Ray Mack, W5IFS

17060 Conway Springs Ct, Austin, TX 78717; w5ifs@arrl.net

SDR: Simplified

In this issue, we will look at some of the fundamentals of sampling theory and how the theory works in practical situations. First, let's look at the results of our lab from last month.

Lab 1 Results

There are a few changes necessary to the schematic diagram of Figure 5 in the first column, in the Jan/Feb 2009 issue.[1] First, I missed that pin 6 appears twice on the op-amp when I proofread my copy. The inverting input on the top op-amp (U2A) should be pin 2. Second, the circuit operates better if R3 and R4 are 10 kΩ instead of 100 Ω.

Figure 1 shows the oscilloscope output of the circuit as I presented it in the last issue. The top trace of the scope plot is the modulated waveform from the sound card and the bottom trace is the demodulated signal. Once the values are tweaked for optimum performance, you see three levels in the bottom trace. You see distinct levels for the one and the zero, with a blip for the dc level in between. This blip can be used to synchronize the data bits in this particular modulation scheme.

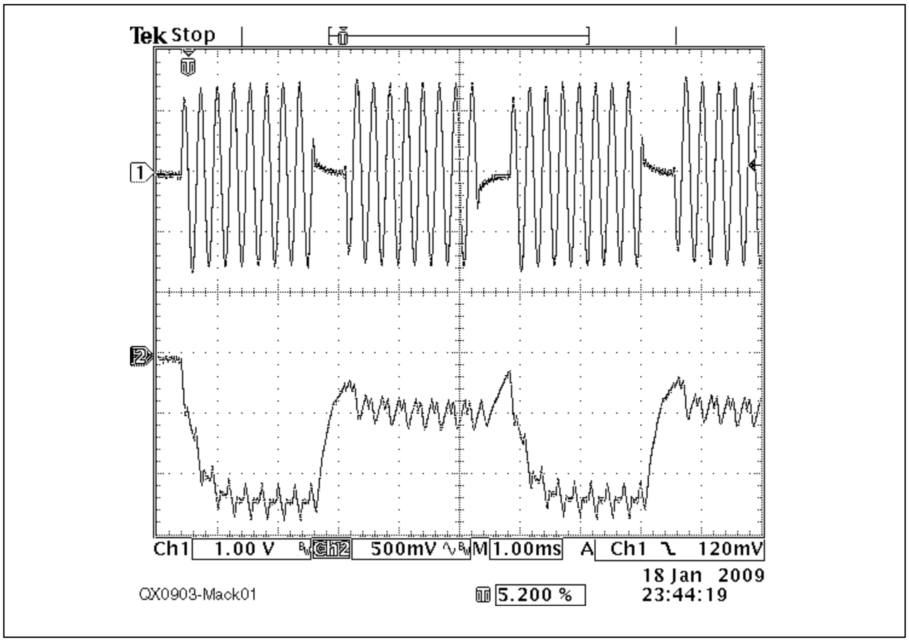

Figure 1 — The top trace of this oscilloscope plot is the modulated waveform from the sound card and the bottom trace is the demodulated signal.

Time Domain and Frequency Domain

There are two ways to look at electrical signals. The first way we learned was to measure voltage or current versus time. Oscilloscopes measure voltage versus time and chart recorders measure current versus time. This is a time domain representation. The second way to look at signals is voltage versus frequency. Panadapters and spectrum analyzers show voltage versus frequency. This is a frequency domain representation. Both time domain and frequency domain tools are used with digital signal processing.

The Nyquist Criterion

One of the most common applications of digital signal processing involves sampling a continuous analog signal using an analog to digital converter, processing the samples using a digital computer, and converting the samples back into a new continuous analog signal. Both conversion processes can lose information if not done correctly. In the analog world, we call loss of information "distortion."

