



GRAND PRIZE WINNER

Stealth Telephone Screener

Tired of receiving bothersome automated phone calls from telemarketing companies armed with predictive dialing systems, Richard decided to fight technology with technology. The result? A PIC-based call screener that's been consistently successful at blocking unwanted automated calls.

Almost everyone has been bothered, at some point in time, by annoying phone calls from telemarketers. After the marketers had convinced themselves that their time is more valuable than yours, they began using predictive dialing systems, which were implemented to place numerous calls at one time and connect only the first caller to a live telemarketer. The other calls were dropped, leaving many people to find only dead air on the line and wonder why so many people had been hanging up on them. I've received my share of these calls, so I recently decided to do something about them.

There are many screening products on the market today. Most of them

require either the caller to know a special code to dial before connection or the callee to manually screen the call and press a button to ignore it. Other systems play a sequence of three special information tones, which are used to indicate a disconnected number, in an attempt to fool a dialing machine into hanging up. I wanted something that would neither require manual input nor offend people who don't appreciate being screened—especially those who would rather not leave a message on an answering machine if they don't know a special code.

The Stealth Screener was born out of these requirements (see Photo 1). It answers the phone with a natural-

sounding "Hello," and then waits for a response. A predictive dialer will remain silent, causing the screener to hang up after a short delay. If the caller plays an automated recorded message, the screener will hang up if there is continuous, unbroken speech for a long period of time. Otherwise, if it hears an appropriate response, it will play a second greeting message (e.g., "Just a minute"), and then initiate ringing tones while ringing through its internal speaker. Any other



Photo 1—The prototype was wire-wrapped and installed in a 5" x 7" plastic box. The high-voltage components are along the right side next to the battery compartment.

phones should have their ringers turned off, so only the screener's ring is heard.

A block diagram of the system is shown in Figure 1. The hardware is based on a Microchip PIC16F872, which contains a 10-bit A/D converter and PWM D/A converter. There are a 128-Kb I²C EEPROM and a pair of MCP6022 dual op-amps. A 9-V battery powers the entire system.

HEARING VOICES

The Stealth Screener's most important characteristic is the voice detection algorithm shown in Figure 2. I used a commonly known method of tracking the audio's speech and noise levels and computing a threshold based on the values.

The levels are derived from the root mean square (RMS) energy of the input signal. As each sample is acquired, its square is computed and

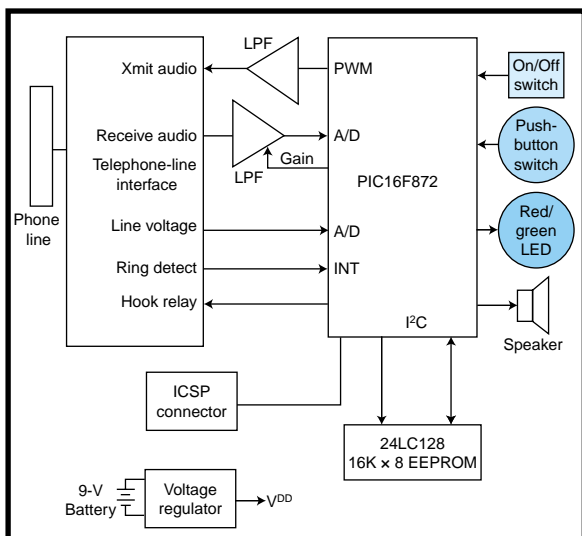


Figure 1—The architecture is straightforward. The phone line isn't isolated, so be sure to unplug it before connecting the ICSP cable.

summed. Then, at the end of each block of 200 samples, the square root of the sum is computed. I used a reverse look-up table for the square root because computing it algorithmically would have been too slow.

The RMS energy is then run through a pair of averaging filters. The speech filter has a fast attack time and a slow decay to track the peaks of the audio envelope. To track the valleys, the noise filter has a slow attack and fast decay.

The VOX turn-on threshold is set to the larger of one-sixteenth of the speech level and two times the noise level, which is subject to minimum

and maximum limits. After it's turned on, the turn-off threshold is one-eighth of the overall highest speech level. There are various ways to fine-tune the thresholds beyond the current algorithm, but they aren't necessary.

