circuit for FPGA or generic microprocessor applications. The features of the circuit in **Figure 1** increase the regulator's stability beyond what you can achieve with the standard Linear Technology application-note circuit.

The LTC3701 switching regulator, IC_1, provides two independently adjustable output voltages with very high voltage accuracy at a cost compatible with consumer-type applications. Because of cost constraints, it does not provide the soft-start or shutdown features present in other switching regulators. This design adds three discrete transistors to the conventional regulator circuitry to provide both arbitrary power-on-sequencing control and a simultaneous-shutdown feature. Q_3, Q_4, and Q_5 are inexpensive discrete

Figure 1



Adding a few transistors to a switching regulator adds power-sequencing and shutdown control to a power supply.

devices that control the voltage on the regulator's I_{TH}/Run pins. The I_{TH}/Run pins of IC_1 provide an external compensation to the internal feedback loops; they can also serve to shut down the device when you pull them to ground. A microprocessor's TTL/CMOS-compatible input signal (Sleep) controls the power state of the circuit. You can put the circuit into shutdown mode by either letting the Sleep pin float high or pulling it higher than approximately 1.5V. Q_3 then connects the I_{TH}/Run1 pin to ground, which causes the V_{CORE} core-voltage supply to shut off. The V_{CORE} voltage then drops toward ground, and Q_4 stops conducting when V_{CORE} falls below approximately 0.8V. The gate of Q_5 pulls to the 5V unregulated input voltage, and Q_5 shorts the I_{TH}/Run2 pin to ground, which turns off the 3.3V regulator. The circuit is now in standby mode, and both power supplies are off.

Pulling the Sleep pin lower than approximately 0.8V turns on the power supply and sequences the voltages in the following manner: Q_3 stops conducting, and the voltage on the I_{TH}/Run1 pin can rise, thanks to internal current sources in IC_1. The V_{CORE} voltage regulator then starts to operate, and V_{CORE} rises to its set voltage, 1.2V by default. Q_4 starts conducting as soon as V_{CORE} rises above approximately 0.8V. This action turns off Q_5 and allows the I_{TH}/Run2 pin voltage to start rising. The 3.3V power supply thus turns on. The combined effect of driving Q_4 and Q_5 from the V_{CORE} voltage is that the 3.3V I/O voltage always turns on only after the V_{CORE} voltage attains an established level. The end result is to sequence the power supplies over a period of 4 msec (**Figure 2**).

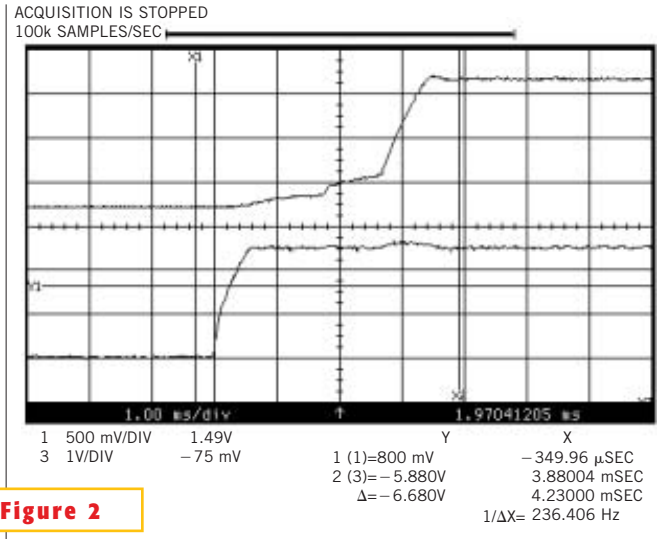


Figure 2 The 3.3V supply turns on several milliseconds after V_{CORE} attains an established level.

The circuit is symmetric, and changing the base drive of Q_4 and interchanging the drain signals of Q_3 and Q_5 reverses the sequencing order of the power supplies for chips that require the I/O voltage to rise before the core voltage. You can adjust the value of R_1 to generate any core voltage above approximately 1V. You may need to adjust the value of R_9 if your design requires core voltages below approximately 1V. You can replace Q_3 and Q_5 by potentially cheaper industry-standard 2N2007 devices at the expense of slightly higher capacitive loading on the I_{TH}/Run pins of IC_1. C_2 and C_6 are compensation capacitors that the Linear Technology literature does not mention but that are highly effective in preventing subharmonic oscillation arising from dynamic current loading on the outputs. (See the Linear Technology Web site for information on subharmonic oscillation.)

