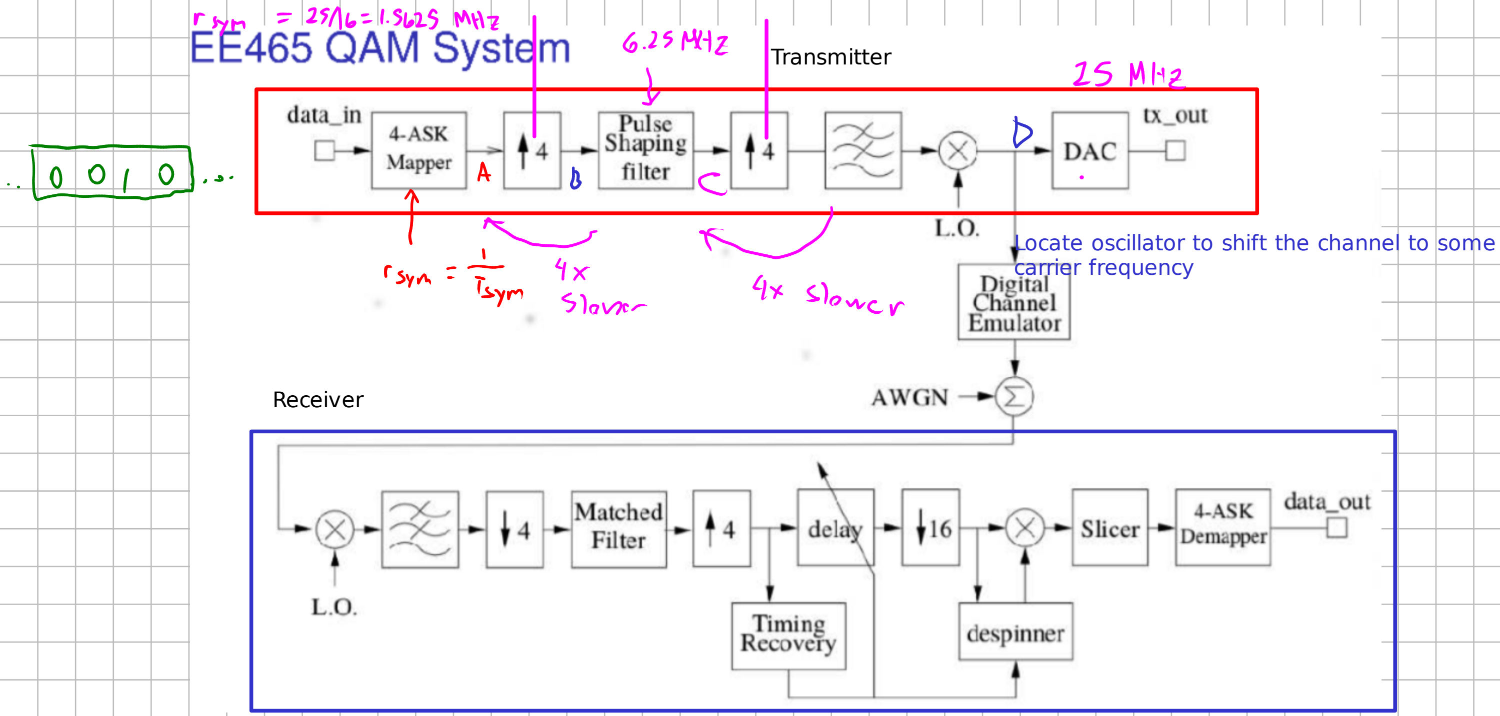
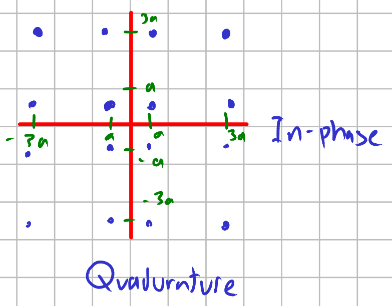


This is what we are building in this class; this class is focused on coaxial cable transmisison media used in internet services/digital cable services

16-QAM transmits waveforms like so



During any transmission interval, we are going to be picking 1 out of the 16 waveforms in the constellation.

WE can treat the in-phase & quadurature component independently

There are 4 possible values for the In-phase component

The way the system works in the transmittier, data_in is a stream of bits that we wish to transmit to a receiver

Transmitter takes long stream of bits and parititions into segments

for a 16-QAM we can convey 4 bits during each signal interval 2^4 = 16

transmitter takes string of bits and breaks it into groups of 4

For each 4 bit segement, it is passed into a mapper

Mapper translates the 4-bits into one of the 16 points in the QAM constellation.

