

We are in the process of building an automated component inspection system (CIS). Last time we built a PC-based capacitance meter. Now we will combine the capacitance meter with a volt/ohmmeter on a single PC board. The result will be an accurate, low cost, computer-assisted test instrument that interfaces through a standard PC parallel port.

Our CIS meter measures three basic quantities: resistance, voltage, and capacitance. Resistance may vary from 0 to 20 megohms. Voltage may vary from 0 to 2 volts DC, but simple peripheral circuitry can be added to increase range. Capacitance may vary from 20 pF to 20 μ F.

Multimeter basics

The preferred instrument for measuring resistance and voltage is a digital multimeter, or DMM. All DMM's are based on a simple, single-range voltmeter. The nice thing is that converting a humble DMM into a measurement powerhouse requires only a handful of resistors and some switches.

Intersil's 7106 is the basic building block of many meters. The 7106 contains an analog-to-digital converter (ADC) and it provides a three-digit output capable of driving an LCD directly.

A basic 7106 voltmeter that requires only three resistors and five capacitors is shown in Fig. 1. To calibrate the circuit, you must set the reference voltage at the wiper of R2 to 1.000 volt. That setting results in an overall range of 0.0 to 2.000 volts.

Increasing range

What if we want to measure a voltage greater than 2.000? To do so, we use a simple voltage divider, as shown in Fig. 2. With that simple addition, the incoming voltage is divided by the ratio of the two resistors to produce a voltage with a corresponding value. In this case, the resistance ratio is 10:1, so the maximum voltage increases by a factor of ten, giving a total range of 0-20 volts.

Measuring resistance

The 7106 compares the incoming voltage to the reference voltage between pins 35 and 36. By

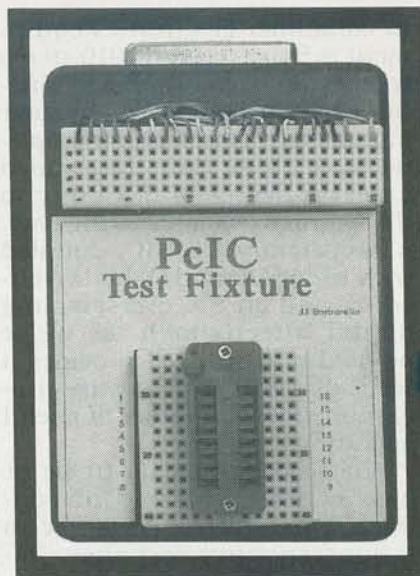
altering the circuit slightly, we can create an ohmmeter, as shown in Fig. 3. That circuit gives us another voltage divider. If the unknown resistance (R_{IN}) equals R_{REF} the same voltage will appear across both the reference and the measurement inputs, resulting in a reading of 1.000. As the value of R_{IN} varies, the reading will vary accordingly. As for range, if the value of R_{REF} were 1K, the range would be 0 to 2K. For a range of 200K, R_{REF} should have a value of 100K.

Computer interfacing

By now you're probably thinking, "That's all well and good, but how do you hook up a 7106 to a PC? The output lines only drive an LCD!" Actually, what may at first seem like an odd approach is both simple and practical. To see why, let's look at a standard LCD readout.

As shown in Fig. 4-a, the LCD consists of three seven-segment digits and a single leading "1," making a total of 22 segments. Figure 4-b shows which segments are used to display each integer from zero to nine; Fig. 4-c shows the same values, but without the use of segments c and d. The 4-c versions aren't pretty, but each one is unique, so a computer could read the 7106 segment outputs and translate the values into a more comprehensible form. By ignoring two segments from each digit, sixteen lines are required: five for each digit plus one for the leading digit. Reducing the number of lines in that way is significant because it allows us to use a simple scheme to read each display segment into the computer and then determine what the composite reading was.

A pair of 4051 analog multiplexers does the trick. The 4051 connects one of eight inputs to a single output according to the binary value at three control inputs (A, B, and C). Using two 4051's with common control lines, we can monitor the status of sixteen inputs using only three output and two input lines from the computer. As shown in Table 1, when A, B, and C are all low, input 0 is connected to the output; when A, B, and C are all high, input 7 is connected to the output.



In our circuit, we connect the sixteen signal inputs of the two 4051's to the segment-drive outputs of the 7106, and use the computer to control the A, B, and C control inputs. We also use the computer to monitor the pin-3 outputs of the 4051's.

