**CMPEN 462** **Wireless Communication systems** **Mar 03, 2024**

# Mini-Project #1: Basic OFDM Transmitter

Name: Shiv Patel, PSU ID: sup446

**Number 1:** At first I took my name and concatenated it with my name to get WirelessCommunicationSystemsandSecurityShivPatel and converted it to 8-bit ASCII using a function in Matlab. This step was important because it transforms the text into a binary format that can be processed for transmission in a wireless communication system. After converting the text to ASCII, the binary representation was not long enough to fill the required data for one subframe's worth of OFDM transmit symbols for the 64QAM modulation scheme. To fix this the ASCII values were repeated enough times to meet the required but length which came out to 448 times. This repetition ensured that there was enough data to simulate the transmission over a 1 millisecond duration, which is the standard length of a subframe in LTE communication.

**Number 2:** I took the bit stream that was generated from the ASCII values of my initial phrase and applied the 4 different modulation schemes (BPSL, QPSK, 16QAM and 64QAM) to modulate the data. This modulation process converts the binary data into the correct form to transmit over a communication channel. By using the different modulation types, I prepared multiple versions of the data each with varying levels of spectral efficiency to noise and interference.

**Number 3:** I converted the serialized bit stream into parallel streams for different modulation schemes. This "serial-to-parallel" conversion was important because it aligned the bit stream with the OFDM requirement to modulate multiple subcarriers simultaneously. Each parallel stream corresponds to a subcarrier's data in one OFDM symbol, and the modulation scheme applied (BPSK, QPSK, 16QAM, 64QAM) which dictated how bits are mapped to symbols on these subcarriers. This optimizes the transmission by utilizing the available bandwidth and preparing the data for the IFFT process.

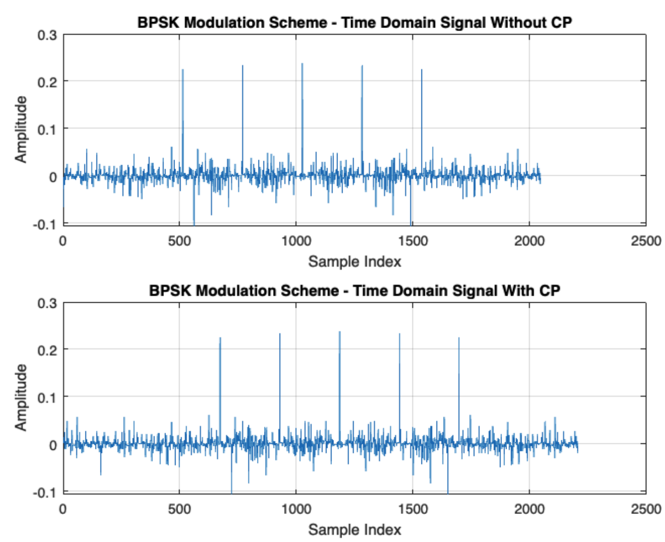
**Number 5:** I did the inverse fast Fourier transform on the modulated data for each modulation scheme to convert the frequency domain signal into time domain OFDM symbols. This prepared the data to be transmitted with minimal noise interference between carriers. The IFFT ensured that the subcarriers in the OFDM signal are orthogonal to each other, which allowed for a denser packing of subcarriers in the frequency system.

**Number 6:** I added a cyclic prefix to each OFDM symbol to mitigate the inter-symbol interference caused by the multipath propagation. This maintained the orthogonality of subcarriers in a multipath environment. The first OFDM symbol got a longer CP to account for the maximum delay spread of the channel while the rest got a standard CP length. To generalize what the CP was doing is that it as copying the end part of the OFDM and appending it to the front which created a buffer period between symbols that helped in combatting the ISI.

**Number 7:** I added each modulation scheme into a buffer which can then be displayed. Since this assignment was specifically looking for the 64QAM I only displayed the 64QAM OFDM symbols.

