Digitally Controlled Sine and Square Wave Generation

AN2086

By: Jerry Wasinger
webmaster@web-ee.com
Associated Project: Yes
Associated Part Family: CY8C25xxx, CY8C26xxx

Summary
This Application Note describes a simple method of generating sine waves over four decades from a square wave input. Furthermore, a method of interfacing and reading a digital encoder is described which allows the period of the sine wave to be controlled.

Introduction
According to Fourier Analysis, there is an infinite number of sine waves inside of the humble square wave. For example, a square wave of frequency *f* can be expressed in a Fourier series as Equation (1):

$$\frac{4}{\pi}\sum_{n=1,3,5...}\frac{1}{n}\sin(2\pi nft) \quad \textbf{(1)}$$

Expanded, Equation (1) becomes Equation (2):

$$\frac{4}{\pi}\sin(2\pi ft)+\frac{4}{3\pi}\sin(6\pi ft)+\frac{4}{5\pi}\sin(10\pi ft)+... \quad \textbf{(2)}$$

If we set *f* to 1 kHz and plot the first 11 terms of the above, we would have the waveform shown in Figure 1:

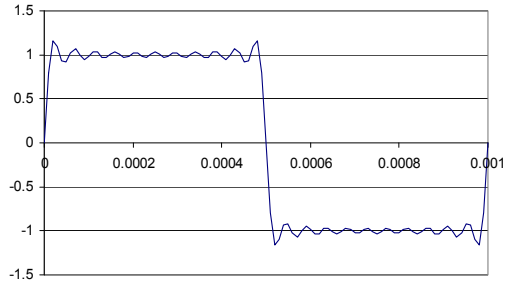

Figure 1: Waveform with *f* Set to 1000

The above is the addition of a 1 kHz sine wave with increasingly smaller sine waves that have frequencies of 3 kHz, 5 kHz, 7 kHz, and so on. Even after 11 terms, we have a pretty jagged waveform.

To generate a sine wave from a square wave, we are only interested in the first term of the Fourier series, which is also called the fundamental frequency. If we could isolate that term of the equation, we could produce a pure sine wave with amplitude of 4/π.

BPF2 to the Rescue
One method of isolating a single frequency component from a series of harmonics is to use a bandpass filter. Figure 2 shows a bandpass filter centered at 1 kHz with a Q of 4. Superimposed on this graph are points representing the amplitudes of the first four terms of our Fourier series:

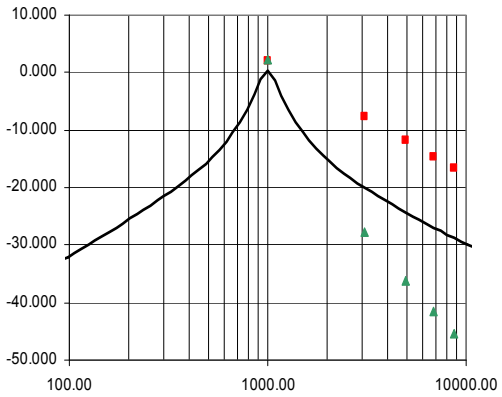

Figure 2: BPF with Four Terms of Fourier Series

The red squares are the amplitudes, in dB of the sine wave coefficients required to generate a square wave. The green triangles are the same amplitudes after the attenuation of the bandpass filter. We see from this that the first harmonic is about 30 dB below the fundamental. The waveform resulting from this process will be what we want, a sine wave.

Implementation

The bandpass filter in the PSoC Designer software has the nice property that allows the center frequency to be controlled by the sampling clock. The Q of the filter remains fixed regardless of the sampling frequency. By using a single clock to drive the filter-sampling rate and to generate the square wave input, we can build a digitally controlled sine wave generator. A block diagram of this approach appears in Figure 3:

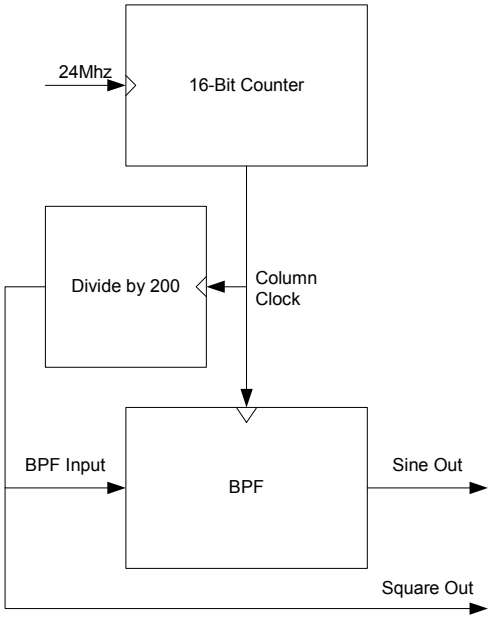

Figure 3: Block Diagram of Digitally Controlled Sine Wave Generator

The schematic of our circuit is shown in Figure 5. The only required component besides the PSoC chip is the digital encoder used to control the counter period. The PSoC chip resource utilization can be seen in Figure 6. The design was implemented using a 16-bit counter, an 8 bit counter, a programmable gain amplifier, and a pair of bandpass filters.

