In most keypads, pressing a key closes a contact that bridges two lines in an xy matrix. If you use a microcontroller to detect a key closure, checking the states of (x+y) lines requires an equal number of I/O pins. Occupying only one free I/O pin, the circuit Figure 1 communicates with a microcontroller by generating a single pulse each time someone presses a key. The pulse’s width is proportional to the number of the pressed key, and the microcontroller identifies the pressed key by measuring the pulse’s width.

IC2, a CMOS LMC555 version of the popular 555 timer, operates as a monostable one-shot multivibrator. In the circuit’s resting state, a transistor internal to IC2 at Pin 7 shunts C6, and IC2’s output at Pin 3 remains at logic low. Pressing any key on the keypad connects two resistors from two groups—R1 and R2 in one
group and $R_3$, $R_4$, and $R_5$ in the other—in series with $R_x$. The sum of the two resistors varies in 10-kΩ increments, and the total resistance is proportional to the number of the pressed key.

Pressing any key draws current through $R_y$, and the selected keypad resistors and raises the voltage at IC$_1$’s Pin 7. After $C_1$ charges, introducing a short delay that’s sufficient to eliminate keypad-switch contact-closure bounce, CMOS comparator IC$_1$ detects the small voltage drop established across $R_x$. The output of IC$_1$ (Pin 6) goes from 5 to 0V, which in turn triggers Pin 2 of IC$_2$. Timer IC$_2$’s output (Pin 3) goes high and begins to charge capacitor $C_6$ at a time constant that depends on the selected key. When the voltage across $C_6$ reaches two-thirds of $V_{CC}$, or 3.333V, Pin 3 goes low and discharges $C_6$. The following equation calculates IC$_2$’s output pulse width, $T$: $T = 1.1 \times R_S \times C_6$, where $R_S$ equals the sum of the selected keypad resistors and ranges from 10 to 120 kΩ. The pulse width spans a range of 110 to 1320 μsec in increments of 110 μsec.

The smallest relative change in pulse width occurs at the longest pulse ratio, 110/1320, or 8.33%. This ratio provides sufficient margin to allow use of standard ±1% tolerance or better components for those in Figure 1 that are ±0.5 and ±1%. Resistors $R_{13}$ and $R_{14}$ compensate for variations in IC$_2$’s internal voltage dividers by forcing the voltage at Pin 5 to two-thirds of power-supply voltage $V_{CC}$.

The keypad circuit’s output pulse drives the external interrupt input, RA$_2$, of a Microchip 16F630 microcontroller. Listing 1, available at the online version of this Design Idea at www.edn.com, presents an interrupt routine for the 16F630 that measures the pulse width, verifies that its tolerance is within ±40 μsec, and returns a numerical value of 1 to 12 that corresponds to the pressed key. As a safeguard against erroneous data, the routine returns an error code if the pulse width falls outside certain limits.

### Calculator program evaluates elliptic filters

Fernando Salazar-Martínez, Alan Altamirano-Cruz, and David Báez-López, Department of Engineering Electronics, University of the Americas, Puebla, Mexico

Many designers consider the elliptic-transfer function to be the most useful of all analog-filtering functions, because of its steep roll-off at the band edges. You can use a Texas Instruments model V200 Voyage programmable calculator and the program in Listing 1 at the Web version of this Design Idea at www.edn.com to evaluate a lowpass elliptic filter by finding its characteristic’s poles and zeros. To do so, this program implements Darlington’s algorithm (Reference 1). The program accepts as input the filter’s maximum passband-attenuation ripple in decibels, its stopband and passband frequencies in radians per second, and its order, or number of poles (Figure 1).

As an example, calculate the zeros, poles, and stopband attenuation of an elliptic, fifth-order, analog lowpass filter with maximum gain of 0.1 dB and stopband frequency of 1.05 radians/sec. Figure 2 illustrates the calculator’s display screens during program execution.

**Figure 1** The characteristics of an elliptic filter’s amplitude response include in-band ripple, passband-attenuation and stopband frequencies, and stopband attenuation.

**Reference**


**Figure 2**

These screens show the calculator’s display from the introductory menu (a), entering filter parameters (b), calculating values for filter-response zeros (c), calculating value for out-of-band attenuation (d), and calculating values for filter-response poles (e).
Dynamic-load circuit determines a battery’s internal resistance

Jim Williams, Linear Technology Corp, Milpitas, CA

The simplest model of a battery comprises an ideal voltage source that connects in series with a resistance whose value—often a few milli-ohms—depends on the battery’s electrochemical condition and construction. If you attempt to use an ordinary ac milliohmmeter containing a kilohertz-range ac excitation source to measure a battery’s internal resistance, you get erroneous results due to capacitive effects, which introduce losses. A more realistic battery model includes a resistive divider that a capacitor partially shunts (Figure 1). In addition, a battery’s no-load internal resistances may differ significantly from their values under a full load. Thus, for greatest accuracy, you must measure internal resistance under full load at or near dc.

The circuit in Figure 2 meets these requirements and accurately measures internal resistance over a range of 0.001 to 1Ω at battery voltages as high as 13V. One section of an LTC6943 analog switch, IC2a, alternately applies 0.110 and 0.010V derived from 2.5V voltage reference IC3 and resistive divider R2, R3, and R4 to IC1’s input.

