BY KRIPASAGAR VENKAT • TEXAS INSTRUMENTS

Glass-breakage detector uses one microcontroller

A GLASS-BREAKAGE DETECTOR CAN DETECT WHEN A WINDOW OR DOOR BREAKS IN A HOME OR BUSINESS, SERVING AS A MONITORING DEVICE TO ENHANCE SECURITY BY DETECTING ILLEGAL ENTRY.

A glass-breakage detector works either independently or in conjunction with other antitheft devices to form a security system. The detector essentially captures and analyzes any acoustic activity and reports whether glass breakage has occurred. Due to their mode of operation, these detectors depend heavily on the quality of sound events, posing numerous challenges to the designer. The detector must also be able to reject all failure alerts—sounds that are not true glass breakage. This article discusses an efficient and robust glass-breakage detector using a low-cost microcontroller.

Microcontrollers are low-end processors that find use in applications such as simple digital real-time clocks and complex smart-metering systems. Microcontrollers suit these applications because they cost less, consume less power, and are easier to use than most other types of digital processors. In simple applications with limited requirements, it is easy to achieve low cost and low power. However, with the trend toward using microcontrollers in complex applications, it becomes a challenge to maintain low cost and achieve low power. Engineers must now try to get the best performance with the lowest possible cost. To achieve this goal, they face microcontroller-architecture restrictions, such as lower on-chip memory, a limited peripheral set, lower operational speed, and a smaller pin count. Engineers must optimize everything these microcontrollers offer for use in fairly complex applications, such as the glass-breakage detector.

DESIGN CONSIDERATIONS

A robust glass-breakage-detection algorithm should be able to easily distinguish valid glass breakage from other sound events. All glass-breakage-detection algorithms capture sound events, analyze their time and frequency components, and make a decision. Glass-breakage sounds vary by type of glass, thickness, acoustic environment, distance, the object the would-be thief uses to break the glass, and other factors. All glass-breakage-detection algorithms are inherently similar but vary slightly depending on conditions, so one algorithm will not work for all conditions. Installers usually fine-tune the algorithm during final installation in a home or business.

You can analyze a valid glass-breakage signal in the time domain or the frequency domain. **Figures 1** and **2** show a typical glass-breakage signal in the time and the frequency domains, respectively. This sound falls well within the audio spectrum of 20 Hz to 20 kHz. The time-domain waveform relates to the sound that a listener hears, and the frequency-domain waveform gives the complete frequency content of the signal. These plots provide valuable information in the design of an efficient algorithm for breakage detection. The time-domain plot indicates that the waveform is dense and that a lot of activity occurs in short intervals. This activity relates to the fact that the signal contains a lot of high-frequency components and that the waveform has a lot of ze-

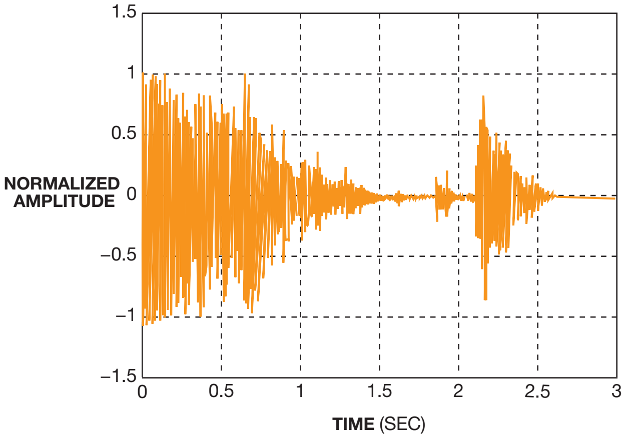

Figure 1 A typical glass-breakage signal in the time domain falls well within the audio spectrum of 20 Hz to 20 kHz and relates to the sound that a listener hears.

