

# INTERFACE YOUR COMPUTER TO THE "POOR MAN'S SPECTRUM ANALYZER"

*Add features found in expensive commercial units to your "Poor Man's Spectrum Analyzer"*

---

Since the publication of the "Low-Cost Spectrum Analyzer with Kilobuck Features" by Robert Richardson, W4UCH, in the September 1986 issue of *HAM RADIO*,<sup>1</sup> there have been several follow-up articles about modifying and improving this interesting project.<sup>1,3,4</sup> I purchased the tuners and receiver board from WA2PZO\* and built my own version. I combined the VHF "cable ready" tuner and the VHF/UHF tuner (the VHF section isn't used) in a single enclosure and now have a spectrum analyzer that covers from 1 to 900 MHz in two ranges.

## Background

I added a sawtooth generator to produce the frequency sweep described by Joe Carr in the September 1987 issue of *HAM RADIO*.<sup>3</sup> This provided a good oscilloscope display of the frequency spectrum. A turn-counting dial, attached to the multi-turn frequency-control potentiometer, let me set the center frequency accurately after calibrating the dial. The more expensive spectrum analyzers offer additional features like calibrated display (amplitude and frequency), programmable start and stop frequencies, and the capability to plot the spectrum display.

I wanted to add these features to my unit,

so I interfaced the "Poor Man's Spectrum Analyzer" to my IBM AT compatible computer and wrote a control program in Microsoft® Quick Basic.\* The interface requires a digitizer board that's mounted inside the computer, a frequency control board mounted with the spectrum analyzer, and an interconnecting cable.

The computer controls user-selected start and stop frequencies through the parallel printer port. After the amplitude versus frequency data has been acquired, it's plotted on the screen. The "print screen" command then plots the data on paper via the printer. **Figure 1** shows my spectrum analyzer and the computer interface.

The program lets you select the start frequency in two bands (1 to 500 MHz and 393 to 900 MHz) for use with the two tuners available from WA2PZO. You choose the start frequency, and the program establishes it using a 12-bit D/A converter (DAC). The frequency sweep uses an 8-bit DAC to obtain high resolution over a limited frequency range above the start frequency. The maximum permissible stop frequency is calculated based on the start frequency, and then displayed. After you select the desired stop frequency, the program makes one sweep and displays the plot on the monitor screen.

\* WA2PZO, Science Workshop, Box 393, Bethpage, New York 11714.

\* IBM AT is a registered trademark of IBM Corporation. Quick BASIC and GW-BASIC are registered trademarks of Microsoft Corporation.

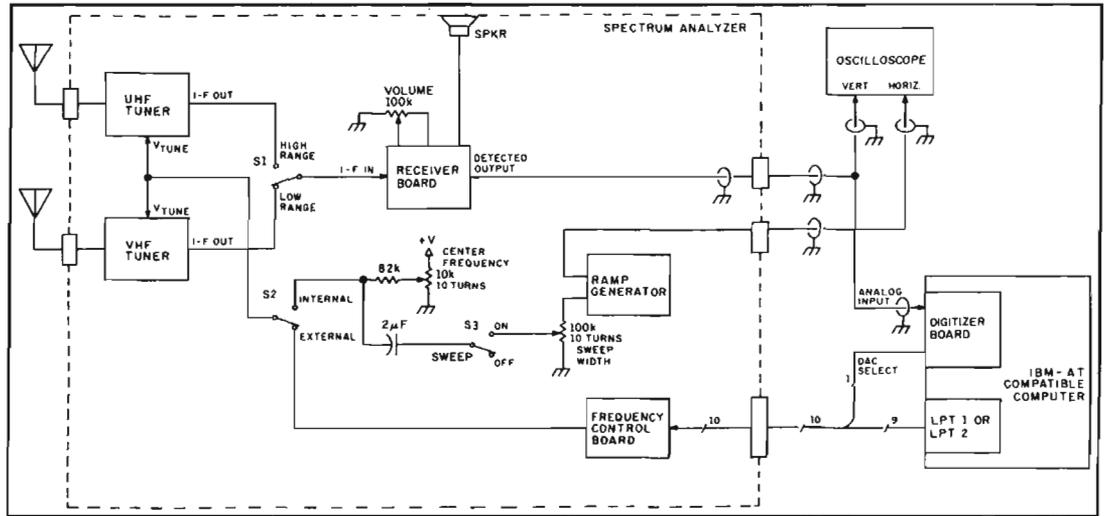


Figure 1. Spectrum analyzer.

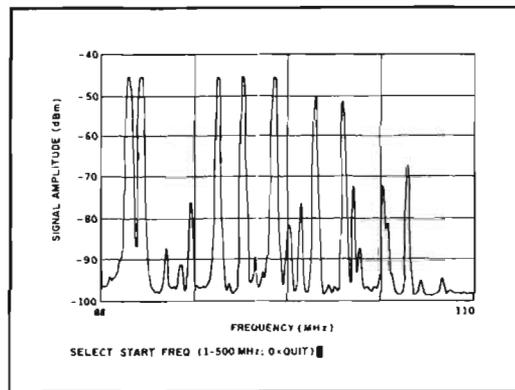


Figure 2. Spectrum plot using the VHF tuner.