Amplifier IC1, power MOSFET Q1, and associated components form a closed-loop current sink that provides an active load for the battery under test via Q1’s drain. Diode D1 provides reversed-battery voltage protection; a 1.5kΩ Sense resistor monitors the battery’s voltage. Amplifiers IC4A and IC4B form two stages of current amplification, and LTC1150 output amplifiers provide the required gain.

These components form a switched-current sink D1, which provides the necessary active load for the battery under test. Diode D1 provides reversed-battery voltage protection. A 1.5kΩ Sense resistor monitors the battery’s voltage. Amplifiers IC4A and IC4B form two stages of current amplification, and LTC1150 output amplifiers provide the required gain.

For improved accuracy, analyze the battery’s voltage drop at a frequency near dc.
Battery automatic power-off has simpler design

Yongping Xia, Navcom Technology, Torrance, CA

A previous Design Idea describes a simple way to automatically turn off a battery after a preset on period to save battery life. This Design Idea presents a simpler way to perform the same function (Figure 1). Two gates of IC1, a quad two-input NAND Schmitt trigger, form a modified flip-flop. When you apply a 9V battery to the circuitry, the output of IC1A goes high because the initial voltage on C1 is zero. The output of IC1B is low, which feeds back to IC1B through R2. C3 charges up through R3. The output of IC1C goes high because R6 is connected to ground. A P-channel MOSFET switch, Q1, is off, and the output of IC1D goes high, which in turn charges C4 through R5.

When you push momentary switch S1, IC1A’s output goes low because both of its inputs are high, and this output forces IC1B’s output high. The value of R2 is much smaller than R3, so that C3 holds a logic-level high when S1 stays on. When S1 goes off, C3 discharges through R3. You can turn off the MOSFET switch in one of two ways. When tantalum capacitor C2 is charged such that the voltage on IC1C’s input becomes lower than its threshold $V_{IH}$, IC1C’s output changes from low to high; this action turns off the MOSFET switch. C2 and R6 determine the duration of this automatic protection. The voltage at amplifier IC5’s positive input and the voltage drop across R8 determine the load applied to the battery. In operation, the circuit applies a constant-current load comprising a 1A, 0.5-Hz square wave biased at 100 mA to the battery.

The battery’s internal resistance develops a 0.5-Hz amplitude-modulated square-wave signal at the Kelvin connections attached to the battery. A synchronous demodulator comprising analog switches S5 and S3 in IC1B and chopper-stabilized amplifier IC5 processes the sensed signal and delivers a 0 to 1V analog output that corresponds to a battery-resistance range of 0 to 1Ω.

Via transistor Q3, amplifier IC5’s internal approximately 1-kHz clock drives CMOS binary divider CD4040, IC6, which supplies a 0.5-Hz square-wave clock drive for the switches in IC7. In addition, a 500-Hz output from IC5 powers a charge-pump circuit that delivers approximately ~7V to IC8’s negative power-supply input and thus enables IC8’s output to swing to 0V.

The complete circuit consumes approximately 230 μA, allowing nearly 3000 hours of operation from a 9V alkaline-battery power supply. The circuit operates at a supply voltage as low as 4V with less than 1-mV output variation and provides an output accuracy of 3%. The circuit accommodates a battery-under-test voltage range of 0.9 to 13V, but you can easily alter discharge current and repetition rate to observe battery resistance under a variety of conditions.

![Figure 1](image-url)
ic turn-off. With the values shown, the turn-off takes approximately six minutes. Meanwhile, the high-to-low transition on IC1b’s output forces IC1a and IC1b back to standby status through C4.

Alternatively, you can manually turn off the MOSFET switch by pushing S1. Because the voltage on C3 is low, closing S1 forces IC1a’s outputs high and IC1b’s outputs low. The high-to-low transition on IC1b’s output forces IC1c’s output to be high, which turns off the MOSFET. Because the value of C5 is fairly large, D1 provides a quick discharge route, and R4 limits the discharge current.

This circuitry consumes less than 0.2 \(\mu\)A of power during standby operation. Because the MOSFET switch has a low on-resistance, it has only a 2-mV loss when the load current is 100 mA. Add an LED with a current-limiting resistor in series to the load side if you need a power-on indicator.

**Reference**

---

**Figure 1**
Under control of its host processor, digital potentiometer IC, adjusts the processor’s core power-supply voltage.
voltage from exceeding 1.2V if a software glitch occurs.

A digital potentiometer typically presents a highly variable absolute resistance value but can accurately set its internal resistance ratio. In this design, the AD5258’s internal resistor forms a voltage divider with an external resistor to set the output voltage. To improve the ADP3051’s output-voltage accuracy, the ADSP-BF531 uses a simple algorithm to compute and store an appropriate maximum resistance for a given operating voltage in the AD5258’s nonvolatile memory via its I2C port.

Using the AD5258 with an external resistor provides hardware protection to prevent the output voltage from going above 1.2V. If the AD5258 is set to zero resistance, the resulting output voltage is $0.8V \times (0\Omega + 10 \text{k}\Omega)/10 \text{k}\Omega = 0.8V$. If you set it to its maximum resistance of 5 k\Omega, the resulting output voltage is $0.8V \times (5 \text{k}\Omega + 10 \text{k}\Omega)/10 \text{k}\Omega = 1.2V$. When the embedded processor directs the AD5258 via its I2C port to ramp the core voltage from 0.8 to 1.2V, IC1’s output voltage monotonically increases within 40 µsec (Figure 2).

![Figure 2](image.png)

**Figure 2**

Applying power-supply voltage to the host processor ramps voltage from 0.8 to 1.2V with only 60-mV overshoot.