The Nyquist criterion describes what is necessary so that information is not lost in the initial analog to digital conversion pro-

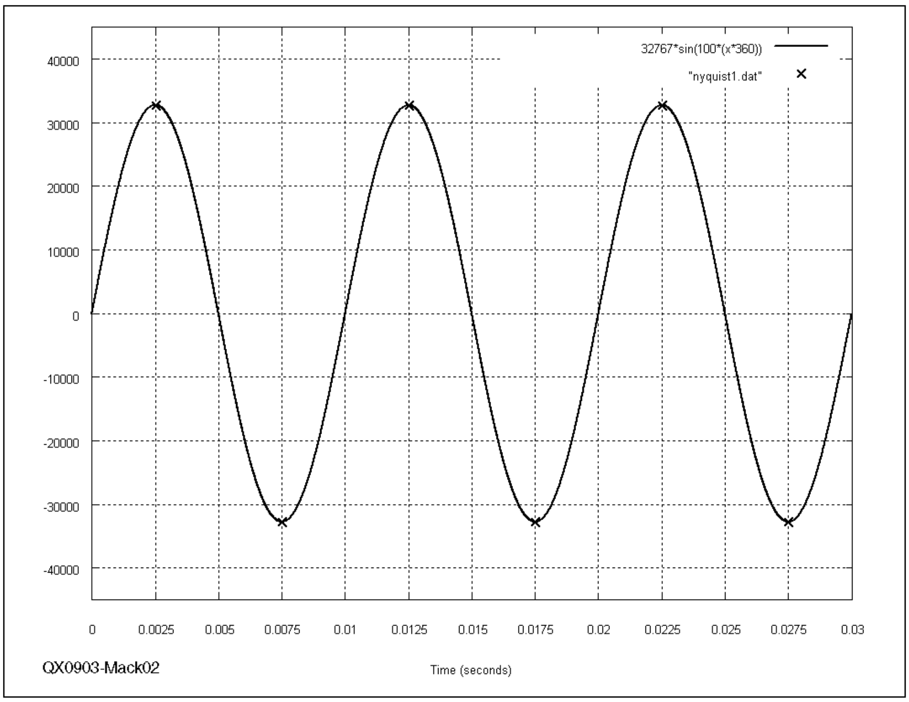

Figure 2 — This sine wave represents a 100 Hz signal. The x marks at the crests and troughs represent one set of sampling points when the sampling rate is 200 samples per second (200 Hz). These samples can define a 100 Hz sine wave with an amplitude equal to the original signal — the ideal situation.

[1]Notes appear on page 44.

cess. Nyquist is probably the best known and most misquoted of the foundations of digital signal processing. Nyquist states that a signal must be sampled at a rate ***greater than*** twice the highest frequency component of the input information. It is totally incorrect to say that it must be greater than ***or equal***, which is how many authors describe Nyquist. In fact, practical systems require the sampling frequency to be much more than twice the highest input frequency. The Nyquist criterion also states that a signal that is sampled at a rate greater than twice the highest input frequency can be completely reconstructed as an analog signal without loss of information. There are some impractical assumptions in this second half of the Nyquist criterion, however.

Let's look at an example of why the "equal to" condition is just plain wrong. We'll take a 100 Hz sine wave and sample it 200 times per second. We can do this manually by simply plugging in values to the formula for a sine function and evaluating for a sequence of times that are 5 milliseconds apart. We will look at three different cases, however, for the position of our first sample relative to where the sine function begins. Our first case samples the sine function at times 2.5 ms, 7.5 ms, 12.5 ms, 17.5 ms, 22.5 ms and 27.5 ms. Figure 2 is a *Gnuplot* of the input sine and the samples. You can see that you capture all of the maximum and minimum values of the sine function, so you should be able to exactly reconstruct the waveform with the proper filtering. This reconstructed sine will have the input frequency of 100 Hz and have the exact same phase. It will also have a peak amplitude equal to the value of the samples.

The second sample sequence is for times 0, 5 ms, 10 ms, 15 ms, 20 ms, 25 ms and 30 ms. You can see from Figure 3 that this sample sequence of the very same sine function yields a sample result indistinguishable from a dc value of zero! This result is called aliasing. Aliasing is the name of the effect in which sampling converts one frequency to a lower frequency. In this case, 100 Hz is converted to zero hertz.

