**Proposal to Modify the ISI Penalty calculation in the current GbE Spreadsheet Model**

David Dolfi
Agilent Technologies
April 6, 2000

There are two factors in the current GbE spreadsheet's treatment of the ISI penalty which are mathematical approximations, and to which I will propose corrections to be outlined below. While the corrections are small (of the order of a fraction of a dB in the ISI penalty), there are situations where this degree of error is important.

The ISI penalty couples to other penalties in the model, in particular to the Baseline Wander, so that an error in the ISI penalty can amplify error in other penalties to add further inaccuracies. In addition, it is desirable to limit the ISI penalty itself, irrespective of the magnitude of the other penalties or the total power budget. In the current model, there is a general rule to limit the ISI penalty to less than ~ 3.6 dB. Therefore, a fraction of a dB is important in determining exactly where, in terms of the allowed link length, the ISI penalty reaches this limit.

I want to stress that the inaccuracies I am addressing here are purely mathematical in nature; that is, I am not proposing any new model or physical approach. Therefore, there should be no requirement of experimental verification of these changes.

1. Functional form of ISI penalty -

The current spreadsheet model [1] assumes that the system impulse response can be approximated by a Gaussian of the following form:

$$h(t) = \frac{e^{-\frac{t^2}{2\sigma_c^2}}}{\sqrt{2\pi}\,\sigma_c} \qquad (1)$$

where $\sigma_c$ is the rms width of the impulse response. This rms width is related to the system 10-90 risetime, $T_c$, by:

$$T_c = b_g \sigma_c \qquad (2)$$

where bg $\cong$ 2.563 for a Gaussian response [2]. $T_c$ is the parameter currently used in the model to characterize the total system response. The ISI penalty, $\boldsymbol{P_{ISI}}$, can be written in terms of $\sigma_c$ as [1]:

$$\boldsymbol{P_{ISI}} = \frac{1}{1 - E_m(t_0)} \text{ where}$$

$$E_m(t_0) = 2\left[1 - \int_{-\infty}^{\infty} df \left(\frac{\sin(p f T)}{pf}\right)\left(e^{-\frac{(2pf s_c)^2}{2}}\right) e^{2pift_0}\right] \qquad (3)$$

The time $t_0$ is the time at which the eye is sampled, and T is the baud period. This integration can be performed to yield:

$$\boldsymbol{E}_m(t_0) = 2[1 - h_e(t_0)] \qquad \text{where}$$

$$h_e(t_0) = \frac{1}{2}\left[\operatorname{erf}\left(\frac{2t_0 + T}{\sqrt{8}s_c}\right) - \operatorname{erf}\left(\frac{2t_0 - T}{\sqrt{8}s_c}\right)\right] \qquad (4)$$

where erf (x) represents the error function. Because of the symmetry of the impulse response around the origin, the middle of the eye occurs at the origin, so that $t_0 = 0$. Hence:

$$h_e(t_0) = h_e(0) = \operatorname{erf}\left(\frac{T}{\sqrt{8}s_c}\right) \qquad (5)$$

where we have used the fact that erf is an odd function (erf(x) = -erf(-x)). We can combine (2), (3), (4), and (5) to write:

$$\boldsymbol{P_{ISI}} = \frac{1}{[2h_e(0) - 1]} \qquad \text{where}$$

$$h_e(0) = \operatorname{erf}\left(\frac{b_g T}{\sqrt{8}T_c}\right) \qquad (6)$$

which represents a self-consistent solution to the ISI penalty within the framework of the assumption of a Gaussian system response, embodied in equation (1). The current spreadsheet model uses an approximation to equation (6), given by [1]:

