BUILD AN ALPHANUMERIC PAGER DECODER

Catch beeper messages from the airwaves with this device and view them on your computer screen.

BY ALAN D. JONES

Digital pagers, or beepers as they're commonly called, have become exceedingly popular in the past decade. Pagers of all kinds are clipped to the belts of millions of people today, and it's quite likely that one of those people is you.

A signal intended for a digital pager consists, among other things, of an individual pager address followed by the message to be displayed on the pager. Note that the word "digital" here refers to any pager that receives binary data, including both numeric (digits only) and alphanumeric (full text) pagers. The majority of messages in a broadcast will be simple telephone numbers or digit codes, but people with alphanumeric pagers are increasingly making use of detailed text messages, often of considerable length.

But how do you know if the messages being transmitted are being received correctly? There's no backup message generated by paging services, and you therefore have no way of determining if you're getting the whole message. What's needed is a way of getting a second look at a message.

The *Alphanumeric Pager Decoder*

described in this article, together with a scanner radio and a PC, will permit you to decode the messages that are transmitted to about 80 or 90 percent of beepers that are commercially available, and view them on your computer screen. You could, for example, keep a log on disk of all messages transmitted to your own pager and verify that all were received by the beeper. Also, corporations could keep time-stamped logs of all messages sent to their in-house pagers.

Units like the Decoder are quite effective, and for this reason are even used by law-enforcement agencies. With such high-tech help, the good guys in blue can keep tabs on the dealings of known nefarious characters.

How Pages are Transmitted.

Paging channels can be found scattered around the VHF (152 MHz) and UHF (454 MHz) bands. In most metropolitan areas, a large number of paging channels can be found within the dedicated paging band from 929 to 932 MHz. To discover the frequency used by a specific paging service, just look at one of their pagers. There will almost always be a sticker indicating

the reception frequency.

Pager signals of the type we are interested in here are modulated by the "direct FSK" method. This means that the RF carrier is switched between center frequency + 4.5 kHz and center frequency − 4.5 kHz, to represent binary "1" and "0," respectively. Bit rates currently range from 300 to 2400 bits per second.

