

# TEST DIGITAL CIRCUITS WITH THE SMARTPROBE

*Replace your old logic probe with one that comes with a cap and gown—and the ability to resolve voltages to four levels as well as remember them!*

While a logic probe is an inexpensive piece of test gear that can be very handy when working with digital circuits, there are both positive and negative aspects to it. On the positive side, it's easy to use and gives you a quick view of what's going on in digital circuits. On the negative side, you get only three LEDs that tell you if the point being probed is low, high, or is pulsing between low and high. Of course, if all of the LEDs are off, the reading might be "floating" between the voltage thresholds for zero and one. That assumes that the instrument is working properly and that you've set it up correctly—like making sure not to accidentally switch the sensitivity to "CMOS" while testing a TTL-based circuit!

It's probably fair to characterize most basic logic probes as "dumb" instruments. Wouldn't it be nice to have a "smart" probe that gives you both visual and audible indications? How about adding the ability to display the status of the last 20 or so readings? While we're at it, let's add the ability to view the relative frequency and pulse width of signals on an oscilloscope-type display? Of course, it would be nice if you got all of those additional capabilities for about the same cost as that standard "dumb" logic probe.

The SmartProbe device presented here is such a device. Using an inexpensive LM339 comparator IC and a handful of other components, it connects to the parallel port of a PC. Like a standard logic probe, it senses the logic 0 and logic 1 "windows" of both TTL and CMOS devices, providing visual

JAMES J. BARBARELLO

feedback of the detected logic state. The software lets your PC select a different audible tone for TTL and CMOS levels; keep a history of the last 20 readings; and show amplitude, frequency, and pulse-width characteristics on an oscilloscope-type display. Since it is a PC-based device, you can modify it to control other devices based on the sensed levels. It will work with any PC, from an old XT to tomorrow's newest model. The SmartProbe uses readily available components and doesn't require any special construction techniques. Even without any "junk-box" parts, the total cost is under \$15.

**How It Works.** The schematic diagram for the SmartProbe is shown in Fig. 1; refer to it during the following discussion. The heart of the circuit is IC1, an LM339 quad comparator. That chip has four identical comparators in it. When the voltage to the positive input is greater than the voltage to the negative input of any particular comparator, its output goes high. When it is lower, the output goes low. Although the

outputs of an LM339 do not actively produce a voltage to indicate a high output, we'll assume at this point in the description that they can do that; the outputs of IC1 will be discussed in greater detail later in this section.

The SmartProbe's tip is connected to one of the inputs on each comparator. Note that two of the comparators ("b" and "c") have the probe voltage going to their positive input, while the other two have the probe voltage going to their negative input. The other comparator inputs are connected to a voltage divider formed by R5-R9; that divider provides a reference level for each comparator. The resistor values chosen set the logic-low and logic-high voltage levels for both TTL and CMOS devices. With a 5-volt source, the reference voltage for comparator "a" is about



