HARDWARE HACKER

Understanding shaft encoders, more on E.T. watching, working with Gray codes, finding Commodore ICs, underground radio communications and more

By Don Lancaster

Seems to be a lot of interest in shaft encoders these days, so that's what we will center this month's column on. Then we'll end up with some shafts of a wildly different type.

But first some news and updates.

The old Apple IIe Absolute Reset (Modern Electronics, February and March 1985) will *not* work on the new IIe with the "upgraded" ROM set, nor will it work on a IIc. Instead, "alike but different somehow" patches are needed. Full details on the required upgrade will be presented next month.

Here's some more on E.T. watching. You might find two additional books of interest. The first is called *The Early Years of Radio Astronomy* by W. T. Sullivan and published by *Cambridge University Press* ($39.50). The second is titled *Serendipitous Discoveries in Radio Astronomy* by K. Kellerman and B. Sheets, and published by the *National Radio Astronomy Observatory* ($7). You'll find reviews on these in *Science,* Issue 4701, May 17, 1985, Volume 228, pp. 854-856.

I have got some brand-new Applewriter/Laserwriter Utility diskettes all ready to go. You simply will not believe what these gems can do. The *Gila Valley Apple Growers Association* will be most happy to send you a free packet of samples showing what you can do with an Apple IIe and a Laserwriter creatively interacting with each other. Things chosen, naturally, that are insanely easy to do on a IIe or IIc and extremely difficult on a Mac, using both *WPL* and *Postscript* working together, of course.

As usual, this is your column, so write or call your hardware hacker questions and problems to my address in the box at the end.

On to this month's goodies.

What is a Shaft Encoder?

A shaft encoder is any scheme to find out something useful about something that is turning. You might be interested in the speed, the direction, the absolute position, the relative position or phase, or possibly even such exotic things as acceleration, torque, or strain.

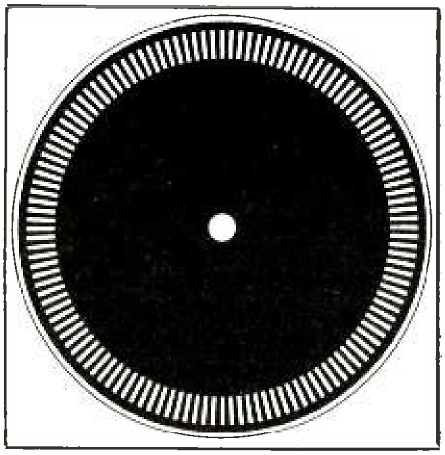


Fig. 1. An encoding disc for an incremental shaft encoder.

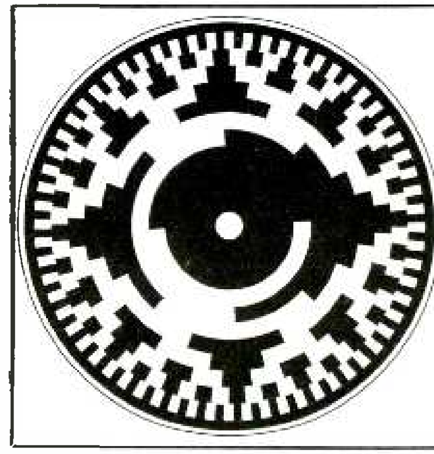


Fig. 2. An encoding disc for an absolute shaft encoder.

As an example, years ago I was working on a car safety radar and needed a way to measure vehicle speed. This was long before such niceties as integrated circuits or sanely priced electro-optic goodies existed. What I did was take an old surplus dc motor and removed everything but the brushes, the shaft, and the communtator. I then shorted every third bar of the 18-bar commutator together. The beast was then cut down and repackaged so you would unscrew the speedometer cable, add the encoder, and then replace the cable. As the shaft rotated, you got six commutator shortings, and thus six pulses per revolution.

Today, most shaft encoders are done using electro-optics. An optical disk that has clear and opaque areas usually interrupts a beam of infrared light. You can put an infrared LED on one side of the disk and an infrared sensor on the other, or else you can use an opto-interrupter. This one has a slot in it that accepts the edge of the disk. The advantage of an opto-interrupter is that all the electronics is in one $2 package; the disadvantage is that only the outside track of an encoder can normally be read.

Important suppliers of shaft encoders and opto goodies include *General Electric, General Instrument,* and *Hewlett-Packard.* Or write or call Forrest Mims (c/o Modern Electronics), since he is much more into opto stuff than I am.

Important advantages of opto shaft encoders are that they do not load the shaft, do no wear, and are simple, clean and reliable. They preform reasonably well when somewhat dirty or misaligned.

