by Don Lancaster

Magic Sinewaves

Electric cars, solar energy converters, induction motor speed controls, and energy efficiency improvers all share a common problem: Create power sinewaves which are cheap, clean, and efficient.

I have recently been exploring the new *magic sinewave* approach that forms an obsessively compelling new route towards power electronic solutions. A fresh approach that offers great heaping bunches of multi-million dollar new *Midnight Engineering* opportunities.

Usually, you'll want to start with some sort of a raw dc supply and a pair of SPDT switches known as an *H-Bridge Drive*. The object of the game is to flip the switches in such a way that a motor or transformer load thinks it is seeing a clean sinewave of set amplitude and frequency.

Ideally, you'll want to flip your switches as *few* times per cycle as possible. For each switch flip will cost you dearly in high frequency switching losses.

Magic sinewaves to the rescue

A *magic sinewave* is just a long series of repeating ones and zeros. Hundreds of them. Magic sinewaves usually repeat continuously to produce a waveform of just the right amplitude and frequency. By selecting the correct magic sinewave, the distortion can be (A) carefully controlled and (B) made utterly and amazingly low.

To use a magic sinewave, you stuff it into a small lookup table in a low end microcontroller. PIC or otherwise. Select an amplitude and shove it out one or more ports. Then you stall to hit your desired frequency. Simple and cheap.

Compared to classic PWM or *Pulse Width Modulation* sinewave methods, there's ridiculously less high frequency energy involved. Efficiencies can be much higher. You often can get by with smaller heatsinks and cheaper power stages. There's no modulation or subcarrier. And no load integration beyond a simple filtering. Nor any offset, noise, quantization, or hysteresis. Or anything analog.

As a bonus, magic sinewaves easily handle single or three-phase power. And can *eliminate* motor capacitors.

Fourier and his series

The key to understanding magic sinewaves is the classic *Fourier Series*. Which says that any repeating waveform is really a fundamental and some harmonics.

For instance, a square wave can be shown to consist of a fundamental frequency sinewave, a third harmonic at 0.33 of the fundamental, a fifth at 0.20 and so on.

A square wave is simple and cheap. But it has several flaws when it comes to ac loads. First, the low harmonics are very strong and hard to filter. Especially over a wide speed range. Unchecked, low harmonics can cause severe noise, vibration, heat, and efficiency problems.

Second, you only get one square wave amplitude. That of the power supply. And third, each transition involves a double switch flip. Single flips are more efficient.

Fourier tells us that we can get rid of even harmonics by making sure our patterns have a property called *half wave symmetry*. When you also seek out waveforms with *quarter wave symmetry*, only sine terms will remain for those odd harmonics. Greatly simplifying your analysis.

Your goal is to come up with a magic sinewave list of "enough" amplitudes, "minimum" jitter, "good" spacing, and "acceptable" distortion levels. Having the "highest" possible efficiency and the "lowest" number of transitions per cycle. Using the "fewest" switch flips.

Uh, let's see. Up at a twelve bit word, we end up with four solutions of four amplitudes. All of which have zero evens and a zero third harmonic. 0 1 1 1 1 0 0 -1 -1 -1 -1 0 ferinstance. And this choice involves only single switch flips. (The -1 means your current goes in the opposite direction through the bridge.) We've just analyzed 4096 magic sinewaves and found 4 more or less useful ones.

When you get up to the thirty bit words, you'll find nine solutions that have a zero third and a zero fifth harmonic. Most of which involve single switch flips. Here you are analyzing 1,073,741,824 magic sinewaves and finding nine sort of useful ones. But if you got this far, you probably discovered you only had to work with the top half of the waveform. Analyzing a mere 32,768 cases.

Forcing zero harmonics

As you make your words longer, the number of useful magic sinewaves increase at an infuriatingly slow rate. It's not until you get up into several hundred bits that you start to get into the really good stuff.

Exhaustive searching rapidly gets out of hand. Even a 60 bit word has 1,152,921,504,606,846,976 possible states.

One trick is to try and force zero low harmonics. To do this, your word length has to be a multiple of the harmonic being cancelled. The only lengths that let you cancel a third and a fifth are 15, 30, 45, 60… For these lengths, it turns out that certain combinations of ones and zeros must match in specific ways. Leaving you far fewer candidates.