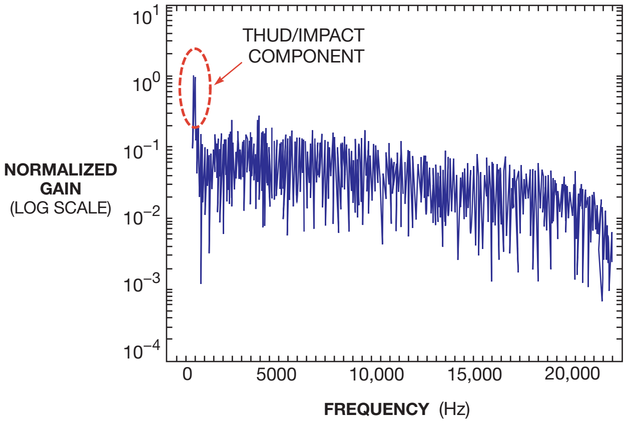

Figure 2 The frequency-domain waveform gives the complete frequency content of the signal.

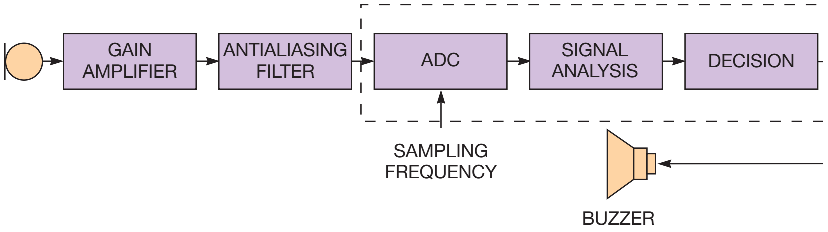

Figure 3 A microphone captures sound events, and an antialiasing filter following a gain amplifier handles signal amplification and filtering of high-frequency components.

ro crossings and high peaks. These characteristics, although they provide good information, seem to mimic white noise, and it is a challenge for designers to distinguish between the waveform's characteristics and white noise.

The frequency response involves similar challenges. The glass-breakage signal's components spread over the entire spectrum with fairly equal energy, which is typical of white noise. However, a peak occurs at approximately 200 to 300 Hz; this peak—the frequency component of the sound caused by the initial impact to the glass during breakage—provides the much-needed distinction. The impact is a low-frequency signal among all the high-frequency glass-breakage sounds that follow. You can view this impact, or thud, as the sound that occurs when an object hits the glass. It is difficult to recognize this information in the time-domain waveform, but you do know that this sound precedes all other sounds during breakage.

SYSTEM COMPONENTS

A glass-breakage detector must always be on and should be able to process any sound activity in real time. However, you can turn off some of the detector's blocks or put them into low-power modes when they are not operating. A microphone captures sound events, and an antialiasing filter following a gain amplifier handles signal amplification and filtering of high-frequency components (**Figure 3**). The antialiasing filter rejects any frequencies above the audible range of 20 kHz and avoids violating the Nyquist criterion during digitization of an analog signal.

The blocks within the dashed lines are parts of the processor, which can be an ASIC, a microcontroller, or a DSP. The ADC converts the analog signal to a digital sample for processing in the digital domain. The sampling frequency, F_S, depends on the frequency content of the signal. Because this circuit uses a 20-kHz antialiasing filter, the sampling rate must be at least 40 kHz to preserve the original signal's content and integrity. The signal-analysis block encompasses all of the signal processing necessary for the detection or rejection of glass breakage. Once this detection completes, the decision block activates an indicator, such as an LED or a buzzer, to indicate glass breakage.

HARDWARE SPECS

Most glass-breakage detectors operate from batteries; for sufficient battery life, the design should have low power consumption. The choice of all hardware components in this design depends on their ability to contribute to the low-power design. An analog signal starts at the microphone and ends at the ADC. The choice of the microphone is important because its performance contributes to the success of any glass-breakage detector. The microphone should also be able to capture and preserve key sound components, such as impact and other high-frequency components that the detector algorithm uses. The microphone must be on most of the time to capture any sound activity and hence must consume less power to bring down the overall system current. The gain amplifier—usually an operational amplifier in inverting or noninverting mode—has a gain higher than unity. The op amp must provide sufficient gain to the sound—on the order of tens of millivolts—the microphone captures. The op amp must always remain on and must have a small turn-on current. The antialiasing filter is also an op amp, filtering in the analog domain, and is usually a simple first- or second-order-unity-gain lowpass filter.

The most important choice in this design is that of the signal processor. You can use ASICs, microcontrollers, or DSPs, depending on the application. Like smoke detectors, most glass-breakage detectors are placed inside homes or offices at locations that ensure security and safety. However, they require battery power so that you can place them anywhere without worrying about their proximity to power outlets and to ensure that they will continue to operate in the absence of power on the mains.

You must choose a low-power, programmable, easy-to-use,

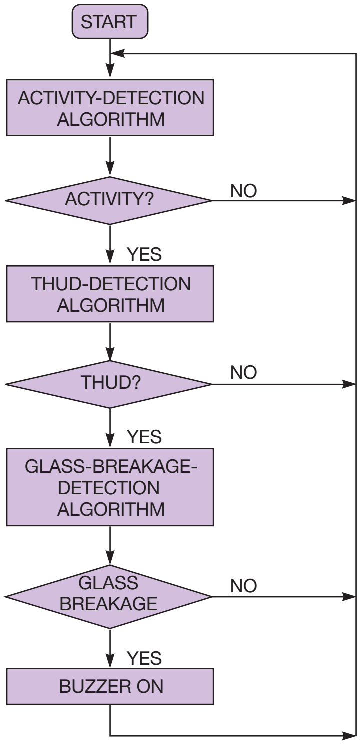

Figure 4 The high-level software flow of the algorithm includes activity detection, thud detection, and glass breakage in the order of occurrence in time.

TABLE 1 CURRENT AND TIMING CONSIDERATIONS

Condition/mode	Peripherals on	Clocks	Current	On time
Low-Power Mode 3	Timer on (up mode)	MCLK=DCO=off SMCLK=DCO=off ACLK=VLO≈12 kHz TACLK=ACLK=VLO	0.6 µA	2.5 msec
Activity detection/ AM1 (Active Mode 1)	Microphone on Op amp 0 on	MCLK=DCO=12 MHz SMCLK=DCO=12 MHz ACLK=VLO≈12 kHz	4.8 mA	20 µsec
Thud detection/ AM2 (Active Mode 2)	Microphone on Op amp 0 on Op amp 1 on Timer on (up mode) ADC 10 on	MCLK=DCO=8 MHz SMCLK=DCO=8 MHz ACLK=VLO≈12 kHz TBCLK=SMCLK=8 MHz	4 mA	32 msec
Glass-breakage detection/ AM3 (Active Mode 3)	Microphone on Op amp 0 on ADC 10 on	MCLK=DCO=12 MHz SMCLK=DCO=12 MHz ACLK=VLO≈12 kHz	5.8 mA	60 msec

and inexpensive processor with good processing capabilities for real-time operation. The microcontroller is the best of these choices because it meets all of these requirements. Some microcontrollers also integrate analog peripherals, which further reduce overall system cost.

SOFTWARE SPECS

An antialiasing filter with a 20-kHz cutoff frequency filters the analog signal from the microphone. To digitize this signal, the sampling rate must be greater than 40 kHz, and the ADC must be able to support that rate. For real-time operation, the filter must complete the required processing between successive-sampling instants. For example, if the maximum CPU frequency is 12 MHz, the number of CPU cycles between successive samples is only 300, which is tight for signal processing. You can choose a processor that supports a higher CPU clock for increased CPU cycles. Doing so would increase power consumption, however, and therefore decrease battery life. Hence, one must make a trade-off between the algorithm's complexity level and battery life.

The thud occurs at the beginning of a glass-breakage sound. This thud signal is present in most sounds, such as the noise that occurs when a door or cabinet closes, an object hits the ground, or someone claps his or her hands or knocks on a door. However, these sounds lack the high-frequency components of typical glass-breakage signals. Other sounds, such as the noise a coffee grinder makes, loud music, motorcycles in motion, or the noise a wineglass makes when it breaks, have similar high-frequency components but no thud component.

The glass-breaking algorithm exploits the fact that these two types of components are on either side of the frequency spectrum and occur independently of each other in time.

