2013

ON REDUCING THE DECODING COMPLEXITY OF SHINGLED MAGNETIC RECORDING SYSTEM

Awad, Nadia

ON REDUCING THE DECODING COMPLEXITY OF SHINGLED MAGNETIC RECORDING SYSTEM

by

Nadia Awad

Ph.D.

May 2013

ON REDUCING THE DECODING COMPLEXITY OF SHINGLED MAGNETIC RECORDING SYSTEM

A thesis submitted to Plymouth University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Nadia Awad

May 2013

School of Computing and Mathematics

Faculty of Science and Technology

Plymouth University, UK

Abstract

On Reducing the Decoding Complexity of Shingled Magnetic Recording System

Nadia Awad

Shingled Magnetic Recording (SMR) has been recognised as one of the alternative technologies to achieve an areal density beyond the limit of the perpendicular recording technique, 1 Tb/in^2, which has an advantage of extending the use of the conventional method media and read/write head.

This work presents SMR system subject to both Inter Symbol Interference (ISI) and Inter Track Interference (ITI) and investigates different equalisation/detection techniques in order to reduce the complexity of this system.

To investigate the ITI in shingled systems, one-track one-head system model has been extended into two-track one-head system model to have two interfering tracks. Consequently, six novel decoding techniques have been applied to the new system in order to find the Maximum Likelihood (ML) sequence. The decoding complexity of the six techniques has been investigated and then measured. The results show that the complexity is reduced by more than three times with 0.5 dB loss in performance.

To measure this complexity practically, perpendicular recording system has been implemented in hardware. Hardware architectures are designed for that system with successful Quartus II fitter which are: Perpendicular Magnetic Recording (PMR) channel, digital filter equaliser with and without Additive White Gaussian Noise (AWGN) and ideal channel architectures. Two different hardware designs are implemented for Viterbi Algorithm (VA), however, Quartus II fitter for both of them was unsuccessful. It is found that, Simulink/Digital Signal Processing (DSP) Builder based designs are not efficient for complex algorithms and the eligible solution for such designs is writing Hardware Description Language (HDL) codes for those algorithms.

To my beloved

Mother

and

sister; Fatin

for their support, encouragement and constant love

Acknowledgments

First and foremost, praises and thanks to Allah, the Almighty, for His showers of blessings throughout my research work.

I would like to express my deepest appreciation to all those who helped and supported me in my research.

I will always be in debt to my first supervisor Mohammed Zaki Ahmed for his excellent guidance, encouragement, full support, deep intuition, and constant help throughout my research. His mark in my PhD time and in my life will never be forgotten.

My second supervisor, Paul Davey, thanks a lot for the useful discussions which have broadened my knowledge. Professor Martin Tomlinson, much appreciated your support.

My deepest thanks and gratitude extend to Dr. Mustafa M. Aziz at Exeter University who introduced me to my supervisor Zaki without whom I would not achieve what I achieved now.

I would also like to thank the Ministry of Higher Education and Scientific Research in Iraq for the financial support.

I would also like to offer my sincere thanks to all my colleagues in Smeaton 208 for their support. My colleague, Is-Haka Mkwawa, thanks a lot for your help on Latex, support, proof-reading my thesis and various discussions. You are always willing to help and give the best suggestions and solutions.

My greatest thanks must go to all Iraqi's friends and colleagues at Plymouth University especially Samahir, Hanady, Haseba, Ali Al-Timemy, Saif, Salah and Adhraa for their help and support which made my time in Plymouth so enjoyable.

Lastly, I would like to thank my family for all their love and moral support and sacrifices. My mother, brother and sisters have given so much to bring me up. You are the people who have made me who I am and I would be nothing without you. I love you more than you can imagine.

Author's Declaration

At no time during the registration for the degree of Doctor of Philosophy has the author been registered for any other University award without prior agreement of the Graduate Committee.

This study was financed under the Ministry of Higher Education and Scientific Research in Iraq scholarship for "Data Communications".

A programme of advanced study was undertaken, which included the extensive reading of literature relevant to the research project, development of softwares based simulations, development of practical designs and attendance of international conferences on magnetic recording area.

The author has submitted papers at The 2012 IEEE Intermag Conference at Vancouver, Canada, The 2013 IEEE Joint MMM Intermag Conference at Colorado, USA and The 2013 3rd International Symposium on Advanced Magnetic Materials and Applications.

In addition the author attended The 2011 IEEE Intermag Conference at Taipei, Taiwan.

Word count of main body of thesis: 41004

Signed.....

Date.....

Contents

List of Tables

List of Figures

Abbreviations

2-D Two-Dimensional

ADC Analog-to-Digital Converter

APP A Posteriori Probabilities

AWGN Additive White Gaussian Noise

BER Bit Error Rate

BPM Bit-Patterned Media

CTF Continuous Time Filter

DAC Digital-to-Analog Converter

DSP Digital Signal Processing

FIR Finite Impulse Response

FPGA Field-Programmable Gate Array

GMR Giant Magneto-Resistance

GPR Generalized Partial Response

GPR3 Class-III Generalized Partial Response

GUI Graphical User Interface

HAMR Heat-Assisted Magnetic Recording

HDD Hard Disk Drive

HDL Hardware Description Language

ISI Inter Symbol Interference

ITI Inter Track Interference

LMR Longitudinal Magnetic Recording

MAMR Microwave Assisted Magnetic Recording

MAP Maximum A Posteriori

ML Maximum Likelihood

MLSD Maximum-Likelihood Sequence Detection

MMSE Minimum Mean-Square Error

MR Magneto Resistive

OTI Off-Track Interference

pdf probability density function

PMR Perpendicular Magnetic Recording

PR Partial Response

PR4 Class IV Partial Response

PRBS Pseudo Random Binary Sequence

PRML Partial Response Maximum Likelihood

RLL Run-Length-Limited

RTL Register-Transfer Level

SMR Shingled Magnetic Recording

SNR Signal to Noise Ratio

SUL Soft Magnetic Underlayer

TDMR Two-Dimensional Magnetic Recording

VA Viterbi Algorithm

VGA Variable-Gain Amplifier

VHDL Very high speed integrated circuits (VHSIC) Hardware Description Language

Chapter 1

Introduction

The digital lifestyle which includes accessing the internet, using cash machines, watching Hollywood movies and trading stocks online demand tremendous amounts of digital information. Most information is stored in four types of physical media that are: paper (books and newspapers), film (photographs and cinema), optical (DVDs) and magnetic (PC disk drives and camcorder tape) [21]. However, magnetic storage is the largest medium used for storing information [21].

According to the study of University of California-Berkeley in 2003, the world produces between 1 and 2 exabytes (exabyte is a billion gigabytes or 10^{18}) of unique information per year (about 250 megabytes per person) [21]. Then, another study was forecasted explosive growth of the digital universe from 281×10^{18} (281 exabytes) in 2007 (about 45 GB per person) to 1.8×10^{21} (1.8 Zetta) Bytes in 2011 [8]. Thus, it is necessary to ensure the rapid increase in the capacity of HDD.

In 1999, `The First 100 Years` book was published on the 100^{th} anniversary of the data storage technologies. Data storage area could be considered as a very mature area whose trend in magnetic recording has focused on smaller, faster, cheaper and denser devices.

A brief history of Hard Disk Drive (HDD) is outlined in this chapter followed by reviewing the magnetic recording technologies. The main purpose of this work and the contributions that the author achieved are outlined in aim and objectives and contribution to knowledge sections.

1.1 History of Hard Disk Drive

This section highlights some of the historical events that happened in the field of hard disk storage which had resulted to increase the storage capacity such as:

- 1888- the first magnetic recording head was invented by Oberlin Smith. Then in 1898, Valdemar Poulsen built the first functioning recorder using Oberlin head design [10].

- 1956- the first commercial HDD was invented by IBM with a storage capacity of five million characters [22].

- 1961- Model 1301 disk drive was shipped by IBM with a capacity of 56 MB [10].

- 1971- Magneto Resistive (MR) head was discovered which accelerated HDD development 20 years later [23]. However, the MR effect was first found in 1857 [2] and the 2007 Nobel prize in physics was awarded to scientists who discovered the MR effect [24]. The first use of the MR head in commercial disk drive was developed by IBM in 1992 [10].

  The design of MR head is based on the concept of the ability of metals to change their resistivity in the presence of a magnetic field [2] [25].

  During readback process, the recorded magnetic patterns that are represented by the alternating magnetic pols N and S create fringing magnetic fields. The read head senses these fields which causes a change in the resistance of the stripe on the MR head. Consequently, an output voltage is produced that varies according to that magnetic patterns [24].

- 1972- The structure of ML estimator was introduced [26].

- 1974- BCJR algorithm, named after the authors (Bahl, Cocke, Jelinek and Raviv) of [15], was developed to minimize the probability of symbol (or bit) error on trellis based constraints.

- 1976- A magnetic head for perpendicular recording was proposed by Iwasaki, *et al* [27]. It consists of two poles, the main pole for recording and the auxiliary pole. This head produces a field whose perpendicular component has an intensive and sharp distribution.

- 1979- VA was possible to apply in magnetic recording systems for detection purposes [10].

- 1979- IBM 3370 introduced thin film heads technology however working on thin-film head structures was started in the late-1960s [28].

- 1990- First HDD with Partial Response Maximum Likelihood (PRML) algorithm was introduced by IBM [10].

- 1992- Marcellin and Weber introduced two-dimensional modulation code which increased the data storage density by working on multiple tracks in parallel [29]. However, Swanson and Wolf introduced a new class of two-dimensional Run-Length-Limited (RLL) recording codes that added an additional constraint across each track [30].

- 2001- Different types of target for PRML channel were investigated in [31]. The paper confirmed that the Class-III-based Generalized Partial Response (GPR3) target had a better performance.

- 2001- Wood, *et al.* in [32] confirmed that the media characteristics of perpendicular recording have a significant effect on the areal densities.

- 2002- Seagate applied Heat-Assisted Magnetic Recording (HAMR) technology into HDD by cramming one Terabit into a square inch [33].
- 2005- Toshiba shipped the first HDD using PMR technology. This new technology was applied into two high capacity drives: 1.8″ 40/80 GB [34].
- 2006- A new optical drive element was developed by Fujitsu using HAMR [35].
- 2007- HGST announced industry's first Terabyte hard drive using PMR technology [36].
- 2008- Seagate unveiled the industry's first 1.5TB desktop and 0.5TB notebook hard drives [37]
- 2010- Western Digital announced shipping 3TB internal hard disk drive [38].
- 2011- Seagate shipped the first 4TB external HDD [39].
- 2012 - Western Digital announced the first 2.5-inch, 5*mm* thick drive, and the first 2.5-inch, 7*mm* thick drive with two platters [40].
- 2012 - TDK demonstrated 2TB on a single 3.5-inch platter using HAMR technology [41].

1.2 Communication Channel for Magnetic Storage

Communications mean transmission of information from a source to a destination through a channel. Depending on the physical medium that uses to transmit these information, different types of channels are there such as:

1. **Wireline channels**: like wire lines for voice signal transmission of telephone network and twisted-pair wire lines and coaxial cable for electromagnetic channels.

2. **Fiber optic channels**: the information is carried on a modulated light beam. The bandwidth of these channels is larger than coaxial cable channels which offers a wide variety of telecommunication services such as voice, data, facsimile and video.

3. **Wireless electromagnetic channels**: transmit the information using electromagnetic waves through the atmosphere (free space) by radiating them via antenna.

4. **Underwater acoustic channels**: data transmission is performed by sensors placed under water which transmit the data to the surface of the ocean.

5. **Storage channels**: include magnetic tape (digital audio tape and video tape), magnetic disks, optical disks and compact disks which can be considered as communication channels where the data are written on a disk and then read out at a later time.

The difference between data storage channel and communication channel is the domain in which the communication occurs. For the data storage channel, the communication is represented in time domain where, the information is transported from one time to another. However, different domains are used for the communication channels as mentioned above. Furthermore, there is a big difference between the two channels in terms of Bit Error Rate (BER) . In communication systems, the BER would be 10^{-5} or 10^{-6}. In contrast, 10^{-12} error rate or better is the BER requirement for storage systems [42].

1.3 Data Storage Devices

Permanent storage device stores data (information) permanently unless it is physically changed or deleted by the user. Therefore, this information will not be lost when the power is switched off. On contrast, in read access memory (RAM) the information will be lost when the power is turned off. The most common permanent storage devices are: magnetic

disk, optical disk, magnetic tape, flash memory, floppy disk, solid-state drive, hybrid drive and external drives.

These devices could be classified according to their capacity, portability and the access method. Access method means the way by which data would be accessed which is either directly or sequentially. Magnetic tape is an example of sequential access media. In order to get a specific data, the user should roll through the tape until the desired data is located. However, for the direct access devices, the user can get a particular data directly without passing through a sequence. Best example of direct access devices is the HDD. The following section depicts the parts of the HDD.

1.4 Hard Disk Drive Components

HDD consists of many parts as depicted in Figure 1.1. The main part is the platter [43]. Platters are rotating disks made of an aluminium or glass and ceramic substrate which are held by a spindle spinning at an extremely high speed. Platters, unlike the floppy disks, cannot be flipped or bent that is why they have been called hard disks. Normally, HDD have 1 to 10 identical platters each one has one write/read head. The magnetic layer on the platter is divided into a very small room, each room stores one bit of data. The two sides of each platter are magnetized to store the information and this would increase the storage density. Also, the key component of the HDD is the head that performs writing and reading functions. Head writes the data by converting the electrical current into a magnetic field which magnetizes the storage medium into two opposite directions. To read the recorded data, the head senses the magnetic flux from the media. Head flies over the disk surface with a distance called flying height which is about 20μin [10]. The first head used was Poulsen-type head [23], where longitudinal recording was done by this head. Then using ring-type head improved the recording performance [23]. Lee, *et al.* [44] proposed a

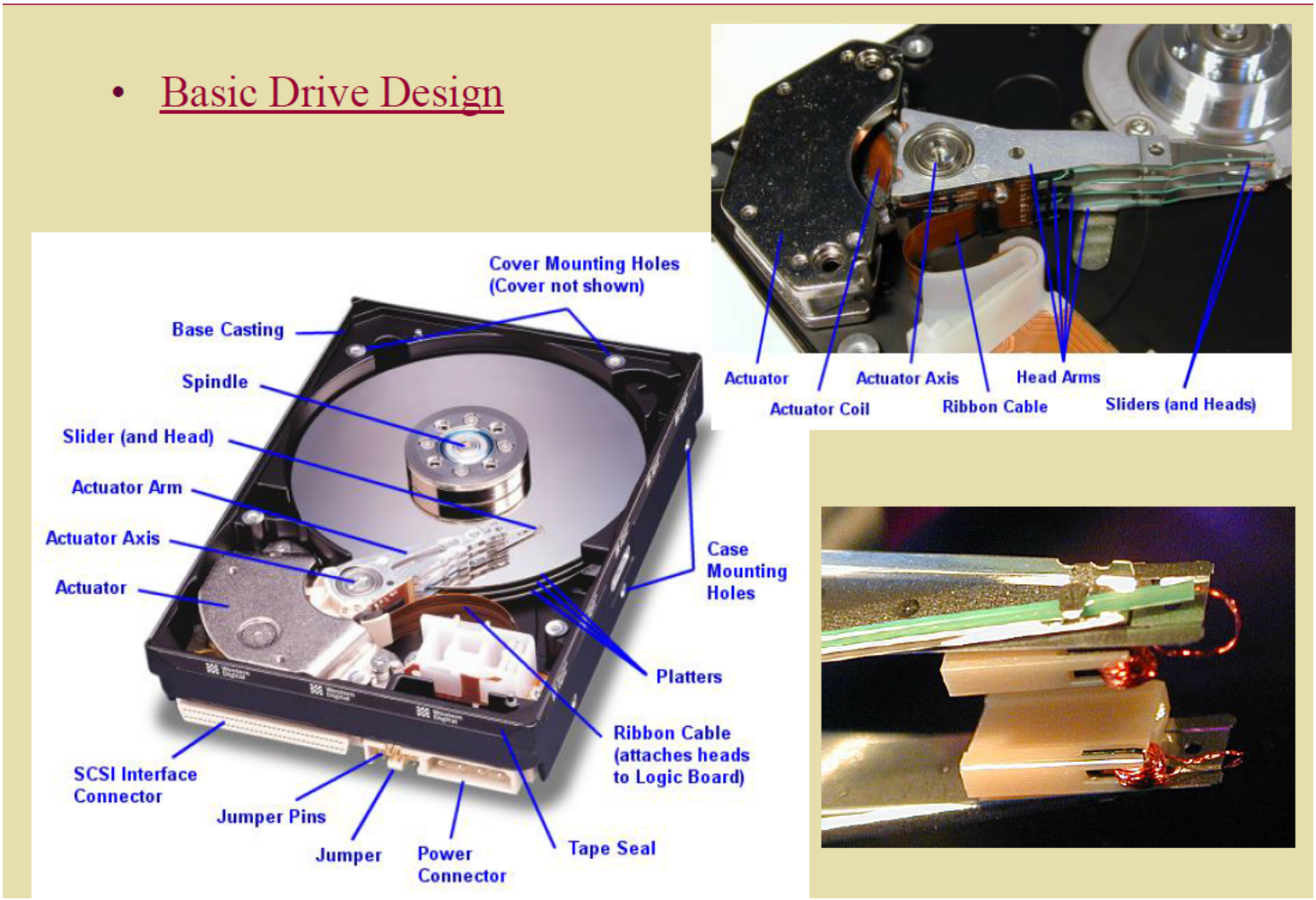

Figure 1.1: Hard Disk Drive Parts [2]

modified ring-head. The proposed channel model was for a single-layered perpendicular magnetic recording with a modified ring-head. The ring-head has been trimmed at the top pole edge to get a new head with improvements of the perpendicular head field and its gradient.