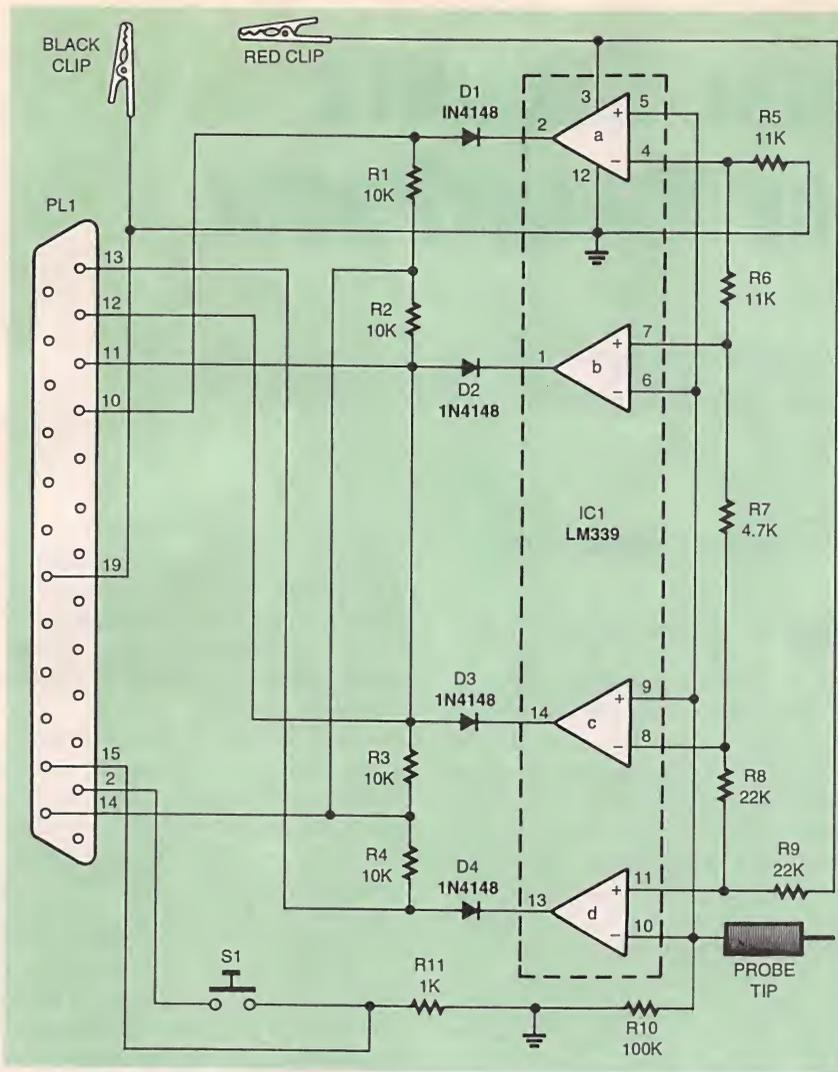


Fig. 1. The SmartProbe is built around a set of four comparators that can sense four different voltage levels. Those levels are set to the low and high logic levels of TTL and CMOS devices. Using an LM339 as well as a set of diodes lets the SmartProbe work with higher voltages without damaging the TTL circuitry in a PC's parallel port.

0.8 volts (TTL low) and about 2 volts for comparator "c" (TTL high). Since CMOS uses a percentage of the supply voltage to determine the logic-low and logic-high thresholds, comparator "b" is referenced to 30% of the supply voltage while comparator "d" is set to 70% of the supply voltage.

To better understand the action of the comparators, we'll apply a voltage to the probe that is between ground and 15% of the power supply. Since that level is lower than the reference voltage at R5/R6, IC1-a's output will go low. The same thing will happen with IC1-c since the input level is also lower than the reference voltage at R7/R8. The opposite occurs with IC1-b and

IC1-d because the probe voltage is being applied to the inverting inputs. Table 1 shows the resulting outputs on IC1 for the various voltage thresholds.

Note that with the probe unconnected, R10 provides a ground reference, effectively keeping all of the comparator inputs at ground, avoiding any spurious responses.

A trigger switch, S1, is connected to two pins of PL1. Pin 2 supplies volt-

TABLE 1—COMPARATOR RESPONSES

Probe Voltage	"a"	"b"	"c"	"d"
0%–15%	0	1	0	1
15%–30%	1	1	0	1
30%–40%	1	0	0	1
40%–70%	1	0	1	1
70%–100%	1	0	1	0

age to S1 (which will be set through the software) and pin 15 reads the current state of S1 (closed or open). When S1 is opened, R11 keeps pin 15 at ground level.

Using an LM339 for IC1 is the secret to powering the SmartProbe with voltages (as high as 18 volts for CMOS) that would otherwise destroy the circuitry in the computer's parallel port. Although the output of IC1 does not supply an active voltage when high (only grounding the output when low), blocking diodes D1–D4 and their associated pull-up resistors, R1–R4, give the open-collector outputs of IC1 additional protection in case something goes wrong with IC1.

### Reading SmartProbe Information.

First, let's review how a parallel port on a PC works. We'll assume that we're using LPT1; the decimal address of that port is 888. That port address is for outputting to the data lines of the printer. We want to read the five input pins that normally send feedback from the printer to the computer. That address is the base address + 1, or 889 in our example. If we want to look at a particular pin, we simply AND the read value with a "masking" value that strips away the data that we're not interested in. For example, if we want to look at pin 10, we would AND the reading from input port 889 with the value 64. If the result is 64, pin 10 is high; if it is zero, then the pin is low. The state of the other pins are forced to zero by the AND operation.

An oddity to remember is that pin 11 inverts its signal. Our software has to consider that in order to read data from the SmartProbe correctly.