There are times, though, when mechanical shaft encoders may be the better choice. One example is in a wilderness data acquisition system that is totalizing wind speed from an anemometer. Sometimes you simply cannot afford the luxury of continuously lighting one or more LEDs for a month or more from a small battery. In this case, a "zero-power" mechanical contact may be the best choice.

You will find two basic types of shaft encoders. An *absolute* encoder has many sensing channels and always knows where it is at. For instance, a four-channel absolute encoder could give you the 16 points of the compass rose needed for a wind vane sensor.

The second type is the *incremental* encoder. An incremental encoder usually takes a single channel, or at most a very few channels. Incremental encoders only deliver pulses. An incremental encoder is ideal for something like an anemometer wind pickoff. You can count a few pulses to obtain instantaneous wind speed, or count all pulses for totalized wind over a given period. Instantaneous wind speed would be useful for weather forecasting, while totalized wind is of interest to wind energy people.

An incremental encoder can become more or less of an absolute encoder if you add counting electronics to totalize the counts and remember where you are. This can be done either with add-on hardware or with software routines in a microcomputer. Counting is often far cheaper than going to a true absolute encoder, particularly if long run of wire or a long distance is involved.

The weather station stuff by the *Heath Company* gives a good example of both types of encoders, done elegantly and simply. In fact, the same circuit board can be used for either type of encoder. Once again, *Hewlett-Packard* has some ready-to go shaft encoders, such as the HEDS-5000 series. These have recently been available on a $20 special promotion. H-P has announced a lower-cost series of digital potentiometers as well.

Robotics makes lots of use of shaft encoders. It is often much easier to measure where you are and correct a position with feedback, compared to being extremely accurate in the first place. The use of feedback to correct speed or position is sometimes referred to as a "closed-loop servo" system.

Figure 1 illustrates the wheel pattern for an incremental shaft encoder. Figure 2 is the pattern for an absolute encoder. Note than though the absolute encoder seems coarser by a factor of two, both disks are eight-bit encoder wheels with a resolution of one part in 256, roughly $\pm 0.7°$. By the way, I have some software that makes designing master artwork for low and medium resolution encoder wheels utterly trivial and super cheap. How about six cents per original art master? Write or phone if you are interested in this software.

Oh yes—a gotcha learned the hard way. Infrared light sometimes behaves totally differently than regular light. Opaque may not be. Clear may not be. Be sure to check your encoder disk to make sure that what you think is clear is in fact transparent to infrared and what seems opaque really blocks infrared light. Certain plastics will actually behave exactly opposite of what you might expect; others will not give enough difference between "black" and "clear" to be of any use to you.

On to some fine points on encoders that create questions of their own . . .

How can I measure Direction With an Incremental Encoder?

Thought you would never ask. This very subtle problem has a most elegant solution. Position and direction are two different types of information, or two more or less independent variables. Thus, it is not reasonable to expect to handle both with a single channel. One or the other will end up ambiguious. Instead, you use two channels that are phase shifted by 90°. These are usually called sine and cosine channels and are set up so that the one channel changes in the middle of the black or the clear space of the other.

Now, if the sine channel changes while the cosine channel is black, you are going, say, counterclockwise. If the sine channel changes while the cosine channel is white, you are going clockwise.

Figure 3 shows how you can use a single D-type flip-flop to extract direction information from a pair of sine and cosine channels on an incremental encoder.

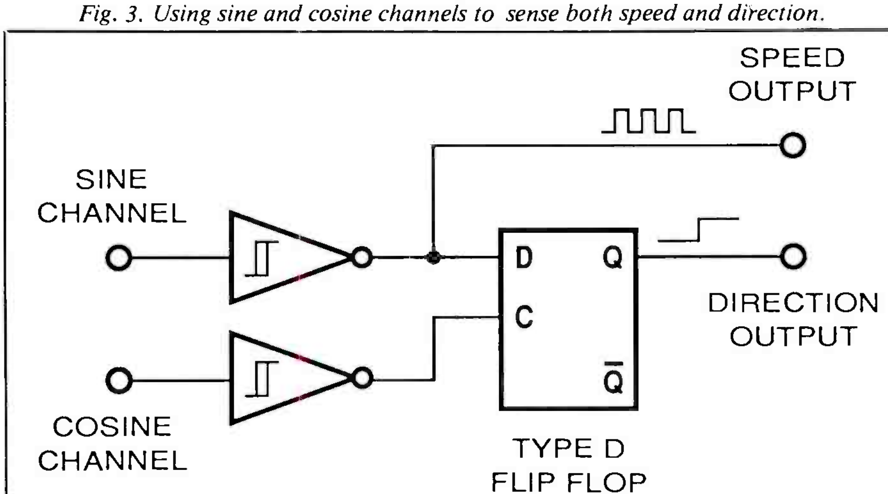
In this case, both channels are first conditioned with Schmitt inverters to eliminate noise. The sine channel is used as the actual speed pulse output. Every time the cosine channel changes, it samples the sine channel and stores it in the flip-flop.