Recording data on a hard disk drive is accomplished by writing it on the recording medium. The medium would be magnetized into two opposite directions by passing a current into the write head. The two opposite directions representing either 0 or 1. The recording medium should be thin and highly coercive to allow recording of large amount

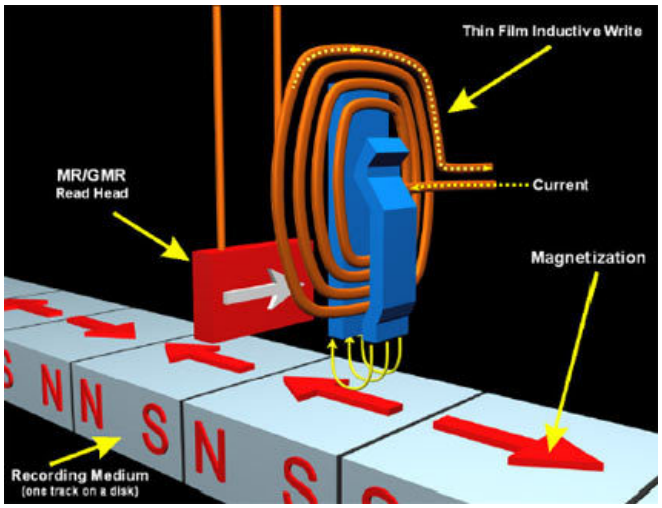

Figure 1.2: Longitudinal Magnetic Recording Technology [3]

of data onto the disk surface. Firstly, the coated r-Fe_2O_3 film of thickness less than 1μm was used. Then, composite anisotropy medium composed by double layers of Fe-Ni and Co-Cr thin films were prepared by Iwasaki, *et al.* [27] which had a significant effect in increasing the storage capacity.

1.5 Longitudinal Recording Technology

Depending on the way the recording medium is saturated, there are three types of magnetic recording which are longitudinal, perpendicular and transverse. The recording types are classified according to the direction of magnetization relative to the direction of the medium motion [22]. Longitudinal thin film recording has been the standard technique in the HDD industry for more than 50 years [45]. In Longitudinal Magnetic Recording (LMR) technique, the data bits are aligned horizontally in relation to the plane of the disk as shown in Figure 1.2. This method has reached its physical limit and the maximum storage density could be offered is 100 Gbit/in^2 by using *Ku* materials (uniaxial magnetocrystalline anisotropy) [46][47]. Therefore, the necessity of a new method was arisen to meet data

growth requirements. PMR, which was known before longitudinal recording [48], was the alternative attention for HDD industry.

In general, increasing the storage capacity needs several requirements such as:

1. Medium requirements which include thickness with high coercivity. Coercivity means the capability of a ferromagnetic material to be demagnetized. Demagnetization in the medium decreases the remanent magnetization and establishes circular magnetization mode which in turn reduces the reproduced signals [49]. Consequently, reducing the thickness of the recording medium could prevent the establishing of the circular magnetization.

2. Low thermal stability: the magnetic medium consists of grains. Increasing the storage capacity requires shrinking the grains further which leads to the thermal stability problem. That means the grains would be much taller than they are wide. Therefore, a specific volume should be available for those grains to be thermally stable [9]. Also, making the grains so small would affect the energy of the signal itself. In other words, signal energy becomes very small compared to the ambient thermal stability which in turn requires powerful signal processing tools to retrieve the recorded bits [50].

3. Head requirements like strong writing field of the inductive head.

4. Good Signal to Noise Ratio (SNR) which satisfy BER requirements. However, error rate of 10^{-12} or less is the need for storage systems [51].

5. Advanced signal processing tools to reliably retrieve the readback signal. Increasing the storage density would cause many sources of noise and distortion such as transition jitter noise, electronics noise in the head and preamp circuits, recording medium

noise [10], ISI and ITI . Consequently, appropriate mitigation techniques are required to retrieve back the recorded signal.

Iwasaki, *et al.* [49] analysed the demagnetization mechanism of the longitudinal recording in high density storage. The conclusion that came from this analysis was that the longitudinal recording is no longer the best method in terms of high storage density. An experiment of combining perpendicular anisotropy films with magnetic heads was performed by Iwasaki, *et al.* which confirmed that the perpendicular recording had a magnetization mode which increased the remanent magnetic moment of signals.

To counter the above problems, PMR technology was introduced by Iwasaki. The following section will explain this technology (PMR) and then shows the limitations and alternatives for it.

1.6 Perpendicular Recording Technology

It is mentioned that the first HDD using PMR technology was shipped in 2005 which boosted the capacity to 80 GB [34]. The longitudinal recording method was replaced by perpendicular recording because it reached its physical limits due to the superparamagnetic effects [48][31] [52]. The superparamagnetic effect is a trade-off between three parameters: the thermal stability of the media, the write ability of the media with a narrow track head and the media SNR [8]. However, perpendicular and longitudinal recording methods have complementary relationship even in the aspect of engineering. The perpendicular recording magnetization mode is suitable for recording of the digital signal while for the analog signal, the longitudinal recording magnetization mode is more suited [27]. In this technique, the data bits are aligned perpendicularly in relation to the plane of the disk as shown in Figure 1.3 which allows picking up more data on a disk surface compared with the longitudinal magnetic recording [53].

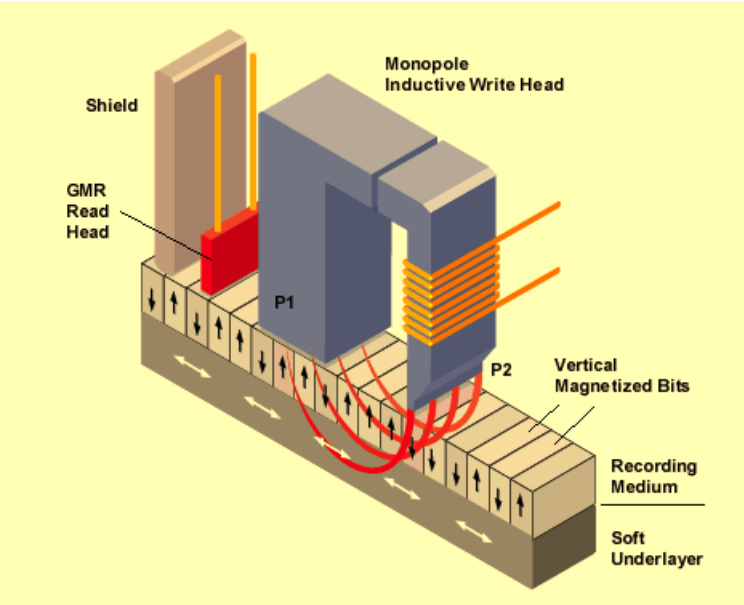

Figure 1.3: Perpendicular Magnetic Recording Technology [4]

1.6.1 PMR Advantages Over LMR

PMR have several advantages over LMR such as:

- Perpendicular recording system uses single-pole type write head which gives twice the field that a ring head produces [32][50]. Higher write field would increase the write efficiency [10] and subsequently increase the storage density.
- In PMR, the bits are thermally more stable than in LMR as the adjacent bits are placed side by side rather than end-to-end which allow attraction the bits instead of rejection.
- Adding a Soft Magnetic Underlayer (SUL) beneath the recording layer (PMR includes two groups with and without SUL [10]) offers a larger effective write field which has a significant effect on increasing the data storage [53].
- Both technologies use the same read head which is Giant Magneto-Resistance (GMR)

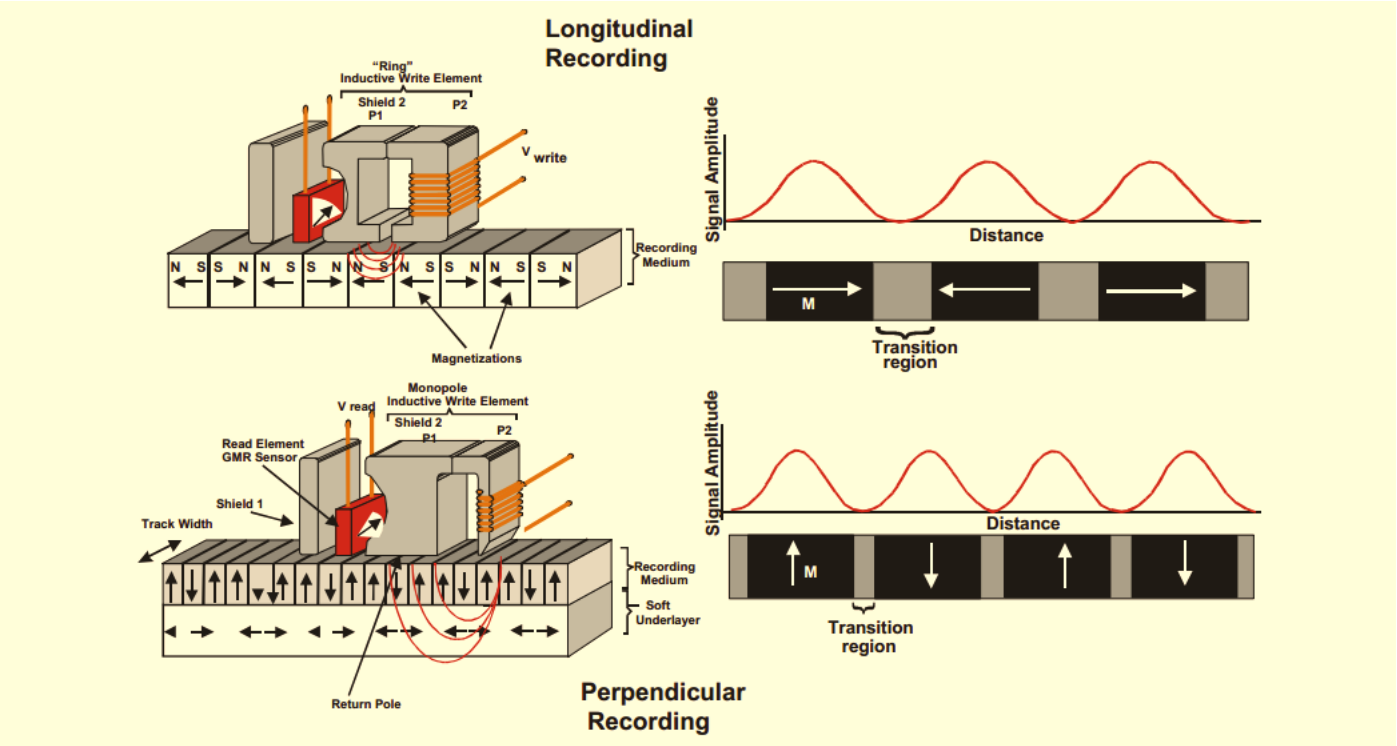

Figure 1.4: Main Differences between LMR and PMR [5]

playback sensor.

The main differences between the two technologies are depicted in Figure 1.4. All these benefits make PMR more interested than LMR and one of the assuring magnetic recording systems for the HDD.

Using PMR, two ways could help to increase the storage capacity of HDD which are either increasing the areal density of the platter or increasing the number of platters.

Areal density is defined as the number of bits per inch per track times tracks per inch [45] which is given in bits/mm^2, Mbit/in^2, Gbit/in^2 and Tbit/in^2 [54].

However, PMR technology has some limitations against increasing the storage capacity such as the instability of the medium in case make magnetised areas are very small. Also, because of the other technologies like: Bit-Patterned Media (BPM) , HAMR and Microwave Assisted Magnetic Recording (MAMR) require the medium to be redesigned

which in turn require high expenses. Thus, the direction now is towards adding platers to increase the storage capacity rather than cramming more bits on the same platter [55].

Wood [50] suggested that achieving 1 Tbit/in^2 is feasible for PMR method. The fundamental obstacle to go beyond this achievement for this technology is the superparamagnetic effect [50].

Recently, some HDD industry projects have started which aim at pushing the storage density beyond the superparamagnetic limit for example: in 2010, the Storage Research Consortium project aimed at 2 Tbit/in^2, in 2013 the New Energy and Industrial Technology Development Organization target at 5 Tbit/in^2 and 10 Tbit/in^2 by 2015 for Information Storage Industry Consortium [8].

1.7 Future Technology Options

Some approaches have been considered as an alternative to the conventional technique (PMR) such as BPM, HAMR/MAMR and SMR . BPM looks a promising technique to continue beyond the 1 Tbit/in^2 and eliminates the limitations of PMR that were mentioned in section 1.6. In this technology, the magnetic layer (medium) must be radically redesigned i.e. continuous thin film should be replaced with a discrete magnetic material [45]. However, this approach faces some challenges which make it difficult to implement. The first challenge is manufacturing the medium and engineering it into islands [56] as shown in Figure 1.5 [45]. BPM suggests storing one bit in one magnetic island [8] unlike PMR where each bit must cover roughly 100 grains due to the randomness of the grain sizes and shapes [57]. The other considerable challenge is the write-process must be synchronized carefully with the locations of the medium islands i.e. each single-domain magnetic island should store one bit.

In HAMR the significant component is the recording head [8]. The writing area is mag-

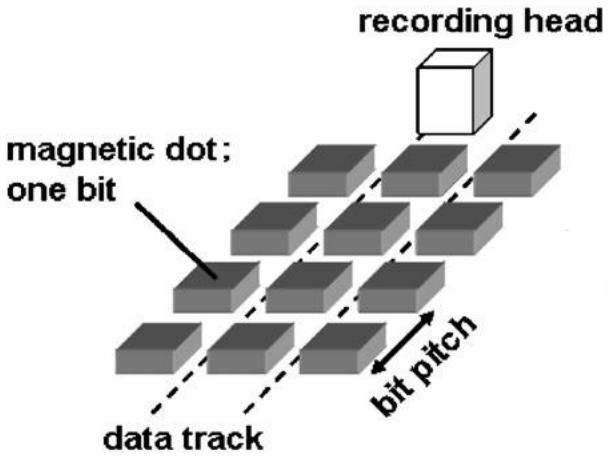

Figure 1.5: Schematic View of the Bit-Patterned Media [6]

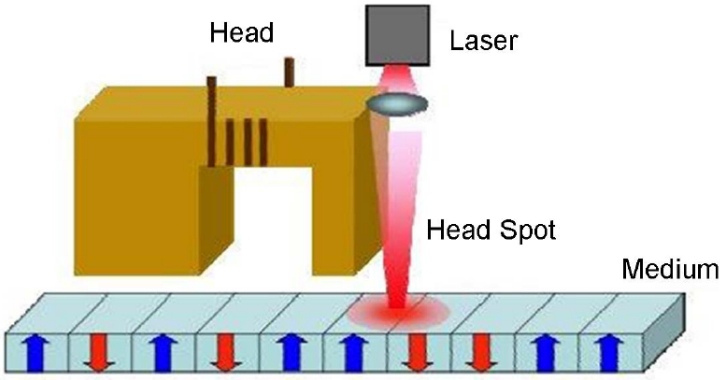

Figure 1.6: Schematic View of HAMR Technology [7]

netized by a laser embedded in the write head as shown in Figure 1.6, which heats the medium then cooling it back to an ambient temperature to get the required magnetization. Figure 1.7 illustrates schematically the three future techniques options.

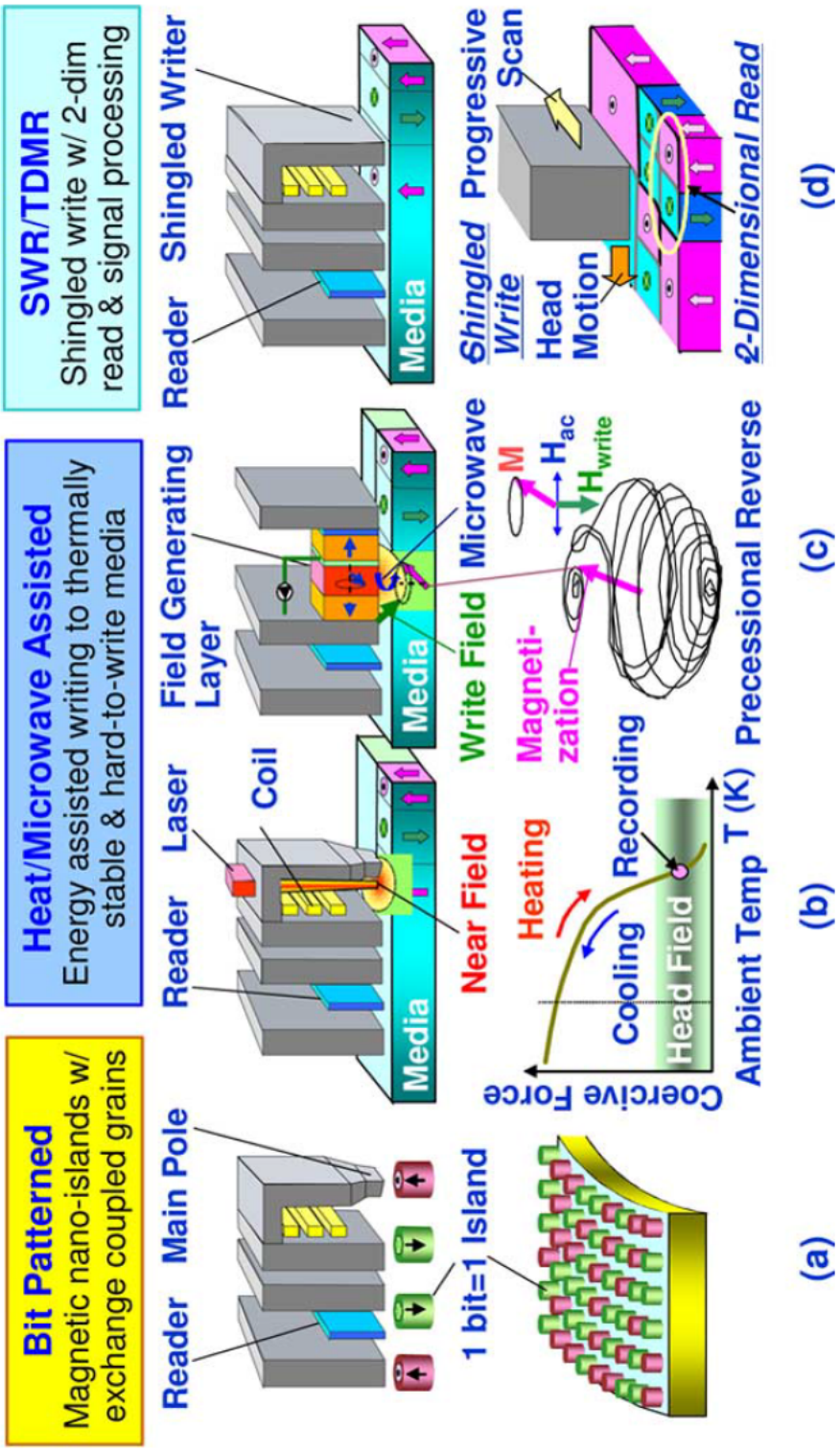

Figure 1.7: Future Techniques Options for HDD Storage [8]

1.8 Shingled Writing Technology

The current magnetic recording technology, PMR, has reached a limitation which is imposed by superparamagnetic effect, the media trilemma [56]. While the theoretical limit of this technology is 1 Tbit/in^2, the current drives capacity is 400 Gbit/in^2 [56] [58]. However, the highest demonstrated areal density that could be achieved with continuous perpendicular medium is 612 Gbit/in^2 [8].

