Adam Chapman and Ryan Gill
Professor Vu
EE-107 Final Project
December 11th, 2016

DESCRIPTION

The purpose of this project was to simulate a communications system in MATLAB. This was accomplished by modeling the following stages that are typical in a communications system: image pre-processing, conversion to a bit stream, modulation of the data with a pulse shaping function, transmission through a channel, addition of noise, equalization, matched filtering, sampling and detection, conversion back to an image, and image post-processing. Each module has its own MATLAB function. Other helper functions were also written. The following is a brief description of how each module was implemented. The image pre- and post- processing steps are omitted because they were provided by the instructor and were not altered.

Conversion to a bit stream: This step was implemented by writing several helper functions and using them to perform the conversion step by step. ImagePreProcessing converts the image into an 8x8x625 array of Discrete Cosine Transform (DCT) blocks, which will be converted into bits for transmission. The ConvertToBitstream file function calls the ConvertBlocksToBits function on N blocks at a time. The ConvertDCTBlockToBits function is then called on each of the N DCT blocks. The end result is that a row in bit_matrix corresponds to N DCT blocks.

Modulation: Two pulse shapes were used for this project: the Half-Sine Pulse (hereafter referred to as “HSP”) and the Square Root Raised Cosine pulse (hereafter referred to as “SRRC”). The equations from the project descriptions were used to write the HSP and SRRC functions, which generate the pulses with a given set of parameters (sampling frequency and bit duration, and truncation length and rolloff factor for SRRC). The SRRC function is normalized so that a pulse has the same energy as the HSP pulse. This will enable the performance of the two pulses to be compared later.

Channel: A channel frequency response was given in the project description and was used to simulate echoes that might be seen in a real communications channel.

Noise: Gaussian noise with mean 0 and standard deviation $\sigma$ (the square of which is a parameter) was added to the signal to simulate random noise that appears in communications channels.

Equalization: Two equalization filters were utilized: the Zero Forcing Filter and the Minimum Mean Square Error filter (referred to as “MMSE” hereafter). Both filters counteract the effects of the channel, and the MMSE goes further by mitigating the effects of noise.

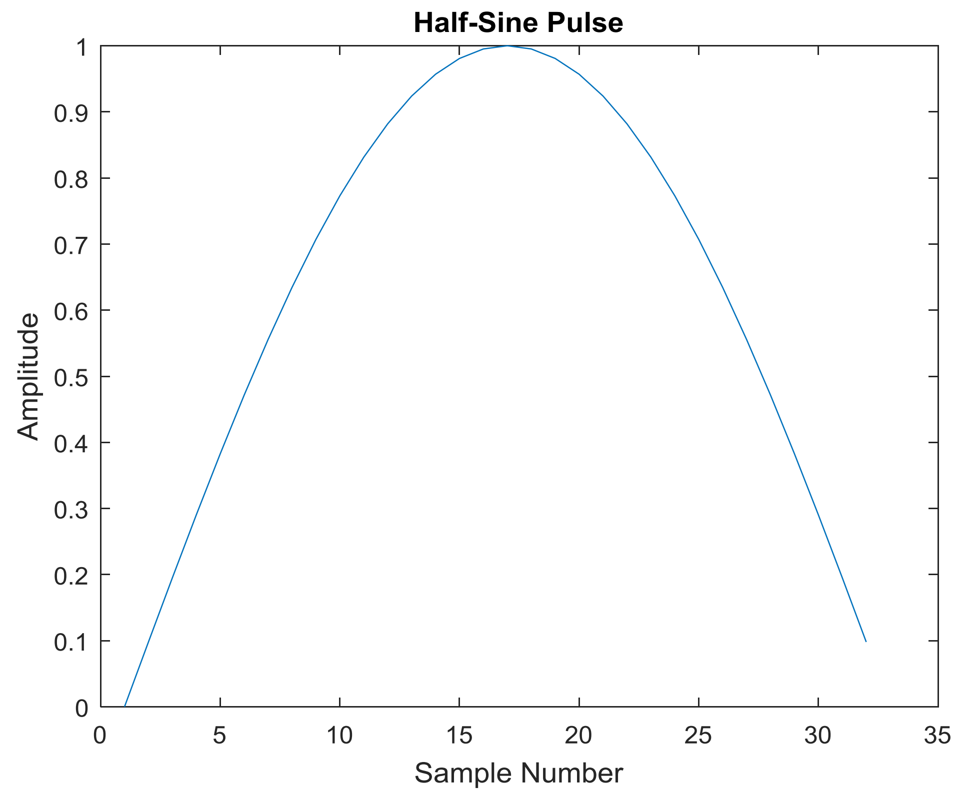
Matched Filter: Two matched filters were modeled: one for HSP and one for SRRC. The output of the matched filter was computed by convolving the input signal with the appropriate pulse shaping function.

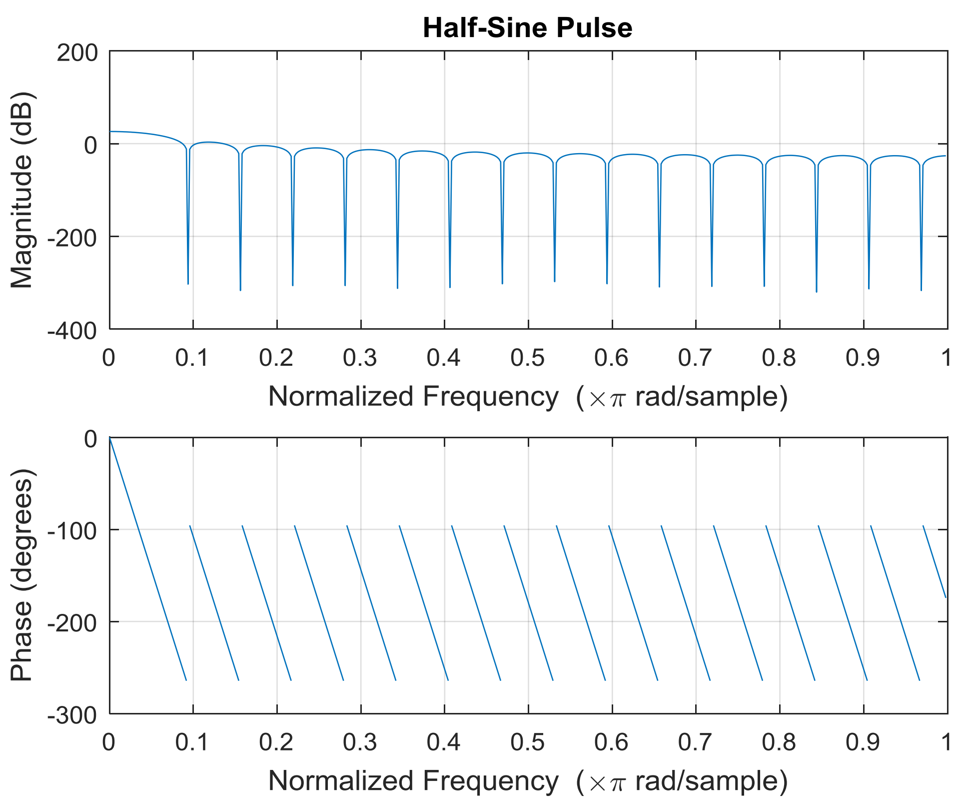
Sampling and detection: The delays in the signal chain were used to determine the ideal sampling time at the receiver for each pulse shape. A decision was made between a 1 and a 0 bit, and the results were stored in an array called received_bits.

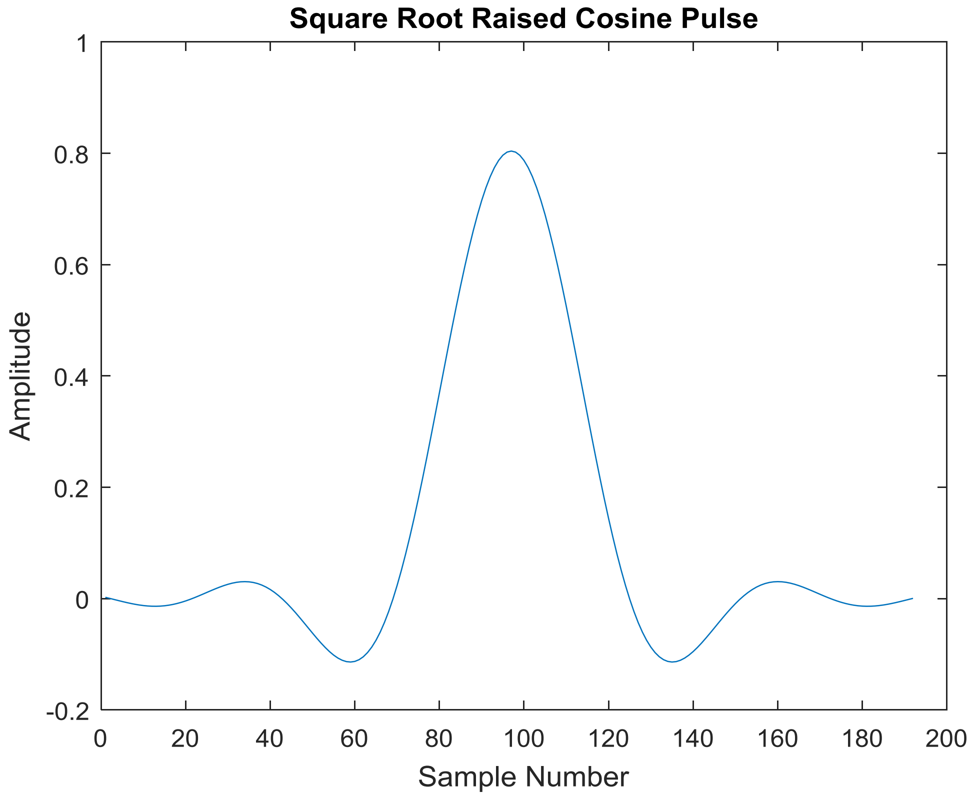
Conversion to image: The reverse of the operations in the conversion to a bit stream function were implemented to transform the received data back into the original image format for viewing.

