



AMERICAN UNIVERSITY OF BEIRUT

OPTIMAL VLSI ARCHITECTURE FOR A 2X2 MIMO  
DETECTOR

by  
LAMA AHMAD SHAER

A thesis  
submitted in partial fulfillment of the requirements  
for the degree of Master of Engineering  
to the Department of Electrical and Computer Engineering  
of the Faculty of Engineering and Architecture  
at the American University of Beirut

Beirut, Lebanon  
May 2014

AMERICAN UNIVERSITY OF BEIRUT

OPTIMAL VLSI ARCHITECTURE FOR A 2X2 MIMO  
DETECTOR

by  
LAMA AHMAD SHAER

Approved by:

\_\_\_\_\_  
Dr. Mohammad Mansour, Associate Professor  
Electrical and Computer Engineering

  
Advisor

\_\_\_\_\_  
Dr. Ali Chehab, Associate Professor  
Electrical and Computer Engineering

  
Member of Committee

\_\_\_\_\_  
Dr. Rouwaida Kanj, Assistant Professor  
Electrical and Computer Engineering

  
Member of Committee

Date of thesis/dissertation defense: April 30, 2014

AMERICAN UNIVERSITY OF BEIRUT

THESIS, DISSERTATION, PROJECT RELEASE FORM

Student Name: \_\_\_\_\_ SHAER \_\_\_\_\_ LAMA \_\_\_\_\_ AHMAD \_\_\_\_\_  
Last First Middle

Master's Thesis       Master's Project       Doctoral Dissertation

I authorize the American University of Beirut to: (a) reproduce hard or electronic copies of my thesis, dissertation, or project; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

I authorize the American University of Beirut, **three years after the date of submitting my thesis, dissertation, or project**, to: (a) reproduce hard or electronic copies of it; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

Lama Shaa

15/05/2014

Signature

Date

---

## ACKNOWLEDGMENTS

I want to thank my advisor Professor Mohammad Mansour for his guidance and help. I want to thank committee member Professor Ali Chehab for his great help and support. Thank you for always pushing me forward and for believing in me. I want to thank committee member Professor Rouwaida Kanj for her valuable feedback regarding my work.

I would also like to thank my parents, my role models, for believing in me and for constantly supporting me. Thank you for listening to me and thank you for giving me the right advice.

I would like to thank my friends for bearing long conversations about my research. Thank you for being my source of comfort. Thank you for making the journey worthwhile. I love you so much.

# AN ABSTRACT OF THE THESIS OF

Lama Ahmad Shaer for Master of Engineering  
Major: Electrical and Computer Engineering

Title: Optimal VLSI Architecture for a 2x2 MIMO Detector

In communications systems, increasing the data rate of transmission has become a vastly growing field of study. Multiple-Input Multiple-Output systems emerged as a promising approach in such a field. However, increasing the number of transmitted vectors makes the detection issue even harder.

At the transmission side, the data is changed into a binary sequence of bits to be later modulated. Then, the signal is sent over the channel where some noise will be added to the signal distorting it.

At the receiver side, the receiver receives these noisy symbols displaced from their initial position in the constellation. Therefore, recovering the original symbol coordinates on the constellation is not an easy task especially when there are multiple transmission sources and multiple receivers. In this thesis, we consider 2x2 MIMO system and we propose an efficient detection method whose architecture was implemented in VHDL.

# CONTENTS

|   |    |
|---|----|
| ACKNOWLEDGEMENTS .....                                  | v  |
| ABSTRACT .....  | vi |
| LIST OF ILLUSTRATIONS.....                              | ix |
| Chapter   |    |
| I.INTRODUCTION.....                                     | 1  |
| A. Motivation.....                                      | 1  |
| B. Basic MIMO Channels.....                             | 2  |
| C. Detection Problem .....                              | 6  |
| D. Thesis Contribution and Organization.....            | 6  |
| II.EXISTING DETECTION SCHEMES.....                      | 9  |
| A. Existing Detection Algorithms.....                   | 9  |
| B. Existing Architectures for Detection Algorithms..... | 12 |
| III. THE ALGORITHM AND THE ARCHITECTURE.....            | 15 |
| A. The Algorithm.....                                   | 15 |
| 1. Matlab Simulation Results.....                       | 19 |
| IV. IMPLEMENTATION OF THE ALGORITHM.....                | 21 |
| A. Basic Components.....                                | 21 |
| 1. Basic Adders and Multipliers.....                    | 21 |
| 2. Two's Complement and Shifters.....                   | 24 |
| B. Stages formulation and Implementation.....           | 24 |
| 1. Stage1 formulation and Implementation.....           | 24 |
| 2. Stage2 formulation and Implementation.....           | 29 |

|   |    |
|---|----|
| 3. Stage3 formulation and Implementation..... | 32 |
| 4. Overall Design.....                        | 38 |

|                                    |    |
|------------------------------------|----|
| V. CONCLUSION AND FUTURE WORK..... | 42 |
|------------------------------------|----|

|                     |    |
|---------------------|----|
| VI. REFERENCES..... | 43 |
|---------------------|----|



# ILLUSTRATIONS

| FIGURE  | PAGE |
|---|------|
| 1. Elements of a communication system[29].....            | 2    |
| 2. From Digital to Analog[27].....                        | 4    |
| 3. Constellation of QPSK[27].....                         | 5    |
| 4. Constellation of 8-PSK[27].....                        | 5    |
| 5. Constellation of QAM[27].....                          | 6    |
| 6. Architecture for Sphere Decoder[12].....               | 13   |
| 7. Flow Graph for the Algorithm.....                      | 18   |
| 8. Results of Detection Rate.....                         | 20   |
| 9. One bit full adder[28].....                            | 22   |
| 10. 8-bit full adder[29].....                             | 22   |
| 11. The 8-bit Baugh Wooley Multiplier Algorithm[18].....  | 23   |
| 12. Architecture for Computing $x2_{real}$ .....          | 26   |
| 13. Architecture for Computing $x2_{imaginary}$ .....     | 27   |
| 14. Output of Stage1.....                                 | 28   |
| 15. Architecture for slice operation.....                 | 30   |
| 16. Output of first part of stage2.....                   | 31   |
| 17. Final Answer for stage2.....                          | 32   |
| 18. Architecture for computing A.....                     | 34   |
| 19. Architecture for computing B.....                     | 35   |
| 20. Final answer for stage 3 real part.....               | 36   |
| 21. Architecture for computing the imaginary part.....    | 37   |
| 22. Final Answer for the imaginary part.....              | 38   |
| 23. Architecture for implementing the overall design..... | 40   |
| 24. Final Design.....                                     | 41   |

# CHAPTER I

## INTRODUCTION

### **A. Motivation**

In communication systems, increasing the rate of data transmission has become a vastly growing field of study. Methods of signal detection arose as an essential component of the wireless services. Accordingly, multiple-input multiple-output systems emerged as a promising approach in such an expanding field [2]. The growth of the system capacity with the number of parallel sub-channels increased the focus on the Multiple-Input Multiple-Output systems. However, increasing the number of transmitted vectors increases the accompanying detection predicament [1].

At the transmission side, the data produced is converted into a binary sequence of bits using a source encoder. The output of the source encoder is then fed to the channel encoder, where some redundancy is required so as to avoid noise effects. This output is later fed to the digital modulator, which maps the binary stream into signal waveforms [3]. There are three basic types of digital modulation: Amplitude shift Keying, Frequency Shift Keying, and Phase Shift Keying. Our focus will be mainly on Quadratic Amplitude Modulation which is a variation of the Phase Shift Keying [4]. Afterwards, the signal is sent over the channel and some noise will be added to the signal that will distort it and displace the initial position of a symbol in the constellation. The receiver receives a constellation with symbols displaced due to the noise effect.

Therefore, recovering the original signal is no easy task especially when there are multiple transmission sources and multiple receivers. Therefore, it is necessary to design an efficient detection scheme that will recover the original signal or trace back the displaced symbol in a constellation. The purpose of this thesis is to present a novel architecture for an efficient detection algorithm that will reduce the cost of computation. In addition, the aspects of power and speed will be targeted in the architecture.

## **B. Basic MIMO Channels**

The basic elements of communication system are shown in the figure below. These include Information source and input transducer, a source encoder, a channel encoder, a digital modulator, a channel, a digital demodulator, a channel decoder, a source decoder, and an output transducer.

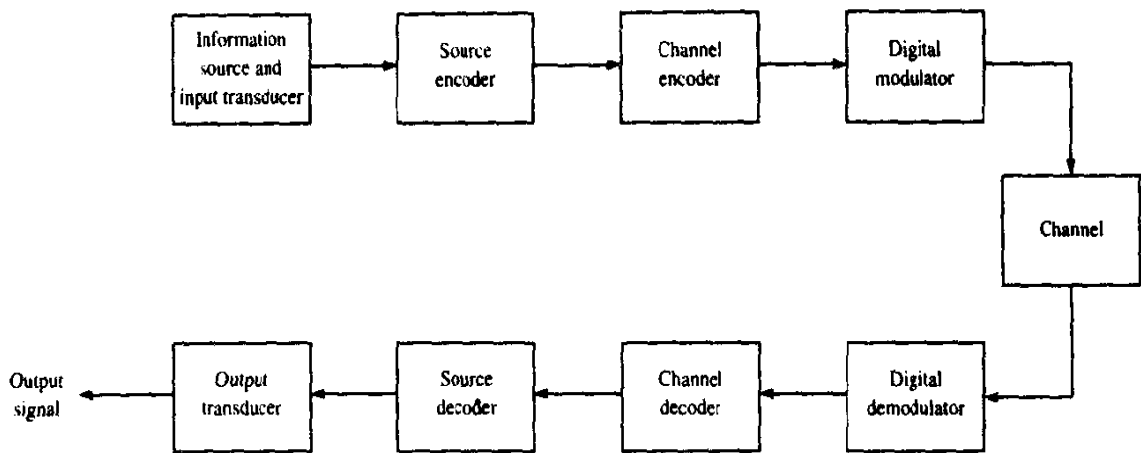


Figure 1: Elements of a communication system

At the source, the messages produced are converted into a sequence of binary digits using the source encoder that are fed to the channel encoder. The channel encoder adds some redundancy to the binary sequence which allows the receiver to overcome noise added to the channel. The binary sequence is passed to the digital modulator. It maps the binary sequence to waveforms that are to be sent over the channel. The channel is the physical medium whereby the signal is sent. The digital demodulator processes the waveform and reduces it to a binary sequence of numbers. It is passed to channel decoder that attempts to reconstruct the original sequence. The source decoder accepts the output sequence of the channel decoder and from knowledge of source encoding method reconstructs the signal.