As a result, demands for a new recording technology have arisen to cover the 30-50% annual growth [50][56] [8] of the data density. SMR is the more powerful candidate among different technologies to replace the conventional magnetic recording. SMR is easy to implement as it can be built on the existing technology so that there is no need to change the media of PMR method. However, a few changes are required on the write head because the writing needs much stronger magnetic field compared to the reading process. The fundamental concept of SMR is overlapping of tracks, like shingling (laying) tiles on a roof as shown in Figure 1.8. The stronger and asymmetric magnetic field offered by the write head of shingled system would help making the overlapping possible. The shingled-writing is like overlapping of newly written tracks with preceding tracks leaving only a small strip of the previous track as a record of the original data. Placing tracks closer together would allow more data to be recorded. This small strip of the written track could be read by the current GMR head. Nevertheless, there are still some disadvantages in this technology such as the writing should be done sequentially. But this technology still offers random access readout where the readback process is unaffected [56][8]. In addition, a considerable disadvantage which is ”update-in-place” is no longer possible [9]. In other words, rewriting a single track or a portion of it would require recovering the information written on the subsequent adjacent tracks.

Another technology has been proposed to extend the existing heads and media to higher

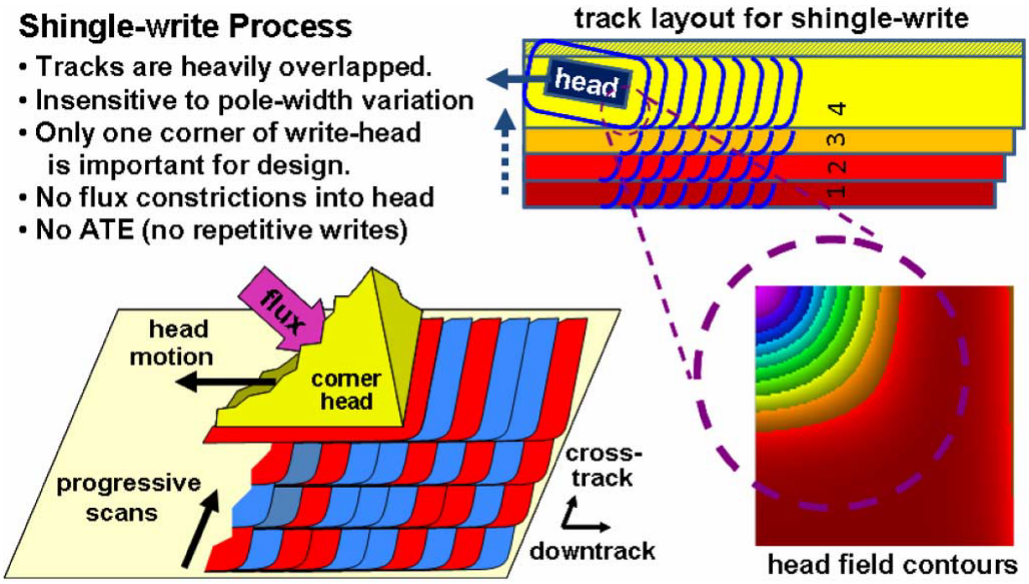

Figure 1.8: Shingled Magnetic Recording Technique Process [9]

storage density, namely Two-Dimensional Magnetic Recording (TDMR). TDMR combines two techniques: SMR and Two-Dimensional (2-D) readback which requires a powerful signal processing. However, TDMR could be used alone or in conjunction with SMR [8].

1.9 Aim and Objectives

Many areas are recognized as the key components that play a significant role in disk drive capacity such as recording technique, materials, manufacturing and signal processing tools. The improvements in head, media, mechanical designs and advances in signal processing techniques could achieve high recording density [59]. Also, using magnetic recording system that record each bit of data on very few magnetic grains together with signal processing system that have the capability to recover the data reliably would achieve the highest areal densities [50]. Although the current disk drives are approaching 400 Gb/in^2, Wood, *et al.* [50] predicated that the limit of the conventional magnetic recording is around 1 Tbit/in^2.

The main obstacle of the growth of HDD capacity is the thermal stability (superparamagnetic effect) of the magnetic grains that comprise the magnetic material [50]. The superparamagnetic effect causes erasing of the recorded data after a short period of time [10]. Increasing the storage density requires shrinking the grains size which in turn decreases the magnetic anisotropy energy of the grains (KuV, where Ku is the anisotropy of the grain and V is its volume) to become less than K_BT (where K_B is Boltzmann's constant and T is the absolute temperature) then the magnetization will be switched [45][60].

In addition, the other obstacle that prevents data storage growth is the ITI. The rapid growth of digital data requires higher storage density which in turn requires either reducing track widths, or reducing the space between tracks, or a combination of both. Perpendicular recording technology stores the data in concentric circular tracks which are reasonably isolated with guard bands [61]. Shingled systems can increase the storage capacity by reducing those guard bands. In shingled magnetic recording, the data can be written sequentially using a wide write head with tracks squeezing (no guard bands)[62] [63]. However, this squeezing causes tracks to overlap one another which in turn increases ITI that will lead to performance degradation.

The aim of the research is investigating the ITI and reducing the complexity of the shingled system decoder. To achieve this aim, the work is carried out by the following objectives:

1. Investigation and implementation of PRML channel model as a platform in order to accomplish the data recovery process where this process indicates the performance of the magnetic recording system. PRML channel implementation is developed as follows:

    - Designing and simulation of PMR channel.

- Implementation of Partial Response (PR) equaliser to reshape the PMR channel response into a specific target response.
- Investigation and implementation of VA as an optimal sequence decoder to recover the written data. Then this algorithm is modified to reduce BER which would improve the performance of the PRML channel.
- Investigation and implementation of Maximum A Posteriori (MAP) algorithm as an optimum bit based decoder. This algorithm is also modified for the same channel and a comparison is made between the two algorithms.

2. Investigation of shingled system as an alternative technique of perpendicular recording method to increase the disk density. Shingled technology can be built on the same media of perpendicular recording with few changes to the write head.

3. Implementation of shingled system by extending the PRML system model of Step 1 into a two-track model to include the ITI of the adjacent track. Implementation of the two-track system model requires extension of the channel trellis of one-track system model. Consequently, a complicated 16-state trellis with four incoming paths and four outgoing paths at each state would be constructed which will make the complexity of the new system decoder trellis very high. Therefore, an intensive work is required in order to reduce the complexity of this trellis.

4. Implementation of one-track system model in hardware to measure its complexity.

1.10 Contributions to Knowledge

This thesis covered many aspects of decoder design for perpendicular recording and shingled-writing systems. The original contributions to the knowledge of the author are listed below:

1. Two decoding techniques used for perpendicular magnetic recording which are the optimal sequence decoder (VA) and the optimal bit-based decoder (MAP). The algorithms are modified based on the pdfs histogram of the output samples of the ideal channel (target). The derivation of maximum likelihood sequence for ML decoder is given considering the pdfs distribution of the ideal channel samples. From the derivation, it is found that the new maximum likelihood sequence involves additional term which will require adding one sum in the path metric computations of the channel trellis. VA is modified by adding the new term to the path metric computations. MAP is modified by adding the new term to the transition probabilities computations.

2. Investigation of the ITI in shingled magnetic recording systems requires construction two-track one head system model. The ITI can be considered as either a noise added to the AWGN of the system or a signal which would increase the SNR.

   If the ITI is treated as a noise, the BER exceeds 10^{-1} at 6 dB SNR. However, if the system has no ITI, the BER is almost 10^{-3} for the same SNR.

   If the equaliser views the ITI as a signal, then 16-state trellis with four outgoing paths from every state and four incoming paths to every trellis state is required to accomplish all the possibilities of the channel. MLSD is applied to find the most likely path sequence in the trellis of the shingled system. Consequently, different decoding techniques have been found. The decoding techniques show different performances with different complexities. The conventional technique (without simplification) shows an optimal performance with high complexity. However, the one common factor technique has the closest performance to the optimal one where the difference between the two performances becomes larger at higher amount of ITI (more than 30%).

In terms of complexity, the one common factor approximation has the lower complexity. However, the complexity could be reduced further if some terms of the approximation are completed off-line. Consequently, the complexity is reduced more than three times compared with the complexity of the conventional technique. The simulation results showed that a trade-off can be made between the BER and the complexity.

The author has presented this contribution at:

- ***Intermag 2012 Conference***.

- ***Joint MMM/Intermag 2013 Conference***.

3. In order to measure the complexity of one-track one head system model, PRML channel is verified in hardware using Altera Field-Programmable Gate Array (FPGA) development board. PRML system is divided into four components for implementation purpose in Simulink/DSP Builder softwares which are: PMR channel, PR equaliser, ideal channel (target) and VA components. These components have been verified in hardware expect VA architecture. Consequently, a conclusion has been made which is Simulink/DSP Builder softwares are ideal for low complexity designs. However for complex designs, HDL code is the eligible solution.

   The author has published this work at 3rd International Symposium on Advanced Magnetic Materials and Applications 2013.

1.11 Thesis Structure

This thesis is organized in the following way:

Chapter 2 presents a general knowledge of PRML system in magnetic recording systems which include equalisation and detection algorithms. The aim of this chapter is to provide a background information about signal processing which will help simulating the PRML

system to perform the data recovery process.

Chapter 3 discusses PRML system implementation to evaluate the system performance in terms of BER graphs. Also in this chapter, two optimum detection algorithms will be modified in order to improve the PRML system performance.

Chapter 4 investigates SMR system and discusses adding ITI amount into the one-track model that described in Chapter 2 and then extend it into two-track system. New trellis, 16-state, with four incoming paths and four outgoing paths at each state is constructed. Six novel decoding techniques have been found with very high complexity. Intensive work has been done in order measure and then reduce their complexities. One technique (one common factor) had a better performance which was very close to the optimal one. The complexity of this technique has been measured and it is found that some computations can be completed off-line while the others can be completed on-line. Off-line means while the equaliser output sequence is being equalised, the computations that do not depend on the equaliser output sequence can be computed. Consequently, the complexity of one common factor method has been reduced further. The simulation results showed that a trade-off can be made between the BER and the complexity.

Chapter 5 provides an introduction to the simulation softwares that will be used to implement PRML system in hardware. Those softwares are Simulink of Math-Works, ModelSim and Altera Quartus II softwares. Also, it explains some features of STM32F4 High-Performance Discovery board and Altera DE1 development and education board. The two boards will be used to verify the hardware design of PR equaliser model.

Chapter 6 discusses many hardware architectures of PRML system including PMR

channel, PR equaliser without and with AWGN, ideal channel (target) and 4-state trellis structure. Those architectures are implemented in Simulink/DSP Builder softwares and then in Quartus II. A conclusion has been made which is designing using Simulink/DSP Builder softwares is ideal for the designs with low complexity however for the complicated designs the alternative solution is using HDL.

Chapter 7 the work presented in this thesis is concluded. Where this chapter is summarised the main contributions of this work and gives some suggestions as a future work to improve and develop the research in this area.

Chapter 2

Overview of PRML Read Systems

2.1 Introduction

Generally, PRML systems have been used in communication channels since 1960s and then in magnetic recording channels in 1970 as a technique to combat the ISI effect. In the PRML system, Viterbi detector which is an ML detector, is applied to convert the received sequence into a corresponding sequence of detected bits. MAP algorithm is also applied to the data storage channels as an optimum detector based on bit detection rather than sequence detection.

The most important part of any data storage system is the process of recovering the written data that is transmitted through the magnetic recording systems. Therefore, this chapter will present in detail the process of recovering and will review every part of the PRML system which includes the two detection algorithms.

2.2 Data Storage Channel Model

The similarity between digital communication system and data storage system is in the main goal, namely the reliability in receiving the transmitted information (data). In both systems, information is transmitted through noisy channels and then recovered by using efficient means. The difference between the two scenarios is that the information in communication systems is transported from one location to another at the same time. Whereas in the data storage system, transmitting the information occurs from one time to another at

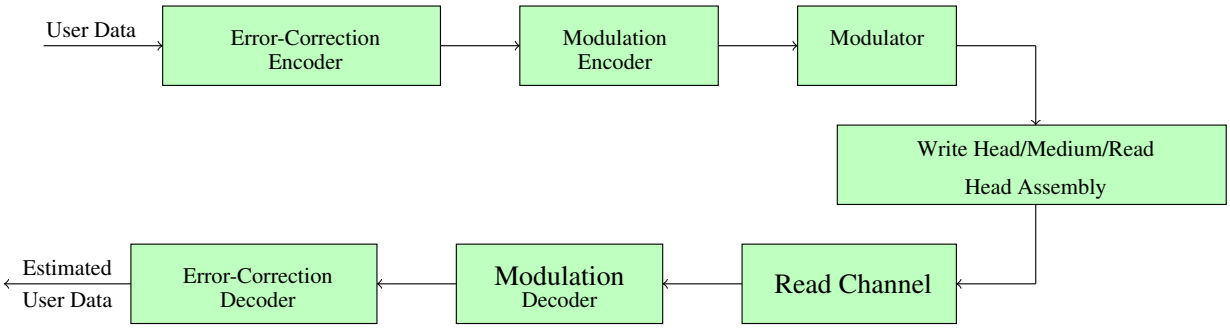

Figure 2.1: Data Storage System Block Diagram [10][11]

same location [51][64].

Figure 2.1 depicts data storage system model. The information bits are encoded and protected by error correction codes. Modulation coding is applied to control the minimum and maximum distances between consecutive magnetic transitions. In other words, using modulation coding would help eliminate the arbitrary sequence of user data from a recorded stream. The minimum distance constraint is used to avoid media noise and non-linearity associated with the crowded magnetic transitions. The maximum distance constraint is applied to ensure the existence of the signal at readback. The constraints of RLL(d,k) code are the most known constraints for data storage channel where d is the distance that separates 1's and k for 0's. The modulated data sequence is converted into a rectangular current waveform and then stored in the medium as a magnetization form. The read waveform is a linear superposition of shifted transition responses where every single written transition is a transition response [10].

2.2.1 Longitudinal Magnetic Recording Channel Model

Figure 2.2 shows the simulated LMR channel model. The $(1-D)$ block is inherent with the longitudinal magnetic channel due to the differentiation property in the readback process [65] produced by the magnetic read head. Magnetic recording channel can be considered

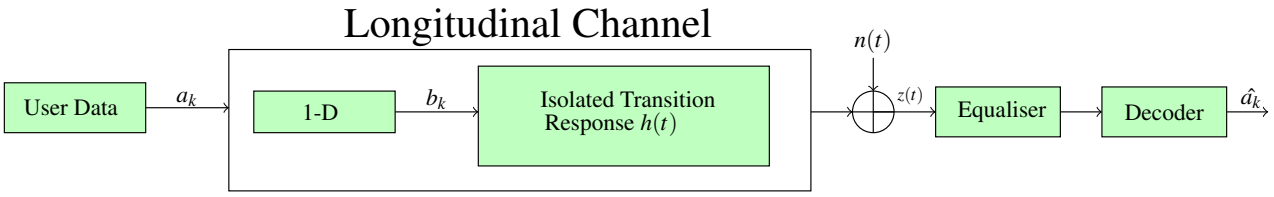

Figure 2.2: Simulated Longitudinal Recording Channel Model

as a linear time-invariant system [66] [54]. As a result, the read waveform of longitudinal channel is [10]

$$z(t) = \sum_k (a_k - a_{k-1}) h(t - kT) + n(t) \tag{2.1}$$

Where a_k is the user data which is a sequence of 1's and 0's, $b_k = a_k - a_{k-1}$, $h(t)$ is the isolated transition response, T is the bit interval and $n(t)$ is AWGN due to the read head and electronics [10].

In LMR, the MR read head generates an output voltage from a magnetic transition. Consequently, each magnetic transition produces an opposite output voltage of the previous transition and zero output voltage for no magnetic transitions regions. Therefore LMR readback signal is a dc-free signal. Contrary to PMR, the output voltage will be obtained from the regions of constant magnetic polarity while at the magnetic transition region the MR read head constructs zero output voltage. Thus, PMR readback signal is a non-zero dc-response [67].

The reading head impulse response, $h(t)$, is governed with the shape of magnetization and it is modeled as a Lorentzian pulse [10]. As it was mentioned in Chapter 1, the writing process is accomplished by the write head. It records the data on the disk surface by magnetizing the storage media, that moves below the head, into two opposite directions representing either bit 0 or bit 1. Subsequently, the read head reads the recorded data by sensing the change of the magnetic field arising from the disk. The read out signal could be characterized by the transition responses $h(t)$ and $(-h(t))$. Here, $h(t)$ indicates a positive

transition from bit 0 to bit 1 and $(-h(t))$ means the transition is from bit 1 to bit 0.

The Lorentzian pulse of the longitudinal recording transition response is expressed as [10]

$$h(t) = \left(\sqrt{\frac{4E_t}{\pi PW_{50}}}\right)\left(\frac{1}{1+(2t/PW_{50})^2}\right) \tag{2.2}$$

Where $E_t = \int \mid h(t) \mid^2 dt$ is the amount of energy in the transition response and PW_{50} is the width of Lorentzian pulse, $h(t)$, at half of its peak value. PW_{50} is related to the head and the media and not related to the operating normalized density therefore, PW_{50} is fixed for a fixed head and media combination [10]. The term $\sqrt{\frac{4E_t}{\pi PW_{50}}}$ can be set to 1 as a normalization factor. Thus, transition response could be reduced to:

$$h(t) = \frac{1}{1+(2t/PW_{50})^2} \tag{2.3}$$

The response of two adjacent transitions, dibit, can be expressed as [22][54]

$$d(t) = \frac{1}{1+(\frac{2t}{PW_{50}})^2} - \frac{1}{1+(\frac{2(t-1)}{PW_{50}})^2} \tag{2.4}$$

The longitudinal channel has been simulated using MATLAB written by the author in order to find the impulse response of this channel as follows [10]

$$x_i = \frac{1}{1+(2i/(D_s \cdot osr))^2} \tag{2.5}$$

Where $i = 0, 1, 2, ...$, x_i is the i-th sample in the oversampling domain, $D_s = 0.1$ and osr =65. $D_s = PW_{50}/T$ is the normalised density where PW_{50} is normalised to the bit period, T, and osr is the number of samples.

