Simulink Tx

Digital Signal Processing Lab

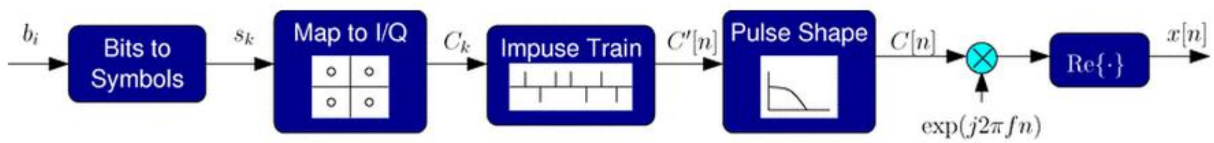
12/2/2013

Jacobs University Bremen
Grishma Raj Pandeya and Joshan Chaudhary

1. Explain in one or two paragraphs what you learned about communications systems that you didn't know before.

We took matlab course with Dr. Oswald last year but the usage of simulink's dsp toolbox was completely a new experience. We learned that we could model a communications system (a transmitter to be precise) using simulink's toolbox. It was really fascinating especially when we got to see how the waves and ……changed when we changed certain parameters in the toolbox and the effects of certain components in the system (give some examples)

Since this experiment was single-link communications system operation we got to learn about the various steps (with what happens in each step) involved in the process. They are explained below.



Above diagram graphically represent the crucial steps involved in the transmitting process. The steps involved are listed below:

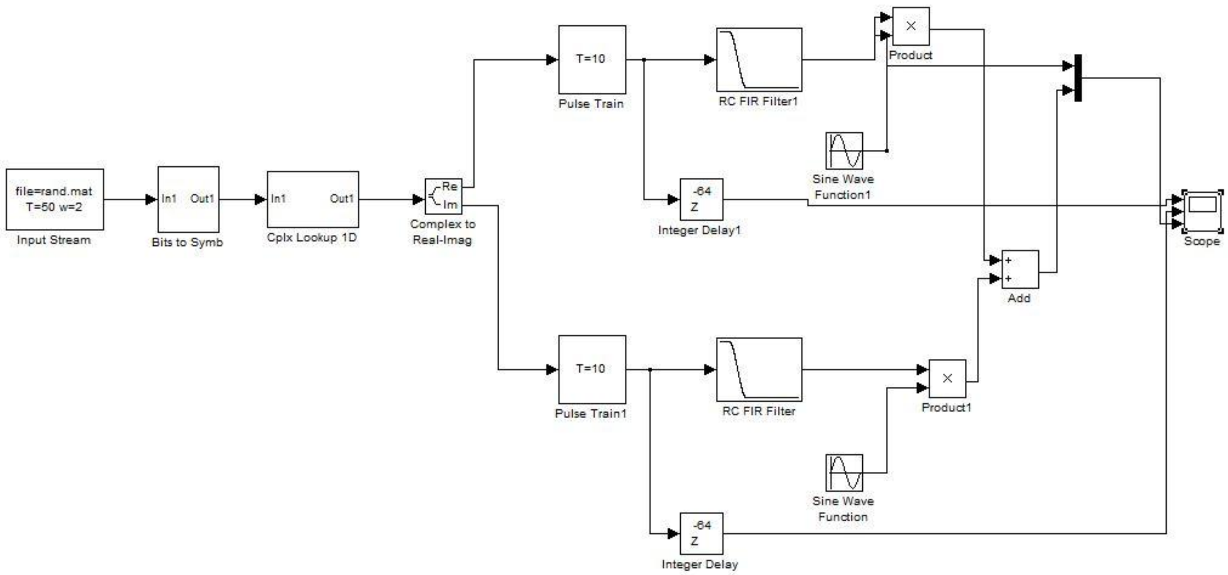
a) Symbol Mapping
b) Constellation (I/Q) mapping
c) Pulse shaping
d) Up-conversion

Given enough SNR (signal to noise ratio), we can actually transmit more than two different signals through a channel and detect which one has been sent to the other side. So, we map stream of bits to symbols. We would require 2^(N) symbols for N bits which are represented as sk, which are basically integers.