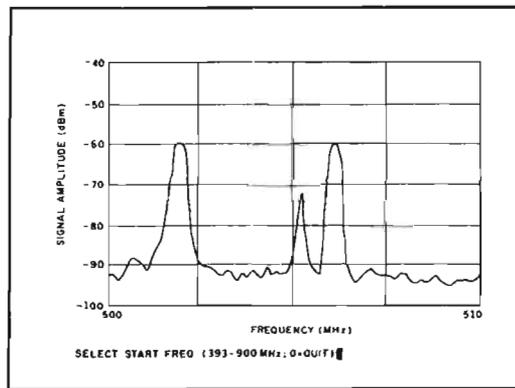


Figure 3. Spectrum plot using the UHF tuner.

Figure 2 shows a typical plot from the spectrum analyzer program using the VHF tuner. This sample shows the FM broadcast band with a part of the aircraft band as received in the Huntsville, Alabama area. Figure 3 is a typical plot using the UHF tuner. In this plot, you can clearly see the video, color, and audio carriers of television channel 19.

## Digitizer board

The schematic for the digitizer board is shown in Figure 4. Integrated circuit U3 is an 8-bit A/D converter that requires no special logic for interfacing with the computer data bus. The output latches of the A/D converter have high-impedance outputs that connect directly to the computer's data bus. The A/D converter only outputs data when addressed by the computer. Potentiometer R1 sets the reference voltage for the converter and thus determines the converter's full-scale setting. U1 and U2 are address decoders wired so an address of 22F hexadecimal (559 decimal) will start the conversion.

## DAC operation

While in operation, the computer treats the digitizer board as just another memory location. The board only digitizes the analog input when memory location 22F is addressed. U4 is used as a switch to select the appropriate DAC on the frequency control board when the computer sends frequency control data. When the computer addresses memory location 22B hexadecimal (555 decimal), the DAC-select line is a logic high. When the computer addresses memory location 22D hexadecimal (557 decimal), the DAC-select line is a logic low. More about the DAC-select line later.

## Digitizer construction

The digitizer board is mounted inside the computer in one of the expansion slots. Each expansion slot has two receptacles for the edge connectors on the expansion boards. The digitizer board requires just one edge connector and uses only the receptacle