Increasing the storage density requires increasing the recorded bits per the interval $w, w =$

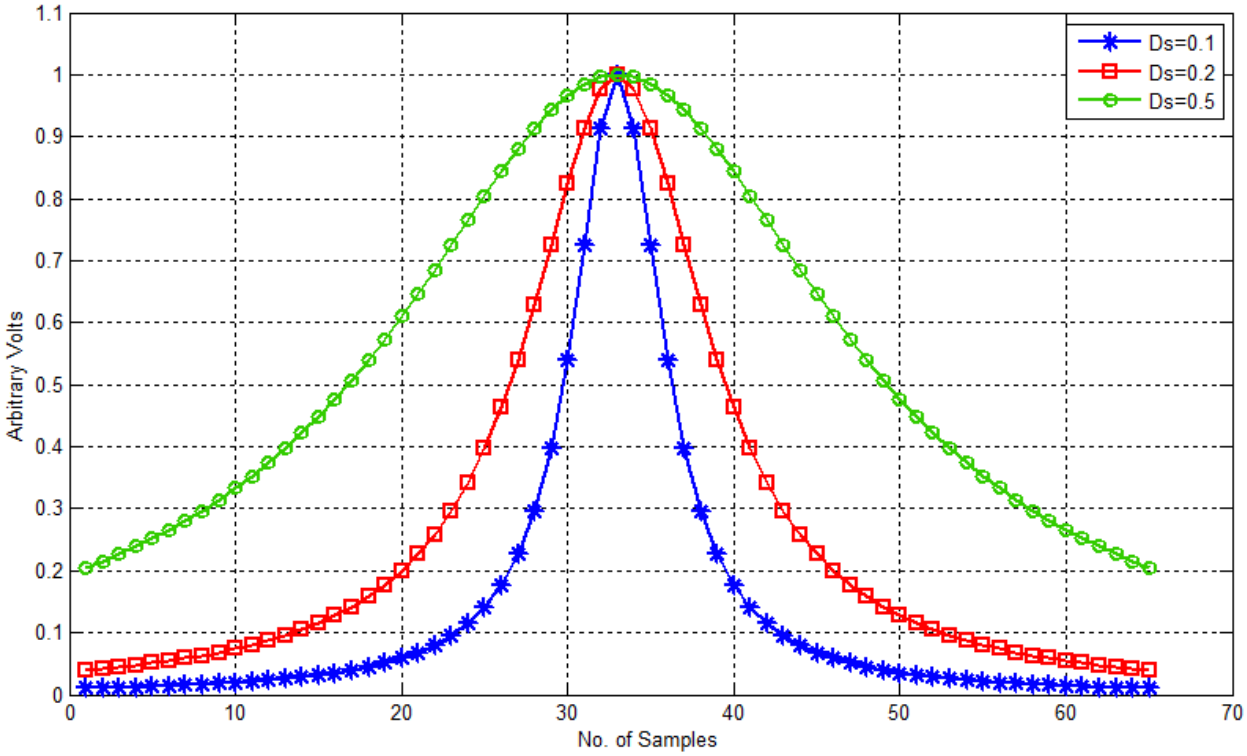

Figure 2.3: Sampled Transition Response of Longitudinal Recording at Different Linear Densities

PW_{50}, which in turn causes the overlapping and then more ISI as shown in Figure 2.3.

The communication channels, including data storage channels, are characterized as band-limited filters. Accordingly, such channels are described by their frequency response $H(f)$, which is [10]

$$H(f) = A(f)e^{j\theta(f)} \tag{2.6}$$

Where $A(f)$ is the amplitude response and $\theta(f)$ is the phase response. Sometimes group delay or envelope delay is used instead of phase response, that is [10]

$$D(f) = -\frac{1}{2\pi}\frac{d\theta(f)}{df} \tag{2.7}$$

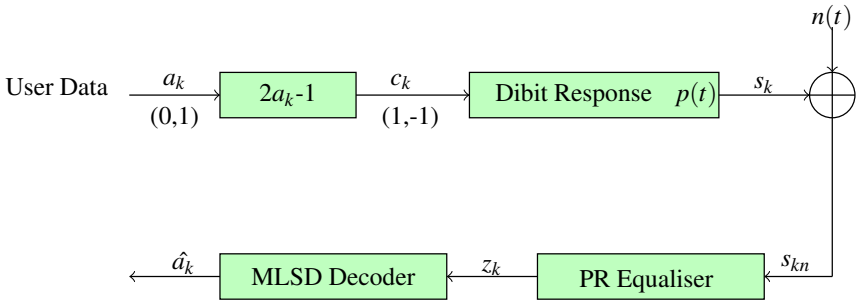

Figure 2.4: Simulated Perpendicular Recording Channel Model

The frequency response of Equation (2.4) is [22][54]

$$H(w) = (1 - e^{-jwT})\frac{PW_{50}}{2}\pi e^{-w\frac{PW_{50}}{2}} \tag{2.8}$$

Where w is the frequency term and T is the sampling period.

2.2.2 Perpendicular Magnetic Recording Channel Model

Figure 2.4 depicts the simulated PMR system model. The implementation of the PMR channel in MATLAB will be discussed in Chapter 3. The readback signal can be described by [10]

$$z(t) = \sum_k c_k p(t - kT) + n(t) \tag{2.9}$$

Where, $p(t - kT) = h(t - kT) - h(t - (k-1)T)$. For each single written transition, the transition response is modelled as [10]

$$h(t) = V_p.erf\left((2\sqrt{ln2})\frac{t}{PW_{50}}\right) \tag{2.10}$$

where $erf(x)$ is the error function, stated as:

$$erf(t) = \frac{2}{\sqrt{\pi}} \int_0^t e^{-x^2} dx \tag{2.11}$$

V_p is the peak voltage value of the transition response and PW_{50} is the time taken for $h(t)$ to go from $-V_p/2$ to $+V_p/2$.

For simulation purposes, the transition response can be approximated by a hyperbolic tangent function, *tanh* [52]

$$h(t) = V_p \tanh\left(\ln(3)\frac{t}{D_s}\right) \tag{2.12}$$

Figure 2.5 shows the sampled transition response for perpendicular recording. The dibit response, $p(t)$, is:

$$p(t) = h(t) - h(t-T) \tag{2.13}$$

Increasing the normalised density would decrease the amplitude of the dibit response which in turn reduces the power of the readback signal [10] as shown in Figure 2.6.

2.3 PRML Method

Peak detection and RLL coding have been used for decades as a reliable technique with 100 000 bits/mm^2 storage density [59]. The concept of the peak detector, symbol-by-symbol detector, is based on detecting the isolated peak of each transition written on the magnetic media. The detector works effectively with low recording density where the pulses are obviously separated and each transition written results in a relatively isolated voltage peak [10][12][42]. At high recording density, the problem of ISI arises and the pulses overlap each other which would effect the capability of the peak detector to recover the recorded data reliably. Consequently, high density storage requires more sophisticated detection methods in order to mitigate the ISI effects. Therefore, it was necessary to find another

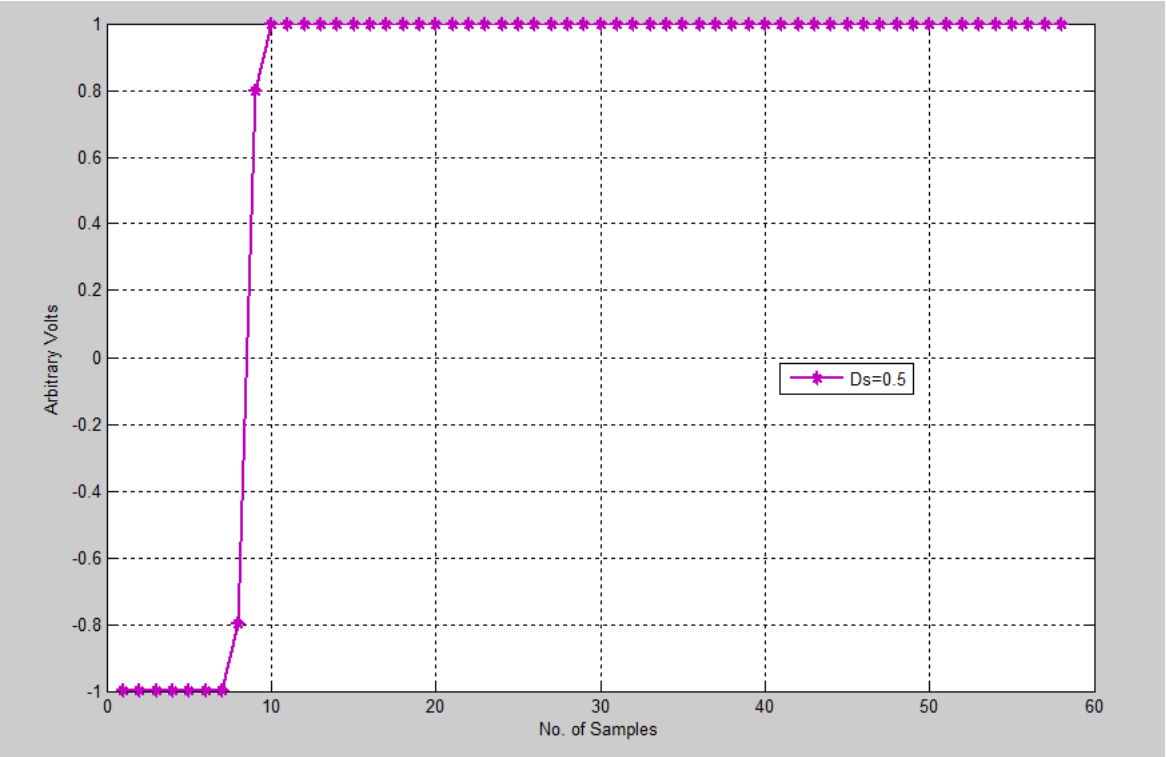

Figure 2.5: Sampled Transition Response of Perpendicular Magnetic Recording System

technique that would overcome the peak detector degradation when the amount of ISI is considerably large. For that reason, the PRML method was proposed and then became the dominant detection scheme in commercial HDDs.

PRML method is used as a signal processing technique for high density longitudinal and perpendicular recording systems. The digital cassette tape recording device was the first storage product which used PRML and was introduced by Ampex in 1984. It was reported in [68] that, using the PRML method on an experimental disc recording system achieved 56 Mb/in^2. The first HDD -IBM 0681 was shipped in 1990, used PRML technique [69]. At the beginning of 1990's, most of the drives industry had been applying the peak detection method with an exception of a few IBM drives using PRML technique and later on the entire industry was replaced with the PRML method [10].

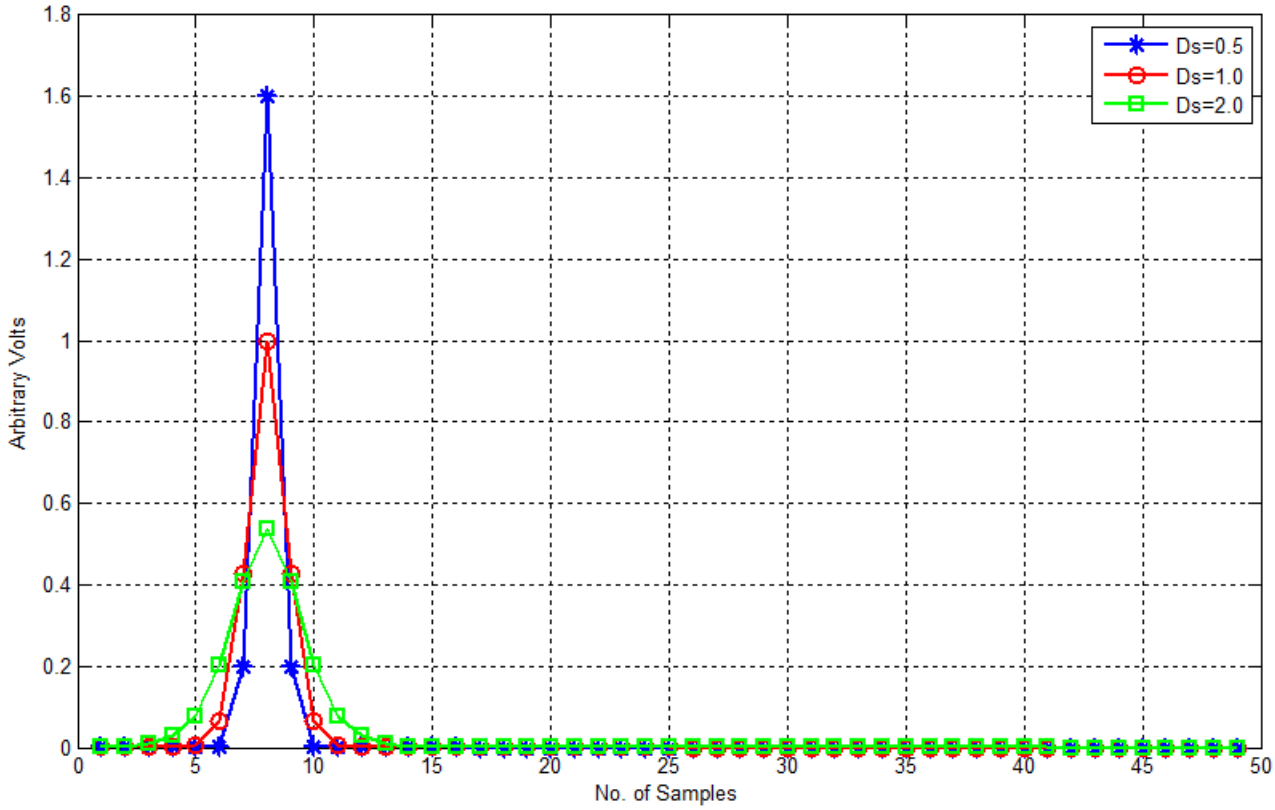

Figure 2.6: Sampled Dibit Response of Perpendicular Recording at Different Linear Densities

Moreover, in 1970 the theoretical studies done by Kobayashi and Tang at IBM suggested that the maximum likelihood sampling detector and PR equalisation was possible to apply to the magnetic recording channels [65]. Also, Cideciyan, *et al* in [59] have showed that applying PR signalling in combination with Maximum-Likelihood Sequence Detection (MLSD) allows further progress in increasing the storage density and recovering the recorded data reliably over the conventional technique (RLL coding and peak detection method). This work [59] also proved that the PRML technique is suitable for digital magnetic recording applications.

The frequency responses of magnetic recording systems that use inductive or MR head (for

longitudinal recording) have similar characteristics, both have a null at DC. Also, the frequency responses decrease exponentially with frequency at high frequencies. Therefore, magnetic recording channels are considered as band-pass channels [59]. A question arises if the sampling frequency in the PRML system is high enough to recover the recorded transition. The Nyquist sampling theorem, states a principle of analog signals digitization which is the sampling frequency should be at least twice the highest frequency contained in the signal. For instance, if a channel bandwidth is WHz then the maximum binary transmission rate with zero ISI is $1/T = 2W$ symbols/sec [10]. Therefore, sampling once per channel period T is appropriate only if the spectrum of the analog readback signal is concentrated below the frequency $f_{max} = 1/2T$ [12].

Figure 2.7 shows the block diagram of a typical PRML channel. Variable-Gain Amplifier (VGA) , analog equaliser, Analog-to-Digital Converter (ADC) and digital equaliser blocks transform the readback signal into a partial response signal. The magnetic read head produces an analog readback signal, with different levels of amplification which in turn leads to get different peaks of the readback signal. VGA compensates that variation in the signal peaks and amplifies them to have a certain and constant level of amplification. A clock signal is used to sample the analog waveform to obtain the discrete samples. The clock and gain recovery blocks provide a control signal to the VGA. The analog equaliser (Continuous Time Filter, CTF) block which has a specified frequency response and specific bandwidth, would determine the spectral components of the readback signal by filtering out the unwanted components that are beyond the specified bandwidth. The reason for that is to satisfy the Nyquist theorem. The analog equaliser performs another function which is modifying the frequency response of the channel. Modification is necessary to adjust the shape of the readback signal from the magnetic head. Then the analog equaliser output signal is sampled with the ADC and the sampling process is initiated by a clock signal. The

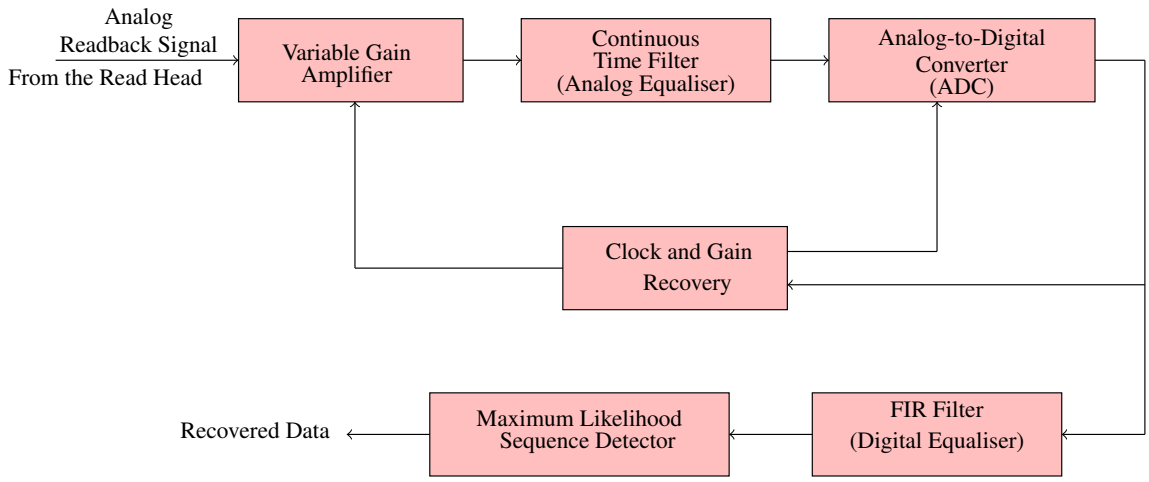

Figure 2.7: Block Diagram of Typical PRML Channel [12]

rate of the clock signal is one sample per channel bit period where this rate is adjusted by the clock recovery loop. Now, the signal at the ADC output is a stream of digital samples or discrete-time data. The possibilities of ADC are between 0 and $2^n - 1$, where n is the number of bits in ADC. Another digital filter is applied to remove the unwanted samples which in turn improve the quality of analog equalisation that performed by the CTF [12]. The purpose of this process is to reshape the partial response channel into a predefined response and then apply it to the ML detector to recover the original data [16]. Depending on the desired time and frequency response of the signal, different classes of PR targets for equalisation exist and this will be explained in the following section.