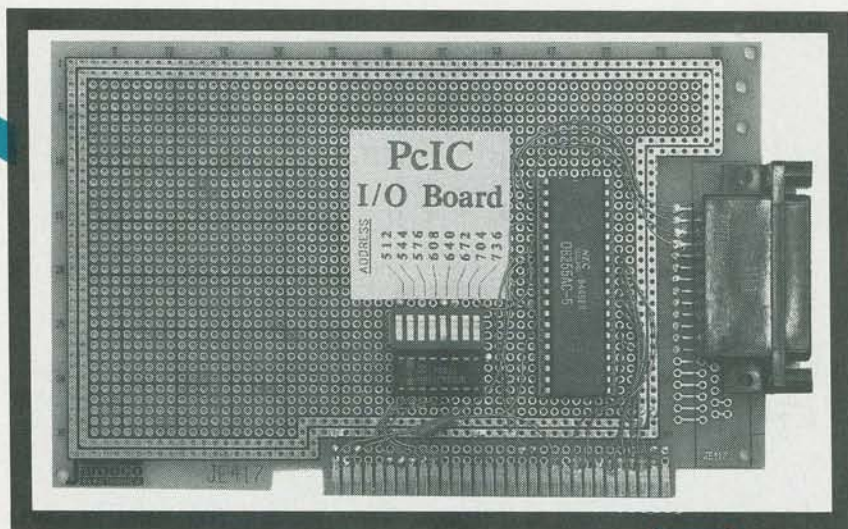
The circuit

The complete circuit is shown in Fig. 5. The circuit is slightly different from the basic circuit in that the reference and voltage inputs are routed to a three-pole, double-throw switch (S1), which selects ohms or volts. In the "volts" position, the CALIBRATE potentiometer (R10) is connected to the reference inputs of the 7106. In the "ohms" position, the reference inputs are connected to one of several reference resistors, as selected by analog-switch IC7.

A 4030 EXCLUSIVE-OR gate (IC2) converts the segment-drive signals to DC levels. The 7106 provides segment drive and BP (backplane) signals, both of which are square waves. When a segment is to be illuminated, the phase of the segment output is shifted 180 degrees with respect to BP. By sending the BP and segment signals through an EXCLUSIVE-OR gate, a steady low is obtained when a segment is off, and a high when it is on. That steady-state level can then be read by the PC through the PE and SLCT inputs.

A normally-open pushbutton switch (S2) allows a diode or

EXPERIMENTING WITH PC-BASED TEST EQUIPMENT



This time we build a low-cost PC-based voltage, resistance, and capacitance meter.

JAMES J. BARBARELLO

transistor junction to be forward-biased, thereby allowing measurement of the voltage drop across the junction. We incorporated the capacitance meter (discussed in the May 1991 issue of **Radio-Electronics**) in IC4 and IC6.

In the schematic, note that there are two ground circuits. The reason is that the 7106 requires a supply greater than 6.5 volts, but the PC needs standard TTL levels. In addition, the meter inputs are referenced to the 7106's INLO input, which differs from the digital ground. We use a 9-volt DC wall transformer to power the 7106 and associated IC's, and Zener diode D1 to generate the required TTL levels for the PC interface.

Construction

We recommend PC-board construction for this project. Foil patterns are provided if you wish to make your own board.

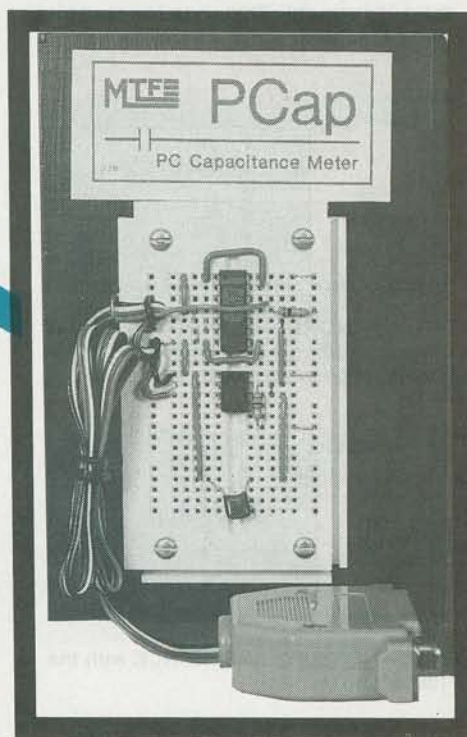
Begin construction by installing the 12 jumpers, as shown in Fig. 6. Next, install sockets for all seven IC's. Except for IC4, all IC's are static-sensitive—especially

IC1—so use standard precautions when handling those devices. Install the remaining components.