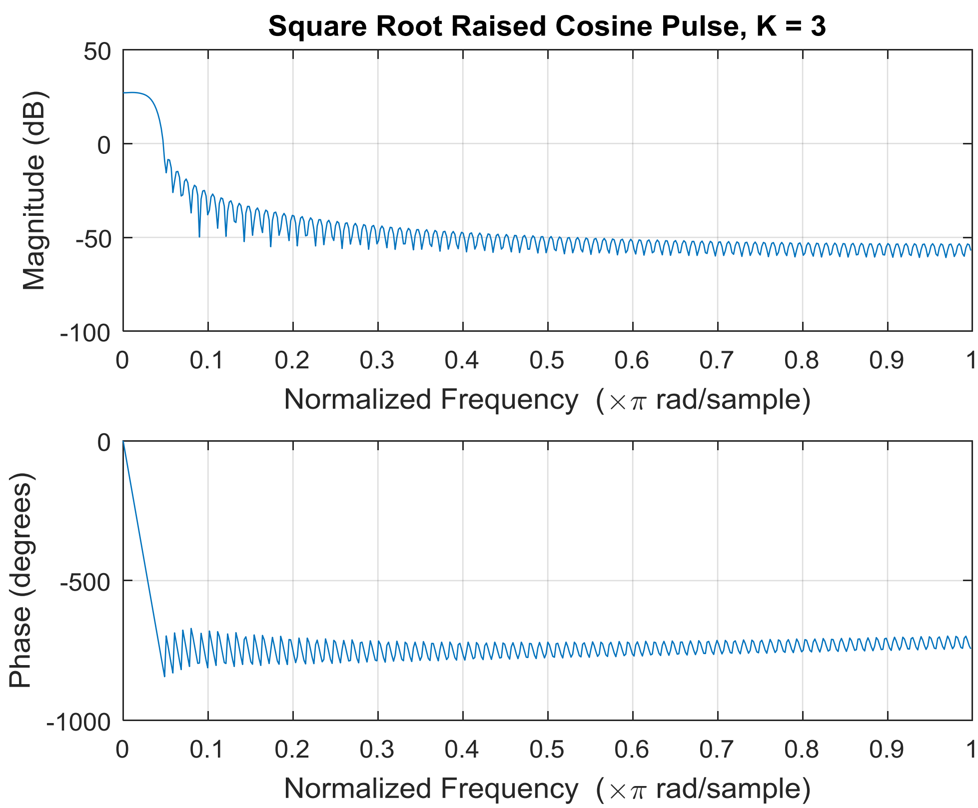
QUESTIONS

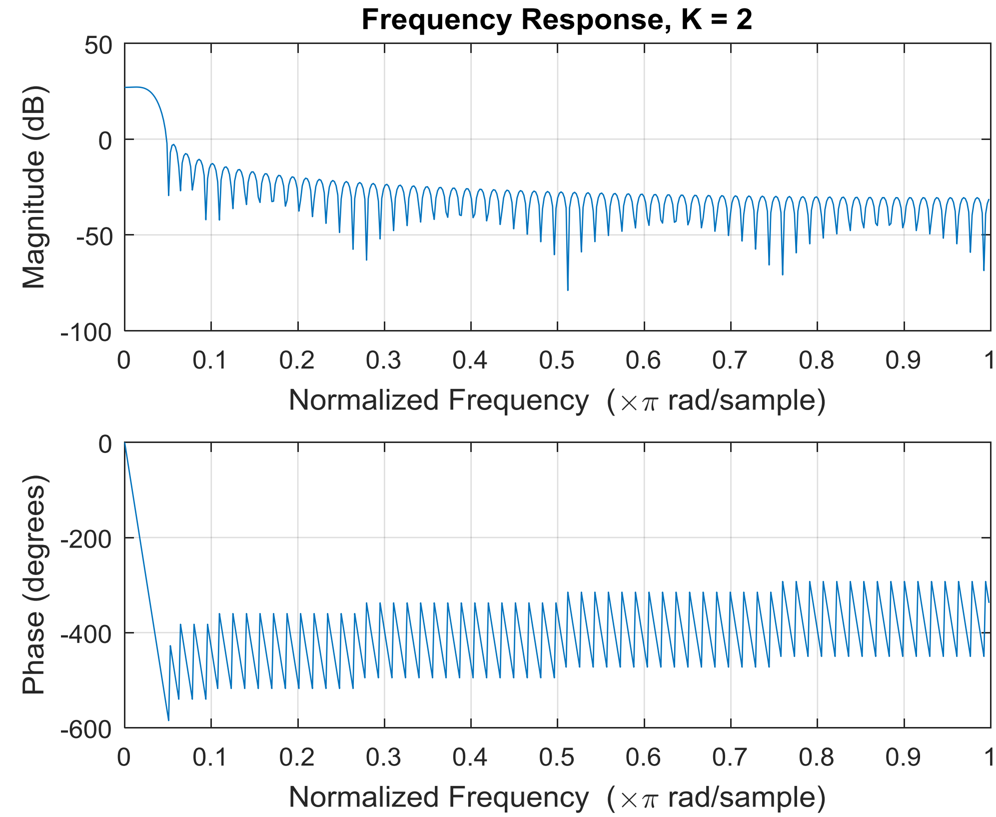
Q1) The pulse-shaping functions and their frequency responses are given below. The plots of magnitude vs frequency show that HSP uses a larger bandwidth SRRC. This implies that in situations where the available bandwidth is limited, the SRRC pulse should be considered for pulse shaping. Increasing the length of the truncated SRRC pulse (increasing K) decreases the bandwidth of the signal. This is because as a signal gets larger in the time domain, it gets smaller in the frequency domain. Sharp cutoffs require more frequencies, so as the truncation length increases and the SRRC pulse approaches 0, less frequencies are required. Changing the rolloff factor $\alpha$ has a similar effect. Increasing $\alpha$, which causes the wave to oscillate more quickly, also increases the required bandwidth. This makes sense because increasing the rate of oscillation would imply that higher frequencies are present in the signal. Plots are given below for different values of K and $\alpha$.


Half-Sine Pulse
Amplitude
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
5
10
15
20
25
30
35
Sample Number

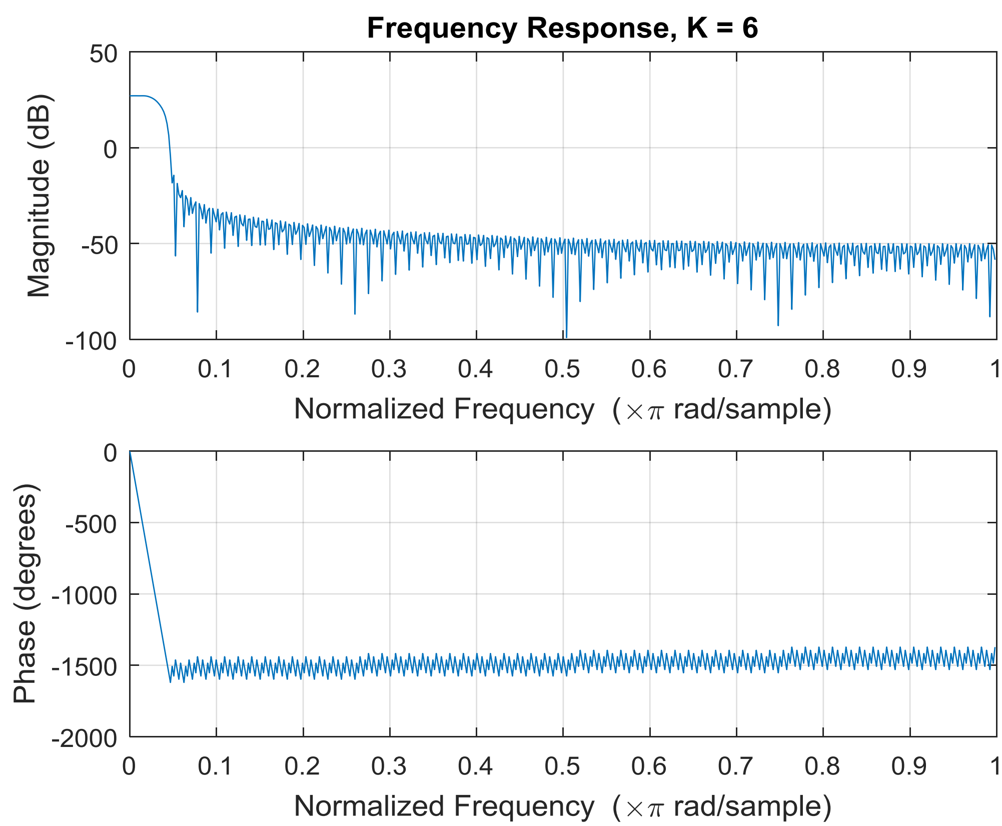

Half-Sine Pulse
Magnitude (dB)
200
0
-200
-400
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)
Phase (degrees)
0
-100
-200
-300
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)


Square Root Raised Cosine Pulse
Amplitude
Sample Number
1
0.8
0.6
0.4
0.2
0
-0.2
0
20
40
60
80
100
120
140
160
180
200

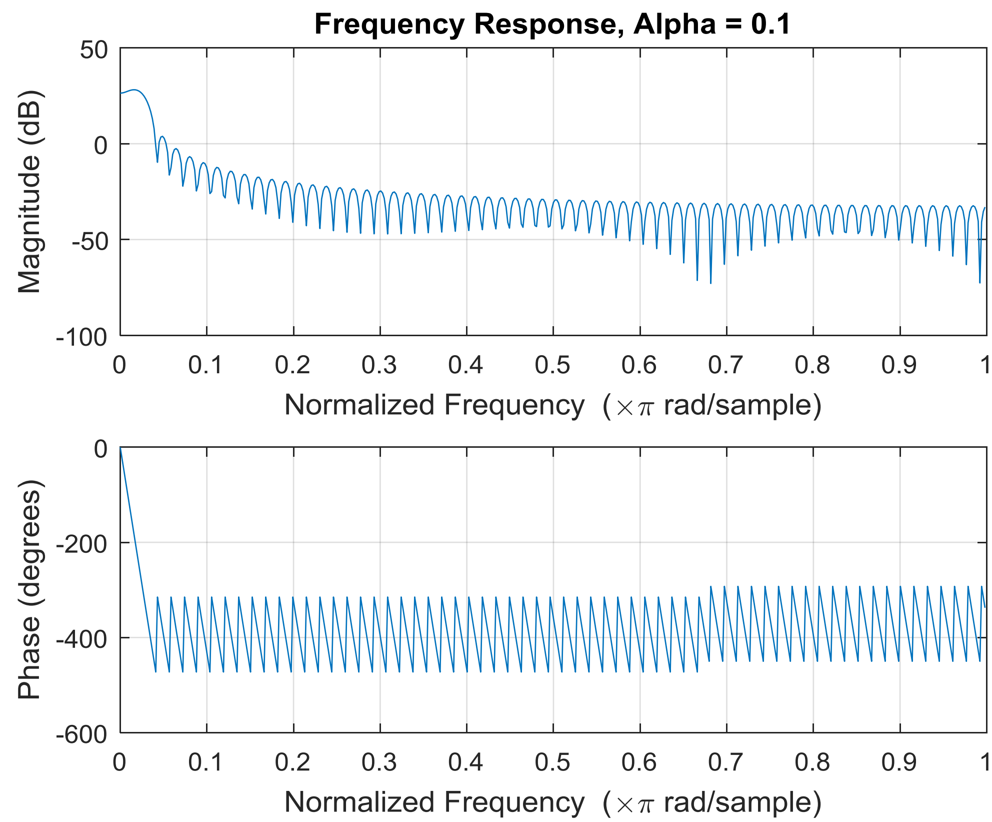

Square Root Raised Cosine Pulse, K = 3
Magnitude (dB)
50
0
-50
-100
Normalized Frequency (×π rad/sample)
Phase (degrees)
0
-500
-1000
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)

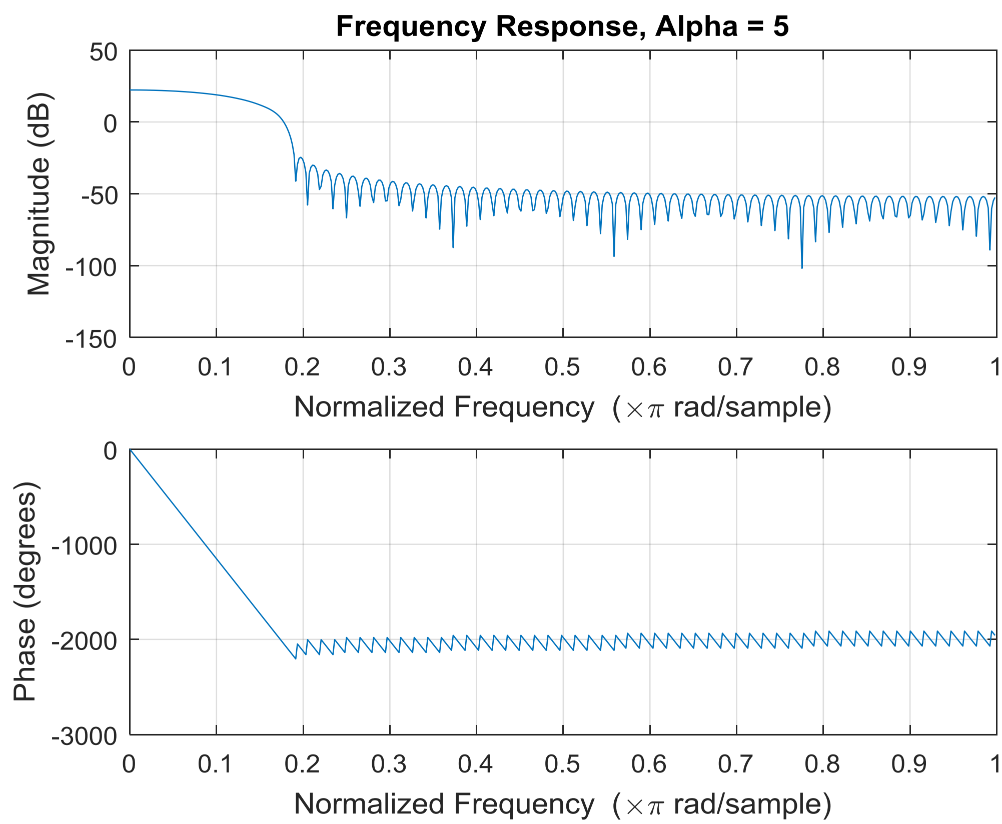


Frequency Response, K = 2
Magnitude (dB)
50
0
-50
-100
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)
Phase (degrees)
0
-200
-400
-600
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)