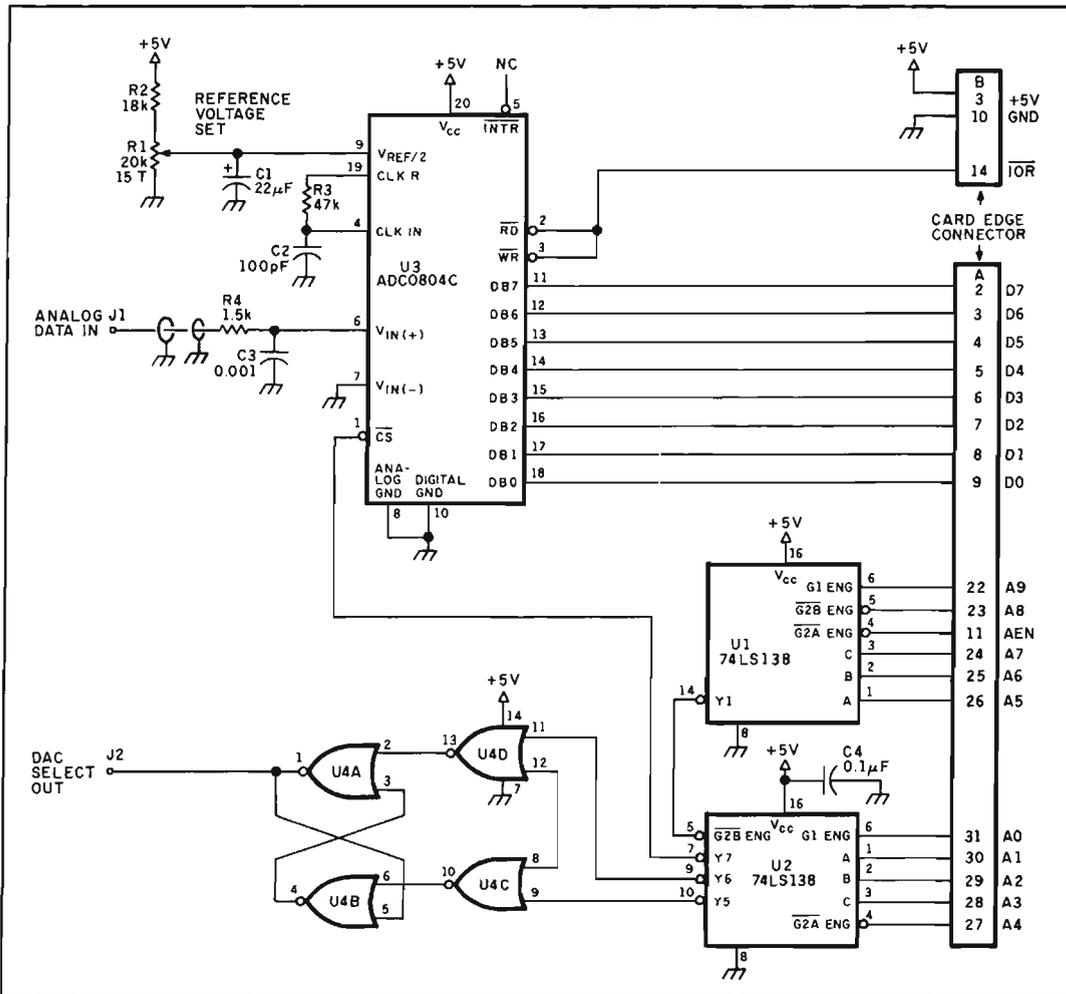


Figure 4. Schematic for the digitizer board.

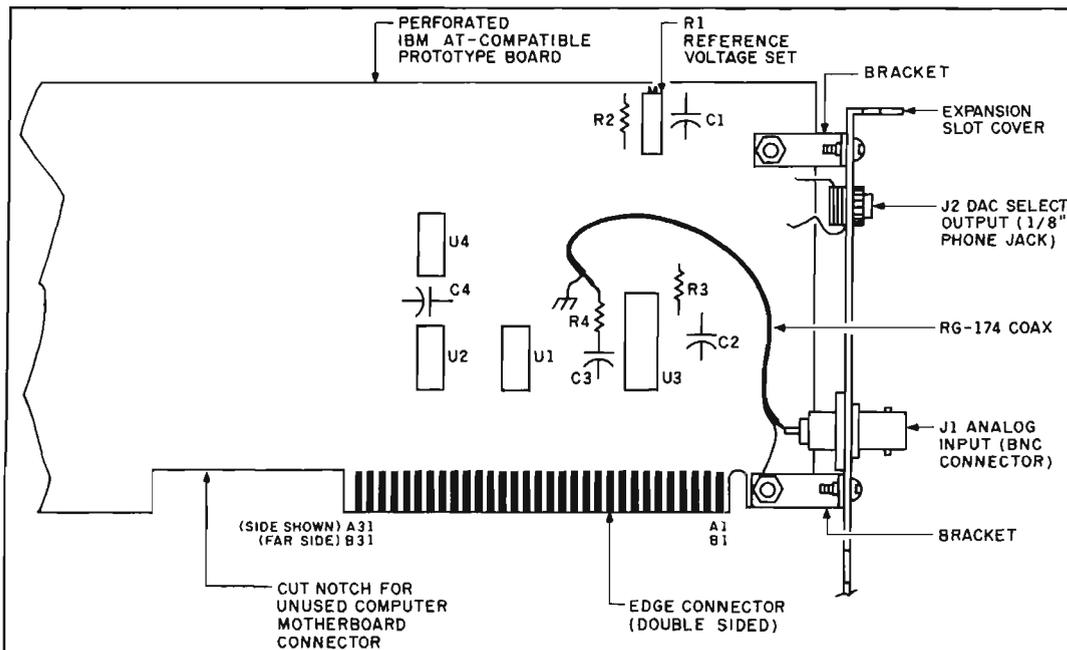


Figure 5. Layout of the digitizer board.