The process of converting information so that it can be sent over a medium is called modulation. It takes your voice and maps it into some aspect of a sine wave that is sent

over the channel leaving the actual voice behind. The sine wave on the other side is remapped to a near copy of the voice. The sine wave is called the carrier. All of these techniques vary an aspect of a sinusoid (amplitude, frequency, phase). There are three basic types of modulation techniques: Amplitude Shift Keying, Frequency Shift Keying, and Phase shift keying.

In Amplitude Shift Keying the amplitude of the carrier is changed in response to the information. In FSK we change the frequency in response to information, one particular frequency for a 1 and another frequency for a 0.

In phase shift keying we change the phase of the sinusoid. To transmit a 0 we shift the sinusoid by 180 degrees. ASK and PSK are combined to produce the Quadrature amplitude modulation.

The I and Q channels allow us to define the signal as a vector with coordinates.

A symbol is a representation of bits that the medium transmits to convey the information.

The bits that it stands for are not being transmitted, what is transmitted is the symbol or waveform.

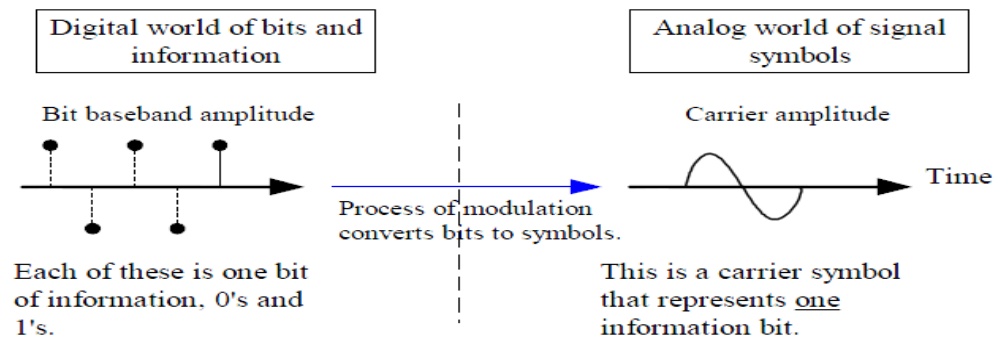


Figure 2: From Digital to Analog[27]

If we have two symbols S1 and S2, we utilize only one sinusoid as a basis function.

Quadrature Phase Shift Keying: An extension of the Binary Phase Shift Keying. It is used when the order of modulation is 4. The modulated signal is described as follows:

$$S_i(t) = A_c p_s(t) \cos(2f_c t + 2\pi i/M) \quad (1)$$

The corresponding constellation is shown below:

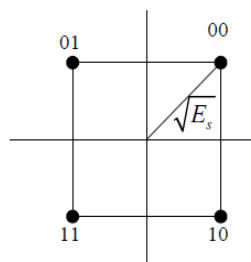
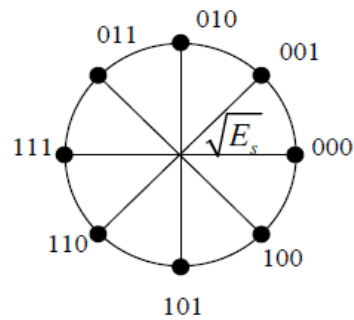


Figure 3: Constellation of the QPSK[27]

When the order of modulation is 8 for a three bit representation, the constellation is as shown below.



**Figure 4: Constellation of 8-PSK[27]**

Quadrature Amplitude Modulation (QAM) is a combination of ASK and PSK. PSK allows the changing of the amplitude and the phase. Hence, all the points lie on a circle so the I and Q are related to each other and is represented as the following equation:

$$S(t) = (\sqrt{2E_s/T}) * \cos\theta(t) * \cos(2\pi f_c t) - (\sqrt{2E_s/T}) * \sin\theta(t) * \sin(2\pi f_c t) \quad (2)$$

Therefore, we are allowed to modulate by varying the phase and amplitude and the constellation is shown below:

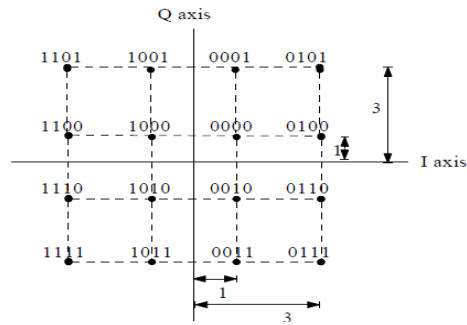


Figure 5: Constellation of QAM[27]

MIMO systems were first described by Foschini and Gans in [13] and it refers to having multiple inputs and multiple outputs. A MIMO system includes multiple transmitters and multiple receivers.

### C. MIMO Detection Problem

A MIMO system with  $n_T$  transmit antennas and  $n_R$  receive antennas is described using the following formula:

$$\mathbf{Y} = \mathbf{H}\mathbf{x} + \mathbf{n} \quad (3)$$

where  $\mathbf{Y}$  denotes the received symbol vector;  $\mathbf{Y} = [y_1, \dots, y_R]^T$  and  $\mathbf{H}$  is the  $n_R \times n_T$  channel matrix where the complex valued elements  $h_{ij}$ , which represent the complex fading gain from the  $j$ -th transmit antenna to the  $i$ -th receive antenna. As for  $\mathbf{x}$ , it represents the baseband signal vector transmitted by elements chosen from the



constellation of the QAM,  $x=[x_1 \ x_2 \ \dots \ x_T]^T$ . Regarding  $n$ , it is the complex white gaussian noise that contains the noise elements added,  $n=[n_1 \ n_2 \ \dots \ n_R]^T$ .

Since a 2x2 MIMO detector is considered in this thesis, then  $Y$  is a 2x1 matrix,  $H$  is a 2x2 matrix,  $x$  is a 2x1 matrix, and  $n$  is a 2x1 matrix.

In particular,  $Y$  is a 2x1 matrix that includes the coordinates of the received signals on the constellations.  $H$  is the 2x2 matrix that includes the fading factors. Furthermore,  $x$  is a 2x1 matrix that includes the coordinates of the sent signals at the transmitter side, and  $n$  is a 2x1 matrix that represents the noise that is added while sent on the channel.

The detection problem described in [13] is briefly summarized as follows. There exists a pair of coordinates of points belonging to the 64-QAM constellation that minimizes the Euclidean distance between the matrix that includes the received symbol and the product of the channel matrix and the matrix including the pair of the coordinates which represent the detected symbol coordinates.

#### **D. Thesis Contribution and Organization**

Since the detection problem is a very recent hot topic, many algorithms emerged to tackle this issue. However, each of these algorithms has a drawback in terms of complexity of computation or the power consumption. The thesis presents a novel technique to find the optimum solution for the detection problem with minimal number

of required computations. Since power is associated with the number of required computations, then reducing the computations will certainly result in a considerable power reduction. The thesis is organized as follows:

### **Chapter Two: Existing Detection Schemes and Architectures**

- Various algorithms that target the problem of detection are presented. Also, the architectures for such schemes are also presented

### **Chapter Three: The Algorithm and the corresponding Architecture**

- The proposed algorithm is presented with the explanation
- Each of the building blocks for the algorithm is presented and discussed
- The suggested architecture for implementing the algorithm and the approach for implementing it.

### **Chapter Four: Implementation of the Proposed Architecture**

- The implementation of the different blocks of the architecture. Each stage is presented individually alongside the overall design.

### **Chapter Five: Conclusion and Future Work**

- This chapter concludes the thesis and presents the possible future work that could be done to improve the algorithm in future works.

## CHAPTER II

### EXISTING DETECTION SCHEMES AND ARCHITECTURES

In this chapter, an overview of the existing detection algorithms is used along with the architectures –introduced in the literature. The first section introduces the detection algorithm and their significance and approaches used in them. In the second and third sections, the analytical and statistical techniques are discussed along with their advantages and disadvantages. The fourth section concludes the chapter by motivating the introduction of a novel architecture for MIMO detector.

#### A. Existing Detection Algorithms

Many detection schemes have been proposed. MIMO detection schemes can be classified into two main categories: Maximum Likelihood (ML) methods and sub-optimal methods.

##### ML Methods

In [16], Maximum Likelihood Rule implies that the optimal MIMO detector is the detector that minimizes the average probability error as given below:

$$P(e) \triangleq P(\hat{x} \neq x)$$

In [17], the Maximum Likelihood Rule refers to finding the  $\hat{x} = \underset{x \in C}{\operatorname{argmin}} \|Y - Hx\|^2$ .

$C = \chi^m$  and  $\chi$  is the Pulse Amplitude Modulation of signal set.[16].

One type of ML methods is the exhaustive search ML method. The ML detector is optimal in terms of the Bit Error Rate but it has a very high computation cost. This approach assumes that all possible  $x$  vectors belong to a finite set of points. In a 2x2 MIMO channel it assumes that the received  $x_1$  and  $x_2$  are certain points in the constellation. Therefore for every possible point in the 64-QAM constellation it roams through the 64 points in the QAM constellation and calculates the distance between the following both points as given below.

$$d(x_1, x_2) = \|Y - HX\|^2$$

The final ML solution corresponds to that of the minimum  $d(x_1, x_2)$  where  $Y$  is the 2x1 matrix that represents the received vector.  $H$  is the 2x2 matrix that include the complex coefficients for the channel.  $X$  is the 2x1 matrix that represents the sent signal from each transmitter. These signals are  $x_1$  and  $x_2$ .

The automatic sphere decoder is another ML detector. The sphere decoding algorithm mainly begins from the root node and skims through the remaining nodes of the tree so as to compute the respective weights of the connected branches and nodes. It skims through just enough points on the tree to allow it to locate a point that meets the ML solution. The ASD approach utilizes the tree method. It selects a root node. Then it expands a selected node and inserts children into the node list. Then it selects the node

with the smallest weight until the leaf node is selected. Then it reports the optimal solution and the search radius[11].

#### Suboptimal-ML Methods

Another type is the matched filter detector. The matched filter is mainly a linear filter which maximizes the SNR. The matched filter's functionality is similar to a correlation where the unknown signal undergoes a convolution with the filter that has an impulse response that is a mirror and time shifted copy of the original signal. This operation is mostly useful when the original signal is known[8].

There is also the Zero-Forcing (ZF) Detector. The ZF detector gives a performance that is sub-optimal, however it provides a reduction in terms of computational complexity.[9]It sets the interference amplitude to zero. It inverts the channel response and rounds to the closest symbol in the available. The channel matrix is now symmetric an inversion is employed. Then pseudo inversing is then employed using an inversion step [10].