Table 2 is simply Table 1 with the logic levels replaced with the values that our software would see when the input port is read. An additional column has been included: a sum of the read values. For example, if we read port 889, AND that data with 240 (to mask our four input pins), and find that the result is 208, we know that the SmartProbe is reading a voltage that is between 30% and 40% of its power supply.

## PARTS LIST FOR THE SMARTPROBE

### SEMICONDUCTORS

IC1—LM339 quad comparator, integrated circuit  
D1—D4—1N4148 silicon diode

### RESISTORS

(All resistors are 1/4-watt, 5% units unless otherwise noted.)

R1—R4—10,000-ohm  
R5, R6—11,000-ohm, 1/4-watt, 1%  
R7—4700-ohm  
R8, R9—22,000-ohm  
R10—100,000-ohm  
R11—1000-ohm

### ADDITIONAL PARTS AND MATERIALS

PL1—DB-25 male connector  
S1—Single-pole, single-throw, momentary-contact switch, normally open  
DB-25 non-shielded hood, 9-conductor cable, black and red mini-alligator clips, wire, hardware, etc.

**Note:** The following item is available from James J. Barbarello, 817 Tennent Road, Manalapan, NJ 07726: A 3½-inch floppy disk containing the SmartProbe software plus additional material (both source and executable code), Part No. SP-S, \$12.00. Check or money order must be payable in US funds. NJ residents must add appropriate sales tax. The author will be happy to answer any questions that are accompanied by a stamped, self-addressed envelope.

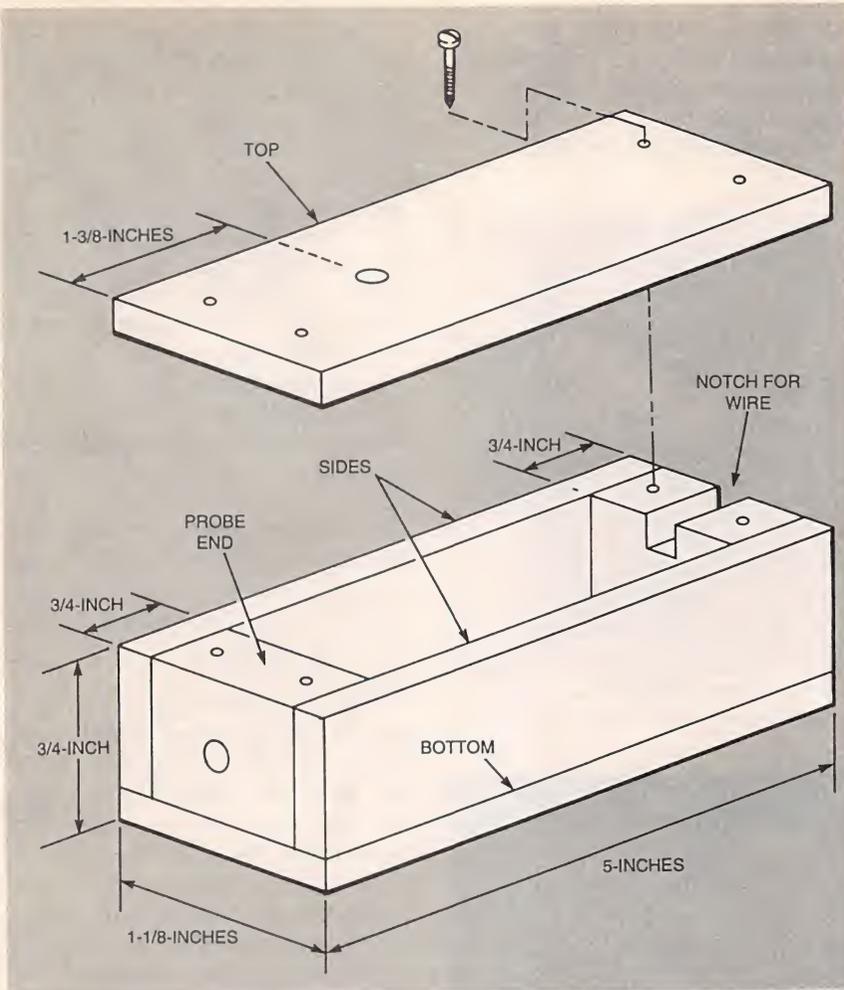


Fig. 2. The SmartProbe's case can be made from pieces of wood or plastic that are glued together. The result is a custom-designed unit that is comfortable to hold and use.