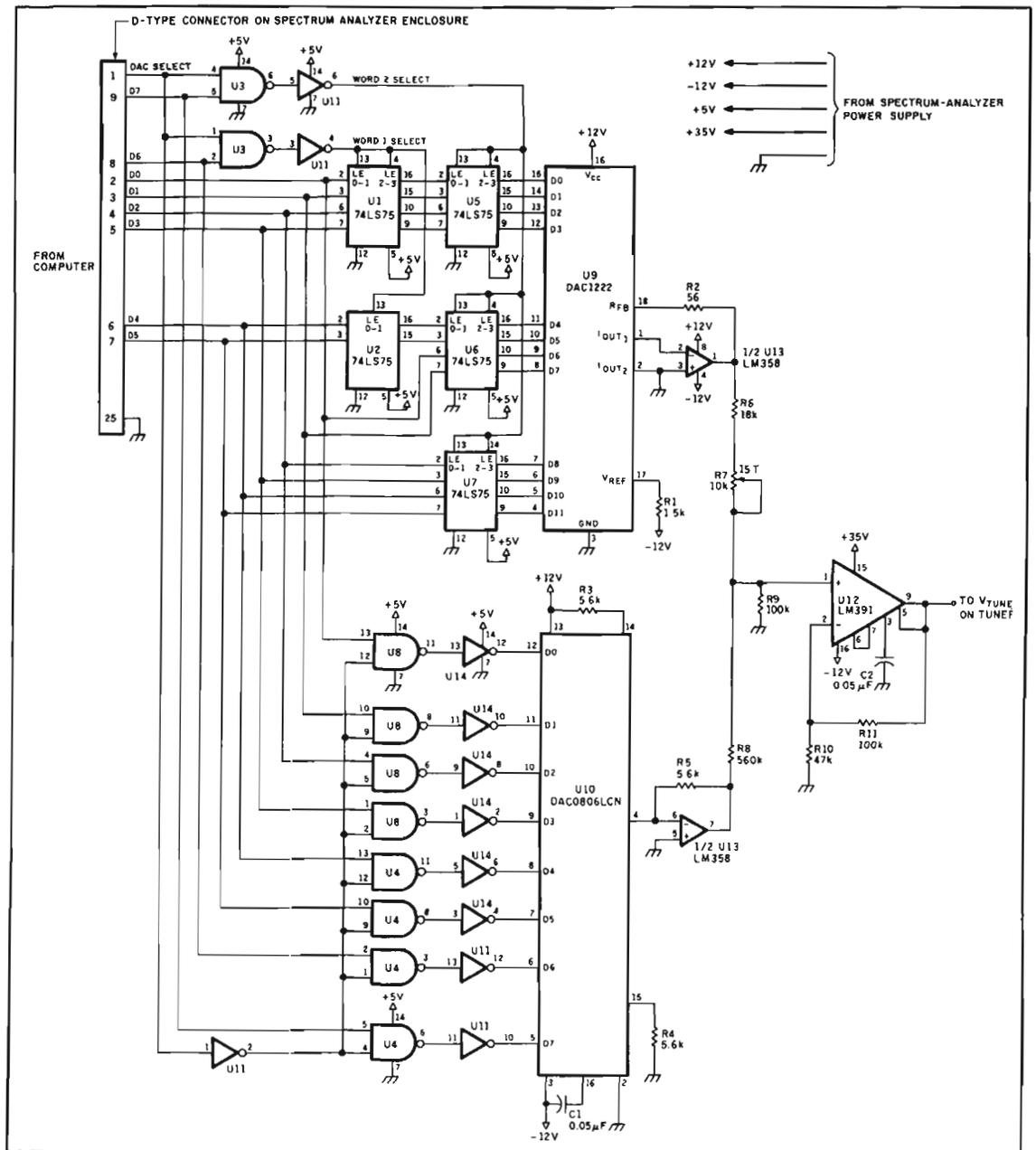


Figure 6. Schematic for the frequency control board.

at the rear of the computer.

I built the board on a perforated prototype board made especially for the computer. These boards are available from most electronics hobby stores and mail-order suppliers for about \$25. I used point-to-point wiring. All ICs are mounted in sockets to facilitate repair.

The component layout isn't critical. **Figure 5** shows the layout and edge-contractor pinouts of the digitizer board. Note that the edge connector is double sided. In the schematic, pin numbers A1 through A31 refer to edge-contractor pins on the component side of the board. Pin numbers B1

through B31 refer to edge-contractor pins on the solder side. I made a cutout in the board to clear the unused expansion slot receptacle. R1 is mounted near the top of the board. With the computer cover removed, you can adjust R1 while the board is mounted inside the case.

## Installing the digitizer board

The analog input and DAC-select line are the only connections external to the computer. I attached the slot cover from the slot in which the digitizer board is installed to the rear end of the board (see **Figure 5**).

Two small brackets stand the cover off from the board so the cover fits properly against the computer rear frame when the digitizer board is in the machine. I mounted a BNC connector on the slot cover for the analog input and a 1/8-inch phone jack on the slot cover for the DAC-select signal.

## Frequency control board

The schematic for the frequency control board is shown in **Figure 6**. The circuit uses two DACs. A 12-bit DAC decodes the start frequency; an 8-bit DAC sweeps the frequency from the start to the stop frequency.

## System logic

Both DACs are controlled through the computer's parallel-printer port. The DAC-select signal generated on the digitizer board chooses the DAC to be addressed by the printer port. When the DAC-select line is a logic high, the 12-bit DAC is selected. Conversely, when the DAC-select line is a logic low, the 8-bit DAC is selected.

The parallel-printer port is an 8-bit port. To control the 12-bit DAC, two 8-bit words — Word 1 and Word 2 — are sent from the computer. Each word contains six bits of data. The remaining two bits identify whether Word 1 or Word 2 is being sent. Bits D0 through D5 from the parallel-printer port contain the data. Bits D6 and D7 are the word identifiers. The identifier for Word 1 is a logic 0,1 for bits D6 and D7, respectively; the identifier for Word 2 is a logic 1,0 for bits D6 and D7, respectively.