As well there is the Minimum Mean Square Error (MMSE) detector. This detector minimizes the error due to noise and the interference combined. . It implements a code structure that is vertically layered where independent such that each block is associated with a transmitter .The transmitted layers are detected using successive interference cancellation. However, this implies that tedious calculations are required. Thus using the MMSE approach simplifies the task, since it decreases the mean square error

between the transmitted symbol and the output of the detector. In particular, the MMSE approach uses a sorted QR decomposition technique [6].

Another approach is the Lattice Reduction Aided Detection (LRAD), which reduces the complexity of the receiver structure for a MIMO orthogonal frequency division multiplexing. This algorithm presents a novel approach that results in a similar performance to the maximum likelihood detector. The operations at the receiver consist of scaling, shifting, and equalizing in the new basis. Subsequently, the receiver performs a slicing operation, whereby it returns to the original basis and undoes the scaling and shifting [7].

### **B.Existing Architecture for Detection Algorithms**

Several architectures have been proposed to implement decoder algorithms. One main architecture is that of the sphere decoder algorithm. A parallel implementation is proposed. The original node is first chosen. Other branches are then examined in parallel and the smallest one is computed. The architecture inspects a single node at each clock cycle. Figure 1 shows the suggested architecture.

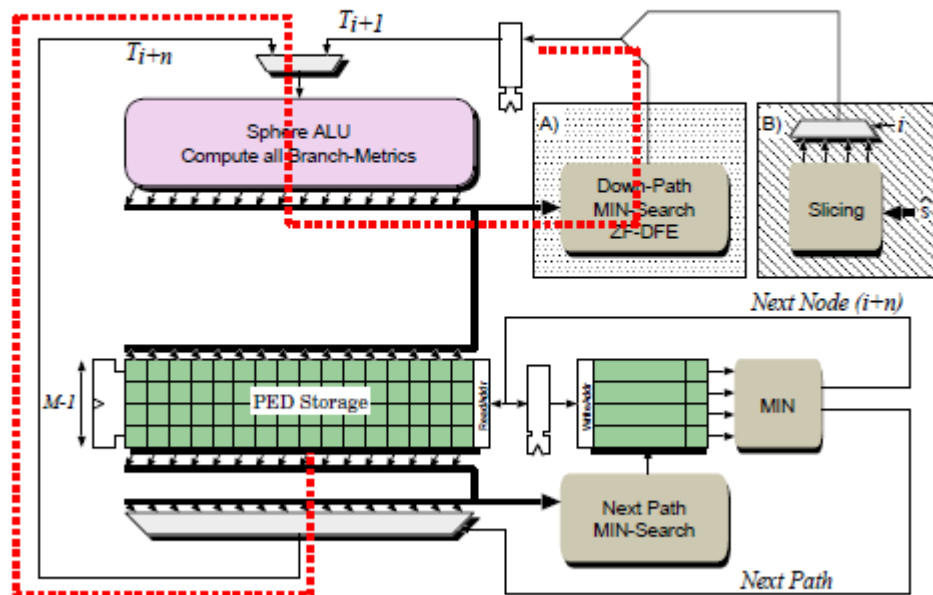


Figure 6: Architecture for the Sphere decoder[12]

The architecture includes a sphere ALU that computes partial Euclidean distances of all the paths coming from the original node concurrently. Moreover, the down path unit is used to compute the smallest of these distances. If a dead end is reached, the decoder has to explore the nodes of the current graph to find a path that meets the sphere constraint.[12]

Maximum Likelihood and Sub-Optimal methods are approaches to relocate the displaced symbol in the received constellation. However, they require a lot of

computation. The main detection problem is summarized as follows. There exists a certain combination of  $(x_1$  and  $x_2)$  that belongs to a 64- QAM constellation and minimizes  $\|Y - HX\|^2$ , where  $Y$  is a  $2 \times 1$  vector that has the received points on the 64- QAM.  $H$  is a  $2 \times 2$  matrix that represent the channel matrix coefficients.  $X$  is the  $2 \times 1$  vector that contains  $(x_1$  and  $x_2)$  that represent the coordinates of the points on the constellation.

The algorithm adopted in our problem is the Fitz algorithm. The Fitz algorithms aims at finding the best  $x_1$  and  $x_2$  that minimize  $\|Y - HX\|^2$ . This will be discussed in the next section.



## CHAPTER III

### THE ALGORITHM AND THE CORRESPONDING ARCHITECTURE

After studying the different approaches used in the literature for the detection issue, this chapter will present an overview of the proposed approach based on the Fitz Algorithm. First, a general overview of the algorithm is presented. Then, design requirements and possible choices of each of its building blocks are studied separately.

#### **A. The Algorithm**

The main aim behind the proposed algorithm is to reduce the complexity of computing the Maximum Likelihood Rule. If it were to be solved in a brute way the computational cost would be [13].

A 2x2 MIMO transmission system is described as  $Y = Hx+n$ .

Y is a 2x1 matrix that includes the coordinates of the received signal,  $Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$ .

H is a 2x2 matrix that contains the complex coefficients for the channel  $H =$

$$\begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}.$$

x is a 2x1 matrix that contains the complex coordinates of the transmitted signal  $x =$

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}.$$

Therefore  $Y = Hx+n$  can be represented as follows

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} \quad (4)$$

We start deriving the needed formula for the algorithm as follows:

$$\text{Let } H_1 = \begin{bmatrix} h_{11} \\ h_{12} \end{bmatrix}$$

$$\text{Let } H_2 = \begin{bmatrix} h_{21} \\ h_{22} \end{bmatrix}$$

$$Y = H_1x_1 + H_2x_2$$

$$\Rightarrow Y - H_1x_1 = H_2x_2$$

$$\Rightarrow x_2 = \frac{H_2^*(Y - H_1x_1)}{\|H_2\|^2} \quad (5)$$

So far, we have derived  $x_2$  in terms of  $x_1$ . However there are 64 possible locations for  $x_1$  on the constellation. So, we need to get all 64 possible pairs of  $x_1$  and  $x_2$ . So, we need to repeat computing the obtained formula (5) 64 times.

Given the 64 pairs of  $x_1$  and  $x_2$  we need to calculate the Euclidean distance,  $\|Y - Hx\|^2$ .

This should also be calculated 64 times.

The following flow graph will describe the process of the algorithm.

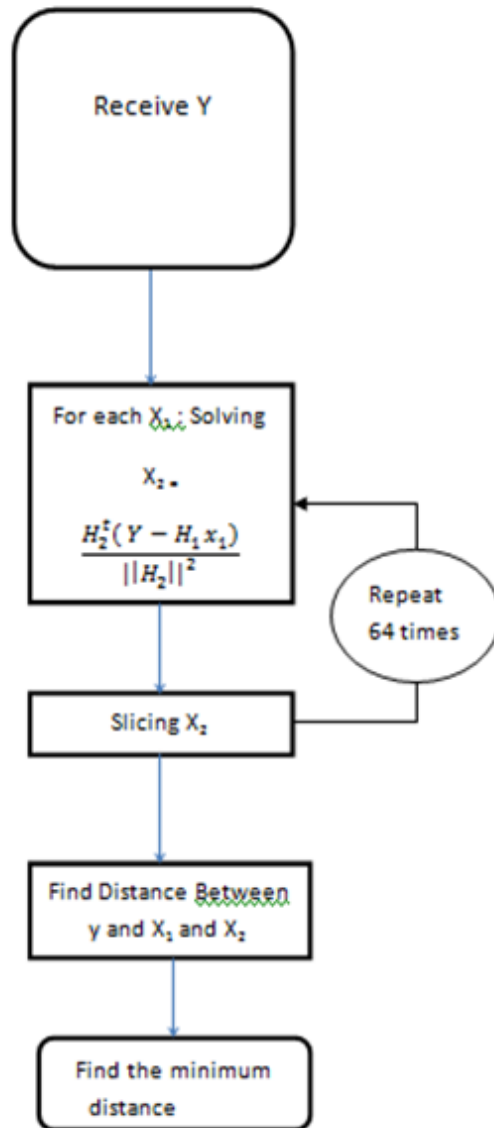


Figure 7: Flow Graph for the algorithm

The algorithm assumes that the received 2x1 matrix includes the coordinates of the points on the 64-QAM constellation.

The algorithm firstly loops around 64 points for  $x_1$  in the first constellation and each time computes  $x_2$  which represent the coordinates of the symbol in the second constellation using the formula obtained in (5).

After that a function called slice is used in order to round the coordinates of  $x_2$  to the coordinates of a point representing a symbol on the constellation. For each point it finds the distance between  $x_1$  and  $x_2$ . Then choose the combination that corresponds to the minimum distance.

### ***1. Matlab Simulation Results***

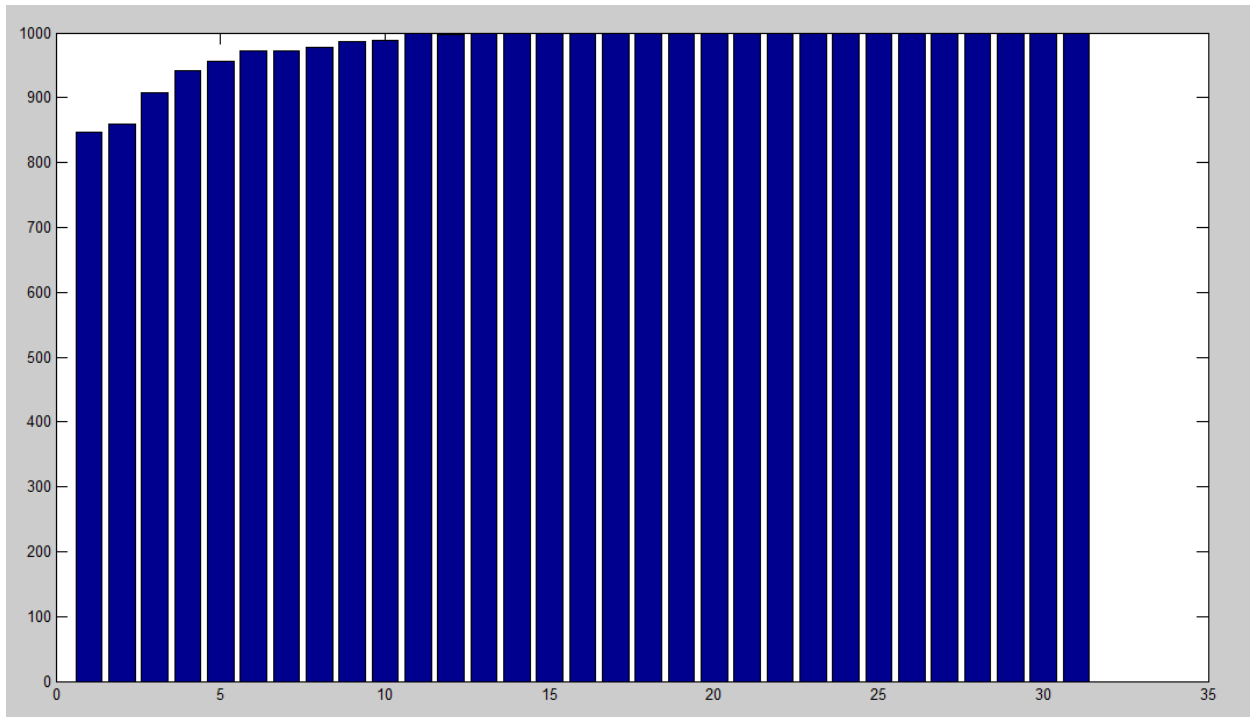
In order to verify the performance of the algorithm, Matlab was used. The basic way used to verify that our code is working was by inputting the channel matrix  $H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ .

Hence  $y_1 = x_1$  and  $y_2 = x_2$ . After ensuring that  $y_1 = x_1$  and  $y_2 = x_2$ . Now, the algorithm can be used for the testing of the vectors.

The challenge in the matlab code was determining the noise factor. According to [22], the ideal SNR for the awgn added to the signal while being transmitted is between 10-

15. After doing simulation the ideal SNR was 12 as it produced the highest detection rate when testing 1000 times.

The following graph shows the result obtained:



**Figure 8: Results of Detection rate**

According to the graph shown above, after inputting 1000 test vectors an SNR of 12 is the best to be used.

Further testing was done to the algorithm, where different test vectors were inputted and the algorithm proved to have a very high detection rate.

## CHAPTER IV

### IMPLEMENTATION OF THE PROPOSED ALGORITHM

After presenting the different blocks of the algorithm and the different approaches for implementing each of these blocks, this chapter will look at the implementation of each of the blocks in VHDL.

#### **A. The Basic Components**

Prior to implementing the main building blocks of the proposed architecture, the basic components consisting of the simple commonly used circuits was implemented and tested to simplify the construction of the complex blocks. The basic components are mainly composed of the following: two's complement, 1-bit full adder, 8-bit full adder, 16-bit full adder, 8-bit Baugh Wooley Multiplier, 16-bit Baugh Wooley Multiplier, and the shifters. In the following subsections, a brief overview of the implementation of these components is shown.

##### ***1. Basic Adders and Multipliers***

In this subsection, a brief overview of the basic gates is presented. Figure 8 shows the 1-bit full adder used in the design. The inputs are two one bits (the inputs) and the carry in, and the outputs are the sum and the carry out. Figure 9 shows an 8-bit full adder used in our design. A 16-bit full adder as used in our design. The inputs of the 8-bit full

adder are the 8 bit representation of the two input numbers and the carry in[19]. The outputs of the 8- bit adder are the sum and carry out. As for the 16 bit full adder, the inputs are the 16 bit representation of the value of the numbers to be added and the carry in. The outputs are the carry out and the sum.

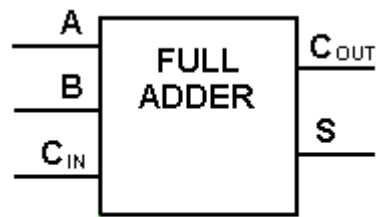


Figure 9: One bit full adder[28]

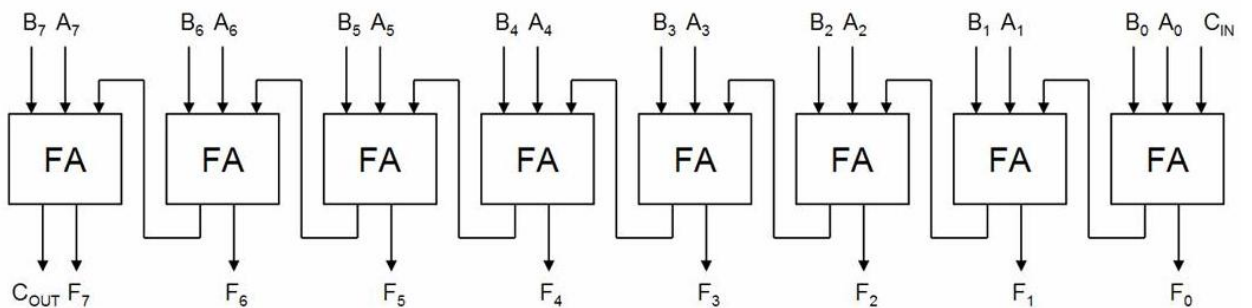
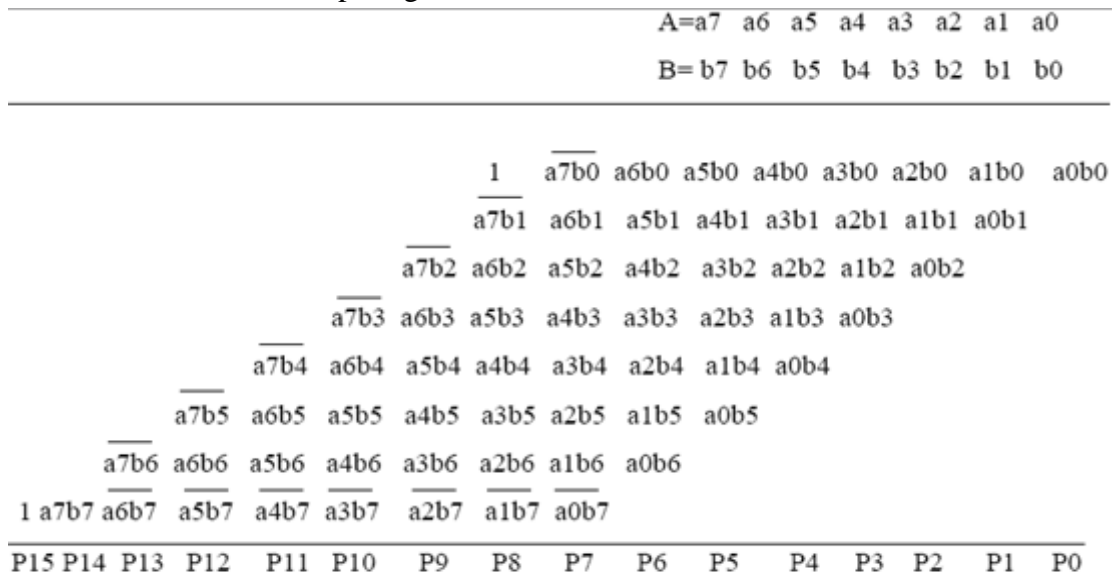


Figure 10: 8 bit full adder[29]



As for the multipliers, the multiplier used is the Baugh Wooley multiplier[15]. The Baugh Wooley multiplier is a tree multiplier was presented in Figure 10. The algorithm works in the following way[18]:

- 1- It computes the partial products and negates the last partial product except the last step.
- 2- In the last step all the partial products are negated except for the last term.
- 3- Adds 1 while computing 8<sup>th</sup> and last term.



**Figure 11: The 8 bit Baugh Wooley Multiplier Algorithm[18]**

Other existing multipliers such as Booth multiplier [23] or the Wallace trees [24].However, the Baugh Wooley multiplier algorithm is very efficient and easier to implement. It handles the sign bit of the multiplicand and the multiplier efficiently. Parallel Multipliers with booth encoding[25] can be used since they accelerate the

multiplication process. Also, bit serial multipliers can be used [26] since they use pipelining. The Wallace tree is known for their optimal computation time[24].

## ***2. Two's Complement and Shifters***

In this subsection, the two's complement and shifters are presented. The two's complement was implemented by inverting the bits and adding one[20].

As for the shifters, the shift left and the shift right operation that are needed in stage 2 of the design. The shifters have zero cost in hardware.

### **B. Stages Formulation**

#### ***1. Stage 1 Formulation and Implementation***

In the first stage of the design, the value of  $x_2$  is computed using the formula obtained in (5). However, further mathematical derivations need to be done and the output of stage one of the design should include the real and imaginary part of the computed  $x_2$ .

$$Y = \begin{bmatrix} y_{1R} + y_{1C} \\ y_{2R} + y_{2C} \end{bmatrix}$$

$$H = \begin{bmatrix} h_{11R} + i h_{11C} & h_{12R} + i h_{12C} \\ h_{21R} + i h_{21C} & h_{22R} + i h_{22C} \end{bmatrix}$$

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}.$$

$$m = \frac{H_2^t}{\|H_2\|^2} = \begin{bmatrix} a + bi \\ c + di \end{bmatrix}$$

The inputs needed for stage 1 for computing  $x_2$  :a, b, c, d,  $y_{1R}$ ,  $y_{1C}$ ,  $y_{2R}$ ,  $y_{2C}$ ,  $x_{1R}$ ,  $x_{1C}$ ,  $h_{12C}$ ,  $h_{12R}$ ,  $h_{11R}$ ,  $h_{11C}$ .

The outputs for stage 1:  $x_{2R}$ ,  $x_{2C}$

$$x_{2R} = [a^* h_{11R} * x_{1R}] + [a^* h_{11C} * x_{1C}] + [b^* h_{11C} * x_{1R}] + [b^* h_{11R} * x_{1C}] - [c^* h_{12R} * x_{1R}] + [c^* x_{1R} * h_{12C}] + [d^* x_{1C} * h_{12R}] + [d^* x_{1R} * h_{11C}] \quad (6)$$

So, to calculate the real part:

$$x_{2R} = [a^* h_{11R} + b^* h_{11C} + c^* h_{11C} + d^* h_{12C}] * x_{1R} + [a^* h_{11C} + b^* h_{11R} + c^* h_{12C} + d^* h_{12R}] * x_{1C} \quad (7)$$

$$x_{2C} = [b^* y_{2R}] - [b^* h_{11C} * x_{1R}] + [b^* h_{11C} * x_{1C}] - [a^* h_{11C} * x_{1R}] - [a^* h_{11R} * x_{1C}] + [a^* y_{2C}] + [d^* y_{2R}] - [d^* x_{1R} * h_{12R}] + [d^* x_{1C} * h_{12C}] - [c^* x_{1C} * h_{12R}] - [c^* x_{1R} * h_{12C}] + [c^* y_{2C}] \quad (8)$$

Since  $x_2$  is being computed in terms of  $x_1$ , all the products that do not include a multiplication with  $x_{1R}$  or  $x_{1C}$  can be ignored.

The following is a reduced form of the formula (8)

$$x_{2C} = [(b^* h_{11R}) + (a^* h_{11C}) + (d^* h_{12R}) + (c^* h_{12C})] * x_{1R} + [(b^* h_{11C}) - (a^* h_{11R}) + (d^* h_{12C}) - (c^* h_{12R})] * x_{1C} \quad (9)$$

The product of the inner terms is presumed to be pre computed.

