

ADSL Consortium
DSL Crosstalk Simulation and Calibration



Last Updated: May 11, 2004

*ADSL Consortium
University of New Hampshire
Research Computing Center
InterOperability Laboratory*

*121 Technology Drive, Suite 2
Durham, NH 03824
Phone: +1.603.862.2911
Fax: +1.603.862.4181
www.iol.unh.edu*

DSL Crosstalk Simulation and Calibration

Timothy R. Clark

InterOperability Laboratory, 121 Technology Drive, Suite 2, University of New Hampshire, Durham, NH 03824, USA.

Email: trclark@iol.unh.edu

ABSTRACT:

The performance of DSL transceivers can be impaired by interference from other transceivers operating over wires in the same binder group. It is therefore important to test DSL transceivers in the presence of crosstalk, which can be simulated in a programming environment such as MATLAB. The simulated crosstalk can then be exported to an arbitrary waveform generator (AWG) for injection onto the line through a high-impedance noise injection circuit. This document describes a method of crosstalk simulation and calibration whereby white gaussian noise (WGN) is passed through a frequency-domain filter that gives the noise the spectral characteristics of crosstalk. The crosstalk is calibrated by measuring its power with a vector signal analyzer (VSA) and adjusting the output level on the AWG to achieve the desired noise power. By using MATLAB to estimate the simulated crosstalk power in the same manner as the VSA, the calibration can be performed in MATLAB without taking actual measurements.

KEYWORDS:

Crosstalk, NEXT, FEXT, ADSL, SHDSL, VDSL

1 Crosstalk Simulation

The crosstalk simulation procedure proposed in this paper is shown in Figure 1. White gaussian noise is filtered in the frequency domain to give it the spectral characteristics of the desired crosstalk. This requires multiplying the spectrum of the white noise by the spectrum of the crosstalk, then transforming the result into the time domain.

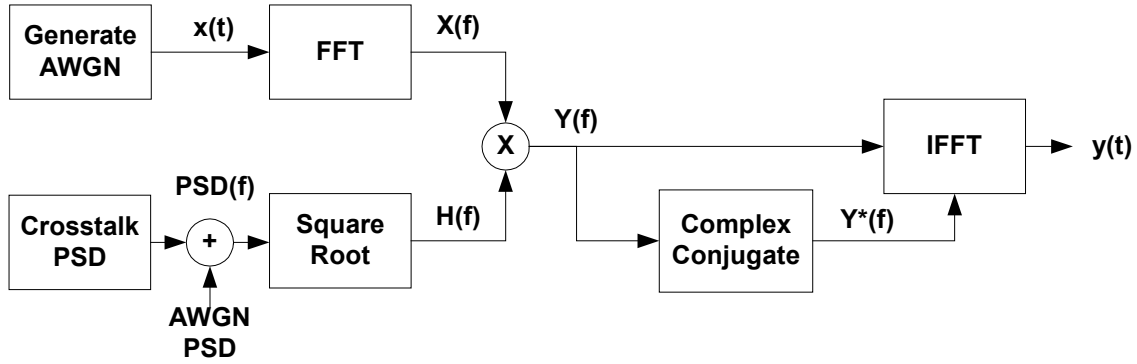


Figure 1 – Block Diagram of Crosstalk Simulation Procedure

1.1 Crest Factor

The first consideration in simulating crosstalk is the number of points N in the output signal. N is limited by the maximum waveform memory of the AWG, but must be large enough to provide WGN with a minimum crest factor. Crest factor (also called peak-to-average ratio) is the ratio of peak amplitude to RMS voltage, calculated as follows:

$$\text{Crest Factor} = \frac{V_{peak}}{V_{RMS}} \quad (1)$$

The crest factor can be thought of as a measure of how “spikey” a signal is. A signal with few but large deviations from the average value will have a high crest factor.

