

Digital Correction For Improved Spectral Response In Signal Generation Systems

George R. Spalding and Randall L. Geiger
Iowa State University

Abstract-Signal generation systems using Direct Digital Synthesis (DDS) techniques have received a great deal of attention recently due to their flexibility, repeatability, and wide range of stable operating frequencies. The spectral accuracy of these systems is limited by the analog components used to convert the signal into continuous-time. This work will demonstrate how the addition of digital signal processing and a feedback loop around the analog signal path can increase the spectral quality.

I. DIRECT DIGITAL SYNTHESIS

Direct Digital Synthesis (DDS) is a signal generation technique where the desired waveform is created in digital format and then converted to a continuous time signal by a data converter and reconstruction filter [1,2]. In many cases, DDS implies the use of a phase accumulator that provides a memory address from the signal frequency input information [3]. In this paper, DDS will be used to describe any system that derives the analog output by first constructing a digital signal.

The limiting factor in most DDS systems is the inability of the analog-to-digital conversion circuitry to maintain the high spectral purity possible in the digital signal [4,5]. This analog "back-end" consists of a Digital-to-Analog Converter (DAC) that transforms the signal from digital to sample-data format, a reconstruction filter which removes the unwanted high-frequency aliased components, and usually some form of compensation filter.

The DAC's spectral performance is determined by both its DC specifications (integral and differential non-linearity, gain and offset errors) as well as its high-frequency, transient or settling characteristics. Both low and high frequency non-linearities can produce new spectral components, harmonic and non-harmonic in nature. In a system trying to produce the highest possible frequencies, it is important to push the update rate of the clock as high as possible. When this is done it is very difficult to keep the non-linear settling (or glitch) from becoming a serious problem. Calibration, which can remove most DC errors, is not effective against error sources such as glitch and code dependent settling. Instead of correcting these errors, the current approach is to try and minimize them during the circuit design stage [6].

Most DAC's, instead of trying to approximate impulses, output rectangular pulses at each clock period that give

the common "staircase" approximation to the signal. The frequency response of this technique, often referred to as the zero-order-hold approach, suffers from what is known as $\sin(x)/x$ distortion [7]. In an effort to correct these errors, an analog compensation block or filter is usually added to the DDS system.

Recently it has been demonstrated that a digital filter can be used to prewarp the DAC input signal to compensate for the $\sin(x)/x$ distortion [8,9]. The drawback to this technique is that it assumes rectangular output pulses, which are only approximated well by DACs whose output has settled for the majority of the clock period. This tends to limit the maximum update rate of the system.

II. DIGITAL PREDISTORTION

The idea behind digital predistortion is to compare the systems analog output spectrum to the desired spectrum, and use the results to change the digital input signal so that the output spectrum more closely resembles the ideal. One way in which this can be done is to resample the analog output signal, convert it into a digital signal, and compare its discrete Fourier transform to that of the digital input. The process can be made more computationally efficient if the spectrums are evaluated with the Fast Fourier Transform (FFT).

One of the main concerns is how to use the spectral comparison to alter the initial digital signal. There are a number of possible algorithms, but one of the simplest is to consider each spectral component as a vector, with corresponding magnitude and phase, and to use vector subtraction at each frequency in the FFT output to find the error vector at that frequency. This error vector is then converted back into the time domain in the form of a digital signal and subtracted from the ideal input signal. The process can then be repeated as necessary to further increase the spectral purity of the output. This iterative process is necessary because there is no way to be sure that the correction signal will completely eliminate the spectral errors, or in fact, that the DAC won't distort the correction signal, thus producing yet more unwanted spectral components. A block diagram of a possible algorithm is shown in fig. 1.

One restriction must be placed on the type of signal that this algorithm will handle efficiently. Many DAC errors, such as differential non-linearity and major-carry glitch,

contribute strong discontinuities and code sequence dependencies to the transfer characteristic. This means that even a slight change in the digital input code could result in a large spectral error. The algorithm will work best if the portion of the signal that the FFT evaluates includes the “Digital Repetition Period” (DRP) or time it takes for the digital signal to start running through the exact same code sequence. For a sign wave, this period will only be the same as the analog output period if the clock frequency is an integer multiple of the signal frequency. For more complex signals, the algorithm will only work at maximum efficiency on the portion of the signal evaluated by the FFT, and if the signal is aperiodic or not coherent with the clock signal, the FFT would have to be as long as the desired duration of those signals. The computation time of FFTs grows exponentially with the number of points, so only a relatively small DRP can be handled.