The 6-bit data of Word 1 from the parallel printer port are temporarily stored in latches U1 and U2. The data are latched under command of the Word 1 latch signal generated by U3 and U11. The 6-bit data stored in U1 and U2 are then stored in latches U5 and U6. The 6-bit data of Word 2 from the parallel-printer port are stored in latches U6 and U7 under command of the Word 2 latch signal generated by U3 and U11. Thus the 12-bit data required by the 12-bit DAC are stored in latches U5, U6, and U7, following two consecutive outputs from the parallel-printer port. The 12-bit DAC (U9) will then output a current proportional to the 12-bit word present at its inputs. U13 converts U9's output current to an output voltage.

The DAC-select signal is inverted and applied to one input of NAND gates U4 and U8. When the DAC-select signal is low, these gates are activated and the 8-bit data from the parallel-printer port are applied to 8-bit DAC U10's inputs. The computer generates a series of 8-bit words and outputs

them to U10 to generate a ramp at U10's output. U13 converts U10's output current to an output voltage.

The voltage-converted outputs of the two DACs are summed by non-inverting amplifier U12. (U12 is an audio power driver intended for use in high-fidelity audio amplifier designs.) Because it can operate with high power supply voltages, I used it in this application as a high-voltage op amp.

The tuners used in the spectrum analyzer require tuning voltages ranging from zero to about 30 volts to cover their wide frequency range. U12's output can produce the required tuning voltages and drive the tuners directly. The op amp doesn't need to deliver a large current because tuning is accomplished inside the tuners with reverse-biased variable-capacitance diodes. Potentiometer R7 is used to calibrate the start frequency.

## Control board construction

I built the frequency control board on a perfboard using point-to-point wiring. All integrated circuits are mounted in sockets to facilitate repair. Once again, the board layout isn't critical. The board is mounted inside the spectrum analyzer enclosure. I added switch S2 in series with the tuning inputs of the two tuners so I could select manual (internal) or computer (external) control. See **Figure 1** for details.

## Interconnecting cable

The computer and the frequency control board in the spectrum analyzer are connected via a 10-conductor cable. I mounted a 25-pin D-type connector on the spectrum analyzer enclosure for connection to the cable. One end of the cable has a 25-pin D-type connector compatible with the one on the spectrum analyzer. The computer end of the cable has two connectors. One is a 25-pin D-type compatible with the computer's parallel-printer port. The other connector is a 1/8-inch phone plug. This plug carries the DAC-select signal and fits into jack J2 on the computer slot cover mounted on the digitizer board. The pinouts for the D-type connectors on the spectrum analyzer and computer end of the cable are shown in **Figure 7**.

## Computer program

I originally wrote the computer program in Microsoft Quick BASIC for use with an EGA monitor so I could tap into the high-resolution graphics capability. I've included line numbers on all lines of the code (Quick BASIC doesn't require line numbers for all lines) so the program can run with the more

common BASIC versions like Microsoft GW-BASIC.

The program runs much slower in GW-BASIC. This means the frequency sweep generated by the computer will be slower, and you may miss some of the intermittent signals you want to "catch" in the frequency spectrum. For the most part, the speed of the BASIC program or the computer isn't a critical parameter.

I first wrote the program for use with parallel-printer port 2 (LPT2), with the spectrum analyzer connected to LPT2 and the printer connected to parallel-printer port 1 (LPT1). Using this arrangement eliminates the need to switch connections between the printer and spectrum analyzer at the parallel-printer port when I want a hard copy of the spectrum plot. I also have a version for use with LPT1 if you have only one parallel printer port.

The computer program contains about 300 lines of code and is too long to be included with this article, but I'll send you a copy of the program. My address appears at the beginning of the article. A self-addressed stamped envelope would be appreciated. I can also supply the program on a diskette for \$5. Please state whether you want the LPT1 or LPT2 version.

A flowchart for the computer program is shown in **Figure 8**. The program begins with the establishment of four arrays. Arrays A and B contain the digitized data. The digitized spectrum analyzer data are stored immediately upon acquisition in Array A. Array B stores the digitized data after it's scaled to an equivalent dBm input to the spectrum analyzer. Arrays A and B both

contain a maximum of 256 elements, with each element corresponding to a sample of spectrum analyzer's detected output. Because an 8-bit DAC is used for the fre-

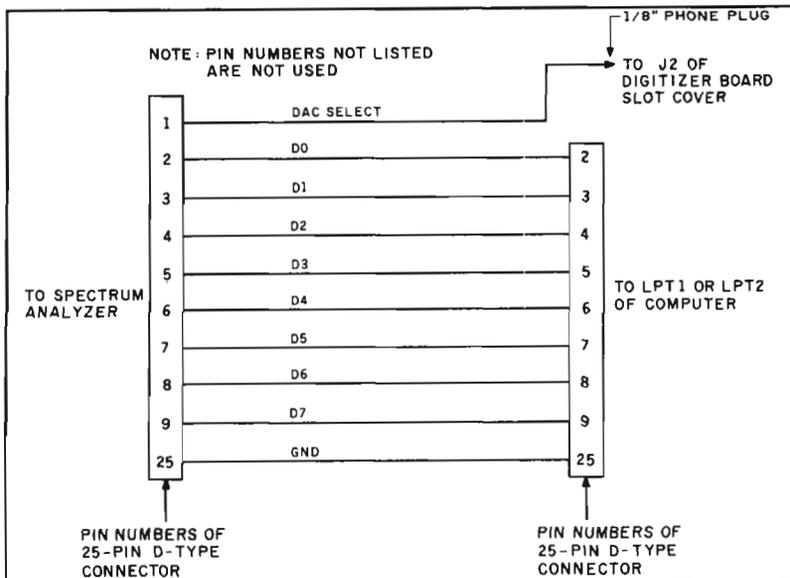


Figure 7. Interconnecting cable.