Information can be transmitted using both the phase and amplitude of the signal (for most of the channels and systems). There is a convenient way of representing transmitted signals in the complex plane in terms of the I = in-phase (real part) and Q = quadrature (imaginary part) components. What we do here is take each symbol sk and map its corresponding complex baseband value Ck in the complex plane. The next step involved is pulse shaping. We need pulse shaping filters here. In order to keep the bandwidth limited we employ pulse shaping filters to create a desirable shape in a more confined spectrum which still conveys the same information. We don't transmit signals at baseband, we shift the resulting complex baseband waveform C[n] to a higher frequency which is done by the final multiplication block.

2. Show a picture of your final QPSK transmitter. Explain briefly what the different blocks are for. Explain how you checked that it was working correctly. A plot of the output showing important signals would be very helpful.

The picture of our QPSK transmitter is as follows:



The different blocks used and their uses are as follows:

- Input Stream: reads a file named 'rand.mat' which has a stream of 1's and 0's. The bit period is 50 samples and output width is 2.
- Bits to Symbols: maps the bits to constellation diagrams. It takes a vector of N bits and maps them to the indices 0 to 2^N-1 with the least significant bit appearing first.
- Cplx Lookup 1D: plots the complex symbols to the real and imaginary axis as [1 j -1 -j]
- Complex to Real-Imag: splits the complex values into real and imaginary parts which are then treated separately.
- Pulse train: generates a pulse train at the symbol period on both real and imaginary part of the transmitter.
- RC FIR Filter: used for pulse shaping. Low pass filter filters the high frequency components (edges).
- Integer Delay: adds some delay to the pulse trains so that the comparison of the pulse train with the modulated signals becomes easier. Here we delay the pulse train by 64 samples.
- Sine wave function: does the up-conversion. Here, we multiplied the real part of the signal by a cos and the imaginary part by –sin(by adding a phase of pi/2).
- Product: outputs the product of the message signal and the carrier signal.
- Add: adds the real and imaginary parts after up-conversion
- Multiplexer: combines multiple input signals into a vector. Used for signal routing
- Scope: displays the waveforms of the various outputs