TALKING BACK

The Screener records and plays two short announcements that are stored as 4-bit ADPCM samples in an external EEPROM. The quality of ADPCM is somewhat below "toll-quality" PCM, which is the standard used by telephone companies for digital voice transmission; however, it's better than many of the more complex compress-

sion algorithms used by some answering machines and cellular phones.

The compression algorithm was adapted from the code in a Microchip application note after it was optimized for 10-bit uncompressed samples. [1] Two samples were packed per byte and stored in the EEPROM.

Speech data is saved in a 64-byte ring buffer on its way to or from the EEPROM. This allows for the use of the EEPROM's internal 64-byte write buffer, which can write an entire buffer's worth of data in one 5-ms write cycle. With an 8-kHz sample rate, 128 samples (64 bytes) take 16 ms to collect, so there is plenty of time to write it out, even when allowing for I²C bus overhead.

Speech sampling is achieved at 8 kHz via the 10-bit A/D converter. I chose this rather than 8 bits to allow the Screener to work with a wider range of input levels from different phone lines. Playback uses PWM in 9-bit mode, using four times oversampling. The result is a 32-kHz PWM sample rate (but still an 8-kHz data rate), which eases the analog filtering requirements over the 16-kHz rate that would be necessary for 10-bit mode.

IN CONTROL

A single push button switch allows you to select the number of rings before answering (if pressed when the On/Off switch is on), or to record a new pair of greeting messages (if pressed when off).

The battery voltage is checked each time you pick up the phone. If it's low, the ringing sound played through the speaker is an on-off-on-off double ring, instead of the usual 2 s on, 4 s off.

SOFTWARE DETAILS

The code was written in assembler using Microchip's MPASM 3.20. You may download the code from the *Circuit Cellar* ftp site.

Debugging was accomplished the old-fashioned way by replacing the MCU with a PIC16F876 and using its serial port to output debugging data. Although it wasn't as powerful as using an ICD system, it proved ade-