The gate-drain-source capacitance of Q_3 and Q_5 also add to the stability of the loop filter. Note that sequencing the turn-on ramps of the power supplies also has the benefit of reducing the inrush current into the power supply by staggering this current and preventing simultaneous current loading of the primary bypass capacitors by both power supplies. The selected component values allow for more than 2A of current on the 3.3V line and more than 3.5A of current on the V_{CORE} line.□

MathCAD functions perform log interpolation

James Bach, Delphi Delco Electronics Systems, Kokomo, IN

MathCAD provides a number of interpolation and curve-fitting functions, so that, given a set of X-Y data points, you can estimate the Y value for any given X coordinate. Unfortunately, these functions work poorly with data that is to be displayed in a nonlinear (logarithmic) manner. Examples of these functions are:

- Log-Lin: phase/magnitude-versus-frequency (Bode plots);
- Log-Log: impedance-versus-frequency (reactance plots); and
- Lin-Log: impedance-versus-tem-

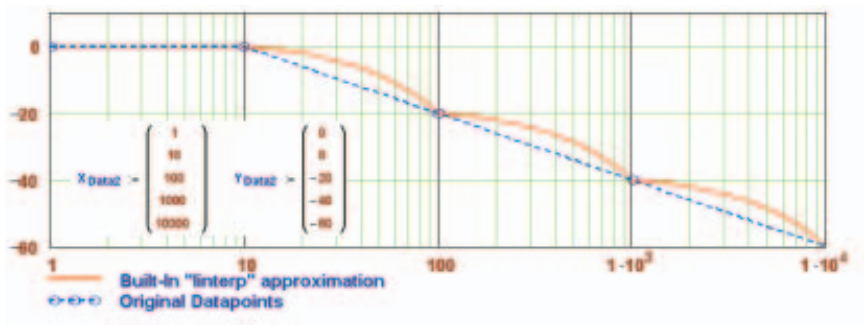


Figure 1 At X coordinates between data points, MathCAD's linterp function creates a bulging effect.

(continued on pg 94)

perature (thermistor data).

Using the built-in "linterp" function, MathCAD estimates and plots the data (**Figure 1**). As you can see, at X coordinates between the original data points, the "linterp" function creates a "bulging" effect. The following trio of simple interpolation functions allows the correct interpolation of nonlinear data on its appropriate scale. These routines function by prewarping the incoming-data matrices before feeding them into the existing "linterp" function; for logarithmic Y-axis functions, you raise 10 to the result of the "linterp" function to restore the values to the proper decade:

$$LogLogInterp(Vx, Vy, X) = 10^{linterp(\overrightarrow{\log(Vx)}, \overrightarrow{\log(Vy)}, \log(X))} \quad Log\ X / Log\ Y$$

$$LogLinInterp(Vx, Vy, X) = linterp(\overrightarrow{\log(Vx)}, Vy, \log(X)) \quad Log\ X / Lin\ Y$$

$$LinLogInterp(Vx, Vy, X) = 10^{linterp(Vx, \overrightarrow{\log(Vy)}, X)} \quad Lin\ X / Log\ Y$$

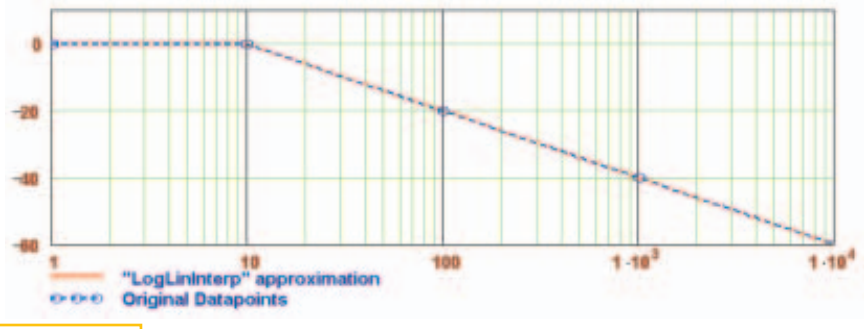


Figure 2 Using the LogLogInterp function, the bulges in Figure 1 disappear.

