COMMUNICATION STRAND

YEAR 3 PROJECT

Performance of a communication scheme protected by repetition code

Supervised by: *Dr. Stephane LeGoff*

© Submitted by:

Ahmed Ali Al-Khallafi ID#: 1209

Mohamed Ali Al-Nuaimi ID#: 1231

Mohamed Ahmed Al-Jarwan ID#: 1233

ACKNOWLEDGMENTS

We would like to express our appreciation for the contribution of **Dr. Stephane LeGoff** who supervised our project and given us the original ideas about this project. We also indebted to the staff for their advice and assistance in this project. Also we would thank **DR.Shihab Jimaa** for all his instructions and cooperation through out the project.

In particular, we wish to thank every one for their helpful comments and suggestions concerning any problems, which we faced in our project. We also express our thanks for constructive feedback and we also appreciate the suggestions made by our friends in the college.

Finally, we must thank **Etisalat College** for supporting us with equipment and a place to carryout in our project. We hope you enjoy this project as much as we enjoyed finishing it.

Table of Contents

Table Of Figures

ABSTRACT

The aim of this project is to design a system that is protected by a repetition code. This repetition code enhances the overall performance of a communication system by repeating the code and transmitting it more than once for more accuracy and hence the detector detects the possible errors.

The project started by identifying the requirements by asking the supervisor what the expected final product is. After that, a specification of the system is compiled in such a way that it tells an approximate scenario of the coming steps. Then, designing the system according to the requirements and specifications in order to implement them and then testing them.

This system is required to simulate the transmission of binary sequence. The system is implemented using **SimuLink** software available inside the **Matlab** software. This is done by sending a random Binary Sequence as a source of information bits that is transmitted through an Additive White Gaussian Noise (AWGN) channel to the detector at the other end of the system.

The Bit Error Rate (BER) of the transmitted sequence is proportional to the variance of the AWGN channel. The BER traced by the system can be minimized through several techniques; here the repetition-coding scheme is used to reduce the BER.

Chapter 1

1.0 INTRODUCTION

1.1 Objectives:

This report has been written in order to show detailed information of the performance of a communication scheme protected by repetition code. The report aims to design a communication system that indicates how errors could be minimized using repetition code.

The repetition code is supposed to offer very high accuracy levels of transmission for digital (Binary) data that is widely used in telecommunication system nowadays.

This report is written to clarify one possible technique to enhance the possible errors that occurs during transmission and correcting them using the hamming distance error correction technique.

It shows detailed information not only about how the different parts of the system are connected but also it describes how and why it was implemented.

1.2 Overview of the report:

This report contains many chapters. Every chapter is organized in such a way that the first page consists of a list of sections that will be described. Then, in the following pages, an introduction of the part (or block) that is to be discussed.

After the introduction there is some theory background that helps to understand how this particular block can be implemented.

The implementation of the block is described in detail. Finally, after these chapters, chapters of testing and discussion will be introduced.

1.3 specifications:

Part1: -

-Sending a single Binary Sequence generated in the transmitter.

-Detecting the transmitted signal and hence calculating the Bit Error Rate.

Part2: -

-Inserting a rate-1/3 repetition encoder inside the transmitter.

-Insert the corresponding decoder inside the receiver, which has to be designed using some logic gates.

-Insert an additional BER meter to determine the BER before and after detection.

Chapter 2

2.0 Part One.

2.1 Theories and simulations of part one:

2.1.1 The Block Diagram of part one

The initial block diagram of the general communication system model of the first part of this project is shown in the block diagram in figure1, below:

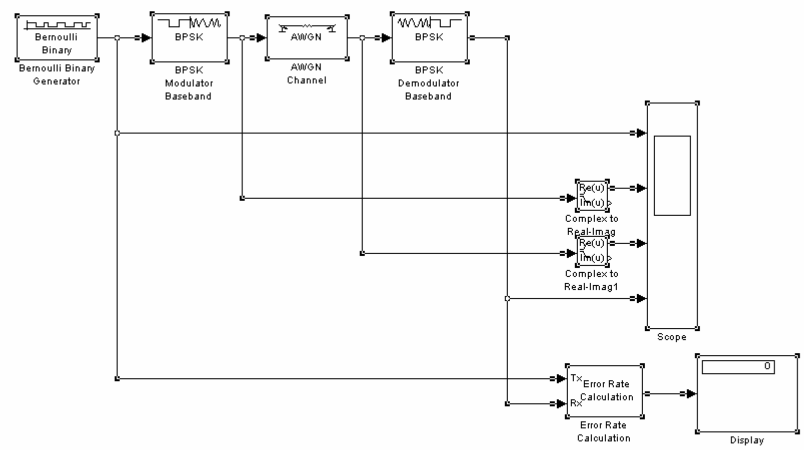

Figure 1 - Generic communication system model

The Block Diagram consists of the following major elements:

1- Bernoulli Binary Generator

The Bernoulli Binary Generator block generates random binary numbers using a Bernoulli distribution. The Bernoulli distribution with parameter p produces zero with probability p and one with probability 1-p. The Bernoulli distribution has mean value 1-p and variance p(1-p). The Probability of a zero parameter specifies p, and can be any real number between zero and one.

2- BPSK Modulator Baseband

The BPSK Modulator Baseband block modulates using the binary phase shift keying method. The output is a baseband representation of the modulated signal. The input must be a discrete-time binary-valued signal. If the input bit is 0 or 1, respectively, then the modulated symbol is exp(jθ) or -exp(jθ) respectively, where θ is the Phase offset parameter.

3- AWGN Channel

The AWGN Channel block adds white Gaussian noise to a real or complex input signal. When the input signal is real, this block adds real Gaussian noise and produces a real output signal. When the input signal is complex, this block adds complex Gaussian noise and produces a complex output signal. This block inherits its sample time from the input signal. This block uses the DSP Blockset's Random Source block to generate the noise.

4- BPSK Demodulator Baseband

The BPSK Demodulator Baseband block demodulates a signal that was modulated using the binary phase shift keying method. The input is a baseband representation of the modulated signal. The input can be either a scalar or a frame-based column vector. The input must be a discrete-time complex signal. The block maps the point’s exp (jθ) and -exp (jθ) to 0 and 1, respectively, where θ is the Phase offset parameter.

5- Complex to Real-Imag

The Complex to Real-Imag block accepts a complex-valued signal of any data type, including fixed-point data types, except int64 and uint64. It outputs the real and/or imaginary part of the input signal, depending on the setting of the Output parameter. The real outputs are of the same data type as the complex input. The input can be an array (vector or matrix) of complex signals, in which case the output signals are arrays of the same dimensions. The real array contains the real parts of the corresponding complex input elements. The imaginary output similarly contains the imaginary parts of the input elements

6-Scope, Floating Scope

The Scope block displays its input with respect to simulation time. The Scope block can have multiple axes (one per port); all axes have a common time range with independent y-axes. The Scope allows you to adjust the amount of time and the range of input values displayed. You can move and resize the Scope window and you can modify the Scope's parameter values during the simulation.

7-Error Rate Calculation

The Error Rate Calculation block compares input data from a transmitter with input data from a receiver. It calculates the error rate as a running statistic, by dividing the total number of unequal pairs of data elements by the total number of input data elements from one source.

8-Display

The Display block shows the value of its inputs, which are here the transmitted and received signal. This display is called Monte Carlo Simulation and it shows the number of bits sent, the number of errors and the Bit Error Rate.

2.1.2 The theory of the system's block

As we can see in the system's Block Diagram consists of 4 basic elements and sub-elements for displaying the results and calculating Bit Error Rate. We built this Block using the Matlab.

The main goal here is to transmit a random sequence of bits. The bit emitted between *kT* and (*k*+1)*T* where *T* designates the duration of the bit, is termed $b_k \in \{0,1\}$. In the sequence $\{b_k\}$ a 0 is represented by a voltage 0 V, while a 1 is represented by a voltage 1 V. The distribution of the this sequence is such that Pr{0} = Pr{1} = ½.

This sequence is sent to a Binary Phase Shift Keying (BPSK) Modulator operating as the following:-

-If $b_k = 1$, the modulated signal is $S(t) = (+)\cos(2\pi f_o t)$

-If $b_k = 0$, the modulated signal is $S(t) = \cos(2\pi f_o t + \pi) = (-)\cos(2\pi f_o t)$

Theoretically, the term $\cos(2\pi f_o t)$ is not considered since it complicates the simulation process and hence the BPSK Modulator simply generates , between *kT* and (*k*+1)*T*, a binary symbol X_k such that X_k = -1 or 1. The modulated sequence $\{X_k\}$ is transmitted through a classical Additive White Gaussian Noise channel. This channel adds a Gaussian variable to the modulated signal. Therefore the received sample after transmission through the channel is given by: -

$$Y_k = X_k + N_k$$

Where X_k = -1 or +1 V, and N_k is a Gaussian variable with Zero mean and variance σ^2 whose value determines the quality of the channel under test. A high noise variance corresponds to a noisy channel i.e. poor quality channel, while a small noise variance corresponds to good quality channel.

The role of the BPSK demodulator is firstly to produce a the sample Y_k from the analog signal coming from the physical channel and processed through the following processes

1) Multiplication by the carrier, 2) Matched filtering, 3) And sampling to obtain the transmitted signal Y_k.

At the receiver, the decision circuit compares the received sample Y_k with the a threshold equal to 0, as the following: -

-If $Y_k > 0$, then the decoder decides that the transmitted bit b_k was 1.
-If $Y_k < 0$, then the decoder decides that the transmitted bit b_k was 0.

2.1.3 The simulation of the system's block

The first element in this block is the Bernoulli Binary Sequence Generator. This generator will feed the modulator with a random sequence of bits, which can either assume 0 or 1. The distribution of the sequence here is such that the probability of 0 EQUALS the probability of 1 and it is equal to ½ (Half). After that this sequence is fed to the second block, which is a Binary Phase Shift Keying Modulator, which modulates the baseband signal.

The third element in our system is a classical Additive White Gaussian Noise channel. This type of channel almost simulates the transmission conditions for a signal, which can be affected by noise. The main feature of using this channel is that it adds a Gaussian variable to the modulated signal. From this block we can vary the variance of the channel and therefore we can see the effect of changing the variance and its affects on the noise.

After the signal has been subjected to the noise, it is received through a Binary Phase Shift Keying Demodulator. As we can see on the Block Diagram that the output of each element is fed to the Scope which displays the different shapes of the signal for each block, to obtain our calculations and readings we used the Marco Polo meter which calculates the number of bits, errors and the error/bit rate.

In order to obtain the different readings we changed the variance from 0.2 to 10, where the scope is used to show the different waveforms after being transmitted through the channel. Figures 2, 3 and 4 show the different waveforms for the variances 0.2, 5 and 10. The results and calculations are shown in the next couple of pages.

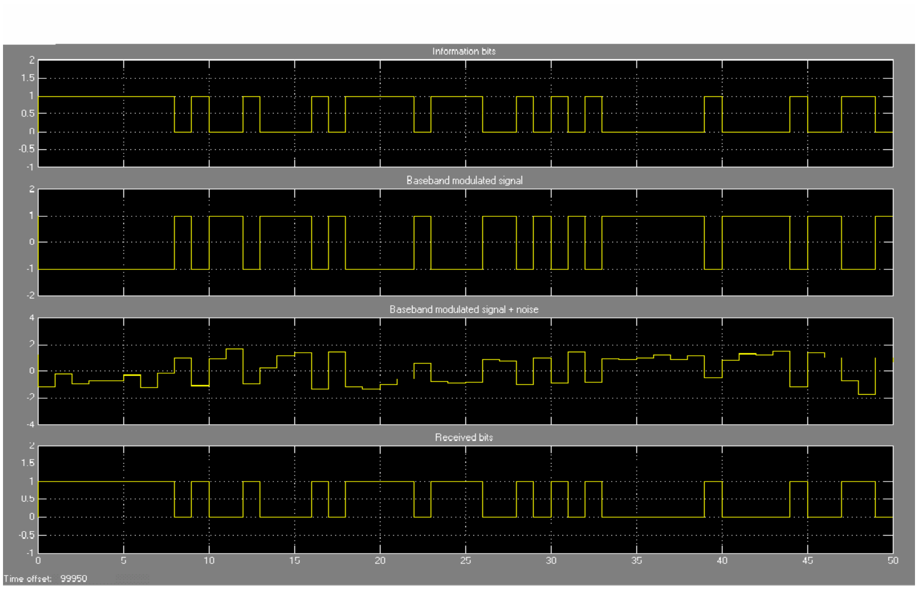

Figure 2 – The scope display at 0.2 channel variance

As we can see from the figure in the previous page, the transmitted signal (the top waveform) is almost unaffected by noise (the bottom waveform) and can be easily recovered at the receiver end. Figure 3 and 4 below illustrates the effect of noise as we increase the variance to 5 and 10 respectively to each figure. It can be seen that the signal will be distorted greatly as we increase the variance.

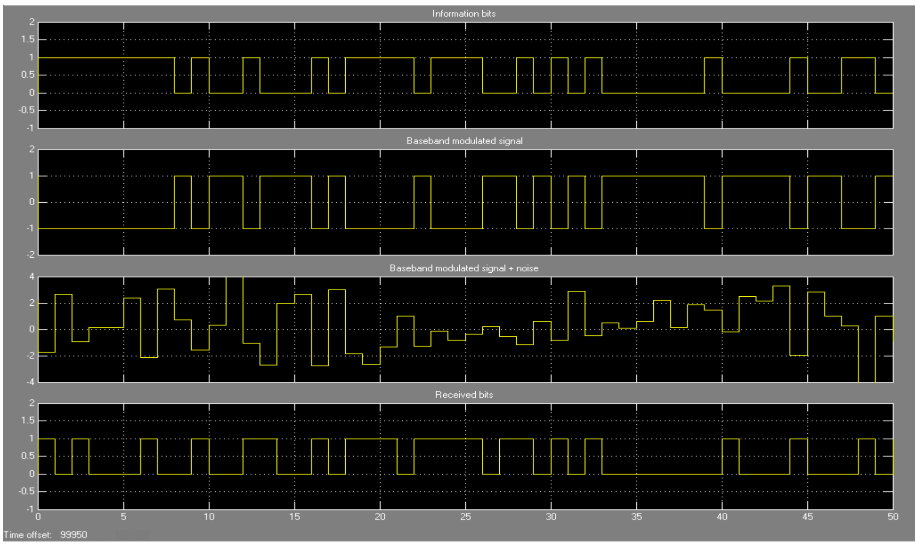

Figure 3 – The scope display at 5-channel variance

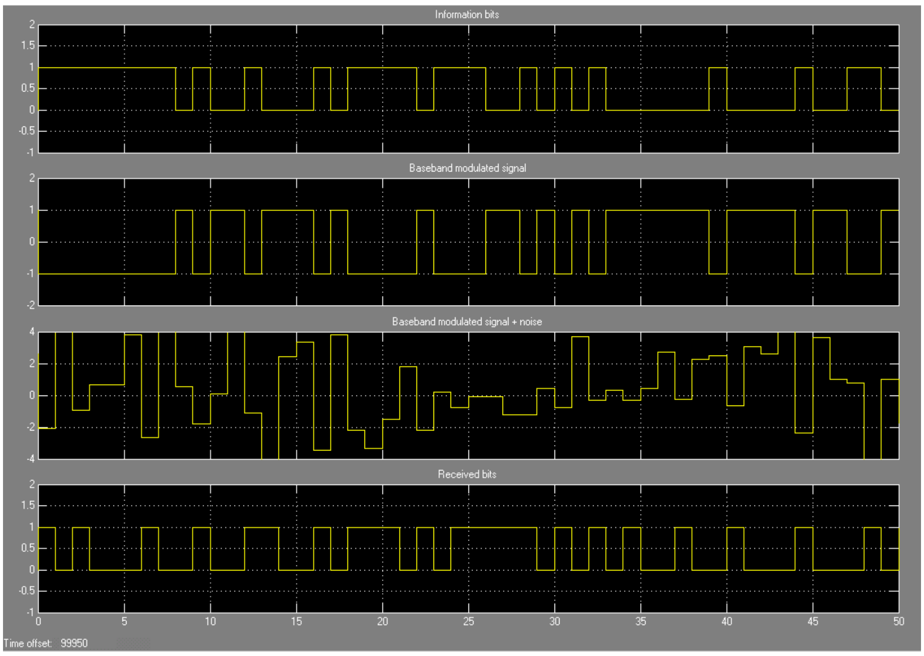

Figure 4 – The scope display at 10-channel variance

It can be seen clearly from the table shown below, when we increase the variance the signal will be more affected by the Gaussian Noise. The scope is here used to give us a visual output of each block.

The graph shown below is the Bit Error Rate Versus 10Log(1/Variance), this graph shows the effect of increasing the variance which corresponds to an increase in the noise level. The BER here is plotted against 10 Log (1/Variance) because this ratio is directly proportional to the Signal to Noise Ratio.

Table 1 – Results for different variance values

Variance	*Number of Bits*	*Number of Errors*	*Number of Errors/Bit*
0.2	100000	75	0.00075
0.4	100000	1297	0.01297
0.6	100000	3488	0.03488
0.8	100000	5736	0.05736
1	100000	7850	0.0785
2	100000	15910	0.1591
3	100000	20630	0.2063
4	100000	23850	0.2385
5	100000	26330	0.2633
6	100000	28130	0.2813
7	100000	29610	0.2961
8	100000	30770	0.3077
9	100000	31810	0.3181
10	100000	32660	0.3266

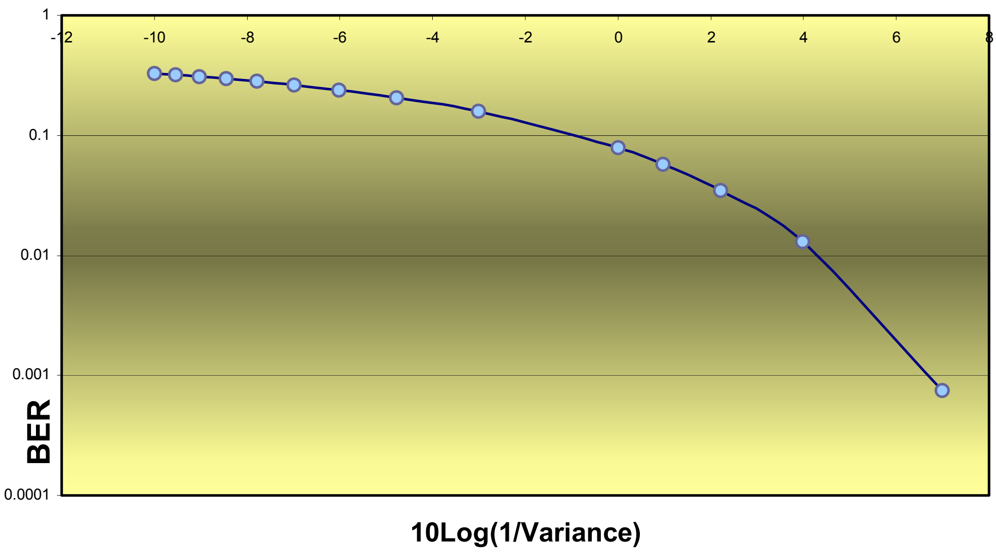

Figure 5 – BER Vs 10Log(1/Variance)

Chapter

3

3.0 Part Two.

3.1 Introduction:

The second part of the project is mainly about reducing the BER obtained in the first part.

The reduction of noise that was introduced by the channel would be explained in this section. A suitable technique to reduce the BER is utilizing a process of sending more bits (redundant) to be added to the information bits in order to protect the information bits during the transmission, this technique is called error-correcting code.

Before sending information bits through the channel, these bits are encoded (the design of the encoder will be explained in more details further up in this chapter). Some redundant bits are added to the stream according to a specified coding rule so that information bits will be protected during the transmission.

At the receiver side, a decoder is inserted at the demodulator output; this decoder will use the redundant bits to correct some of the errors that occurred during transmission.

3.2 Bits transmission:

3.2.1 1/3 Repetition Code:

A simple repetition code was used, which is an example of error-correcting code. Repetition codes which is a straightforward idea is to repeat every bit of the message some prearranged number of times, a 1/3 repetetion code was used here.

An explanation of the 1/3 repetetion code is as following with the aid of the tables.

-To transmit a 0, we send three 0s.
-To transmit a 1, we send three 1s.

Table 2 - The repetition code `R3'.

Source sequence **s**	Transmitted sequence **t**
0	000
1	111

If the source message s = (0 0 1 0 1 1 0) was transmitted over a binary symmetric channel with noise level f=0.1 using this repetition code. the channel adds a sparse noise vector $\mathbf{n}$ to the transmitted vector (in modulo 2 arithmetic, *i.e.*, the binary algebra in which 1+1=0).

s	0	0	1	0	1	1	0
t	000	000	111	000	111	111	000
n	000	001	000	000	101	000	000
r	000	001	111	000	010	111	000

Figure 6 - form of bits sent in the system.

3.3 Decoding the received vector:

The optimal algorithm looks at the received bits three at a time and takes a majority vote. The majority vote decoder is shown in figure 7. If all bits are 0, we decode the triplet as a 0.

If, as in the second triplet, there are two 0s and one 1, we decode the triplet as a 0-- which in this case corrects the error. The number of changes in positions or values is called the hamming distance.

Received sequence r	Decoded sequence ŝ
000	0
001	0
010	0
100	0
101	1
110	1
011	1
111	1

Figure 7 - Decoding algorithm for R3.

3.3.1 Hamming Distance:

Definition: The *Hamming distance d H* (***x*** *,* ***y***) between words ***x*** and ***y*** is the number of positions in which they differ.

- *d H* (1111011, 1111111) = 1
- *d H* (1234567, 1111111) = 6

3.4 Design and Implementation of the system:

3.4.1 Systems block diagram:

Below we can see the system of the second part, which was designed using **SimuLink**.

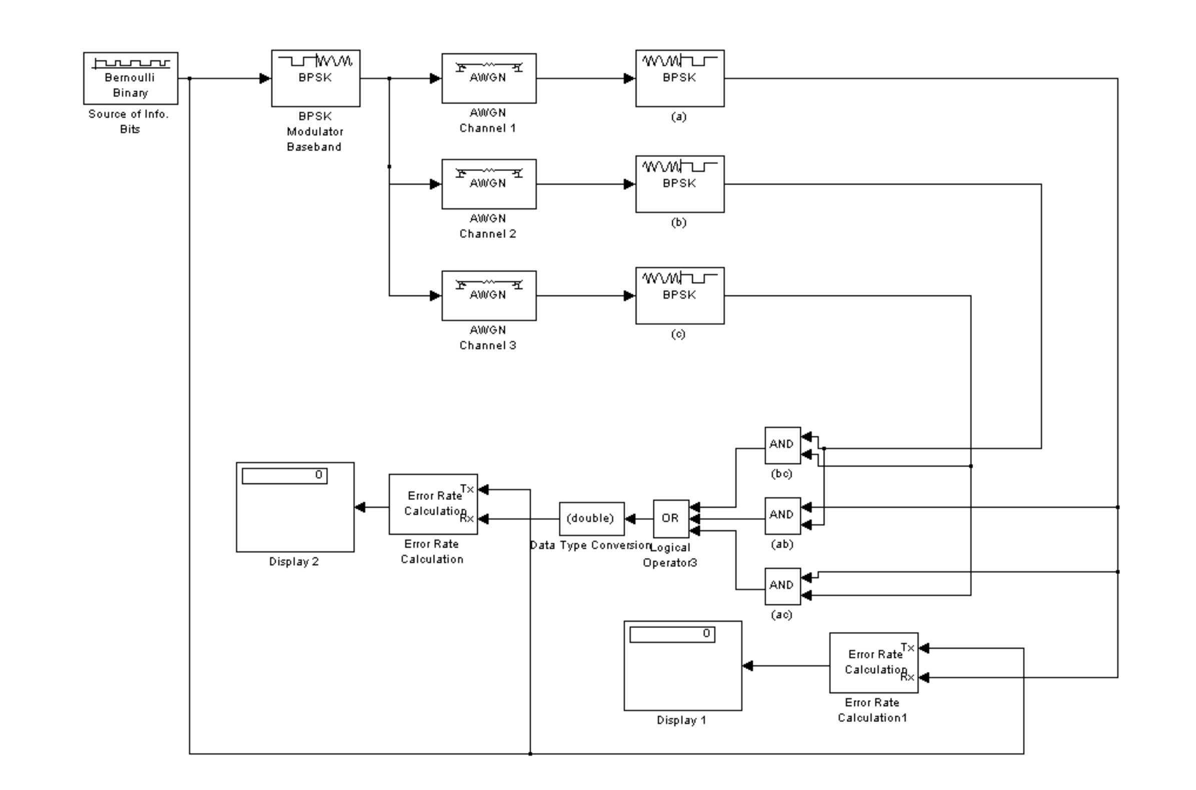

Figure 8 – Part 2 System's block diagram.

3.4.2 Systems operation:

A rate of 1/3 repetition encoder is inserted inside the transmitter. The difference from the first system can be noticed from here, were 3 bits were transmitted instead of one bit in order to reduce the BER.

The information bits are transmitted toward the Binary Phase Shift Keying Modulators that make the bits ready for transmission.

The bits move through the 3 Additive White Guassian Noise Channels (AWGN) to the Binary Phase Shift Keying (BPSK) were the bits are passed through this demodulator and returns to its original form when they were transmitted.

At the receiver side, the designed decoder is placed. The decoder is designed using three 2-input AND gates and one 3-input OR gate. The first BER (Bit Error Rate) meter is connected at the decoder input, while the second BER meter is placed at the decoder output.

The main job of the decoder is to detect the BER using the hamming distance and locate the error; the signal is then corrected to its original form using the decoder. Here it is noticed that the BER has decreased in a large ratio than the first system.

Results were obtained by changing the variance from 0.2 to 10 in order to have a range of readings and use the reading to compare with results obtained in the previous system before correcting the code and using the repetition technique.

The number of bits, number of errors and the bit/error rate (BER) were measured using two-error rate calculating blocks and two displays. The first displays the error before it gets in the decoder and been corrected and the second is located in the output of the decoder to measure the number of errors that has been obtained after using the decoder.

3.4.3 Design of the decoder:

In order to correct or reduce the BER a decoder was needed. The design of the decoder was simply using 3 AND gates and a single OR gate. In order to reach this design a truth table was made as you can see below.

Table 3: the truth table of the decoder

a	b	c	O/P (Decoder)
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

The next stage was is to simplify the results obtained from the truth table using the K-MAP technique as we can see below.

K-MAP:

c \ ab	00	01	11	10
0			X	
1		X	X	X

Figure 9 – K-map of the decoder

As shown above, the final result for the decoder design was obtained and therefore it is ready for implementation.

Dec. = bc + ab + ac

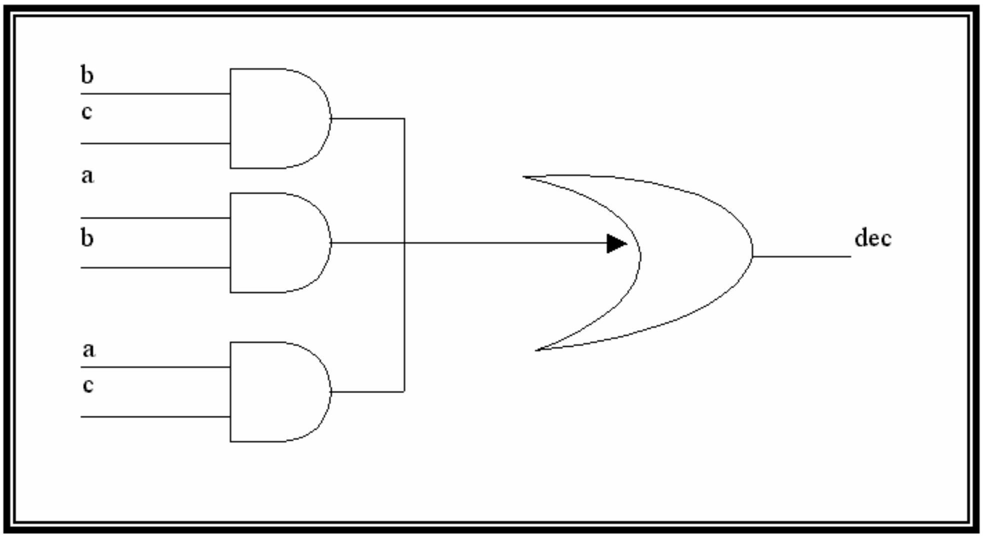

Figure 10 - Decoder used in the second part of the system.

3.5 Results of Part 2:

Results obtained in this part of the project were the most important part in this project, because from the results obtained we noticed how the repetition code helped us to have more accurate transmission.

The following table shows the result of each variance obtained in this part of the project, and compares the BER before and after decoding as shown one the next page in Table 4.

Table 4 - Final results of the project.

Variance	*10 Log (1/Variance)*	*BER (After Decoding)*	*BER (Before Decoding)*
0.2	3.9794	0.00049	0.01297
0.4	2.2185	0.00364	0.03488
0.6	0.9691	0.00938	0.05736
0.8	0	0.01695	0.0785
1	-3.0102	0.06719	0.1591
2	-4.7712	0.1104	0.2063
3	-6.0205	0.1438	0.2385
4	-6.9897	0.1724	0.2633
5	-7.7815	0.1936	0.2813
6	-8.4509	0.2111	0.2961
7	-9.0308	0.2258	0.3077
8	-9.5424	0.2394	0.3181
9	-10	0.2502	0.3266

As mentioned now from the above table that the BER decrease by more than 70% in low variances and by almost 50% in the higher variances. This shows us how usefull was the decoder and how useful is the repetition code in order to decrease the BER.

Although after introducing the decoder in the system and using three bits instead of one, the BER cannot be removed from the system as noticed through this project.

The main problem of not reducing the BER to 0% us that in some cases the decoder can not always judge the number of errors in the signal transmitted, or in other words that the decoder corrects the signal to certain measures.

Sometimes the signal is affected by a great deal by the nose that it is altered from its original form in more than two bits, that makes the decoder corrects it to an opposite value.

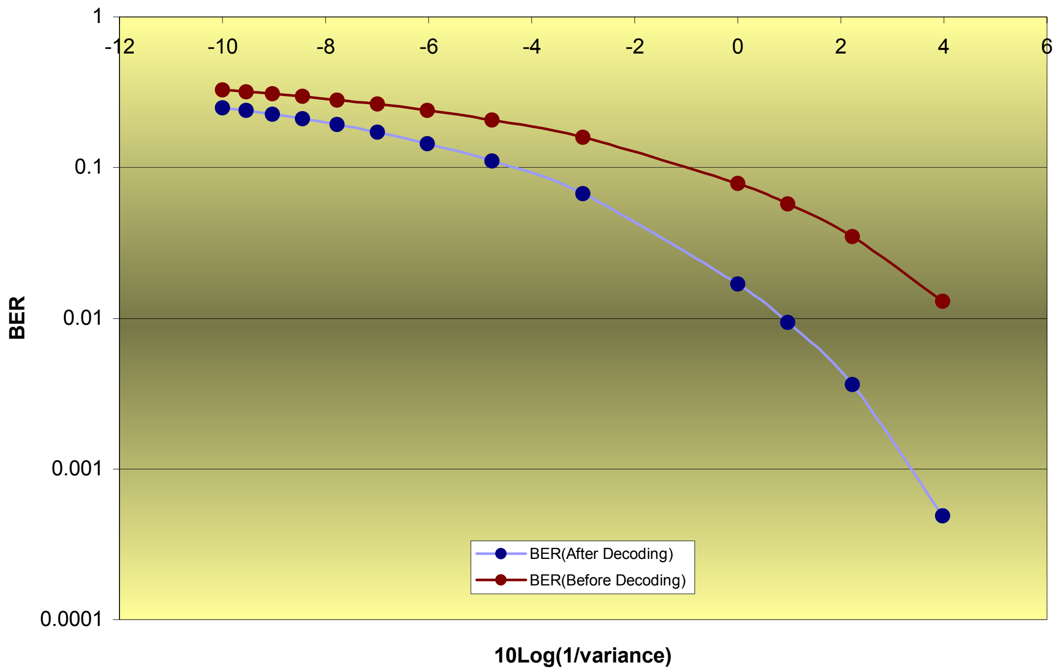

Figure 11 – BER (Before & After Decoding) Vs 10 Log(1/Variance)

The above graph shows the BER before and after decoding versus 10log(1/variance). It is noticed that the difference between the first system and the second system. This graph gives a good visual idea of how the repetition code scheme is useful in decreasing the BER.

Chapter 4

4. TOOLS

The only software used through out the work in the project was MATLAB version 6. It mainly a math program that makes the complicated calculations and numbers easy to play with.

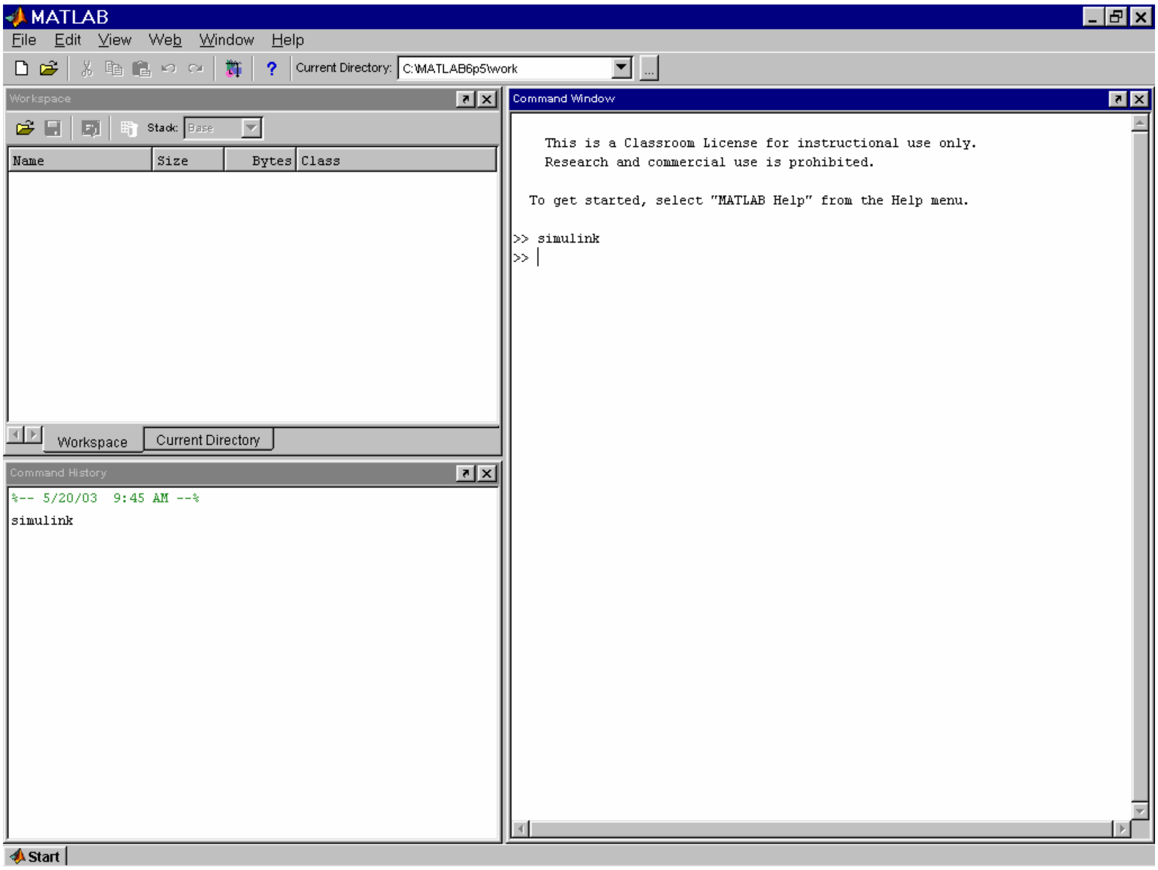

Figure 12 - MATLAB interface.

Simulink is a program that runs as a companion to **MATLAB**, these programs are developed and marketed by the **MathWorks, Inc. Simulink** and **MATLAB** form a package that serves as a vehicle for modeling dynamic systems.

Simulink provides a graphical user interface (GUI) that is used in building block diagrams, performing simulations, as well as analyzing results. In Simulink, models are hierarchical so a system can be viewed at high level.

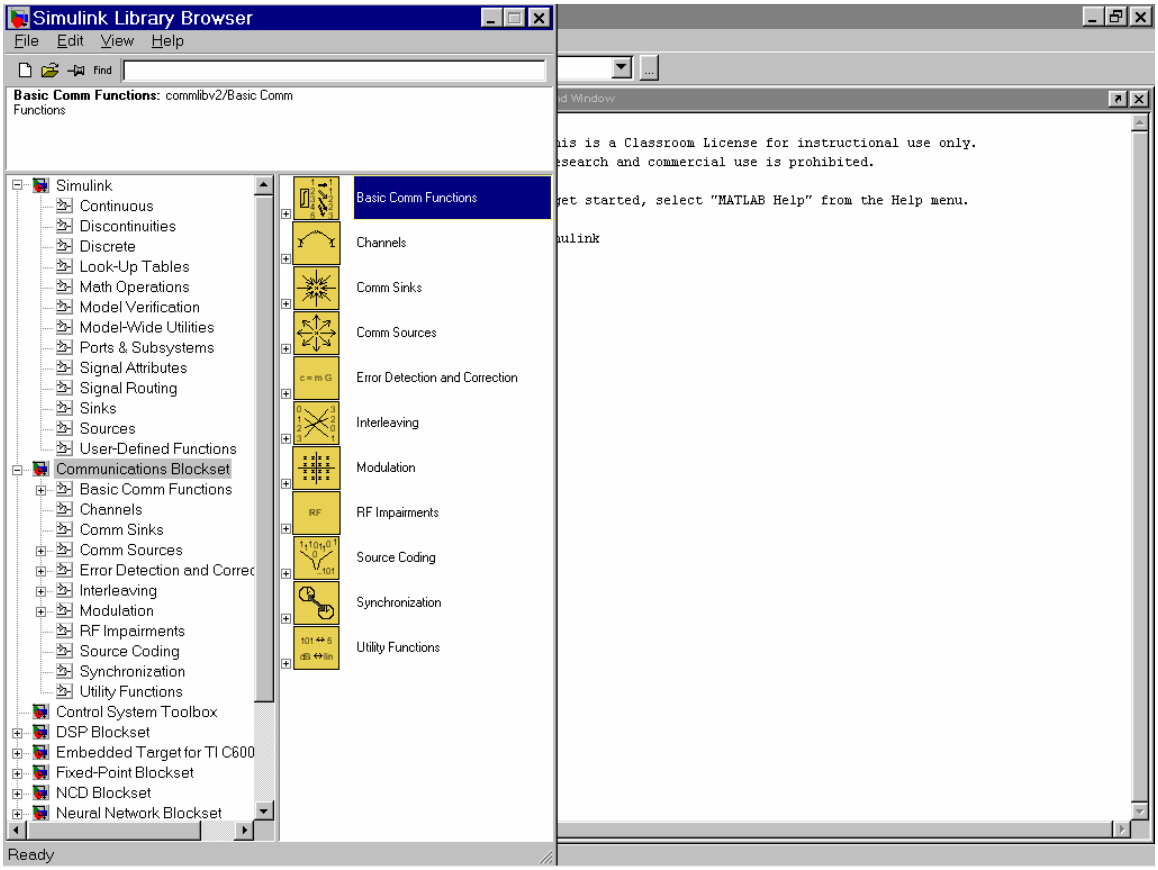

Figure 13 - **Simulink** Library Browser.

As seen above, the Simulink Library Browser which was used to obtain all the blocks in the system. This library has a whole range of communications devices that can be used in any design.

After finishing the design the main task was to change the variance of the AWGN channel. Below is shown the process of changing the variance.

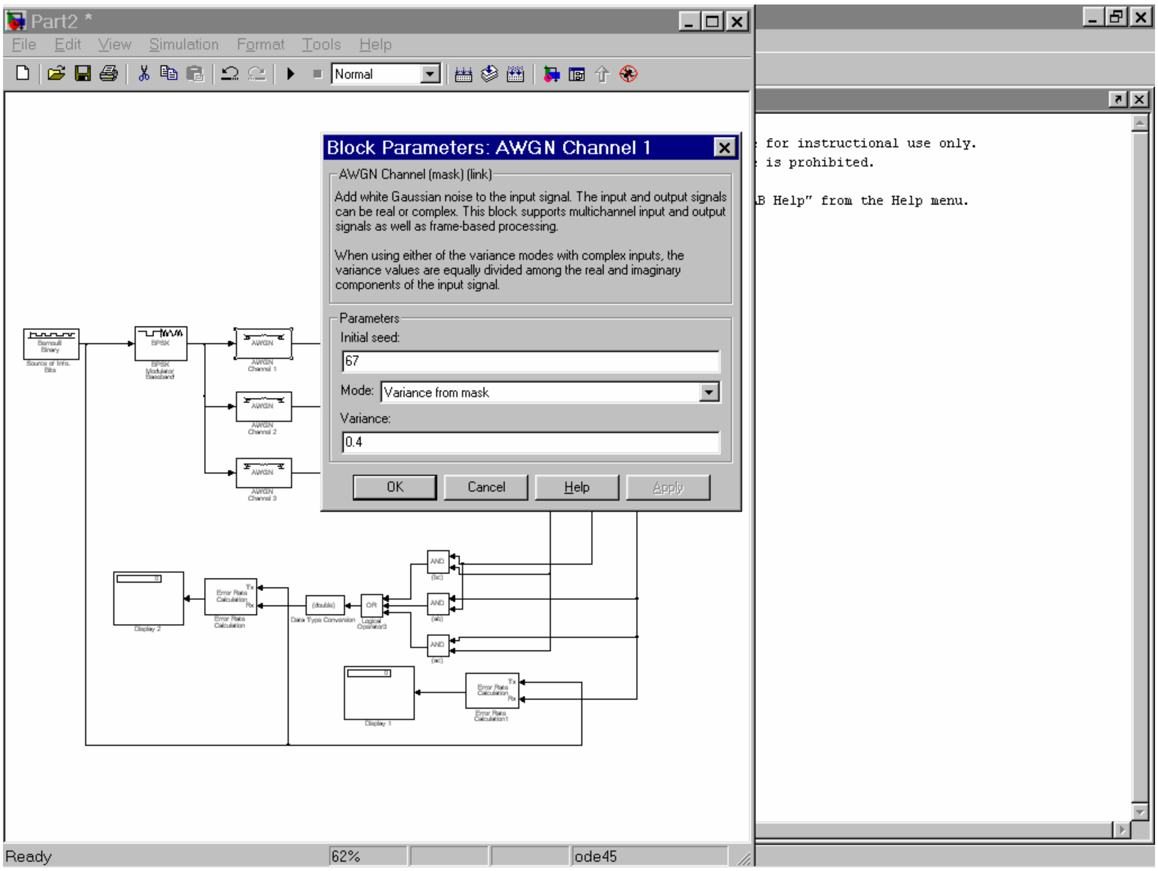

Figure 14 - Work in progress through the simulations.

Chapter

5

5.THE CONCLUSION

In conclusion, the work in the project went on perfectly with out facing large problems that delayed the work. The most difficult part of the project was the design of the decoder, which our supervisor helped us in it.

A great benefit obtained from the work in this project, specially learning how to use the Simulink and the designing process. The project gave us good information on how to correct the signal transmitted using the repetition code.

Through out our research on further information for the project, we noticed other techniques rather than the repetition code which can be tested and implemented in the future in order to obtain better results. The techniques are as the following:

-The SPC (*single- parity- check code*) code.
-The (7, 4) Hamming Code.
-Error- Correction Capability.
-Shannon Capacity [1948].

Section

6

6.THE APPENDIX

The scope display of channel variances from 0.4 to 9 excluding the value of 5 channel variance which is already included in the report

Chapter

7

7.REFERNCES

1. Analytical techniques in telecommunications.
By: Fraidoon Mazda.

2.Audiovisual telecommunications.
By: N.D Kenyon and C. Nightingale.

3.MATLAB
Help Menu.

4.Internet.