Four major multiplications and two major additions are required to be implemented in VHDL of the first stage.

$$\text{Let } P1 = a^* h_{11R} + b^* h_{11C} + c^* h_{11C} + d^* h_{12C}$$

$$\text{Let } P2 = a^* h_{11C} + b^* h_{11R} + c^* h_{12C} + d^* h_{12R}$$

$$\text{Let } P3 = (b^* h_{11R}) + (a^* h_{11C}) + (d^* h_{12R}) + (c^* h_{12C})$$

Let  $P4 = [(b^* h_{11C}) - (a^* h_{11R}) + (d^* h_{12C}) - (c^* h_{12R})]$

Figure 11 includes the architecture for computing the real part of  $x_2$  and Figure 12 includes the architecture for computing the imaginary part of  $x_2$ .

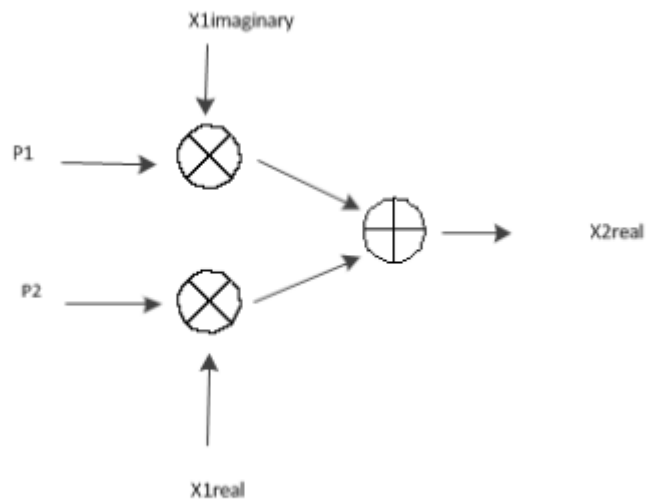


Figure 12: Architecture for computing  $x2real$

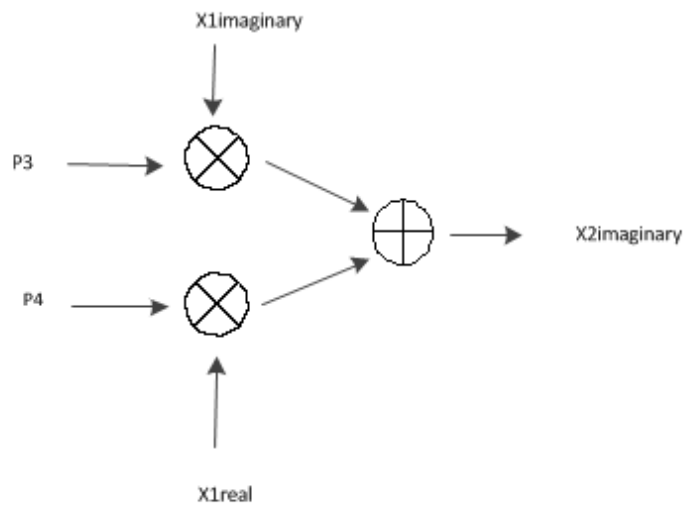


Figure 13: Architecture for computing  $x_{2imaginary}$

## VHDL

This stage requires four 8-bit multipliers (Baugh Wooley multiplier) and requires two 16-bit full adder.

The inputs in VHDL are the P1, P2,  $x_{1real}$ ,  $x_{1imaginary}$ , P3, P4 and the outputs are  $x_{2real}$  and  $x_{2imaginary}$ .

The value of each of the inputs is as follows

- $P_1 = 10100110$
- $P_2 = 10010110$
- $P_3 = 10010110$

- P4=10100111
- X1real =00000101
- X1imaginary= 00000011

The output is as shown in the figure and the values of the output are as follows:

- X2real= 111110000101100
- X2imaginary= 111110000110001

The output of the test bench of stage one is shown in the figure below  
Testbench :

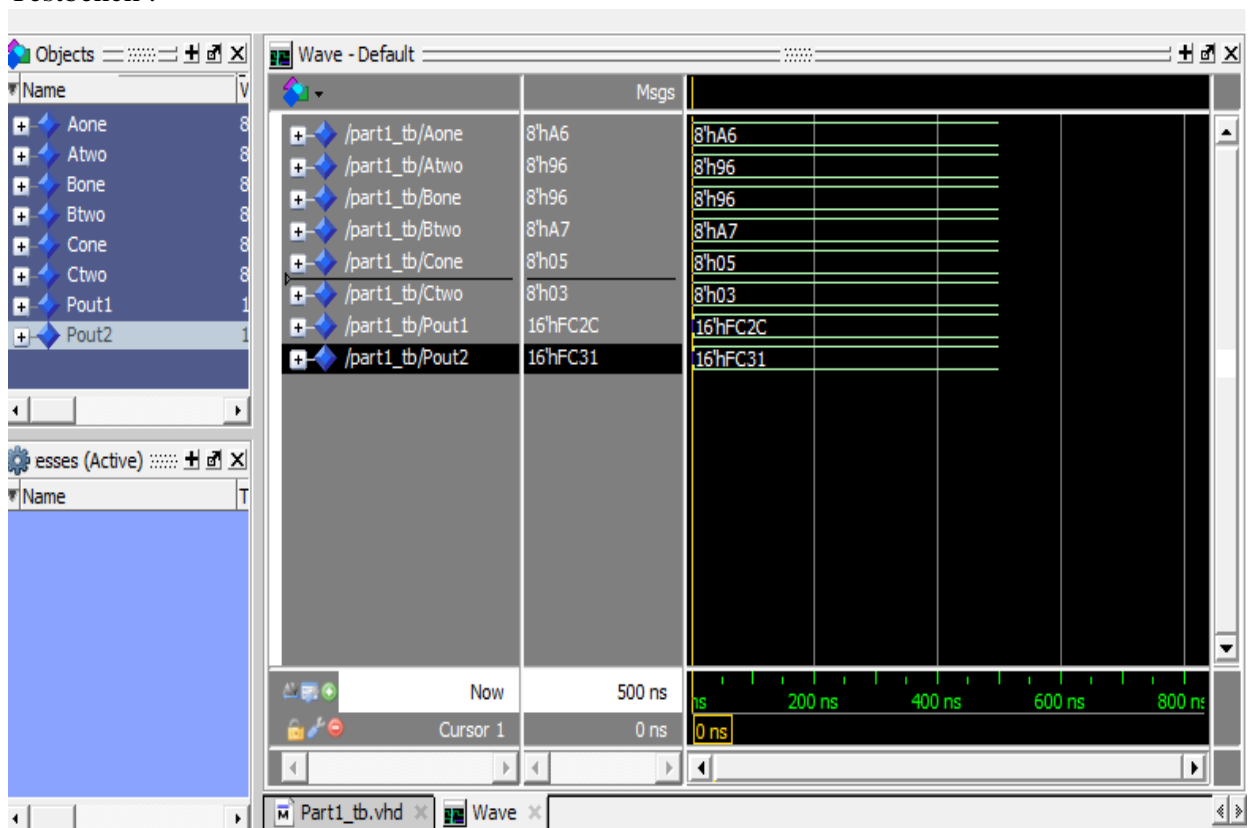


Figure 14: Output of stage 1

## **2. STAGE 2 formulation and Implementation**

In the second stage of the design, as shown in Figure7, the obtained value of  $x_{2real}$  and  $x_{2imaginary}$  are sliced. Slicing implies routing back the value of  $x_2$  to one of the possible coordinates of the 64-QAM constellation.

Derivation of the slice operation is as follows:

Since the range for the x coordinate is between -7 and 7, then we need to add 7 and divide by 2. We multiply the obtained number by 2 and add 7.

$$\begin{aligned} Y &= 2 * [(x+7)/2] - 7 \\ &= 2 * [(I.F +7) /2] - 7 \\ &= 2 * [(I' .F /2] - 7 \\ &= 2 * [I'' .F'] - 7 \\ &= 2I'' - 7 \text{ when } F' \leq 15 \text{ **OR** } 2(I'' +1) - 7 \text{ otherwise} \end{aligned}$$

**Direct Flow Graph:**

The general slice operation requires an adder, a shift left operation, a round operation, a shift right, and finally an adder. The shift right operation corresponds to a multiplication by 2. Meanwhile, the shift left operation corresponds to a division by 2.

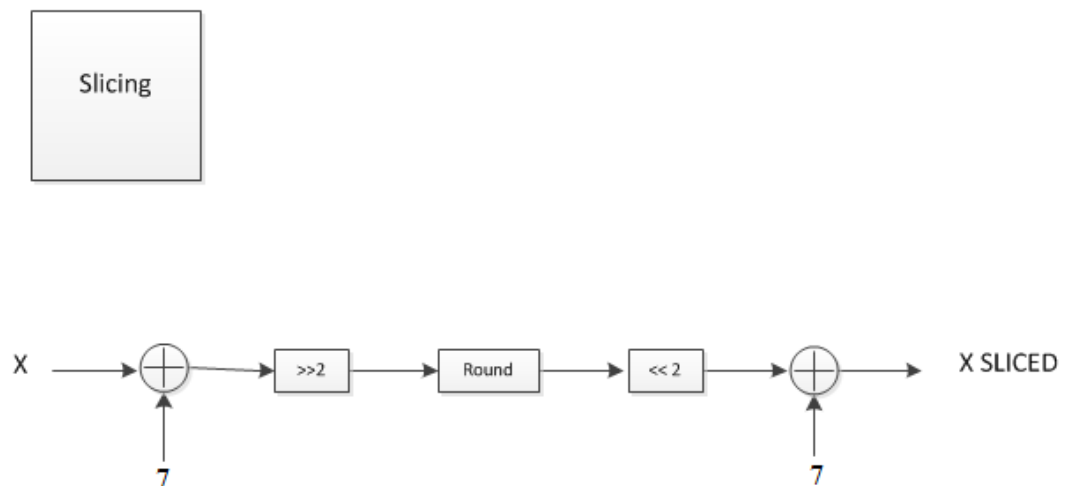


Figure 15: Architecture for slice operation

Regarding the round operation, it is done using the following method.

- If the least significant bit of the fraction part (I) is 0 it rounds it to zero.
- If the significant bit of the fraction part (I) is 1 it adds one to the fraction.

The four least significant bits are considered to be the fraction part.



This requires two Baugh Wooley multipliers and the cost incurred by the round operation which is a comparator.,

The round operation requires a comparator which is implemented by subtracting the number from the closest point and looking at the sign bit.

In the second stage the output of the third stage is added to 7. Then is right shifted once. After that the output is extracted from the testbench file. And rounded then is is right shifted then subtracted from 7.

The input for the stage2 is Lone and the output is Kout.

