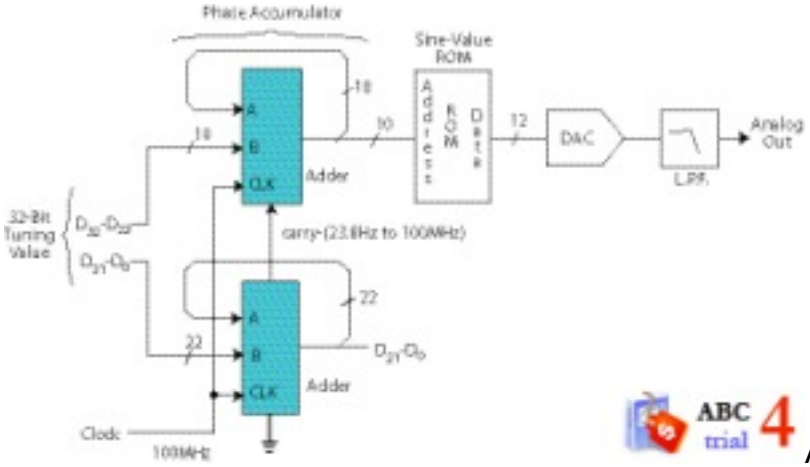
Cover Story: DDS ICs and IP Make Waves

Jon Titus, Senior Technical Editor

DDS ICs and IP Make Waves

by Jon Titus, Senior Technical Editor

You can think of direct digital synthesis as a technique that lets a digital value control the frequency of a sine wave. At its simplest, a DDS circuit involves a binary counter, a ROM programmed with equally spaced sine values for one full wave, and a digital-to-analog converter to convert the stored sine values to voltages. The frequency of the counter's clock determines the sine-wave frequency, but that's an inflexible arrangement.



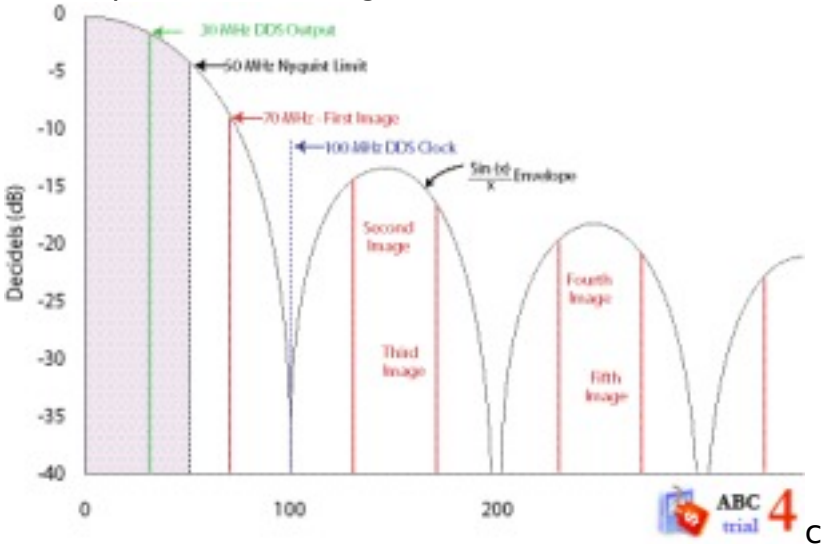
A better approach replaces the counter with a binary adder, often called a phase accumulator, which sums an external tuning value and the previous result (Figure 1). A counter increments by one for each clock pulse, but the adder can sequence through every second, third, fourth...32nd, 131st, or other sine value up to the maximum value the adder can accept. You might ask, "Doesn't this technique skip though sine values?" Yes, but at regular intervals. Remember the Nyquist criteria: You can produce a useful sine wave with only a few samples. An appropriate reconstruction, or recon, filter at the DAC's output completes the job.

You can buy a variety of DDS ICs that include adders, control circuits and a DAC

that produce a sine wave or sine and cosine waves--ideal for I/Q communication schemes. DDS devices may use a sine-value look-up table or an algorithm to produce sine values. Don't worry if you plan to work with an FPGA. Vendors offer tools and IP that implement DDS functions for you. (See For Further Reading below for articles that describe DDS techniques and operations.)

Because a DDS device works with sampled data, designers must take into account the Nyquist sampling theorem. Consider a DDS circuit clocked at 100 MHz and programmed to produce a 30 MHz sine wave. This circuit has a Nyquist limit of 50 MHz for the output signal, but 40 MHz would be a more realistic upper limit.