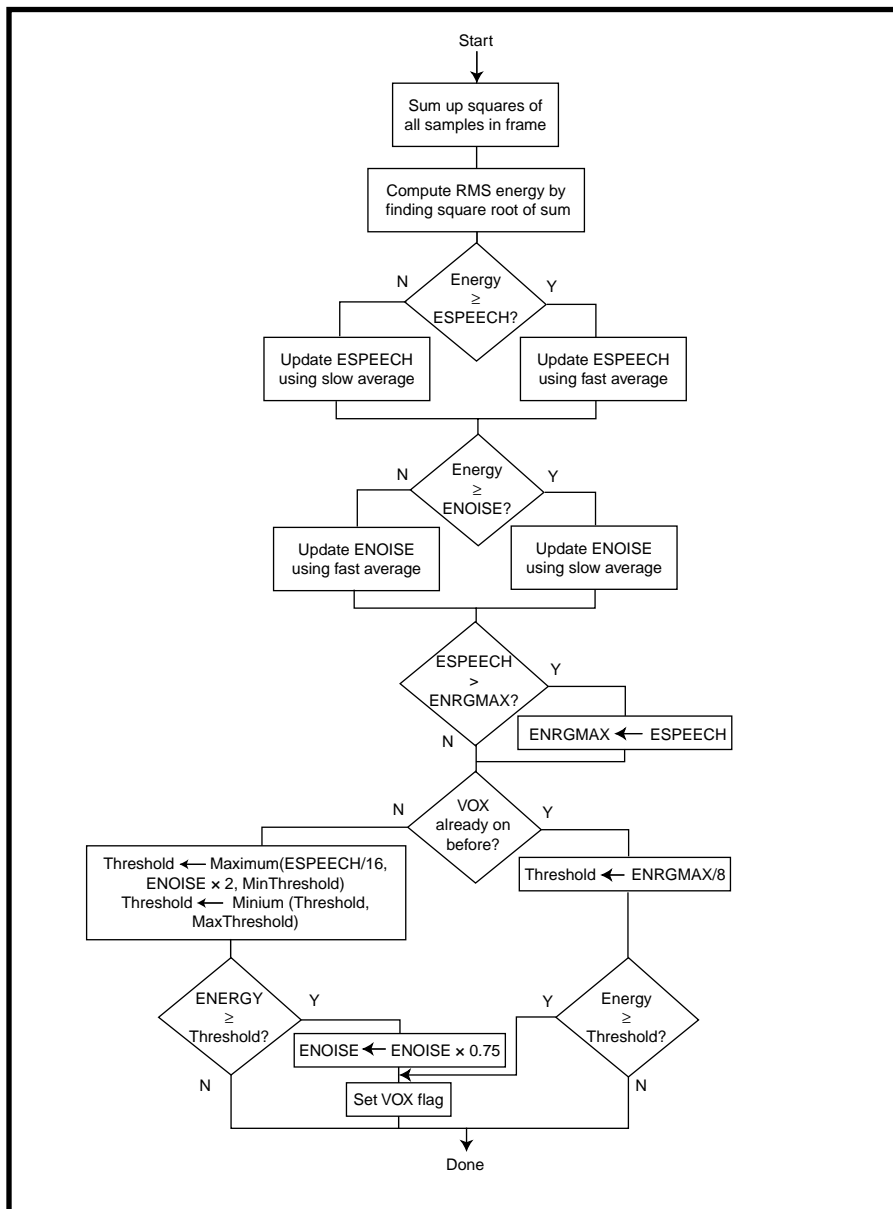


Figure 2—The VOX algorithm distinguishes between speech and background noise levels, and adjusts its thresholds accordingly.

quate enough to monitor the flow of software during development. The code uses most of the flash memory, the entire RAM, and a few EEPROM locations for configuration data.

After first applying power, the internal peripherals are initialized and a quick check is made to confirm that the external EEPROM is alive. Then, Timer2 starts, which functions as the primary time base. The 16.384-MHz oscillator is divided down to 32 kHz for the PWM clock and then to 8 kHz for the audio sample interrupt routine. This is divided down further to execute the frame and debounce routines one time every 200 samples, or every 25 ms. All of the hardware inputs and other status conditions are funneled through the debounce routine and fed to the main control code via two status bytes. This simplifies the software, because all of the control flow is determined by those two bytes.

MAIN ROUTINE

The main routine waits for the phone to ring or for the push button to be pressed. Then, it wakes up, executes the appropriate function, and goes back to sleep.

When the phone rings, it waits for a user-selectable number of rings before answering. Next, the Screener picks up the phone line, plays the first greeting to the caller, and waits for a response (see Figure 3). If it doesn't hear any sounds after 5 s, it hangs up and waits for another call.

If the Screener hears a voice, it waits up to 10 s for at least 1.5 s of silence. If there isn't a break in the speech, which is typical of a recorded message, then it hangs up. When a caller passes both of these obstacles, they are assumed to be a real person, and the Screener rings through its speaker.

The Screener immediately hangs up for the following reasons: if another phone on the same line is picked up; if the phone-line voltage is interrupted, indicating that the caller has

hung up; if the push button is pressed; or if the On/Off switch is turned off. Rather than check for each of these conditions throughout the main routine, a task-abort interrupt flag (TASK-INT) contains bits for each abort condition, which are set as needed. Then, the frame update routine checks each of the conditions and aborts the main routine if any become true. Figure 3 shows the currently active conditions for each operating mode.

If the push button is pressed and held down for several seconds while the On/Off switch is on, you can change the number of rings before answering by pressing the button one time for each ring. If the switch is off at the time, you can record a new pair of greetings using a locally connected phone.

FRAME UPDATE

The frame update routine runs every 25 ms. Rather than calling it directly

from the interrupt routine, it's called regularly by the main routine by checking the frame update flag (FRM-FLG) whenever there's a loop waiting for an event. FRMFLG is set by the interrupt routine every 200 samples.

This routine reads and debounces the inputs, and checks for an abort condition (see Listing 1). The two switches, ring detector, VOX, and the line-voltage detector are debounced using two separate on and off times for each input. Thus, multiple functions can be based on a single input. For instance, when the phone rings, the short debounce-time ringing bit would follow the ring cadence, and the longer one would indicate the overall ringing period.

INTERRUPTS

The highest priority is the sample interrupt, which runs at 8 kHz; it handles the A/D and PWM samples along with reading and writing to the external EEPROM. The mode flag determines which action will be taken depending on the current operating mode.

The VOX is the most time-intensive function, because each sample must be squared and summed as it's acquired. The routine was carefully optimized so that the interrupt routine loading is only 64%, leaving plenty of CPU cycles for the main routine.