There are 13 connections to P1, two connections for power, and 11 other connections. Prepare 24 lengths of wire, stripping $\frac{1}{8}$ " of insulation from both ends of each wire. Connect the designated wires to P1 and S1.

If there is a plug on the lead from the power cube, remove it and strip $\frac{1}{8}$ " of insulation from the two leads. Using a voltmeter, determine which is positive and which is negative, and attach each to the appropriate pad on the PC board. Don't mix them up!

You can use whatever you like for input terminals. The author found it convenient to use a solderless breadboard strip with four rows of connection points. He ran leads from the \pm resistance/voltage inputs to one pair of rows, and from the \pm capacitor inputs to the other. That scheme allows quick insertion and removal of test components.



A separate set of terminals provides a quick connection for diode/transistor testing.

Check your work carefully, making sure all semiconductors and the power leads were installed correctly, and that there are no solder bridges or opens on the PC board. If you have an oscilloscope handy, apply power and check for a square wave between pins 38 and 40 of IC1; the frequency should be between 40

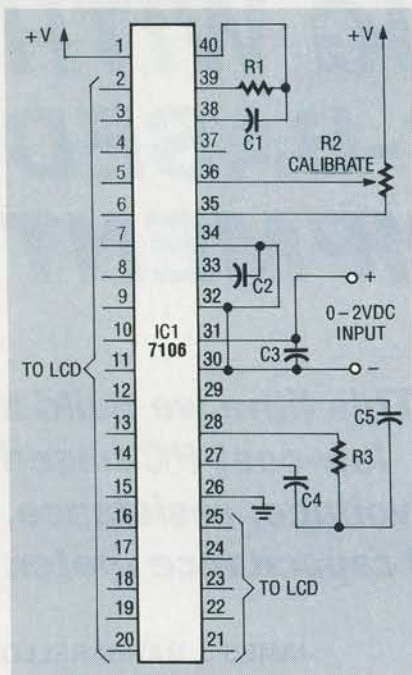


FIG. 1—A BASIC DMM can be built around Intersil's versatile 7106 and a few resistors and capacitors.

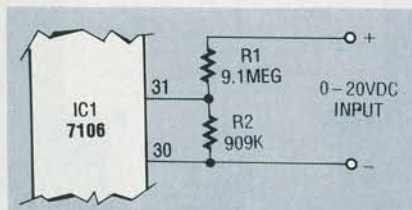


FIG. 2—INCREASE INPUT-VOLTAGE range by adding a voltage divider. In this circuit, range increases by a factor of ten.

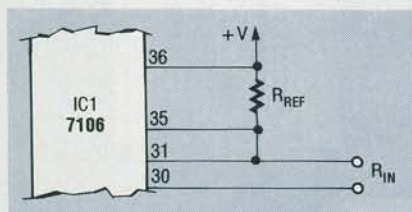


FIG. 3—MEASURE RESISTANCE with the 7106 as shown here.

and 50 kHz. That measurement is not strictly necessary, but making it confirms that the circuit is operating.

Software

The software consists of several independent QuickBASIC programs for calibrating the device and making different measurements. Unfortunately, there is not enough space to publish all of the programs. However, the software is available from the RE-BBS (file: PCTEST2.ZIP, 516-293-2283, 1200/2400, 8N1)

TABLE 1—4051 DECODING

A	B	C	"0"	"1"	"2"	"3"	"4"	"5"	"6"	"7"
0	0	0	X							
0	0	1		X						
0	1	0			X					
0	1	1				X				
1	0	0					X			
1	0	1						X		
1	1	0							X	
1	1	1								X