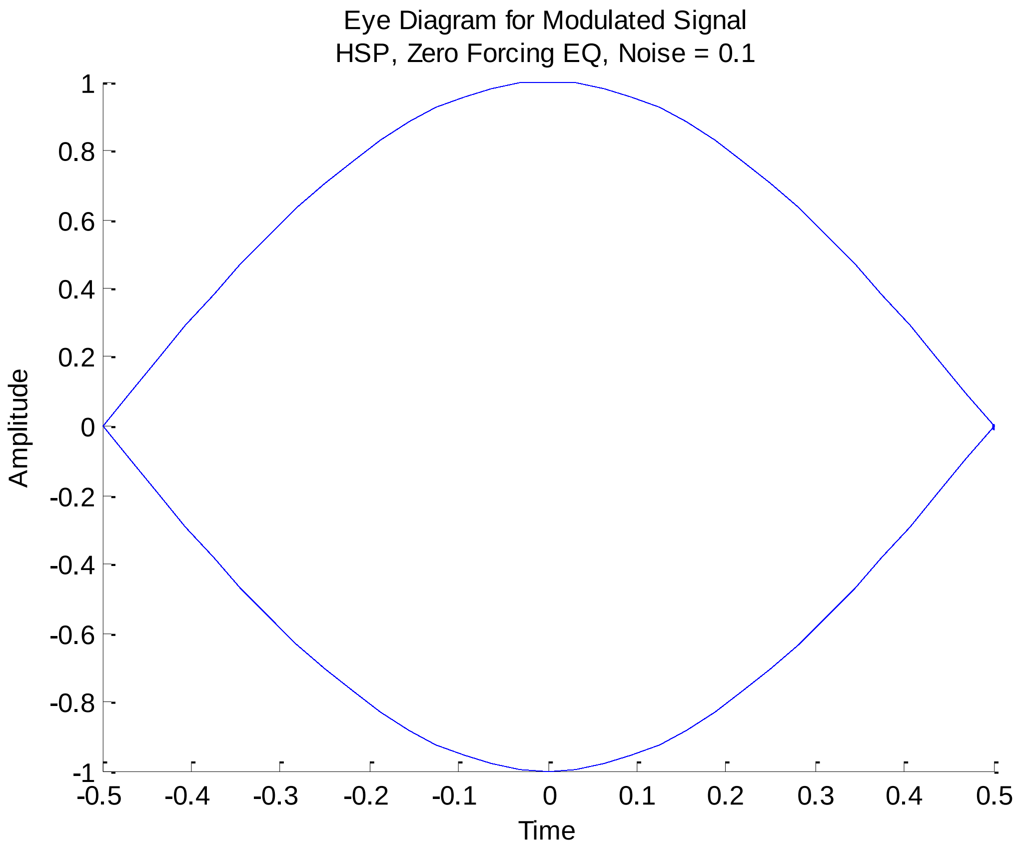



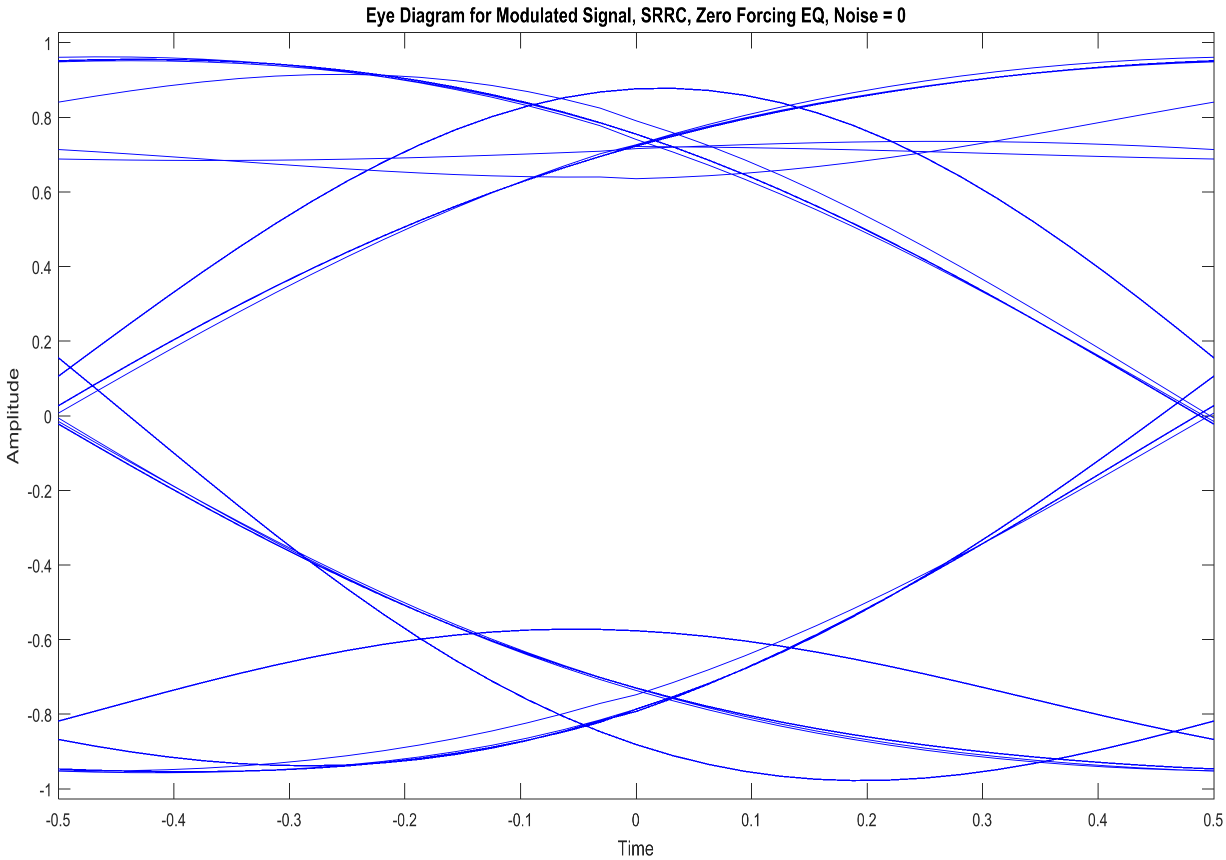
Frequency Response, K = 6
Magnitude (dB)
50
0
-50
-100
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)
Phase (degrees)
0
-500
-1000
-1500
-2000
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)


Frequency Response, Alpha = 0.1
Magnitude (dB)
50
0
-50
-100
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)
Phase (degrees)
0
-200
-400
-600
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)

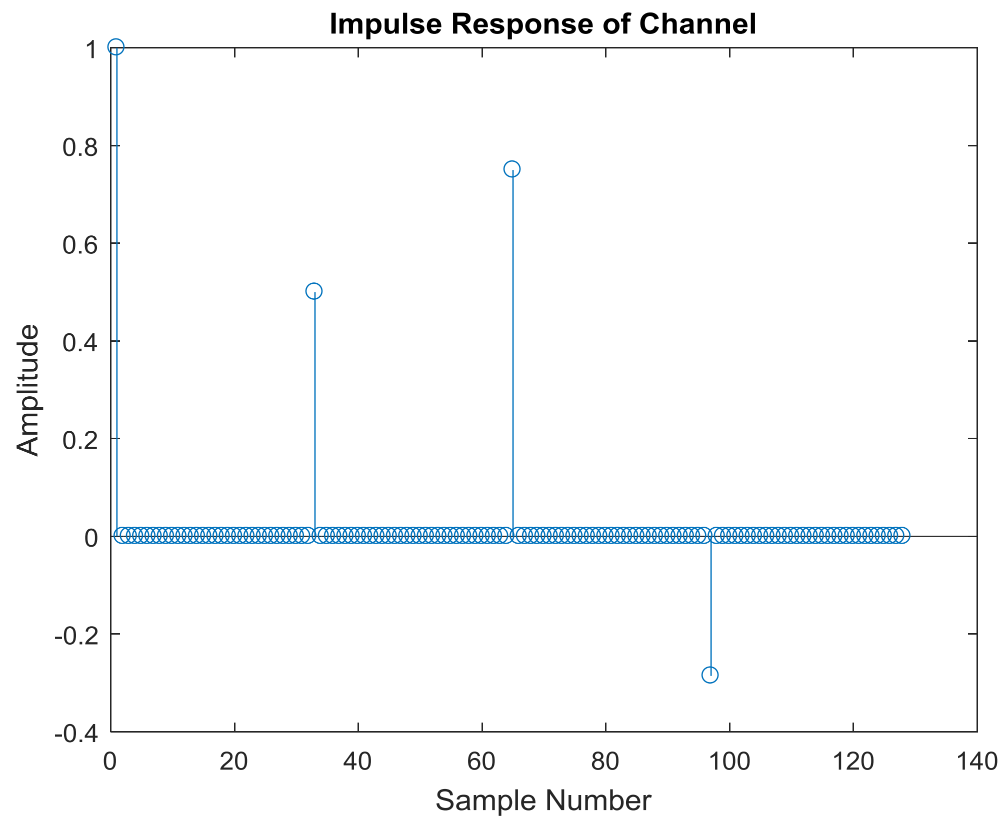

Frequency Response, Alpha = 5
Magnitude (dB)
50
0
-50
-100
-150
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)
Phase (degrees)
0
-1000
-2000
-3000
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)

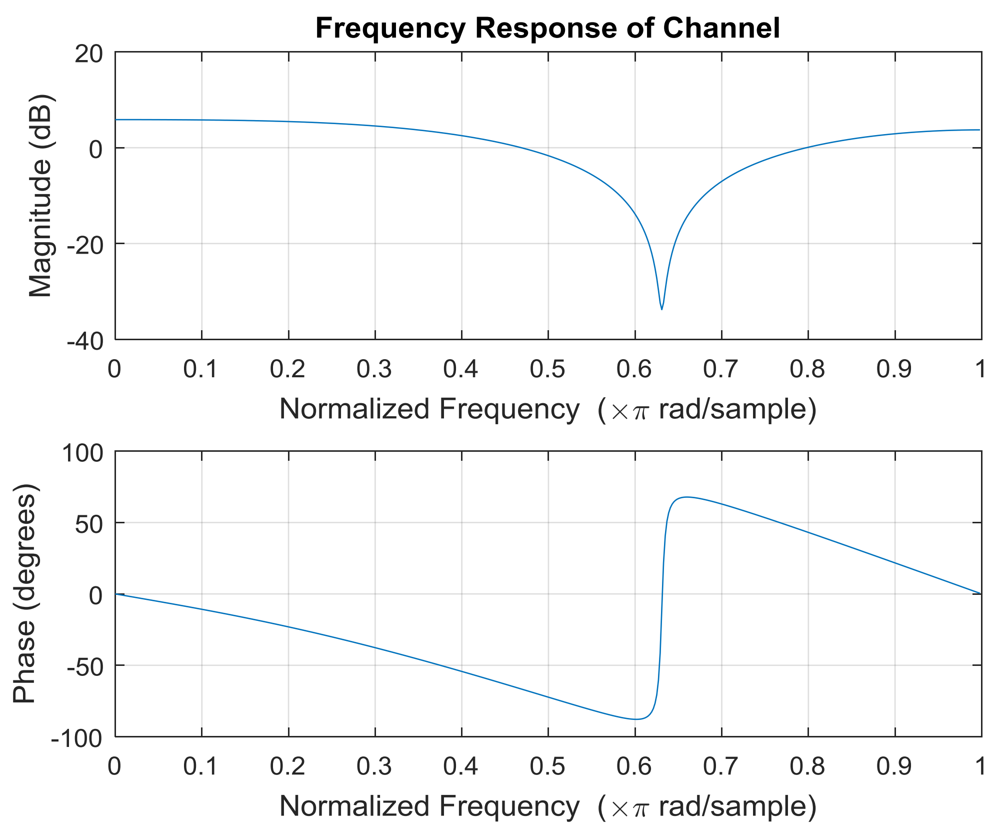
Q2) The eye diagrams for HSP and SRRC are plotted below. The eye opening is visible in both plots, but the diagram for HSP is much simpler. This is due to the fact that square root raised cosine pulses extend past their own bit duration and into neighboring bits. This is dependent on the truncation length K. Increasing K will cause bits to extend further and therefore interfere with more neighbors. This problem will be taken care of by the matched filter, which will cause the net response to be that of a raised cosine pulse.



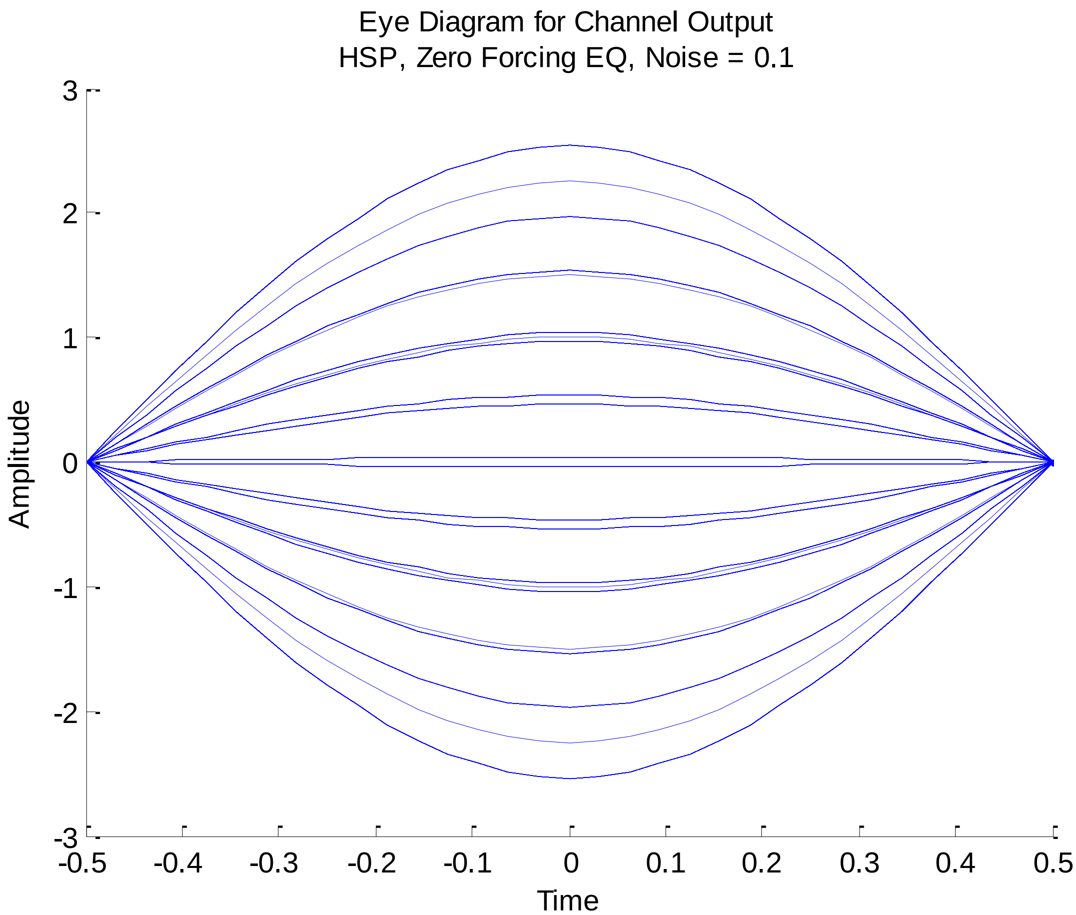


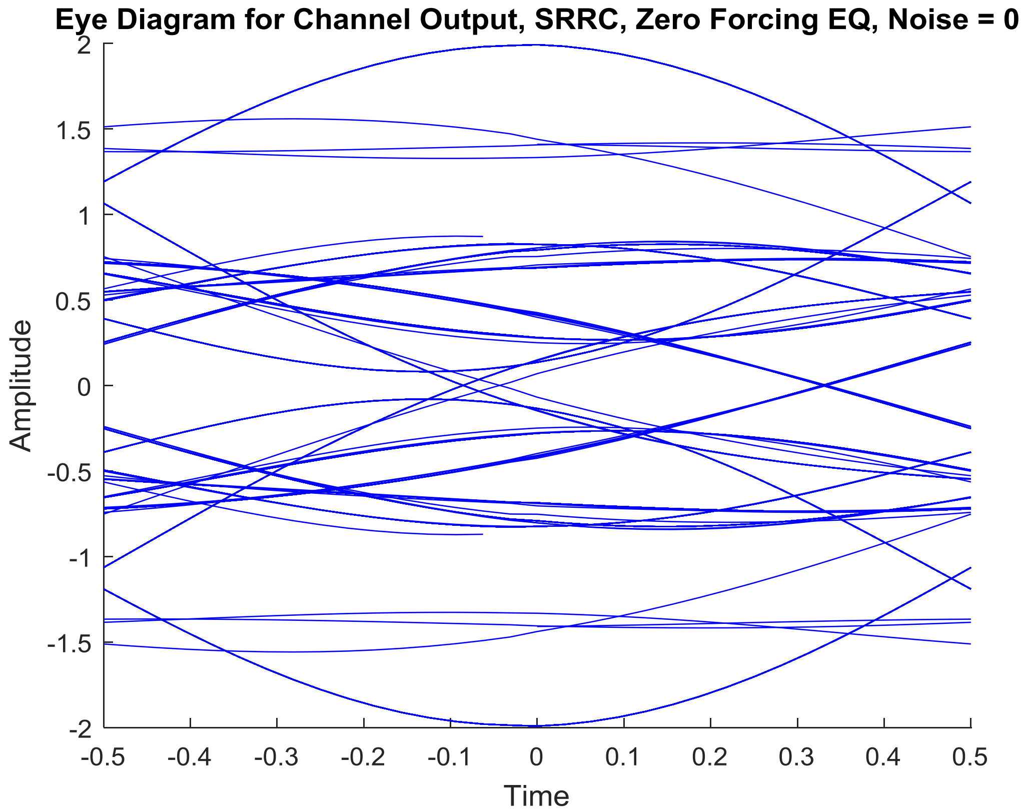

Q3) The impulse response and phase response of the channel are plotted below. Note that the frequency response of the channel is periodic, so only one period is shown.