How those numbers relate to the logic levels of the SmartProbe are detailed in Table 3. For example, if the SmartProbe is sensing a voltage between 40% and 70% of its power supply, the read value is 240. That value represents a valid logic high for TTL circuitry, but is below the logic-high threshold for CMOS devices.

One final detail is how S1 is read: simply mask the input data with 8. A value of 8 means that S1 is closed; if it is open, the value will be zero.

**Construction.** Building the SmartProbe is very simple and straightforward. None of the circuitry is sensitive or high-speed. You can use a piece of perfboard and standard construction techniques.

Before building the circuit, we will need the case to know how much room we have to work with. The author's prototype was built from

1/8-inch-thick wood. The overall dimensions are shown in Fig. 2. Those dimensions result in a case that is large enough inside to hold all of the electronics, yet is comfortable to hold like an oversized pencil. The author's prototype can be seen in Fig. 3, showing all of the parts that go into building the SmartProbe.

A notch will be needed in the

back end of the case for the power clips and the cable that will be attached to the parallel port. Don't forget to drill a hole for S1. If you mount S1 on the bottom side of the case, it will be easier to assemble the unit—no wires will need to be attached to the top of the case.

The probe tip itself can be a solderless tip jack or, if one is not avail-



Fig. 3. The author's prototype uses a stacked-board arrangement for the circuit board; a single board will fit in the space just as well.

able, it can be formed from a machine screw. If you're using a tip jack, drill the mounting hole slightly smaller than the threads on the probe tip. To form a probe tip from

the machine screw, file its end down to a point, forming a smooth taper approximately  $\frac{3}{4}$ -inch long from the pointed end. Drill a clearance hole in the probe end of the case.

Coat the mating surfaces of the ends, sides, and bottom pieces with wood glue and clamp together for about an hour. Remove the clamps and mount the top piece on the assembly with four sheet metal or wood screws. Sand all of the surfaces smooth and chamfer or round all 12 edges. Apply a finish such as paint or varnish to seal the surfaces of the wood. When dry, remove the four screws and the top.

Mount the probe tip to the case. For a tip jack, simply thread it into the hole that was drilled earlier. For a machine screw, install two washers and a nut on the end of the screw, place it inside the case, and pass the pointed end of the screw through the hole in the probe end. Loosely install a second nut on the part of the probe screw that is outside of the case.

With the case finished, you can cut a piece of perfboard for the circuit that fits the available space. If you mount S1 on the board, you can use the switch to hold the board inside the case.

The power wires can be color coded red and black with similar-colored alligator clips. A short length of insulated wire connects the probe tip to the circuit board. If you are using a machine screw, wrap the bare end of the wire around the screw between the two washers and tighten one of the nuts; the outside nut might be easier to reach. The data cable should be shorter than ten feet—four to six feet is a common length. Once it is wired up, install a DB-25 hood on PL1.

Once everything is mounted in the case, inspect your work before closing it up.

### Testing and Using the SmartProbe.

Basic software for the SmartProbe is available from the Gernsback FTP site ([ftp.gernsback.com/pub/EN/smartprobe.zip](ftp://ftp.gernsback.com/pub/EN/smartprobe.zip)); it contains the set of compiled programs that will let the SmartProbe be used in its basic functions. The software available from the source given in the Parts List includes additional programs as well as source code. While the main application can be run on even the oldest non-graphics PC, the Scope function (discussed later) requires an EGA or

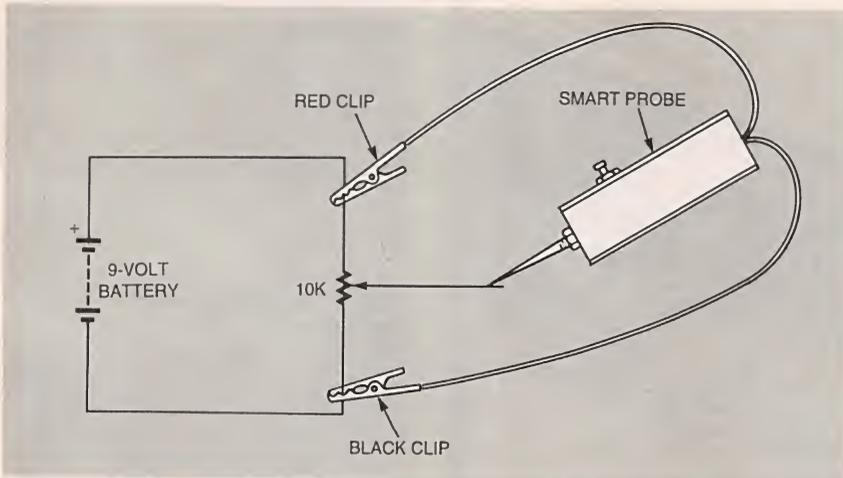


Fig. 4. To test the SmartProbe, a simple variable voltage is applied to the tip. When the software indicates that the SmartProbe has changed from one level to the next, the voltage is measured with a voltmeter to verify that the unit is working correctly.