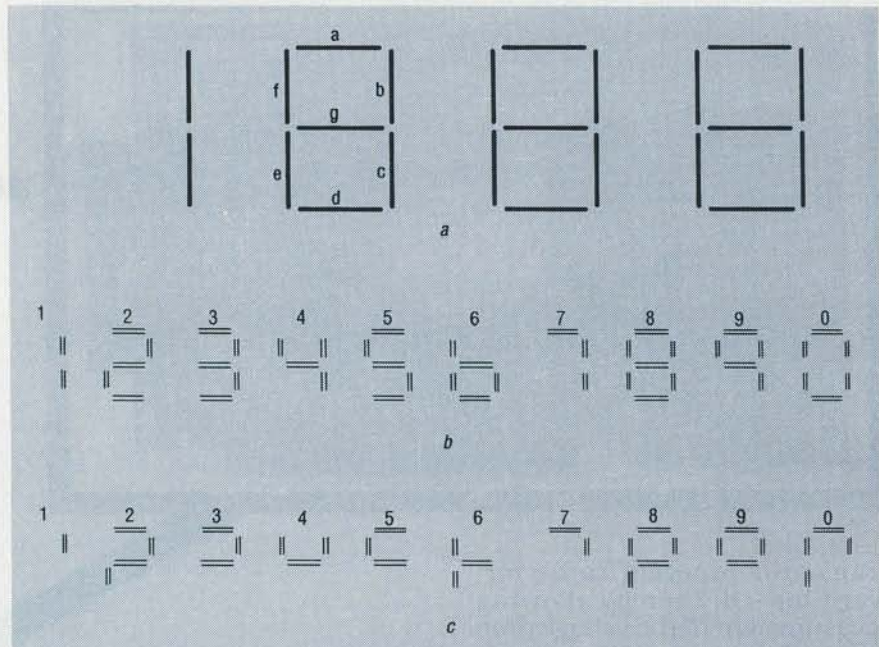


FIG. 4—THE 7106 DRIVES A 3½-DIGIT LCD DISPLAY (a). Each integer from 0-9 is formed from a unique combination of segments (b); by ignoring segments c and d, each digit remains unique, but fewer lines allow a cleaner PC interface.

PARTS LIST

All resistors are ¼-watt, 5%, unless otherwise noted.

- R1, R4, R7—100,000 ohms
- R2—470,000 ohms
- R3, R9—10 megohms
- R5, R11, R12—1000 ohms
- R6—1 megohm
- R8—10,000 ohms
- R10—100,000 ohms

Capacitors

- C1—100 pF, 5%
- C2, C3—0.1 µF, 10%
- C4—0.047 µF, 10%
- C5—0.47 µF, 10%

Semiconductors

- IC1—7106CPL 3½-digit LCD A/D converter
- IC2—CD4030 quad XOR gate
- IC3, IC5—CD4051 8-channel multiplexer
- IC4—555 timer
- IC6, IC7—CD4066 quad bilateral switch

D1—1N4734 5.6-volt Zener diode

Other components

- P1—DB25 male connector
- S1—3-pole, double-throw slide switch
- S2—normally-open SPST pushbutton switch

Miscellaneous: 9-volt DC, ½-amp wall transformer; case; solderless breadboard strip; PC board.

Note: A compiled version of the software that also contains data-logging capability is available on 5¼" double-density PC diskette for \$12.00 from JJ Barbaro, RD#3, Box 241H, Tennent Road, Manalapan, NJ 07726. The author will be happy to answer any questions. Please include a self-addressed stamped envelope for reply.