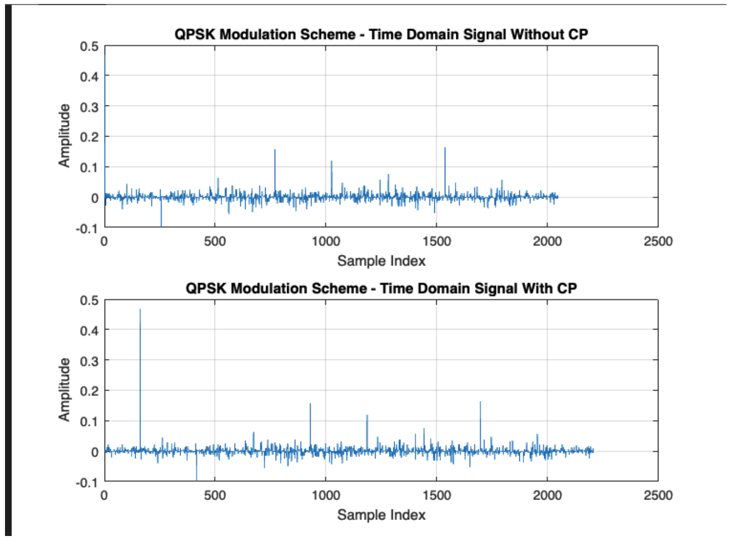
**Number 8:** This is the step that caused me to submit my assignment late. Primarily because it showed me that the code (4 different versions) was entirely incorrect, and even with this version, I am still unsure. I plotted each modulation scheme, and each figure has 2 subplots within them. The top graph shows the time domain without the cyclic prefix and the bottom shows the time domain with the Cyclic Prefix.

For the BPSK Modulation my graph looked like this:



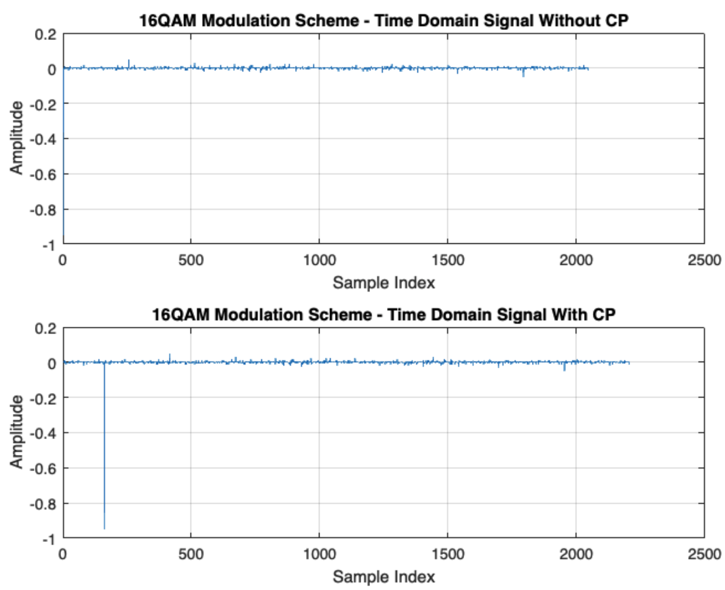

You can see that the high amplitudes have been shifted down. This means there was an addition of the CP to the data.

For the QPSK Modulation my graph looked like this:

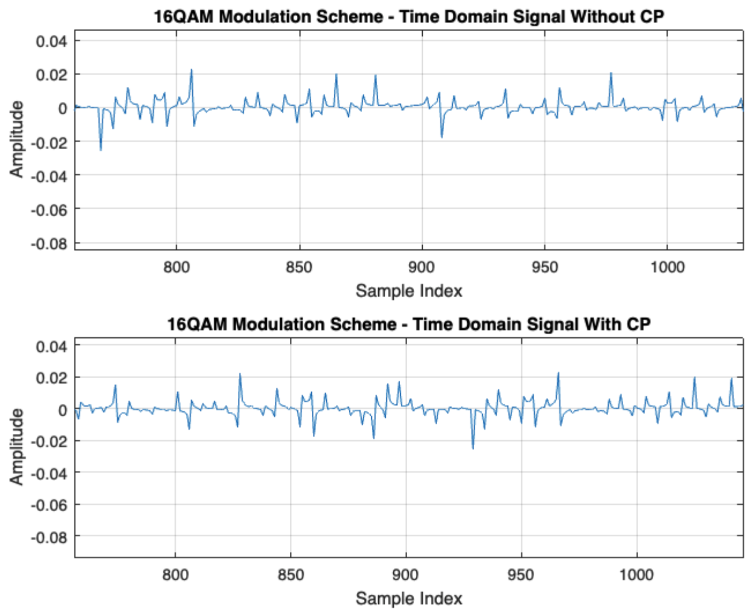



In this graph, I noticed an extremely high peak, which I am not entirely sure where it comes from, but it stayed consistent in the following graphs as well. The main point to notice here is that the graph is significantly less noisy than the BPSK.