Using the BPF2 User Module data sheet, a filter centered at 1 kHz with a **Q** of 4 and an over sampling rate of about 50 was designed. The clock driving the filters has to be four times this sampling rate so the 16-bit counter is used to generate the 200 kHz signal. To generate our 1 kHz square wave, an 8-bit counter (Counter8_1), fed by the output of the 16-bit counter, was set up to divide by 200. The compare value was chosen to provide a square wave output.

The output of the Counter8_1 was connected to P0[0]. This, in turn, was externally connected to the input of the PGA amplifier at pin P0[3]. To avoid saturation of the filters, the gain of the PGA was set to 0.75. The PGA output is then fed to the input of the first filter, which is then followed by the second filter. The sine wave output is made available at pin P0[5].

Changing the value of the period register in the Counter16 User Module will produce a proportional change in the period of the output waveforms. The output frequency can be found by using Equation (3):

$$f = \frac{120{,}000}{p} \quad \textbf{(3)}$$

The variable *p* is the value stored in the period register. By using an over-sampling ratio of 50:1, the theoretical frequency range of this design is 2 Hz to 20 kHz or four decades. However, due to switch-cap droop errors, the filters may not operate well below 200 Hz. Going above 20 kHz exceeds the maximum clock rate of the BPF2 stages. The lower limit is set by the maximum value of the period register. A log-log chart showing the relationship between output frequency and period is shown in Figure 7.

Taking Control

To provide a simple means of changing the output frequency, a digital encoder is connected to Port 1. The encoder is a pair of switches that open and close in quadrature as the knob is rotated. Referring to Figure 4, we see that clockwise rotation produces a pair of waveforms with Channel A leading Channel B. Channel A lags Channel B with counterclockwise rotation.

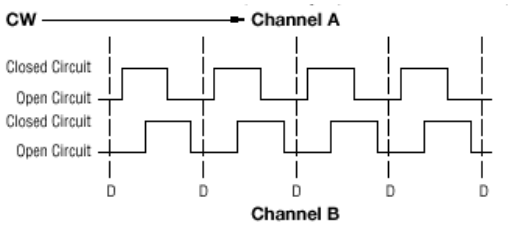

Figure 4: Waveform from Employing Digital Encoder

The built-in pull ups of the PSoC chip were used to minimize parts' count and pin P1[0] was set to interrupt on the falling edge.

Upon interrupt, Counter8_3 is used to provide about a 500 usec delay to suppress switch bounce. P1[0] and P1[2] are then sampled. If P1[0] is not low then we assume that the interrupt was caused by switch bounce. If P1[0] is low then we sample P1[2]. If P1[2] is high then rotation is CCW and we increment the period value. If P1[2] is low, rotation is CW which leads to a decrease in the period value.

Further Refinements

If you want to minimize board space, this design could be completely contained in the 8-pin variant of the PSoC chip (CY8C25122).

There's no reason that an encoder has to be used as the input mechanism. Another device or microprocessor could be used to provide the quadrature waveforms. Or, an RS232 block could be implemented in the PSoC chip that, with some additional software, could provide very fast updates to the period register.

Finally, the precision of the system could be better regulated with the addition of an external crystal to provide the system clock.