1.2 AWG Memory Limitation

A Tektronix AWG2005 has a maximum waveform memory of 65,535 points. This is not a sufficient number of points to provide gaussian noise with a crest factor of at least 5.0, the required minimum in numerous standards [1], [2]. However, gaussian noise can be statistically altered to produce a “near-gaussian” noise with a crest factor greater than 5.0 with just 65,535 points. The crest factor of the white noise sequence can be increased by selectively altering samples so that the peak value is increased and/or the RMS value is decreased. However, altering selected samples results in an amplitude distribution that is no longer gaussian. Standards bodies usually specify an amplitude distribution mask (see

Figure 2) under which the amplitude distribution must fall for the white noise to be considered “near gaussian.” This procedure can be performed in advance of any filtering, so that the same white noise sequence can be used to produce all crosstalk scenarios.

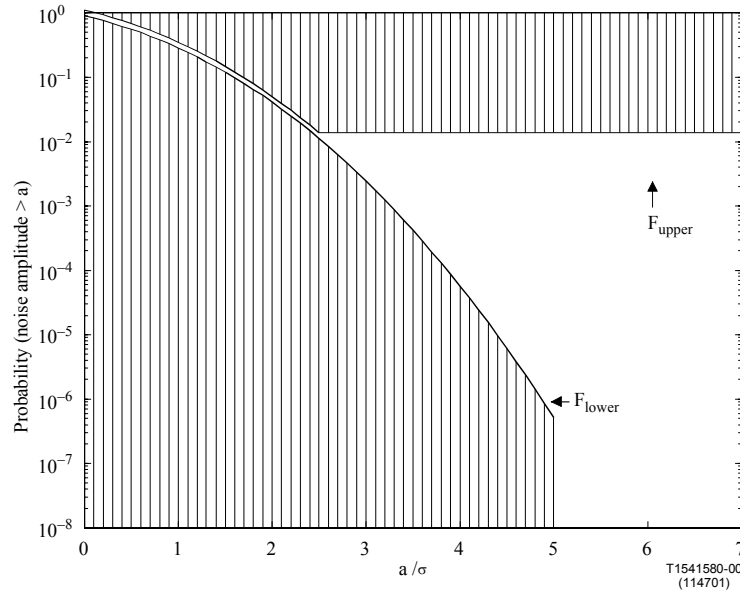


Figure 2 – Amplitude Distribution Mask [1]

1.3 Defining Crosstalk in the Frequency Domain

Once the number of data points has been determined, the frequency vector can be defined. The frequency vector will depend on N , the sampling frequency f_s , and the desired bandwidth f_0 of the output noise. Sampling frequency should be selected to satisfy the Nyquist criterion, i.e. $f_s \geq 2f_0$. If the signal is critically sampled, i.e. $f_s = 2f_0$, the increment of values in the frequency array is then f_s / N . In this case, the lowest value in the frequency array would be f_s / N and it would increase in steps of f_s / N to a maximum value of f_0 .

The next step is to define the crosstalk filter as a function of the frequency vector f . This is most conveniently accomplished by defining a filter transfer function $H(f)$ in terms of the desired power spectral density (PSD). If the input to the filter is a white

noise process with unit variance, then the PSD is equal to the magnitude squared of the filter transfer function [4]. The filter transfer function is then

$$H(f) = \sqrt{PSD(f)} \quad (2),$$

where $PSD(f)$ is the PSD in Watts/Hz. The PSD of crosstalk is modeled by multiplying the transmit PSD of the interfering technology by a crosstalk coupling function. The crosstalk coupling function is used to model how energy from one wire radiates to a nearby wire and is different for near-end crosstalk (NEXT) and far-end crosstalk (FEXT). A NEXT coupling function depends only on frequency and the number of disturbers but a FEXT coupling function is also dependent on the channel transfer function. This is because it originates from the far end of the line and is attenuated by the channel before it reaches the transceiver. For example, n -disturber NEXT/FEXT, as observed from the remote end of the network, can be modeled as

$$PSD(f) = PSD_{Upstream} \times |H_{NEXT}(f, n)|^2 + PSD_{Downstream} \times |H_{FEXT}(f, n, H_{channel}(f), l)|^2 \quad (3),$$

where $PSD_{Upstream}$ and $PSD_{Downstream}$ are the transmit PSDs of the crosstalkers, H_{NEXT} and H_{FEXT} are the crosstalk coupling functions, $H_{channel}(f)$ is the channel transfer function, and l is the coupling path length. The ADSL upstream transmit PSD is multiplied by the magnitude-squared of the NEXT coupling function (and the downstream PSD by the FEXT function) because the point of reference is the remote side of the network. Figure 3 illustrates how NEXT and FEXT appear on the remote end of the network.