Due to the rectangular shapre of the constellation, we can treat the in-phase and quadurature components as being independent, we can build them instead of being a 16 Mapper in our transmitter, we treate the in-phase and quad channels as independent components throughout the above system

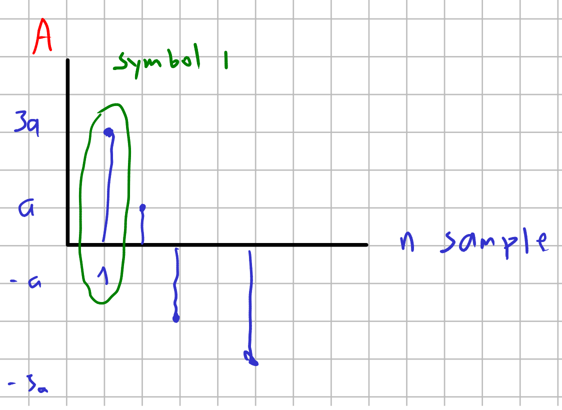
so the block diagram of the transmitter only considers one of the 2 channels. Presumable of the 4 bits, 2 of them correspond to quad channel and the other correspond to the in-phase channel.

the 2 bits are passed into a mapper, which takes the 2 bits and maps them to one of the 4 possible signal levels: 3a, a, -a, -3a

This process is repeated for each symbol taht we wish to transmit in the system.

This mapper operates @ rsym or the symbol rate which is 1/(symbol period). Every symbol period, the mapper takes 2 bits and maps them to an amplitude

This is implemented in the FPGA and is a discrete time system where it takes on 4 possible amplitudes



To construct this system, we need a pulse shaping filter

if we look at the spectrum of signal A in the frequency domain, it would look like a white noise signal OR flat from 0 to 2pi

It looks like this since the input stream of bits is randoms

In real transmission system, we don't get to use up the entire transmission medium for ourselves

ex: If we are broadcasting a television channel, we would want to have many channels shared across this transmission medium but the spectrum is flat which makes overlap and interfere with each other

we use a PS filter to limit the spectrum and create a waveform shape to created the specific waveformw e want to transmit so this pulse shaping filter creates the shape of the waveform

theres different pulse shaping filters we can use but the common on is the SRRC filter

a SRRC filter in the transmitter and receiver, jointly look like a raised cosine; if we convolute a SRRC with a SRRC, the output is a Raised cosine

we want an overall RC filter since it obeys the Nyquist property for no ISI

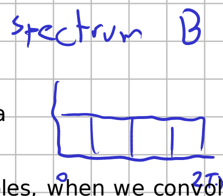
we upsample by 4 to give filter space to work with (inserts [n-1] 0's between each sample)

when you upsample by n, the spectrum is compresed by a factor of n and we have n copies of the original spectrum

Since the spectrum @ A is flat from 0 to 2pi, then the spectrum @ B is flat from 0 to 2pi but theres 4 copies of this compressed spectrum that spread across from 0 to 2pi

Random signals with a flat spectrum

ex: Going to inject an impulse w/A = 3a, then a 2nd impulse that occurs 5 samples later, and so on

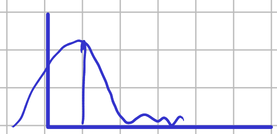


PS filter will have an impulse response defined by the SRRC filter, since its a discrete time filter, its a sampled version of the filter's response

the width of the pulse/Nsps, is going to be greater than 4, since we injected a sample every 4 samples, when we convolve the impulse w/the filter's response which will output a SRRC pulse centered @ that impulse

the second impulse comes in and is impulsed @ n=5, when we sum these impulses, we see that they will overlap and interfere with each other

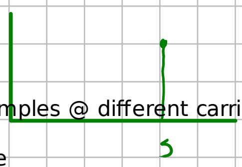
This interference is known as ISI (inter symbol interference), the interference of one sample with another. ISI limits the performance of our system

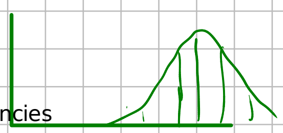


We will design our filter in such a way that it has an SRRC filter response

"stopband leakage"

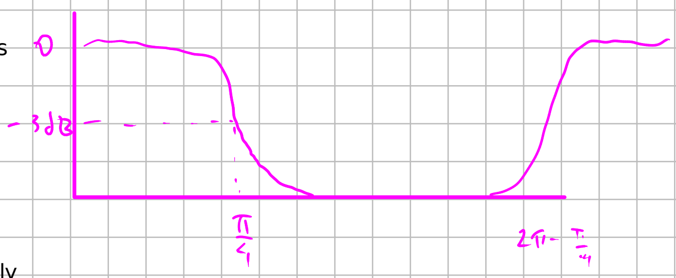
before sending to the DAC, we need to upsample our signal more in order to match the DAC sampling rate (25 MHz).





