**EE 460 Project FAQ**

**Q1:** What is the oversampling factor, M? I didn't see it anywhere in Chapter 15.

**A1:** M is determined from the baud rate (or period) and the sampling frequency. Both of these are given.

**Q2:** What is the carrier frequency?

**A2:** The signal is transmitted at a carrier frequency (unspecified, and doesn't matter), and then the receiver's analog front-end brings the signal down to an intermediate frequency of $f_{IF}$. Then, the signal is (under)-sampled at a frequency of $f_s$. Obviously, $f_s < f_{IF}$, so we have a situation of sub-Nyquist sampling. The signal that you are given is after this sampling (i.e. $r[k]$ shown in Fig 15.2). You should be able to figure out the "new" effective carrier frequency after aliasing.

**Q3:** How do I pick 'delta' for the equalizer?

**A3:** You should have observed (at least!) one important fact from the homework: If you make the equalizer long enough, it should still open the eye for a range of deltas. On the other hand, if you make the equalizer too long, it may suffer from slow convergence and excess MSE.

**Q4:** I decided to use the Costas loop but when I demodulate the signal, the copy that is supposed to appear at DC disappears!

**A4:** Make sure you're not doing **pre-processing** with the Costas loop. It's not a good idea because it introduces $\pi/2$ ambiguities.

**Q5:** How can I be sure that my carrier recovery algorithm is working?

**A5:** One way to test if your carrier phase tracking is working is this: if you plot the spectrum of your demodulated signal (before any filtering), the portion of the signal at baseband should be twice the height of the signal at twice the effective carrier frequency (or its aliased equivalent).

**Q6:** I want to use a trained equalizer like LMS (i.e. not a blind equalizer like DMA or decision direction). So, I need to perform correlation before the equalizer to obtain the location of the training data. Should I quantize my signal before correlation?

**A6:** Quantizing effectively throws away information, and there's no need to quantize before correlating. For example, if the Gaussian noise added to a -1 PAM symbol causes it to be shifted to +0.1, it would get quantized to +1. However, if it is left unquantized, the uncertainty of the value is incorporated into the correlation (i.e. +0.1 is close to a decision boundary, so it's a less reliable value. . . and doesn't get weighted as much).

**Q7:** How can I be certain that my receiver will work on the 'mystery signal'?

**A7:** I might suggest this sequence of steps:

- Get your receiver working error-free with the easy test vector.
- Change your initial guesses of carrier phase and timing, and confirm that your receiver is robust to poor initializations.
- Get your receiver working error-free with the medium test vector. This step may require significant tuning of your stepsizes.
- Again, confirm that your receiver is not sensitive to initial parameter values.
- After your receiver works error-free with the medium test vector, make sure that it still works with the easy test vector.
- Once your receiver works error-free on the easy and medium test vectors (for any initial parameter values), you are ready to conquer the hard test vector.
- Your receiver may not be able to get completely error-free performance on the hard vector, but it should obtain decent performance (BER $< 0.01$).
- After all tuning and tweaking, confirm that your receiver still works on all three test vectors.

**Q8:** Do I need to do any magnitude scaling in my receiver? It seems the constellation doesn't line up with $\pm 1, \pm 3$?

**A8:** While it's true that the sampled signal may not line up with $\pm 1, \pm 3$, you should not use any *fixed* magnitude scaling (though this might be useful in the beginning during texting). The AGC on the front end of the receiver (Fig 15.2) adjusts the gain of the incoming signal so that it has unit norm. Subsequently, your equalizer should correct for any magnitude scaling that may be necessary.

**Q9:** My receiver works on the "easy" test vector, but not on the "medium". Why?

**A9:** This is a difficult question to answer, but a good place to start is stepsize tuning. Examine the trajectories of your adaptive parameters, and confirm that they look sensible.

**Q10:** My receiver works on the "medium" test vector, but not on "hard". Why?

**A10:** Again, this probably amounts to stepsize tuning. The hard vector is more dynamic (i.e. has faster moving impairments), and so you may need to increase your stepsizes slightly in order to track the time-variation. Also, it is possible that your equalizer is too long. After you get your signal working reasonably on the hard vector, I suggest decreasing the size of the equalizer so long as the performance stays roughly the same. When deciding the equalizer lengths, there is a trade-off in ability to equalize vs. ability to track. A longer equalizer has more parameters to adapt, and thus may have a harder time tracking. An equalizer that is too small will not be able to sufficiently equalizer the channel.

**Q11:** My correlation fails. How can I improve it?

**A11:** While the receiver will be not perform well in general if you attempt block-processing, the correlation is one exception. Assuming your timing algorithm is tracking and doing downsampling appropriately, you can expect preamble to occur with the same regular period. Thus, the correlation is the one place you can "cheat" and do block processing. With a little common sense, you should be able to derive a robust correlation scheme that uses knwoledge of the largest peak to find the first preamble.

**Q12:** I'm not sure if my equalizer is working. How can I check?

**A12:** If carrier recovery and timing seem to be working, and you're not getting a good BER, this may be the case. One thing you can do is plot the squared-error history of the equalizer to see if the error is indeed getting smaller. To do this, you will need to change the equalizer code a bit so that you store the error after each iteration. So the variable e will need to be a vector indexed by the loop counter. After your equalizer has run, you can issue the command: `plot(e.^2)` and see if the error is decreasing.

**Q13:** My receiver seems to be working, but cuts off the last few symbols. Why?

**A13:** There could be several reasons. There is a difference between the `filter` and `conv` command. For a filter of length $N$ and a data sequence of length $M$, the `filter` commands return a sequence of length $M$, whereas `conv` returns a sequence of length $N+M-1$. Thus, `filter` doesn't flush out the delay elements unless you explicitly use two output arguments to obtain the final conditions – stick to `conv`, it's easier. Also, you may need to re-examine your loop indices, as it's possible that your loops are not running long enough. If all else fails, you can always attach zeros (maybe 200 or so?) to the end of the received signal in the first line of your code.