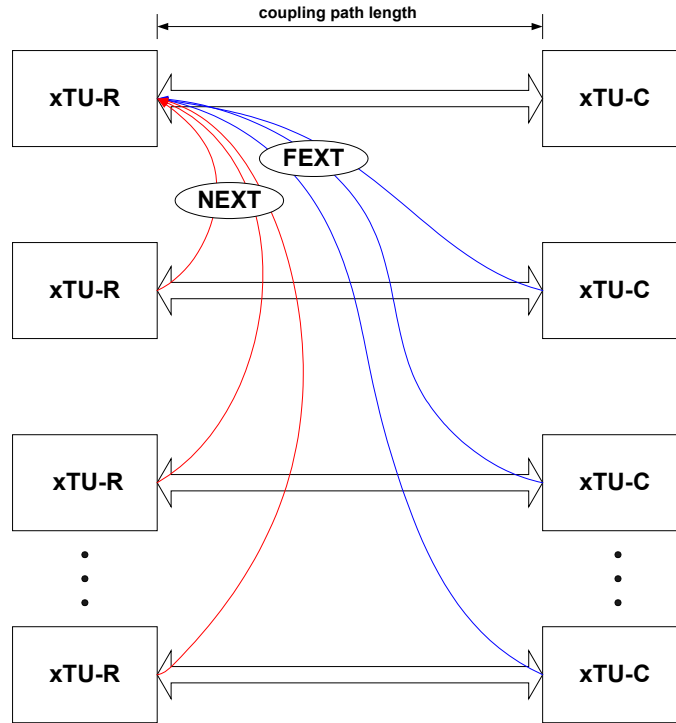


Figure 3 – Crosstalk Coupling

The white noise is represented by $x(t)$. Its one-sided spectrum $X(f)$ is obtained through applying the Fast Fourier Transform (FFT). To apply the filter, $X(f)$ is multiplied by $H(f)$, yielding the filtered spectrum $Y(f)$. The filtered signal $y(t)$ is obtained from $Y(f)$ by applying the Inverse Fast Fourier Transform (IFFT). Since $y(t)$ is to be generated with an AWG, it must be a real signal. If $y(t)$ is to be real, then $Y(f)$ must be a two-sided spectrum with Hermitian symmetry. This means that the magnitude of $Y(f)$ is an even function of f and the phase of $Y(f)$ is an odd function of f , i.e.

$$|Y(f)| = |Y(-f)| \tag{4a}$$

$$\angle Y(f) = -\angle Y(-f) \tag{4b}$$

By appending the mirror image of the one-sided spectrum's complex conjugate, a two-sided spectrum with Hermitian symmetry is obtained. Figure 4 shows the two-sided

spectrum of 24-disturber ADSL NEXT/FEXT as observed from the remote end of the network. Note that the plots satisfy Equation 4.