Our third sample sequence samples the sine function at 1.25 ms, 6.25 ms, 11.25 ms, 16.25 ms, 21.25 ms, and 26.25 ms. Figure 4 shows the sample sequence. You can reconstruct this signal and create a 100 Hz sine with proper filtering, but notice that you will get a sine wave that has an amplitude only 0.707 of the original, and the result will be out of phase with the original by 45°.

So, from now on we will be certain to sample our input signals at some frequency greater than two times the highest input signal. Let's look at a 10 kHz sine wave, and see what happens if we sample it at a rate of 22,050 samples per second. Figure 5 is an oscilloscope plot of the 10 kHz signal sampled at 22.05 kHz and played by our sound card. The sample rate is 10% above

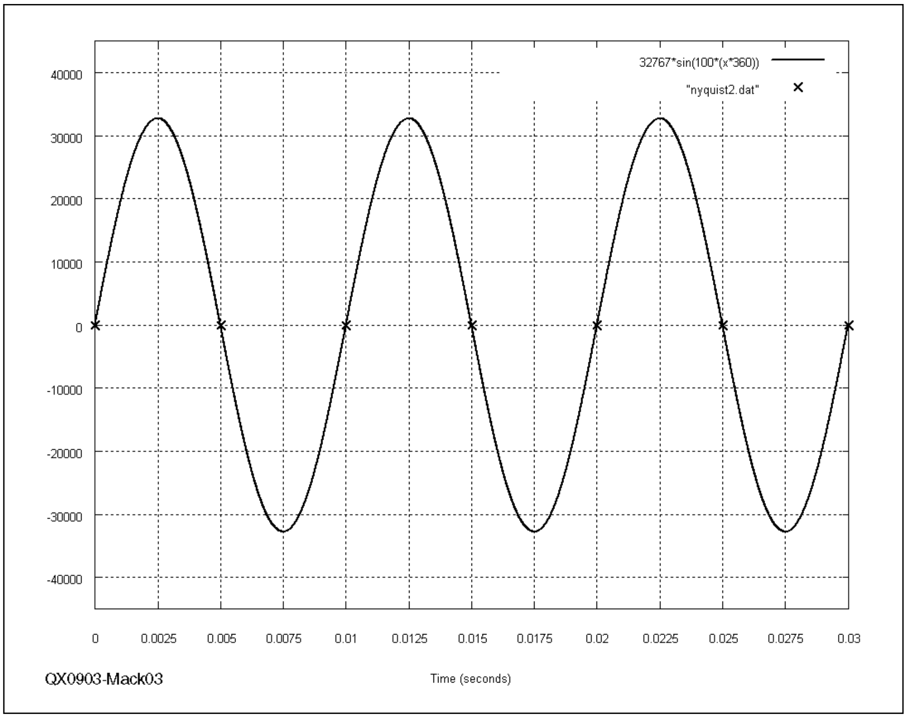

Figure 3 — In this oscilloscope plot, the same 100 Hz signal from Figure 2 is again sampled at a rate of 200 Hz, but this time the samples are taken to match the zero crossing points. This time the samples appear to define a dc signal with zero amplitude.