Impulse Response of Channel
Amplitude
Sample Number
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0
20
40
60
80
100
120
140

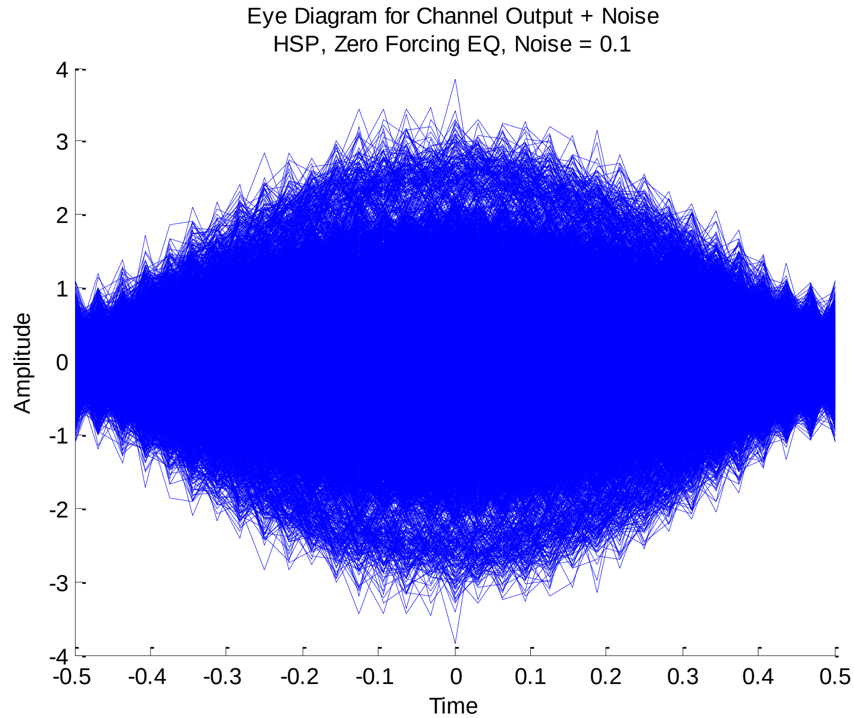

Frequency Response of Channel
Magnitude (dB)
20
0
-20
-40
Normalized Frequency (×π rad/sample)
Phase (degrees)
100
50
0
-50
-100
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)

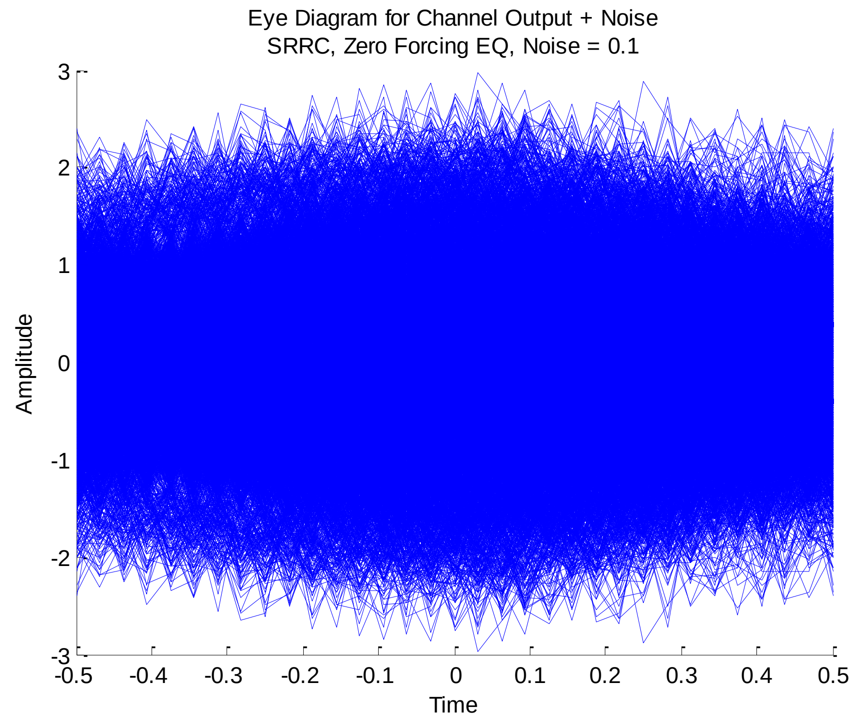
Q4) The eye diagrams for the channel outputs prior to the addition of noise are given below. HSP has many allowable signal levels due to the echoes of the channel. Since HSP is zero outside its own bit duration, these echoes will line up with neighboring pulses, either increasing or decreasing their amplitude but leaving their shape unchanged. This is not the case with SRRC, where neighboring pulses extend into each other. For SRRC, the echoes cause the eye diagram to become more jumbled. In both cases, the eye closes, making it necessary to compensate for the channel effects using an equalizer.






Q5) The eye diagrams after the addition of noise are given below. Here, the noise power was chosen to be 0.1. The noise causes the eyes to completely close, since they were already closing prior to the addition of noise and the noise causes each waveform to change randomly. In generating the eye diagram, neighboring bit durations will have different and random amounts of noise, making most amplitudes possible at that stage. This will make it necessary to counter the effects of noise at a later stage.

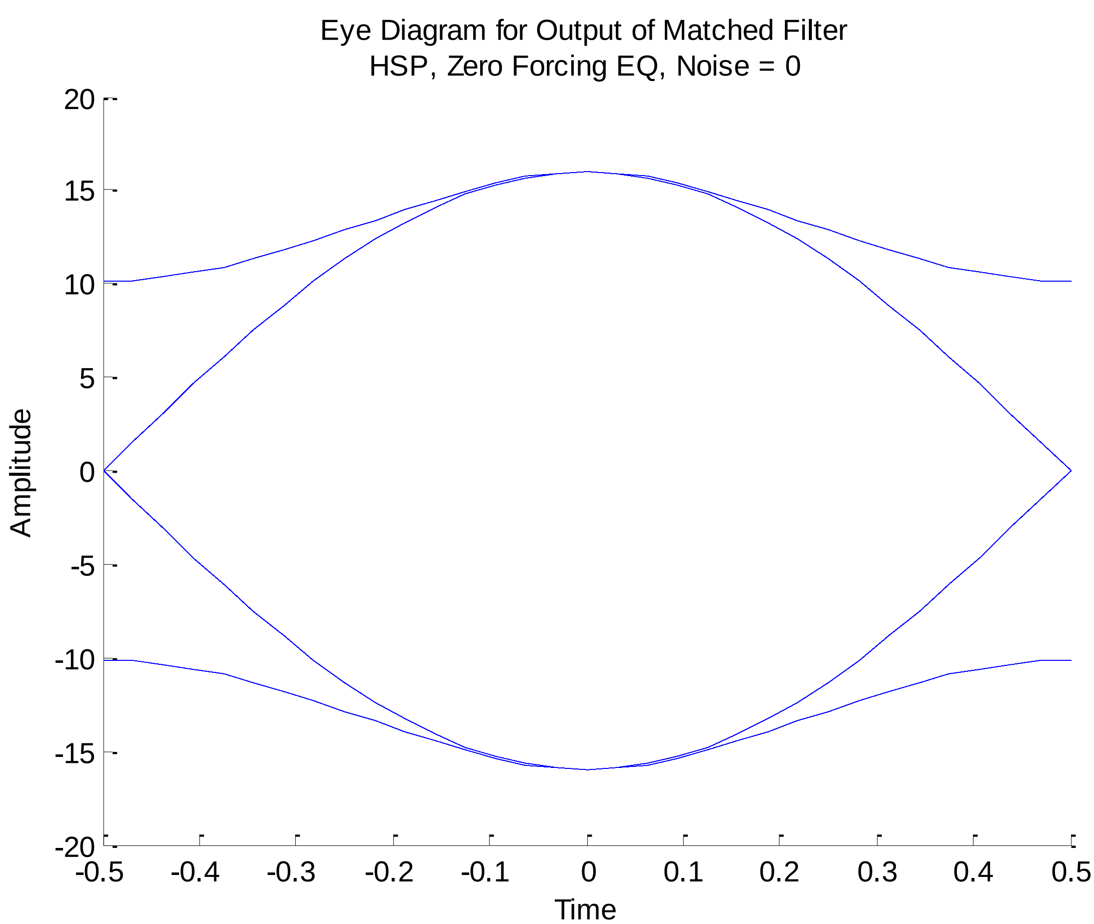


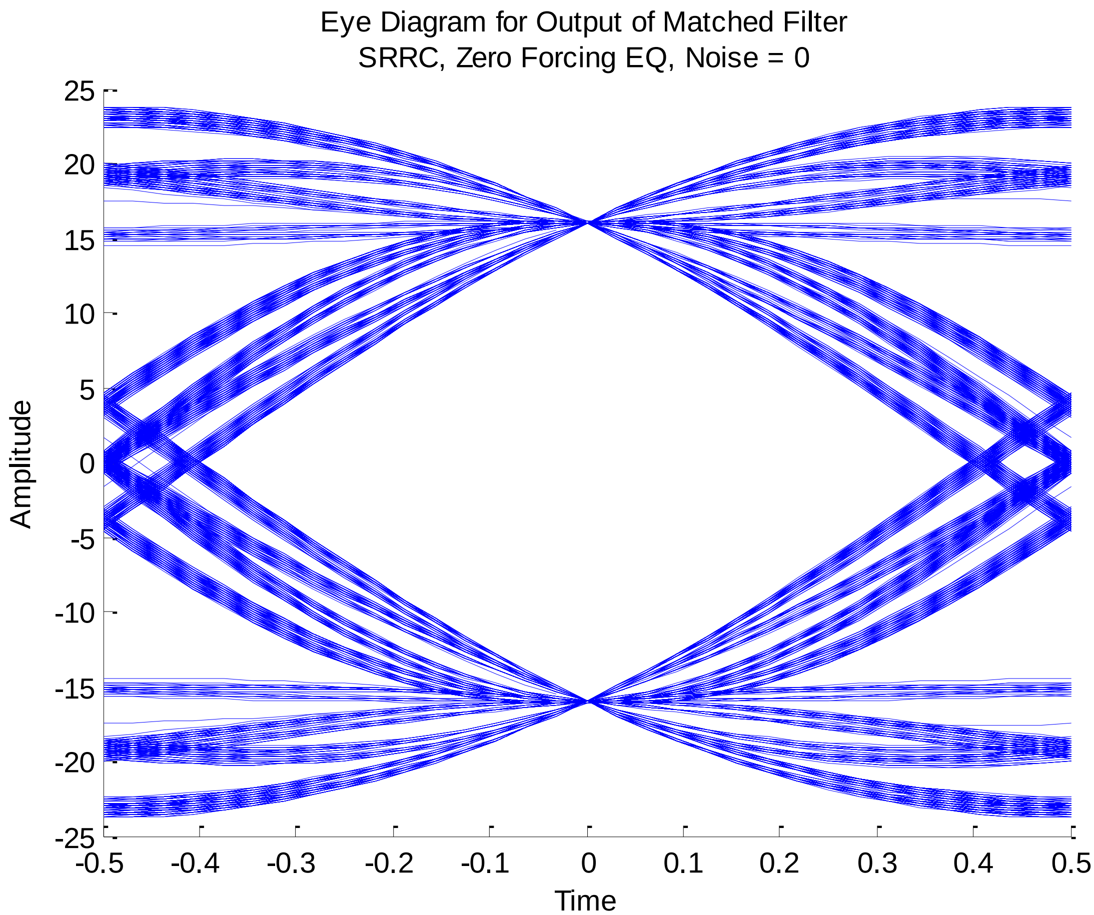




Q6) The impulse responses and frequency responses of the matched filters are the same as those of the pulse shaping functions (HSP and SRRC). This is because both pulses are symmetric, so the function $g_R(t) = g_T(t-T)$ will result in $g_R(t) = g_T(t)$. See the first four figures.

