Using pressure transducers, call progress detectors, adapting surplus drives, electronic halftone secrets

By Don Lancaster

We've gotten quite a bit of mail involving several of our previous columns. Lots of you have asked for more sources of cubic spline information. One good reference is *Fundamentals of Interactive Computer Graphics* by J. Foley And A. Van Dan, published by Addison Weseley.

Several others have asked for additional information on pressure transducers, so we'll try to pick up on that topic this month. I am working up a construction project or two on this, but lately I have been up to my ears in laser printing, so it may take a while.

Don't forget that I still have lots of freebies for you. These include laser printer demo packs, word processing utilities, shaft encoder software, EPROM programming adaptors, and bunches more. Just write or call the helpline number listed at the column's end.

On to this month's goodies . . . .

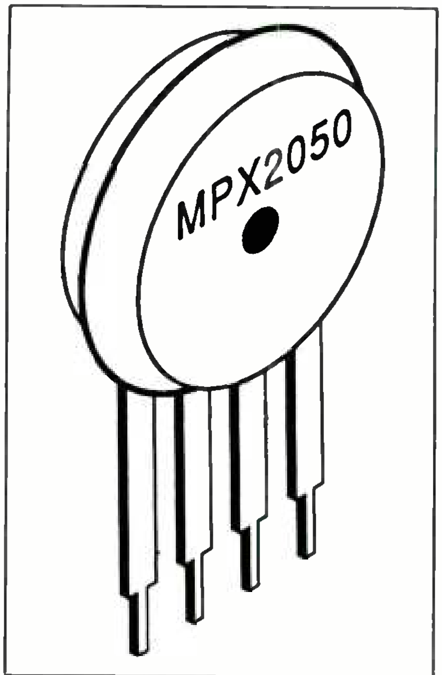


Fig. 1. A pressure transducer.

Tell me more about pressure transducers

A pressure transducer is any device that will accurately convert a pressure difference into a proportional current or voltage. As we saw two columns back, one important use of pressure transducers involves stream gauges. Other uses include weather forecasting, cave surveying, altimetering, tactile robotic sensing, auto emission controlling, medical instrumentation, weighing, and industrial process controlling.

While not yet super cheap, some pressure transducers have dropped enough in price that they have become most interesting and most challenging hacker components. Some sensors are now priced in the \$20 to \$30 range, and this is almost certain to drop further.

How can you measure pressure?

One very old method uses a column of mercury working against a vacuum. As the column moves up and down, column height can be measured and then related to atmospheric pressure. Another method requires a sealed bellows. The bellows expands or contracts with pressure variations. Bellows position can then be measured and related to pressure.

The modern electronic way of sensing pressure involves a new family of integrated circuits called silicon pressure transducers. Two of the leading suppliers are *Motorola* and *Micro Switch*, although there are a few others.

Important advantages of silicon pressure transducers include their repeatability, relatively low cost, high accuracy, and the ability of some models to internally compensate against fluctuations in temperature.

Figure 1 shows a typical pressure transducer. There are many different packages available, depending on what kind of pressure is to be sensed and the kind of environment in which they are used.

A silicon sensor is actually very simple. A piece of silicon is etched so it forms a very thin diaphram, somewhat similar to a drumhead. As the pressure on one side changes, the diaphragm flexes and changes its size. This flexing is sensed by a resistance strain gauge bridge implanted directly on the diaphragm. Laser-trimmed resistors are sometimes added for calibration and to compensate for temperature variations.

As pressure changes, the diaphragm flexes, the resistor values change, and a differential output voltage is produced. You typically get 40 or 50 millivolts out for full-scale deflection.

The sensor is connected as a simple bridge. You apply a regulated + 10 volts. A small differential voltage appears across the positive and negative outputs. This voltage is proportional to pressure. The common-mode (fixed offset) voltage at these outputs is typically one-half the supply voltage.

Normally, the output signal is isolated and single-ended with a first operational amplifier stage and is then offset and amplified with a second stage. The 0-to-5-volt dc output level can be A/D converted or used directly as an analog signal or fed to a meter.