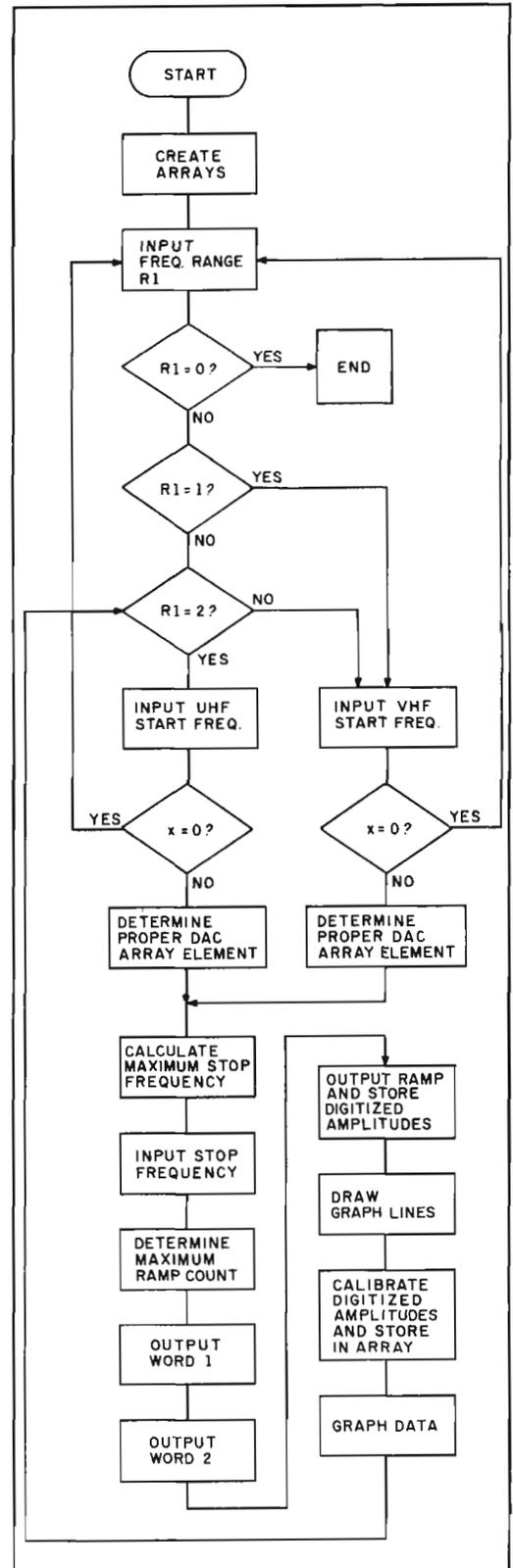


Figure 8. Flowchart for the computer program.

quency sweep, a maximum of 256 samples is possible (that is, 2 to the 8th power). Therefore, 256 array elements are required. Arrays C and D have 4098 elements each. These arrays contain the data for the two words that are sent to the 12-bit DAC for establishing the start frequency. The data for Word 1 are stored in Array C and the data for Word 2 in Array D. The program automatically creates the data for arrays C and D and stores them in the proper locations within these arrays before the program continues. The program then prompts you for the desired frequency range. There are three possibilities: 0 to quit, 1 for the VHF tuner (1 to 500 MHz), and 2 for the UHF range (393 to 900 MHz). You'll then be prompted for the desired start frequency.

After you've input the start frequency, the look-up tables in the program are used to determine the elements of Array C and Array D corresponding to the desired start frequency. The program includes two look-up tables — one for the VHF tuner and the other for the UHF tuner. The tuning curves for the two tuners aren't linear across the wide tuning ranges. **Figure 9** shows a plot of the tuning voltage versus input frequency for the VHF tuner. As you can see, the curve is very nonlinear. The UHF tuner has a similar tuning voltage versus input frequency characteristic. I've divided the tuning curve of each tuner into several small linear segments and stored equations for these segments in the look-up tables. The program selects the proper segment based on the selected start frequency, and determines the one element out of 4098 that corresponds to the frequency nearest the one selected. The element number (1 to 4098) is stored in variable f1.