GLASS-BREAKAGE ALGORITHM

Figure 4 shows the high-level software-flow diagram of the algorithm, including activity detection, thud detection, and glass breakage in the order of occurrence in time. Approximately every 2.5 msec, the microphone and op amp 1 turn on to check for any sound activity. In the absence of any significant activity, they turn off, and the microcontroller goes into a low-power state. If significant activity occurs, the software proceeds to thud detection, during which the ADC turns on, followed by signal processing to check for the thud component. If a thud is present, the algorithm proceeds to glass-breakage detection. Otherwise, the algorithm reverts back to activity detection. If glass-breakage detection is successful, an onboard LED or buzzer activates to indicate this event. The glass-breakage detector then reverts back to activity detection.

Activity detection compares the ADC's input values to prefixed thresholds on either side of zero to distinguish

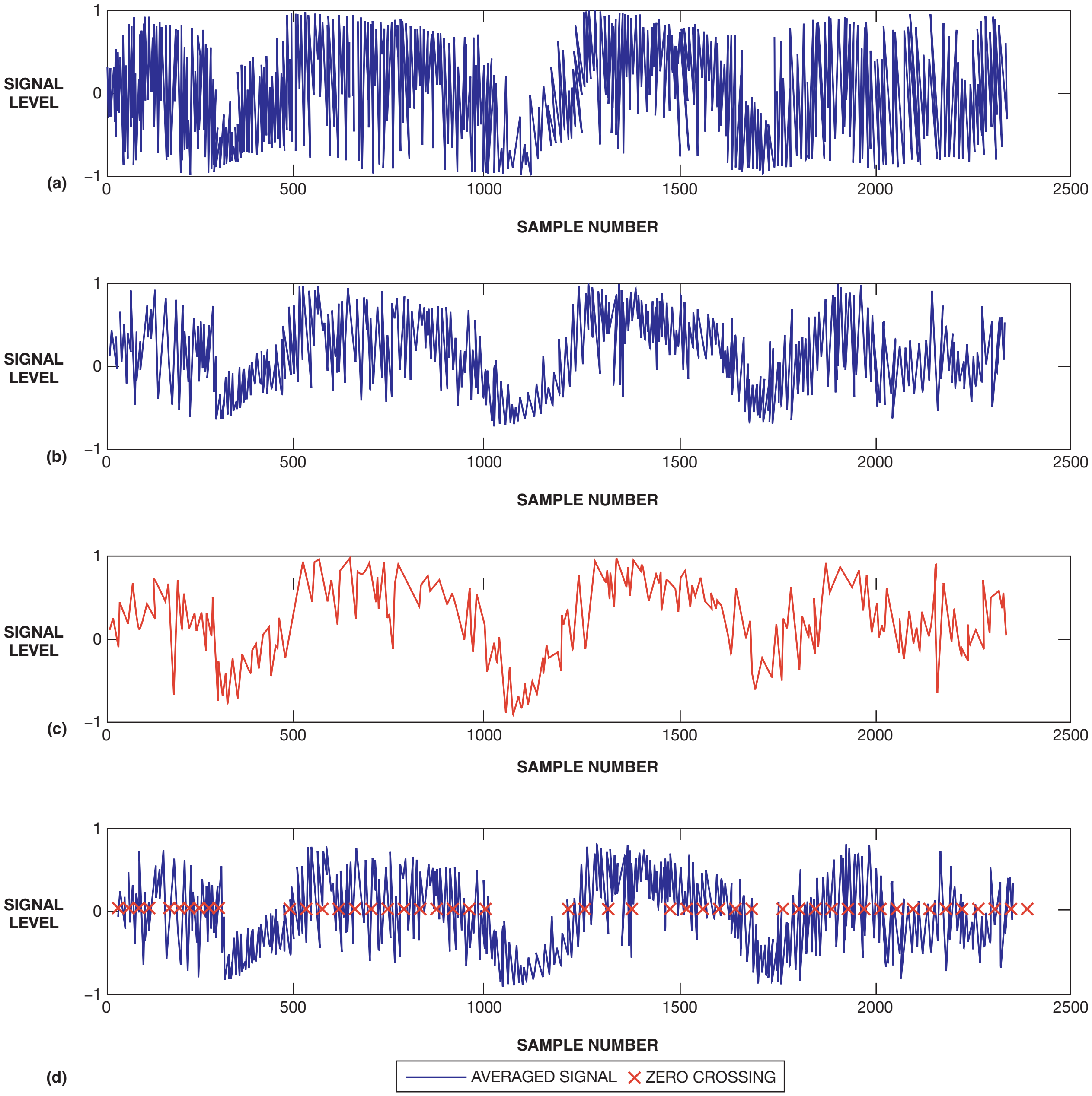

Figure 5 The first stage of processing occurs on every sample once the algorithm detects a thud. This stage uses a 20-kHz antialiasing filter and increases the ADC's sampling frequency to 40 kHz.

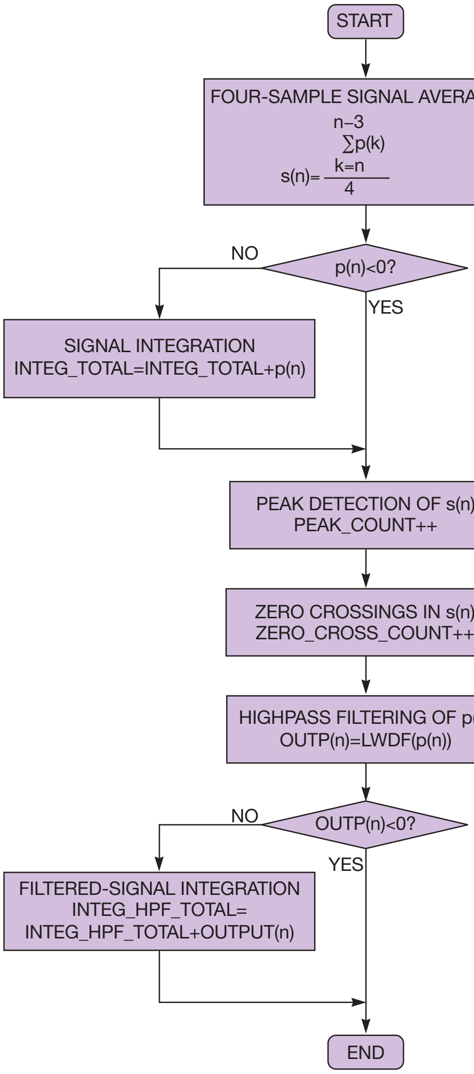

Figure 6 The operations for this stage include signal averaging, zero-crossing detection, and peak detection, which occur for approximately 60 msec, or approximately 2400 samples.

a true signal from noise. The thud component occurs only during initial impact, and a digital lowpass filter with a cutoff frequency of 350 Hz filters only the first few samples of the incoming signal. The system accumulates and averages the filtered samples and compares them to a prefixed energy threshold. If the energy exceeds this threshold, the system initiates a thud component and the glass-breakage-detection algorithm. The digital lowpass filter must be small yet effective, so the sampling frequency for these initial samples remains at only 4 kHz. However, this section of the algorithm uses an antialiasing filter with a cutoff frequency of 2 kHz rather than 20 kHz for typical antialiasing filters.

The glass-breakage-detection algorithm is more complex than the thud detection and includes two signal-analysis parts. One is the first stage of processing and occurs on every sample once the algorithm detects a thud. This stage uses a 20-kHz antialiasing filter and increases the ADC's sampling frequency to 40 kHz. The operations for this stage include signal averaging, zero-crossing detection, and peak detection, which occur for approximately 60 msec, or approximately 2400 samples. Once the first stage is complete, the second stage initiates to complete the entire signal analysis.