2.3.1 Partial Response Equalisation

Instead of eliminating the ISI, PR equalisation controls the amount of ISI that enters the detector [10]. The equalisation process is regarded as a ”partial response” equalisation rather than ”full response” equalisation because the later means removing the whole ISI [66]. The basic idea of the PR equaliser is leaving a controlled amount of ISI to the detector to combat [42]. This amount is allowed in order to match the PR signal spectrum to the

recording channel characteristics [59]. However, Kobayashi was the first one who noted applying the PR signals in magnetic recording systems in order to increase the storage density [70]. The magnetic recording channel could be transformed into a PR channel as long as it is satisfied two fundamental features; the superposition of voltage pulses from adjacent transitions is linear and the shape of the readback signal from an isolated transition is exactly known and determined [12]. The frequency spectrum of a linear magnetic channel is usually defined as the Fourier transform of its pulse response [12][45]. An experimental spectrum has been obtained from a channel which had a random pattern of transmitted data. The pattern had an equal number of positive and negative transitions therefore no spectrum content at zero frequency (DC content is zero) but the readback signal has high frequency components. Since that pattern contains all spectral components with frequencies of $1/2nT$, where $n = 1, 2, 3, ...$, this spectrum should have the Fourier transform of the single (isolated) pulse [12]. Also, the spectrum is concentrated below $1/2T$, though the tail of it was outside the $1/2T$ range. According to that the PRML channel is appropriate for magnetic recording systems as long as the frequency spectrum can be readily equalised by CTF with a cutoff frequency of $1/2T$ [12][45].

Kretzmer defined the classes of PR in [71] where the first widely used PR channel is Class IV Partial Response (PR4) channel [12]. The performance of PR4 is matched to the spectrum of a typical readback pulse over a range of recording densities [70]. In PR4 system, each voltage has two samples i.e. if a transition of magnetization occurs it gives a "1" at the transition location and another sample at the next sample period. This can be described by the " spreading" operator $(1+D)$. Thus, the PR4 can be described as $(1-D)(1+D) = 1-D^2$ polynomial. Table 2.1 summarizes the PR channels where in $1990's$, the PR4 channel became the darling of the magnetic data storage industry [12].

However, different PR schemes can be described by the following polynomial [71]:

Table 2.1: Partial-Response Channels

Name	Polynomial	Impulse Response Samples
PR1	1+D	...0 1 1 0...
PR2	$(1+D)^2$= $1+2D+D^2$	...0 1 2 1 0...
PR3	(1+D)(2-D)= $2+D-D^2$	...0 2 1 -1 0 ...
PR4	(1+D)(1-D)=$1-D^2$	...0 1 0 -1 0 ...
PR5	$-(1+D)^2(1-D)^2=-1+2D^2-D^4$	...0 -1 0 2 0 -1 0...

$$F(D) = (1-D)(1+D)^n, n = 1,2,3,... \tag{2.14}$$

where D is the delay operator, n is a positive integer that controls the amount of ISI [10] and $(1-D)$ is the differentiating operator. The $(1+D)$ operator represents the separation of the transition sample over the neighboring bit periods. By ignoring the differentiating operator, $(1-D)$, and looking at the second part of the above polynomial, samples of the isolated pulses will be obtained. For example, the polynomial of PR4 channel is $(1+D)$ which corresponds to $n = 1$, the samples would be [1 1]. For $n = 2$, the samples of the isolated pulse is given by $(1+D)^2 = 1+2D+D^2$ polynomial which represents [1 2 1]. This family of PR channel is called the "extended partial response 4" or EPR4 channel. If $n = 3$, the polynomial will be $(1+D)^3 = 1+3D+3D^2+D^3$ and the samples of the isolated pulse are [1 3 3 1]. This type of PR channels is called the E^2PR4 channel. Table 2.2 lists the popular PR channels used in magnetic recording systems [12].

2.3.2 PRML Target Selection

The PR equaliser attempts to reshape the magnetic recording channel into a specific response. This specific response is called the target. The targets are often described by a

Table 2.2: The Popular PR Channels Used in Magnetic Recording

Name	Polynomial	Isolated Pulse samples	Impulse Response samples
PR4	(1-D)(1+D)	...0 1 1 0...	...0 1 0 -1 0...
EPR4	$(1\text{-D})(1\text{+D})^2$	...0 1 2 1 0...	...0 1 1 -1 -1 0...
E^{2}PR4	$(1\text{-D})(1\text{+D})^3$	...0 1 3 3 1 0...	...0 1 2 0 -2 -1 0...

polynomial in terms of T_c-second delay operators D [66] and classified into two types:

1. The PR targets with a polynomial of the form [67]

$$G(D) = (1-D)\sum_{l=0}^{L-1} g_l D^l \tag{2.15}$$

2. The Generalized Partial Response (GPR) targets with a polynomial of the form [67]

$$G(D) = \sum_{l=0}^{L-1} g_l D^l \tag{2.16}$$

Where g_l are positive, integer-valued coefficients and L is the target response length. In LMR systems, MR read head produces a pulse in response to a magnetic transition on the media. Constant magnetic polarity regions generate zero output voltage. Each transition gives an opposite voltage output of the last transition. In other words, the readback signal of the LMR system is dc-free signal. The $(1-D)$ term indicates that the first polynomial has a dc-free response which matches the low-frequency response of the overall response. Because of the term $(1-D)$ is implicit in the LMR magnetic channel, the first polynomial as described in Equation (2.15) is more suitable to the LMR targets than the second. However, in PMR channel dual-shielded MR reader produces the output voltage from the regions of constant magnetic polarity while at a magnetic transition the head gives zero

voltage. The output signal in PMR system is nonzero dc-response i.e. has a dc component [31][67]. Thus, the second type as described in (2.16) is more suited to the PMR channel as its main signal energy is at low frequencies. Therefore, making the first polynomial suitable for the PMR systems, the low-frequency energy should be filtered causing a loss in useful energy [67].

Sawaguchi, *et al.* investigated different polynomials of GPR targets [31]. It has been found that PR1-based DC-full GPR channel with a positive-coefficient target response had a match performance with the original waveform with an AWGN. Also, GPR3 channel with suppressed low-frequency components had a significant SNR improvements resulted from the better match of this channel with DC-distorted PMR channel.

Shah, *et al* presented a new method of designing the GPR targets for PMR channels. The method has been based on maximising the ratio of minimum squared Euclidean distance of the PR target to the noise penalty that is introduced by the PR filter. The method is suitable for the target that does not include noise prediction [72].

In this research work, a class of the GPR targets is used to simulate the PMR channel.

Ide in [73] proposed PR targets polynomial for perpendicular recording:

$$G(D) = (1+D)^P(D^Q-1)(1-D) \tag{2.17}$$

The dibit signal is a superposition of two single pulses; a single pulse and a time-shifted single pulse with opposite polarity. Data of the single pulse can be expressed as $(1+D)^P$ and (D^Q-1) for the superposition of two single pulses. Therefore, the data assignment to the dibit response is $(1+D)^P(D^Q-1)$, where P is the number of data assigned to a single pulse and Q represents the amount of time shift of the single pulse at the superposition. Depending on the values of P and Q, Equation 2.17 could have different forms which they are listed in Table 2.3.

Table 2.3: PRML Channels for Perpendicular and Longitudinal Magnetic Recording

	P	Q	$\mathbf{D}^0$	$\mathbf{D}^1$	$\mathbf{D}^2$	$\mathbf{D}^3$	$\mathbf{D}^4$	$\mathbf{D}^5$	$\mathbf{D}^6$
PMR1	1	2	-1	0	2	0	-1		
PMR2	1	3	-1	0	1	1	0	-1	
PMR3	2	2	-1	-1	2	2	-1	-1	
PMR4	2	3	-1	-1	1	2	1	-1	-1
PMR4	3	2	-1	-2	1	4	1	-2	-1

2.3.3 Maximum Likelihood Sequence Detector Method

Forney introduced the structure of ML sequence estimator for a digital pulse-amplitude-modulated sequence in the presence of finite ISI and white Gaussian noise [13]. The structure consists of a linear filter (whitened matched filter), a symbol-rate sampler, and a recursive nonlinear processor (Viterbi algorithm) as shown in Figure 2.8. This ML structure estimates the entire transmitted sequence [13] and it is defined as selecting the sequence that maximize the $p[y(D)|x(D)]$ probability, where $x(D)$ is the transmitted sequence and $y(D)$ is the received sequence, or finding the sequence that minimizes the probability of error for a received sequence of bits [10] as shown in Figure 2.9 [14]. The received signal is assumed to be corrupted by AWGN with zero mean and spectral density of $N_0/2$ W/Hz. Therefore, the output of the sampler (ADC) is the signal sequence

$$y_n = \sum_{k=-\infty}^{\infty} a_k x_{n-k} + v_n, -\infty < n < \infty \tag{2.18}$$

The sequence x_n is the sampled autocorrelation function of $h(t)$, which is,

$$x_n = \int_{-\infty}^{\infty} h(t)h(t+nT)dt \tag{2.19}$$

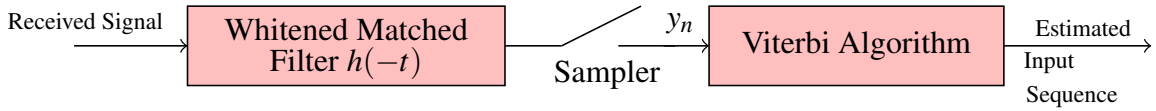

Figure 2.8: Optimal Detector Diagram [13]

And v_n is AWGN which is defined as

$$v_n = \int_{-\infty}^{\infty} w(t)h(t-kT)dt \tag{2.20}$$

The sampled signal sequence y_n is expressed as

$$y_n = a_n x_0 + \sum_{k=-\infty}^{\infty} a_k x_{n-k} + v_n, k \neq n \tag{2.21}$$

The term $a_n x_0$ represents the desired signal component and the second term is the ISI term. Practically, the summation of the ISI part can be truncated to a finite number of terms. The estimated data sequence is obtained by passing the sampled data y_n to the MLSD. An efficient algorithm for implementing MLSD is the VA [10]. Increasing the normalized density, D_s would increase the ISI. For example, if $D_s = 0.5$ the ISI is limited into two bits on either side of the desired bit however when $D_s = 3$ the ISI extends to about 10 bits on either side of the desired bit. In other word, if ISI spans 10 symbols then 2^{10} matched filters are needed which is practically not feasible way. Consequently, for long ISI spans, the matched filter is no longer practical. Therefore for magnetic recording systems, the matched filter of Figure 2.8 was replaced by a PR equaliser, a Finite Impulse Response (FIR) filter.

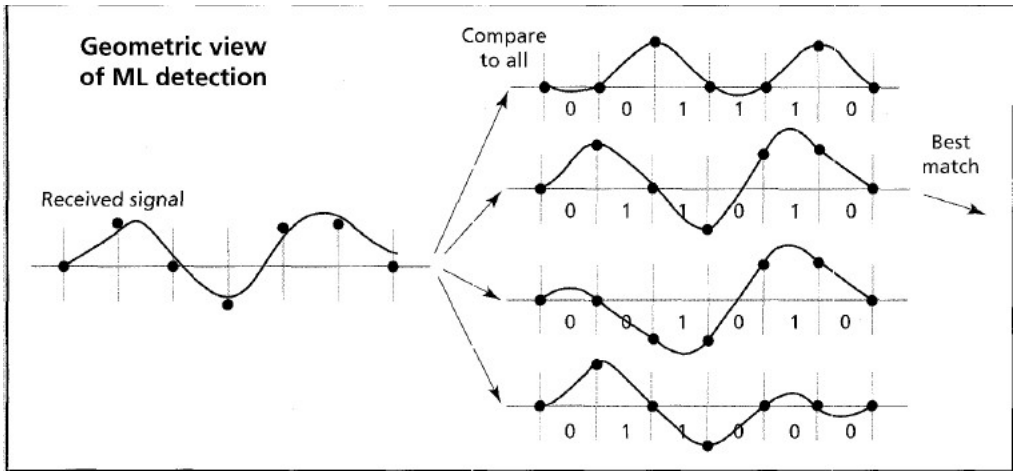

Figure 2.9: Geometric view of ML detection [14]

2.4 Detection Algorithms

In this section, the optimal detection algorithms that optimized for AWGN channel which are VA and MAP will be reviewed. However, the implementations and the modifications of these algorithms will be discussed in Chapter 3.

2.4.1 Viterbi Algorithm

In 1967, the so-called VA was introduced as a technique of decoding convolutional codes [74]. Then in 1972, Forney showed that VA solves the MLSD problem for ISI channel with AWGN [13]. Kobayashi and Tang recognised the possibility of applying this algorithm in magnetic recording systems for detection purposes [65]. Combining the Viterbi detector with PR equaliser in data storage channels to mitigate the ISI effects resulted in numerous commercial products [10]. Viterbi algorithm as an efficient algorithm has been used to implement the MLSD based on Hamming distance or Euclidean distance metrics. The complexity of VA can be classified into: (1) memory where for each state, the algorithm requires M memory location; (2) computation: in each unit of time, the algorithm must

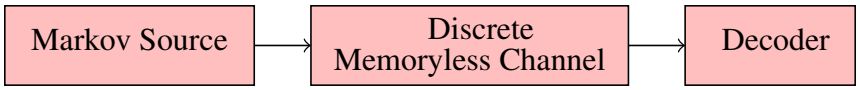

Figure 2.10: Schematic Diagram of transmission System [15]

make M^2 additions at most, one for each transition, and M comparisons among the M^2 results [54].

2.4.2 MAP Algorithm

Estimating the A Posteriori Probabilities (APP) of the states and transitions of Markov source for discrete memoryless channel was considered in [15] where an optimal decoding algorithm, BCJR algorithm, was applied to decode linear blocks and convolutional codes to minimize symbol error probability. The schematic diagram of the BCJR algorithm is shown in Figure 2.10 [15]. VA minimizes the probability of the sequence error while MAP minimizes the probability of the symbol (or bit) error. This algorithm will be discussed further in Chapter 3.

2.5 Readback Process of Perpendicular Recording System

The most essential part of magnetic recording systems is the data recovery (readback) process. Readback process techniques reflect the significant role of signal processing in magnetic recording systems. Read channel consists of band-limiting filter, sampler, equaliser and detector (it includes modulation encoder/decoder and , possibly, some auxiliary inner error correction or error detection encoder/decoder). The reading process starts with a magnetic read head, magnetic transducer, that converts magnetic information of the medium (tape or disk) to electrical signal and then pass it to the read channel. For both recording technologies, longitudinal and perpendicular, the basic shielded read head design is the same but the amount of flux is different. In perpendicular recording, the thicker recording

layer has increased the flux in the read head. In 1971 MR read head was invented to detect magnetic fields from the media. MR heads have some advantages over the inductive reader heads such as the signal amplitude is larger than in inductive head[10][12]. Also, this signal does not depend on the velocity which makes MR heads more interested for smaller disks with lower linear velocities. Then discovering GMR heads made a significant contribution in increasing the storage density[12].

2.6 Summary

In this chapter, PRML channel is introduced in detail and

- A review of the old writing technology, longitudinal magnetic recording, is introduced as it is necessary for comparison purposes with the current technology which is the perpendicular recording technology.

- The essential part of any magnetic recording system is retrieving the recorded information which reflects the role of the signal processing in the replay process. Therefore, PRML read channel is developed and explained in this chapter as a system that is used to recover the recorded data.

- PRML channel consists of two parts: PR equaliser and detector. The function of PR equaliser is modifying the impulse response of the channel and reshaping it into a per-defined target.

  The detector such as ML is applied in order to find the maximum likelihood sequence of the transmitted sequence.

- VA and MAP algorithms are applied to the magnetic recording systems as an optimal detection algorithms.

- For PMR channels, GPR targets are more accepted than PR targets.

Chapter 3

Implementation of PRML Channel

3.1 Introduction

In this chapter, simulation of the PMR channel is developed and described. Also, this chapter presents implementation of two detection algorithms which are VA and MAP. The implementation of the modified VA and MAP algorithms are introduced in detail. A comparison of their performances is made. Each block of the PRML system model is implemented in MATLAB (version 7.8.0) written by the author as a simulation environment. A frame of binary data with length of 1000 bits is randomly generated as an input sequence to the PRML channel model. The simulation would stop when 100 frames are in error for a particular SNR. The simulation runs for different values of SNR and the performance of the PRML channel is represented as BER vs. SNR graph.

3.2 Simulation of Perpendicular Recording Channel

In order to evaluate the performance of PMR channel in terms of BER against SNR, the channel model is implemented in MATLAB. Figure 2.4 shows the simulated PMR channel model.

A sequence of random binary data, a_k, is generated in order to represent the digital data that is intended to be recorded on the magnetic media. The MATLAB random function *randint* generates 1000 binary bits as a frame and this frame represents a sector of information on

the HDD. Every simulation starts by supplying the magnetic recording system with a randomly generated frame for a particular SNR value.

The a_k sequence is mapped to sequence of $[-1, +1]$ to simulate the write current, c_k, according to $(2a_k - 1)$, where the binary data sequence is fed into the write current driver to generate a two-level signal waveform called the write current [10]. The obtained sequence c_k represents the magnetization states where $+1$ means a positive transition from bit 0 to bit 1 occurs while -1 shows a negative transition from bit 1 to 0 [10].

For each single written transition, the transition response can be approximated by a *tanh* function as stated in Equation (2.12). The peak voltage, V_p can be set to "1" for amplitude normalisation purposes. The normalised density , D_s, is set to 0.5. The response of two adjacent transitions $p(t)$ is shown in Equation (2.13).
The sequence s_k is obtained as a result of the convolution of the mapped sequence, c_k, and the channel dibit response, $p(t)$.

Different sources of noise arise in magnetic recording systems such as the electronic noise in the GMR sensor and the preamplifier that amplifies the incoming signal and a noise from the medium magnetization variation [10]. The AWGN is added to the output of the channel (replay signal) to approximate the electronic noise, where in this work, the electronic noise is assumed as a dominant noise. The readback signal is sampled with a rate $1/T$ where T is the bit interval [10].
Then a digital equaliser is applied to equalise the readback signal samples into a desired GPR target to generate an output signal (response), z_k, which resembles the desired target output.