Next, the maximum allowable stop frequency, based on the selected start frequency, is calculated and displayed. This frequency is the upper limit of the permissible frequency sweep, based on the start frequency selected. (Remember that the frequency sweep is generated by an 8-bit DAC.) The computer outputs a series of 8-bit words ranging from 000 to a maximum of 256 to generate a ramp which is applied to the tuners' input. A count of 256 is the maximum value available with 8 bits and thus represents the largest ramp that can be produced. This means a count of 256 will represent the maximum available frequency span.

Because the tuning curves of the tuners are nonlinear, a maximum frequency span corresponding to a count of 256 isn't constant across the tuners' frequency range. Therefore, the maximum allowable stop frequency must be calculated for each start fre-

quency selected.

The program prompts you for the desired stop frequency. After this frequency is input, the program converts it to the nearest number from 000 to 256 corresponding to that frequency. That number is stored in variable f2. The program then outputs data to the frequency control board telling it to set up the start frequency using the 12-bit DAC. The appropriate element of Array C, corresponding to Word 1, is sent first. This is followed by the appropriate element of Array D, corresponding to Word 2.

After a brief delay routine is completed, the frequency ramp output is generated, and the detected output amplitude of the spectrum analyzer is sampled by the digitizer board and stored in Array A. The detected output is sampled and the data stored for each element of the ramp. The ramp can contain a maximum of 256 elements (that is, the computer counts up to a maximum of 256), so the maximum number of digitized samples equals 256.

Next, the program creates the display screen by drawing the border and amplitude lines and labeling the axes. The digitized samples stored in Array A are then "calibrated" and converted to dBm (decibel referenced to 1 mW) using another set of look-up tables. I created the look-up tables for the VHF and UHF tuners from measurements I made of the tuners using a laboratory-grade signal generator and oscilloscope. The calibrated amplitudes are stored in Array B.

Finally, the data stored in Array B are graphed on the screen, and the program returns to the first prompt for a start frequency. At this point, you can output the graph to the printer using the "print screen" command if you wish.

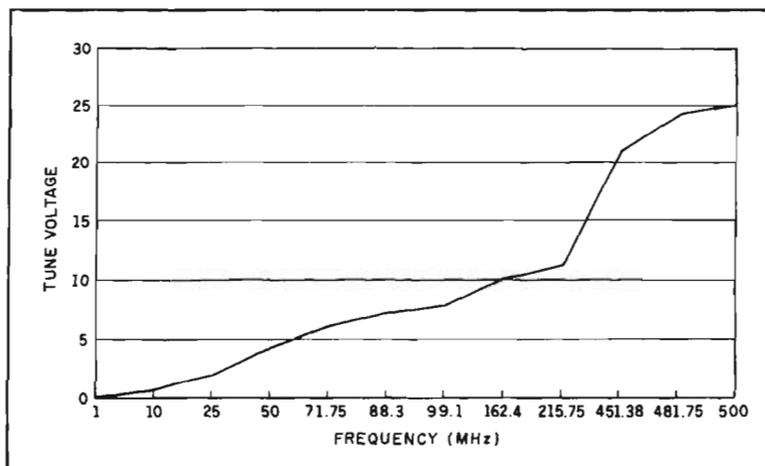


Figure 9. VHF-tuner curve.

## Installation and alignment

Once you've assembled the digitizer board, carefully inspect it for any wiring errors, shorts, or bad solder joints. Install the digitizer board in the expansion slot desired. Adjust potentiometer R1 on the digitizer board to a point approximately midway between the extremes of its range.

### Digitizer alignment

To align the digitizer board, you must apply +2.5 volts DC to its analog input. I used a 12-volt DC power supply and a potentiometer connected as a voltage divider between the power supply output and ground. The potentiometer's wiper was connected to the BNC connector on the digitizer board through a coaxial cable. An oscilloscope was used to measure the voltage at the wiper precisely. To avoid loading down the circuit, use a high-impedance device like an oscilloscope or digital voltmeter (DVM) to make this measurement. Verify that the voltage is about +2.5 volts before connecting this voltage to the computer.

### Computer interface

Turn on the computer and load the spectrum analyzer program into BASIC. Connect the test voltage to the BNC-input connector on the digitizer board and readjust the voltage for precisely +2.5 volts DC.

Run the spectrum analyzer program. The words "PLEASE WAIT" should appear on the screen. The computer is now filling Arrays C and D with the proper data for the 12-bit DAC. This takes a little time, especially if you're using GW-BASIC. The wait message appears to let you know that the computer is working.

In a few moments, the screen prompts you to select the desired frequency range. Type a 1 to choose the VHF range. The computer now prompts you for the desired start frequency. Type 100. The computer asks you for the desired stop frequency. Type 132, and a graph will appear on the screen. After a pause, a horizontal line corresponding to the input voltage level will be plotted on the graph. Below the graph, you'll again see a computer prompt for the desired start frequency.

The test voltage of +2.5 volts DC corresponds to a -60 dBm input to the VHF tuner. If the line that was plotted on the graph isn't drawn at the -60 dBm mark, readjust R1 on the digitizer board and again select 100 as the start frequency and 132 as the stop frequency. Observe the new plot. Repeat this operation until the line is plotted at the -60 dBm mark.

