Simulate digital filters with PSpice

Dobromir Dobrev - May 14, 2015

PSpice has become an industry standard tool for analog circuit simulations. Analog circuits have real physical realizations and can be simulated with their schematic representation, and the frequency response is a consequence of the circuit time constants. In contrast, digital filters perform mathematical operations on a sequence of samples. The digital filter’s time constants are hidden in the sampling interval \( T \). So, digital filters are usually simulated only mathematically by their transfer functions, and to do this, it is important to have an easy way of simulating the sampling delay \( T = 1/f_s \), introduced by the sampling rate \( f_s \), because this delay defines and scales the overall filter response.

![Image of a circuit diagram](https://example.com/circuit.png)

Usually, the Laplace transform is used for behavior modeling of analog circuits because it transforms the time domain into the complex frequency s-domain. The frequency response of a digital filter serves a particular case, and can be derived from the Time-shift theorem (Delay theorem) of the Laplace transform. The theorem states that, if a time function \( f(t) \) is delayed by a time \( T \) in the time domain, the result in the frequency domain is a multiplication by \( e^{-sT} \), see Eq. (1).

\[
\text{If } f(t) \xrightarrow{\text{Laplace}} f(s), \text{ then } f(t-T) \xrightarrow{\text{Laplace}} e^{-sT}f(s) \quad (1)
\]

The term \( e^{-sT} \) is often referred to as a delay operator, and if it is replaced by the symbol \( z \), as in Eq. (2), the Laplace transform rises to the so called z-transform. Thus, coming back to the time domain, \( z^n \) corresponds to the delayed \( n^{th} \) sample, \( z^0 \) is the current sample, and \( z^n \) denotes the future \( n^{th} \) sample.

\[
z^n = e^{-sT} \quad (2)
\]

This Design Idea presents a fast, simple, and intuitive approach for digital filter frequency response simulation in PSpice. The PSpice analog behavior modeling symbol library \( abm.slb \) contains the LAPLACE part (Laplace Voltage Controlled Voltage Source), where any s-domain transfer function can be written in its numerator (NUM) and denominator (DENOM) forms. To simulate a z-domain transfer function, first, the sampling interval \( T \) should be defined in the circuit parameter list. Next, the z-domain transfer function should be written in the LAPLACE part by replacing \( z^n \) with Eq. (2). In PSpice, this replacement can be done by defining a function (Eq. (3)), and this function is the heart of the Design Idea.

\[
\text{.FUNC } z(n) \{ \exp(n*s*T) \} \quad (3)
\]
For example, if \( z(-10) \) is written somewhere in the numerator or denominator forms, PSpice will replace \( z(-10) \) with \( e^{10sT} \) (\( s \) is the Laplace variable used by the LAPLACE part). Eq. (3) can be placed in an included file, or more conveniently, in the template of a new schematic part, FUNCTIONS.

The PSpice schematic for AC simulation of a digital moving-average filter is shown in Figure 1. The moving-average filters reject any signals with frequencies in the filter zeros.

For example, let the sampling frequency be \( f_s = 2\text{kHz} \), or \( T = 0.5\text{ms} \). To reject 50Hz power-line (PL) interference, the samples from one period must be averaged. At 2kHz sampling rate, one PL period consists of \( 20\text{ms}/0.5\text{ms} = 40 \) samples. The averager can be simulated directly by a transfer function (output MAV40T), or by a structure close to its real algorithmic realization using delay blocks, gain blocks, and summing and difference junctions (output MAV40R). The simulation has notches at all harmonics of PL interference, as seen in Figure 2.

Figure 1  PSpice schematic for AC line interference averager (comb filter)
The presented approach is fast and easy. I have used it for 10 years, and was inspired to share my knowledge after discovering the Design Idea in Reference 1. Other examples of digital filters simulated with the presented approach can be found in Reference 2. And here are some design files.

References:

1. Lopez D., Transmission lines simulate digital filters in PSpice, EDN, 2008

Also see:
- Use PSpice to model distributed-gap cores
- Inline equations offer hysteresis switch in PSpice
- The power of moving-average digital filters