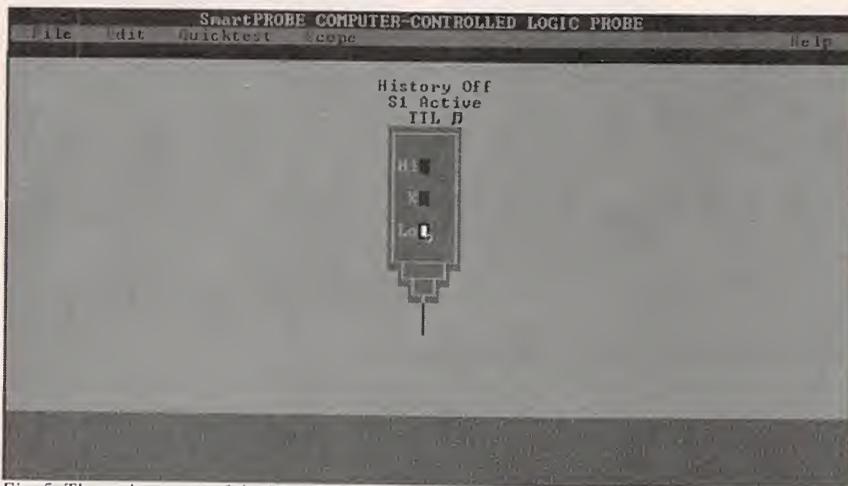


Fig. 5. The main screen of the SmartProbe software lets you easily choose the various settings and options.

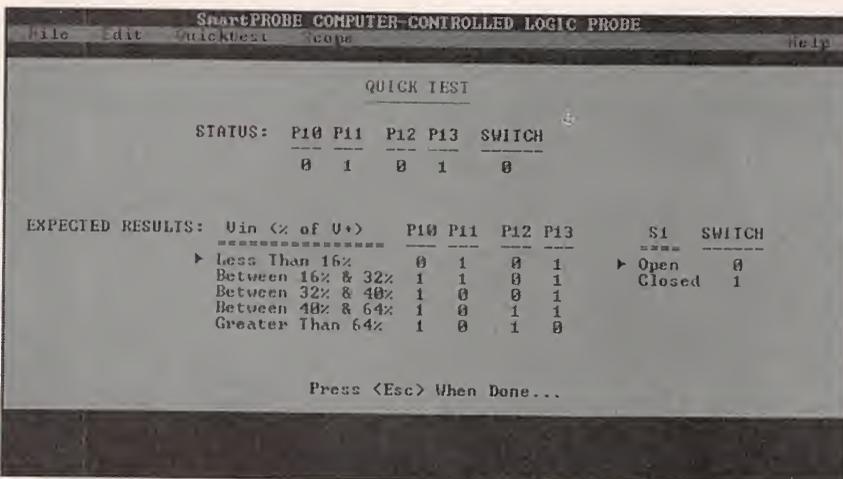


Fig. 6. When testing the unit, this screen tells you what the SmartProbe is reporting. The information at the bottom of the screen is a handy reference for testing the hardware.

TABLE 2—PARALLEL PORT RESPONSES

Probe Voltage	Pin 10	Pin 11	Pin 12	Pin 13	Read Value
0%–15%	0	0	0	16	16
15%–30%	64	0	0	16	80
30%–40%	64	128	0	16	208
40%–70%	64	128	32	16	240
70%–100%	64	128	32	0	224

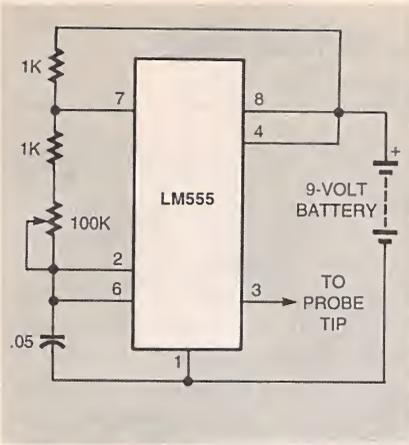


Fig. 7. This simple LM555-based oscillator shows the SmartProbe's ability to act like a poor-man's digital-storage oscilloscope.

better monitor.