## Frequency-control board alignment

Remove the test voltage from the input to the digitizer board and connect the spectrum analyzer detected output to the digitizer input board. Connect the 10-conductor cable from the spectrum analyzer to the computer parallel-printer port. Next, connect the DAC-select line to the digitizer board's 1/8-inch phone jack. Configure the spectrum analyzer so the frequency control board controls the VHF tuner's frequency, and the output of the VHF tuner is connected to the receiver board. With my spectrum analyzer, all I have to do is select low range (VHF) and external control with the two toggle switches, S1 and S2, and reconnect the antenna to the VHF tuner (see **Figure 1**). Adjust R7 on the frequency-control board to the center of its range.

Perform the frequency-control board alignment using a known frequency as an input to the spectrum analyzer. Try the frequency of an FM commercial radio station. At the prompt for select start frequency, type in the frequency of a local FM station. (I used 89.3 MHz.)

At the select stop frequency prompt, type in the maximum stop frequency that the computer calculated and displayed. After a pause, you should see a spectrum plot displayed on the screen. When the plot is complete, listen to the audio output from the spectrum analyzer. The spectrum analyzer will be "resting" on the start frequency that you selected.

If your R7 setting is correct, you should hear the radio station in the speaker. If you don't hear the proper station, readjust R7 on the frequency-control board. Align the frequency-control board using several different stations as your start frequency. Adjust R7 so you can hear the majority of them to some extent.

The 12-bit DAC used for the start frequency control generates only discrete voltages. It won't be able to position the start frequency precisely on all selected start frequencies. Consequently, you won't hear all the FM stations that you use for calibration clearly. If R7 is properly adjusted, the frequencies 89.3 MHz and 96.1 MHz are easy to hear when used as start frequencies. Use these frequencies for calibration, if they are available in your area.

As an alternative, you can use a signal generator as the signal source, adjusting R7 so the spike for the generator frequency is plotted at the left of the graph. If you choose this approach, you should also try several different frequencies. Adjust R7 for

the best compromise.

That's all you have to do to align the digitizer and frequency-control boards. Now, at the start frequency prompt, type 0. The computer asks you to select the desired frequency range. Type 0 to end the program.

## Conclusions

I've found that these modifications provide an interesting versatility to WA2PZO's spectrum analyzer.<sup>4</sup> While the computer

interface doesn't make the "Poor Man's Spectrum Analyzer" equal in performance to more expensive laboratory grade equipment, it does allow some reasonably accurate measurements. ■

### References

1. Robert M. Richardson, W4UCH, "Low Cost Spectrum Analyzer with Kilobuck Features," *HAM RADIO*, September, 1986, pages 82-90.
2. Joe Carr, K4IPV, "Building the 'Poor Man's Spectrum Analyzer,'" *HAM RADIO*, March 1987, pages 99-104.
3. Joe Carr, K4IPV, "Revisiting the 'Poor Man's Spectrum Analyzer,'" *HAM RADIO*, September 1987, pages 49-52.
4. Murray Barlowe, WA2PZO, "Add a Digital Readout to the 'Poor Man's Spectrum Analyzer,'" *HAM RADIO*, September 1988, pages 84-93.

### Parts List: Digitizer Board

*C1* 22- $\mu$ F tantalum  
*C2* 100-pF disc ceramic  
*C3* 0.001- $\mu$ F disc ceramic  
*C4* 0.01- $\mu$ F disc ceramic  
(All capacitors 12 volts DC or higher)  
*J1* BNC-type connector (chassis mount)  
*J2* 1/8-inch phone jack (chassis mount)  
*R1* 20-k, 15-turn potentiometer  
*R2* 18 k  
*R3* 47 k  
*R4* 1.5 k  
(All fixed resistors 1/4-watt, 10 percent)  
*U1* ADC0804C (8-bit analog-to-digital converter)  
*U2* 74LS02 (quad 2-input NOR gate)  
*U3,U4* 74LS138 (1-of-8 decoder/demultiplexer)

### Parts List: Frequency Control Board

*C1,C2* 0.05  $\mu$ F disc ceramic  
(All capacitors 35 volts DC or higher)  
*R1* 1.5 k  
*R2* 56  
*R3,R4,R5* 5.6 k  
*R6* 18 k  
*R7* 10-k, 15-turn potentiometer  
*R8* 560 k  
*R9* 100 k  
*R10* 47 k  
*R11* 100 k  
(All fixed resistors 1/4-watt, 10 percent)  
*U1,U2,U5,U6,U7* 74LS75 (4-bit bi-stable latch)  
*U3,U4,U8* 74LS00 (quad 2-input NAND gate)  
*U9* DAC1222 (12-bit binary multiplying D/A converter)  
*U10* DAC0806LCN (8-bit D/A converter)  
*U11,U14* 74LS04 (hex inverter)  
*U12* LM391 (audio power driver)  
*U13* LM358 (low power dual op amp)