If you are always going forward, the sine channel is always sampled when it is black, and you get a high direction output. If you are always going backwards, the sine channel is always sampled when it is white, for a low direction output.

Incidentally, this sine and cosine business also shows up in such diverse places as singe-sideband communication and various radar systems.

Do we really need two tracks on the encoder? We could used only one track and space the two sensors half a slot apart. That gets mechanically tricky. Instead, why not let the sine and cosine channels look at different slots at the same time? You can then space your opto-interrupters, say, 10½ slots apart, or even halfway around plus or minus half a slot. Thus, only one incremental encoder pattern is need to get both sine and cosine channels, if you are careful as to where and how you position your sensors.

Fig. 3. Using sine and cosine channels to sense both speed and direction.



Question: What happens if the hole is off center? How far off can you be for a given wheel size and resolution? Sometimes a single hole is placed in a second channel to give you an absolute "zero" reference in an otherwise incremental encoder. This lets you find out where you are every now and then. Some floppy disk systems also use this technique.

What is a Gray Code?

A Gray code is a very old and very special computer code that today nicely solves a sticky absolute shaft encoder problem.

There used to be all sorts of weird, wooly and wonderful computer codes that served many obscure and arcane uses. Thankfully, most of these are long gone. For instance, there was the EBDIC code used to purposely confuse IBM customers. There was an excess-three code that slightly simplified the hardware needed for decimal arithmetic. There were all sorts of binary-coded-decimal codes, such as the 1-2-2-4 or 1-1-2-5 codes that simplified early digital counting displays. And finally, there was, and still is, the Gray code.

Let's go back to our absolute shaft encoder. Suppose we tried to use the "obvious" straight binary code for the wheel. Say further that it is an 8-bit wheel with 256 positions of absolute encoding. What happens when you go from $FF to $00 on the wheel? Before the motion, we are in position decimal 255 or hex $FF. After the transition, we are in position 0 or $00. But there is no way that all the bits will simultaneously change from ones to zeros. Some bits will be slightly ahead and some will be slightly behind at the intended transition point. This happens because of mechanical alignment and channel sensitivity varitions.

As a result, when you are on the hairy edge between $FF and $00, wildly wrong results will be output. Should these results actually get used, very ungood things could happen very quick like—ferinstance, on a steel rolling mill that just got told to go forward and reverse at the same time.

0000	0000	0000
0000	0000	0001
0000	0000	0011
0000	0000	0010
0000	0000	0110
0000	0000	0111
0000	0000	0101
0000	0000	0100
0000	0000	1100
0000	0000	1101
0000	0000	1111
0000	0000	1110
0000	0000	1010
0000	0000	1011
0000	0000	1001
0000	0000	1000
0000	0001	1000
0000	0001	1001

etc ...

Fig. 4. Gray code changes only one bit at a time, eliminating false counts.

Instead, how about an encoder pattern that changes only one bit at a time for each count advance? Since only one bit can change, it either does or does not do so. There is no possibility of a wrong code at any time, since you either get the "old" value or the "new" value. The absolute position code either changed or it did not. A Gray code is a code that lets only one bit change at a time, regardless of absolute position.

Figure 4 is a listing of Gray code. Note that each successive state is only one bit different from the previous state. The process can continue for as many bits as you like.

You can get from a Gray code to a straight binary with either hardware or software. The only hardware needed is a stack of exclusive-OR gates. The equivalent software can be some shifts and EOR commands or a simple table lookup.

While it is equally easy to get from binary to a Gray code, reverse conversion is rarely needed. If you need to do this, exclusive-OR gates, shifts and EOR commands, or table lookup can be used.

If you look back at Fig. 2, you will see the Gray code of Fig. 4 wrapped around the wheel. Note that as you rotate the wheel, only one bit changes at a time.

Start with the "all-black" position pointing a tad above due east, and work your way around the disk counterclockwise. Only one bit changes at a time. Note that all more significant bits change in the middle of either the black or white area of the least-significant outside track.

Virtually all absolute encoders use a Gray code or some varition of one. Once again, this prevents any possible wrong answers when on the hairy edge between two different positions.

Thus, Gray lets you clearly tell black from white.

Read any Good Data Books Lately?