Using the newly created LogLogInterp function, the straight-line data is displayed (**Figure 2**).□

Scheme improves on low-cost keyboard

Martin O'Hara, Telematica Systems Ltd, Cranfield, Bedfordshire, UK

You can easily improve on a previous Design Idea to produce a slightly simpler resistor arrangement with better timing balance between switches, using a single resistor value (**Reference 1**, **Figure 1**). The use of a single resistor value, R_S, in a series chain for the switch resistors gives the timing parameters a simpler format and should reduce bill-of-materials cost. The timing balance between switches should now also be more even. The improved balance eases extending the keyboard for adding key inputs. The additional benefit of this arrangement is to make the circuit easier to adapt for faster or slower microprocessors, because you can easily adapt the circuit by changing the single switch resistor or capacitor values to alter the charge-discharge characteristics (**Figure 2**). It can also make building the circuit into a keypad housing easier, especially if you use membrane keys. The entire circuit is an SMD assembly with just two component types: switch and resistor. In the original idea, in cases of multiple keys being pressed, the timing is some odd multiple of parallel resistors and could accidentally represent a key that was not selected. With the arrangement of **Figure 1**, the lowest order key dominates; hence, the keypad has hard-wired priority setting and always results in a selected key timing, and no intermediate timing period should occur.□

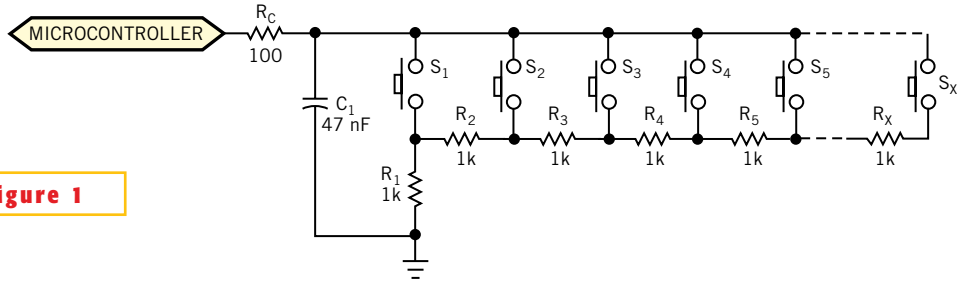


Figure 1 This simple keypad arrangement uses a single resistor value to select the switches.

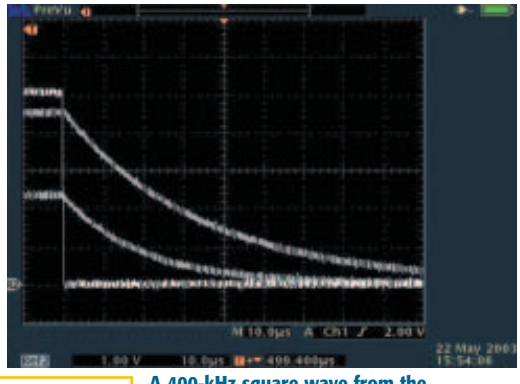


Figure 2 A 400-kHz square wave from the microprocessor shows no key, key 1, and key 5 pressed.

Reference

1. Thevenin, Jean-Jacques, "Novel idea implements low-cost keyboard," *EDN*, April 3, 2003, pg 69.

ADC interface conditions high-level signals

Moshe Gerstenhaber and Stephen Lee, Analog Devices, Wilmington, MA

DESIGNERS who build equipment for the industrial market share a widespread problem. At one extreme, they must build equipment that supports ±10V bipolar voltages, often riding on a high common-mode level, a requirement enforced by 30 years of legacy industrial equipment. At the other extreme, the analog signal needs conditioning to match the full-scale range of a low-voltage, single-supply ADC. Designers need to scale and level-shift signal levels throughout their system to accommodate the high voltage levels that sensor manufacturers dictate and the low voltage levels that the ADC dictates. Operating from a single 5V supply, the circuit in this Design Idea provides an interface of large bipolar inputs to a single-supply, low-voltage, differential-input ADC. The circuit in **Figure 1** comprises two difference amplifiers, connected in antiphase. The differential output, V_1-V_2, is an attenuated version of the input signal: $V_1-V_2=(V_A-V_B)/5$.