There are three fundamentally different types of pressure transducers. A *relative* transducer measures the pressure difference between its two sides. If both sides of the transducer are made available, you can measure the relative difference between two pressures.

A second type of transducer is called a *gauge* transducer. With a gauge transducer, the second port is vented to ambient air. The pressure difference between your input and the current atmospheric pressure is then measured.

A gauge transducer can be either *vacuum ported* or *pressure ported*, depending on whether you want to measure pressures above or below that of ambient.

The third, and rarest, sensor is called an *absolute* pressure transducer. With this type of transducer, one port is permanently connected to the best possible vacuum attainable. The output signal then measures absolute input pressure.

Note that most silicon pressure transducers start out as relative devices. Giving access to the second side leaves you with a relative transducer. Venting the second side to ambient gives you a gauge transducer, while sealing the second side in a

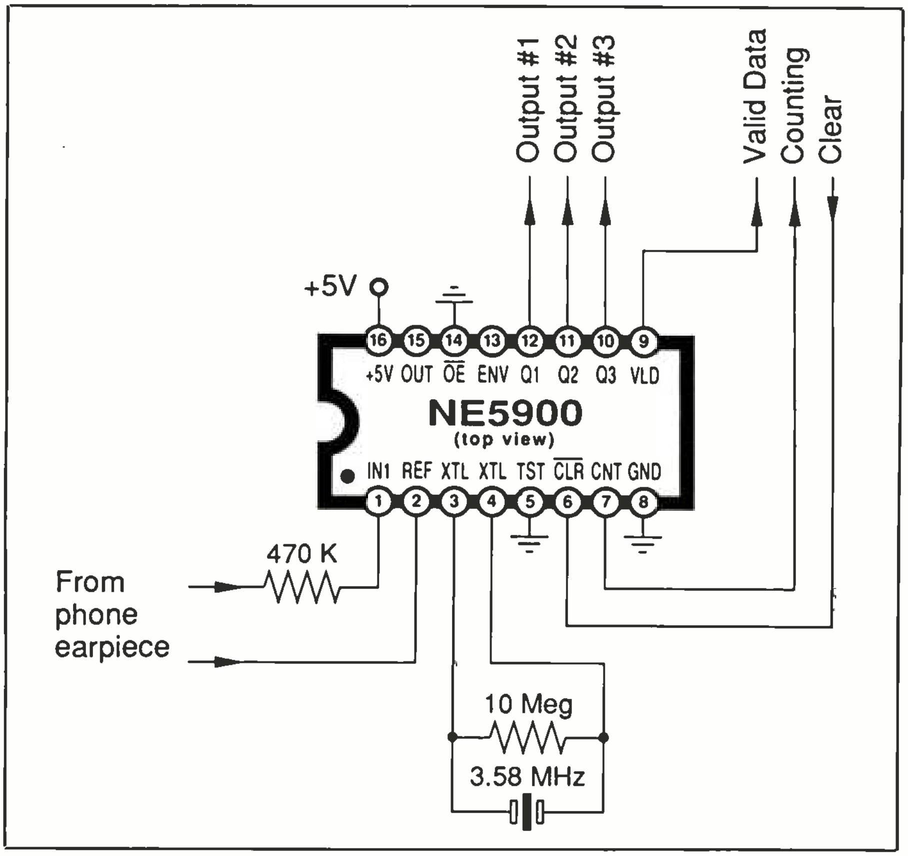


Fig. 2. How to use an NE5900 call progress detector.

good vacuum gives you a way to measure absolute pressures.

Full-scale range varies with your choice of pressure transducer. Motorola's MPX2050 series is rated at 0 to 7.5 psi—an ideal range for stream gauges, electronic emission controls and robotic sensors. Other full-scale ranges are available for other uses.

Normally, you are limited to a 100-percent overload before the transducer will fail, so you always want to pick the most sensitive unit you can, consistent with the maximum pressure with which you want to work. The transducer itself is protected with a silicone gel coating that apparently resists clean water, some other liquids, and many weaker chemicals.