The most commonly used data format is POCSAG (Post Office Code Standard Advisory Group). This coding standard was devised in a series of meetings of industry representatives hosted by British Telecom in 1978 and 1980. POCSAG is a 32-bits-per-word synchronous error-correcting code using a 17-word frame. It is broadcast at 512, 1200, and 2400 bits per second. On most paging channels you will hear the signal switching rapidly between different bit rates as various pagers are addressed. To learn to recognize POCSAG by its distinctive sound, just use the search mode on your scanner in the 929- to 932-MHz range. About 90 percent of the signals you hear will be POCSAG at one of the three standard bit rates (see the "POCSAG Messages" box for more information).
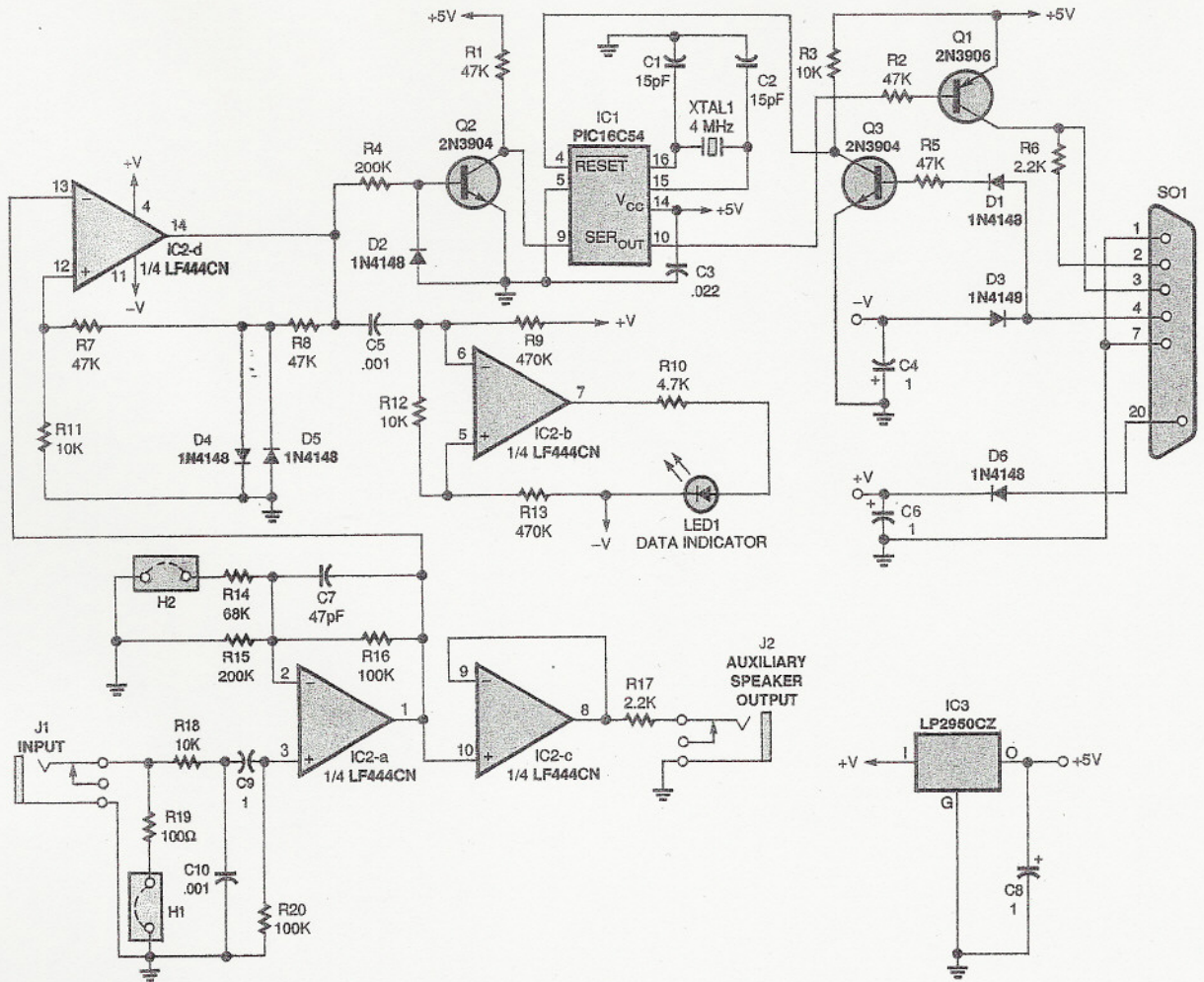
Fig. 1. Here is the schematic for the Alphanumeric Pager Decoder. Many sections of the circuit take their power from two pins of a PC serial port, through DB-25-connector SO1. Regulator IC3 also converts that power to a 5-volt source for use by other parts of the circuit.

Using Your Scanner. A perfectly demodulated paging signal should appear on an oscilloscope as a series of rectangular pulses of varying width (squarewaves). This signal exists at some point within the circuitry of any radio receiver. Unfortunately, most low- to moderate-cost scanners were designed specifically to listen to voice transmissions. As a result, that nice clean squarewave passes through several stages of lowpass and high-pass audio filtering to optimize the sound quality from the speaker. The result is that the signal available at the external speaker or earphone jack bears little resemblance to the original source.

One of the functions of the Decoder is that of reconstructing the original data waveform from the highly distorted waveform available at the audio output of a typical scanner. Ex-

amination of the outputs of several popular scanners reveals certain common characteristics. One of these is that all DC information is lost; if a long string of 1's or 0's is encountered (no bit transitions), the output quickly settles to a center zero point regardless of the polarity of the binary data. Another is that any bit transition causes an initial spike of the proper polarity followed by a moderate-to-severe overshoot of the opposite polarity. The second overshoot (back in the original direction) is usually well-damped and of much lower amplitude. To make matters worse, the "ringing" frequency is often roughly equal to half the bit rate of a 1200-bps data stream, causing transition-induced spikes to sometimes superpose themselves on the first overshoot from a previous transition.