The input Lone which is the output for x2real obtained in stage 1 of the design is 1111110000101100.

The output obtained is shown in figure 13 and it is: 0111111000011001

The following is the testbench for part2 and the code.

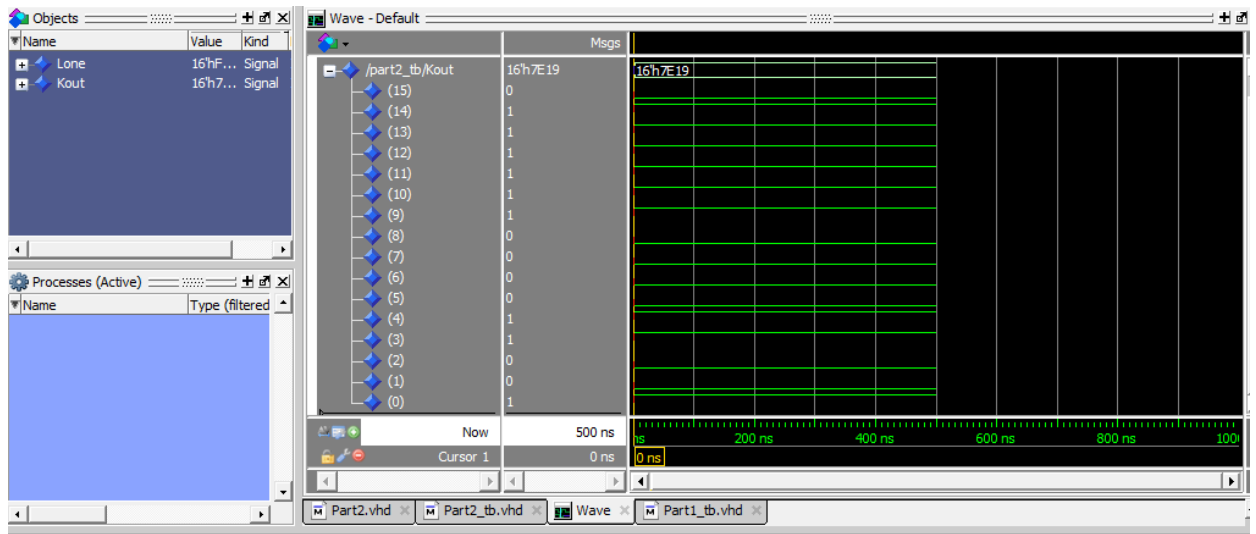


Figure 16: Output of first part of stage2

Stage 2 is split into two parts the first part performs an addition to 7 and a shift right.

The output is 0111111000011001

The Round is performed manually. Since the last four bits are 1001 then we round it to

1111

So, the new output is 0111111000101001

This is fed to the second part where its subtracted from seven then fed to the shiftleft

two component

The following is a screenshot of the final answer:



Figure 17: Final Answer for stage2

The final answer of stage 2 is 1111110001100000

### 3. Stage3 Formulation and Implementation

In stage 3 of the design, as shown in Figure 7, the Euclidean distance is required to be found. This is required since the main aim of the thesis is to solve the Maximum Likelihood Rule using minimal number of computations.

Similar derivation to the first stage occurred:

$$d = (y_{2R} - h_{12R}x_{1R} + h_{12C}x_{1C} - h_{22R}x_{2R} + h_{22C}x_{2C})^2 - (y_{2C} - h_{12C}x_{1C} - h_{12C}x_{1R} - h_{22R}x_{2C} - h_{22C}x_{2R})^2 + i(2 * (y_{2R} - h_{12R}x_{1R} + h_{12C}x_{1C} - h_{22R}x_{2R} + h_{22C}x_{2C}) * (y_{2C} - h_{12C}x_{1C} - h_{12C}x_{1R} - h_{22R}x_{2C} - h_{22C}x_{2R}))$$

For the real part

$$D_{real} = (y_{1R} - h_{11R}x_{1R} + h_{11C}x_{1C} - h_{21R}x_{2R} + h_{21C}x_{2C})^2 - (y_{1C} - h_{11C}x_{1R} - h_{11R}x_{1C} - h_{21R}x_{2C} - h_{21C}x_{2R})^2$$

This is equivalent to  $a^2 - b^2 = (a-b)(a+b)$

We can simplify this to:

$$(y_{1R} - h_{11R} x_{1R} + h_{11C} x_{1C} - h_{21R} x_{2R} + h_{21C} x_{2C} - y_{1C} + h_{11C} x_{1R} + h_{11R} x_{1C} + h_{21R} x_{2C} + h_{21C} x_{2R}) (y_{1R} - h_{11R} x_{1R} + h_{11C} x_{1C} - h_{21R} x_{2R} + h_{21C} x_{2C} + y_{1C} - h_{11C} x_{1R} - h_{11R} x_{1C} - h_{21R} x_{2C} - h_{21C} x_{2R})$$

$$(-h_{11R} x_{1R} + h_{11C} x_{1C} - h_{21R} x_{2R} + h_{21C} x_{2C} + h_{11C} x_{1R} + h_{11R} x_{1C} + h_{21R} x_{2C} + h_{21C} x_{2R}) (-h_{11R} x_{1R} + h_{11C} x_{1C} - h_{21R} x_{2R} + h_{21C} x_{2C} - h_{11C} x_{1R} - h_{11R} x_{1C} - h_{21R} x_{2C} - h_{21C} x_{2R})$$

Then a similar grouping of the variables having  $x_{1R}, x_{1C}, x_{2R}, x_{2C}$  as their coefficient and the above equation reduces to

$$(x_{1R}(-h_{11R} + h_{11C}) + x_{1C}(h_{11C} + h_{11R}) + x_{2R}(-h_{21R} + h_{21C}) + x_{2C}(h_{21C} - h_{21R})) (x_{1R}(-h_{11R} - h_{11C}) + x_{1C}(h_{11C} - h_{11R}) + x_{2R}(-h_{21R} + h_{21C}) + x_{2C}(h_{21C} - h_{21R}))$$

Which simplifies further to:

$$(x_{1R}(-h_{11R} + h_{11C}) + x_{1C}(h_{11C} + h_{11R}) + x_{2R}(-h_{21R} + h_{21C})) (x_{1R}(-h_{11R} - h_{11C}) + x_{1C}(h_{11C} - h_{11R}) + x_{2R}(-h_{21R} + h_{21C}))$$

In this case we have four major additions and 7 major multiplications

$$\text{Let } A = x_{1R}(-h_{11R} + h_{11C}) + x_{1C}(h_{11C} + h_{11R}) + x_{2R}(-h_{21R} + h_{21C})$$

$$\text{Let } B = (x_{1R}(-h_{11R} - h_{11C}) + x_{1C}(h_{11C} - h_{11R}) + x_{2R}(-h_{21R} + h_{21C}))$$

Figure 18 and Figure 19 show the architecture for computing A and B.

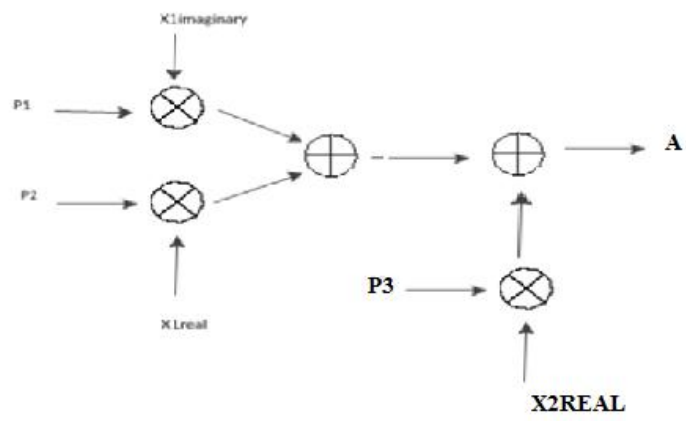


Figure 18: Architecture to Compute A

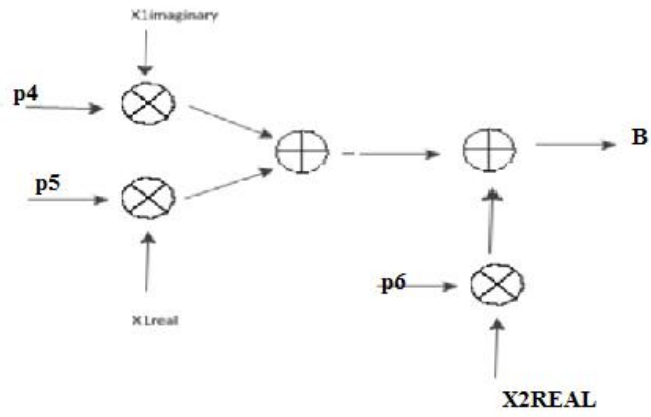


Figure 19: Architecture for Computing B

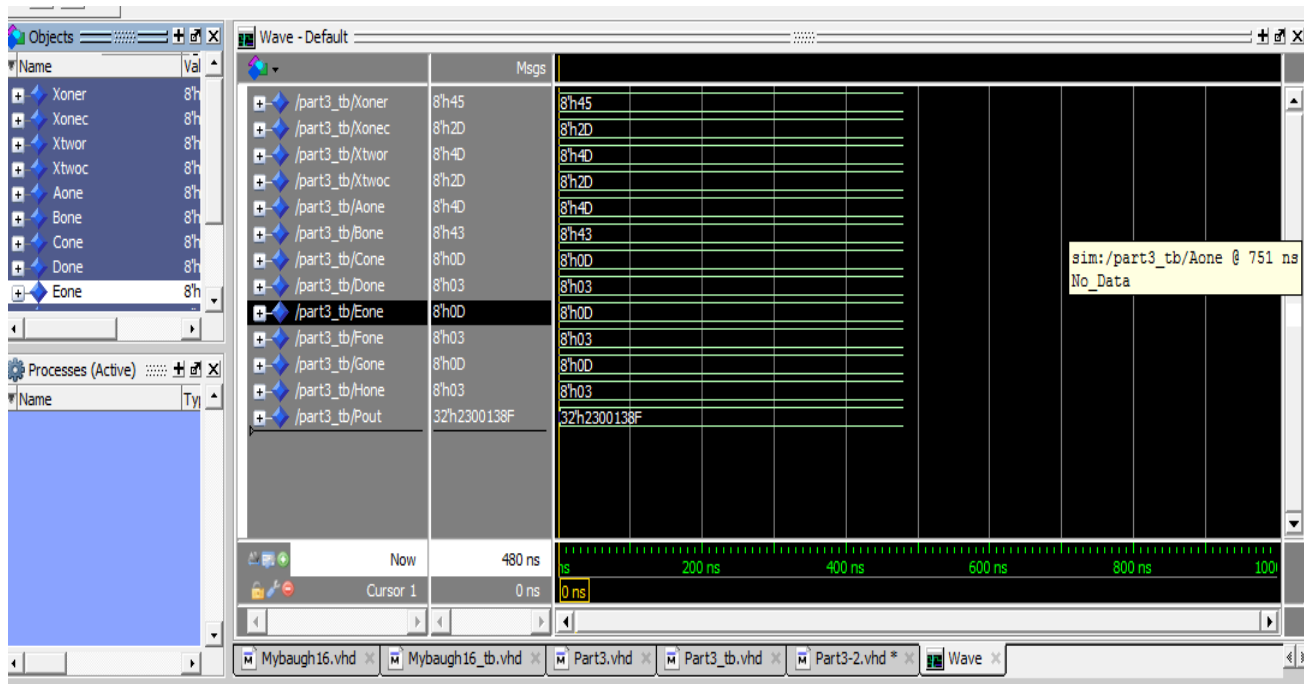


Figure 20: Final Answer for stage 3 real part

Regarding the imaginary part:

$$x_{2C} = 2 * (y_{1R} - h_{11R} x_{1R} + h_{11C} x_{1C} - h_{21R} x_{2R} + h_{21C} x_{2C}) * (y_{1C} - h_{11C} x_{1R} - h_{11R} x_{1C} - h_{21R} x_{2C} - h_{21C} x_{2R})$$

$$x_{2C} = 2 * (y_{2R} - [h_{12R} * x_{1R}] + [h_{12C} * x_{1C}] - [h_{22R} * x_{1R}] + [h_{22C} * x_{1C}]) * (y_{2C} - (h_{12R} * x_{1C}) - (h_{12C} * x_{1R}) - [h_{22R} * x_{1C}] - [h_{22C} * x_{1R}]) ;$$

Reduced:

$$x_{2C} = 2 * ([h_{12R} * x_{1R}] + [h_{12C} * x_{1C}] - [h_{22R} * x_{1R}] + [h_{22C} * x_{1C}]) * ((h_{12R} * x_{1C}) - ((h_{12C} * x_{1R}) - [h_{22R} * x_{1C}] - [[h_{22C} * x_{1R}]]);$$

We need 6 major additions and 7 major multiplications and a shift right operation.

I assumed for simplicity that the factor in between brackets is precalculated and hence similar to stage one is predefined.

$$\text{Let } C = [(h_{12R} * X_{1R})] + [h_{12C} * X_{1C}] - [h_{22R} * X_{1R}] + [h_{22C} * X_{1C}]$$

$$\text{Let } D = ((h_{12R} * X_{1C})) - ((h_{12C} * X_{1R})) - [h_{22R} * X_{1C}] - [[h_{22C} * X_{1R}]$$

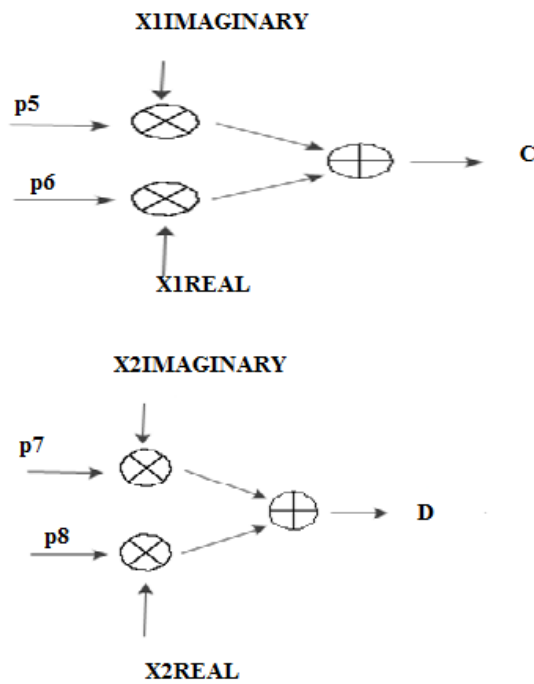


Figure 21: Architecture for Computing the Imaginary part



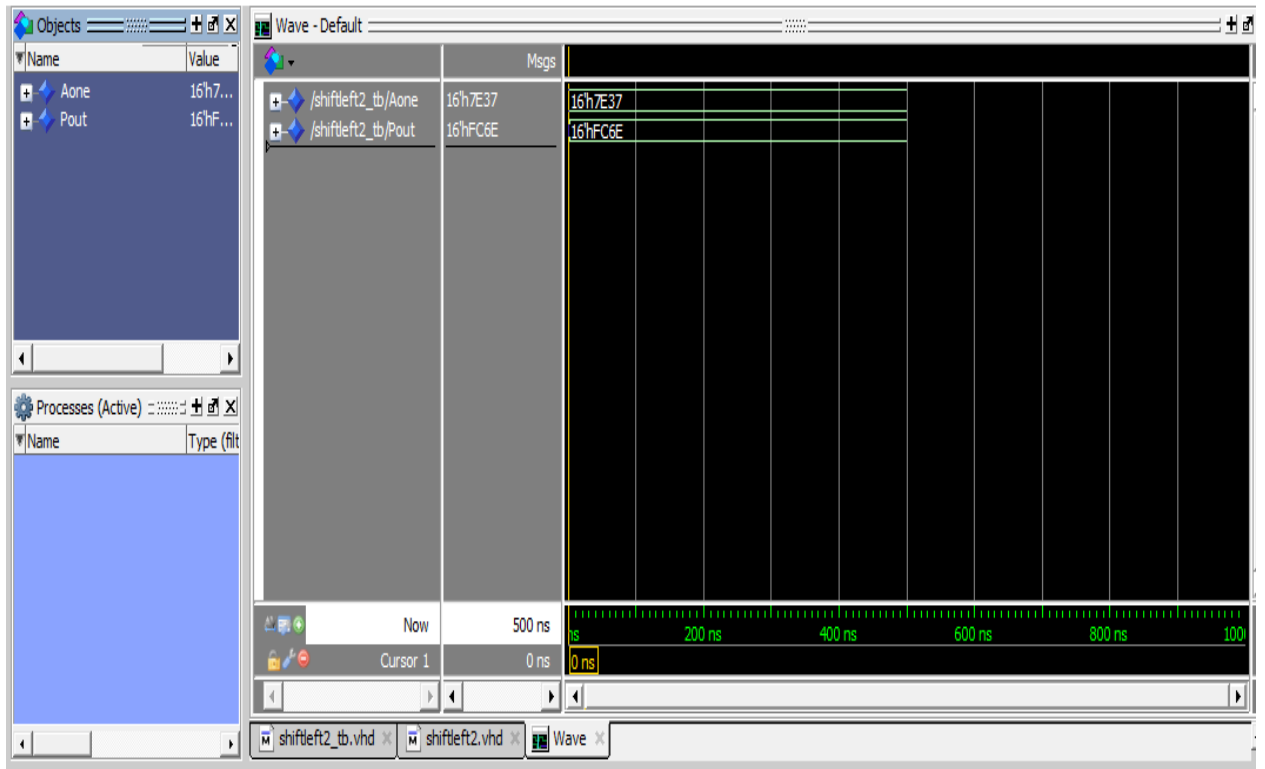


Figure 22: Final Answer for imaginary part

#### 4. The Overall Design

In the following subsection a brief overview of the overall design is described along with an illustration showing the flow of data into the different building blocks.

The first building block is the Detection Algorithm block where the data is inputted.

The main aim of this block is to perform the computation of  $x_2$  using the formula () as shown in the description of the first stage.

The second building block is the Slicing block. The purpose of this block is to map the retrieved output of the Detection Algorithm block to one of the possible constellation points. This is done in two steps as mentioned in the description of the second stage.

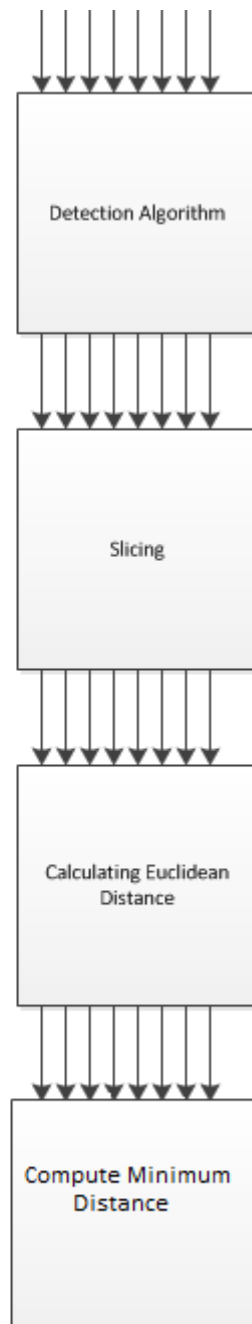
The third building block is the Calculation of the Euclidean Distance. The purpose of this block is to compute the Euclidean distance which is required to obtain the optimal solution that satisfies the Maximum Likelihood Rule.

The final building block is the calculation of the minimum Euclidean Distance. The purpose of this block is to compute the minimum of the 64 points obtained.

It should be noted that 8 arrows are being fed to the building block. This means that we should feed 8 inputs eight times to perform the computation for all 64 possible computations.

In order to compute all 64 computations this can be done as shown in Figure 18. The figure shows that 8 of the overall block (as shown in Fig 19) are needed to implement the overall design.

In Fig.19, there are 8 inputs to be fed to each of these blocks. Therefore, the arrows indicate the inputs. Each input is the real and imaginary part.