To better understand the predistortion technique, consider the following example. A signal generation system consists of ideal components except that the DAC has a transfer characteristic described by the equation:

$$y = x + 0.02x^2 \quad (1)$$

where y is the analog output and x is the digital input. If the input is a simple sine wave that is coherent with the clock, prior to correction the output spectrum will be ideal except for a second harmonic and DC offset, both 40dB below the level of the fundamental:

$$x = \cos(\omega t) \quad (2)$$

$$y = \cos(\omega t) + 0.02\cos^2(\omega t) \quad (3)$$

$$y = \cos(\omega t) + 0.01\cos(2\omega t) + 0.01 \quad (4)$$

On the first iteration, the algorithm will attempt to compensate for the distortion by subtracting the unwanted component from the digital input signal:

$$x' = \cos(\omega t) - 0.01\cos(2\omega t) - 0.01 \quad (5)$$

When this new signal passes through the DAC, the output is determined by equation #1 as:

$$y' = \cos(\omega t) - 0.01\cos(2\omega t) - 0.01 + 0.02[\cos(\omega t) + 0.01\cos(2\omega t) + 0.01]^2 \quad (6)$$

Which, to the first order, cancels the distortion. Note that the squaring of the input creates cross products between the fundamental and the correction signal, similar to intermodulation distortion in multi-tone signals. These cross products will figure into the correction signal on the next iteration, assuming that they are of a significant amplitude.

The correction algorithm has two effects. The first is to reduce the magnitude of the spectral distortion, and the second is to push the majority of these components higher in frequency. If there is a limited bandwidth of interest (as there is in most applications) the errors may soon be pushed out of it, but it is clear that as the signals and error sources become more complex, the algorithm will have to iterate many times to drive down the distortion sufficiently. It is also possible that the algorithm will not converge.

Although the above example is very simple, it demonstrates the basic technique, and shows some of its advantages and limitations. The closed loop nature of this technique should allow it to reduce all of the analog distortion components simultaneously, including $\sin(x)/x$ distortion (regardless of shape), magnitude and phase distortion, as well as DC and AC DAC non-linearities. It can even remove perceived delays in the signal due to digital registers and analog time constants.

There are certain error sources, however, that the algorithm cannot fully overcome. Errors due to quantization, the basic limit to the resolution of the DAC, cannot be corrected. Gaps in the transfer characteristic of the DAC that prevent it from spanning the full output range are also a problem, although simulation suggests that the effects of these errors may be reduced by the algorithm. Such gaps are frequently caused by large positive DNL errors, while negative DNL errors - leading to non-monotonicity - do not seem to limit the performance of predistortion. It is a fairly easy task to design a DAC that spans the full output range.

III. SIMULATION RESULTS

In an effort to test the predistortion algorithm, a simulator was written in C. The flow of the program is the same as that shown in fig. 1. The DAC model includes the following error sources:

DC Gain, offset, integral non-linearity, differential non-linearity, quantization.

AC $\sin(x)/x$ distortion, limited algorithmic glitch and settling non-linearities.

Data skew and random errors, simulating timing and component mismatches, have yet to be implemented, and the system has no reconstruction filter. The program also generates various digital input signals, and includes an FFT routine that can handle up to 2048 points.

Fig. 2, 3 and 4 show the simulator’s response to various error sources, while Table 1 summarizes these and other results. In all cases, the bandwidth of interest is 40MHz, the clock rate is 128MHz, the FFT observation window is 10^{-6} seconds, and the error threshold is 10^{-5} ($-100dB$). The DAC output range is $\pm 1.5V$.

IV. ADVANTAGES AND LIMITATIONS

Under the right conditions, digital predistortion has the ability to drastically increase the spectral performance of a DDS signal generation system. In addition to eliminating the need for $\sin(x)/x$ compensation, it also lowers the demands on the DAC and output reconstruction filter, while allowing for increased clock rates without performance degradation. Although the algorithm does not rely on the DAC outputting rectangular pulses, for efficient operation it does assume that the gain of the data conversion is approximately one, which would not be true with certain pulse shapes.

Unfortunately, these advantages come at the cost of significant hardware overhead, because every new signal must be run through the algorithm. The two largest contributors are the FFT computation circuitry and the Analog-to-Digital Converter (ADC) which resamples the analog output. The ADC's accuracy has to be commensurate with that desired by the system, or the loop will converge to the signal that compensates for transfer characteristic errors in both the ADC and the DAC. Such an ADC is a very challenging design problem. The speed of the ADC, however, does not need to be significantly faster than the DAC. If we assume that the reconstruction filter chops off all the signal above $f_s/2$, then the ADC only has to accurately obtain the signal in this limited bandwidth.