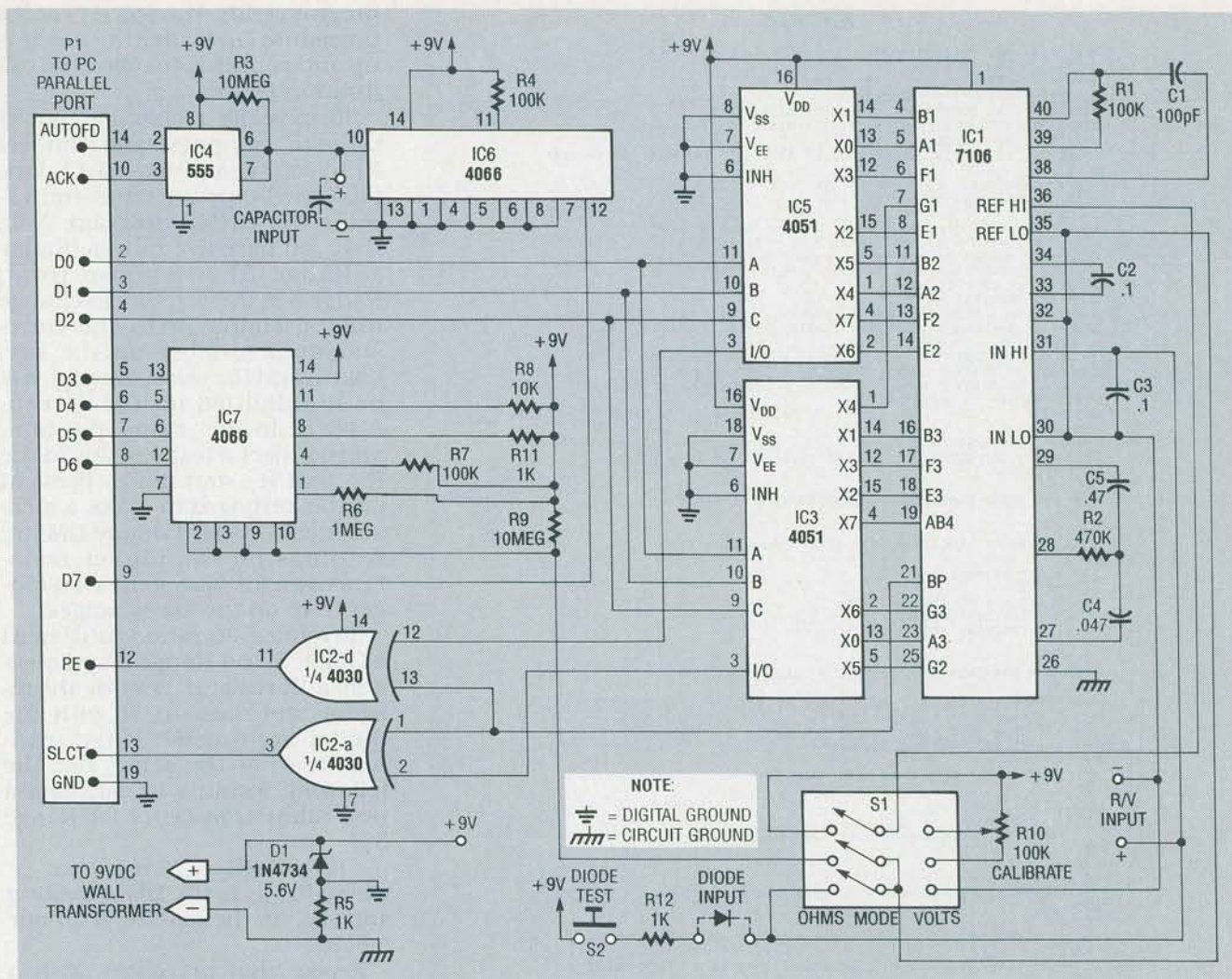


FIG. 5—COMPLETE SCHEMATIC. Data lines D0-D3 select one of eight segment-output lines through the 4051's (IC5, IC6). Range selection is accomplished by data lines D3-D6, which select one of four resistors through analog-switch IC7. The selected resistor, if any, is paralleled with R9, thereby decreasing effective resistance.

and from the author, as mentioned in the parts list. Listing 1 shows the resistance-measurement program.

Calibration and use

Both the hardware and the software must be calibrated. Begin hardware calibration by connecting a DMM across the wiper and the lower end of R10, and set S1 to the "volts" position. Apply power and adjust R10 until the DMM reads 1.000 volt.

Begin software calibration by executing the OHMFAX program. The screen will initially show null calibration factors for the five resistance ranges, R1-R5. The program asks whether you want to revise those values; press Y and enter the following initial values: R1, 10000; R2, 1000; R3, 100; R4, 10; R5, 1. The Revise question will appear again; press N this time. Then the program asks whether to save the values you just entered; press Y. Now the

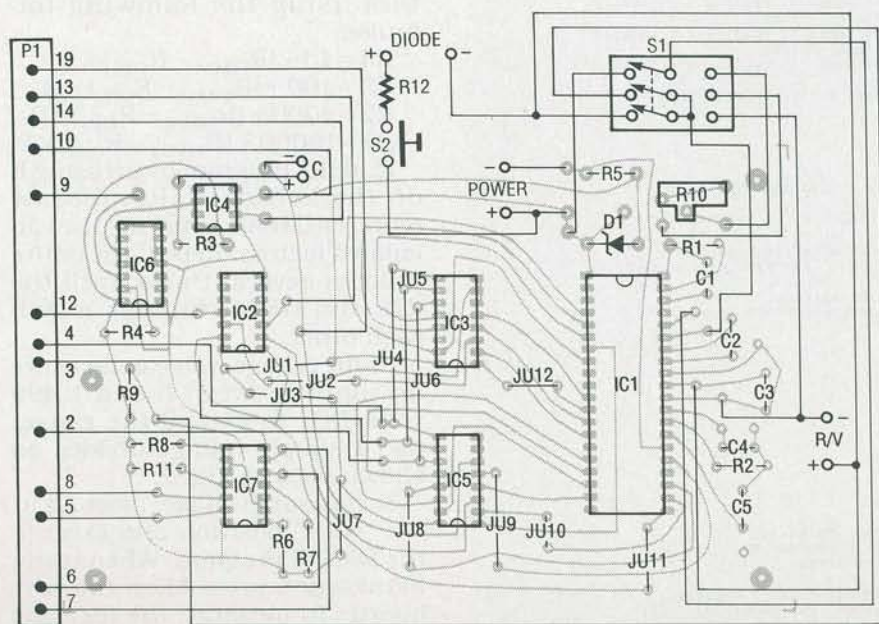


FIG. 6—MOUNT ALL COMPONENTS as shown here, installing the twelve jumpers first.

