EEE 482F LABORATORY 2

Comparison of Inter-Symbol Interference for Digital Transmission

Exercise 1

- Set up a unipolar PN code generator to vary between 0 and 1V at a bit rate of 10Hz. Pass the output of the PN code generator through a raised cosine filter. The filter can be found in the Communications menu in the Linear Filter System library, and the parameters can be left as default to start with. Make sure you select the "parametric filter", since this will allow you to se the alpha factor, bit time, and so on.
- In setting up this simulation, carefully examine the simulation parameters such as time duration, simulation sample rate (and hence time increment) in relation to the waveform that you are actually simulating. Why did you choose the values you did, or, were automatically selected by the system? (Hint: look at the clock token in the standard toolbar.) Be prepared to answer questions on this topic.
- View the PN code and the output of the filter. In the analysis window, view the eye diagram of the output of the filter. This is done using the 'Time Slice' function in the Style library on the calculator. Set the start time to 0 and the increment to 0.1s (the bit period). Make sure the plot is set to "repeat" i.e. you will see many bit times superimposed.
- Adjust the parameters of the raised cosine filter and describe qualitatively what you are seeing in the time domain. Review the theory in the textbook (Chapter 7). Which alpha factor gives the best eye diagram. What criterion do you use to describe a good eye diagram?
- Change your sink viewer to look at the spectrum of the raised cosine filtered signal. Note the jagged spectrum. Explain qualitatively. Try different alpha factors. Now increase the simulation time span by a factor of 10 (i.e. more bits through the system). Look at the new spectrum. Explain what you see, and now vary the alpha factor and observe the spectral width.

Exercise 2

- Using the same PN code generator output as used in the previous exercise, set up a duobinary system (described in Section 9.10 in Stremler). Do not implement the precoding method at this stage.
- Pass the duobinary code through a Gaussian or Butterworth filter with response much wider than needed for the bit stream (what is that?), as well as a filter matched to the bit stream. View the output of the filter and the eye diagram. Compare this to the eye diagram generated in Exercise 1.
- Compare the spectrum of the raised cosine filtered stream with alpha =0 and 1, versus the spectrum of the unconstrained duobinary stream. Comment.

Exercise 3

- Using a bit to symbol token from the Communications Library, convert the PN sequence to an amplitude M-ary code with 4 levels.
- Pass the output through a filter with the same parameters as before. View the eye diagram and compare it to the previous waveforms and spectra.

Call the tutor/demonstrator when you are done and be ready to answer a couple of questions on what you have done.

For enrichment

Have a look at the effect of Gaussian noise on the eye diagrams you have generated above. The noise should, of course, be inserted before the channel filters. You might also want to try some noise with a matched filter. Here, use a simple PN sequence and the corresponding matched filter. Is the output what you expect?