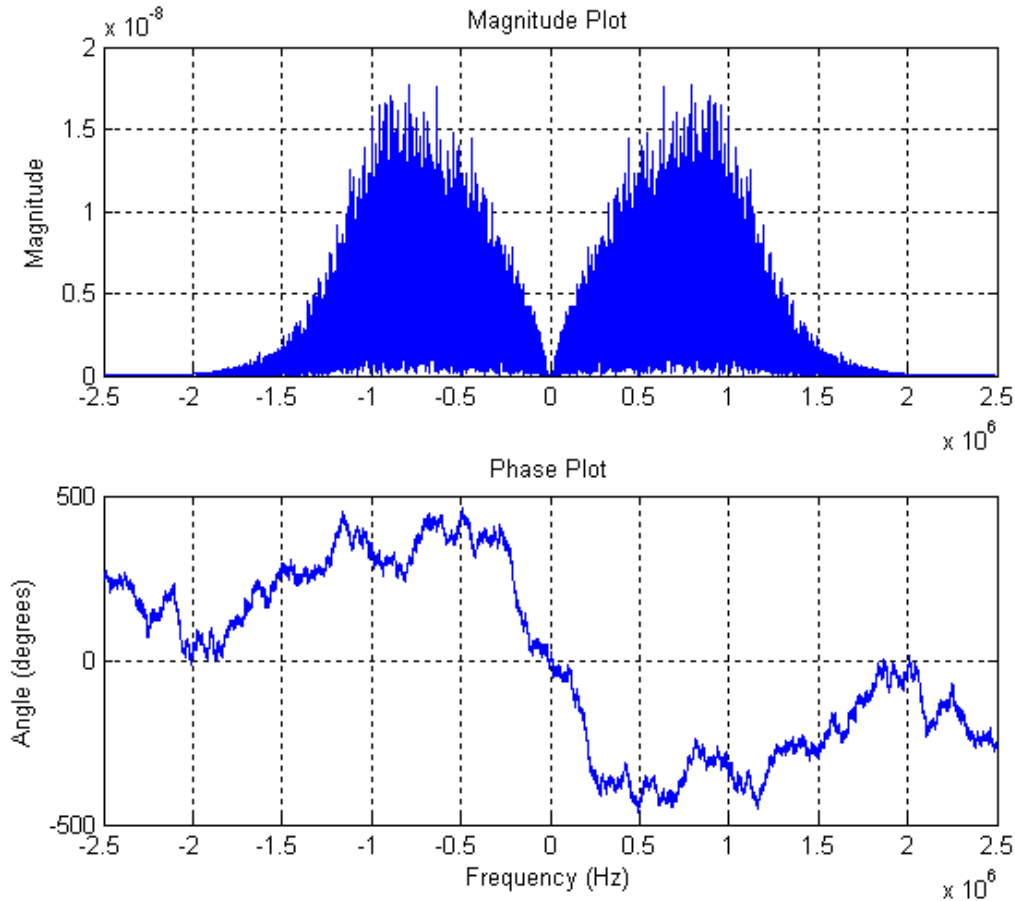


Figure 4 – Hermitian Symmetry

2 Crosstalk Calibration

The procedure outlined above gives the crosstalk the appropriate spectral characteristics but not the right amount of power. Calibrating the crosstalk involves finding the signal amplitude so that the measured power is equal to the desired power. The crosstalk power is measured at the noise injection circuit output, across the calibration termination, as shown in Figure 5.

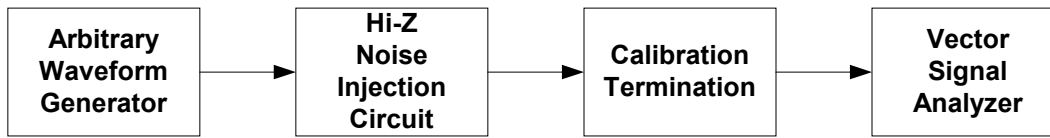


Figure 5 – Block Diagram of Crosstalk Calibration Procedure

2.1 Calibration Termination

The theoretical crosstalk PSD is typically specified as the power delivered to a single-transceiver load. However, since the crosstalk is measured across two transceivers in parallel (see Figure 6), the measured total power needs to be double the theoretical total power, or +3 dB higher. In addition, testing specifications usually require the crosstalk power to be increased by a testing margin, which needs to be added to the total power during calibration. Also, the crosstalk must be measured with sufficiently small resolution bandwidth (RBW) to resolve all of the spectral characteristics.