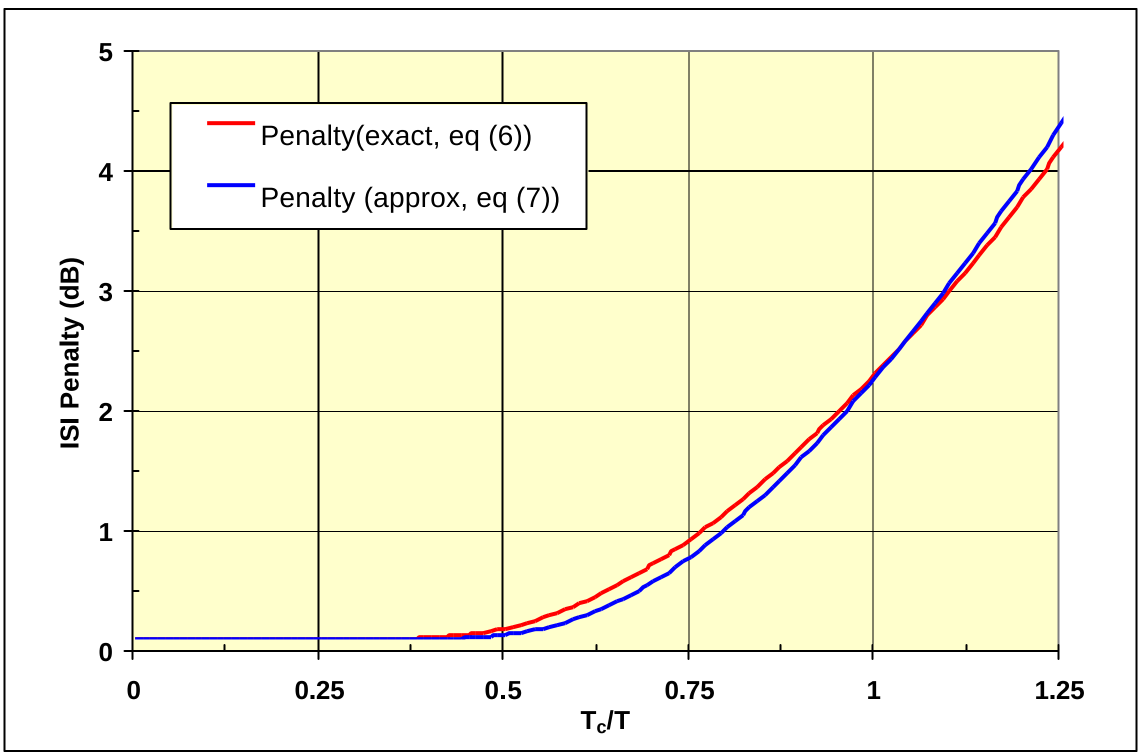
$$\boldsymbol{P_{ISI}} = \left[1 - 1.425\, e^{-1.28\left(\frac{T}{T_c}\right)^2}\right]^{-1} \qquad (7)$$

The ISI penlaty in dB (for either expression) is given by:

$$P_{ISI}(\text{dB}) = 10\log_{10}(P_{ISI}) \qquad (8)$$

The difference between (6) and (7) can be seen on the graph below, where both equations are graphed in dB according to (8) as a function of ($T_c/T$). For small values of this parameter, corresponding to a rapid system risetime relative to the baud period, the approximation (7) is slightly smaller than the expression (6), underestimating the ISI penalty. However, as the system risetime grows relative to the baud period, particularly in the region where the ISI penalty is > 3 dB, the approximation (7) is larger than (6), overestimating the ISI penalty by a non-negligible amount (several tenths of a dB). Since this is precisely the region in which the ISI penalty becomes critical, I would propose using the expression (6) in the spreadsheet rather than the approximation (7) to it.

A potential implementation difficulty, which led to the original use of the approximation (7), is the calculation of the error function. However, this function can be easily calculated in Excel for all versions from Excel 97 onward. The Appendix outlines in detail how this can be done.



2. Correction to Receiver contribution to system response:

The current GbE spreadsheet calculates the rms contribution of the receiver bandwidth to the net system response via the following formula [1]:

$$T_c^2 = c_g^2\left(\frac{1}{BW_m^2}+\frac{1}{BW_{ch}^2}\right)+T_s^2+\left(\frac{0.35}{BW_r}\right)^2 \qquad (7)$$

where $T_c$ is the system 10-90 risetime as before, $BW_m$ and $BW_{ch}$ are the fiber modal and chromatic dispersion bandwidths, respectively, and $T_s$ is the source 10-90 risetime. The factor $BW_r$ is the receiver bandwidth. The factor $c_g$ is the ratio of the bandwidth to the rms impulse response width, which is assumed Gaussian for both the chromatic and modal dispersion. The different factor multiplying the receiver bandwidth $BW_r$ is due to the fact that it is assumed to have a raised cosine (rather than Gaussian) response. If $BW_r = BW_r^{3dB}$, the 3 dB receiver bandwidth, it can easily be shown that this factor is:

$$T_r = \frac{a_{rc}^{3dB}\, b_g}{BW_r^{3dB}} \qquad \text{where}$$

$a_{rc}^{3dB} = s_r BW_r^{3dB}$ is the factor for a raised cosine response which relates the rms response width to the corresponding 3 dB bandwidth, $s_r$ being the rms response width for the receiver.