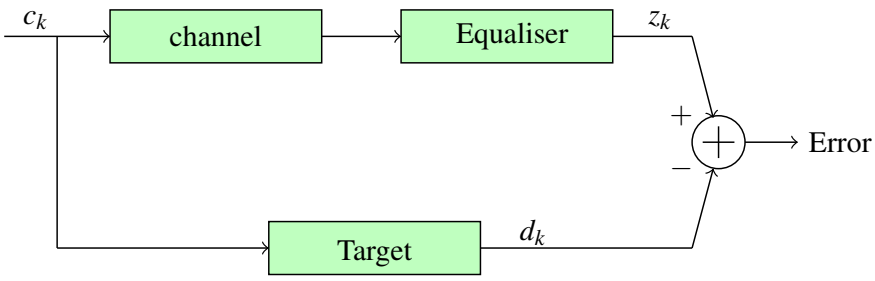

Figure 3.1: MMSE Equaliser Design [16]

Finally, MLSD is applied to deliver a solution which is used for performance evaluation.

3.3 Implementation of the Digital Equaliser

It is mentioned in Section 2.3.2 that GPR targets are more suited to the PMR channel as the readback signal of PMR channel has a DC component which represents the main signal energy. In order to find a suitable target for the PMR channel model, the following procedure is used in this work. Figure 3.1 shows the block diagram of the method that is used to find any target that suits a channel model. It is also mentioned in same section that the function of the equaliser is reshaping the channel to predetermined target with integer-valued coefficients. The total length of the impulse response of that equaliser is $(2K+1)$ [16] which is determined by the number of taps for a FIR filter. Also the delay lines in the FIR filter, should not be less than the duration of the pulse to be equalised. Typically 6 - 10 programmable taps are suitable for the FIR filters [54]. In this work, $K = 3$ is utilized to design a 7-tap PR equaliser.

The following procedure is used to shape the PMR channel which means finding the PR equaliser coefficients h, $\begin{bmatrix} h_0 & h_1 & h_2 & h_3 & h_4 & h_5 & h_6 \end{bmatrix}$.

1. From Equation (2.12) and Equation (2.13), evaluate the dibit response $p(t)$ for a given D_s.

2. From that response, $p(t)$, construct a 7 by 7 matrix X.

$$\begin{bmatrix} x(7) & x(8) & x(9) & x(10) & x(11) & x(12) & x(13) \\ x(8) & x(9) & x(10) & x(11) & x(12) & x(13) & x(14) \\ x(9) & x(10) & x(11) & x(12) & x(13) & x(14) & x(15) \\ x(10) & x(11) & x(12) & x(13) & x(14) & x(15) & x(16) \\ x(11) & x(12) & x(13) & x(14) & x(15) & x(16) & x(17) \\ x(12) & x(13) & x(14) & x(15) & x(16) & x(17) & x(18) \\ x(13) & x(14) & x(15) & x(16) & x(17) & x(18) & x(19) \end{bmatrix}$$

3. Create a vector Y which is the target coefficients.

$$Y = \begin{bmatrix} 0 & 0 & 4 & 6 & 4 & 0 & 0 \end{bmatrix}$$

4. Arrange the matrices as shown below:

$$\begin{bmatrix} x(7) & x(8) & x(9) & x(10) & x(11) & x(12) & x(13) \\ x(8) & x(9) & x(10) & x(11) & x(12) & x(13) & x(14) \\ x(9) & x(10) & x(11) & x(12) & x(13) & x(14) & x(15) \\ x(10) & x(11) & x(12) & x(13) & x(14) & x(15) & x(16) \\ x(11) & x(12) & x(13) & x(14) & x(15) & x(16) & x(17) \\ x(12) & x(13) & x(14) & x(15) & x(16) & x(17) & x(18) \\ x(13) & x(14) & x(15) & x(16) & x(17) & x(18) & x(19) \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 4 \\ 6 \\ 4 \\ 0 \\ 0 \end{bmatrix}$$

5. Finally, find the equaliser coefficients by getting the inverse of matrix X and then multiply it by vector Y, $h = X^{-1}Y$.

The values of the equaliser coefficients that found above depend upon the recording density, D_s. The Minimum Mean-Square Error (MMSE) criterion is used to find the best target function that suites the PMR channel [16]. Minimizing the mean squared error between the

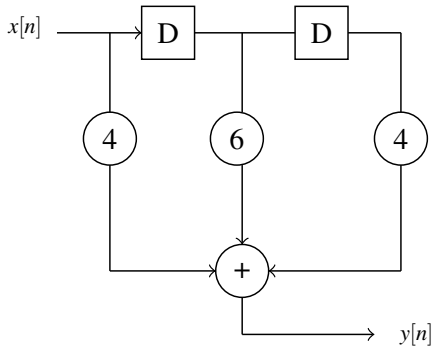

Figure 3.2: Block Diagram of the Ideal Channel

Table 3.1: States Table of The Ideal Channel

c_k	**Initial State**	**Final State**	d_k
-1	-1 -1	-1 -1	-14
+1	-1 -1	+1 -1	-6
-1	+1 -1	-1 +1	-2
+1	+1 -1	+1 +1	+6
-1	-1 +1	-1 -1	-6
+1	-1 +1	+1 -1	+2
-1	+1 +1	-1 +1	+6
+1	+1 +1	+1 +1	+14

equaliser output sequence and the desired signal (target output sequence) means the two sequences are identical.

The polynomial of the GPR target that is used in this work is

$$P(D) = 4 + 6D + 4D^2 \tag{3.1}$$

The block diagram of this polynomial is shown in Figure 3.2. The channel memory is 2 and the number of states is 4 which are (-1 -1, -1 1, 1 -1, 1 1). The state table of the ideal channel (target) is shown in Table 3.1. Consequently, a 4-state trellis is constructed with

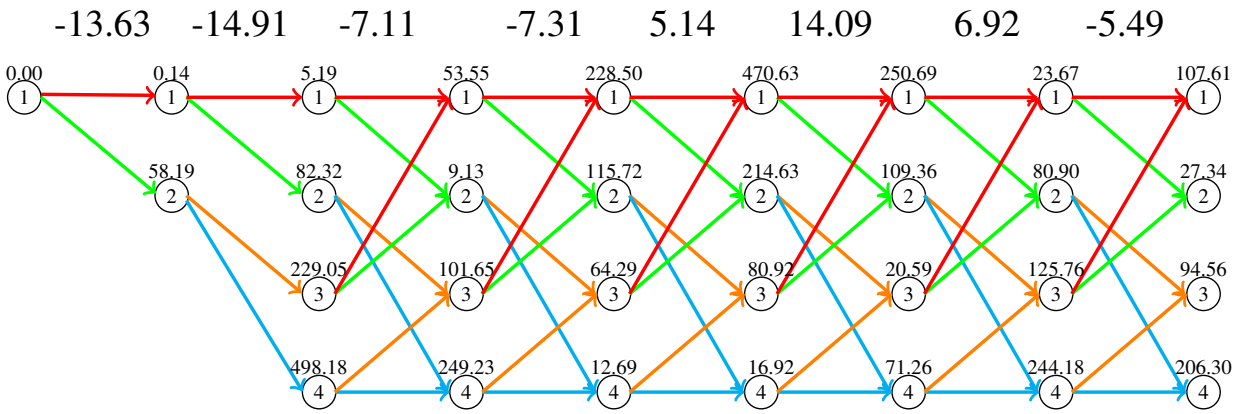

Figure 3.3: The GPR trellis diagram

two incoming paths and two outgoing paths at each state. Figure 3.3 shows the channel trellis for a specific sequence of the equaliser output.

3.4 Frequency Response of the Ideal Channel

This section presents finding the frequency response of the ideal channel (FIR filter) that shown in Figure 3.2 whose impulse response is (4 6 4). For discrete-time system, the difference equation that defines the output of the FIR filter in terms of its input is defined as [75]

$$y[n] = b_0 x[n] + b_1 x[n-1] + \ldots + b_n x[n-M] \tag{3.2}$$

Where

$x[n]$ is the input signal.

$y[n]$ is the output signal.

b_i are the filter coefficients or taps weight.

M is the filter order.

Generally, M^{th} order FIR filter has an impulse response with a length of $(M+1)$ samples. These samples called filter weights, filter coefficients or filter tap coefficients/weights [76].

Finding the frequency response of the ideal channel is explained in the following procedure:

1. ***Finding the difference equation***.

   The difference equation of the impulse response of this channel is expressed as:

$$y[n] = 4x[n] + 6x[n-1] + 4x[n-2] \tag{3.3}$$

2. ***Converting the equation domain into frequency domain***.

   In frequency domain, z-transform is used to describe discrete-time systems. The z-transform of a discrete-time signal, $x[n]$, is defined as [77]:

$$X(z) = \sum_{n=0}^{N-1} x[n]z^{-n} \tag{3.4}$$

   Therefore, z-transform of Equation 3.3 is

$$Y(z) = 4X(z) + 6X(z)Z^{-1} + 4X(z)Z^{-2} \tag{3.5}$$

3. ***Finding the transfer function***.

   The transfer function of the above equation is

$$H(z) = \frac{Y(z)}{X(z)} = 4 + 6Z^{-1} + 4Z^{-2} \tag{3.6}$$

4. ***Getting the symmetric form***.

   FIR filters with symmetric coefficients would guarantee linear phase. Linear phase means that all frequency components of the input signal will have the same delay which means the phase distortion will be avoided. Therefore, in order to get the

symmetric form of that equation, one position shifting is required. Thus,

$$H(z) = [4Z^{+1} + 6Z^{0} + 4Z^{-1}] \times Z^{-1} \tag{3.7}$$

5. ***Converting z-transform into polar form***.

   For good analysis, the transfer function would be represented in the polar coordinate which has form of *(magnitude* . e^{jphase}). Writing each z in polar form: $z = Ae^{j\theta}$, $e^{j\theta} = cos\theta \mp jsin\theta$ where A is the magnitude of z, j is the imaginary unit and θ is the angle or phase in radians which equals $2\pi f/F_s$.

   Therefore,

$$H(z) = 4 + 6e^{-j\theta} + 4e^{-j2\theta} \tag{3.8}$$

$$H(z) = 4 + 6e^{-j2\pi f/F_s} + 4e^{-j4\pi f/F_s} \tag{3.9}$$

$$H(z) = [4e^{j2\pi f/F_s} + 6 + 4e^{-j2\pi f/F_s}] \times e^{-j2\pi f/F_s} \tag{3.10}$$

$$= 4cos(2\pi f/F_s) + 4jsin(2\pi f/F_s) + 6 + 4cos(2\pi f/F_s) - 4jsin(2\pi f/F_s) \tag{3.11}$$

6. ***Finally, the magnitude and the phase responses are***

$$H(z) = [6 + 8\cos(2\pi f/F_s)] \times e^{-j2\pi f/F_s} \tag{3.12}$$

The Magnitude is $[6 + 8\cos(2\pi f/F_s)]$, where F_s is the sampling frequency and f/F_s is the normalised frequency. The magnitude is a cosine function has been multiplied by 8 and shifted up by 6 as shown in Figure 3.4. The phase is $(-2\pi f/F_s)$. If $w = 2\pi f$, the radian frequency, then the phase equals $-w/F_s$. The phase response is a linear function of frequency.

Thus, the target used in this work is a symmetrical FIR digital filter with linear phase and

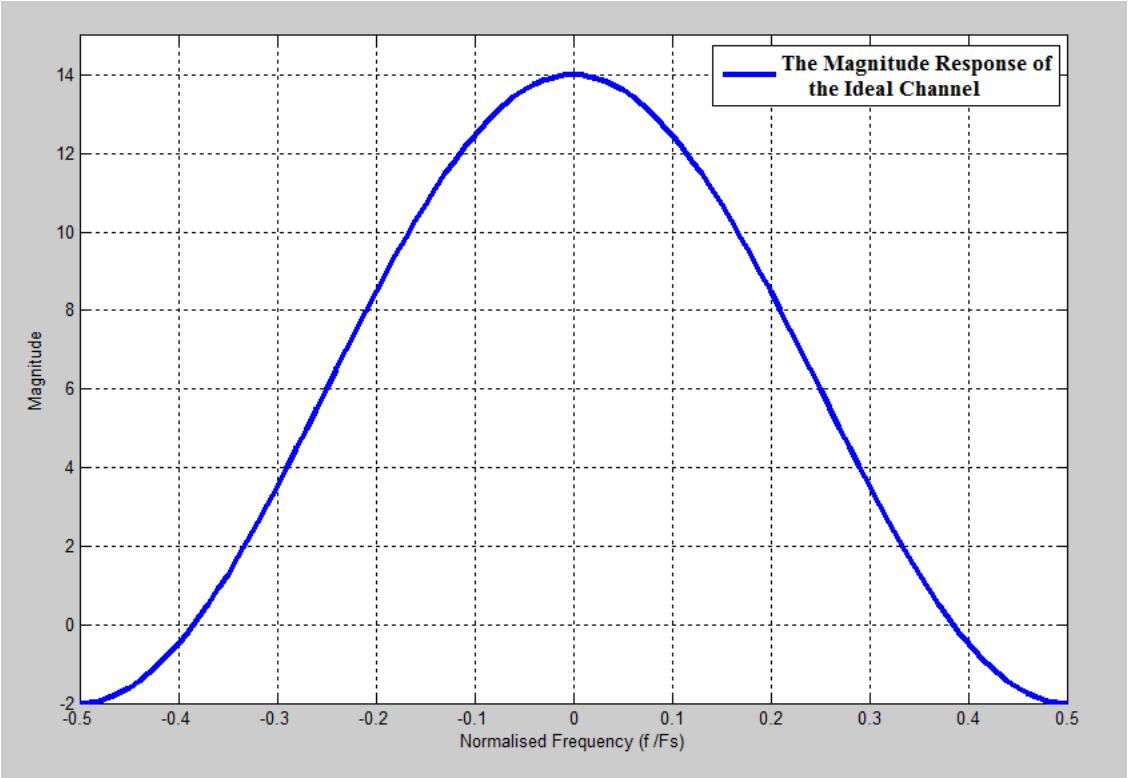

Figure 3.4: The Magnitude of the Frequency Response of the Ideal Channel

its group delay is $(N-1)/2$ samples.

The importance of the frequency domain representation comes from that the requirements for a digital filter are specified in the frequency domain. Designing a FIR filter consists of determining the desired magnitude response and/or the desired phase (delay) response which meets the given requirements on the filter response [78] [75].

3.5 SNR Definition in Magnetic Recording Systems

In digital communication, SNR definition is the ratio of information bit energy to noise spectral density E_b/N_0. This definition is unique for different types of coding and modulation schemes for a given bandwidth. However in magnetic recording channels, SNR

definition is based on the energy of the transition response $h(t)$ which depends on the inherent characteristic of the given head-medium assembly. Therefore, SNR definition is not generic for different techniques that have different code rates and operate at different linear densities. This is one of the difficulties that is faced by the researches working on data storage area from the coding and signal processing perspective. The unique definition of SNR would help them compare the performance merits of different schemes.

SNR definition in magnetic channels is the ratio of the energy in an isolated transition response to the noise E_t/N_0. Many sources of the disturbances in magnetic recording systems like noises (mentioned in Chapter 1) and distortions such as write/read distortions and interferences like ITI [12]. The following equation is the most commonly used SNR definition [12]:

$$SNR = \frac{\text{zero-to-peak signal power}}{\text{noise power}} = \frac{S}{N} = \frac{V_{0-p}^2}{V_{rms,n}^2} \tag{3.13}$$

Where $V_{rms,n}$ could include some other disturbances which are mentioned above. The peak signal voltage can be defined either as the isolated signal voltage peak or as the square wave amplitude recorded at the signal frequency. SNR is most often quoted in the units of dB:

$$SNR(dB) = 10\log\frac{S}{N} = 20\log\frac{V_{0-p}}{V_{rms,n}} \tag{3.14}$$

In this work, AWGN is added to the simulated read channel as a source of noise. SNR definition is found as follows:

$$\frac{E_t}{N_0} = \frac{E_t}{2\sigma^2} \tag{3.15}$$

Where E_t is the magnetic transition energy, N_0 is the power-spectral density of AWGN measured in $watt/Hz$ and σ^2 is the variance of that noise.

$$\sigma^2 = \frac{N_0}{2} \tag{3.16}$$

Table 3.2: Noise Power Values for Different Values of SNR in dB

SNR_{dB}	0	1	2	3	4	5	6	7	8	9	10
σ	0.7071	0.6302	0.5617	0.5006	0.4462	0.3976	0.3544	0.3159	0.2815	0.2509	0.2236

$$SNR_{dB} = 10\log(\frac{E_t}{N_0}) \tag{3.17}$$

$$SNR = \frac{E_t}{N_0} = 10^{SNR_{dB}/10} \tag{3.18}$$

Consequently, the dimensionless SNR is the ratio of impulse response power to the noise power. The noise power σ is calculated as follows:

$$\sigma = \sqrt{\frac{N_0}{2}} \tag{3.19}$$

From (3.18)

$$N_0 = \frac{1}{10^{SNR_{dB}/10}} \tag{3.20}$$

Therefore

$$\sigma = \sqrt{\frac{\frac{1}{10^{SNR_{dB}/10}}}{2}} \tag{3.21}$$

And

$$\sigma = \frac{1}{\sqrt{2 * SNR}} \tag{3.22}$$

The noise has been added by using *randn* function to randomly generate a vector of integer numbers then multiply it by above σ values. The complete PMR channel model that was shown in Figure 2.4 is simulated using MATLAB 7.8.0. The block diagram implementation includes channel, equaliser, MLSD implementation. For detection, two algorithms are implemented which are the VA and the MAP algorithm.