LISTING 1

```

1
2
3 REM*****
4 REM** OHMS.BAS *
5 REM** V900518 *
6 REM*****
7 DEF SEG = 64: DEFINT A-C, I-K: DEFDBL R, V, X
8 DIM a(9), j(7), k(7), r(5), r$(5), m(5), f$(6), ft$(6)
9 OPEN "M", 1, "ohmfax.dat", 50
10 FIELD 1, 10 AS r1$, 10 AS r2$, 10 AS r3$, 10 AS r4$, 10 AS r5$
11 GET 1, 1
12 r(1) = VAL(r1$): r(2) = VAL(r2$): r(3) = VAL(r3$)
13 r(4) = VAL(r4$): r(5) = VAL(r5$)
14 CLOSE
15 a(0) = 15: a(1) = 2: a(2) = 23: a(3) = 19: a(4) = 26
16 a(5) = 25: a(6) = 29: a(7) = 3: a(8) = 31: a(9) = 27
17 m(1) = 0: m(2) = 8: m(3) = 16: m(4) = 32: m(5) = 64
18 r$(1) = "20Meg": r$(2) = "2 Meg": r$(3) = "200 K"
19 r$(4) = "20 K": r$(5) = "2 K"
20 comma$ = CHR$(34) + ","
21 GOSUB S1: C = PEEK(8) + 256 * (PEEK(9)) + 2: B = C - 1: a = B - 1
22 imsk = 5: imskold = imsk: readingold = -999: OUT a, m(imsk)
23 LOCATE 20, 4
24 COLOR 0, 7: PRINT " M "; : COLOR 7, 0: PRINT "easure, ";
25 COLOR 0, 7: PRINT " D "; : COLOR 7, 0: PRINT "atalog, ";
26 COLOR 0, 7: PRINT " E "; : COLOR 7, 0: PRINT "nd, or ";
27 PRINT "Range: "; LOCATE 21, 43
28 PRINT " 1 2 3 4 "; : COLOR 0, 7: PRINT " 5 ";
29 : COLOR 7, 0
30 FOR i = 1 TO 5: LOCATE 20, 43 + (i - 1) * 7: PRINT r$(i): NEXT
31 LOCATE 20, 71: COLOR 0, 7: PRINT r$(5): COLOR 7, 0
32 GOTO tryagain:
33 REM
34 REM*** SETUP COMPLETED - MEASUREMENT LOOP BEGINS
35 REM
36 begin:
37 OUT a, m(imsk): LOCATE 9, 35: PRINT "Measuring... "
38 FOR i = 0 TO 7
39 ii = i + m(imsk)
40 OUT a, ii
41 FOR rr = 1 TO 150: NEXT rr
42 j(i) = INP(889) AND 16: j(i) = j(i) / 16
43 k(i) = INP(889) AND 32: k(i) = k(i) / 32
44 NEXT
45 REM
46 REM*** MEASUREMENT LOOP ENDS - CALCULATE VALUE
47 REM
48 ones = k(0) + k(1) * 2 + k(2) * 4 + k(3) * 8 + j(4) * 16
49 tens = k(4) + k(5) * 2 + k(6) * 4 + k(7) * 8 + j(5) * 16
50 huns = j(0) + j(1) * 2 + j(2) * 4 + j(3) * 8 + j(6) * 16
51 thos = j(7): IF huns = 0 THEN huns = 99
52 FOR i = 0 TO 9
53 IF ones = a(i) THEN ones = i: GOTO JUMP1
54 NEXT
55 JUMP1:
56 FOR i = 0 TO 9
57 IF tens = a(i) THEN tens = i: GOTO JUMP2
58 NEXT
59 JUMP2:
60 FOR i = 0 TO 9