In addition to the 30-MHz fundamental frequency though, the DAC output includes signals at 70 MHz (100-30), 130 MHz (100 + 30), 170 MHz (2*100-30), and so on. The amplitude of these unwanted "images" follows a standard sin(x)/x, or sinc, envelope or roll-off (Figure 2). Thus, DDS



circuits include a low-pass filter after the DAC to remove these unwanted signal components. Some designs, though, make use of one of these "image" signals. A radio receiver, for example, might filter out the 30-MHz fundamental and use the 70-MHz image signal in an IF stage. (The DDS IC can vary the frequency of that signal, too.) A close look at the sin(x)/x envelope also shows how the amplitude of the fundamental frequency decreases as it approaches the Nyquist limit.

The characteristics of the output signal depend very much on the fidelity of the reference clock signal that drives a DDS IC. Thus, any jitter--or phase noise--in the reference-clock signal will degrade the signal output. All clock circuits exhibit some jitter, but devices produced specifically for signal-synthesis can markedly reduce it, as can careful attention to clock-circuit design. Analog Devices, for example, offers clock-generator ICs that reduce jitter as low as 300 fsec.

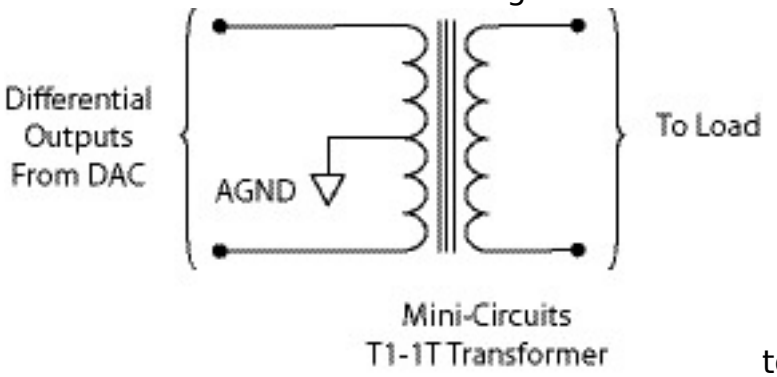
Designers also must consider the spurious-free dynamic range (SFDR) of the signal they want to generate. So when engineers look for spurs in an output signal, they must look beyond the spurs caused by the DDS circuitry itself. Switch-mode power supplies, for example, can create spurs. If you change the digital tuning value of a DDS circuit, spurs related to the DDS circuits will shift, too. You'll see spurs from other circuits remain at fixed frequencies.

The reconstruction filter mentioned earlier can also lead to a design "gotcha." According to Jeff Keip, senior product marketing manager at Analog

Devices, a typical recon filter may not suppress spurious signals for more than several octaves above the filter's cutoff frequency. "Because you have a sampled system, you have signal images that in theory will repeat at higher and higher frequencies. A good recon filter must offer a stop band sufficient to attenuate these higher-frequency signals so they don't mix with your fundamental signal."

Keip stresses that engineers must develop a "signal budget" for a DDS-based application to ascertain what signals will appear and at what amplitude. In this way, engineers can plan their filtering, tuning, clocking, and DDS programming strategy to minimize spurious signals while getting the best possible sine wave.

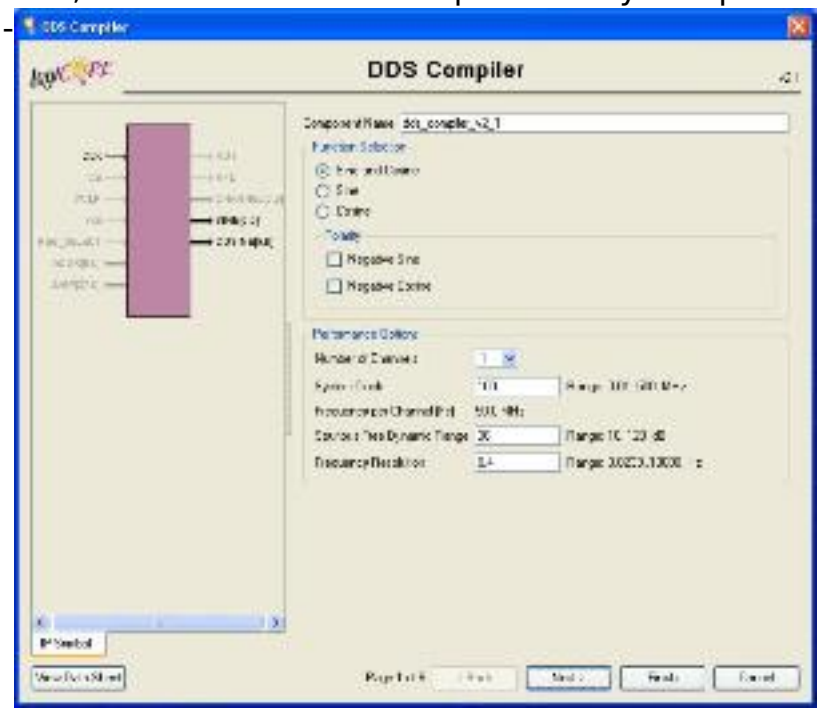
One way to improve signal output involves using an RF transformer on the output of the DDS IC's internal DAC. Using the differential DAC outputs