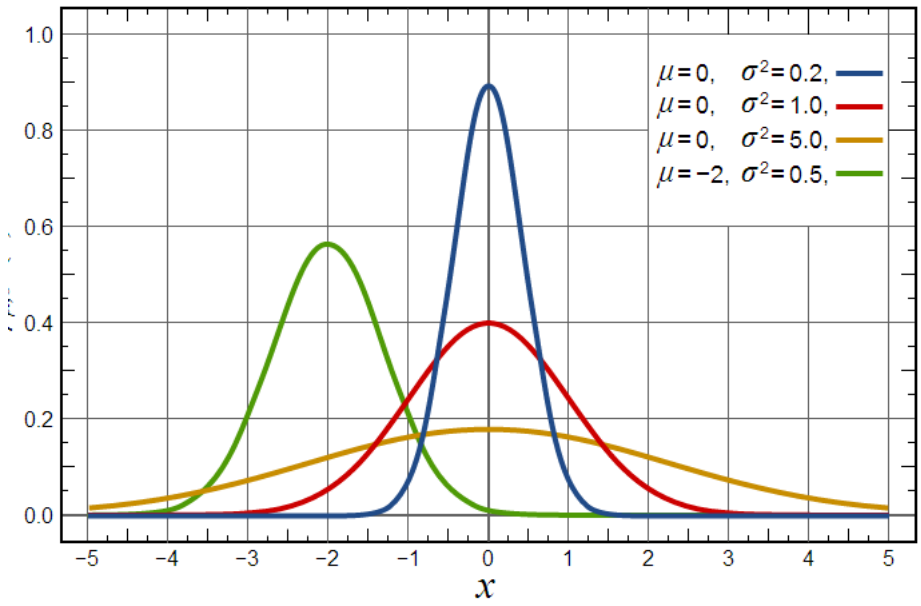

Figure 3.5: A Selection of Normal Distribution pdfs Where Both Mean and Variance Are Varied [17]

3.6 Standard Normal Distribution

Gaussian process $x(t)$ is a random function whose value x at any arbitrary time t is statistically described by Gaussian probability density function (pdf) [79]

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{3.23}$$

The parameter μ is the mean or the expectation of the distribution, σ is its standard deviation and σ^2 is the variance. The standardized or normalised Gaussian density function of a zero-mean process is obtained by assuming that $\sigma = 1$ [79]

$$p(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} \tag{3.24}$$

Where the factor $1/\sqrt{2\pi}$ ensures that the total area under the curve $p(x)$ is one. However, any normal distribution is a version of the standard normal distribution whose domain has been stretched by a factor σ and then translated by μ, that is

$$f(x) = \frac{1}{\sigma} p(\frac{x-\mu}{\sigma}) \tag{3.25}$$

To make sure the integral is still 1, the probability density must be scaled by $1/\sigma$.
Figure 3.5 shows Gaussian distributions for different values of variance and mean which have been applied to Equation 3.23 where the y-axis represents the probabilities values.

3.6.1 Memoryless Channel

Memoryless channel is defined as ISI-free channel where the channel output will be consisting of only the desired symbols [10] and there are no previous samples. Also, it is defined as the added noise is AWGN. In communication systems, additive noise process arises from the electronic components and amplifiers at the receiver which may be characterized as thermal noise. Statistically, thermal noise is described as a Gaussian noise process [79][80]. When this type of the noise is added to the mathematical model of the communication system, the resulting channel is called AWGN channel [80]. This channel model applies to many classes of the communication systems including magnetic recording system. However, in magnetic recording system the noise at the input of the detector is assumed white while it is not white because the equaliser changes the spectral density of the noise [10] as depicted in Figure 3.6. In order to make it AWGN again, the non white noise should be cancelled. Cancellation means two stages decoding process should be performed to that system. The first stage is estimation of AWGN from the decoder output by knowing the difference between the recorded data and the recovered one and then uncorrelated it.

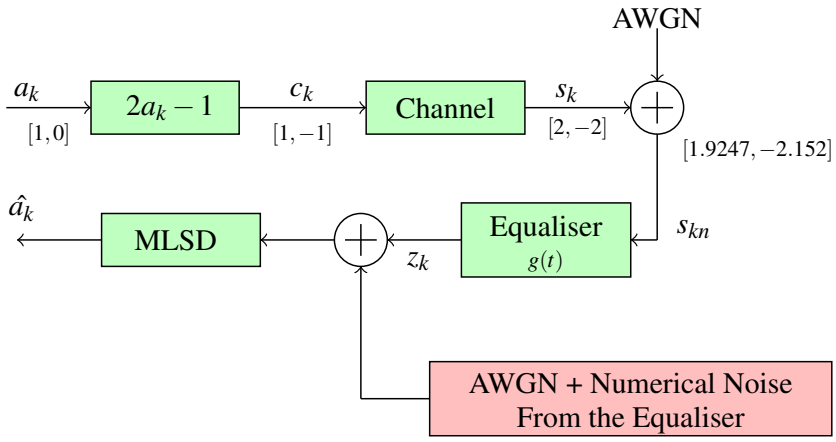

Figure 3.6: Memoryless Channel

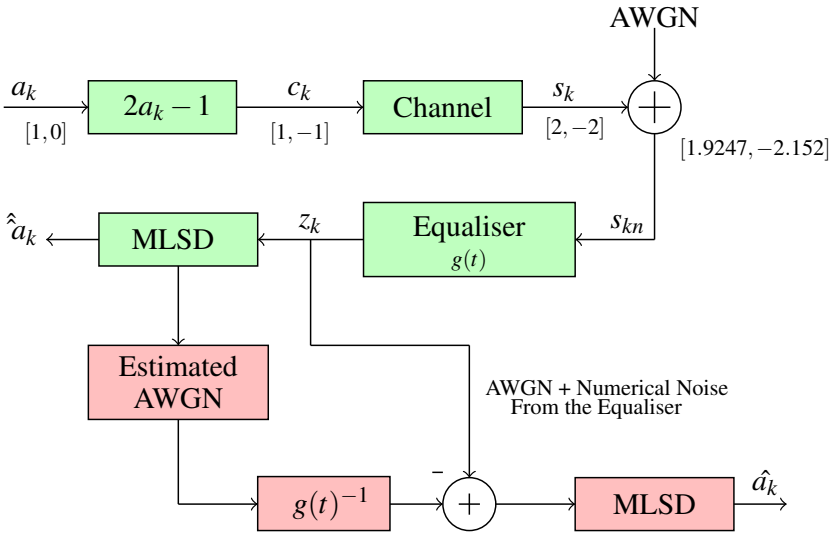

Figure 3.7: Non White Noise Cancellation of AWGN Channel

The second stage is adding the existing noise in addition to the new noise of the first stage into the decoder after subtracting the unwanted noise and then performing the decoding process again. Figure 3.7 explains the cancellation of non white noise in order to make this channel an AWGN channel.

White noise is a type of noise that has the same power spectral density over the frequency spectrum and it consists of different frequencies. The adjective 'white' used for this type of noise is quoted from the white light. White light consists of different colors (frequencies) of light combined together [81].

The following example explains the memoryless property. Suppose that a sequence, c_k, consists of two symbols $+1$ and -1 is transmitted through a memoryless channel which its impulse response consists of one sample (2). If the impulse response of the channel is just one sample, the output of the channel would be free ISI and has values of $(2,-2)$. The pdfs of the received sequence (assuming the added noise is AWGN and $\sigma = 0.7071$) would take Gaussian shape which are:
For the first symbol:

$$p(1.9247|1) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(1.9247-1)^2}{2\sigma^2}} = 0.2400 \tag{3.26}$$

And for the second symbol:

$$p(-2.152|-1) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(-2.152+1)^2}{2\sigma^2}} = 0.1496 \tag{3.27}$$

However, in reality the AWGN channel is not a ISI-free channel and its impulse response has many samples, [0.0027 0.1973 1.60 0.1973 0.0027], rather than the desired one as shown in Figure 3.8.
Consequently, an assumption should be made for the PMR channel which is this channel is memoryless channel and the noise is not AWGN otherwise Equation (3.23) is not valid.

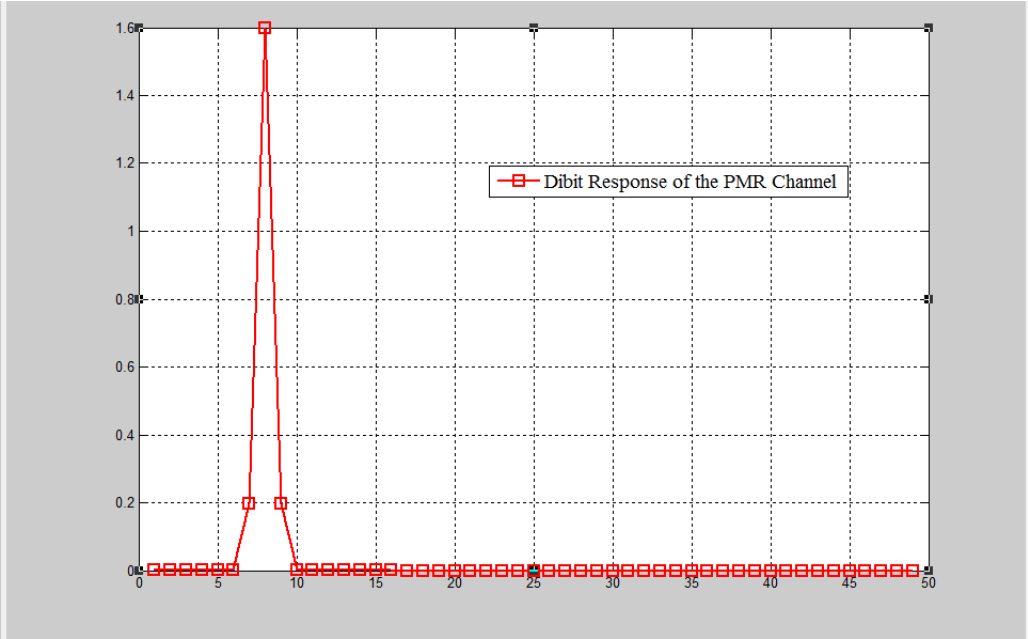

Figure 3.8: Sampled Dibit Response of the PMR channel

3.7 MLSD Decoder

If the transmitted signals were interdependent, a detector that makes its decision based on checking the sequence of the received signals over successive signal intervals will be the optimum detector. MLSD method searches the minimum Euclidean distance path through a trellis that describes the channel [80] of the communication systems including magnetic recording systems. In order to understand the MLSD algorithm and how to implement it, the next section will have an example of how to find the sequence that has the maximum pdf.

3.7.1 Implementation of MLSD

This section introduces the procedure of MLSD implementation. In order to understand the way of implementation, it would be useful to start with two symbols and then generalize it to unlimited sequence. Therefore, an assumption should be made which is the channel

input sequence has just two symbols. Thus, the channel output sequence is a sequence of two numbers s_{kn1} and s_{kn2}. Assume also that the channel noise is white and has Gaussian distribution therefore the noise sequence is also white. The output sequence of the channel will have the following pdfs:

$$pdf_1 = \frac{\alpha_1}{\sqrt{2\pi}\sigma} e^{-\frac{(z_{k1}-s_{kn1})^2}{2\sigma^2}} \tag{3.28}$$

And

$$pdf_2 = \frac{\alpha_2}{\sqrt{2\pi}\sigma} e^{-\frac{(z_{k2}-s_{kn2})^2}{2\sigma^2}} \tag{3.29}$$

Where $\alpha 1$ and $\alpha 2$ are scaling factor. If $\alpha_1 \neq \alpha_2$ then the joint pdfs for these two symbols can be expressed as a product of the their pdfs

$$f = \frac{\alpha_1.\alpha_2}{(\sqrt{2\pi}\sigma)^2} e^{-\frac{(z_{k1}-s_{kn1})^2+(z_{k2}-s_{kn2})^2}{2\sigma^2}} \tag{3.30}$$

The term $(\sqrt{2\pi}\sigma)^2$ is a constant only if σ is constant so that it can be ignored.
To simplify the computations, it might be useful using the natural logarithm (natural $log = log_e$)

$$\log(f) = \log\left(\alpha_1\alpha_2.e^{-\frac{(z_{k1}-s_{kn1})^2+(z_{k2}-s_{kn2})^2}{2\sigma^2}}\right) \tag{3.31}$$

Multiply Equation (3.31) by $2\sigma^2$

$$2\sigma^2\log(f) = 2\sigma^2\log\left(\alpha_1\alpha_2.e^{-\frac{(z_{k1}-s_{kn1})^2+(z_{k2}-s_{kn2})^2}{2\sigma^2}}\right) \tag{3.32}$$

Then Equation (3.32) is simplified using some algebraic properties of the natural logarithm such as:

- *The logarithm of the multiplication of a and b is the sum of logarithm of a and the*

logarithm of b; $log(a\ .\ b) = log(a) + log(b)$.

- *The natural logarithm function $ln(a)$ is the inverse function of the exponential function e^a, for $a > 0$; $e^{ln(a)} = a$.*

- *The logarithm of a raised to the power of b is b times the logarithm of a; $log(a^b) = b\ log(a)$.*

Therefore,

$$2\sigma^2 \log(f) = 2\sigma^2 \log \alpha_1 \alpha_2 + 2\sigma^2 \log[e^{-\frac{(z_{k1}-s_{kn1})^2+(z_{k2}-s_{kn2})^2}{2\sigma^2}}] \tag{3.33}$$

$$2\sigma^2 \log(f) = 2\sigma^2(\log \alpha_1 + \log \alpha_2) - [(z_{k1}-s_{kn1})^2 + (z_{k2}-s_{kn2})^2] \tag{3.34}$$

$$2\sigma^2 \log(f) = \log \alpha_1^{2\sigma^2} + \log \alpha_2^{2\sigma^2} - [(z_{k1}-s_{kn1})^2 + (z_{k2}-s_{kn2})^2] \tag{3.35}$$

Now, suppose a sequence of partial-response equalizer output z_{k1}, z_{k2}, ...z_{km} is received. The joint pdf of that sequence can be expressed again as a product of M marginal pdfs

$$p(z_{k1}, z_{k2}, ...z_{km}|s_{kn1}, s_{kn2}, ...s_{knm}) = \prod_{m=1}^{M} \frac{\alpha}{\sqrt{2\pi}\sigma} e^{-\frac{(z_{km}-s_{knm})^2}{2\sigma^2}}$$

$$= \left(\frac{\alpha}{\sqrt{2\pi}\sigma}\right)^M \exp\left(-\sum_{m=1}^{M} \frac{(z_{km}-s_{knm})^2}{2\sigma^2}\right) \tag{3.36}$$

After following the procedure of two symbols, the obtained equation is

$$2\sigma^2[\log p(z_{k1}, z_{k2}, ...z_{km}|s_{kn1}, s_{kn2}, ...s_{knm})] = -[\log \alpha^{2M\sigma^2} + \sum_{m=1}^{M} (z_{km}-s_{knm})^2] \tag{3.37}$$

$$= -(\log \alpha^{2M} + \log \alpha^{\sigma^2}) - \sum_{m=1}^{M} (z_{km}-s_{knm})^2 \tag{3.38}$$

Table 3.3: The pdfs of The Ideal Output Samples

d_k	-14	**-6**	-2	**+6**	+2	+14
α	1/8	**2/8**	1/8	**2/8**	1/8	1/8

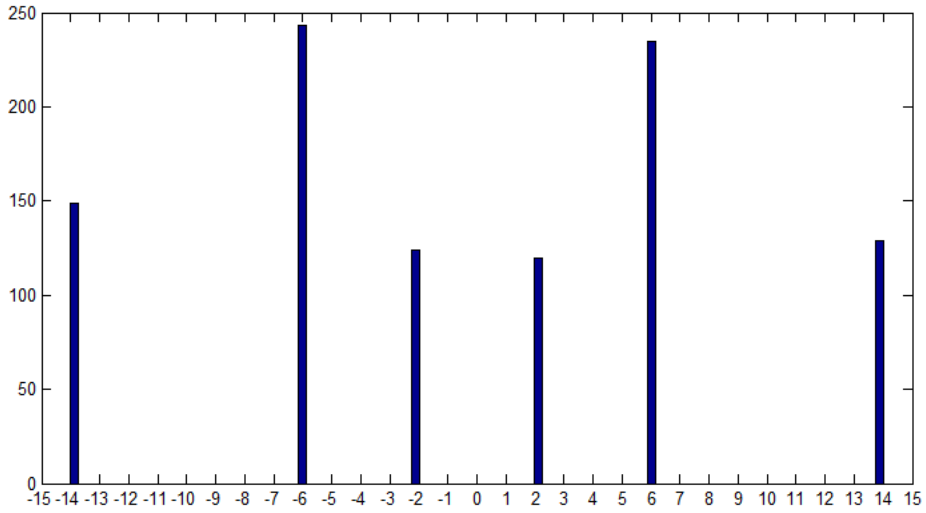

Figure 3.9: The pdfs Histogram of The Ideal Output Samples

The conclusion that arises from Equation (3.38) is to find the corresponding maximum likelihood sequence, minimization of the right hand side of that equation is required.

The next section will introduce the implementation of the first detection algorithm which is VA and then the modified VA. The modified VA includes adding the $(\log \alpha^{2M} + \log \alpha^{\sigma^2})$ term to the path metric which requires add ONE additional sum in the path metric computations. Table 3.3 shows the α values that have been used in the simulations and they are related to the received sequence d_k. Figure 3.9 shows the histograms of α values. The modified VA shows an improvement in the BER and FER graphs for different SNR.

3.8 Implementation of the Viterbi Detector

VA as an efficient algorithm has been used to implement the MLSD based on Hamming distance or Euclidean distance metrics. VA calculates efficiently the probabilities of the whole transmitted sequences and finds their maximum value. In other words, the function of VA is finding a path through a trellis that characterizes a channel with three operations which are add-compare-select that are performed in an iterative way. At each time interval, a metric is computed at each state and the metrics of all paths coming at each state are compared. The best metric is selected and the other paths will be ignored. This best metric is called the survivor which should be stored with its state at each time interval. To employ VA to the PMR channel shown in Figure 2.4, the channel polynomial should be defined. This polynomial is defined in Equation (3.1). The algorithm is described in the following procedure:

(1) At time $t = 0$, initialize the first state as a root state and put its metric to zero.

(2) At time $t = t + 1$, new symbol is received.

- Expand all the paths by one branch and compute the metric for each path entering each state by finding the Euclidean distance between the received sequence and the possible transmitted sequence.

- Add the path metric to branch metric and save the minimum and its state. If two paths are equal, they have the same metric, randomly choose the best path.

(3) increase t and repeat the above procedure until the end of the block.

3.8.1 Modification of Viterbi Algorithm

(1) At time $t = 0$, initialize the first state as a root state and put its metric to zero.

(2) At time $t = t + 1$, new symbol is received.

- Expand all the paths by one branch and compute the metric for each path entering each state by finding the Euclidean distance between the received sequence and the possible transmitted sequence. Add new expression to that metric which is the $(\log \alpha^{2\sigma^2})$ term.

- Add the path metric to branch metric and save the minimum and its state. If two paths are equal, they have the same metric, randomly choose the best path.

(3) Increase t and repeat the above procedure until the end of the block.

3.9 Implementation of MAP Algorithm

MAP technique as an optimal decoder makes bit by bit decisions can be applied for PMR channel system. The algorithm is illustrated in the following steps:

I. Forward Recursion (computing $\alpha's$ probabilities)

1. Initialize the forward probabilities $\alpha's(m)$ and the backward probabilities $\beta's(m)$, m= 0,1,..., M-1 where M is the number of states as in

   $[\alpha(1), \alpha(2), \alpha(3), \alpha(4)]$= [1 0 0 0] .