The Decoder attempts to recon-

struct the original data by the use of a Schmitt trigger. There are two opportunities for trigger points on the audio waveform: the initial spike and the first overshoot. Which one is best depends on whether or not the amplitude of the initial spike is significantly higher than the first overshoot. The general lack of a major second overshoot is the detail that allows us to get away with using the first overshoot instead of the spike. For any particular scanner, direct experimentation will determine the best location for the trigger points.

If your scanner has a "discriminator" or raw-data output, then you should by all means use it with the Decoder. Most do not, however. So, as we'll see later, you will probably have to locate this unadulterated signal within the internal circuitry of the scanner. A little work with a soldering iron can bring out an extra pair of wires (or a jack)

providing the desired output. Because of the audio filtering, you can normally decode 1200-bps and lower transmissions, but it's pretty hopeless to extract useful data at 2400 bps without a direct discriminator output.

Just because you have an accurately reconstructed binary data stream doesn't mean that the problem of interpreting pager signals is solved. We could try level-translating the signal to RS-232 voltages and feeding it into a serial port, but the following must be considered: (1) bit rates may change unexpectedly, and (2) POCSAG is a 32-bit synchronous format that is incompatible with the data input mechanism of the type of UART that is normally used in PCs.

There is some temptation to solve both of these difficulties by feeding the signal to a "handshake" line of the serial port (instead of the normal data line) and using highly timing-intensive software to measure individual bit transitions in order to determine the current bit rate and extract the correctly synchronized data words. In fact, there exist products on the market that do exactly this, and the ingenuity that must have been required to write such software is something to be admired. The disadvantage of this approach is that the software running on the PC must perform timing tasks at the sub-millisecond level in a way that essentially prohibits operation under multitasking operating systems such as Microsoft Windows. Such programs tend to run under MS-DOS only and must "own" all the CPU time in order to function correctly.

The Decoder overcomes this disadvantage by adding one more processing step between the reconstructed data stream and the PC serial port. A Microchip PIC16C54 microprocessor is used to reformat the data into a form that is acceptable to a standard PC UART. In order to retain the versatility and general applicability of this device for future applications in decoding other digital data stream formats, the embedded software for the PIC CPU operates simply as a constant-rate sampler, continuously taking samples of the state of the data stream at approximately 9600 samples per second. Each time a group of eight samples has been accumulated, the group is transmitted to the PC serial port as a conventional asynchronous byte, including start and stop bits, at 19,200 bps. This amounts to four samples per bit at a 2400-bps incoming data rate (more for lower rates), which is adequate for purposes of software-based bit synchronization.

The work of determining the data rate and subsequently converting the data to a usable one-bit-per-data-bit stored format is still handled by the PC itself. The advantage of this method is that, since data is received by the PC serial port in the standard way, the usual operating-system-supplied serial port device drivers can be used to receive and initially buffer the data. This buffer can be occasionally read and analyzed by an application program that, because of the buffering in the device driver, can easily run in a multitasking environment.

The writing of PC software to bit-sync, word-sync, error-correct, decode, and display the data stream from this device would be a project of greater magnitude than the physical construction itself. However, a program that performs these functions under either Microsoft Windows 3.1 or Microsoft Windows 95 is available at no charge via the Internet at http://www.cylexinc.com/download.htm or from ftp.gernsback.com. This program decodes POCSAG at all three standard rates; if you need to decode another coding format, you will need to write your own software for now. Note that any program accepting data from this device must be capable of accommodating a never-ending stream of 19200-baud data and performing a fair amount of computation on that data at real-time speeds. It is necessary to use a reasonably fast PC to run such a pro-