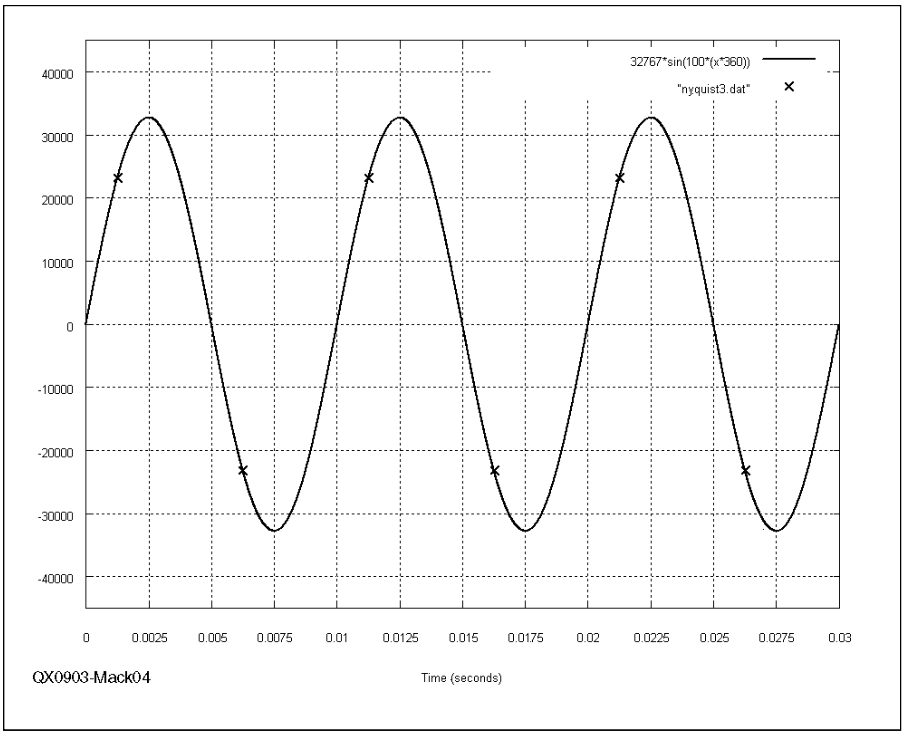

Figure 4 — Here the 100 Hz sine wave is sampled at a rate of 200 Hz, with the samples selected to fall 45° before the crests and troughs. With proper filtering you can create a 100 Hz sine wave from these samples, but the resulting signal will have a maximum amplitude of only 0.707 times the maximum amplitude of the original sine wave. In addition, the new signal will be shifted 45° in phase.

the required amount. You will notice that the signal created from the samples is not a clean sine wave. The difference from a true sine wave illustrates a failure to follow the second requirement of Nyquist: sampling at a rate a little above two times the highest input frequency requires a brick wall filter.

A brick wall filter is an ideal (and physically impossible) filter that has a passband response that is an exactly rectangular shape (Figure 6). The filter on my Dell laptop is nowhere near a brick wall filter. The next two waveforms (Figure 7) are the output of my HP computer for a 10 kHz signal sampled at 22.05 kHz and 20 kHz sampled at 44.1 kHz. Notice that once the initial ramp occurs, then the output signal is a very good sine wave at 10 kHz and 20 kHz respectively. Even though the filter is almost a brick wall, the difference is enough that the first eleven cycles are not a true reconstruction of the input data.

All of the issues we have looked at with respect to Nyquist presume that you are interested in digitizing a signal that spans from dc all the way up to some sample rate less than the Nyquist limit. As we move to looking at RF signals and how to modulate them or demodulate them, we will take advantage of properties of bandpass and other filters, mathematics of sampling, and sample rates that are chosen to keep far enough away from the desired signals so reasonable filters will do the job.

