Transfer Function Analysis Side Bar DSP's offer the unique opportunity to provide their own test instrumentation. In particular, the control system transfer function can be computed without expensive test equipment. In order to perform transfer function analysis within the DSP, it's necessary to generate sine and cosine waveforms.

The method presented here uses a single test frequency at a time. An alternative method excites the system with random noise and uses a DFT to compute the transfer function. The signal level at each frequency must be smaller than the single frequency case to keep the system linear; losing signal processing gain. Overall, more accurate results are obtained using a single frequency at a time approach.

The sine wave will be used to excite the control system using the single voltage injection GFT [3] technique. Mathematically, the excitation would be the cosine wave, however, using a sine wave reduces the initial transient. A signal generator is placed at the loop cut point in series with the control flow. The transfer function is then the loop output divided by the loop input. The measurements at each frequency are

re =integral(sig*sin(wt)), in-phase with the signal
im = integral(sig*cos(wt)), the quadrature signal

where the integral is taken over an integer number of cycles. The signal generation is synchronous with the SMPS sample frequency, so it is possible to select an injection frequency and measurement interval such that the start and end value of the signal are equal. That is n*1/f=m*T, or n/m = f*T. The smallest integer n is m*f*T when m is selected an integer. If f is chosen to be an integer multiple of 1/T, the interval will be reasonably small. If however, f is chosen to be an irrational number (perhaps an irrational selection!) then the condition can only be approximated by truncating f to some desired precision. All of these restrictions are used in order to take advantage of a synchronous signal and test frequency. If that assumption is impossible, then the time domain signals can be passed through a window function [4] pg 468.

The measurement interval should start after the transient residues caused by application of the test signal have died out. The signal amplitude must be low enough to be considered a "small" signal; that is, the result will be a linear function of the signal amplitude. If asynchronous noise is present, the measurement interval can be extended to improve the signal to noise ratio.

Generation of the sine and cosine signals can be accomplished in a number of ways. The brute force method is to use a sine-cosine lookup table. To be perfectly general, the table needs to be made for each test frequency and must be m*n in length. One could restrict the test frequencies, as is done for the discrete fourier transform (DFT). Other methods use z transform properties to make the generator. The one chosen here is to use a transform whose impulse response is sin(wt). The z transform can be found in a table [1] pg 743, which is

$$G=z\sin(wt)/(z^2 - 2z\cos wt+1)$$

"www.intusoft.com\DSP\Scripts\singenZ.txt" gives the code that can be used in scope5 or IsSpice4 to generate the signals.

Note: initialization can take place in a lower sample rate loop so that the user can control the data acquisition process, either manually of using an automatic program.