PARTS LIST FOR THE ALPHANUMERIC PAGER DECODER

SEMICONDUCTORS
IC1—PIC16C54 preprogrammed microcontroller, integrated circuit
IC2—LF444CN quad low-power op-amp, integrated circuit
IC3—LP2950CZ low drop-out 5-volt regulator, integrated circuit
Q1—2N3906 general-purpose PNP silicon transistor
Q2, Q3—2N3904 general-purpose NPN silicon transistor
D1–D6—1N4148 general-purpose silicon diode
LED1—Light-emitting diode, 3-mm diameter, red

RESISTORS
(All resistors are ⅛-watt, 5% units.)
R1, R2, R5, R7, R8—47,000-ohm
R3, R11, R12, R18—10,000-ohm
R4, R15—200,000-ohm
R6, R17—2200-ohm
R9, R13—470,000-ohm
R10—4700-ohm
R14—68,000-ohm
R16, R20—100,000-ohm
R19—100-ohm

CAPACITORS
C1, C2—15-pF, ceramic-disc
C3—0.022-µF, Mylar
C4, C6, C8—1-µF, 16-WVDC, tantalum
C5, C10—0.001-µF, ceramic-disc
C7—47-pF, ceramic-disc
C9—1-µF, nonpolarized Mylar (see text)

ADDITIONAL PARTS AND MATERIALS
XTAL1—4-MHz crystal in HC-49/US package
J1, J2—3.5-mm mono phono jack
SO1—DB-25 female connector with solder-cup terminals
H1, H2—Two-pin header (right-angle break-off pin strip, 0.1-inch spacing)
Printed-circuit materials, DB-25 connector hood designed to accommodate a DB-25 at both ends, jumper shunts (0.1-inch spacing) to fit H1 and H2, cable with a 3.5-mm plug on each end, wire, solder, hardware, etc.

Note: The following items are available from Cylex Inc. (2501 Afton Court, League City, TX 77573-3438; Tel. 800-356-7047; Fax: 713-332-4730): double-sided etched and drilled PC board with matching plastic case—$25.00; kit of parts including pre-programmed PIC16C54 (no board or case)—$35.00; pre-programmed PIC16C54 only—$11.00. When ordering by mail add $5.00 shipping and handling to all orders. Texas residents please add an additional 7.25% sales tax. Visa, MasterCard, American Express, and Discover cards will be accepted. A floppy disk containing the AccuPage Radio Monitor program will be included free of charge with all orders.

gram; we recommend a 33 MHz 486 as a minimum.

Circuit Description. The schematic for the Decoder is shown in Fig. 1. Basically, the circuit consists of four blocks: the power supply, input-signal processing, Schmitt trigger, and digital sampler/UART. All necessary power is drawn from the handshake lines of the serial port itself. The request-to-send line (pin 4 of SO1) provides negative voltage and the data-terminal-ready line (pin 20 of SO1) provides positive voltage (software must set these outputs appropriately). Regulator IC3 creates a 5-volt logic supply for microcontroller IC1. The RTS line (negative supply) doubles as a reset control for IC1 when it is set momentarily positive, while the data-output line from the PC serves as a separate negative supply to produce the necessary voltage swing to drive the PC's data input at RS-232 levels.

Resistor R19 is a simulated speaker load for the scanner if needed. Components R18 and C10 form a lowpass filter to remove the 455-kHz IF components (and harmonics) that are often present at discriminator outputs. Both
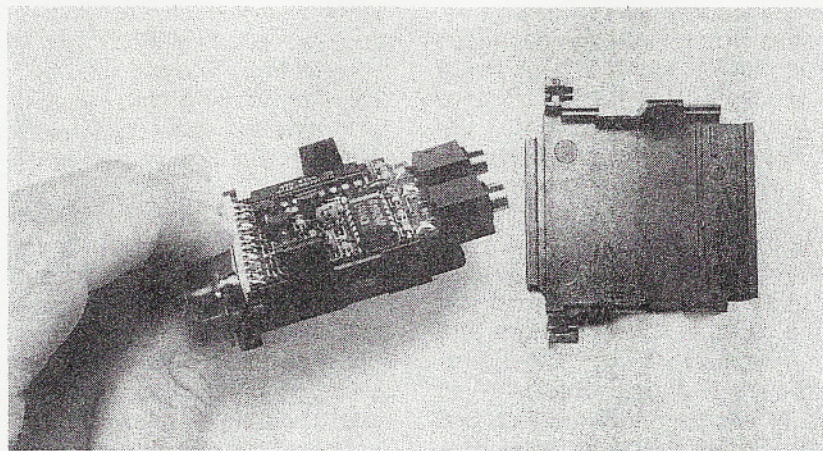
A properly assembled Decoder PC board will fit inside a DB-25 connector hood.