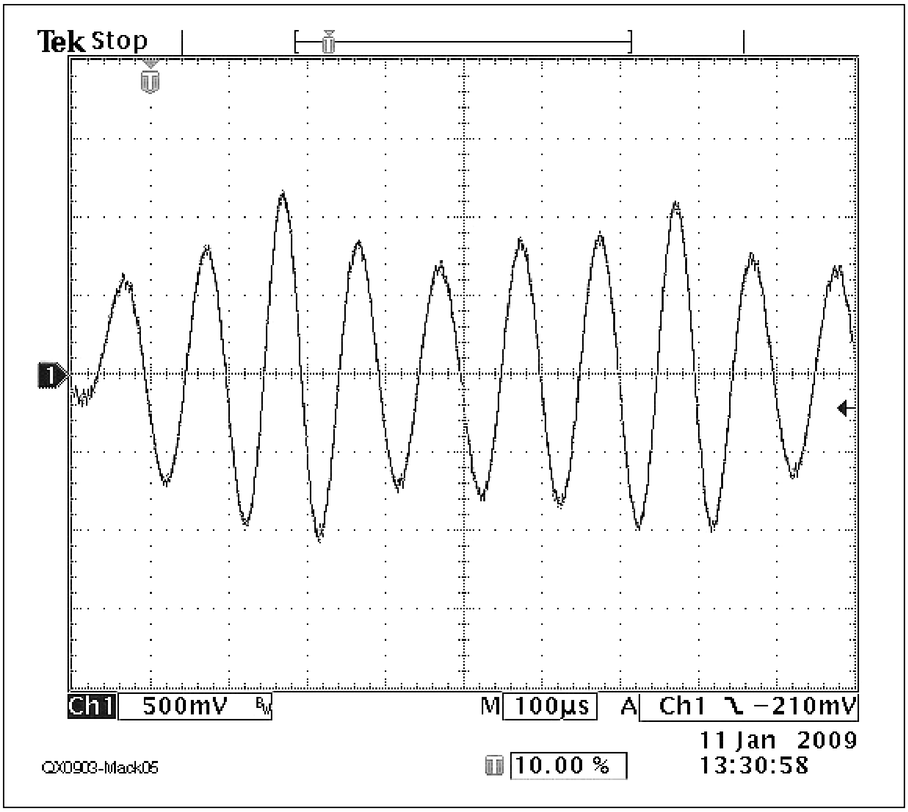

Figure 5 — This oscilloscope plot shows a 10 kHz sine wave that was sampled at 22.05 kHz and played through the sound card on my Dell laptop computer. The reconstructed signal isn't a very good sine wave. The filter on this sound card is not close to a "brick wall" filter.

Fourier and Negative Frequencies

The Fourier transform and negative frequencies are two concepts where the "real world" doesn't seem to square with the mathematics we use. The Fourier transform converts a continuous signal in the time domain to a different continuous function in the frequency domain. Fourier actually discovered two sets of functions. The one we will use for filtering and spectrum display is the Fourier transform. It applies to continuous and non-repetitive waveforms. The other is the Fourier Series, which describes how to create a periodic waveform by adding up a sequence of cosine and sine waveforms that are harmonically related.

We need the concept of I and Q again when dealing with the Fourier transform. Mathematicians call a signal that contains both I and Q a complex signal. That is because they play games with the imaginary value ***i*** (square root of –1) in order to use complex mathematics to generate values that have both magnitude and phase. Electronics folks use ***j*** instead if ***i***, but it is the same thing. We saw in the last column that complex math is really as simple as creating two electronic signals that are a sine wave and a cosine wave, so there is no real magic involved. A Fourier transform takes a complex input in the time domain (I and Q waveforms) and transforms them into a new complex waveform in the frequency domain. This new complex waveform is also just an in phase and quadrature set of data. The combination of the transformed I and Q signals can be used to generate a pair of plots that either show magnitude and phase or we can work directly with just the I and Q components.

The math functions for cosine and sine start at a time of negative infinity and go to positive infinity. The functions that do a Fourier transform also operate from negative infinity to positive infinity. The choice of a real world "zero" time is arbitrary for both the trig functions and the Fourier functions. When we measure a signal with an oscilloscope, we usually place our "zero" time as the left edge of the display and measure time as a positive value relative to that left edge. It is equally reasonable, however, to place "zero" at the middle of the screen and measure both positive and negative time for the data on the screen. If our oscilloscope is set for 1 second per division, our view is limited to –5 seconds up to +5 seconds. All of our digital signal processing will work with signals with positive and negative times and limited amounts of data. We use positive and negative times because all of the functions require the numbers to be symmetrical about "zero." When we do a Fourier transform on signals that cover time before and after zero, we generate a frequency domain representation that contains both positive and negative frequencies.

Now is a good time to look at two more important trig identities:

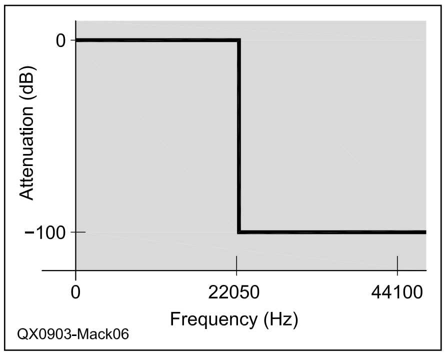

Figure 6 — This graph represents the transition between the pass band and stop band for an ideal "brick wall" Nyquist filter for a 44.10 kHz sample rate.