As before, $b_g$ is the Gaussian factor from (2). The current spreadsheet sets:

$$a_{rc}^{3dB}\, b_g = 0.35$$

for the raised cosine receiver. This factor is incorrect, however. The correct factor for a raised cosine receiver is given by:

$$a_{rc}^{3dB} = .1285$$
$$\rightarrow a_{rc}^{3dB}\, b_g = (.1285)(2.563) = 0.329, \textit{ not } 0.35$$

which is consistent with the corresponding calculation in [2]. While this, like the use of equation (7) above for the ISI penalty, contributes an error of the order of only a fraction of a dB in most cases, it can be an issue when estimating the limits on the ISI penalty or total allowable link length for a particular link type.

As an example, I calculated the ISI penalty for a link of the type 1300 nm on 62.5 MMF, with the following parameters:

wavelength (um): 1.27
Baud rate (Gbd): 3.125
R'cvr BW (MHz): 2500
source linewidth (nm): 0.75
Link length (m): 300
Source rise time (20-80, psec): 100
$T_{DCD\_DJ}$ (psec): 24 psec
Fiber Modal BW (MHz-km): 500

Using the current GbE spreadsheet to calculate the penalties, I obtained an ISI penalty of 3.80 dB. When the two changes above are implemented, this reduced to 3.57 dB, a reduction of 0.23 dB.

I would therefore propose two changes in the current spreadsheet model:

1) That the current approximation to the ISI penalty be replaced with the self-consistent Gaussian penalty.
2) That the correct factor for a raised cosine be used in determining the receiver contribution to the system risetime.

**References:**

[1] David Cunningham, Mark Nowell, and Del Hanson, "Proposed Worst Case Link Model for Optical Physical Media Dependent Specification Development", presented at the IEEE 802.3z Interim Meeting, Jan 27-29, 1997, San Diego, CA. See also, Mark Nowell, David Cunningham, Del Hanson, and Leonid Kazovsky, "Evaluation of Gb/s laser based fibre LAN links: Review of the Gigabit Ethernet model", Optical and Quantum Electronics, 32, pp 169-192, 2000.

[2] Gair D. Brown, "Bandwidth and Rise Time Calculations for Digital Multimode Fiber-Optic Data Links", **JLT**, vol. **10**, no. 5, May 1992, pp. 672-678.

**Appendix:**

Implementation of error function capability in Excel:

These instructions for using the error function [erf (X)] in Excel apply to Excel 97 and Excel 2000. They may apply to earlier versions of Excel as well, but I have only tested them on these two most recent versions of the application.

1. The error function erf (X) of a particular argument X can be calculated in Excel using the syntax:

=SIGN(X)*ERF(0,ABS(X)) (A1)

The sign function SIGN(X) = ± 1, depending on whether X is positive or negative, while ABS(X) is the absolute value of X. This syntax exploits the fact that erf is an odd function. If X has an absolute value greater than ≅ 27 (ie: |X| > 27), the use of (A1) will result in a "#NUM?" error, due to some convergence problems with the calculation. It is highly unlikely that the GbE spreadsheet ISI calculation using (6) will ever result in an argument for the error function of this magnitude, so it should not be an issue for the changes proposed here. For completeness, however, I will note that this problem can be fixed, since, for |X| >> 1, erf (X) = sign(X). Therefore, for arbitrary X, the following, only slightly cumbersome syntax can be used:

=IF(ABS(X)<27,SIGN(X)*ERF(0,ABS(X)),SIGN(X)) (A2)

Equation (A2) will yield the correct value of the error function erf (x) for *any* value of x regardless of its magnitude or sign.

2. If an attempt to use either (A1) or (A2) in Excel results in a "#NAME?" error, it is probable that the error function is not loaded into the program. The error function (as well as Bessel functions and several other special functions) can be loaded into Excel by using the following steps:

a. Under the **Tools** menu at the top of the spreadsheet, select "Add-ins".
b. Under the Add-ins pop-up menu, choose the "Analysis Tool Pack"
c. Click "OK"
d. Close Excel and re-open it. The error function should now be enabled.

Excel should add in this tool pack, which includes the error function, among other things. If Excel cannot find the tool pack, the "Browse" function on the Add-ins menu can be used to find it. However, this should not be necessary. Once the Analysis Tool Pack has been added it will be automatically enabled whenever Excel is launched thereafter. Therefore, once the error function capability is added, it will be a permanent feature of your version of Excel.