Audio recording and playback, including ADPCM coding, is handled entirely at the interrupt level, using a 64-byte circular buffer feeding the I²C EEPROM. Because the I²C bus takes time to transfer data, only one bus transaction is performed per interrupt. A state machine steps through each I/O operation to execute a read or write of the buffer.

Audible ring-back and beep tones are generated with a pair of phase accumulators and a sine wave look-up table. Dividing down the frame counter and toggling the speaker output pin generate the two-tone ringing sound.

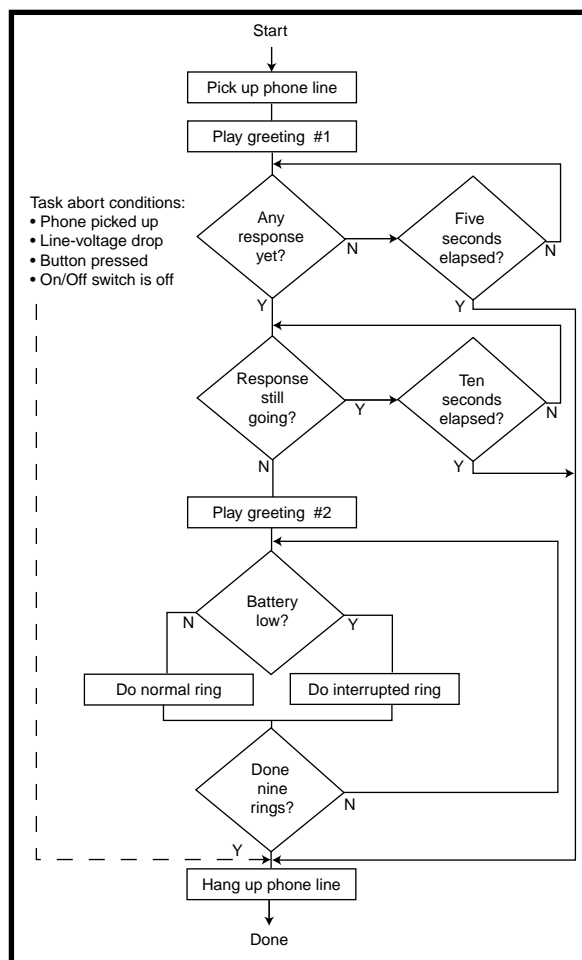


Figure 3—The heart of the Screener is the algorithm that decides if a machine or a human is calling. It is deceptively simple yet quite effective.

Listing 1—The debounce routine checks each of the input conditions (IFLAGS) and sets the status bits (SFLAG1 and SFLAG2) after applying the debounce times from ONTBL1/2 or OFFTBL1/2. This provides two different debounce times for each bit.

```

movlw  CNTRS           //Pointer to debounce counters
movwf  FSR
clrf   TBLIX-d        //Index into debounce threshold tables
                        //First check the current bit
movlw  1<<(8-NSFLAGS) //Get number of bits in use
NEXTBLP movwf MASK-d
andwf  IFLAGS-d,w     //Bit now on?
btfsc  STATUS,Z
goto   NOWOFF         //No
//If the bit is on, increment the on-counters, and set the corre-
sponding SFLAG1 bit if the short debounce time is reached
NOWON  incfsz INDF,w   //Increase On counter
movwf  INDF           //Save if no overflow
incf   FSR,f
clrf   INDF           //Clear Off counter
decf   FSR,f
movf   SFLAG1-d,w
andwf  MASK-d,w      //Was SFLAG1 bit on already?
btfss  STATUS,Z
goto   NOWON2        //Yes
movf   TBLIX-d,w
call   ONTBL1        //Get threshold1
subwf  INDF,w        //ON counter >= threshold1?
btfss  STATUS,C
goto   NOWON2        //No
movf   MASK-d,w
iorwf  SFLAG1-d,f    //Set SFLAG1 bit
//Repeat using SFLAG2 and the longer debounce time
NOWON2 movf   SFLAG2-d,w
andwf  MASK-d,w     //Was SFLAG2 bit on already?
btfss  STATUS,Z
goto   NXTBITI      //Yes
movf   TBLIX-d,w
call   ONTBL2        //Get threshold2
subwf  INDF,w       //ON counter >= threshold2?
btfss  STATUS,C
goto   NXTBITI      //No
movf   MASK-d,w
iorwf  SFLAG2-d,f   //Set SFLAG2 bit
NXTBITI incf   FSR,f
goto   NXTBIT
//If the bit is off, increment the off-counters, and clear the
corresponding SFLAG1 bit if the short debounce time is reached
NOWOFF incf   FSR,f
incfsz INDF,w       //Increase Off counter
movwf  INDF         //Save if no overflow
decf   FSR,f
clrf   INDF         //Clear On counter
incf   FSR,f
movf   SFLAG1-d,w
andwf  MASK-d,w     //Was SFLAG1 bit off already?
btfsc  STATUS,Z
goto   NOWOFF2      //Yes
movf   TBLIX-d,w
call   OFFTBL1      //Get threshold1
subwf  INDF,w       //nn counter >= threshold1?
btfss  STATUS,C
goto   NOWOFF2      //No
comf   MASK-d,w
andwf  SFLAG1-d,f   //Clear SFLAG1 bit
//Now repeat using SFLAG2 and the longer debounce time
NOWOFF2 movf   SFLAG2-d,w
andwf  MASK-d,w     //Was SFLAG2 bit off already?
btfsc  STATUS,Z
goto   NXTBIT      //Yes

```

(Continued)

If the phone line is picked up, the line voltage is sampled one time per frame. The A/D conversion is interleaved between a pair of audio conversions by setting Timer0 to generate an interrupt roughly halfway between audio samples. Note that the line voltage conversion is done at that time. Thus, the timing of the audio samples isn't affected.

THE MICROCONTROLLER

The PIC16F872 uses a 16.384-MHz ceramic resonator, which is a convenient multiple of the 8-kHz sample rate (see Figure 4). It controls an EEPROM via the I²C port and drives a speaker to generate an alternating tone ringing sound. In addition, it controls the phone line hook switch and power to the analog section via a DPDT relay.

I used a low-power relay that requires only 10 mA and is directly driven by a port pin. The On/Off switch is driven by an output port pin, which goes high in Sleep mode to eliminate the pull-up current. The prototype has an ICSP port for easier programming.

PHONE-LINE INTERFACE

In order to connect directly to a telephone line, there are a number of FCC and Underwriters Laboratories (UL) safety and performance requirements. To learn more about these requirements, take a look at the resources listed at the end of this article. Although the FCC's technical requirements are now handled by the Administrative Council on Terminal Attachments and published by the Telecommunications Industry Association, I'll refer to them by their former name.

Often, the most difficult requirements are the line-to-line and line-to-ground surge specifications, which are designed to protect against a lightning strike (FCC) or a direct power line short (UL). Because the Screener doesn't have external connections, the line-to-ground requirements are relatively easy: they're met by using a plastic box and properly isolating any exposed metal parts. UL typically allows the battery contacts not be isolated from the phone line, provided

the battery compartment requires a tool for opening it (e.g., a screwdriver or coin) and the product has a suitable warning message.

The line-to-line surges are somewhat trickier because anything connected to the line could be exposed. The surges are applied both on and off the hook. Different strategies were applied here for each case.

When the Screener is on-hook, the relay will protect anything downstream from the surge. The only two components that will be exposed to high voltage are C8 and R11, which are both rated for 1 kV—the highest surge voltage applied (see Figure 5). Most common 0.25-W resistors are rated for 500 V, so two can be used in series rather than using a more expensive part.

In the off-hook state, a surge suppressor, D2, limits the surge to 18 V. The circuit must withstand a 1000-V, 25-A, type-B metallic surge without damage; therefore, I used a special surge-withstand fuse that was designed to meet FCC requirements (i.e., TIA-968-A section 4.2.3.1, which was formerly FCC part 68, section 302 (c)(1)). All of the wiring from the phone jack to D2 must be able to han-

Listing 1—Continued.

```

movf   TBLIX-d,w
call   OFFTBL2      //Get threshold2
subwf  INDF,w       //Off counter >= threshold2?
btfss  STATUS,C
goto   NXTBIT       //No
comf   MASK-d,w
andwf  SFLAG2-d,f  //Clear SFLAG2 bit
                        //Go to the next bit, if any
NXTBIT  incf   FSR,f
        incf   TBLIX-d,f
        bcf   STATUS,C
        rlf   MASK-d,w      //Any more bits?
        btfss STATUS,C
        goto  NXTBLP       //Yes
        return

```

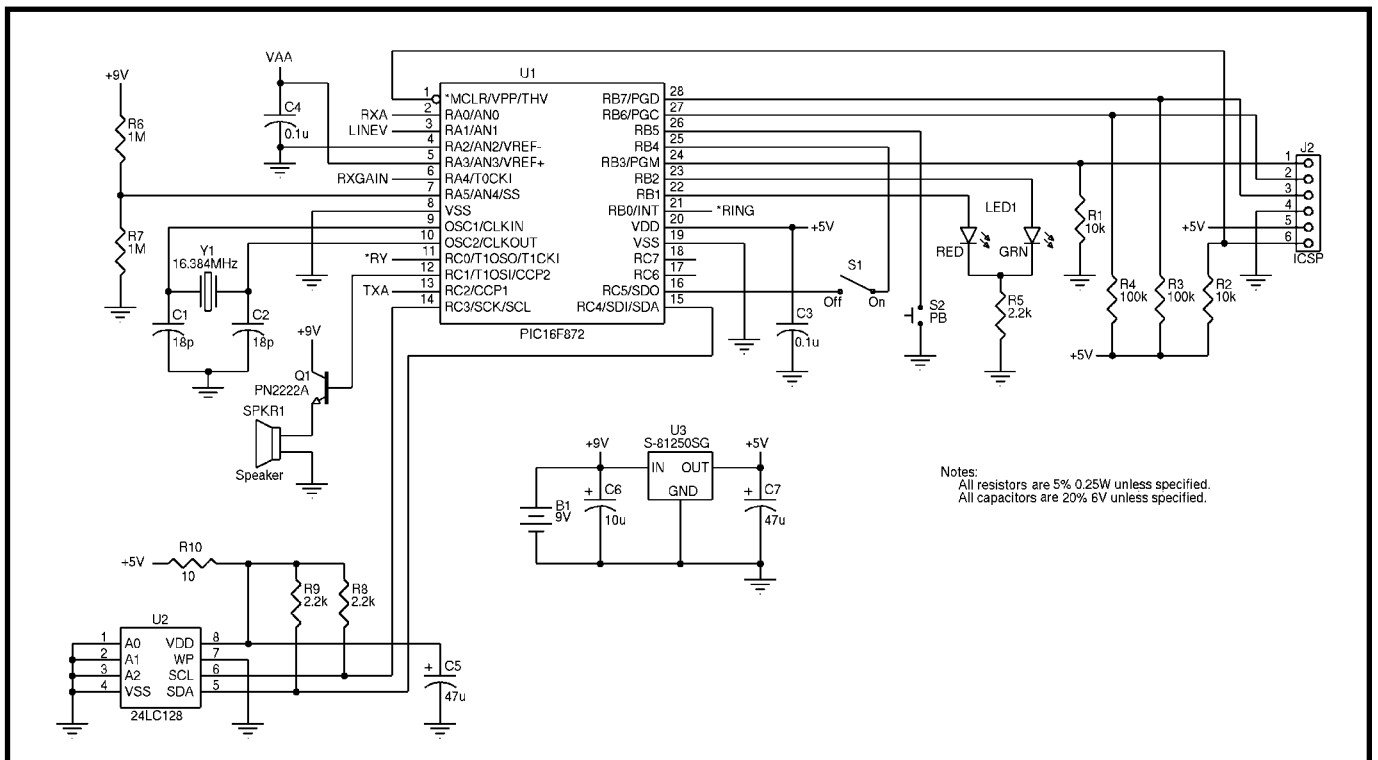
dle this surge current. Otherwise, surge testing may result in a dramatic separation of the wiring from its former connections!

An active current sink (Q3 and nearby components) controls the DC-line current with a dynamic resistance of 200 Ω. This is about two times the resistance of a typical telephone and was chosen so the line voltage would be noticeably higher than normal, making it easier to tell when a phone is picked up.

When the Screener first picks up the line, the current sink doesn't start drawing current, because C11 isn't

charged yet. If no other phone is picked up, the line voltage will rapidly rise until it's limited by D2 (around 18 V). After 10 ms, if it doesn't go high enough, the Screener immediately hangs up the line and waits for another call.