To share more channels, we need to upsample more and shift the samples @ different carrier frequencies

when we do this upsampling, we create more images that need to be filter out

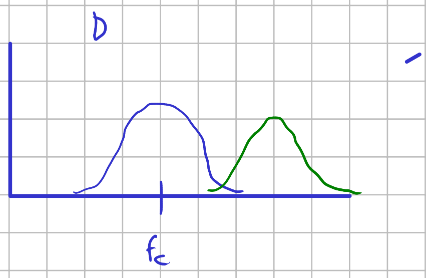


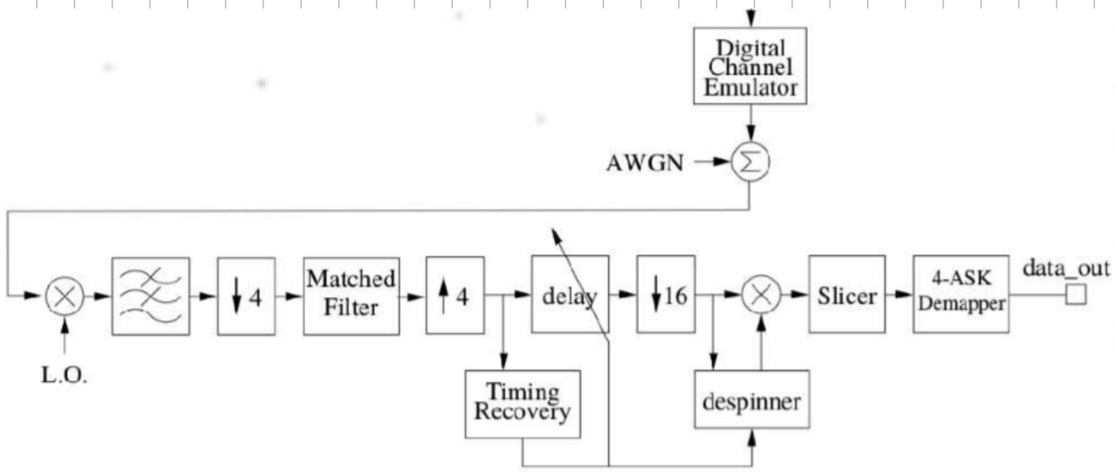
the sampling rate of PS filter will operate @ 6.25 MHz/Msamples

the symbol rate will operate @ 25/16 = 1.5625 MHz

At point D, the shape of the spectrum is similar but has been shifted up to the carrier frequency

Then another channel can be transmitted nearby where it slightly overlaps our channel enough to not cause ISI





Output of transmitter is sent in a real communications system through a channel. The specific aspects of the channel we emulate will be the variable gain (transmission medium) and AWGNhe transmitter

the matched filter in the receiver will approximate a SRRC impulse response. A matched filter is used to project a signal into a basis fucntion to help us decide our signal

Timing recovery block

in a real communications system, its hard to know when to sample the signal. Theres a variable amount of time delay between the transmission and receiver since it depends on many factors.

We need to construct the circuit that looks at the output of the matched filter to determine the optimal sampling rate

The slicer: since we're looking at 4 channels, theres 4 points to examine.

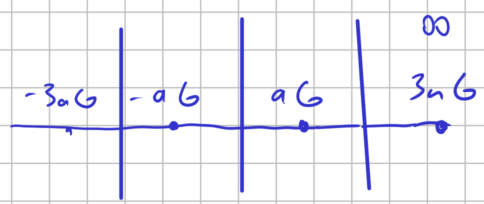
The points won't exactly correspond to what they were in the transmitter.

G for gain

we're constructing these decision rules to determine the signal.

The slicer can estimate the value of a and G is

Once we've done the slicing, we can take the received points and map them



There are many tradeoffs to consider when designing our system ex: to minimize the ISI and make the spectrum as sharp as possible, we need a very long filter with a long IR which costs alot which may not be desireable

there are tradeoffs in the spectral requirements where when you build a channel, you need to make sure the stopband attenuation is small enough so that a signal will not interfere with its adjacent signal

we control the noise with more coeffs, etc but this will cost more resources to implement

the first deliverable focus on the PS filter and matched filter jointly determine the performance of the system

D2 can measure the performance of the system

D3 builds the whole circuit up to matched filter, where we use the MER circuit to measure and evaluate our system
D4 builds sampling/conversion blocks to measure the BER

D5 involves the timing recovery and slicer circuitOrangey