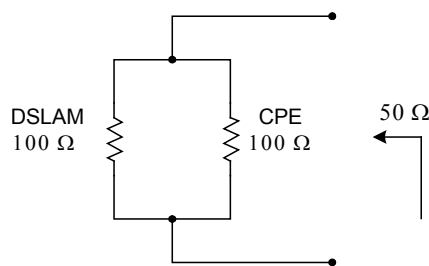


Figure 6 – ADS L Calibration Termination [2]

3 Verifying Simulated Crosstalk

3.1 Resolution Bandwidth

Finally, MATLAB is used to compare the simulated crosstalk with crosstalk measured by the VSA. It is important to reconcile the RBW in MATLAB with that of the VSA, as variations in RBW produce different power measurements. The PSD of the simulated crosstalk can be estimated in MATLAB with a periodogram using the Welch method. Specifying the length and type of window to use in Welch's method can control the RBW of the PSD estimation. The RBW of the periodogram is simply equal to the bandwidth of

the window used. If a Hanning window is used, the bandwidth is equal to 8 divided by the window length, normalized to half the sampling rate [3]. Using Equation 5, an appropriate window length M can be calculated to match the RBW of the VSA.

$$RBW = \frac{8}{M} \cdot \frac{f_s}{2} = 4 \frac{f_s}{M}$$
$$M = 4 \frac{f_s}{RBW} \tag{5}$$

3.2 Insertion Loss of Injection Circuit

The insertion loss of the noise injection circuit also needs to be taken into account if the measured power is to be compared with the simulated power. This can easily be measured by splitting the AWG output into two signals, one of which bypasses the noise injection circuit. Viewing both signals on the VSA simultaneously, the insertion loss is the difference between the total power of the two, and should be uniform with frequency.

3.3 PSD Estimation in MATLAB

The `pwelch` function in MATLAB is used to estimate the PSD of a signal [5]. There are several considerations to take into account to assure an accurate PSD estimate. First, the `pwelch` function must use the `'onesided'` option to specify a one-sided PSD estimate to be compared with the one-sided measurement from VSA. Second, the `pwelch` function estimates the normalized power into 1 Ω . The estimate needs to be divided by the same impedance across which the measurement was taken, the calibration impedance. Third, the insertion loss of the noise injection circuit must be subtracted from the crosstalk power.

3.4 AWG Output Amplitude

Integrating the PSD function yields the signal's total transmitted power. Dividing the total transmitted power of the desired crosstalk by that of the simulated crosstalk yields a factor by which the power must be multiplied to achieve the desired power level. The factor by which the voltage level of the signal must be multiplied is then this power factor

squared. Adjusting the AWG output voltage accordingly, the crosstalk is calibrated without ever having to physically measure the power of the simulated crosstalk.

4 Conclusion

The procedure outlined in this paper allows for the creation of simulated crosstalk with an arbitrary frequency spectrum. This paper addresses most of the requirements for crosstalk-based performance tests as recommended by the various telecommunications standards bodies. This procedure allows the crosstalk to be calibrated without measurement because it uses MATLAB to estimate the PSD using the same method as the measurement equipment. A wide range of crosstalk scenarios can be simulated simply by specifying the frequency-domain filter with the same spectral shape as the crosstalk.

5 Acknowledgements

Matthew Langlois should be acknowledged for his help in writing this paper.

REFERENCES:

- [1] International Telecommunications Union Standardization Sector (ITU-T). Series G: Transmission Systems and Media, Digital Systems and Networks: Digital section and digital line system – Access networks. ITU-T Recommendation G.991.2, *Single-pair high-speed digital subscriber line (SHDSL) transceivers*. February 2001.

- [2] International Telecommunications Union Standardization Sector (ITU-T). Series G: Transmission Systems and Media, Digital Systems and Networks: Digital section and digital line system – Access networks. ITU-T Recommendation G.992.1, *Asymmetric digital subscriber line (ADSL) transceivers*. June 1999.

- [3] Alan V. Oppenheim and Ronald W. Schaffer, *Discrete-Time Signal Processing*, Second Edition, pp. 465-473.

*The University of New Hampshire
InterOperability Laboratory*

[4] Saeed V. Vaseghi, *Advanced Digital Signal Processing and Noise Reduction*, Second Edition, p. 279.

[5] *Using MATLAB Version 6*. The MathWorks, Inc., 2002.