Figure 5 shows a signal representa-

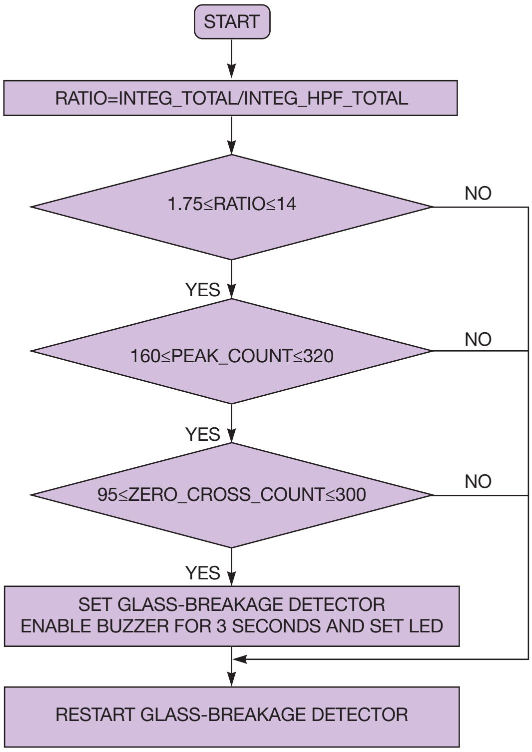

Figure 7 Once the first stage is complete, the second stage initiates to complete the entire signal analysis.

tion during the first stage, and **Figure 6** shows the software flow. The algorithm's p(n) signal denotes the incoming samples, which pass through a simple moving-average filter to reduce noise, yielding the s(n) signal. Integration of the p(n) signal uses only positive samples to calculate signal energy, integ_total, for use in the second processing stage. The s(n) signal then receives peak and zero-crossing counts. To extract the high-frequency components of the incoming signal, you use a highpass filter with a cutoff frequency of one-fourth the sampling frequency and subject each sample of p(n) to this filtering. Simultaneously, only the positive samples of the filtered output are accumulate in the result, integ_HPF_total, which the second stage uses. Stage 1 filtering occurs on every sample and must be complete before the arrival of the next sample, p(n+1), for real-time operation, implying that the total amount of CPU cycles available is only the CPU frequency divided by 40 kHz. Filtering is generally a time-consuming operation. To achieve efficiency, both the lowpass filter and the highpass filter in Stage 1 use lattice-wave digital filters and Horner's algorithm for thud detection.

THE ALGORITHM COMPUTES THE RATIO OF TOTAL SIGNAL ENERGY TO HIGHPASS-FILTERED ENERGY AND CHECKS THE RESULTS AGAINST A THRESHOLD.

Once the first stage of signal analysis processes 60 msec of data, the algorithm proceeds to the second stage of processing. The second stage does not require real-time operation (**Figure 7**). The end of the second stage of signal analysis confirms whether any glass breakage has actually occurred.

The algorithm computes the ratio of total signal energy to highpass-filtered signal energy and checks the results against a threshold. Results show a ratio of 1.75-to-14 for a number of glass-breakage sounds. Similarly, the algorithm checks the number of peaks if it is between 160 and 320 and whether the number of zero crossings is between 95 and 300. A valid glass breakage occurs if the results satisfy each of these conditions. If even one of the conditions fails, the glass-breakage detector reinitializes and returns to activity detection. You