**Figure23: Architecture for implementing overall Design**

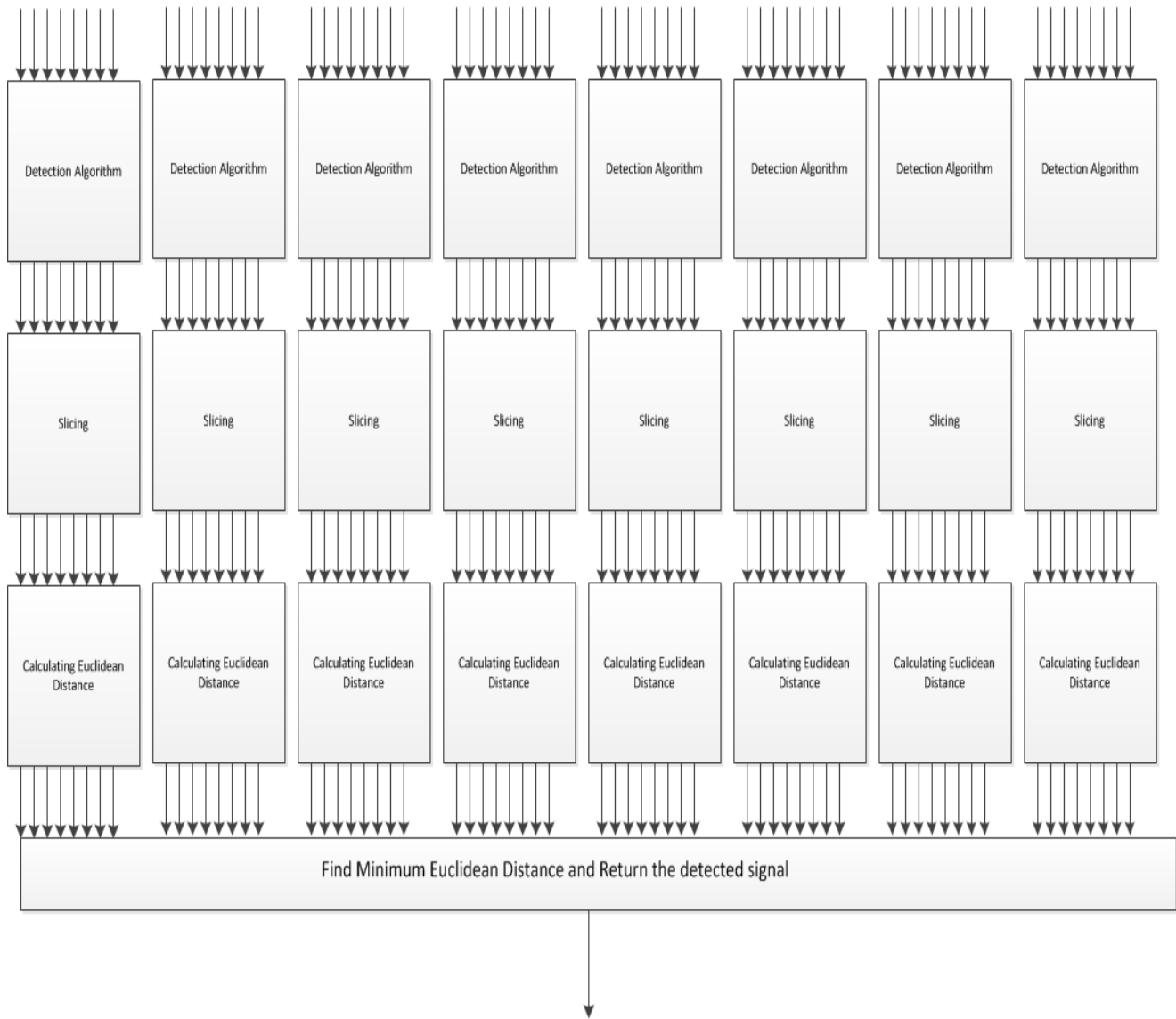


Figure 104:Final Design

## CHAPTER V

### Conclusion and Future Work

Through this work an overview of detection algorithms for MIMO is presented. A new algorithm is approached and a preliminary architecture is presented. The Fitz algorithm reduces the computation significantly as compared to the existing algorithms. The thesis will present a new algorithm and an efficient architecture for the algorithm. The efficiency will be in terms of power and latency.

Future work could focus on finding further more efficient ways of carrying out the channel estimation and data detection and also on achieving better results than those existing for the MIMO case. Also, future modifications could be in terms of hardware resources which could reduce power or increase the speed.

## REFERENCES

- [1] Windpassinger, C.; Lampe, L.; Fischer, R. F H & Hehn, T., "A performance study of MIMO detectors," *Wireless Communications, IEEE Transactions on* , vol.5, no.8, pp.2004,2008, Aug. 2006
- [2] Larsson, E.G., "MIMO Detection Methods: How They Work," *Signal Processing Magazine, IEEE* , vol.26, no.3, pp.91,95, May 2009
- [3] C. Langton, Intuitive Guide to Principles of Communications, Available at [www.complextoreal.com](http://www.complextoreal.com), 2004
- [4]J. Proakis, "Optimum Receivers for the Additive White Gaussian Noise," *Digital Communications*, 4<sup>th</sup> ed. McGraw-Hill, 2000, pp. 233-307
- [5]Hassibi, B. & Vikalo, H., "On the sphere-decoding algorithm I. Expected complexity," *Signal Processing, IEEE Transactions on* , vol.53, no.8, pp.2806,2818, Aug. 2005
- [6]Wubben, D.; Bohnke, R.; Kuhn, V. & Kammeyer, K. -D, "MMSE extension of V-BLAST based on sorted QR decomposition," *Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003 IEEE 58th* , vol.1, no., pp.508,512 Vol.1, 6-9 Oct. 2003
- [7]Berenguer, I.; Adeane, J.; Wassell, I.J. & Xiaodong Wang, "Lattice-reduction-aided receivers for MIMO-OFDM in spatial multiplexing systems," *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium on* , vol.2, no., pp.1517,1521 Vol.2, 5-8 Sept. 2004
- [8] Verma, M. P. K., & Dua, R. L. Performance Analysis Of Energy Detection, Matched Filter Detection & Cyclostationary Feature Detection Spectrum Sensing Techniques. *Indian Streams Research Journal*, (2012)
- [9]Cheng Wang; Au, E.K.S.; Murch, R.D.; Wai-Ho Mow; Cheng, R.S. & Lau, V., "On the Performance of the MIMO Zero-Forcing Receiver in the Presence of Channel

Estimation Error," *Wireless Communications, IEEE Transactions on* , vol.6, no.3, pp.805,810, Mar. 2007

[10]Wubben, D.; Bohnke, R.; Kühn, V.; Kammeyer, K. -D, "Near-maximum-likelihood detection of MIMO systems using MMSE-based lattice reduction," *Communications, 2004 IEEE International Conference on* , vol.2, no., pp.798,802 Vol.2, 20-24 Jun. 2004

[11]K. Su;"Efficient Maximum Likelihood Detection for Communication over Multiple Input Multiple Output Channels", Ph.D. dissertation, Dept. Eng., Univ. Cambridge, England, 2005.

[12]Burg, A.; Wenk, M.; Zellweger, M.; Wegmueller, M.; Felber, N. & Fichtner, W., "VLSI implementation of the sphere decoding algorithm," *Solid-State Circuits Conference, 2004. ESSCIRC 2004. Proceeding of the 30th European* , vol., no., pp.303,306, 21-23 Sept. 2004

[13] T. Kailath and H.V.Poor, "Detection of Stochastic Processes," *IEEE Trans. Inform. Theory*, vol.44, no.6, Oct.1998

[14] Shariat-Yazdi, R.; Kwasniewski, T., "A low complexity VLSI architecture for MIMO sphere decoding algorithm," *Circuits and Systems and TAISA Conference, 2009. NEWCAS-TAISA '09. Joint IEEE North-East Workshop on* , vol., no., pp.1,4, June 28 2009-July 1 2009

[15] Kroft, D., "Comments on "A Two's Complement Parallel Array Multiplication Algorithm"," *Computers, IEEE Transactions on* , vol.C-23, no.12, pp.1327,1328, Dec. 1974

[16] Damen, M.-O.; El-Gamal, H.; Caire, G., "On maximum-likelihood detection and the search for the closest lattice point," *Information Theory, IEEE Transactions on* , vol.49, no.10, pp.2389,2402, Oct. 2003

[17] Wenk, M.; Zellweger, M.; Burg, A.; Felber, N.; Fichtner, Wolfgang, "K-best MIMO detection VLSI architectures achieving up to 424 Mbps," *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on* , vol., no., pp.4 pp.,1154, 21-24 May 2006

[18] Sjalander, M.; Larsson-Edefors, P., "High-speed and low-power multipliers using the Baugh-Wooley algorithm and HPM reduction tree," *Electronics, Circuits and*  
46

*Systems, 2008. ICECS 2008. 15th IEEE International Conference on* , vol., no., pp.33,36, Aug. 31 2008-Sept. 3 2008

[19] Alioto, M.; Palumbo, G., "Analysis and comparison on full adder block in submicron technology," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* , vol.10, no.6, pp.806,823, Dec. 2002

[20] Wires, K.E.; Schulte, M.J.; Marquette, L.P.; Balzola, P.I., "Combined unsigned and two's complement squarers," *Signals, Systems, and Computers, 1999. Conference Record of the Thirty-Third Asilomar Conference on* , vol.2, no., pp.1215,1219 vol.2, 24-27 Oct. 1999

[21] Nirmalkumar S. Reshamwala, Pooja S. Suratia, Satish K. Shah, "Study of ANN Configuration on Performance of Smart MIMO Channel Estimation for Downlink LTE-Advanced System", *IJCNIS*, vol.5, no.11, pp.27-35, 2013.

[22] J.SUN, Y.Dong-Feng, "Neural Network Channel Estimation Based on Least Mean Error Algorithm in the OFDM Systems", *International Symposium on Neural Networks (ISNN) 2006*, Chengdu, China.

[23] Jacobsohn, D., "A Suggestion for a Fast Multiplier," *Electronic Computers, IEEE Transactions on* , vol.EC-13, no.6, pp.754,754, Dec. 1964)

[24] Yuan-Ho Chen; Chung-Yi Li; Tsin-Yuan Chang, "Area-Effective and Power-Efficient Fixed-Width Booth Multipliers Using Generalized Probabilistic Estimation Bias," *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on* , vol.1, no.3, pp.277,288, Sept. 2011

[25] Yajuan He; Chip-Hong Chang, "A New Redundant Binary Booth Encoding for Fast  $2^m$ -Bit Multiplier Design," *Circuits and Systems I: Regular Papers, IEEE Transactions on* , vol.56, no.6, pp.1192,1201, June 2009

[26] Hsu, I.-S.; Reed, Irving S.; Truong, T.-K.; Ke Wang; Chiunn-Shyong Yeh; Deutsch, L.J., "The VLSI Implementation of a Reed-Solomon Encoder Using



Berlekamp's Bit-Serial Multiplier Algorithm," *Computers, IEEE Transactions on* ,  
vol.C-33, no.10, pp.906,911, Oct. 1984

[27] Langton C., "Intuitive Guide to principle of communication" Internet:  
[www.complextoreal.com](http://www.complextoreal.com) [May 15,2014].

[28] John,"Half Adder and Full Adder Circuits" Internet: [www.circuitstoday.com](http://www.circuitstoday.com) [May  
15,2014]

[29] "Attempt to Demonstrate Look Ahead Process" Internet:  
[HTTP://WWW.ECE.UNM.EDU/FACULTY/POLLARD/CLASSES/338/LADEMO/L  
OOKAHEADDEMO.HTM](http://WWW.ECE.UNM.EDU/FACULTY/POLLARD/CLASSES/338/LADEMO/LOOKAHEADDEMO.HTM) [MAY 15, 2014]

[29] John G. Proakis.(November 6, 2007).Digital Communications(fifth).