THE DRAWING BOARD

Smoothing out the sinewave-generator output

ROBERT GROSSBLATT

ANYONE WHO GETS INTERESTED IN ELECtronics and gets really hooked on it will progress through a number of clearly recognizable steps. In the beginning, you buy light-dimmer kits and burn your eyes out trying to read obscure directions written in muddy print. The next step is to buy components and, armed with a chart that lists the resistor color code and a soldering iron, burn your components up trying to build a light dimmer of your own design. Somewhere around here you begin to understand that there's more to electronics than Ohm's law, and you begin to read.

Now, we're all familiar with the truth of Grossblatt's 12th Law: He who doesn't have his head in a book has his head in something else. But the more general the rule, the more exceptions there are to it, and that applies here as well. After you've plowed through enough abstracts and journals, you'll learn how to apply Grossblatt's 27th law: What is written on paper is not carved in stone.

The difference between theory and practice is the difference between brain damage and common sense. The difficult task of plowing through countless reams of paperwork filled with obscure equations can often be eliminated by taking a look at the original problem on a different-color paper or walking away and letting your subconscious take over.

The perfect example of that is the problem facing us at the moment—finding the resistor values for our digital sinewave generator. There are three ways to go about finding the answer. 1) Trial and error. 2) Mind-warping math. 3) Common sense. The first one is OK, but only gives answers for a particular application. The second is OK for people who wear a bathing suit with shoes and socks. That leaves us with the third.

Believe me when I tell you that the standard method for calculating the resistor values involves math so hairy... well, even with a lot of equipment it would be difficult. The Fourier transforms and Fibonacci numbers are the easy part. The hard parts can only be solved using a variable interossiter. (Do any of you remember what that is or know how to spell it?)

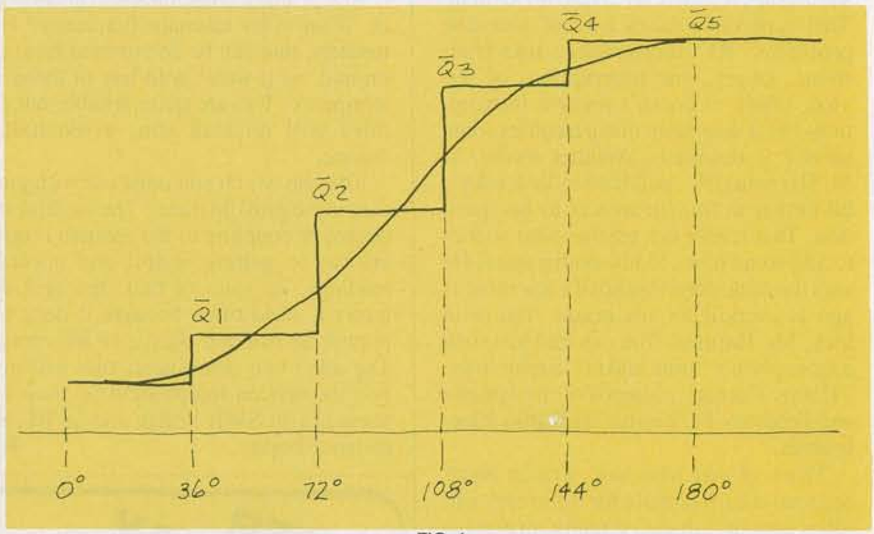

FIG. 1

TABLE 1

OUTPUT	ANGLE	SINE	NORMALIZED VALUE
$\bar{Q}1$	36°	0.588	1.617
$\bar{Q}2$	72°	0.951	1.000
$\bar{Q}3$	108°	0.951	1.000
$\bar{Q}4$	144°	0.588	1.617
$\bar{Q}5$	NOT USED		

FIG. 2

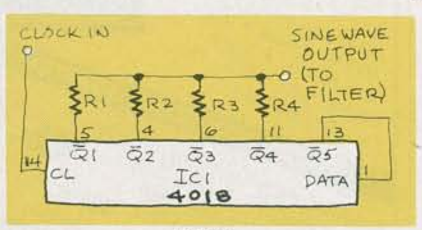

FIG. 3

Getting around the math

But seriously folks, the math is both complicated and unnecessary. We can get within several decimal places of the calculated values by using common sense and a bit of elementary arithmetic. Let's take a good look at the problem. Figure 1 shows the circuit we're going to use; Fig. 2 shows 180 degrees of the waveform that we want, and a couple of helpful hints.