Ring detection is done using a diac, which is a breakdown device that triggers at 60 V. This powers a bidirectional optoisolator driving the PIC's interrupt input. When the phone rings, the signal toggles at 40 Hz. The optoisolator is only used to protect the PIC from lightning surges, because it isn't otherwise needed for electrical



Notes:
All resistors are 5% 0.25W unless specified.
All capacitors are 20% 6V unless specified.

Figure 4—An ultralow-power linear regulator and switchable pull-down for S1 result in miniscule power consumption when the MCU is sleeping.

isolation.

ANALOG PROCESSING

The analog output from the PWM is filtered by a second-order, Sallen-Key low-pass filter with a 4-kHz cut-off frequency. This provides adequate filtering for the 32-kHz PWM sampling rate and AC coupling for the output.

The input to the A/D converter is low-pass filtered at 3.5 kHz after passing through a variable-gain stage. For normal VOX operation, the RXGAIN signal is high, which allows Q4 to turn on and provides a gain of 24.5 (units?) for the caller's voice. When recording a new greeting using a locally connected phone, RXGAIN is low, providing a gain of approximately three. This keeps the audio signal within the range of the A/D converter.

POWER

A 9-V battery, feeding a linear regulator with ultralow quiescent current, supplies the power. The brownout timer is disabled to minimize the standby power consumption, which results in a total draw of approximately 5 μ A while waiting for a call.

When active, the entire system draws 40 mA for about 40 s for each call that rings through or 20 mA for 15 s if it hangs up. A typical alkaline battery will last for a few thousand calls. Hopefully, you don't get that many calls too often!

Battery voltage is monitored through a resistive divider (R6, R7), which uses 1-m Ω resistors to avoid putting a significant load on the battery, because the current is drawn

continuously. However, this large resistance causes an error of as much as 50 mV in the measured voltage because of the 100-nA A/D pin input current. This much error is not a concern, because the low battery threshold is not critical.

CONSTRUCTION

With the exception of a small portion of the phone-line interface, building the Telephone Screener was a mostly straightforward process. The one connection between the top terminal of J1 and K1 is subject to high voltage when the phone line is on-hook, so the corresponding wiring should be kept clear of any other circuitry.