Your first really useful stopping point is up at 210 bits, since 2 x 3 x 5 x 7 = 210. This is the shortest symmetric word that lets you force harmonics 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14, 15, 16, 18, 20, 21, 22, 24, 25, 26, 27, 28 and 30 to zero.

It turns out there's around 2219 or so magic sinewaves of length 210. Of these, ninety or so can be selected for fair amplitude spacing and tolerable mid-teen harmonics.

The first listing of these 210-bit zero harmonic magic sinewaves appeared in the April 1995 issue of *Electronics Now*. Available as HACK87.PDF on *www.tinaja.com.*

A minimum energy alternative

Alas, any time you try to over-optimize any engineering parameter, something has to give somewhere else. Forcing the first ten harmonics to zero certainly works. But forcing to absolutely and exactly zero leaves you with strong odd harmonics in the teens. 11, 13, 17, and 19.

So, a math genius by the name of Jim Fitzsimons came up with another route to magic sinewaves. Known as the *minimum harmonic energy* method. In which you make each one in the word give its all to the fundamental. This leaves each one with neither the energy nor the inclination to create strong harmonics. You still do end up with *some* energy at each and every odd harmonic. But this residual energy can be amazingly low.

The trick with second generation magic sinewaves is apply *noise spectrum translation* techniques. Similar to those used by the CD recording people. Wherein you first generate as little in the way of unwanted signals as you can. And then force any remaining undesirable energy very high in frequency. And as spread out as you can get.

Some filtering will be needed for any magic sinewave. After all, those square corners have to get there somehow. In general, the energy in a properly chosen magic sinewave is fairly equally splatterd among the first several thousand harmonics. But "loudest" in the thirties and forties.

Filtering is often easy to do and generally is much less than what you'd need with PWM.

While excellent results can be had with a simple two pole filter at twice your fundamental, even a motor's or transformer's inductance can be used. As can inertia.

And the results can always be improved upon simply by going to longer magic sinewave sequences.

A few asides: Magic sinewaves are easily generated with several hundred bytes of cheap microcontroller code. From crystals in the lower MegaHertz region. Single, dual, or three-phase. The maximum load switching rates are in the tens of kiloHertz. Average rates are *much* lower.

Finally, I guess you could think of magic sinewaves as a specialized form of *zero carrier PWM*. One with minimum hf energy, no subcarrier, and needing no integration.

Improvements

The first cut at the minimum harmonic energy synthesis method worked like a champ. Except for one tiny detail. The number of transitions in the words skyrocketed. Still better than PWM, but efficiency clearly suffered.

I published these results in the June 1995 issue of *Circuit Cellar Ink*. See MAGICSIN.PDF on *www.tinaja.com*.

The big question was "Could we have *both* minimum transitions *and* low distortion?" The amazing answer is that we can. You can even reduce the number of transitions and distortion together! By going to a new *clumping synthesis* procedure, you can discover magic sinewaves that give you low distortions for the transitions chosen.

At this point in the program, two other major discoveries were made. It turns out that you can *double* the number of amplitudes for any word length magic sinewave. Simply by digging deeply enough into the underlying math. It also turned out that you could further significantly reduce the distortion by an *annealing* process that considers both the fit *and* the total integrated error.

A ferinstance: Hex $0080 700F 81F1 F87F E3FF 9FFE is one quadrant of a 384 bit magic sinewave. Using a simple two-pole filter, the *total* harmonic energy 3 through 17 is an astoundingly low 0.034 percent!

Even *totally unfiltered*, this magic sinewave has a third harmonic way on down at 0.025 percent and a negligible fifth. This particular sinewave happens to output level 106 at 0.843 of the supply voltage.

There are *sixteen* single switch flips per quadrant. A comparable PWM circuit might require *ninety-six* double switch flips each quadrant. While I can't claim that this magic sinewave is *twelve times* more efficient than PWM, your high frequency switching losses might be reduced by a factor of twelve. Or even higher in other cases.

The brothers and sisters of $0080 700F 81F1 F87F E3FF 9FFE form a complete family of 384 bit magic sinewaves. Having 128 evenly spaced amplitudes. Each needing only *twelve bytes* of microcomputer memory.