$\sin(-x) = -\sin(x) = \sin(x + 180°)$ and $\cos(-x) = \cos(x)$

What these two identities mean for us is that a sine wave of –10 Hz (based on our reference cosine wave from last issue) is identical to a sine wave of 10 Hz with a 180° phase shift. The cosine wave has identical phase for both negative and positive frequencies.

The analog image reject mixer uses the property of negative frequencies and a couple of phase shifters to add one set of frequencies and cancel the other set. The Weaver method of single sideband

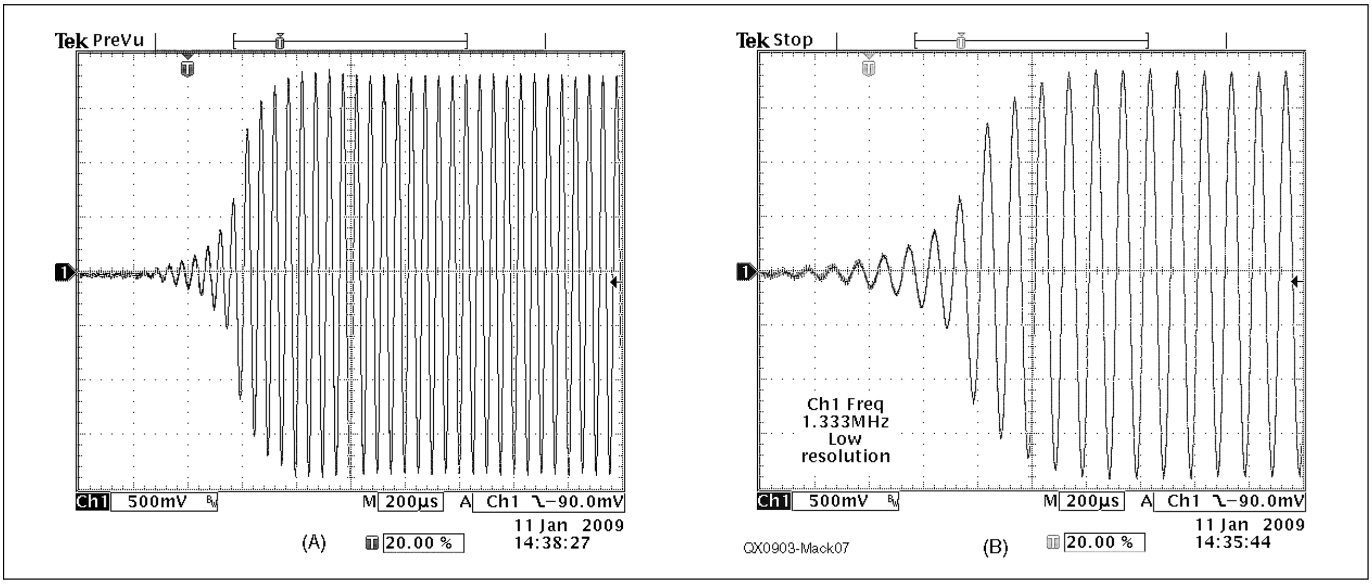

Figure 7 — The oscilloscope plot at A is the output signal from a 10 kHz signal sampled at 22.05 kHz, as processed on my HP computer, with a sound card that provides a much better approximation of a brick wall filter. The plot at B is the output signal from a 20 kHz signal sampled at 44.1 kHz, and played through the sound card in my HP computer.

generation also uses the phase differences between positive and negative frequencies to perform its functions.[2]

You are probably familiar with the Fourier series for a square wave, which consists of the fundamental frequency and all of the odd harmonics. The concept of "zero" changes the Fourier series, however. Here is what I learned as the classic Fourier series for a 50% duty cycle square wave:

$$F(t) = \sin(x) + 1/3 \sin(3x) + 1/5 \sin(5x) + 1/7 \sin(7x) + 1/9 \sin(9x) + \ldots$$

Figure 8 shows a plot of the resulting waveform. Let's look at a similar Fourier series that I first saw in a recent *QEX* article that has all of the same frequencies with the same amplitudes but with different phases:

$$F(t) = \cos(x) - 1/3 \cos(3x) + 1/5 \cos(5x) - 1/7 \cos(7x) + 1/9 \cos(9x) \ldots$$

Figure 9 shows a plot of the resulting waveform for this series. We see that it is the same 50% duty cycle square wave, but with zero in the middle of the high phase. These waveforms are a good example of how changes of where we place zero in the time domain can affect how the frequency domain is presented.