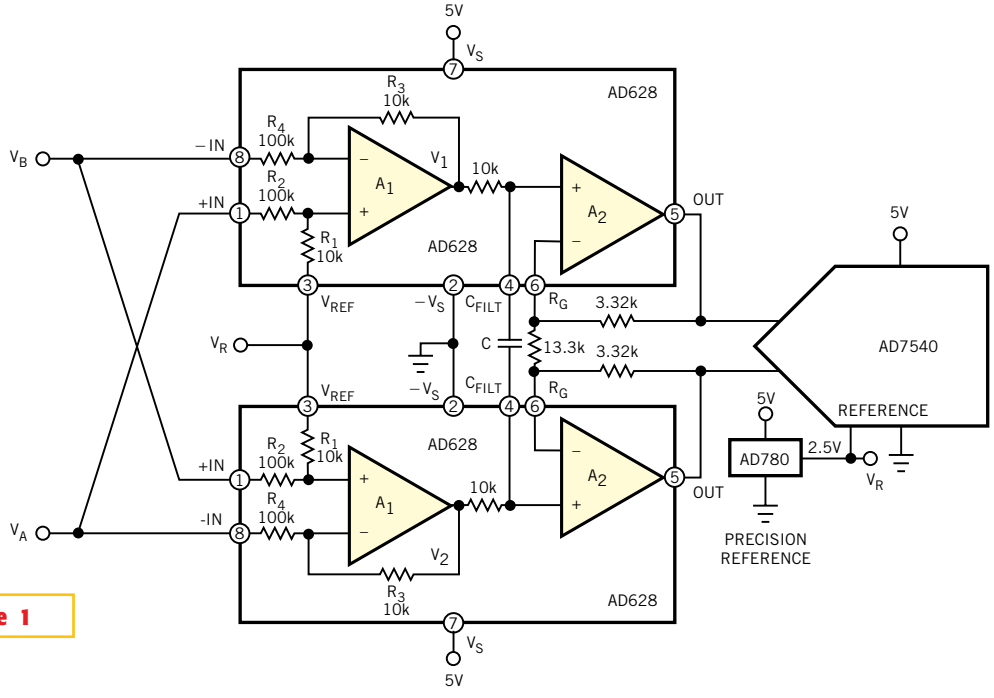


Figure 1 This circuit attenuates and level-shifts a ±10V differential signal while operating from a single 5V supply.

The difference amplifiers reject the common-mode voltage on inputs V_A and V_B. The reference voltage, V_R, which the AD780 develops and the ADC and the amplifier share, sets the output common-mode voltage. A single capacitor, C, placed arcros the C_{FILT} pins, lowpass-filters the difference signal, V_1-V_2. The −3-dB pole frequency is: $f_P=1/(40{,}000\times\pi\times C)$. A_2 amplifies the difference signal by 1.5. Thus, the total gain of this circuit is $^3/_{10}$. **Figure 2** shows a 10V input signal (top), the signals at the output of each AD628 (middle), and the differential output (bottom). The benefits of this configuration go beyond simply interfacing with the ADC. The circuit improves specifications such as common-mode-rejection ratio, offset voltage, drift, and noise by a factor of $\sqrt{2}$ because the errors of each AD628 are not correlated.

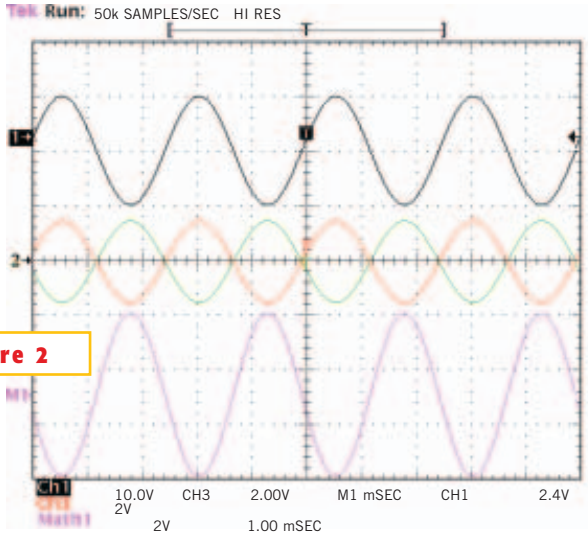


Figure 2 The waveforms show a 10V input signal (top), the signals at the output of each AD628 (middle), and the differential output (bottom).

The output demonstrates 85-dB SNR (**Figure 3**). The two AD628s interface with an AD7450 12-bit, differential-input ADC. The AD7450 easily rejects residual common-mode signals at the output of the difference amplifiers. **Figure 4** shows the common-mode error at the output of the AD628.