A few insider secrets

There are untold zillions of magic sinewaves. Nearly all of which are totally useless. The trick is to find the really great ones. Those having ultra low distortion. To get useful results, strange filtering algorithms and other wondrously bizarre ploys appear to be needed. These ploys can reduce a centuries-long supercomputer task down to checking out a paltry few million candidate waveforms.

Here's a few magic sinewave searching tips I've picked up along the way…

Fourier fer sure – The centermost tool to magic sinewave searching is *Fourier Series* analysis. A classic "trig style" Fourier analysis seems to be a better choice than "FFT" techniques. Mostly because a classic Fourier is way more intuitively obvious. Besides avoiding windowing hassles. For maximum speed, use table lookups.

Use quarter wave symmetry – Always work with a single quadrant. Then reverse the quadrant. Then flip the quadrant pair. Which cuts your analysis word length by a factor of four. Working in quadrants also nicely zeros out all your even harmonics and all cosine terms of odd harmonics.

Pick multiples of 96 bits – The 384 bit word length is one rather useful choice. You will want a factor of four to work with quadrants and slash your microcontroller storage. This also gives you two-phase and split-phase operation. To be three-phase possible, you'll also need a factor of three. And to be three-phase friendly, you'll need yet another factor of eight. The latter to guarantee that all three phases happen to be operating in the same *bit position*.

Work with clumps – Magic sinewave efficiency gets set by your number of transitions. Which in turn gets set by the number of *clumps* of ones in your quadrant. Start off with your transitions and minimize distortion from there.

MAGIC SINEWAVE RESOURCES

Circuit Cellar Ink
4 Park Street #20
Vernon CT 06066
(203) 875-2751

Electronics Now
500-B Bi-County Blvd
Farmingdale NY 11735
(516) 293-3000

Jim Fitzsimons
31125 North 68th Street
Cave Creek AZ 85331
(602) 488-1859

GEnie
401 North Washington St.
Rockville MD 20850
(800) 638-9636

Nuts & Volts
430 Princeland Court
Corona CA 91719
(909) 371-8497

Synergetics
Box 809
Thatcher AZ 85552
(520) 428-4073

Fat clumps to the right **–** This is otherwise known as the *minimum visual pollution theorem*. Chances are a sinewave that "looks" nice will "be" nice. Force each clump of ones to be progressively wider towards the right.

Sum those integers **–** For a given number of clumps *k* and a desired number of ones *n*, only certain combinations of sums-of-integers are possible. Especially when further reduced by the minimum visual pollution theorem.

Two amplitudes for each bit **–** For a given number of ones in a quadrant, there appear to be *two* minimum distortion amplitudes. One can be found by fitting something around $n(1\text{-sin}) + .07$. And a second at $n(1\text{-sin}) - 0.43$.

Shake the box **–** Accurately fitting a sine function is a real good start. But, you'll also want to minimize your *total integrated error* function over the quadrant. Being "right on" each time still lets small errors pile up. The easiest is to *jitter* or *anneal* your clumps. Position each clump a few bits ahead or behind of ideal and observe what happens. Typically, this generates tens of thousands of new magic sinewaves. Including one or two good ones.

For more information

The sidebar above lists some magic sinewave resources for you. I have posted reprints of all my previous stories, working tools, magic sinewave lists, and detailed spectral analysis to my www.tinaja.com.

Full consulting and development services are definitely available. Both through my *Synergetics* and independently from Jim Fitzsimons. Source code, prototype hardware, and co-developer programs are also available. ❖

Microcomputer pioneer and guru Don Lancaster is the author of 33 books and countless articles. Don maintains a US technical helpline you'll find at (520) 428-4073, besides offering all his own books, reprints and various services.

Don has a free new catalog crammed full of his latest insider secrets waiting for you. Your best calling times are 8-5 weekdays, Mountain Standard Time.

Don is also the webmaster of www.tinaja.com where a special area has been set aside for Midnight Engineering readers. You will also find selected reprints of Don's other columns, that Synergetics Consultant's Network plus extensive annotated web site links here.

You can also find Don at his Synergetics, Box 809, Thatcher, AZ 85552. Or email don@tinaja.com