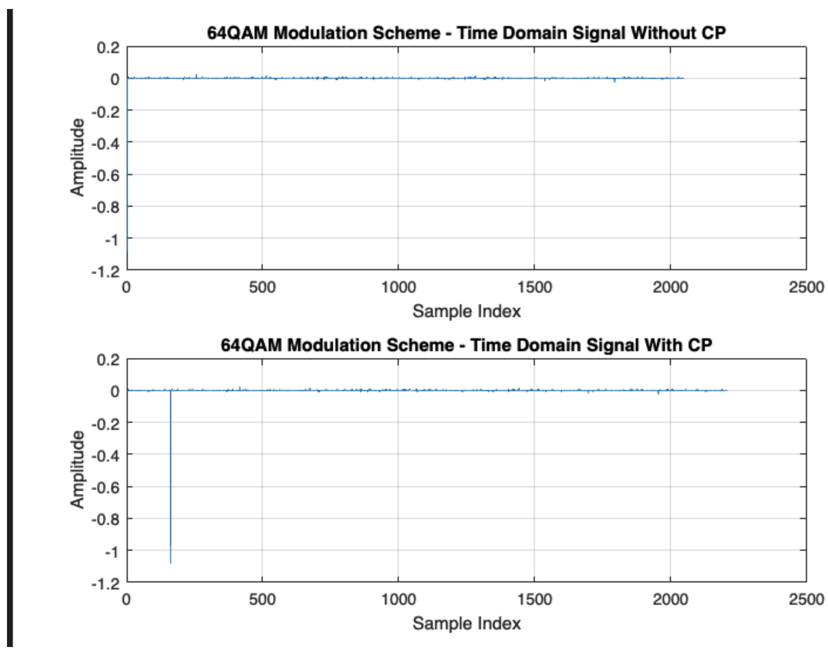
For the 16QAM Modulation my graph looked like this:



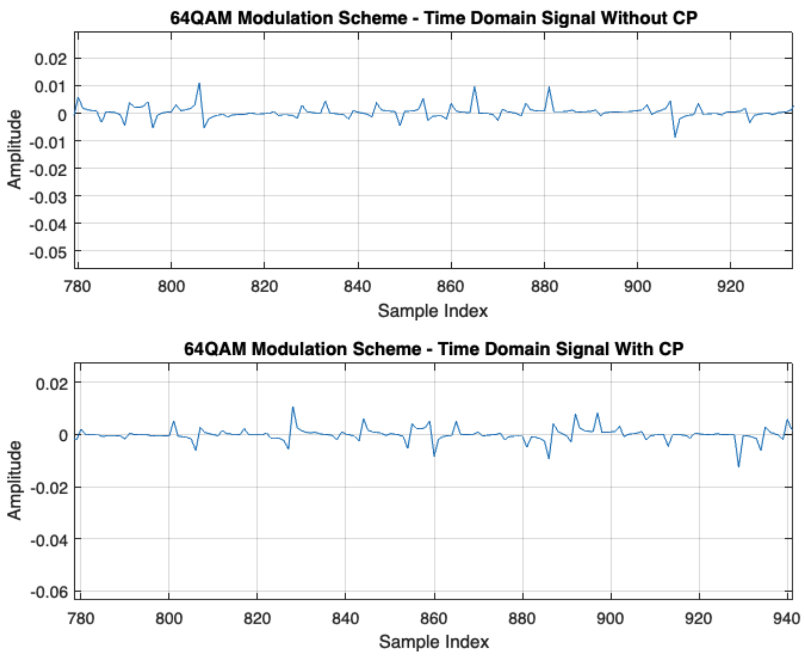

This graph shows a significant decrease in noise but that high peak is still there and this time it is negative, I was not sure how to get rid of it considering it is apart of the initial array. But zooming into the graph you can see the wave form more clearly:




For the 64QAM Modulation my graph looked like this:

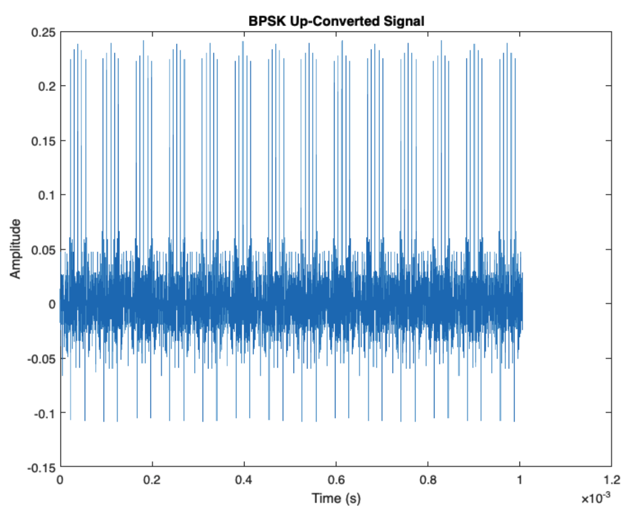



The dip stayed negative in the 64QAM, but the data stream is a lot less noise, because this modulation schemes each symbol represents 6 bits allowing for a denser packing of data and a higher throughput compared to the other modulations. Zooming into the graph you can clearly see the compactness:

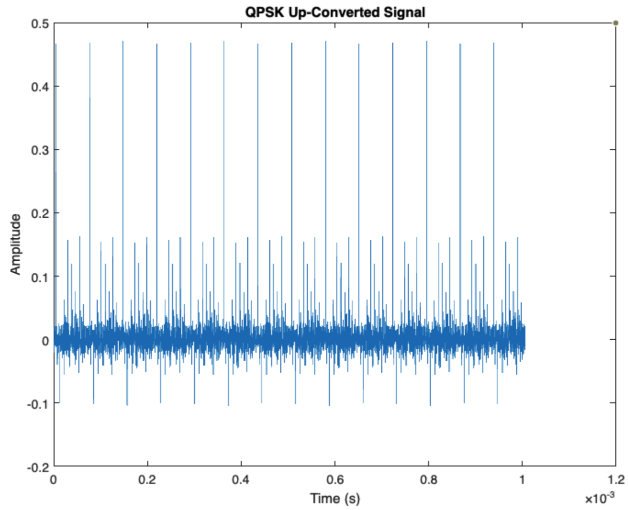



**<u>Number 10:</u>** This question was a little tricky, I am not sure if this is correct but the equation I used to generate the up conversion of the baseband signal was the carrier signal I did: carrier_signal = cos(2*pi*fc*t), where fc was 2.4GHz in the ISM band and t was the vector. I did not draw it but I did graph it using Matlab.

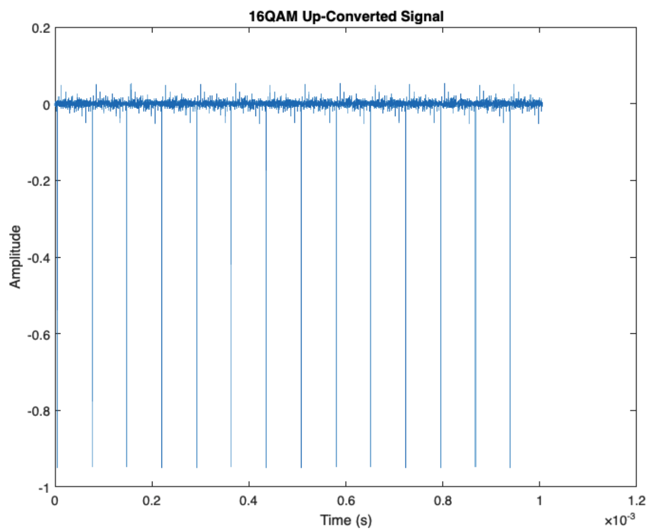
For BPSK I got:



For QPSK I got:




For 16QAM I got:




For 64QAM I got: