ISI suppression in modulation schemes
Third year project I
Bachelor in communication Engineering

Group members:

Nawaf Obaid Saif 1241
Mohammed Al-Qadi 1238
Abdulrhman Yousif 1205

Supervisor
Dr S. Leggof

Submission Date:20th May 2003

Acknowledgments:

A special thank to Dr Leggof our supervisor for helping us during the processes of this project. Also we would like to thank Dr Shihab Jimaa the general supervisor for his advices and comments. Finally we would like to thank all the college staff that helped us during this project.

Abstract:

This project is about the ISI suppression in modulation schemes it's a digital process from the beginning till the end of the project. Its well known that digital communications use square pulses as signals to be transmitted, that would affect the bandwidth because what will happen is that the bandwidth required to transmit such pulses will be infinite. That's why filters are introduced in this case to limit the bandwidth because all communication channels are band limited.

Introduction:

Goals and Objectives:

The aim of this project is to demonstrate the error performance of a practical system over a limited bandwidth channel can be the same as theoretical assumption of infinite bandwidth channel.

The objective of this project is divided in to two parts:

First part:

- **Designing a block system using square pulses as a source and gaussian channel with different values of variance and then observe the bit error performance of this block diagram.**

Second part:

- **Introducing low pass filter to the system to get a limitation of spectrum of the transmitted signal. The raise cosine filter was used as a low pass filter because of its special specification in filtering.**

- **Some simulation will be run to evaluate the bit error rate performance, in order to show the similarity in bit error performance but the difference in the spectrum occupancy.**

The usage of MATLAB in this project:

- **Using the command simulink in MATLAB many communication systems can be built and implemented to see the output and compare the practical part with the theoretical part.**

- **Also MATLAB allow us to compare the results of the block system with out filtering and with filtering in terms of bit error performance and spectrum graph to show the spectrum occupancy.**

- **Introducing every element of the block system and its function in order to achieve what is wanted in the aim and objectives.**

The report will be divided in to three parts: theoretical part, practical part and results and discussion, these parts will be in the body of the report. The last part will be a brief conclusion about the project.

Chapter 1: The theoretical part

1.1 General view of ISI

Many people's concept of digital communications is one of nice square voltage pulses representing 0s and 1s being passed over a piece of cable or perhaps a radio system. In actual fact, this is rarely how digital information is sent.
- Square pulses become rounded pulses when passed through a channel with finite bandwidth.

- **The filtering effects of a channel have an impact on streams of data pulses – bits and symbols.**
- **This introduces intersymbol interference (ISI).**
- **With any practical channel, the inevitable filtering effect will cause a spreading (smearing) of individual data symbols passing through the channel.**
- **For consecutive symbols, this spreading causes part of the symbol energy to overlap with neighboring symbols.**
  - Causing intersymbol interference (ISI). Additionally, filtering in the transmitter or receiver units themselves may also introduce ISI degradation.

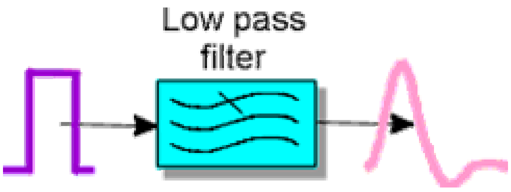


Figure 1: Low pass filter

- **ISI degrades the bit and symbol error rate performance.**
- **ISI can lead to detection error, even in noise free channels.**
- **Systems must be designed carefully.**

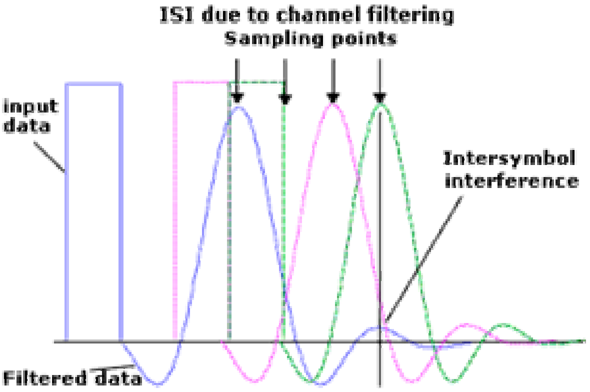


Figure 2: ISI effect

1.2 ISI and Raised Cosine Filter

Why do they call it raised cosine? Because the following response has a cosine function in the frequency domain, although other many other trigonometric representations of this equation that do not have the cosine-squared term, so it is not always clear why these are called raised cosine.

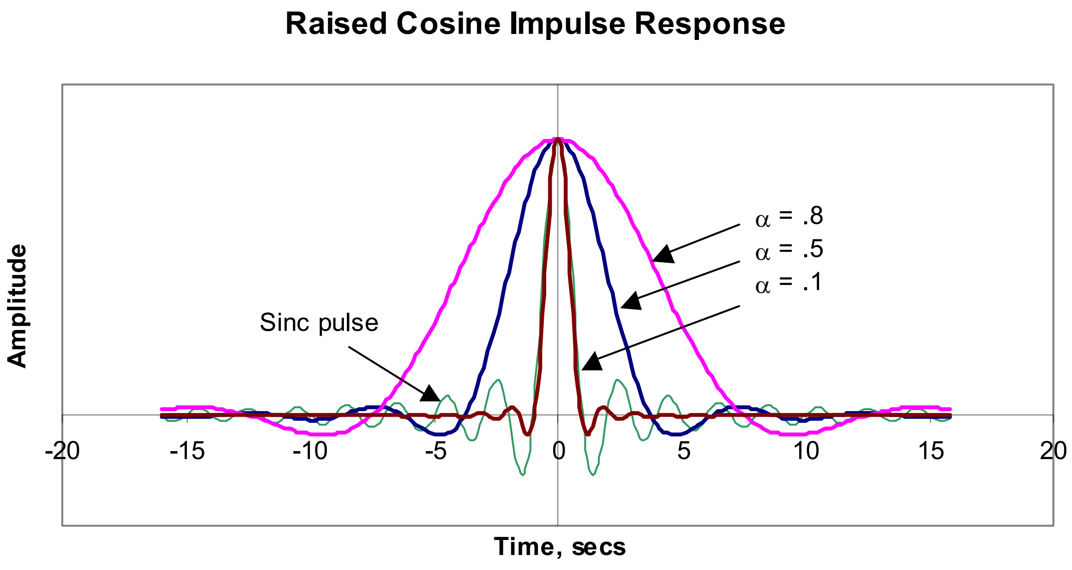


Figure 3: Raised Cosine Impulse Response

Inter Symbol Interference (ISI) and raised cosine filtering

Inter-symbol interference (ISI) is an unavoidable consequence of both wired and wireless communication systems. Morse first noticed it on the transatlantic telegraph cables transmitting messages using dots and dashes.

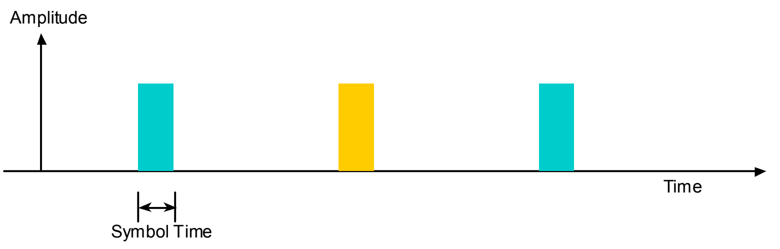


Figure 4: Sent Signal

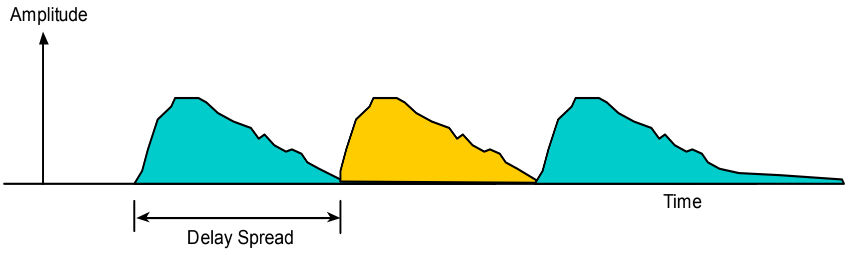


Figure 5: Received Signal

During these early attempts at transmission, it was noticed is that the received signals tended to get elongated and smeared into each other. A short pulse to represent a dot was received as a much-smeared version of the same thing. The problem appeared to be related to the properties of the medium used and the distance of signal travel. To counter this undesired effect, intermediate repeating stations were established and ways had to be devised to reduce this smearing.

The following figure shows each symbol as it is received. We can see what the transmission medium creates a tail of energy that lasts much longer than intended. The energy from symbols 1 and 2 goes all the way into symbol 3. Each symbol interferes with one or more of the subsequent symbols. The circled areas show areas of large interference.

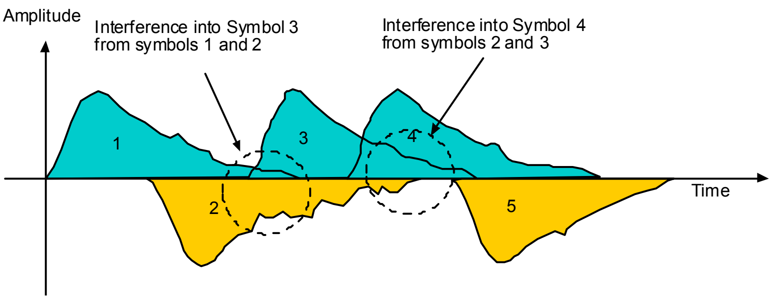


Figure 6: Spreading effect

The figure in the next page shows the actual signal received as the sum of all these distorted symbols. Compared to the dashed line that was the transmitted signal, the received signal looks quite bad. The dots show the value of the amplitude at the timing instant. For symbol 3, this value is approximately half of the transmitted value, which makes this particular symbol is more susceptible to noise and incorrect interpretation.

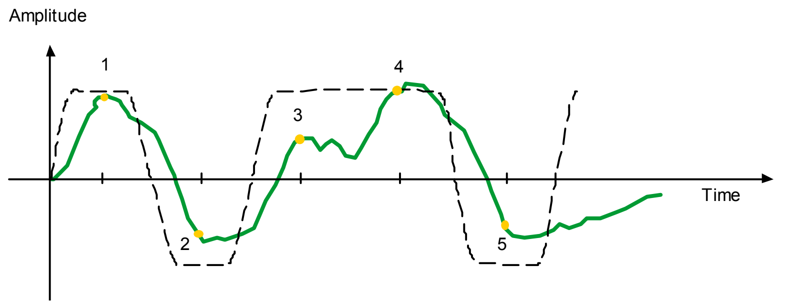


Figure 7: The transmitted signal (doted) and the Received Signal

What can you do about ISI?

The main problem is that energy, which we wish to confine to one symbol, leaks into others. So one of the simplest things we can do to reduce ISI is to just slow down the signal. Transmit the next pulse of information only after allowing the received signal has damped down, as shown below. The time it takes for the signal to die down is called delay spread, whereas the original time of the pulse is called the symbol time. If delay spread is less than or equal to the symbol time then no ISI will result, otherwise yes. Slowing down the bit rate was the main way that ISI was controlled on those initial transmission lines. Then faster chips came and allowed us to do signal processing controlling ISI and transmission speeds increased accordingly.

Slowing down the data rate is an easy but an unacceptable solution. What else can we do to counter ISI? The main tool used to counter ISI is pulse shaping. How can pulse shaping help control ISI? The secret lies in the digital demodulation process used. When the timing pulse slices the signal to determine the value of the signal at that instant, it does not care what the signal looked like before or after it. So if there was some way we could keep the symbols from interfering in such a way that they do not affect the amplitude at the slicing instant, we can counter ISI. Successfully. In the next page the method of pulse shaping to degrade the ISI is shown clearly.

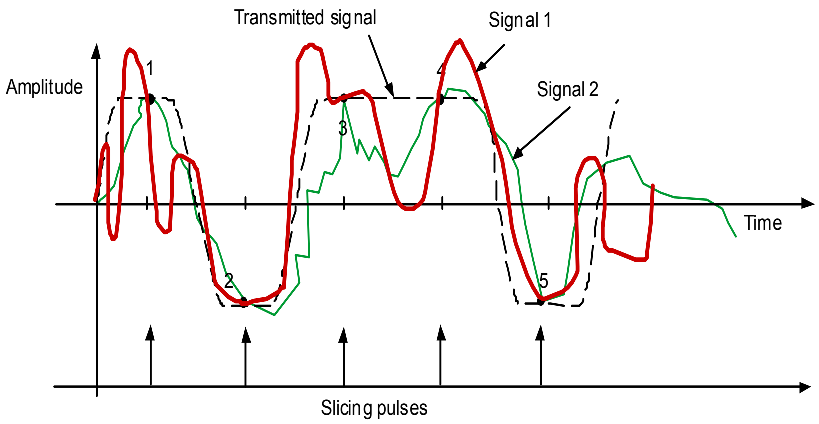


Figure 8: ISI Degradation.

Chapter 2: Practical Part

2.1 The Block System without filtering:

The system block was built using MATLAB and the system consists of an information source, modulator, AWGN channel, demodulator and bit error rate calculation (BER). The scope was used to show the signals that have been transmitted through the system.

The block system is consisted of the following parts:

1. **The information source.**
2. **The modulator.**
3. **AWGN channel.**
4. **The demodulator.**
5. **Bit error rate calculation.**
6. **The time scope.**
7. **Spectrum scope.**

An overview of the block system in MATLAB:

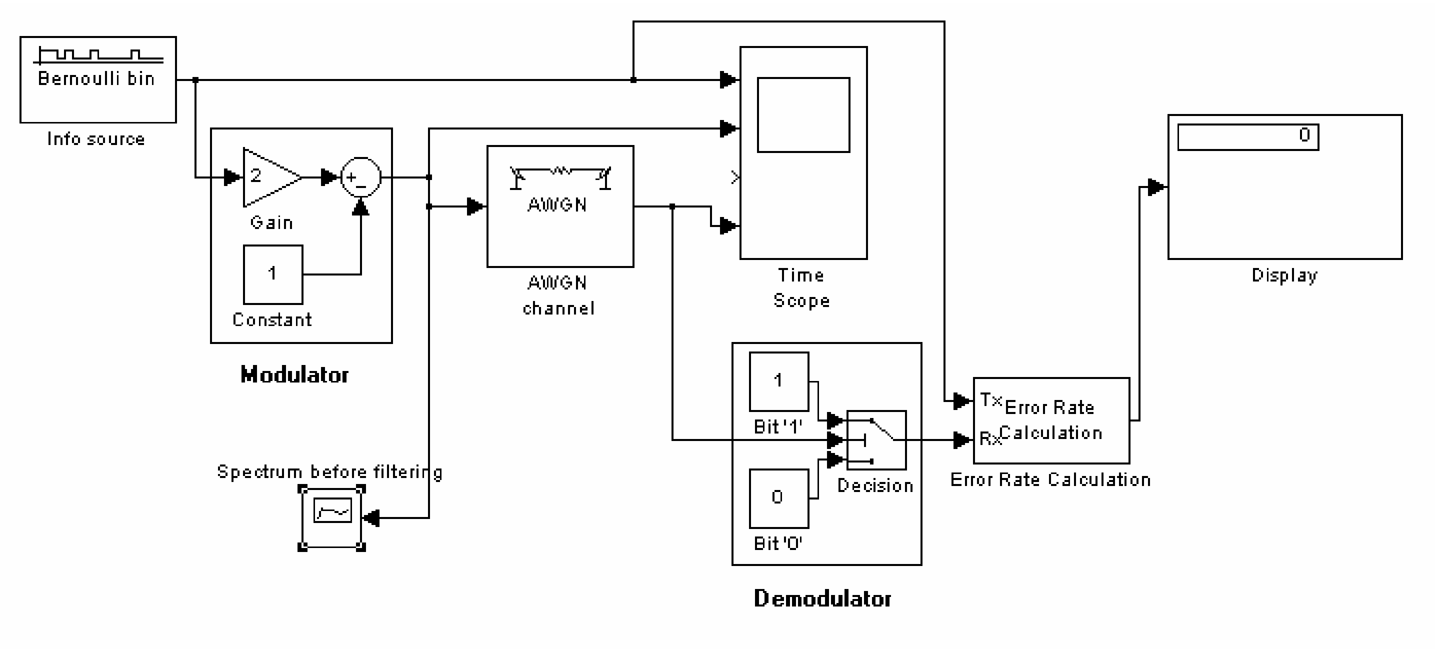


Figure 9: Block System without the Filter

2.2 Description of each block in the system:

1.Bernoulli Binary Generator:

Generate a Bernoulli random binary number.
To generate a vector output, specify the probability as a vector. That generator will generate square pulses.

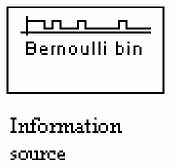


Figure 10: Information Source.

2. Modulator:

Which consists of the following parts:
<u>Gain:</u> Element-wise gain (y = K.*u) or matrix gain (y = K*u or y = u*K).<u>Constant:</u> Output the constant specified by the 'Constant value' parameter. If 'Constant value' is a vector and 'Interpret vector parameters as 1-D' is on, treat the constant value as a 1-D array. Otherwise, output a matrix with the same dimensions as the constant value. <u>Sum:</u> Add or subtract inputs.

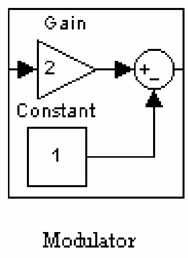


Figure 11: The Modulator.

3. AWGN Channel: **AWGN Channel: Add white Gaussian noise to the input signal. The input and output signals can be real or complex. This block supports multi channel input and output signals as well as frame-based processing.**

When using either of the variance modes with complex inputs, the variance values are equally divided among the real and imaginary components of the input signal.

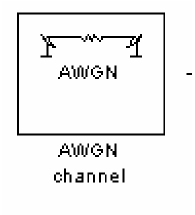


Figure 12: Gaussain Channel.

4. Demodulator: It consists of the following parts: Bit '1' Bit '0'Decision: it decides wither the signal is 1 or 0(i.e. if we had a signal with 0.7 the decision will decide that the signal is 1 and if we had a signal with 0.3 the decision will decide 0)

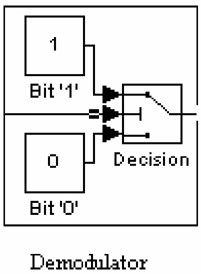


Figure 13: The Demodulator.

5. Bit Error Rate calculation (BER) Error Rate Calculation: Compute the error rate of the received data by comparing it to a delayed version of the transmitted data

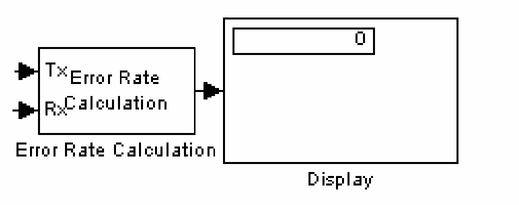


Figure 14: Error Rate Calculation.

6. Time Scope: It will display the signals that have been sent through the system, the display graphs will be introduced in the following part. Many displays can be shown in the time scope before the AWGN and after the AWGN.

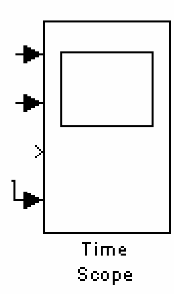


Figure 15: Time Scope.

7. Spectrum scope:
It displays the spectrum of the signals that introduced through the system

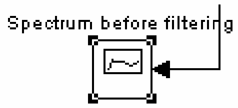


Figure 16: The Spectrum Before Filtering.

2.3 Results gained from Block System.

2.3.1 The Bit Error Rate (BER)
The information source sent number of bits of 6500 and the variance value was varied from 0.5 → 1.5 in step of 0.5 and the results obtained from the BER calculation block are as the following:

Number of Bits	Variance	Number of errors	BER
6500	0.5	487	0.07491
6500	1.0	1015	0.1561
6500	1.5	1373	0.2112

Table 1: BER Results Before Filtering.

It's clear from the table that the BER was efficient with number of errors above 100 and that was needed to have number of errors above 100.

2.3.2 The Time Scope Graphs.
That graph is an example of the first (variance = 0.5) and the rest are the same as this graph. Its clear that the first part is the signal that has been sent through the system,

the second one is the signal after the modulator where we can see that we have a signal with 1 and –1 and that is the job of the modulator. The final graph is the signal with noise (i.e. after the AWGN channel) then the signal at the end will be back the same as before after the demodulator.

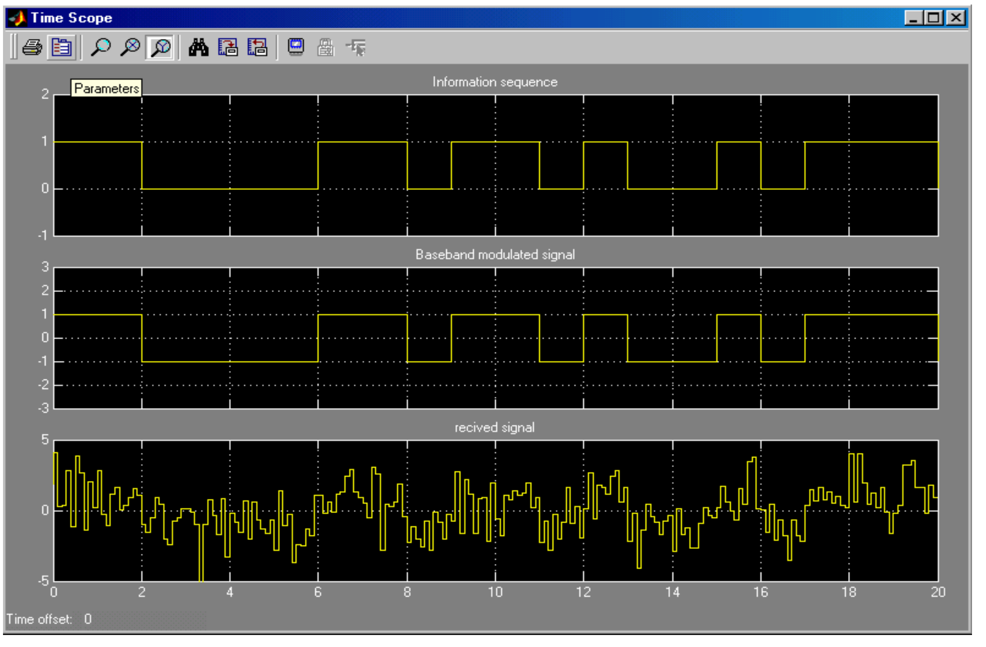


Figure 17: Time Scope Graphs Before Filtering.

2.3.3 The Spectrum Graph.

The spectrum of the signal is shown in the following graph, it's clear that we are having a sinc function with an infinite bandwidth and many harmonics introduced to the spectrum that's not practical at all here where the next part of our project will play its part, which is introducing the filtering effect.

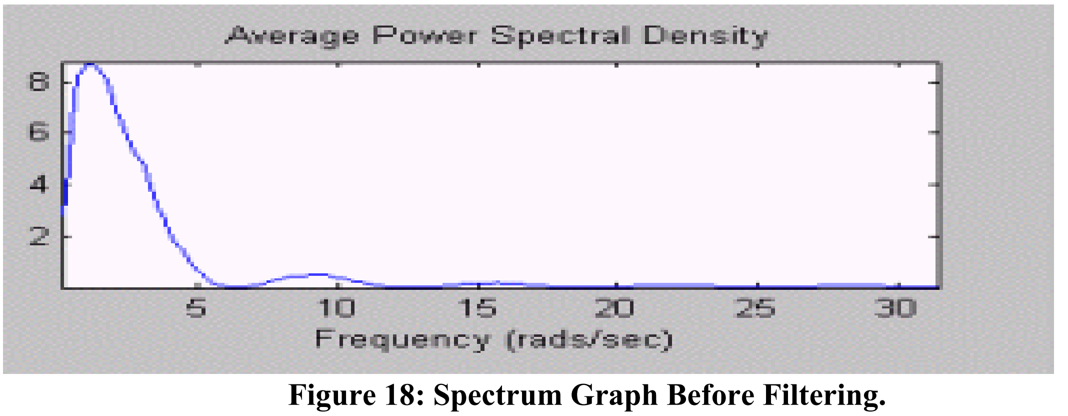


Figure 18: Spectrum Graph Before Filtering.

2.4 The block system with filtering:

The system block was built using MATLAB and the system consists of an information source, modulator, Raised Cosine Filter (RC), AWGN channel, demodulator and bit error rate calculation (BER). The scope was used to show the signals that have been transmitted through the system.

The block system is consisted of the following parts:

1. The information source.
2. The Modulator.
3. Raised Cosine Filter (RC)
4. AWGN
5. The Demodulator.
6. Bit Error Rate Calculation.
7. The time scope.
8. Spectrum Scopes.

An Overview of the Block System in MATLAB:

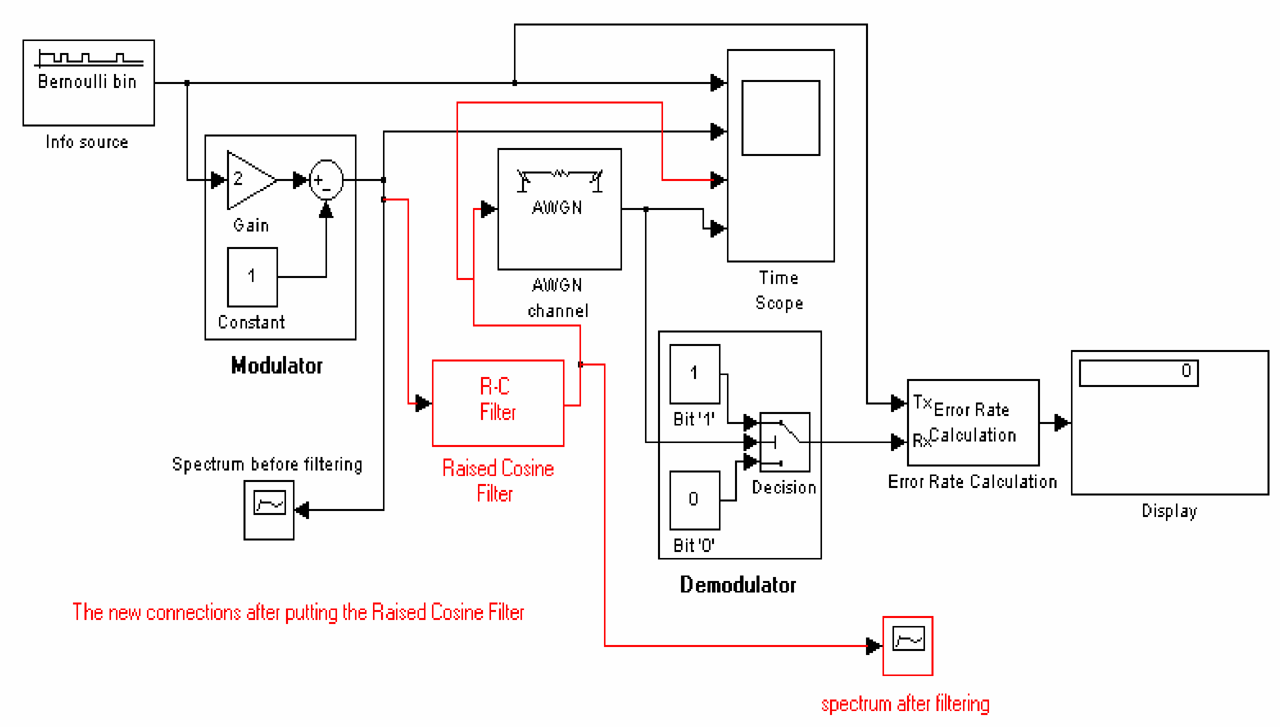


Figure 19: The Block System After Filtering.

The above Block System is different than the previous Block System by the following points:

- **The Raised Cosine filter was added into the system.**

- New connection from the Filter to the Time Scope was established.
- Spectrum Block was added in order to show the spectrum for the signal after filtering.

2.4.1 Description of each block added to the System.

1.R-C Filter (Raised Cosine Filter):

Special kind of filters that filters the signal without the ISI, in other words it reduces the ISI.

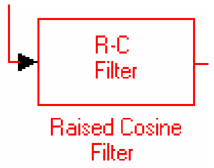


Figure 20:The Raised Filter Block.

2.spectrum scope (after filtering):
Used to show the spectrum of the signal after filtering.

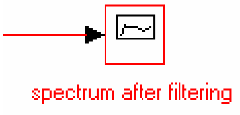


Figure 21:The Spectrum Block After Filtering.

2.4.2 The Bit Error Rate (BER)
The same here the Number of bits is 6500 and the variance value was varied from 0.5 → 1.5 in step of 0.5. The results are as the following:

Number of bits	Variance	Number of errors	BER
6500	0.5	479	0.07371
6500	1.0	970	0.1493
6500	1.5	1282	0.1973

Table 2:BER Results After Filtering.

2.4.3 The Time Scope Graphs.

A raw is added here which is concerned with the signal after the raised cosine filter, its clear that the signal after the filter is smooth and rounded. In this raw we can see that at the same sampling line we will get a value of –1 and +1 always over this raw, which is the same as the square pulses without the filter but the shape is more practical compared with the square pulses.

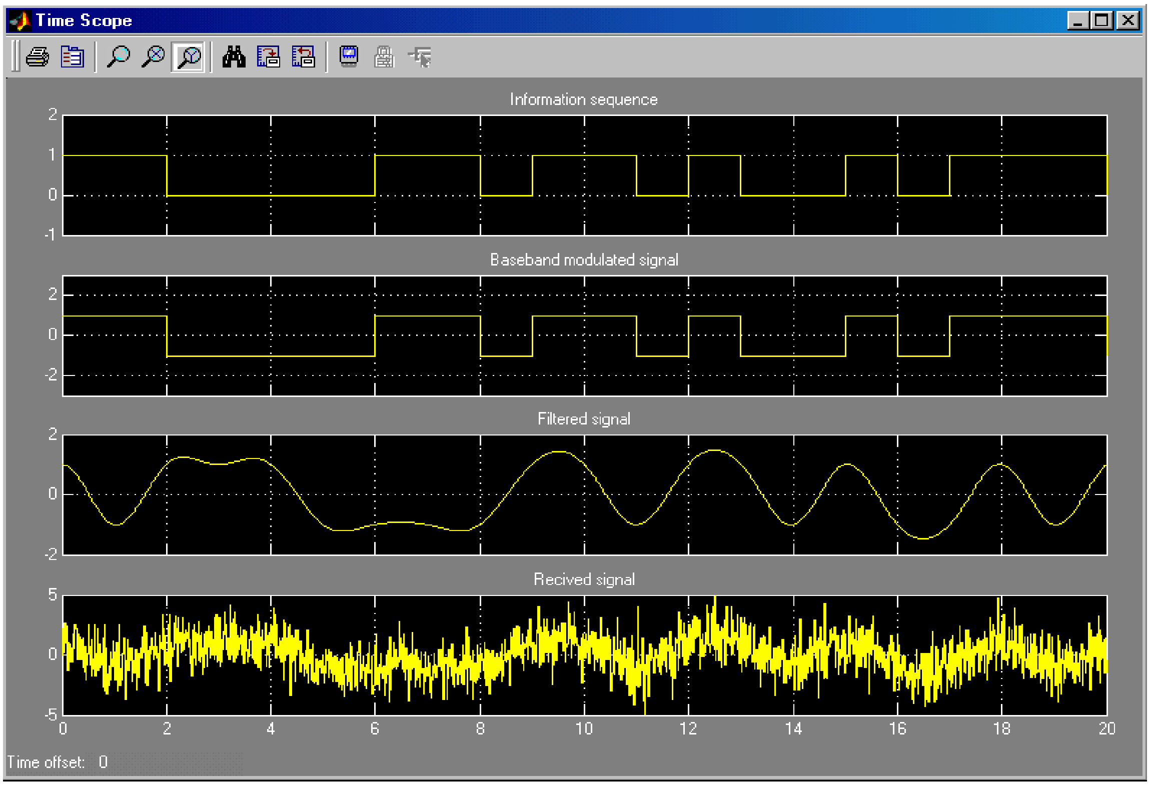


Figure 22: Time Scope Graphs After Filtering.

2.4.4 The Spectrum Graph

Big difference between the spectrum graph with and without filtering as shown in the following graph, all the differences and comparisons will be discussed in details in the following chapter.

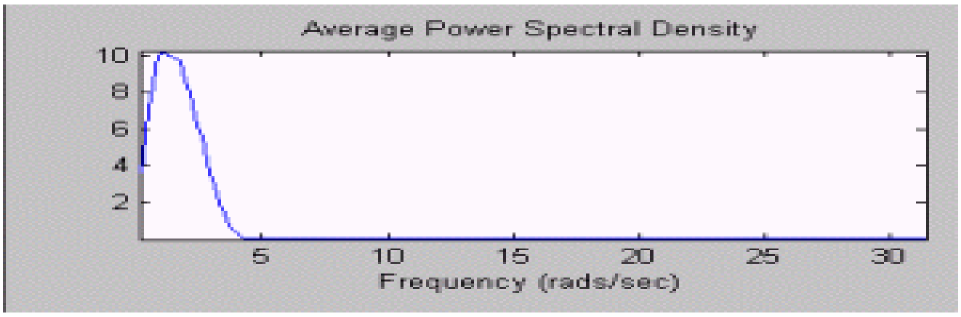


Figure 23: Spectrum Graph After Filtering.

Chapter 3: Results and Discussion.

3.1 Comparison in the BER Before and After Filtering:

Its clear from the table that the Bit Error Rate is different before and after the filtering the results in fact should be the same as the theory which says that we should obtain the same bit error rate before and after the filtering. That means using the filter in practical life can give us a better BER result compared with the system before filtering.

		Before filtering		After filtering	
Number of bits	Variance	Number of errors	BER	Number of errors	BER
6500	0.5	487	0.07491	479	0.07371
6500	1.0	1015	0.1561	970	0.1493
6500	1.5	1373	0.2112	1282	0.1973

Table 3: Comparison Results in BER.

3.2 Comparisons in the Time Scope Graphs.

As mentioned before the filter provides us with a signal as in the figure 25, where we can see the characteristic of the –1 and +1 is still there but with a smoothen signal, that smooth shape is done by the Raise Cosine Filter. The square pulses before the filtering effects are so sharp and very difficult to use in the practical real life, that's why we used the filter to provide us with a practical signal.

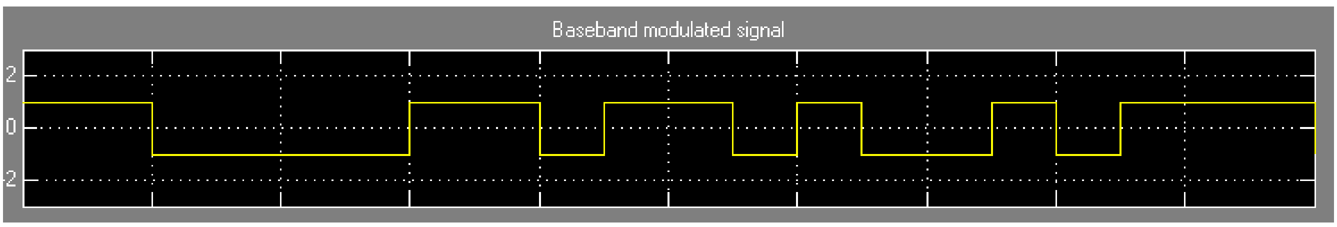


Figure 24: Signal After Modulation.

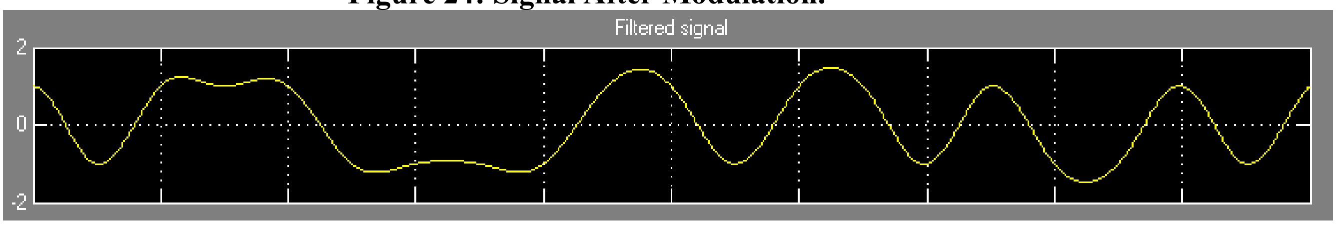


Figure 25: Signal After RC filter.

3.3 Comparisons in the Spectrum Graphs.

In the figure bellow the spectrum graphs are shown before and after the filtering, which in the left graph the spectrum before the filtering in a sinc function with the harmonics, however in the right graph the spectrum can be observed with a smooth sinc function with out the harmonics that means the system with the filter is more practical in terms of the signal output.

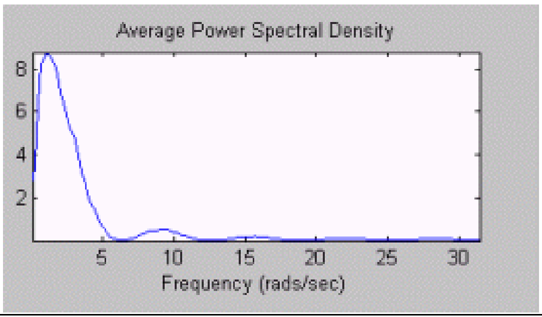
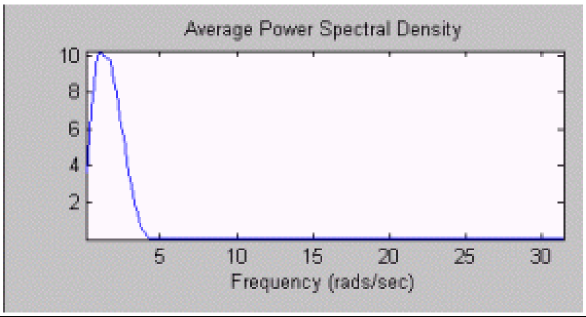
Before filtering	**After filtering**
	

Figure 26: Comparison in Spectrum Graphs.

Conclusion:

- **At the end of this project we got a good overview over the MATLAB software specially in implementing communication blocks.**

- **The MATLAB software crashes for large sent number of bits, so its recommended to have better results of BER to do many readings and then take the average of these results.**

- **The results of the theoretical part and the practical part weren't identical and that's due to the error occurred in the simulations.**

- **The BER before and after filtering should be the same, but we had slight change in the number of the BER and that's might be because of the machine that we are using.**

- **The Only Filter that has the ability to filter the ISI is Raised Cosine Filter, that's why the Filter was introduced into the System.**

- **The Raised Cosine Filter differs according to the value of α where the filter that has α=0 the filtering effect is sharp, while there is the one with α=1 it has a smooth effect of filtering. The one which we used is the α=0.5 which has a medium filtering effect.**

- **It can be recommended in next year to cover the ISI with different RC filter, in other words different values of α.**