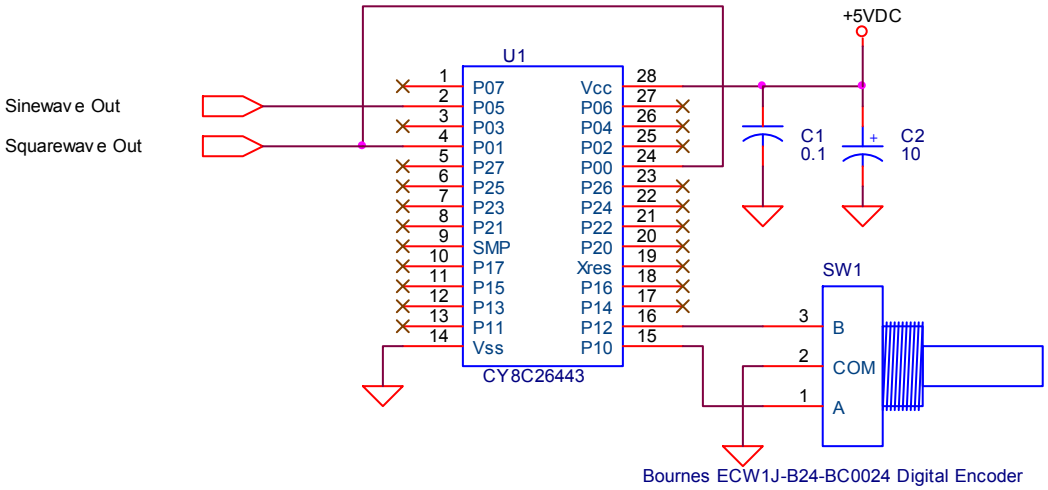

Figure 5: Schematic for Digitally Controlled Sine and Square Wave Generation

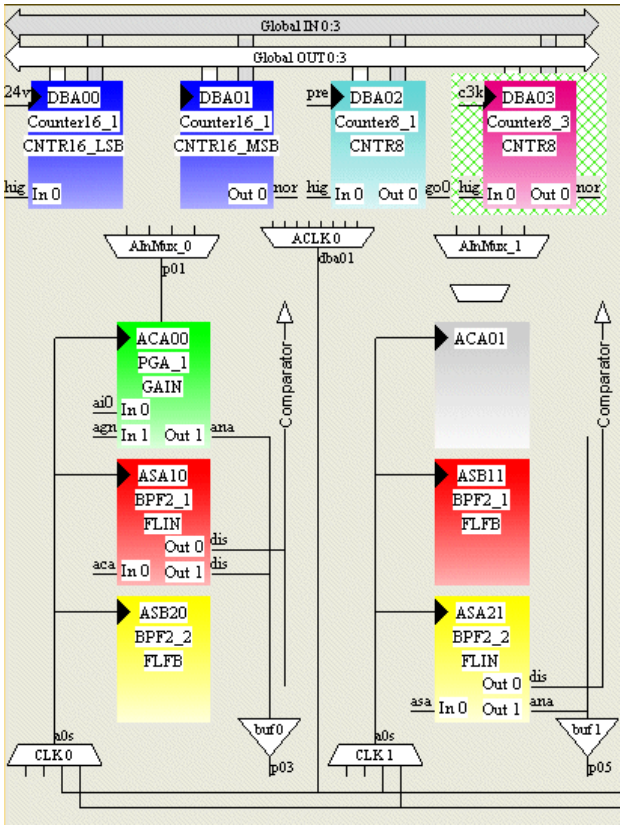

Figure 6: User Module Placement/Resources

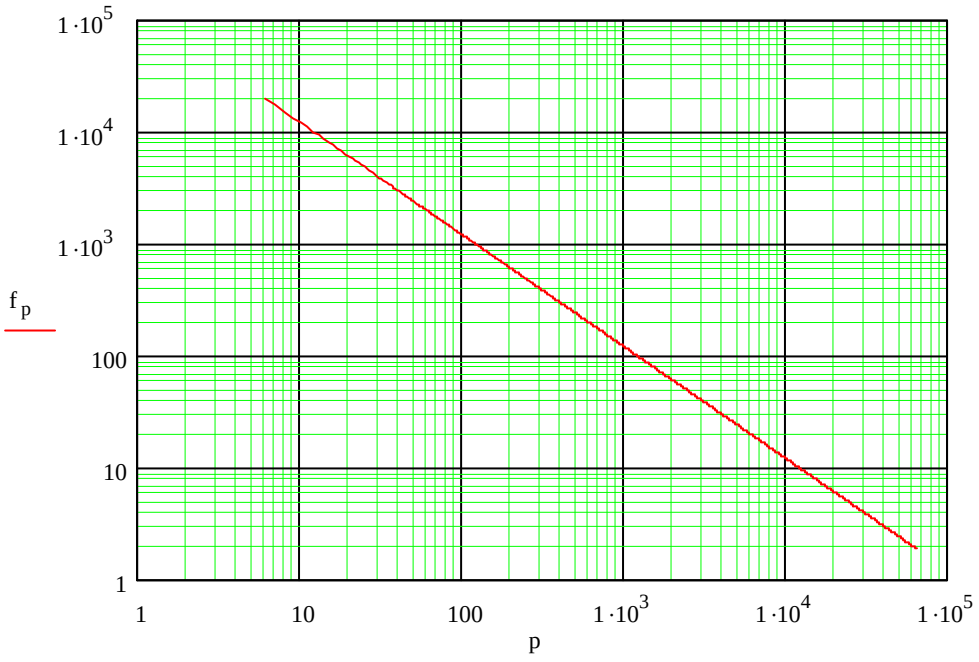

Figure 7: Relationship Between Output Frequency and Period

About the Author

Name:	Jerry Wasinger
Title:	Director, Electrical Engineering at SpectRx, Inc.
Background:	Born and raised in Oklahoma. Spent two years as an art major before changing to EE. Received BSEE from University of Oklahoma in 1978. Received MSEE from Colorado State University in 1982. Designed instrumentation for Hewlett Packard for 7 years. Designed Telecom equipment for Hayes Microcomputer Products for 9 years. Designed Medical Instrumentation for SpectRx, Inc. for 8 years. Run a web site targeted at EE's called WebEE (www.web-ee.com)
Contact:	208.728.6804 webmaster@web-ee.com

Cypress MicroSystems, Inc.
22027 17th Avenue S.E. Suite 201
Bothell, WA 98021
Phone: 877.751.6100
Fax: 425.939.0999
http://www.cypressmicro.com/ / http://www.cypress.com/aboutus/sales_locations.cfm / support@cypressmicro.com