Tellyawhat. Let's have a contest. A free *SAMS* book to the best five hacker ideas that involve pressure transducers. The overall winner gets both an all-expense paid tinaja quest for two (FOB Thatcher, AZ) and some cash-type money if the idea is good enough to qualify as a *Modern Electronics* construction project article.

Fair enough?

Can 80-track disk drives be made Apple compatible?

I've sure gotten a lot of calls on this lately. The quick answer is yes and no. Yes, a knowledgeable hacker can adapt anything to anything if he puts his mind to it. No, I know of no quick and simple way to save time and/or money with these bargain drives.

First, there are the physical differences. *Apple* drives normally use 35 (or rarely 36) tracks, but certain protection schemes on some Apple software make use of half tracks, quarter tracks, and may even use track arcing techniques.

Even if you could get the tracks to line up, it is very unlikely that you could reliably run any and all existing Apple software on the new drive.

The way in which the drives are accessed is fundamentally different as well. On an 80-track drive, much of the operating system that accesses the disk is provided for in hardware inside the drive itself. On an Apple system, most of the operating system is provided for in software residing in the Apple main RAM.

These two different methods of controlling a disk mechanism are philosophically and fundamentally different. It would seem to be a real bear to get between the two.

Note that many Apple programs often make slight to major changes in the operating system they use. This is sometimes done for "protection," whatever that is, and at other times to speed up access or pick up more storage, or whatever.

As a conservative guess, there are probably several hundred or more different Apple disk operating systems in use today. A hardware-based operating system would have to know about all of them and be able to serve them all equally well. Sounds tricky.

A lot of people have also asked where they can get the schematic for the super-secret IWM, or Integrated Woz Machine custom disk controller used in the Apple IIc. This is nothing but an adaptation of the plain old "slot six" disk controller card. The schematic of this card appears in any of the earlier DOS 3.3 manuals.

There are also problems in adapting standard 3.5-inch drives to the IIc or the Macintosh. Most non-Apple 3.5-inch drives spin at a constant speed. Apple spins their 3.5-inch drives at a variable and track-dependent speed. This lets you get more data on the disk, since more ones and zeros can be crammed onto the longer outside tracks.

So, if you are looking for a quick way to save a buck on disk drives, forget it. On the other hand, if you can find a plug-and-go hardware adaptor that is guaranteed compatible, go for it. Better yet, design the adaptor yourself. *If* you can make it simple and cheap enough, there is a big market out there waiting for you.

What is a call progress detector?

When most people make a telephone call, they are usually swift enough to figure out for themselves when the phone is ringing or if the line is busy. This same ability is needed by modems, alarms, auto-dialers, and many microcomputers involved in any sort of telecommunications. Finding out exactly how far a call has gotten is called *call progress detection*, and the electronics needed to supervise a call going through often goes by the name of a *call progress decoder*.

Call progress detection used to be very complex and expensive, but today *Signetics* has a simple and easy-to-use IC called the NE5900 Call Progress Decoder.

Figure 2 shows how to connect the NE5900. This IC needs a single + 5-volt supply and ground. A stock color TV crystal generates the 3.58 MHz needed for internal timing. Two input lines come from the telephone receiver earpiece by way of a 470k resistor. A clearing input is routed from your microcomputer or other controlling electronics every time you want to check on the call status.

There are three main output lines. The binary code on these lines tells you the status of the current call in progress. Figure 3 shows the output codes involved.

There are four likely responses when a call is being placed. The *dialtone* is a continuous tone pair that tells you the line is available. The *audible ring* is a two-note tone that is on for two seconds and off for four seconds. The *busy signal* is a two-note tone that is on for half a second and off for half a second. There is also a possible supervisory *reorder* two-note tone that continuously is on for 0.2 second and off for 0.3 second.

The exact frequencies in use for each response depend on the age of the phone equipment. Fortunately, the time duration of each response is constant and predictable, no matter the age of the phone system. The NE5900 measures the time interval, or more precisely, the cadence of each response.

The chip first does a sloppy bandpass filtering job, using the external resistor and a pair of internal capacitors. A more precise filtering to a passband of 300 to 640 Hz is then done with a precision switched-capacitor filter. The filter's output is then detected digitally, yielding the envelope of whatever tones are present.