Another drawback to digital predistortion is that it only works at maximum efficiency on a limited class of signals, those with relatively small digital repetition periods or of short duration. The algorithms effectiveness on other signals is yet to be explored, but initial research seems to indicate that a portion of the error sources (those that are not responsible for abrupt discontinuities) will still be accounted for.

V. FUTURE RESEARCH

If digital predistortion techniques can be extended effectively to a wider class of signals, its viability in test and signal generation systems would be greatly increased. This could be done by extending the effective FFT window or by generalizing the conclusions drawn from observations of limited duration.

The measurement of the analog output must be further investigated to avoid the necessity of constructing a ADC that is significantly better in resolution, if not speed, than the DAC. Analog spectrum analysis may be an option, and accuracy bootstrapping, which uses an iterative approach to improving the accuracy of both the ADC and DAC, is possible [10].

It is important to explore various DAC designs that could be tailored to the technique to avoid such problems as gaps in the transfer characteristic or strong discontinuities. Also, the algorithm presented here is extremely

simple, and it is clear that more sophisticated routines could be devised to minimize the number of iterations as well as improve the robustness of the technique.

REFERENCES

- [1] J. Gallant, "Devices refine the art of frequency synthesis," in EDN, November 9, 1989, pp. 95.
- [2] M. Thompson, "Low-latency, high-speed numerically controlled oscillator using progression-of-states technique," in JSSC, January 1992, pp. 113-117.
- [3] J. Studders, "Phase accumulator for direct digital synthesis resolves to within 0.7Hz," in Analog Dialogue, vol. 25, no. 1, 1991, pp. 10-11.
- [4] T. H. Higgins, "Analog output system design for a multifunction synthesizer," in Hewlett-Packard Journal, February 1989, pp. 66-69.
- [5] H. Nicholas and H. Samueli, "A 150-MHz direct digital frequency synthesizer in 1.25- μ m CMOS with -90 dBc spurious performance," in ISSCC Dig. Tech. Papers, vol. 34, 1991, pp. 42-43.
- [6] K.C. Hsieh, et al., "A 12-bit 1-Gword/s GaAs digital-to-analog converter system," in JSSC, vol. 22, no. 6, December 1987, pp. 1048-1054.
- [7] R. Gregorian and G. Temes, Analog MOS Integrated Circuits For Signal Processing, John Wiley & Sons, New York, N.Y., 1986.
- [8] H. Samueli, "The design of multiplierless FIR filters for compensating D/A converter frequency response distortion," IEEE CAS Trans., vol. 35, pp. 1064-1066, Aug. 1988.
- [9] T. Lin and H. Samueli, "A 200-MHz CMOS $x/\sin(x)$ digital filter for compensating D/A converter frequency response distortion," in JSSC, vol. 26, no. 9, September 1991, pp. 1278-1285.
- [10] Personal Communication with Eric Soenen of Texas Instruments, May, 1991.

TABLE I Simulation Results

Error Type	Comments	Input Signal	Bits in DAC	FFT Size	Iter	Largest Initial Distortion	Largest Final Distortion
INL	2 nd , 3 rd , 5 th	3MHz @1V	16	128	7	-32.0dB	-101.8dB
DNL	No Gaps	3MHz @1V	16	128	6	-37.2	-100.5dB
Non-Linear Settling	Second Order	3MHz @1V	16	1024	13	-22.0dB	-101.0dB
Combined Errors	Multi-Tone Signal	3M @ 0.5	16	1024	20	-32.2dB	-100.1dB
		7M @ 0.25					
Combined Errors	No Converge	3M @ 0.5	13	1024	N/A	-32.2dB	-85.5dB
		7M @ 0.25					
		12M @ 0.125					