The simple test circuit shown in Fig. 4 will be used to test the SmartProbe. Although a 10,000-ohm value is shown for the potentiometer, any value between 1000 ohms and 1 megohm can be used; the voltage source can be anywhere between 5 and 15 volts.

Connect the red and black power leads of the SmartProbe as shown and plug the SmartProbe into the printer port. After the software has been stored on the PC in a directory of your choice, type "SP" at the DOS prompt to start the program.

Once you see a startup screen with some initial instructions, press any key. The instructions will be replaced by a graphical representation of a logic probe as shown in Fig. 5. Press <ALT> Q to perform a Quick Test. You should see a screen similar to the one shown in Fig. 6. Note that the SmartProbe's current status is to the right of the STATUS indication; expected results are listed in the lower half of the screen. Press S1 on the SmartProbe and notice that the switch indication changes from "0" to "1". Release S1 and the indication should return to "0". Hold the SmartProbe's tip against the wiper of the potentiometer and turn the potentiometer's shaft. The "P10 P11 P12 P13" indications should change to one of the five conditions listed under EXPECTED RESULTS. If you do not get one of those indications, check the unit for component, connection, or construction errors; fix any errors as needed.

When the SmartProbe is giving a correct indication, measure the DC voltage at the point when the SmartProbe changes from one condition to the next. The voltage should be approximately as expected; there might be some variation based on the actual value of the components used. For example, the voltage between the "0 1 0 1" and "1 1 0 1" transition should be about 1.35 volts with a battery voltage of 9 volts. If you do not get those results, check the value and placement of resistors R5-R9 and correct any problems that are found.

To complete initial testing, press the <ESC> key to return to the program's main screen.

Now that you've used the Quick Test function, let's quickly review the other four menu choices (File, Edit, Scope, and Help) available from the main screen (Fig. 5). The File menu (accessed by pressing <ALT> F) contains the DOS Shell and Exit commands. Pressing S for DOS Shell temporarily brings you to a DOS prompt where you can perform normal DOS functions. Type EXIT and press <ENTER> to exit the DOS shell and return to the pro-

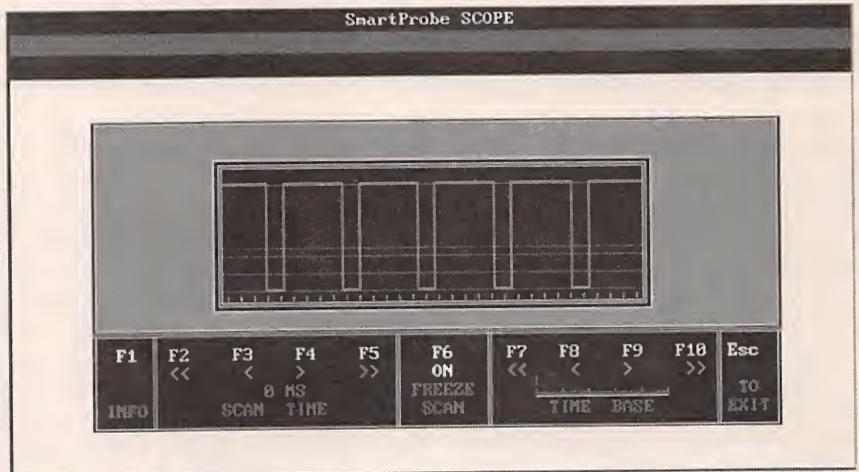


Fig. 8. When using the SmartProbe to check the output of the circuit shown in Fig. 7, the output pulse is accurately displayed.