To use the chip, you first bring the clear input low. Do this every time you want to find the status of a call being placed. This starts a 2.3-second timeout that begins as soon as a tone of any type is received. During the next 2.3 seconds, the input envelope is tested. If it is continuous, the dialtone status code is outputted.

If the envelope is on for only two seconds, the ring status code is outputted. Note that most modem tones are well above the filter passband and will be rejected. If the envelope alternates half a second on and half a second off, the busy code is outputted. Should the envelope alternate 0.2 second on and 0.3 second off or so, then the supervisory reorder status code is outputted.

Finally, if there is extreme noise on the line or if someone is talking, an overflow code is produced. This tells the computer or the controlling electronics that the NE5900 was unable to do its job.

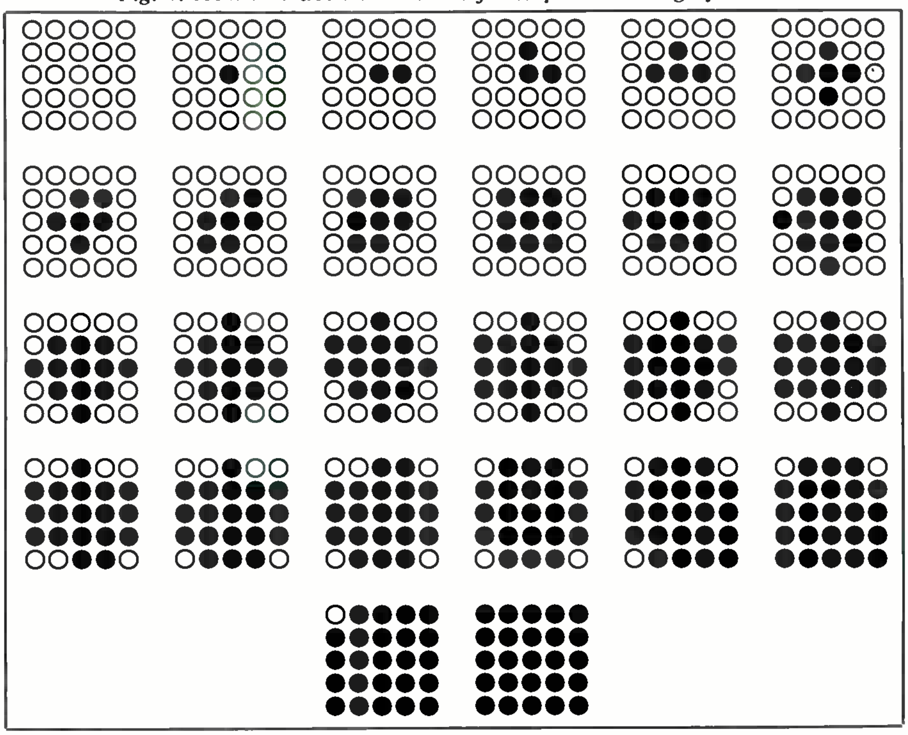
	Q1	Q2	Q3
Dial Tone	0	0	0
Ringing	1	0	0
Busy	0	1	0
Reorder	0	0	1
Overflow	1	1	1

Fig. 3. The call status codes.

There are several other pins on the package that may be of interest to advanced users. An "envelope" output is available that lets you do your own testing for oddball responses. An "enable" pin lets you turn off and on the three status outputs and the envelope line. The enable pin is brought low to activate the outputs. This is handy for tristate bus use on a microcomuter's data bus.

There's also a "data valid" output that

Fig. 4. How a 25 dot electronic halftone produces 26 gray levels.



NAMES AND NUMBERS

Addison Weseley
Microcomputer Books
Reading, MA 01867
(800) 238-3801

Motorola
Box 20912
Phoenix, AZ 85036
(602) 244-6900

Micro Switch
11 West Spring Street
Freeport, IL 61032
(815) 235-6600

SAMS
4300 West 62nd Street
Indianapolis, IN 46206
(800) 428-SAMS

Signetics
811 East Arques Avenue
Sunnyvale, CA 94088
(408) 739-7700

can be used to interrupt a microcomputer, as well as an "analog" output that has only been filtered. A final pin is used for testing and must be held at ground.