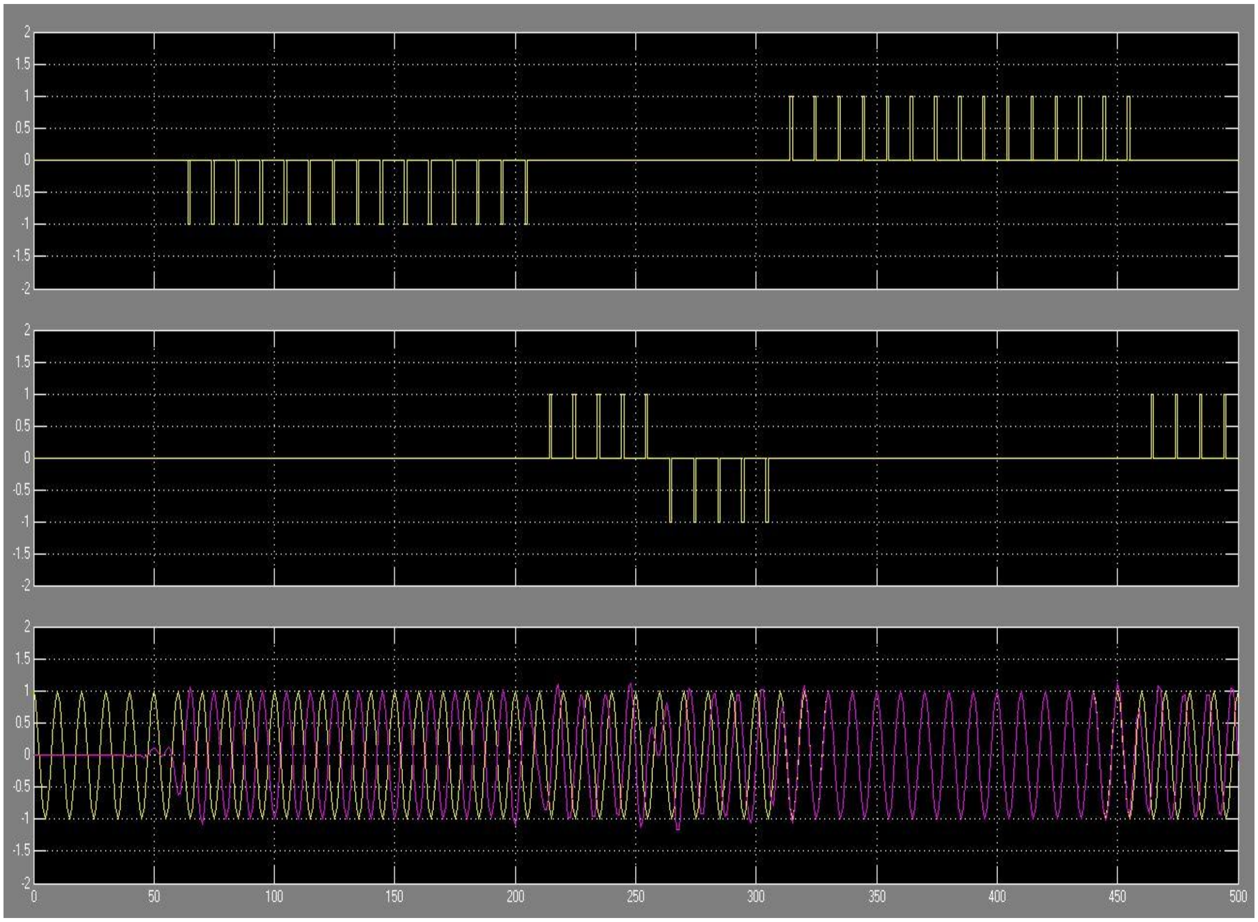


Fig: (from top to bottom) real part of the pulse train, imaginary part of the pulse train and modulated and added signal ready for transmission

We checked that the circuit was working correctly by comparing the modulated signal with the original delayed signal. As seen above in the plot, though it may not seem clear here, when the signal goes to 1 the modulated and carrier signals are in phase but when the signal goes to -1, they are out of phase.

Also, looking at the imaginary part, when it goes to 1 from -1, our modulated and carrier signal goes from leading position to lagging position. Hence, we concluded that the circuit was working fine.

3. Explain why we do pulse-shaping in communications systems, rather than just sending rectangular pulses.

When transmitting rectangular pulses we obtain sync-like spectrum, which is very undesirable for the reason that it is very wide and decays slowly with increasing frequency. But for wireless transmissions where we desire a limited bandwidth such sync like spectrum isn't practical. So, instead of just sending rectangular pulses we do the pulse shaping and then subsequently pass the pulse train through a filter of a desired frequency response of the pulse. This process results a spectrum which is much more confined than the square pulses but essentially carries the same information. So, we do the pulse shaping.

4. Tell why synchronization (e.g. sampling the received signal at the right place) is critical in a communications system that uses pulse shaping.

Synchronization is very crucial in a communications system. Failing to do so will result in the loss of the signal. To successfully receive the transmitted signal, we must sample at the optimal sample points (there should be no inter-symbol-interference).

5. Describe any difficulties you experienced in getting your design to work and how you fixed these problems. I will be really surprised if everything worked perfectly!

As a first time user, we certainly had some difficulties using the toolbox. But with the TAs help we were able to figure it out and navigate through it with a great ease. We had difficulties implementing QPSK because the instructions were not explicit and it was a rather complex design for a beginner. However, TAs helped us with that too. Also, we were not quite sure how to interpret the graphs. With the TAs help we understood everything and got our task done completely. It was a wonderful learning experience.

Reference:

[1] Gentile, K. The care and feeding of digital, pulse-shaping filters

[2] Pulse-Shape Filtering in Communications Systems – Developer Zone – National Instruments

[3] http://www.faculty.jacobs-university.de/jwallace/xwallace/courses/dsp/