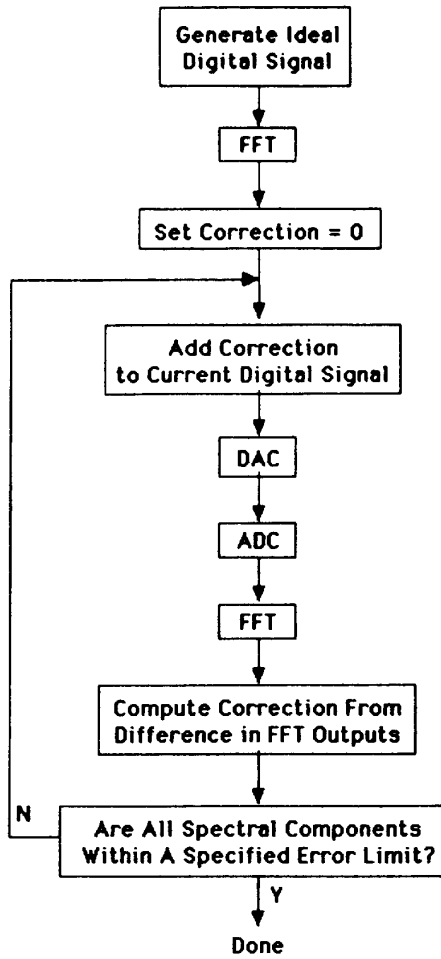


Fig. 1 Block diagram of a predistortion algorithm.

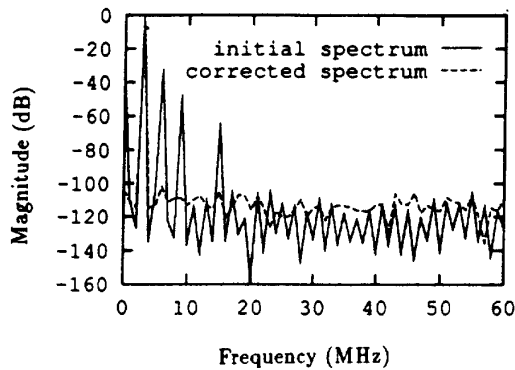


Fig. 4 These spectrums show the ability of the algorithm to handle multi-tone input in the presence of a combination of error sources including INL, DNL, gain, offset, $\sin(x)/x$ distortion, and non-linear settling.

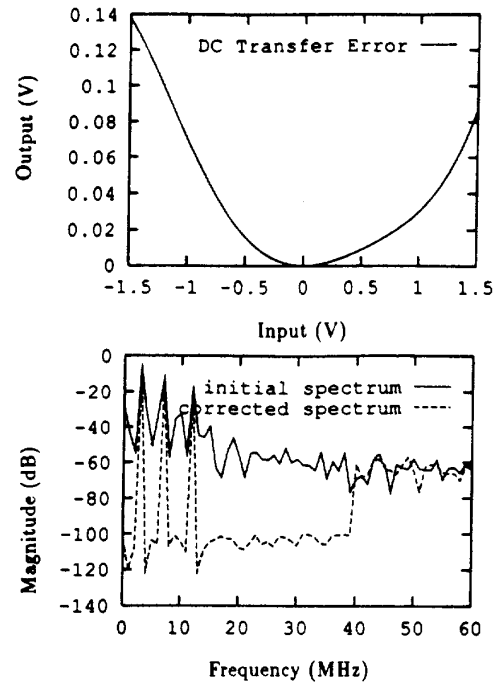


Fig. 2 The INL shown in (a) is composed of 2nd, 3rd, & 5th order terms at 0.05, -0.03, and 0.01 respectively. (b) shows the initial and corrected spectrums.

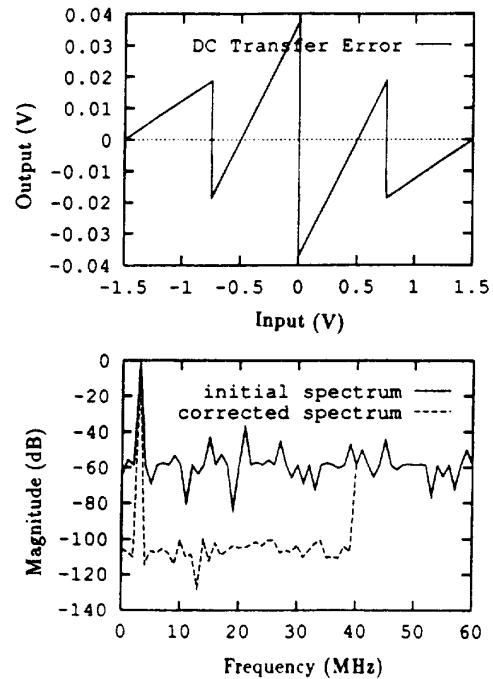


Fig. 3 The DNL shown (a) results in the spectrum (b). Note the effect of limiting the frequency band of interest.