One gotcha: that input 470k resistor is a critical value, since it is used as part of an internal bandpass filter.

How do electronic halftones work?

Today's printers are offering better and better resolution, so it is only a matter of time until you will routinely be able to print your own superb quality photographs or video images. Even today, the *Laserwriter* is capable of directly printing multi-tone images to "fair" quality. If you are willing to 2:1 reduce your final laser artwork, you can easily upgrade this to "good" quality. By a good image, I mean 106 or so gray dots per inch of 33 possible gray levels.

The key to printing an apparent gray scale with a printer that can print only a black or a white dot involves an *electronic halftone*. To understand how you print gray images, you must know just what an electronic halftone is and how it works.

Suppose you abso!utely had to have something truly gray appear on a printed page. The only possible way to do this would be to use gray ink. Should you need several shades of gray, several passes through the press would be needed.

There has to be a better way—and there is. Instead of really printing gray, you print a bunch of tiny black dots. You make the size of the dots larger for darker grays, smaller for lighter areas. The dots are made so small that they exceed the eye's angular resolution. Instead of seeing dots, the eye averages out, or *integrates* the black and white areas, and produces a gray blur.

To prove this to yourself, just drag out a good magnifier and look at some of the photos right here in *Modern Electronics.* Note that there is no gray used *anywhere* in this magazine; it is all done with black and white dots.

In traditional printing, grays are photographed through a magic screen that produces a halftone image. In electronic printing, we fake grays through a somewhat similar electronic halftone process.

Let's throw a couple of terms at you. The *resolution* of the screen is normally specified in dots per inch, or dpi. For instance, a 120-dpi screen gives you 120 dots horizontally and 120 dots vertically for a total of 14,400 dots per square inch.

The number of *gray levels* in a particular screen is set by how many distinct black dot sizes are possible. This is more or less a continuous function with traditional screens. With electronic halftones, you must trade off resolution against gray scale, as we will shortly see.

Halftone screens are normally rotated at a screen angle of 45 degrees. This minimizes any visual distraction the screen might produce. Other angles are sometimes used for special effects or custom work.

Most electronic printers are only capable of placing or not placing a dot in a specific location. Getting from here to an electronic halftone depends on what you want to call a dot. Let's use the Laserwriter as an example. This printer has a 300-dpi resolution, so it can print 90,000 dots per square inch. The trick to an electronic halftone is to use several of these dots bunched together to represent a single gray splotch on the final image.

One possibility is to use a gray splotch that is 5 dots wide and 5 dots high. As Fig. 4 illustrates, you can trick the eye into seeing any of 26 possible gray levels, depending upon how many of the dots are "lit" at any one tme.

Other "dots-per-splotch" values can be picked. The more dots you use, the greater the number of gray levels available, but the cruder the overall resolution. Fewer dots give you better looking and smoother grays, but will restrict the number of gray levels.

There are other restrictions to using very small numbers of dots per gray splotch. Very light grays are often desired. It is difficult to get these light grays with small splotches, first because there are only a few possible levels using only a few dots, and second because the dots tend to *overprint* for uniform line widths.

The order in which you blacken the dots in called the *spot function*. For electronic halftone use, the spot function starts out small and centered, and spirals itself larger with darker and darker grays. You can use any spot function you want for special effects, such as pattern screens, or patterned bit-mapped background fills.

You can easily and instantly change the Laserwriter halftone screens at any time. This is done with a simple text command. Sadly, halftone ability is conspicuously absent from most other low-end-priced laser printers.

Applewriter on an Apple IIe does some outstanding gray-scale work when combined with the Laserwriter. Typical cartoon characters can be done with a file the size of a business memo, and no expensive digitizer is needed. Write or call for a free demo or two. **ME**

NEED HELP?

Phone or write your Hardware Hacker questions directly to:

Don Lancaster
Synergetics
Box 809
Thatcher, AZ 85552
(602) 428-4073