This month's data-book-of-the-month-club selection is the new *Exar* databook. It has bunches of neat things in it involving linear integrated circuits, particularly phase-locked loop and function generator devices.

As usual, you pick up one of these by making a professional request on a business letterhead, or else by phoning the address shown in the names and numbers box. Of, if you can find someone that already has one, use the bingo card in the back of their manual to order your own.

One thing that does need mention, though. Don't take their modern stuff very seriously, except as an excellent tutorial background. Modems done digitally take fewer parts, require no adjustment or setup, will not drift, and generally perform much better. Their analog modem circuits are a very good review of the way things *were*.

Some readers say they're having trouble finding my books in their local bookstores. You can get them directly from Synergetics if you wish. Personally autographed, too.

Where Can I Get Oddball Commodore ICs?

While I haven't personally checked them out, *Boufal Services* lists a wide variety of

NAMES AND NUMBERS

Alberta Safety Division
10709 Jasper Ave., Floor5
Edmonton, Alberta T5J 3N3 Canada
(403)427-0199

Boufal Services
244 Fitzwater St.
Philadelphia, PA 19147
(215) 925-6469

Cambridge Press
32 East 57th St.
New York, NY 10022
(212) 688-8885

Exar Corp.
750 Palomar Ave.
Sunnyvale, CA 94086
(408) 732-7970

General Electric
West Genesee St.
Auburn, NY 13021
(315) 253-7321

General Instruments
3400 Hillview Ave.
Palo Alto, CA 94304
(714) 581-5817

Gila Valley Apple Growers
Box 809
Thatcher, AZ 85552
(602) 428-4073

Hewlett-Packard
640 Page Mill Rd.
Palo Alto, CA 94304
(213) 970-7500

Radio Astronomy Observatory
NRAO Workshop 7
Green Bank, WV 24944
(304) 456-2011

Howard W. Sams & Co., Inc.
4300 West 62 St.
Indianpolis, IN 46206
(800) 428-SAMS

Science
1515 Massachusetts Ave. NW
Washington, DC 2005
(202) 467-4350

Synergtics
Box 809
Thatcher, AZ 85552
(602) 428-4073

Commodore chips and stuff, including the rare and hard to get 6560 and 6567 VIC chips and the 6581 SID chip.

Wonder of wonders, they also have KIM-1's at very attractive prices. The KIM-1 was far and away the greatest microcomputer Commodore ever built. Since then, it has been all downhill.

To this day, there is no better way to begin to learn the fundamentals of machine-language programming than on a KIM-1. The KIM-1 is also an excellent choice for a dedicated micro for a solar controller, a cattle feeder, a weigh station, a pump monitor, or whatever else you can dream up that needs some "plug and go" smarts. Apples and Commodores emulate the KIM beautifully and simply for system development. You can even download or upload through the cassette channels.

Boufal Services also carries most of the manuals, including the obscure and hard-er to find ones.

Check them out.

Is any Information Available On Underground Radio?

Underground radio communications is of interest in mining, cave exploration, and in search-and-rescue operations. Right now, the field seems split in two. On one hand, there are cavers doing their own thing on a limited budget and using older and not very sophisticated techniques. On the other hand, there's the military with extremely expensive equipment and classified information that is difficult if not impossible to access.

Two types of communication are of interst. In people-style communication, you are interested in getting information between two people underground, or one underground and one above ground. Often, coded signals or tones are preferred to voice because of the low frequencies usually involved.

In radar-style communication, you are interested in finding out whether something is on the other side of the rock you are looking through, such as another cave or an unknown cave below the surface. Miners are also interested in voids since they might contain such nasties as methane gas in a coal mine or extremely pressurized gas in a salt or gypsum mine. Or simply an unknown and partially collapsed old digging.

Generally, radio signals go through solid rock vastly better than they can go from the surface through a moist dirt interface into the rock. It is often the first foot or two of radio surface penetration that has the staggering losses. Part of this is caused by moisture, with water having a very high dielectric constant. Another factor is the problem of avoiding reflections between media with different loss factors and dielectric constants.

An excellent example of cavers doing caver-type stuff is the report "Construction and testing of an Underground Radio" by Julian Coward and Ian Drummond of the Alberta Speleological Society. Their report is available from the *Alberta Safety Division,* a branch of the Canadian government.

While they did a really great job on this, I'm left with the feeling that so much more could be done so much better by using the latest in integrated circuits, the newest of antenna and battery technology, and the best of inside military information, particularly if aided by someone with some decent aerospace background.

NEED HELP?

Phone or write your hardware hacker questions directly to:
Don Lancaster
SYNERGETICS
Box 809
Thatcher, AZ 85552
(602) 428-4073