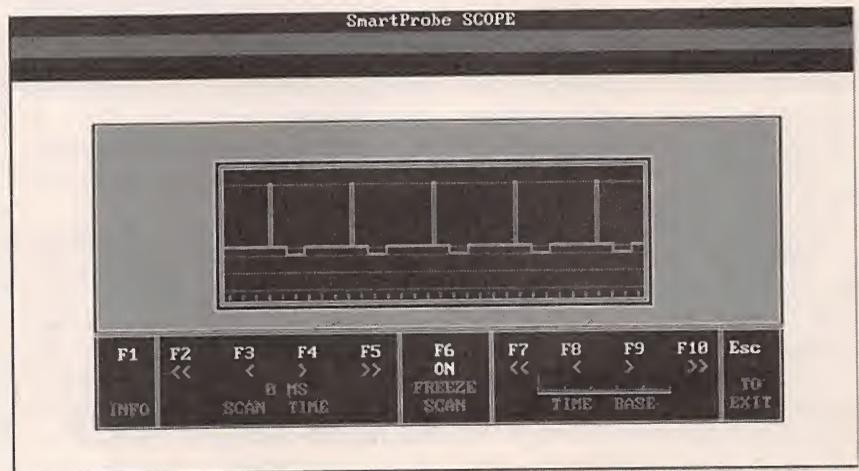


Fig. 9. Since the SmartProbe can sense four different voltage levels, it can be used to read simple analog signals. Here is the signal on the capacitor of the circuit shown in Fig. 7.

gram. To end the program, press X for Exit.

The Edit menu contains five commands: Toggle History, Toggle Sound, Toggle S1, TTL, and CMOS. Pressing H for Toggle History turns the History display on (if it was previously off), or off (if it was previously on). The History On command displays the previous 20 readings; each reading is stored when S1 is closed or the last 20 changes in reading if S1 is disabled. Toggle S1 enables and disables S1. With S1 enabled, a reading will only be taken when S1 is depressed. Disabling S1 allows for continuous readings. Pressing S for Toggle Sound turns the audible indication on or off accordingly. Pressing T for TTL switches the SmartProbe to the TTL mode. Pressing C for CMOS switches the SmartProbe to the CMOS mode. The current status of History, S1 status, Sound, and TTL or CMOS mode are displayed directly above the graphical representation of the SmartProbe.

The Scope function turns SmartProbe into a simple form of digital-signal oscilloscope that can resolve signals up to about 10 kHz. To try out that function, build the test circuit shown in Fig. 7. That circuit is a standard LM555-based oscillator with its output at pin 3. Pin 2 is connected to pin 6 and makes the oscillator run by charging and discharging the 0.05- $\mu$ F capacitor. The results of measuring the signals at pins 3 and 2 of the LM555 are shown in Figures 8 and 9 respectively. Since the output of pin 3 is a digital signal varying between logic low and logic high, Fig. 8 is a true representation of that signal. The signal at pin 2, however, is an analog signal created as the capacitor charges and discharges. It will therefore have a sharp rise and an exponential decay. Figure 9 shows how the SmartProbe creates a digital approximation of that signal. Although it can't show the exponential decay, it can show the width of each pulse and the time between each voltage level. Even with that limitation, the SmartProbe provides a lot of information for the low cost of the unit.

You can select the various scope functions with the ten function keys. For instance, F2 through F5

adjust the scan time (the time between subsequent scans). Since there is no sync input, the display tends to jitter; increasing the scan time reduces that jitter. When you want to examine a particular scan, you can freeze the display by pressing F6. Function keys F7 through F10 control the time base (somewhat like a time base on a standard

TABLE 3—TTL, CMOS LEVELS

Value	TTL	CMOS
16	Low	Low
80	open	Low
208	open	open
240	High	open
224	High	High
0	S1 Open	S1 Open
8	S1 Closed	S1 Closed

oscilloscope). Pressing the Escape key returns you to the main screen.

The Help function is a non-context sensitive help file that provides user information on the operation of the program and on the SmartProbe hardware.

**Expanding the SmartProbe.** Since the SmartProbe is computer controlled, adding new or different functions and features is simply a matter of writing custom programs for those who have the tools and need. For instance, those doing digital-electronics manufacturing and repair are always looking for a quicker and easier way to repetitively check circuits. One such application is included in the software bundle available from the source given in the Parts List. It lets you quickly and repetitively test circuits based on a test specification that you develop (and document) in a plain text file. It provides a running pass-fail indication (with an audible beep if something does not comply) and a printed test report if needed.

You can also use the parallel port's unused output pins (along with appropriate external circuitry) to control external devices based on the levels sensed by the SmartProbe. Even if you choose not to expand its capabilities, you'll find the SmartProbe to be an inexpensive yet extremely versatile addition to your test bench. Why not build one today!