C9 and R20 eliminate any undesired DC components that may exist in the scanner signal. One section of an LF444CN, IC2-a, is simply a buffer and gain block with two jumper-selectable gain settings; IC2-c buffers the signal to the auxiliary speaker output.

Section IC2-d and its associated feedback components form a Schmitt trigger with thresholds of approximately +0.1 and −0.1 volts. Capacitor C5, section IC2-b, and their associated resistors are set up to cause LED1 to flash on each negative transition of the Schmitt trigger output as an aid in setting the scanner output-level control. Transistor Q2 converts the rail-to-rail swing from IC2-d to logic levels for input to microcontroller IC1, a PIC16C54.

The PIC contains on-chip PROM that must be programmed with the small program whose source code and compiled hex file can be downloaded from the Cylex Internet site mentioned earlier, or the Gernsback FTP. Pre-programmed chips are also available from a source mentioned in the Parts List. This program is clock-rate sensitive and will not work unless a 4-MHz crystal is used.

Construction. Layout of the circuit is non-critical. For that reason, any reasonable breadboard assembly technique can be used to build the Decoder. The only precautions you'd have to follow are: (1) Be sure that crystal XTAL1 and its shunt capacitors C1 and C2 are close to microcontroller IC1 with short lead lengths, and (2) Keep C8 close to IC3, and C3 close to IC1. However, when it comes to size, building the circuit on a breadboard is not desirable.

For the most compact assembly, you might want to build the circuit in the fashion the prototype was assembled—on a double-sided printed-circuit board. That way, the entire circuit will fit inside a plastic DB-25 connector shell. If you'd like to etch your own double-sided PC board, you can use the solder- and component-side foil patterns shown in Figs. 2 and 3, respectively. Or, you can order a drilled and etched board from the source mentioned in the Parts List.

If you build the Decoder on the circuit board, use the parts-placement diagram shown in Fig. 4 as a guide. Be careful about the sequence in which you install the parts. Note that some of the discrete components in Fig. 4 are shown made up of dashed lines. Those mount on the solder side of the board directly under ICs. Solder those parts in place and clip their leads before you install ICs. Do not use sockets under the ICs if you plan to enclose the circuit in the plastic shell; there is simply not enough room.