You'll remember that we're not using the Q_5 output of the 4018 because it's a quick and dirty way to make the waveform conform more closely to a sinewave. The resistors on the remaining outputs will determine the shape of the wave we generate but—and this is important—we still have to allow for the time used by the Q_5 output. In other words, no matter how many 4018 outputs we decide to use, it's still going to take 5 incoming clock-pulses to make the 4018 repeat itself. That means any calculations that we do have to take into account the fact that there will be 5 discrete 4018 output states for each 180 degrees of the sinewave.

In practical terms, each incoming clock pulse will control 36 degrees (180/5) of the sinewave. Q_1 will determine the amplitude of the sinewave 36° into the cycle, Q_2 will determine the amplitude of the sinewave at 72°, and so on until we get to

Q_5. Even though we're not using it, we still have to allow for the time it takes for the 4018 to cycle through it.

Make sure you understand that!

Translating that bit of common sense to actual resistor values is really simple. We look up the sine of the angles we're interested in and generate a table like that shown in Fig. 3. We already know the angles we want—they're listed together in the appropriate column in the table. The last column translates that data into something that's easier for us human beings to use. All that we've done is to make the relative proportions a bit more evident by dividing all the sine values into .951.

So, you may well ask, what do we have to do next?

Well, believe it or not, that's all we have to do! All our work is done and the only arithmetic (as opposed to mathematics) we have left is some multiplication. What the last column in the table is telling us is that in order to generate a sinewave using 4 of the outputs from 5 daisy-chained flip-flops, the resistor values have to be in the proportions indicated. Pick a convenient value for R_2 and R_3, do the arithmetic, and you've got your resistor values! Of course, you might have a hard time finding standard-value resistors in the right ratios but that's a common problem—and, naturally enough, it has a common solution. You can use precision resistors if you're rich enough, or trimmers if you're not. In any event, we've got it made!

I know you haven't seen the math we managed to avoid doing, so you can't appreciate the kind of work we saved. What we've done is a classic example of how a common-sense approach to a problem can eliminate a lot of effort and keep the men in the white coats from your door. Let's go through the reasoning behind all that and make sure we understand it.

If the data is recirculated in the 4018, 5 incoming clock cycles have to pass before the output states start to repeat. One complete cycle of the 4018's outputs will be needed for each half of the sinewave we want to generate, regardless of how many of the outputs we actually use. That means that each incoming clock pulse will come when the sinewave we want to generate has advanced *one fifth of half its full cycle* or 36 degrees (180/5).

The amplitude of the sinewave at any point on the curve can be found by looking up the sine of the angle. Once we've listed all the ones we need, we can find the ratios of the resistors we need to generate the wave. See that? It *is* simple!

If you decide you want to use more flip-flops in the sinewave generator that you build, you'll have to recalculate the resistor values. Just go through the same reasoning we outlined and you won't have any problems.

From a practical point of view, I would recommend that you standardize the lowest value resistor at 10K or so and use trimmers to get the other values that you'll need. Just measure the 10K resistors to get the *exact* value, and do the arithmetic to find out where to set the trimmers. Set them out of the circuit and use a bit of nail polish to lock them in place before you put them on the board.

There are other parts to this sinewave generator we're slowly designing: the input clock, frequency selector, and the output filter. The most interesting one is the input clock. With a little bit of thought, we can make it variable so that the frequency selector can be something as simple as a potentiometer. That *was* one of our original design criteria.

Since we've already seen that the input clock has to run ten times faster than the maximum sinewave frequency we want to produce (remember—the 4018 is set up to divide by ten), we need a clock that can be tuned over a 1000:1 range with a twist of the wrist. There are a couple of things that come to mind that will fill the bill, but we ought to think about refinements such as crystal control of the frequency, stability, low-power requirements, and all those other good things.

And that brings us to next month, when we'll start on all that and see if we can put the whole circuit together in the real world. R-E