61 IF huns = a(i) THEN huns = i: GOTO JUMP3
62 NEXT
63 JUMP3:
64 reading = thos * 1000 + huns * 100 + tens * 10 + ones
65 REM
66 REM*** VALUE CALCULATED - DETERMINE IF STEADY STATE READING
67 REM
68 IF reading <> readingold THEN readingold = reading: GOTO begin
69 REM
70 REM*** STEADY STATE READING - DISPLAY
71 REM
72 LOCATE 9, 35
73 IF imsk = 1 THEN PRINT USING "##.## Mohms": reading * r(imsk) /
74 1000000: measuredvalue = reading * r(imsk) / 1000000
75 IF imsk = 2 THEN PRINT USING "###.## Kohms ": reading * r(imsk) /
76 1000: measuredvalue = reading * r(imsk) / 1000
77 IF imsk = 3 THEN PRINT USING "###.## Kohms ": reading * r(imsk) /
78 1000: measuredvalue = reading * r(imsk) / 1000
79 IF imsk = 4 THEN PRINT USING "##.## Kohms ": reading * r(imsk) /
80 1000: measuredvalue = reading * r(imsk) / 1000
81 IF imsk = 5 THEN PRINT USING "###.## Ohms ": reading * r(imsk):
82 measuredvalue = reading * r(imsk)
83 IF huns = 99 AND thos = 1 THEN LOCATE 9, 35: PRINT "OVERRANGE 1"
84
85 BEEP
86 LOCATE 12, 37
87 tryagain:
88 reply$ = UCASE$(INPUT$(1))
89 IF VAL(reply$) < 1 OR VAL(reply$) > 5 THEN GOTO tryagain2
90 imsk = VAL(reply$)
91 LOCATE 21, 43 + (imskold - 1) * 7: COLOR 7, 0: PRINT
92 STR$(imskold) + " ";
93 LOCATE 20, 43 + (imskold - 1) * 7: PRINT r$(imskold)
94 LOCATE 21, 43 + (imsk - 1) * 7: COLOR 0, 7: PRINT " "; reply$ + "
95 ";
96 LOCATE 20, 43 + (imsk - 1) * 7: PRINT r$(imsk)
97 COLOR 7, 0: imskold = imsk: OUT a, m(imsk): GOTO tryagain
98 tryagain2:
99 SELECT CASE reply$
100 CASE "M"
101 GOTO begin
102 CASE "E"
103 CLOSE: END
104 CASE ELSE
105 SOUND 100, 4
106 GOTO tryagain
107 END SELECT
108 GOTO begin
109 END
110 S1:
111 CIS: LOCATE 1, 23: COLOR 0, 7: PRINT SPACES(35)
112 LOCATE 2, 23: PRINT " MANUFACTURING TECHNOLOGY FACILITY "
113 LOCATE 3, 23: PRINT " PC OHM METER "
114 LOCATE 4, 23: PRINT SPACES(35): COLOR 7, 0
115 LOCATE 8, 30: PRINT CHR$(222); SPRINGS(19, 223); CHR$(221)
116 LOCATE 9, 30: PRINT CHR$(222); LOCATE 9, 50: PRINT CHR$(221)
117 LOCATE 10, 30: PRINT CHR$(222); STRING$(19, 220); CHR$(221)
118 RETURN