It is also important to solder all TO-92 parts (Q1, Q2, Q3, and IC3) with their plastic cases all the way down to the board, again because of the shortage of room. Jacks J1 and J2 and SO1 go against the edge of the board and some of their pins solder to each side. Be sure that SO1 is perfectly straight against the board edge or the connector shell will not fit. The LED should protrude through a hole in the connector shell drilled or punched to accommodate it. Also, make sure that C9 is small enough to fit in the space allowed on the board.
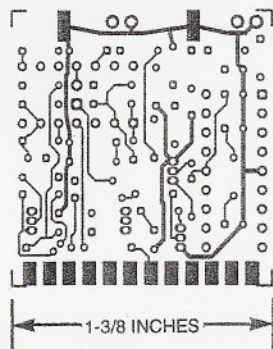
Fig. 2. This is the solder side of the Decoder circuit board.
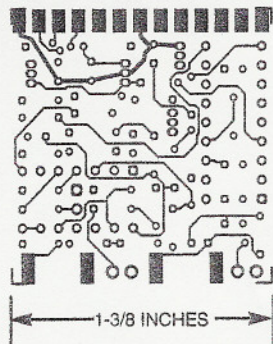
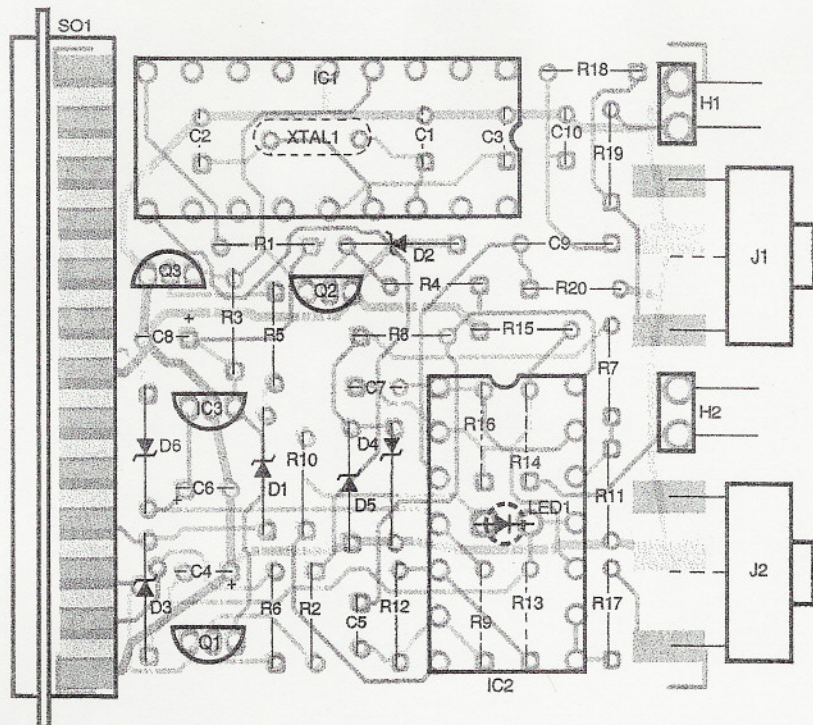*Fig. 3. Here's the component side of the board.*

Fig. 4. Use this parts-placement diagram as a guide when assembling the Decoder on a PC board. Note that the parts drawn with dashed lines are mounted on the component side of the board.

Keeping all those assembly tips in mind, this is the recommended sequence to optimize putting together the circuit board: Solder the two 3.5-mm phono jacks to the board, centering them against the edge. The ground lug on each will need to be bent about 45 degrees in order to touch the pads on the board. Next mount the DB-25 connector to the board using only pins 1 and 13. Temporarily place the board into one side of the plastic shell. If things do not line up properly, reheat the solder joints and adjust the position of the connector. Then solder the remaining pins of the DB-25.

Using the plastic shell half containing the LED hole as a guide, install the LED on the solder side of the board. Be sure to get the polarity correct. We stress this because the installation of IC2 later will make desoldering of the LED difficult. Install C1, C2, C3, X1, R9, R13, R14, and R16 on the solder side of the board as well. Leave about 1 mm of extra lead length on C1–C3 and bend these parts down flat against the board in the direction away from the crystal.

Install all remaining parts on the component side of the board. Be sure that all parts, especially Q1–Q3, and IC3, are flush against the board. Test the circuit before installing the plastic shell; the shell is difficult to take apart.

Checkout and Adjustment. To power up and test the circuit, it is necessary to have either the AccuPage Radio Monitor program for the PC (mentioned earlier) or a test program that sets the COM port as follows: RTS = 0, DTR = 0 for at least 0.1 sec, followed by RTS = 1, DTR = 0 indefinitely. This resets IC1, then configures the lines to provide power. Start with the device connected to the COM port but nothing connected to J1 or J2. Measure the voltage (relative to circuit ground) at IC1 pin 14 and IC2 pin 11; these should be + 5 volts and -6 to -11 volts respectively. The LED should not be lit.