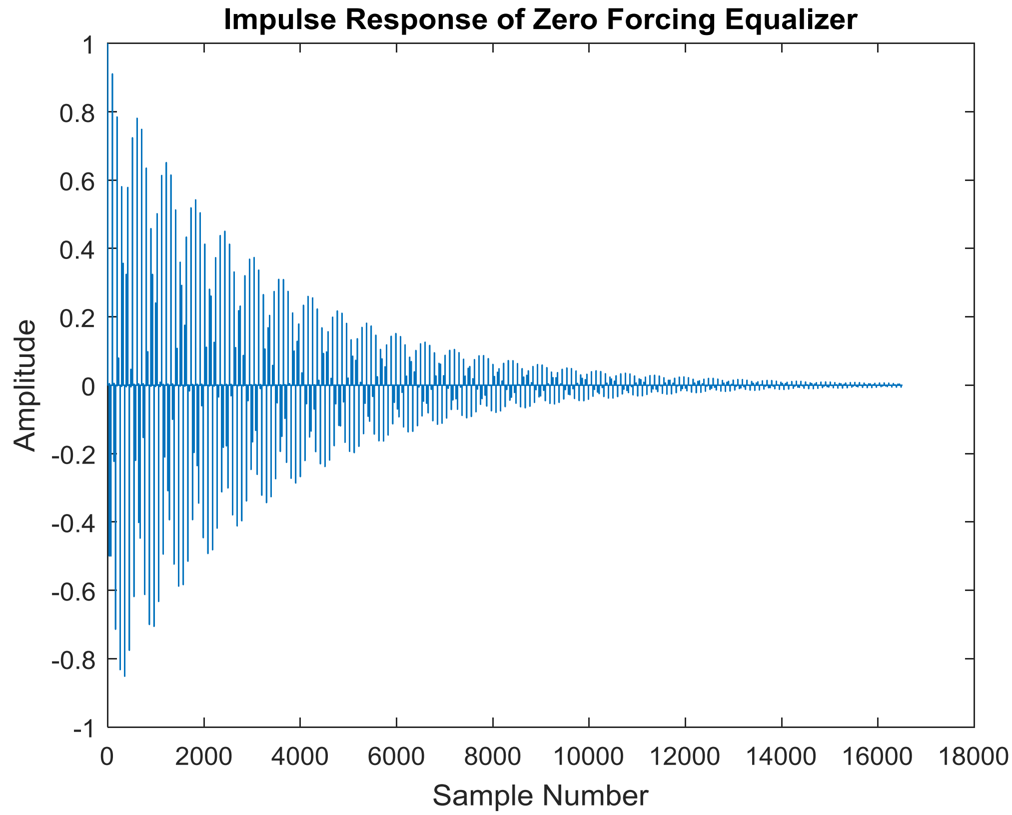
Q7) Note that in our implementation, the matched filter followed the equalizer. This was because it made more sense to us to add the channel effects then undo them with the equalizer first, then move outward in the signal chain. Since convolution is commutative, this should not affect system performance.

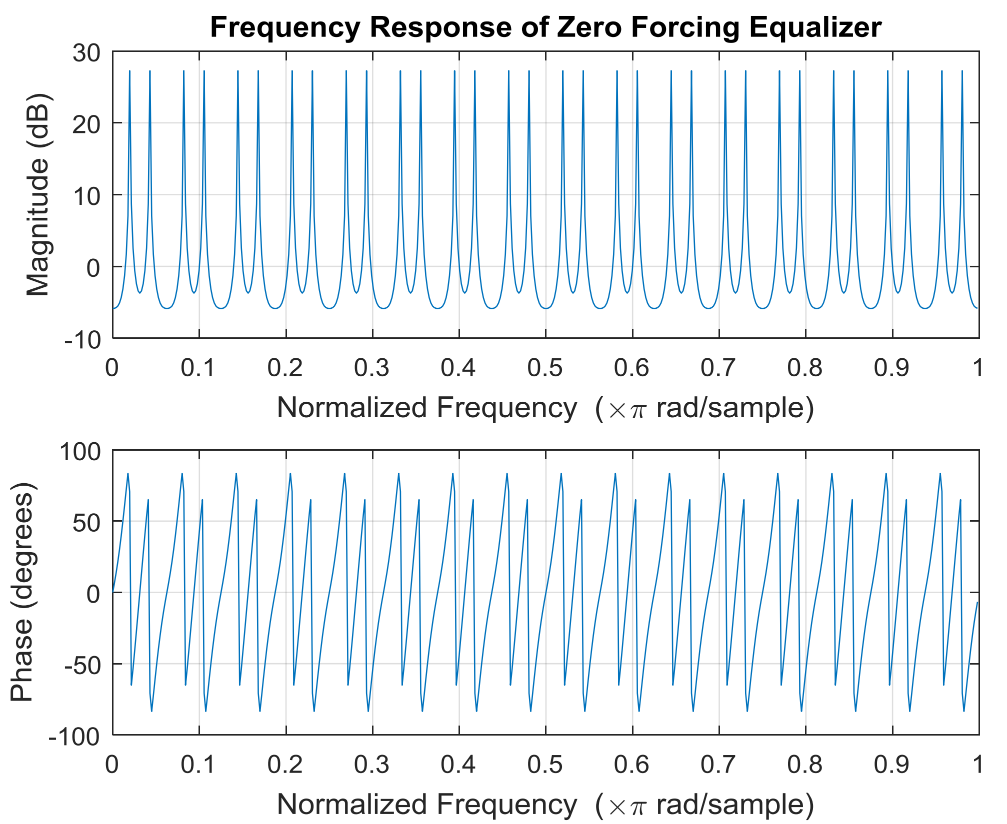
These eye diagrams show that the ideal sampling point is in the center of the eye diagram. This translates to the end of the HSP pulse and at the end of an SRRC bit duration. This is because when you apply the matched filter through convolution it is essentially taking the integral over the pulse. The maximum (minimum for a negative pulse) will occur at the end of the integral, so at the end of the pulse. The center point of these eye diagrams correspond to the end of the pulses. So for example, a half sine pulse is 32 samples long so we sample at the end of it, the 33 sample, then 65$^{th}$, etc. The half sine pulse could be sampled in other locations and still work, but the SRRC pulse needs to be sampled at the end of a pulse to prevent inter-symbol interference. The half sine pulse is contained to a single bit duration so there will always be 0 ISI. After applying the matched filter to the SRRC, sampling at the end of the pulses will have contribution from a single pulse and the value of all the other pulses will be 0. If you sample at other points, the other pulses will contribute and you will get ISI. You can clearly see this in the center of the eye diagram where all the pulses come together in a single point. That point is where there is 0 ISI.



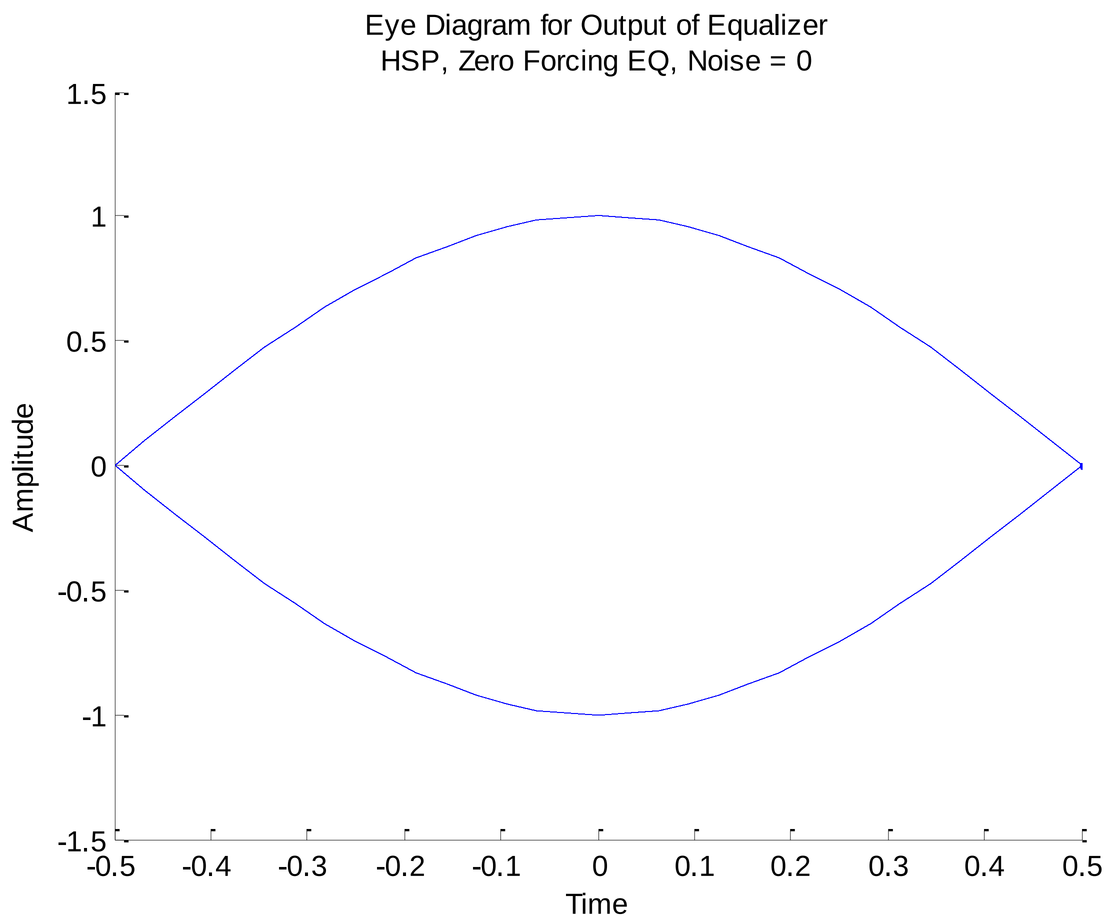


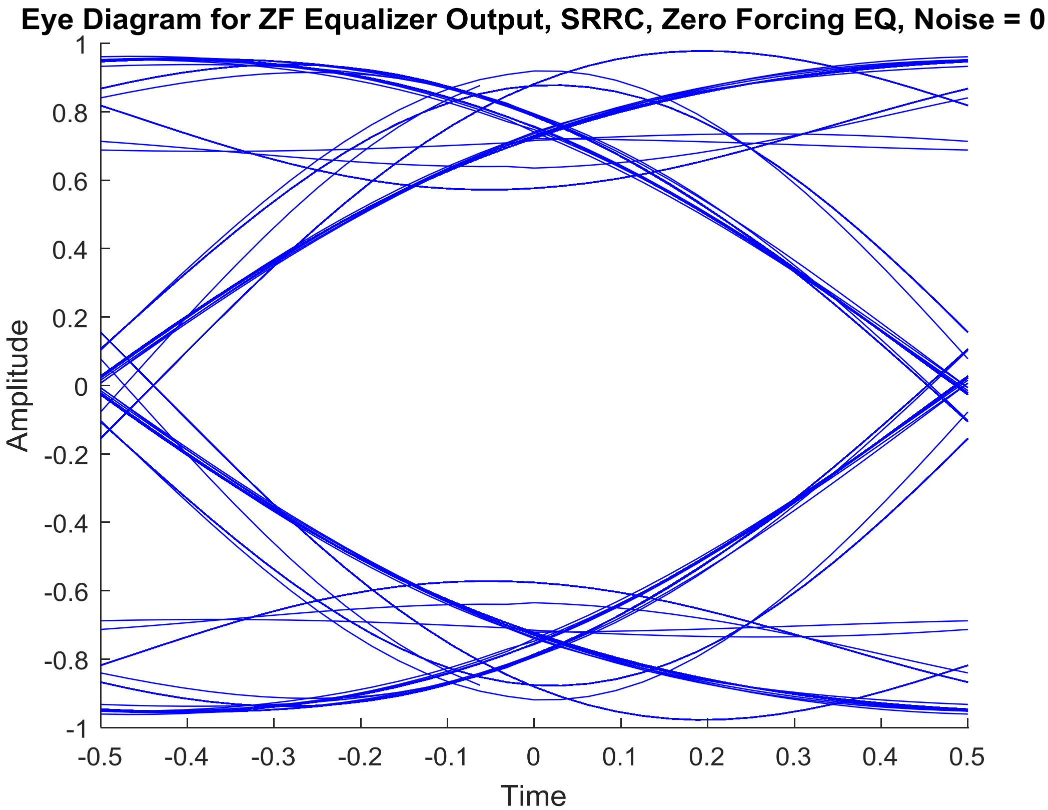

Q8) The Zero Forcing Equalizer is not always stable because there may be zeros in the channel frequency response. One period of the frequency response of the channel was plotted previously. The channel response has periodic deep fades where the gain drops to about -35 dB, which means that the zero forcing equalizer response would have to counteract those deep fades by providing sharp gains at those frequencies (hence the periodic spikes in the zero forcing equalizer response). Requiring such large gains at specific frequencies is highly unrealistic in practice, often making it necessary to use a different equalizer (such as the MMSE equalizer).


Impulse Response of Zero Forcing Equalizer
Amplitude
Sample Number
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2000
4000
6000
8000
10000
12000
14000
16000
18000

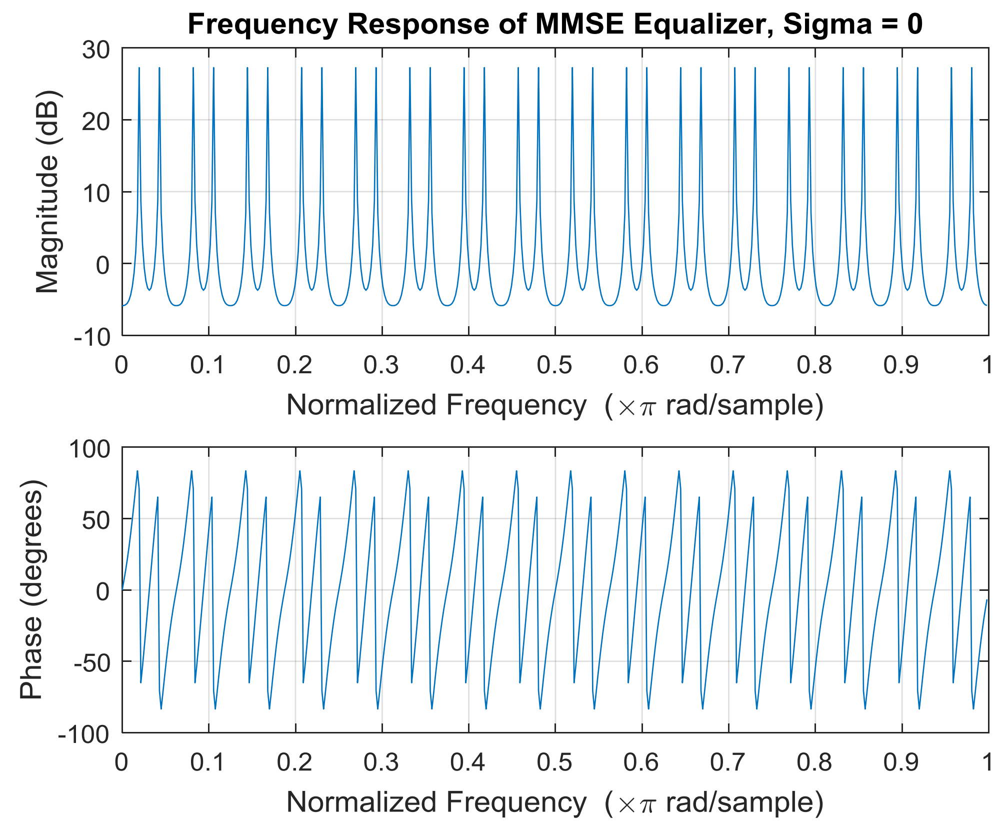

Frequency Response of Zero Forcing Equalizer
Magnitude (dB)
30
20
10
0
-10
Normalized Frequency (×π rad/sample)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Phase (degrees)
100
50
0
-50
-100
Normalized Frequency (×π rad/sample)

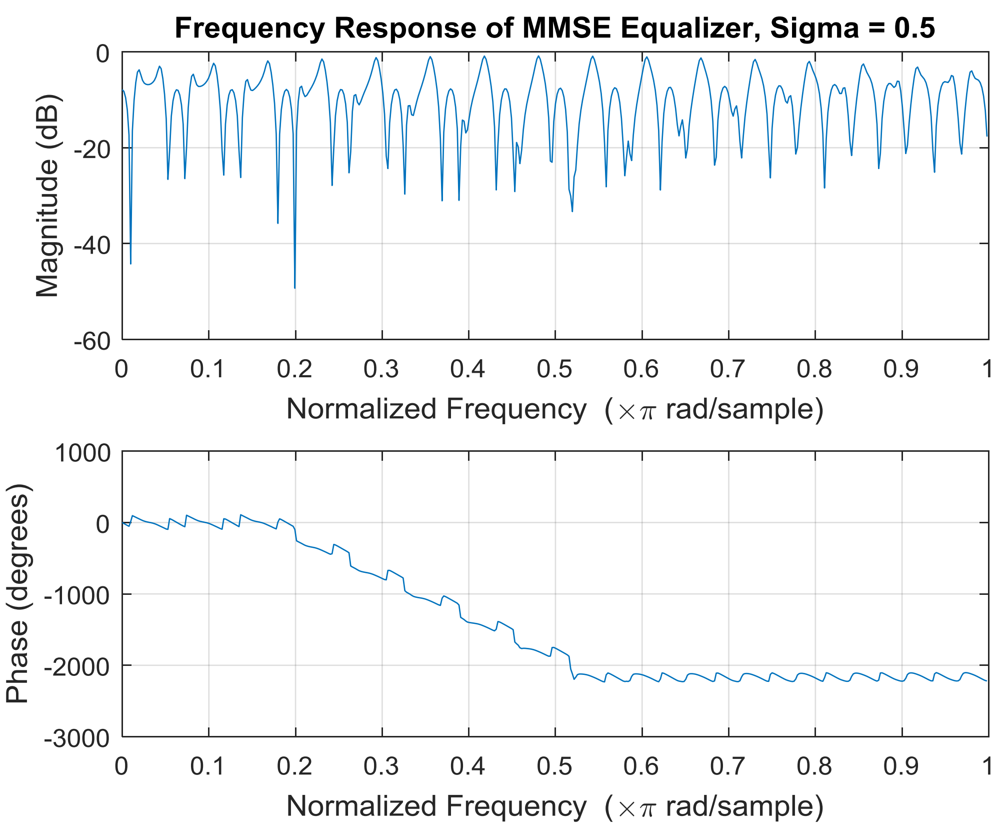
Q9) Eye diagrams at the output of the zero forcing equalizer are shown below. Note again that our equalizer preceded our matched filter. The equalizer removes the echoes from the channel, as can be seen in the following plots. The eye opens back up for both pulses, decreasing ISI and therefore decreasing the error rate of the system.

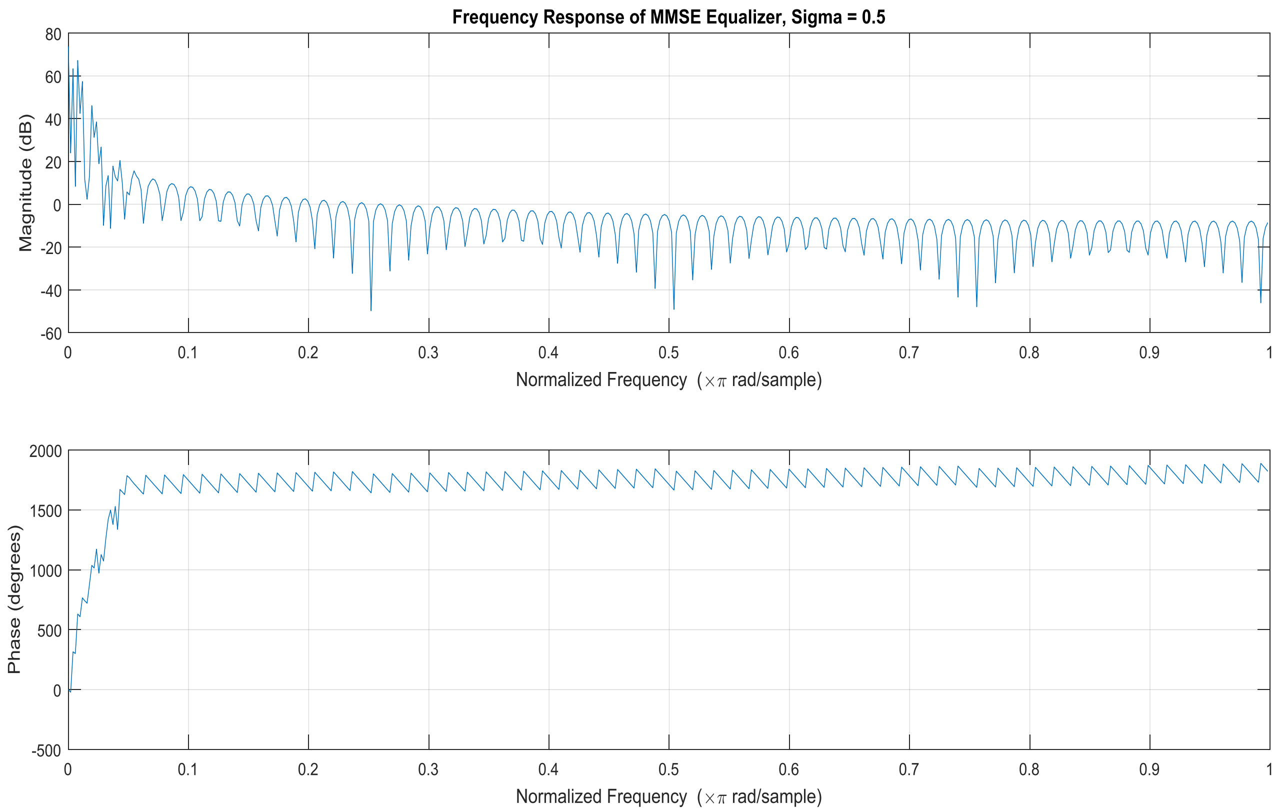


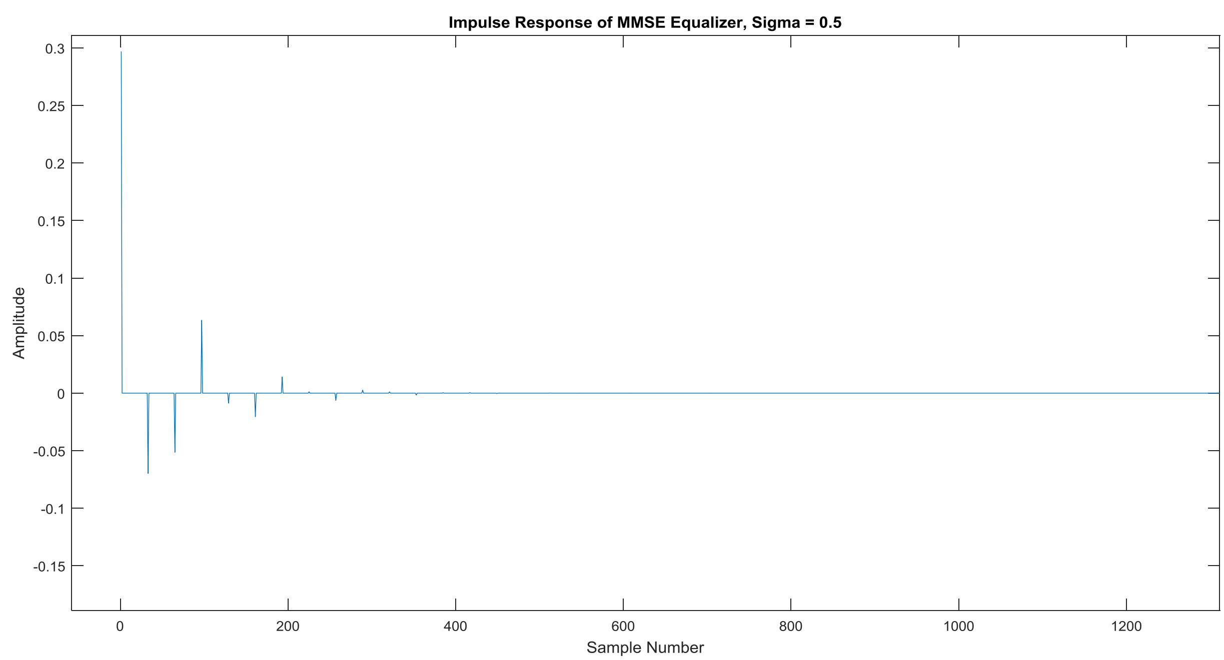



Q10) The frequency response of the MMSE equalizer is plotted below for two values of noise: the first is with zero noise to demonstrate that it is equivalent to the zero forcing equalizer in the noise free case, and the second is to demonstrate what results when noise is added. The addition of noise causes the sharp positive peaks in the frequency response to be suppressed. In the MMSE equalizer frequency response with noise, the gain is below zero for all frequencies, which will ensure stability. This makes the MMSE equalizer much more realistic than the zero forcing equalizer in practical use because noise will always be present in a real communications system. The MMSE equalizer does a better job than the zero forcing does in the presence of noise, because it adapts to the channel and noise power. As you can see in the frequency response of the zero forcing equalizer, a small range of frequencies gets amplified while the rest get attenuated. If there is a lot of noise at this frequency it will also get amplified. In the MMSE equalizer, as the noise power increases, the frequency response approaches that of the channel, however, its phase stays the same. This will attenuate the noise, while still undoing the effects of the channel, making it a better equalizer.

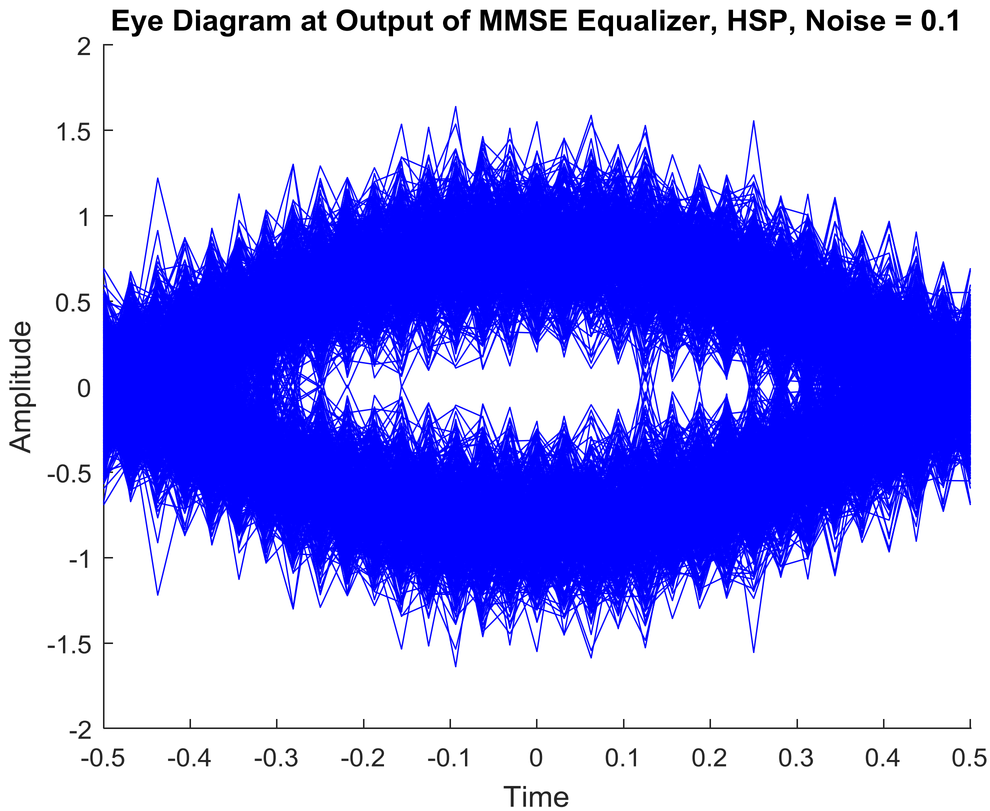

Frequency Response of MMSE Equalizer, Sigma = 0
Magnitude (dB)
30
20
10
0
-10
Normalized Frequency (×π rad/sample)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Phase (degrees)
100
50
0
-50
-100
Normalized Frequency (×π rad/sample)

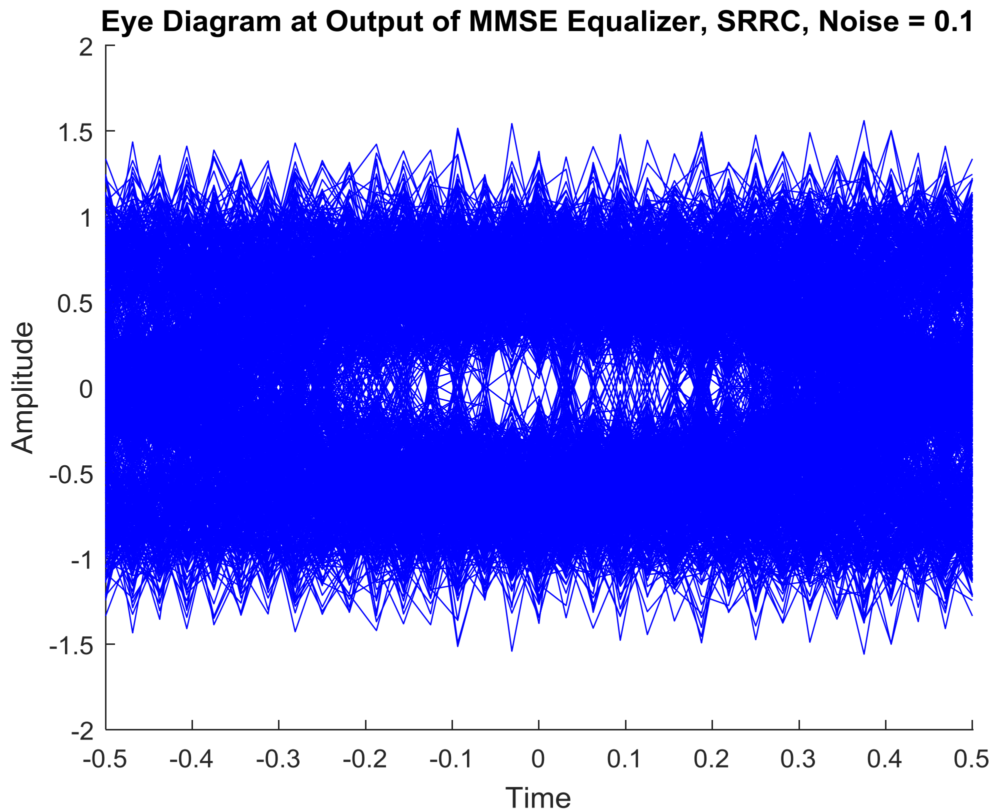

Frequency Response of MMSE Equalizer, Sigma = 0.5
Magnitude (dB)
0
-20
-40
-60
Normalized Frequency (×π rad/sample)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Phase (degrees)
1000
0
-1000
-2000
-3000
Normalized Frequency (×π rad/sample)


Frequency Response of MMSE Equalizer, Sigma = 0.5
Magnitude (dB)
Normalized Frequency (×π rad/sample)
Phase (degrees)
Normalized Frequency (×π rad/sample)


Impulse Response of MMSE Equalizer, Sigma = 0.5
Amplitude
Sample Number

Q11) The eye diagrams for the output of the equalizers are given below. Note that matched filtering has not occurred at this point in the signal chain. The MMSE equalizer causes the eye to open. Compare to the eye diagrams after noise was first added to see the difference. This allows the system to continue to work at higher noise levels by providing a sampling instance in which there is reduced ISI.

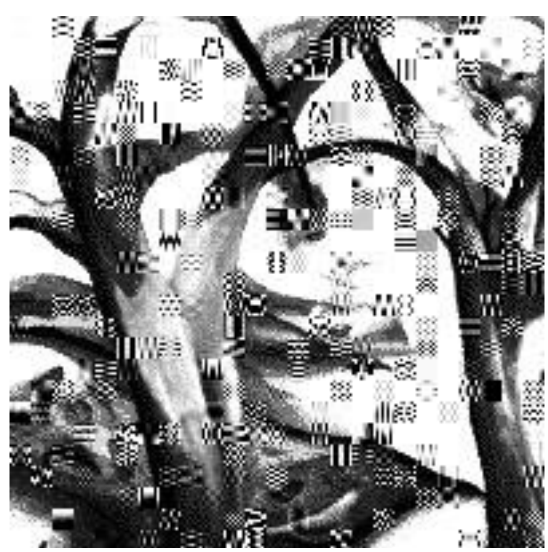





Q12) The following images are the results with perfect reconstruction of the image, fair reconstruction of the image (some errors), and poor reconstruction of the image (lots of errors). This was done for both HSP and SRRC. Extra plots are included to demonstrate that the MMSE Equalizer allows reconstruction at higher noise levels than Zero Forcing.

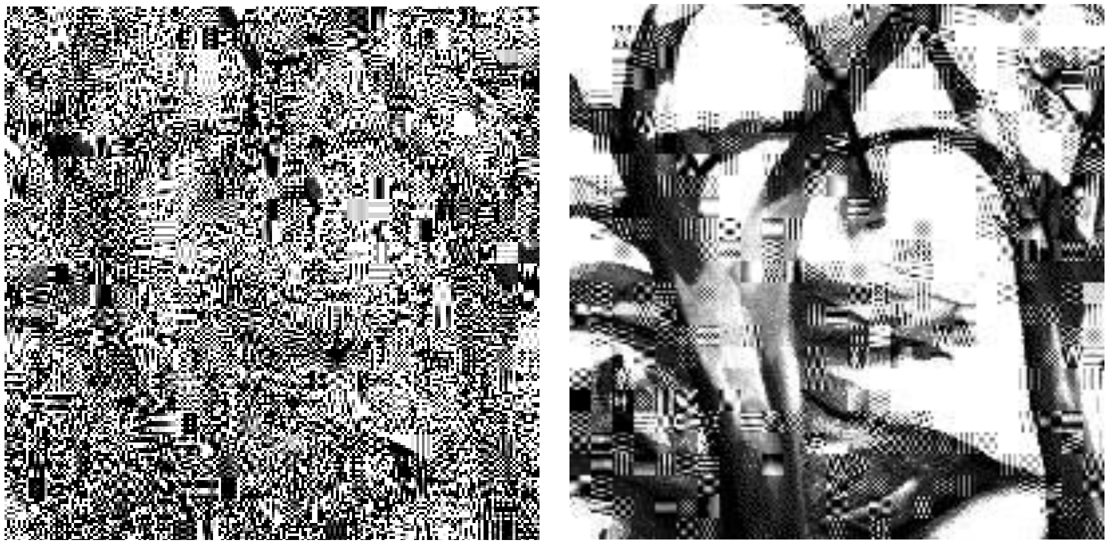
HSP, Zero Forcing EQ, Noise = 0.025

HSP, Zero Forcing EQ, Noise = 0.1



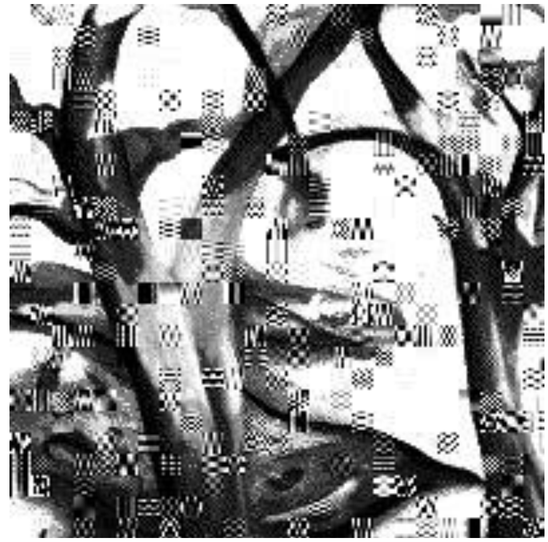
HSP, Zero Forcing EQ, Noise = 0.3

HSP, MMSE EQ, Noise = 0.3

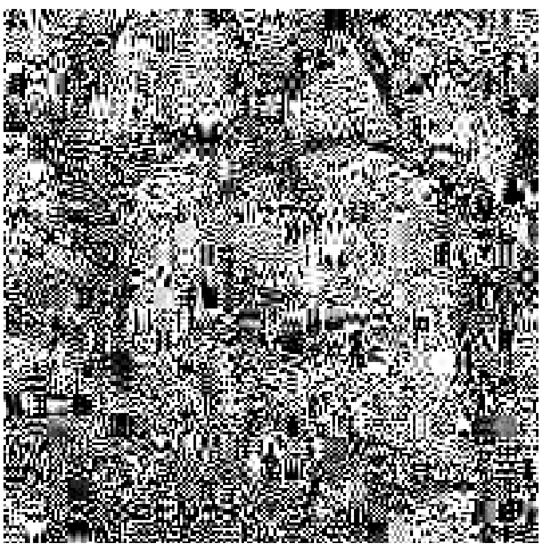


SRRC, Zero Forcing EQ, Noise = 0.025

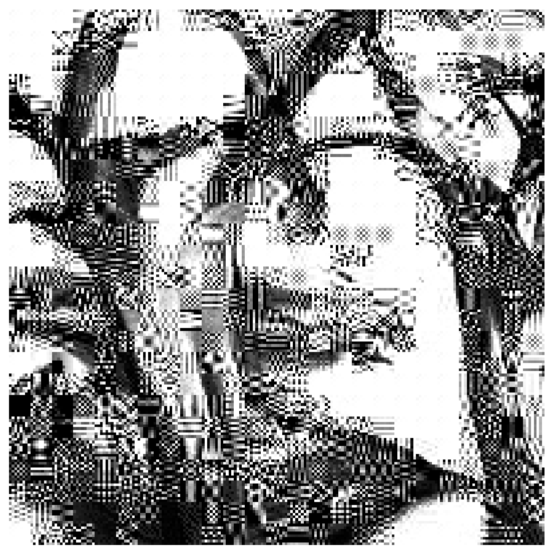
SRRC, Zero Forcing EQ, Noise = 0.1



SRRC, Zero Forcing EQ, Noise = 0.3



SRRC, MMSE EQ, Noise = 0.3

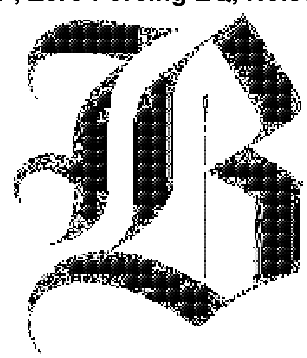


Q13) By transmitting the image using different levels of noise power, the critical $\sigma^2$ for the Zero Forcing Equalizer was determined to be 0.041, while the MMSE was 0.082. HSP was used for both of these measurements. The average power of the HSP signal is 0.5. The SNR is calculated as $SNR_{ZF} = 0.5/0.041 = 12.1591 = 10.9$ dB. For MMSE, the SNR is calculated as $SNR_{MMSE} = 0.5/0.082 = 6.0976 = 7.9$ dB. This means that the SNR must drop to 10.9 dB for errors to appear with the ZF equalizer, while it must drop an additional 3 dB for errors to appear in the MMSE equalizer. This makes sense because the MMSE equalizer accounts for noise while the ZF equalizer does not.

Q14) Two of our own images were chosen for transmission through the channel. Note that since there was error introduced by the ImagePreProcess and ImagePostProcess functions, errors were inherent to the program. Running ImagePreProcess with a picture and using the output called Ztres as the input to ImagePostProcess causes errors, which is illustrated by the following sets of images. In the second set of images with the Boston Bruins logo, it is interesting to note that passing the Ztres matrix produced by the ImagePreProcess function directly into ImagePostProcess results in a completely blank image (as does putting the image through the channel with no noise). However, it can be seen from the second image of that set that adding noise actually helps in discerning the image; the outline of the circle may be seen in the noisy image. The SNR of the first set of images would be infinite, since the noise power is 0. For the second set of images, the SNR is 10*log10(0.5/0.2) = 4 dB.

**Original**

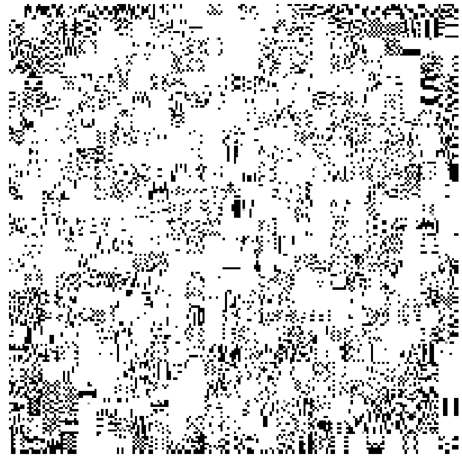
**HSP, Zero Forcing EQ, Noise = 0**



**Output Using Only ImagePre & Post Process**

**Original**

**HSP, Zero Forcing EQ, Noise = 0.2**



**Output Using Only ImagePre & Post Process**

Note: There is a picture above; it came out completely white.

Q15) Nyquist's criterion for zero ISI is that the pulse must be 0 at all bit durations kT, where k is an integer. HSP satisfies this criterion while SRRC does not. Given the success of SRRC in this simulation, this might seem surprising, but since a matched filter was used, the net response is that of a raised cosine (since we have the square root of the raised cosine frequency response multiplied by itself). The raised cosine pulse does satisfy the criterion, which allows SRRC to be an effective pulse shape. Therefore, since the matched filter in our implementation is after the equalizer in the signal chain, we would expect there to be zero ISI at the sampling point. This would not be the case for the output of the equalizer.

Q16) Since the two pulse shapes have the same energy, the pulses should have similar noise performance. What else can we say here? This is supported by the simulation results, which showed that using the same equalizer and noise power resulted in errors appearing in the output images at the same noise power.

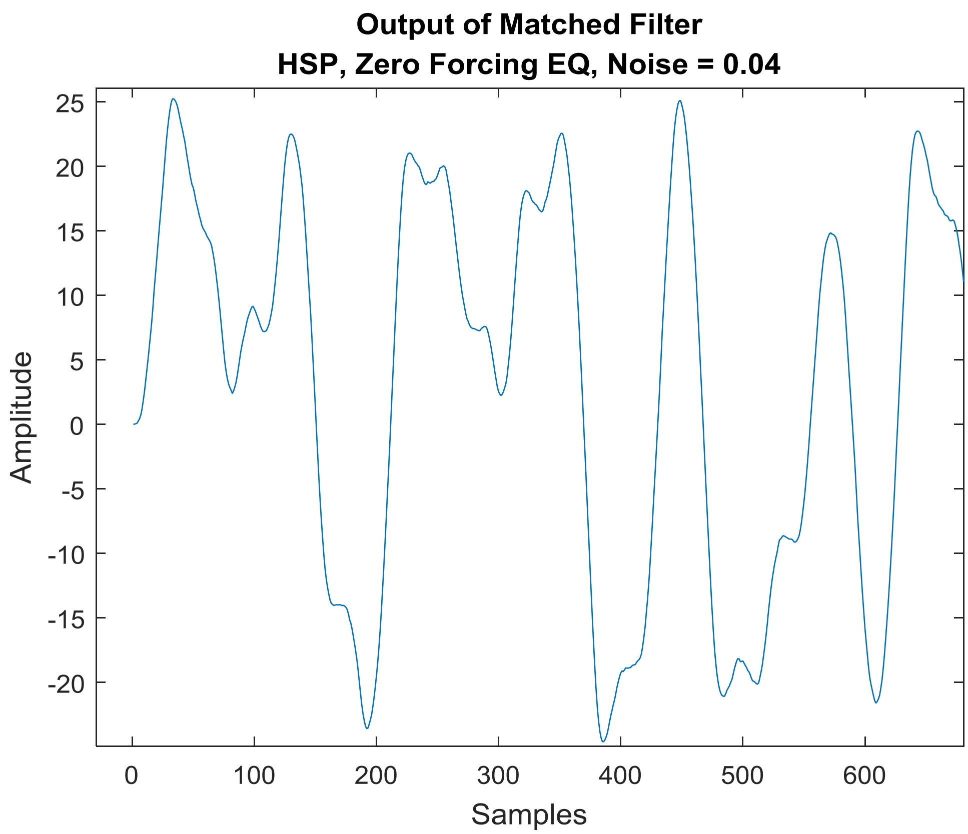
**HSP, Zero Forcing EQ, Noise = 0.04**

**SRRC, Zero Forcing EQ, Noise = 0.04**

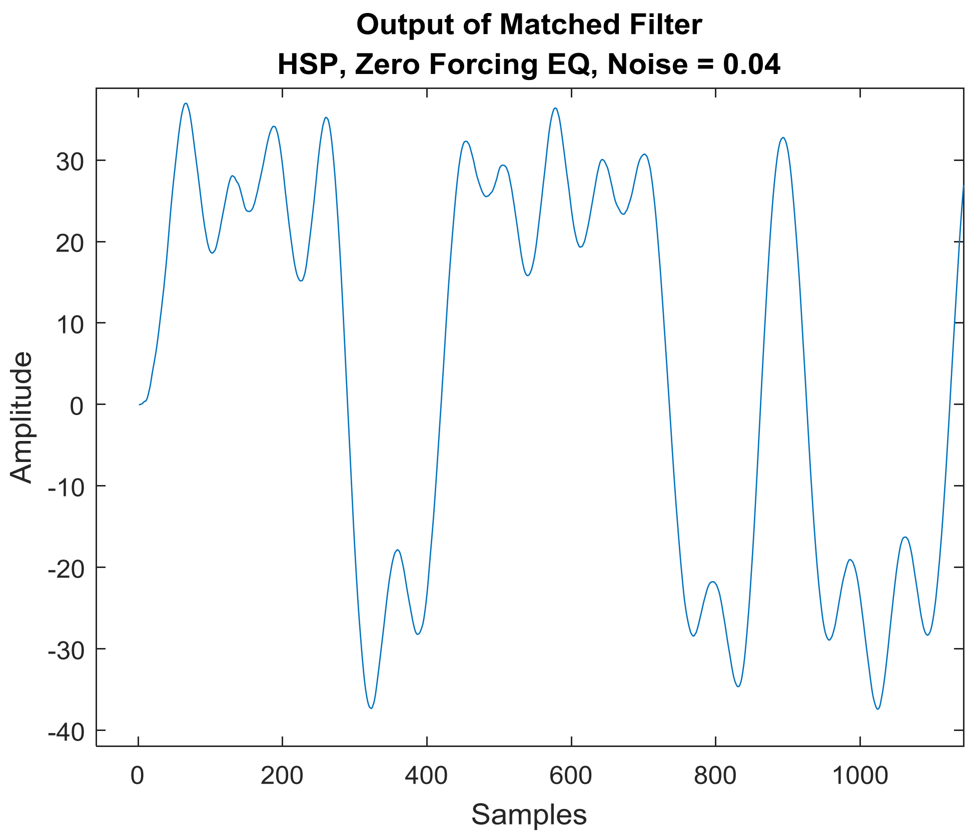
Q17) Plots of the frequency responses of the two pulses may be found in the previous pages. These show that the required bandwidth of the SRRC is smaller than that of the HSP, especially when the truncation length K is increased. This has the effect of reducing the required transmission bandwidth, which is a useful feature in many applications due to the restrictive nature of frequency band allocations.

Increasing T, the pulse length in number of bit durations, improves the performance of the system. The plot below shows the output image after T has been changed from 1 second to 2 seconds. With the same pulse and equalizer as the plot above and to the left, approximately three times more noise power was required for errors to appear. This makes sense because a signal that is changing more slowly would be easier to sample. The outputs of the matched filters (the last stage before sampling) of the system with T = 1 and T = 2 are shown below. For T = 2, the peaks become more defined and the difference between high and low bits becomes more pronounced. This increases resistance to noise.

**HSP, Zero Forcing EQ, Noise = 0.12**


Output of Matched Filter
HSP, Zero Forcing EQ, Noise = 0.04
Amplitude
Samples
25
20
15
10
5
0
-5
-10
-15
-20
0
100
200
300
400
500
600

*T = 1 second*




*T = 2 seconds*

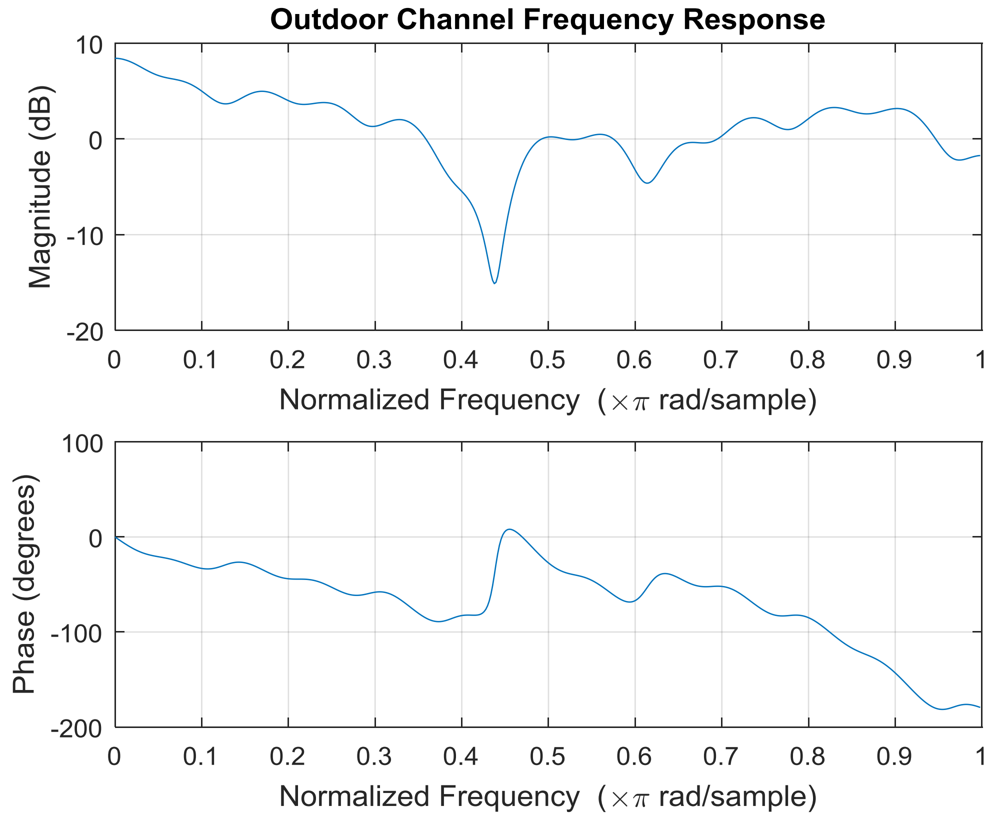
Q18) Based on the above points, and on the fact that literature on the half-sine pulse is more difficult to find than literature on the raised cosine pulse, I would say that the raised cosine pulse is superior. The two pulses have approximately the same resistance to noise, but the raised cosine pulse offers the advantage of requiring a smaller transmission bandwidth for the same power. This is a desirable quality in a pulse shape due to the limitations on bandwidth in communications systems.

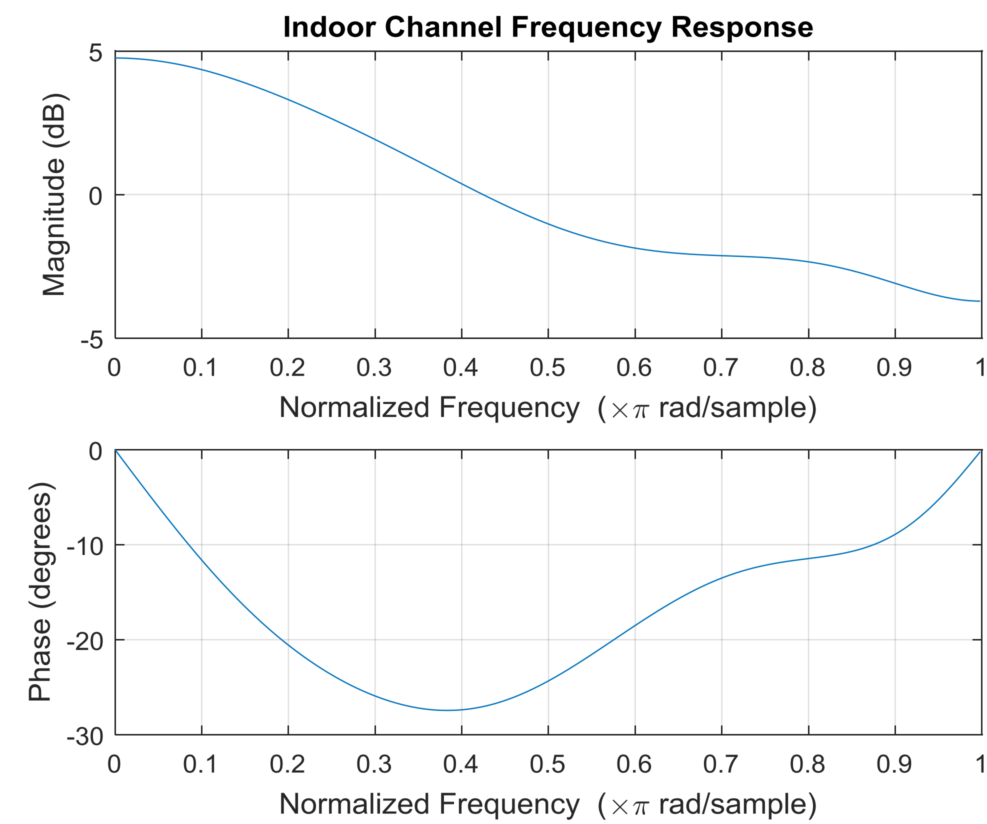
Q19) The program worked seamlessly after changing the channel coefficients that determine its frequency response and the echoes that result. One output image from each channel is included below. Changing the channel coefficients enabled the system to work at much higher levels of noise. Changing the value of N did not have a noticeable effect on system performance, even at levels of noise that caused the system to begin to break down.

The channel frequency responses are plotted below. Note that the outdoor channel has a deep fade at the normalized frequency of 0.43 x π rad/sample. Attempting to broadcast near this frequency would attenuate the signal, degrading the system performance. The indoor channel frequency response is smoother and drops off more consistently.

*a: Outdoor Channel, HSP, Noise = 0.75*

*b: Indoor Channel, HSP, Noise = 0.75*


Outdoor Channel Frequency Response
Magnitude (dB)
10
0
-10
-20
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)
Phase (degrees)
100
0
-100
-200
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)


Indoor Channel Frequency Response
Magnitude (dB)
5
0
-5
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)
Phase (degrees)
0
-10
-20
-30
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Frequency (×π rad/sample)