As with any mixed signal device, the analog and digital power and

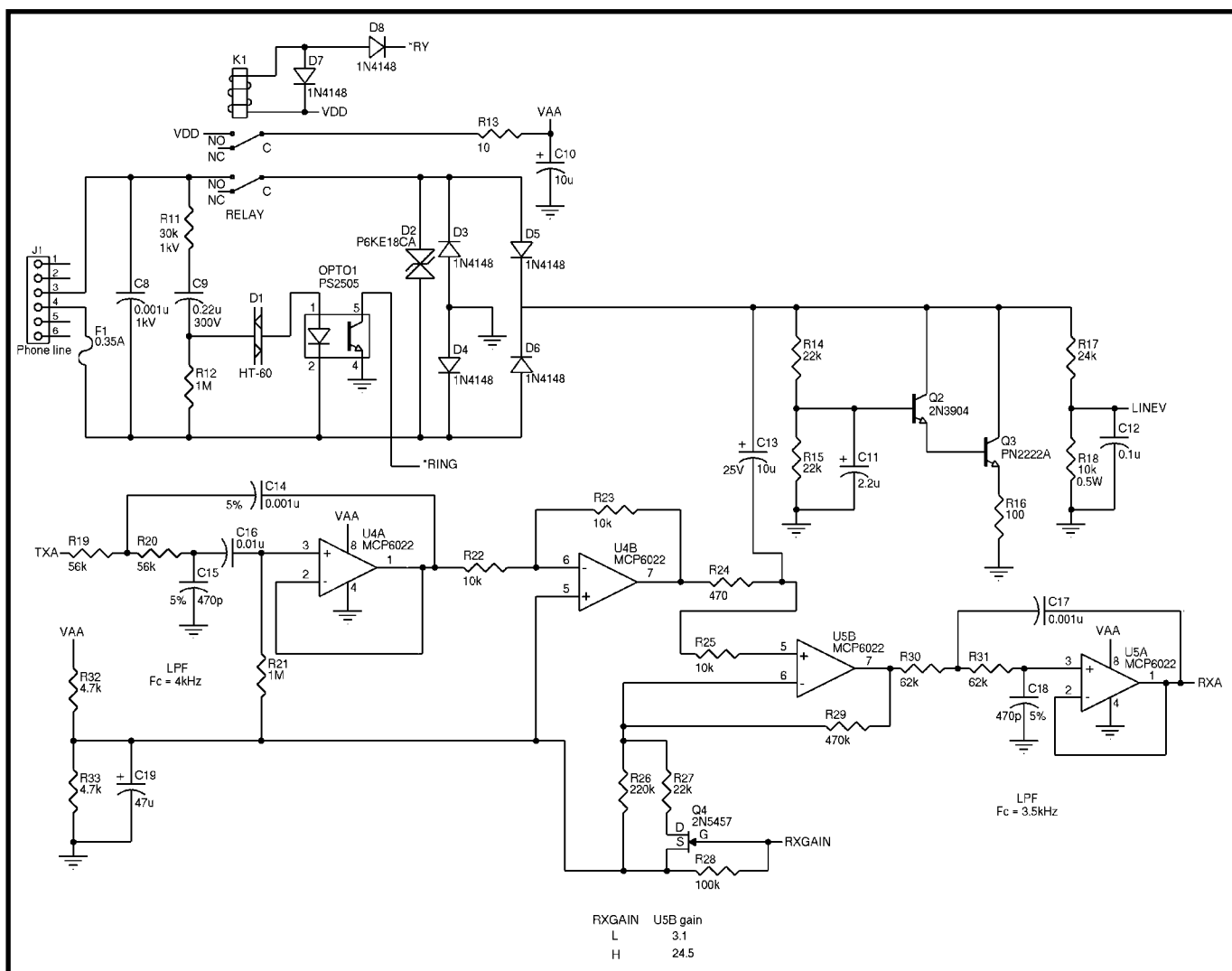


Figure 5—The phone-line interface consists of the hook relay, ring detector, and active current sink. A pair of dual op-amps is used for the antialiasing and reconstruction filters. The receive gain is switchable, allowing either the caller's voice or a locally connected phone (used for recording a greeting) to be the right level for the A/D converter.

grounds should be kept separate, and any high-current paths (e.g., the speaker, EEPROM, and relay coil) should have separate power runs. I still ran into a problem where high-current spikes from the EEPROM were getting into the audio, but adding a filter (R10, C5) to its V_{DD} pin tamed the noise.

There mustn't be any exposed metal parts that are connected to the circuitry, because these would be a shock hazard with the nonisolated interface. This meant that I had to take extra care during debugging to make sure that the ICSP programming port and the phone line weren't plugged in at the same time.

IN MY DREAMS

After several months of use, the Stealth Telephone Screener has successfully blocked all but one of dozens of automated calls, and not even one real human caller. The present version of the hardware won't work with an answering machine, because the Screener would pick up first, and it has no way to transfer the call to the machine. An enhanced version could include a ringing generator instead of the built-in speaker. This would allow a phone or answering machine to ring normally after the call has been screened.

I have attempted to present something to help reduce some of the annoyance in our lives. Someday, perhaps, enough people will use screening devices to render telemarketing a thing of the past—or so I can dream!

Richard Wotiz earned a B.S.E. in Electrical Engineering and Computer Science at Princeton University. For the past 12 years, he's been running his own hardware consulting business that specializes in consumer products and toys. You may reach Richard at dick601@mystics.org.

PROJECT FILES

To download the code, go to ftp.circuitcellar.com/pub/Circuit_Cellar/2003/156.

REFERENCE

- [1] R. Richey, "Adaptive Differential Pulse Code Modulation Using PICmicro Microcontrollers," AN643, Microchip, 1997.

RESOURCES

Telecommunications Industry Association, "Technical Requirements for Connection of Terminal Equipment to the Telephone Network," TIA-968-A, 2002.

Underwriters Laboratories, Inc., "Standard for Information Technology Equipment Including Electrical Business Equipment," UL 1950, 3d ed., 1995 (revised 2001).

SOURCES

TIA-968-A Standard

Administrative Council on Terminal Attachments
www.part68.org/documents.cfm.

UL Standards

COMM 2000 (distributor)
(888) 853-3503
www.comm-2000.com

MCP6022 Op-amp, MPASM 3.20, PIC16F872 Microcontroller, 24LC128 EEPROM

Microchip Technology, Inc.
(480) 792-7200
www.microchip.com

220003 Surge-withstand fuse

Littelfuse, Inc.
(847) 824-1188
www.littelfuse.com