to drive a center-tapped (grounded) coil produces a higher signal swing at the transformer's output (Figure 3). A transformer, such as the Mini-Circuits T1-1T+ or T1-1T, also offers a good way to reduce common-mode signals, such as even-order harmonics. Keip explained, "Some designers choose to eliminate the transformer, but they still want to take advantage of the differential outputs. So they implement differential filters. We found the differential-filter arrangement offers no common-mode rejection and that's a big 'gotcha.'"

As noted earlier, you can design the digital portion of a DDS circuit into an FPGA. The DDS Compiler from Xilinx, for example, lets engineers quickly enter design

parameters and determine whether or not an FPGA can handle them and if so, the FPGA resources the DDS circuit will require (Figure 4). Vinay Jitendra Singh, product marketing manager at Xilinx, stressed the DDS Compiler is only one part of the company's digital-signal-



processing IP library, which includes FFT cores, filters, and other functions.

"Engineers can look at the DDS Compiler data sheet," said Singh. "But if they have the Xilinx software, they could start to use CORE Generator, a tool that includes the DDS compiler. Then they can get implementation information based on their specs."

According to Singh, system designers can start with MATLAB and Simulink using Xilinx System Generator for DSP to develop and test the DDS portion of a design. Then they can implement the DDS circuit on a Xilinx FPGA using the DDS Compiler from the same high-level design environment.

Then the DDS Compiler makes decisions such as whether to use a full-wave or a quarter-wave sine/cosine look-up table or to put the table in block or distributed memory. Singh explained, "Later on if a design runs out of block RAMs, the designers can quickly implement the DDS design with distributed memory. And a team can choose the optimization goals they must meet."

You might think that because your FPGA-based design offers enough logic and memory elements for you to add a DDS circuit, you get it at no cost. Not so: You still must add a precision DAC, voltage reference, recon filter, and other components. On the other hand, an FPGA lets you customize a DDS design for specific requirements.

By necessity, this article provides an overview. Review vendor's data sheets, application notes, and reference designs for implementation details. You will find vendors of DDS ICs and IP have enhanced basic DDS operations to remove error sources, improve tuning ranges, and simplify the use of DDS techniques in modern circuits. Vendors now make dithering, Taylor-series interpolation, auxiliary

accumulators and other advanced capabilities available in easy-to-use forms for non signal-processing experts.

For further reading

"A Technical Tutorial on Digital Signal Synthesis," Analog Devices, Inc. 1999. www.analog.com [1].

"DDS Compiler v2.1," Xilinx. www.xilinx.com/support/documentation/ip_documentation/dds_ds558.pdf [2] (product specification and DDS tutorial)

"System Generator for DSP," Xilinx. www.xilinx.com/support/sw_manuals/sysgen_bklist.pdf [3]

Lacoste, Robert,"Direct Digital Synthesis 101," Circuit Cellar, August 2008. pp. 60--69.

Brandon, Dave, "Determining if a Spur is Related to the DDS/DAC or to Some Other Source," AN-927, Analog Devices.

Brandon, Dave, "Direct Digital Synthesizers in Clocking Applications--Time Jitter in Direct Digital Synthesizer-Based Clocking Systems" AN-823. Analog Devices.

Brandon, Dave, and Ken Gentile, "DDS-Based Clock Jitter Performance vs. DAC Reconstruction Filter Performance," AN-837. Analog Devices.

Links:
[1] http://www.analog.com/
[2] http://www.xilinx.com/support/documentation/ip_documentation/dds_ds558.pdf
[3] http://www.xilinx.com/support/sw_manuals/sysgen_bklist.pdf