Next, tune your scanner to an active paging frequency. Use an appropriate cable to connect the output of the scanner to J1. If you are using the speaker/earphone output of the scanner, install a connecting pin to header H1; remove it if you are using a discriminator output (more on that later). Leave the pin on header H2 off; it is needed only in a small percentage of cases with discriminator outputs and almost never with speaker

outputs. If you are using a discriminator output, then the following paragraph referencing volume-control settings does not apply; you should be able to just plug and go. If the LED does not come on during transmissions, then install the pin on H2.

Set the volume control to its minimum position. Connect a small speaker or earphone to J2. Be sure that the COM port is set correctly and that you check the "Enable" checkbox near the top of the AccuPage Radio Monitor screen. Gradually increase the volume setting until the LED begins to glow continuously during transmissions; stop at this point. Do not change the volume between transmissions; you will just be turning the control with no reference. During a paging transmission, the apparent intensity of the LED should appear to waver in sync with the sound you hear, but it should not go out except between transmissions. (If you have an oscilloscope, connect one channel to pin 1 of IC2 and the other to pin 9 of IC1 to see how the circuit is interpreting the analog waveform.)

It is worth pausing here for a moment to emphasize the importance of clean reception on the scanner. Move or re-orient the antenna as necessary! The sound you hear should be as free as possible from hiss or crackling noises. The sensitivity and selectivity of a wideband receiver is often not as good as that of a single-frequency pager receiver, and getting good reception is subsequently more difficult.

Watch the "Signal" indicators near the top of the AccuPage Radio Monitor screen. The green to red ratio corresponds to the correct to errors ratio. The left indicator is the one to watch initially; the right indicator always starts at "100% bad" and represents a cumulative weighted average over the last several seconds of valid data. We are looking for the left indicator to show mostly green. White indicates no valid data at all. If the indicator remains white, and the LED glows as described, try toggling the "Inverted Data" checkbox. Every scanner is different and the output polarity of yours may be backwards from the program's convention.

If you just can't seem to get any indication on the signal indicator, try increasing the volume setting by tiny

incremental amounts, being sure to try both settings of "Inverted Data" at each position. Once the indicator shows some green, then keep adjusting the volume for best results (minimum red). If you are getting readings but are unable to reduce the "bad" (red) percentage to a small value, then approach from the other direction. Set the Inverted Data checkbox to its opposite setting (to trigger on the overshoot as described earlier) and increase the volume to a much higher setting. Then decrease the volume incrementally, searching for an optimum setting based on the Signal indicators.

The AccuPage Radio Monitor program by default logs and displays all messages that it decodes. If you would like to see only text messages, or to filter the messages so that only those for particular pagers are logged, select "File/Search List" from the menu bar. A typical paging service might easily transmit 100,000 pages per day, and the message log file generated by logging all of them will rapidly grow to a size of many megabytes.

Troubleshooting. The following are some of the most likely reasons why the device might not work properly:

(1) Connector problems at the scanner output.
(2) Wrong COM port set up in software.
(3) "Inverted Data" setting is incorrect.
(4) Computer is too slow (use a 486/33 or faster).
(5) Weak or noisy radio reception.
(6) Wrong settings for pins in H1 and H2.
(7) The received signal is not direct FSK.
(8) Wrong scanner demodulation mode (should be narrow FM).
(9) The received signal is not POCSAG (there are other paging formats).
(10) Soldering problems, wiring errors, damaged components, etc.

Obtaining Unfiltered Scanner Audio. Every scanner is different, and it would be difficult to provide specific instructions for tapping into the unfiltered demodulator output for every type of scanner on the market (though we will give tips for one later on). Here are some general suggestions:

First of all, get a schematic of the scanner! Without this, you still might have success, but it's going to take an oscilloscope and a lot of patience.

Look for the audio amplifier circuit. This will usually consist of two or three stages of amplification, probably with a second-order lowpass filter and a second-order highpass filter somewhere in the chain. Try to obtain a take-off point at the beginning of the amplifier chain. The signal here will probably be only a few tenths of a volt, should look like square waves, and is likely to have a lot of 910 kHz superimposed on it (twice the lowest IF).