The topmost waveform is a 10V, common-mode input signal. The middle waveform, measuring 150 µV, is the common-mode error measured, differentially, from the output of the two AD628s. The bottom waveform, measuring 80 µV, is the resultant common-mode error.□

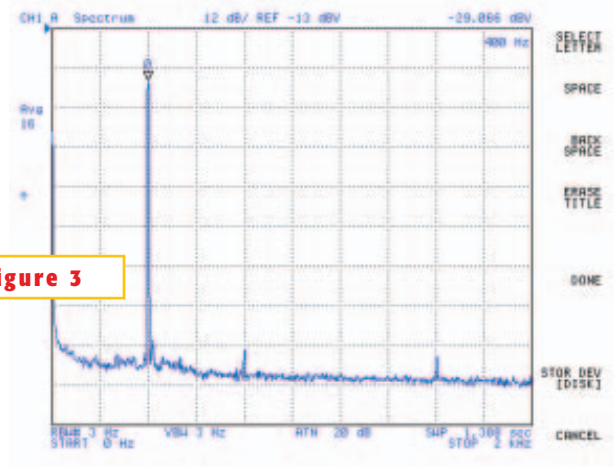


Figure 3

The circuit in Figure 1 has an 85-dBV SNR.

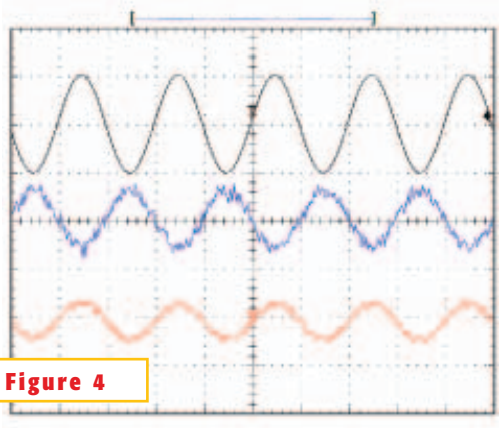


Figure 4

The common mode input (top) measures 20V p-p. The common-mode error of the differential output (middle) is 200 µV p-p. The error of the common-mode output (bottom) is 80 µV p-p.

Two op amps provide averaged absolute value

Dobromir Dobrev, Jet Electronics, Sofia, Bulgaria

THE CIRCUIT in **Figure 1** is useful when you need amplitude demodulation or an averaged absolute-value conversion. The circuit comprises two stages, the first of which, IC_{1A}, is a differential-output absolute-value converter. The second stage, IC_{1B}, is a traditional differential amplifier. The combination of the two stages performs single-ended absolute-value conversion but only if $R_3 >> R_2$. The C_1 capacitors integrate the current flow and yield averaged voltages V_A and V_B. In addition, the capacitors ensure low ac-impedance points at nodes V_A and V_B when the output diodes are reverse-biased. The additional C_2 capacitors in parallel with R_4 resistors impart a second-order-lowpass-filter characteristic to the circuit and remove the remaining ac signal. From a practical point of view, you can choose R_3 to be five to 10 times higher than R_2. The gain of the circuit is $(R_2||R_3/R_1)(R_4/R_3)$. In most applications, you would choose the filter time constants $\tau_1 = R_2||R_3C_1$ and $\tau_2 = R_4C_2$ to be equal. The circuit in **Figure 1** is simple, symmetrical, and cost-effective. It also makes it easy to calculate and adjust the gain using one resistor, R_1. Other advantages are that the circuit has equal delay for positive- and negative-going signals and that it doesn't need matched diodes.□

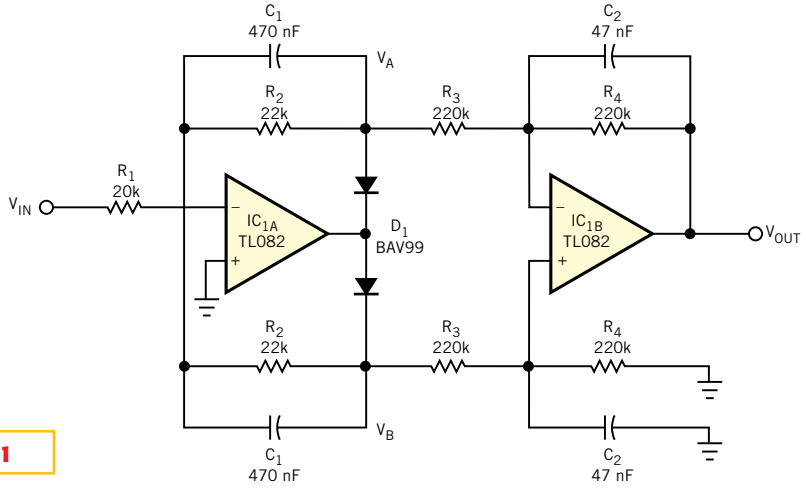


Figure 1

This single-ended, averaged absolute-value converter is useful for amplitude demodulation.