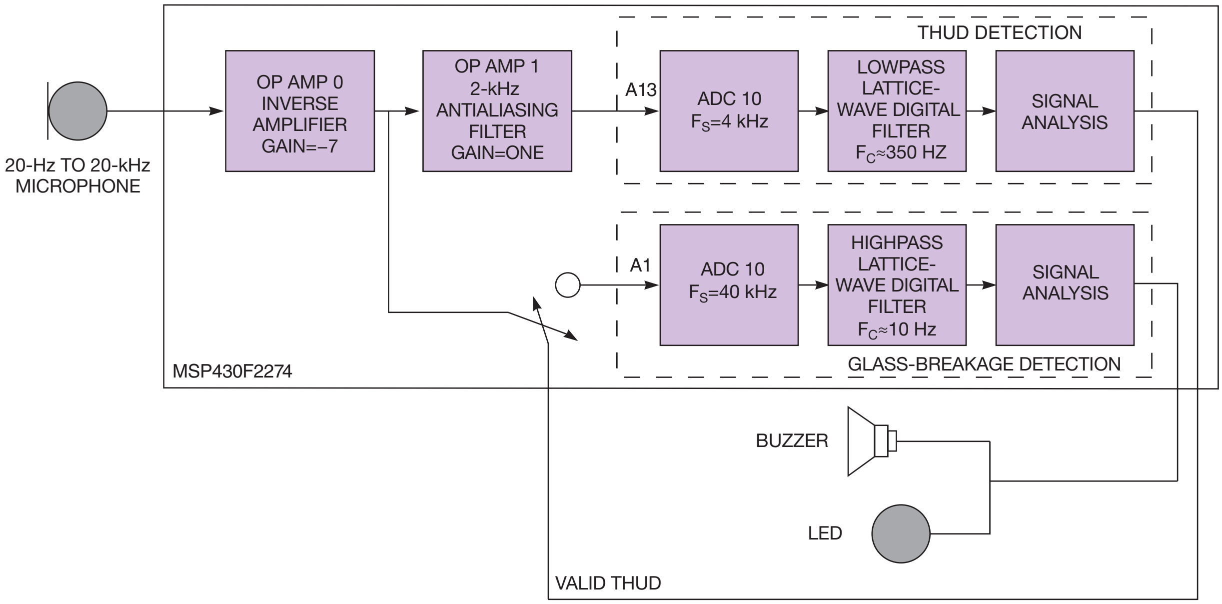

Figure 8 Texas Instruments' low-power, 16-bit MSP430F2274 microcontroller operates at frequencies as high as 16 MHz.

must slightly tweak these thresholds and ranges depending on the room's acoustics, the detector's location, noise in the environment, and other factors.

MICROCONTROLLER IMPLEMENTATION

The ultralow-power MSP430-microcontroller platform from Texas Instruments (www.ti.com) comprises a variety of devices, including the 16-bit MSP430F2274 microcontroller, which operates at frequencies as high as 16 MHz (**Figure 8**). It also has an internal low-power, low-frequency oscillator, which operates at 12 kHz at room temperature; two 16-bit timers; and a 10-bit ADC 10, which supports conversion rates as high as 200 kHz. The ADC works with on-chip, software-configurable operational amplifiers 0 and 1 for analog-signal conditioning. The device consumes 0.7 μA of current during standby-mode and 250 μA during active mode, making it a good choice for battery-powered applications. Because the chosen microphone has a passband of 20 Hz to 20 kHz and the MSP430F2274 integrates only two op amps, you could remove the 20-kHz antialiasing filter from the implementation. Although this removal violates sampling theory, the results do not vary with this absence. However, if another op amp is available, the filter can still be part of this setup.

Op amp 0 works as an inverting amplifier with a gain of seven to provide amplification to the microphone output. Op amp 1 works as a unity-gain lowpass filter, which is a second-order Butterworth type using the Sallen-Key architecture. The filter has a 3-dB cutoff frequency at 2 kHz. The outputs of the two op amps internally connect to channels A_1 and A_{13}, respectively, of the MSP430.

CURRENT CONSUMPTION

The glass-breakage detector's current consumption depends on the low-power modes it uses during its operation and selective turn-on and turn-off of its peripherals. The current-consumption profile of the implementation on the MSP430 for the three modes of operation is discussed in the online version of this article at www.edn.com/ms4375. **Table 1** provides a list of the peripherals and clocks that are on during various modes of operation. These peripherals are significant contributors to the overall current consumption during each stage.

Two AAA batteries providing 800 mAhr of energy power the glass-breakage-detection board. Although it is difficult to predict the battery life of such an application, assuming no glass breakage, the total current consumption is approximately 80 μA to give a battery life of about 416 days. You can further increase battery life by increasing the wake-up for activity detection to more than 2.5 msec; however, this approach increases the possibility of missing a sound event.**EDN**

AUTHOR'S BIOGRAPHY

Kripasagar Venkat is a system-applications engineer for Texas Instruments (Dallas). He holds a master's degree from the University of Texas—Dallas with an emphasis in signal processing and communication.