As an alternative (particularly if you don't have a schematic), look for the FM demodulator. In many scanners, this is the popular Motorola MC3361 chip. If you find one of these in your scanner, then pin 9 is the unfiltered demodulator output. Depending on the external circuitry, it is possible that pin 11 is also a good place to get a signal. Use an oscilloscope if possible to select the "squarest-looking" signal.

In addition to the signal take-off point, you must of course also bring out the scanner's circuit ground. **Warning:** In some handheld scanners we have examined, the "case" side of the external speaker jack is *not* grounded. Look on the circuit boards
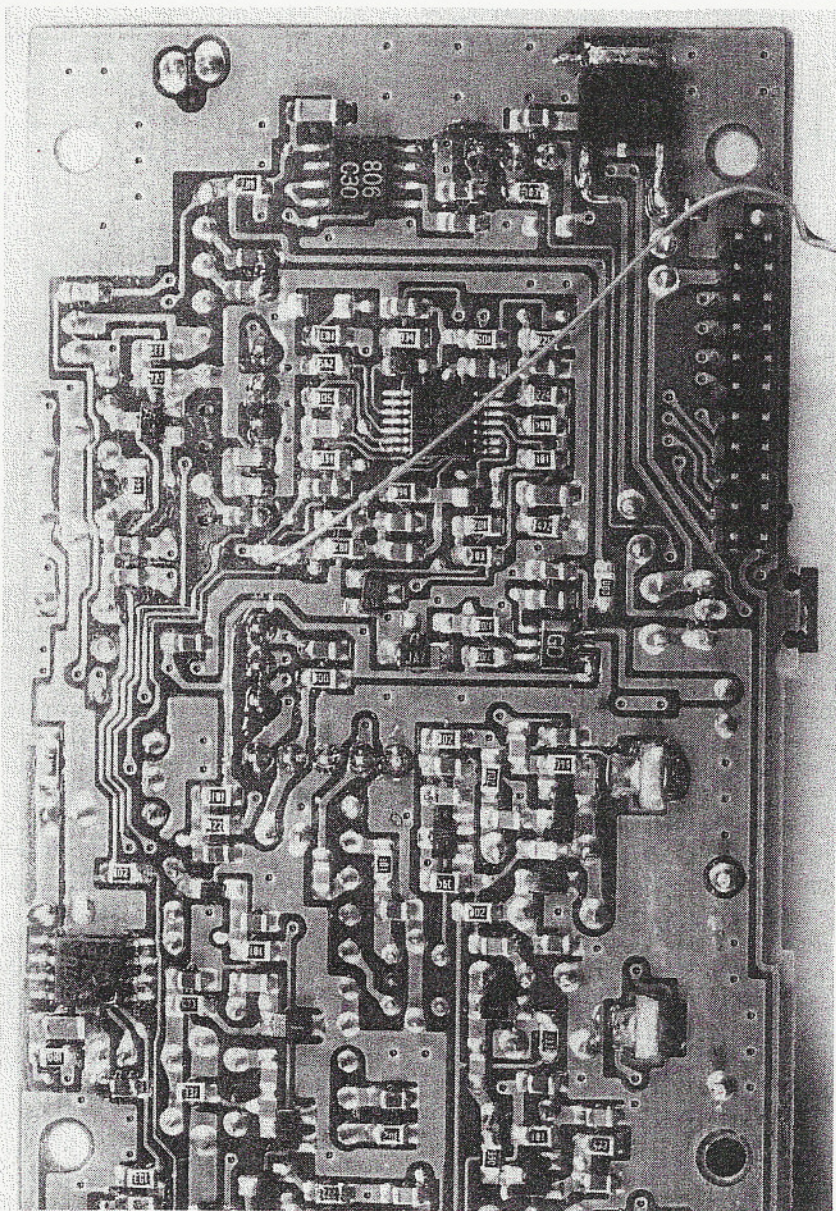
Fig. 5. Once the board from the Uniden SC-150 is removed (see text), use this photo as a guide to locating the signal-connection point for demodulator output.