```

program ends. You will repeat the procedure later, after the meter is up and running, to refine the calibration factors.

To measure resistance, power up your computer and connect P1 to your computer's LPT1 port. Plug in the power cube and execute the OHMS program. Note that you have five ranges (Range 1–Range 5) to choose from; Range 5 is the default. To change ranges, simply press the corresponding number on the keyboard, and the selected range will be highlighted on the screen. Place S1 in the “ohms” position, and connect a test resistor to the R+ and R– terminals. Press M on the keyboard to make a measurement. As with many DMM's, readings on the higher resistance ranges take somewhat longer than on the lower ranges.

To refine calibration you'll need a DMM. Select Range 5 and measure a 1K resistor. Remove the resistor and measure it with the digital multimeter. If the readings are not the same, use the following formula to calculate a new calibration factor for Range 5:

$$R5 = 1 + (R_{DMM} - R_{CIS}) \times 2$$

where R_{DMM} is the DMM reading and R_{CIS} is the reading with our meter.

Repeat that procedure with a 10K resistor for Range 4, a 100K resistor for Range 3, a 1 megohm resistor for Range 2, and a 10-megohm resistor for Range 1. Calculate approximate new factors using the following formulas:

$$R4 = 10 + (R_{DMM} - R_{CIS}) \times 2$$

$$R3 = 100 + (R_{DMM} - R_{CIS}) \times 2$$

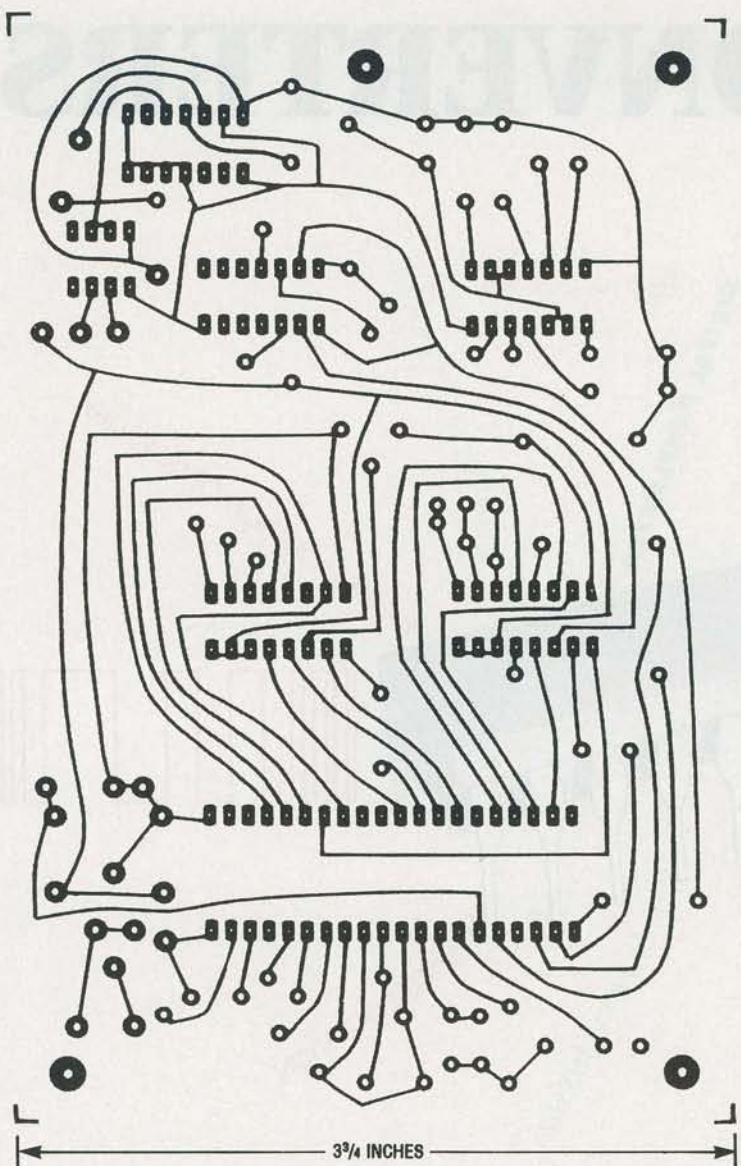
$$R2 = 1000 + (R_{DMM} - R_{CIS}) \times 2$$

$$R1 = 10000 + (R_{DMM} - R_{CIS}) \times 2$$

End the program by pressing E on the keyboard. Recalibrate using OHMFAX, entering the calculated factors. Repeat the entire process several times until the CIS and DMM readings match each other.

As for voltage, you can measure an input between 0 and 1.999 volts DC. To extend the range, use a simple voltage divider, as shown back in Fig. 2.

To measure voltage, place S1 in the “volts” position and execute the DIODE program. When ready to measure, press M on the keyboard. To measure the forward voltage drop across a diode or the



FOIL PATTERN shown actual size.

base-emitter junction of a transistor, insert the device into the appropriate terminals and press S2; doing so forward biases the junction and thereby allows you to measure the voltage drop across it.

Capacitance measurement works as described in May. Place the capacitor to be measured in the C+ and C- terminals; the position of S1 is irrelevant. Execute the CAP program. When ready to measure, press M on the keyboard. Note: The capacitance meter described last time used parallel port line D6 to switch the low range resistor in, but the current version uses line D7. Hence there is a minor change to the CAP program. The line imme-

diately below the RESTART: label must be changed from OUT A,64 to OUT A,128.

Conclusion

This project is a versatile computer-assisted test instrument. You can increase its versatility by modifying the software. For example, you could include data logging or the ability to specify pass/fail criteria for different components. In that way data could be analyzed to determine the quality of components supplied by different vendors. In the final installment of this three-part series, we'll build an IC tester to round out our computer-controlled component inspection system.