Next Time

In the next column we will start working in earnest with a real application of the Nyquist criterion and Fourier transforms. We will look at how Fourier and Nyquist can be used for an under sampling receiver to receive the time standard WWVB at 60 kHz, using a sound card sampling at 48 kHz. We will also go through the steps necessary to set up your computer to use the Blackfin Stamp product.

Since I started working on this column, DigiKey has decided not to stock the AD7476-DBRD board. Analog Devices still makes the board and has some in stock, so you can order a board directly from their Web site. You will need to set up an account. Start from the main page at **www.analog.com/** and select "Buy Online" in the upper right corner. Select "Place Credit Card Order Now" and select "No I am a New User." You will need to fill out the information to register with the site and set your password. Then you can select "Buy Online" again, and place an order for the AD7476-DBRD.

Notes

[1]Ray Mack, W5IFS, "SDR: Simplified," Jan/Feb 2009 *QEX*, pp 53 – 56.

[2]For information about the Weaver method of SSB signal generation, see the Wikipedia entry at **http://en.wikipedia.org/wiki/Single-sideband_modulation**.

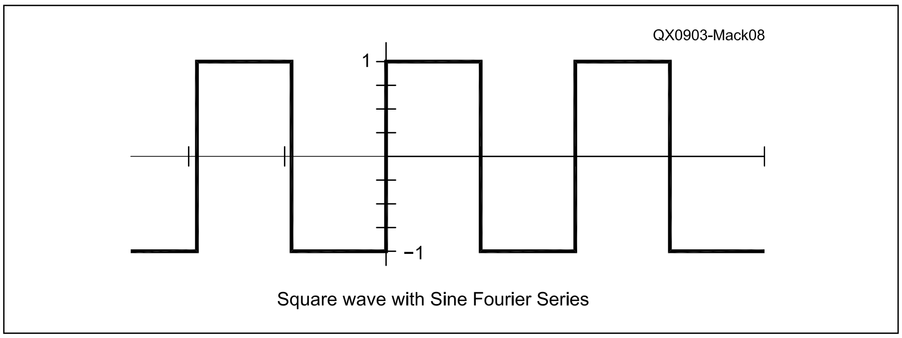

Figure 8 — This is the square wave signal that results from a sine Fourier series. Note the 180° phase shift around the zero frequency axis.

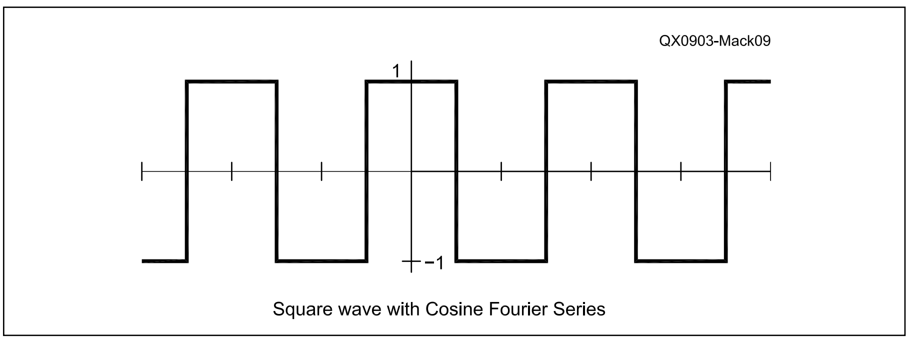

Figure 9 — This is the square wave signal that results from a cosine Fourier series. Note the 0° phase shift around the zero frequency axis.