   $[\beta(1), \beta(2), \beta(3), \beta(4)]$= [1/4 1/4 1/4 1/4].

2. For $t = 0, 1, 2..., N-1$, compute the transition probability $\gamma's$ for AWGN channel using equation

$$p(r|z_k^{m,n}) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(r_n - z_k^{m,n})^2}{2\sigma^2}} \tag{3.39}$$

   Where N is the sequence length, m is the first state and n is the next state.

3. Propagate α using

$$\alpha_{t+1}(m) = \sum_{n=0}^{M-1} \alpha_t(n)\gamma(m,n) \tag{3.40}$$

4. Increase t and repeat step (3) until the end of the block then start the backward recursion.

II. Backward Recursion

1. From $t = N-1$ to 0, find $\beta's$ by using the following equation

$$\beta_t(n) = \sum_{n=0}^{M-1} \beta_t(m)\gamma(m,n) \tag{3.41}$$

2. Compute APP for each bit:

$$p(x) = \sum \alpha(m)\gamma(m,n)\beta(n) \tag{3.42}$$

3.9.1 MAP Implementation: An Example

Following is an example of how to implement the MAP algorithm. The implementation includes calculation of transition probabilities $\gamma's$, forward probabilities $\alpha's$ and backward probabilities $\beta's$.

Unlike VA, the MAP decoder has no knowledge about the previous state and the next state therefore the transition probabilities $\gamma's$ should be firstly calculated. MAP algorithm starts with $\gamma's$ calculations which depend on the noise power, σ, consequently this algorithm is not valid if the system does not has noise (ideal case). On the contrary, VA algorithm could be applied even if the system does not has amount of noise.

1. ***Calculation of transition probabilities***

   Using Equation (3.39) for $z_k = -14.0106$ and assuming that the standard deviation

σ of the Gaussian noise as 0.7071:

State 1

$$\gamma_1 = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(-14.0106-(-14))^2}{2\sigma^2}} = 0.5643 \tag{3.43}$$

$$\gamma_2 = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(-14.0106-(-6))^2}{2\sigma^2}} = 7.6287e^{-29} \tag{3.44}$$

State 2

$$\gamma_3 = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(-14.0106-(-6))^2}{2\sigma^2}} = 7.6287e^{-29} \tag{3.45}$$

$$\gamma_4 = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(-14.0106-2)^2}{2\sigma^2}} = 2.6464e^{-112} \tag{3.46}$$

State 3

$$\gamma_5 = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(-14.0106-(-2))^2}{2\sigma^2}} = 1.263e^{-63} \tag{3.47}$$

$$\gamma_6 = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(-14.0106-6)^2}{2\sigma^2}} = 7.0181e^{-175} \tag{3.48}$$

State 4

$$\gamma_7 = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(-14.0106-6)^2}{2\sigma^2}} = 7.0181e^{-175} \tag{3.49}$$

$$\gamma_8 = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(-14.0106-14)^2}{2\sigma^2}} = 0.0 \tag{3.50}$$

The normalised γ values are:

$$\gamma_{total} = \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 + \gamma_5 + \gamma_6 + \gamma_7 + \gamma_8 = 0.5643 \tag{3.51}$$

$$\gamma_1 = \frac{\gamma_1}{\gamma_{total}} = 0.5643/0.5643 = 1 \tag{3.52}$$

$$\gamma_2 = \frac{\gamma_2}{\gamma_{total}} = 7.6287e^{-29}/0.5643 = 1.352e^{-28} \tag{3.53}$$

$$\gamma_3 = \frac{\gamma_3}{\gamma_{total}} = 7.6287e^{-29}/0.5643 = 1.352e^{-28} \tag{3.54}$$

$$\gamma_4 = \frac{\gamma_4}{\gamma_{total}} = 2.6464e^{-112}/0.5643 = 4.6899e^{-112} \tag{3.55}$$

$$\gamma_5 = \frac{\gamma_5}{\gamma_{total}} = 1.263e^{-63}/0.5643 = 2.2382e^{-63} \tag{3.56}$$

$$\gamma_6 = \frac{\gamma_6}{\gamma_{total}} = 7.0181e^{-175}/0.5643 = 1.2437e^{-174} \tag{3.57}$$

$$\gamma_7 = \frac{\gamma_7}{\gamma_{total}} = 7.0181e^{-175}/0.5643 = 1.2437e^{-174} \tag{3.58}$$

$$\gamma_8 = \frac{\gamma_8}{\gamma_{total}} = 0/0.5643 = 0 \tag{3.59}$$

2. ***Calculation of Alpha Recursion***

The forward probabilities $\alpha's$ should be initialized with:

$[\alpha(1), \alpha(2), \alpha(3), \alpha(4)]$ = [1 0 0 0].

State 1

$$\alpha_{11} = \alpha(1) \times \gamma_1 = 1 \times 1 = 1 \tag{3.60}$$

$$\alpha_{12} = \alpha(3) \times \gamma_2 = 0 \times 1.352e^{-28} = 0 \tag{3.61}$$

$$\alpha_1 = \alpha_{11} + \alpha_{12} = 1 \tag{3.62}$$

State 2

$$\alpha_{21} = \alpha(1) \times \gamma_3 = 1 \times 1.352e^{-28} = 1.352e^{-28} \tag{3.63}$$

$$\alpha_{22} = \alpha(3) \times \gamma_4 = 0 \times 4.6899e^{-112} = 0 \tag{3.64}$$

$$\alpha_2 = \alpha_{21} + \alpha_{22} = 1.352e^{-28} \tag{3.65}$$

State 3

$$\alpha_{31} = \alpha(2) \times \gamma_5 = 0 \times 2.2382e^{-63} = 0 \tag{3.66}$$

$$\alpha_{32} = \alpha(4) \times \gamma_6 = 0 \times 1.2437e^{-174} = 0 \tag{3.67}$$

$$\alpha_3 = \alpha_{31} + \alpha_{32} = 0 \tag{3.68}$$

State 4

$$\alpha_{41} = \alpha(2) \times \gamma_7 = 0 \times 1.2437e^{-174} = 0 \tag{3.69}$$

$$\alpha_{42} = \alpha(4) \times \gamma_8 = 0 \times 0 = 0 \tag{3.70}$$

$$\alpha_4 = \alpha_{41} + \alpha_{42} = 0 \tag{3.71}$$

The normalised α recursion values are:

$\alpha_{total} = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1 + 1.352e^{-28} + 0 + 0 = 1$

$$\alpha_1 = \frac{\alpha_1}{\alpha_{total}} = \frac{1}{1} = 1 \tag{3.72}$$

$$\alpha_2 = \frac{\alpha_2}{\alpha_{total}} = \frac{1.352e^{-28}}{1} = 1.352e^{-28} \tag{3.73}$$

$$\alpha_3 = \frac{\alpha_3}{\alpha_{total}} = \frac{0}{1} = 0 \tag{3.74}$$

$$\alpha_4 = \frac{\alpha_4}{\alpha_{total}} = \frac{0}{1} = 0 \tag{3.75}$$

3. ***Calculation of Beta Recursion***

The backward probabilities $\beta's$ should be initialized with:

$[\beta(1), \beta(2), \beta(3), \beta(4)]$ = [0.25 0.25 0.25 0.25].

State 1

$$\beta_{11} = \beta(1) \times \gamma_1 = 0.25 \times 1 = 0.25 \tag{3.76}$$

$$\beta_{12} = \beta(2) \times \gamma_3 = 0.25 \times 1.352e^{-28} = 3.3799e^{-29} \tag{3.77}$$

$$\beta_1 = \beta_{11} + \beta_{12} = 0.25 \tag{3.78}$$

State 2

$$\beta_{21} = \beta(3) \times \gamma_5 = 0.25 \times 2.2382e^{-63} = 5.5956e^{-64} \tag{3.79}$$

$$\beta_{22} = \beta(4) \times \gamma_7 = 0.25 \times 1.2437e^{-174} = 3.1094e^{-175} \tag{3.80}$$

$$\beta_2 = \beta_{21} + \beta_{22} = 5.5956e^{-64} \tag{3.81}$$

State 3

$$\beta_{31} = \beta(1) \times \gamma_2 = 0.25 \times 1.352e^{-28} = 3.3799e^{-29} \tag{3.82}$$

$$\beta_{32} = \beta(2) \times \gamma_4 = 0.25 \times 4.6899e^{-112} = 1.1725e^{-112} \tag{3.83}$$

$$\beta_3 = \beta_{31} + \beta_{32} = 3.3799e^{-29} \tag{3.84}$$

State 4

$$\beta_{41} = \beta(3) \times \gamma_6 = 0.25 \times 1.2437e^{-174} = 3.1094e^{-175} \tag{3.85}$$

$$\beta_{42} = \beta(4) \times \gamma_7 = 0.25 \times 1.2437e^{-174} = 3.1094e^{-175} \tag{3.86}$$

$$\beta_4 = \beta_{41} + \beta_{42} = 6.2187e^{-175} \tag{3.87}$$

The normalised β recursion values are:

$\beta_{total} = \beta_1 + \beta_2 + \beta_3 + \beta_4 = 0.25 + 5.5955e^{-64} + 3.3799e^{-29} + 6.2187e^{-175} = 0.25$

$$\beta_1 = \frac{\beta_1}{\beta_{total}} = \frac{0.25}{0.25} = 1 \tag{3.88}$$

$$\beta_2 = \frac{\beta_2}{\beta_{total}} = \frac{5.5955e^{-64}}{0.25} = 2.2382e^{-63} \tag{3.89}$$

$$\beta_3 = \frac{\beta_3}{\beta_{total}} = \frac{3.3799e^{-29}}{0.25} = 1.3520e^{-28} \tag{3.90}$$

$$\beta_4 = \frac{\beta_4}{\beta_{total}} = \frac{6.2187e^{-175}}{0.25} = 2.4875e^{-174} \tag{3.91}$$

The values of γ, α and β are normalised for the numerical stability sake.

4. ***Calculating APP***

Once $\gamma's$, $\alpha's$ and $\beta's$ are calculated, the decoder can make a decision whether the received symbol is 1 or -1 by calculating APP:

Finding the probability of 1's

$$p_{11} = \alpha_1 \times \gamma_3 \times \beta_2 = 1 \times 1.352e^{-28} \times 2.2382e^{-63} = 3.026e^{-91} \tag{3.92}$$

$$p_{12} = \alpha_2 \times \gamma_4 \times \beta_4 = 1.352e^{-28} \times 4.6899e^{-112} \times 2.8475e^{-174} = 1.5772e^{-313} \tag{3.93}$$

$$p_{13} = \alpha_3 \times \gamma_7 \times \beta_2 = 0 \times 1.2437e^{-174} \times 2.2382e^{-63} = 0 \tag{3.94}$$

$$p_{14} = \alpha_4 \times \gamma_8 \times \beta_4 = 0 \times 0 \times 2.8475e^{-174} = 0 \tag{3.95}$$

$$App = p_{11} + p_{12} + p_{13} + p_{14} = 3.026e^{-91} + 1.5772e^{-313} + 0 + 0 = 3.026e^{-91} \tag{3.96}$$

Finding the probability of -1's

$$p_{01} = \alpha_1 \times \gamma_1 \times \beta_1 = 1 \times 1 \times 1 = 1 \tag{3.97}$$

$$p_{02} = \alpha_2 \times \gamma_2 \times \beta_3 = 1.352e^{-28} \times 1.352e^{-28} \times 1.352e^{-28} = 2.4711e^{-084} \quad (3.98)$$

$$p_{03} = \alpha_3 \times \gamma_5 \times \beta_1 = 0 \times 2.2382e^{-63} \times 1 = 0 \quad (3.99)$$

$$p_{04} = \alpha_4 \times \gamma_6 \times \beta_3 = 0 \times 1.2437e^{-174} \times 1.352e^{-28} = 0 \quad (3.100)$$

$$App = p_{01} + p_{02} + p_{03} + p_{04} = 1 + 2.4711e^{-084} + 0 + 0 = 1 \quad (3.101)$$

If the probability of 1's is greater than the probability of -1's, then the received bit is 1 otherwise it is -1.

3.9.2 Modification of MAP Algorithm

I. Forward Recursion

1. Initialize the forward probabilities $\alpha's$ and the backward probabilities $\beta's$ as in

   $[\alpha(1), \alpha(2), \alpha(3), \alpha(4)]$= [1 0 0 0].

   $[\beta(1), \beta(2), \beta(3), \beta(4)]$= [1/4 1/4 1/4 1/4].

2. For $t = 0, 1, 2..., N-1$, compute the transition probability γ between the states using the following equations

   For -14, 14, -2, 2

   $$p(r|z_k^{m,n}) = \frac{1/8}{\sqrt{2\pi}\sigma} e^{-\frac{(r_n - z_k^{m,n})^2}{2\sigma^2}} \quad (3.102)$$

   For 6, -6

   $$p(r|z_k^{m,n}) = \frac{2/8}{\sqrt{2\pi}\sigma} e^{-\frac{(r_n - z_k^{m,n})^2}{2\sigma^2}} \quad (3.103)$$

3. Propagate α using

   $$\alpha_{t+1}(m) = \sum_{n=0}^{M-1} \alpha_t(n)\gamma(m,n) \quad (3.104)$$

4. Increase t and repeat step (3) until the end of the block then start the backward recursion.

II. Backward Recursion

1. From $t = N-1$ to 0, find β by using the following equation

$$\beta_t(n) = \sum_{n=0}^{M-1} \beta_t(m)\gamma(m,n) \tag{3.105}$$

2. Compute the a posterior probability for each bit:

$$p(x) = \sum \alpha(m)\gamma(m,n)\beta(n) \tag{3.106}$$

3.10 Simulation Results and Discussion

3.10.1 Distance Metrics Graphs

In order to obtain the shortest path through the trellis that characterizes the channel, the distance between the recorded and the recovered sequences must be a small value.

Figure 3.10 shows the relationship of first and the second minimums of channel trellis with the time intervals before adding AWGN. It is clear that there is a big difference between the first and the second minimums because the two sequences (the recorded and the recovered) are almost the same.

Figure 3.11 shows the same relationship but after adding AWGN ($\sigma = 0.3544$) for VA and the modified VA decoders. The modification has made a big difference in terms of values however there is no a significant difference in terms of performance. This figure shows the efficiency of the trellis in performing the probabilities calculations. The difference between

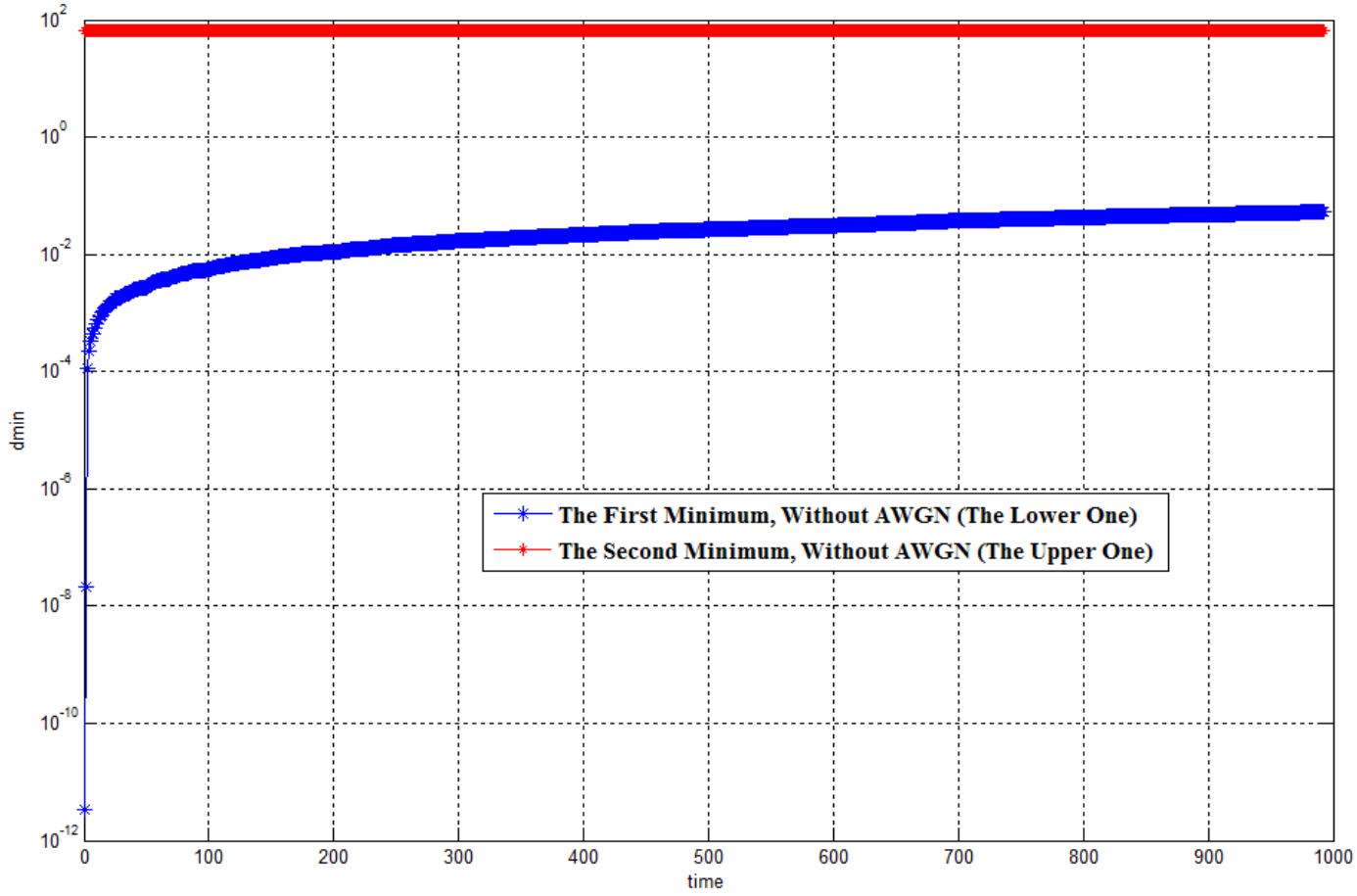

Figure 3.10: The First and the Second Minimums for GPR Trellis Without AWGN

the two minimums is similar, as shown in Figure 3.12, which indicates that the trellis tracks the statistics efficiently.

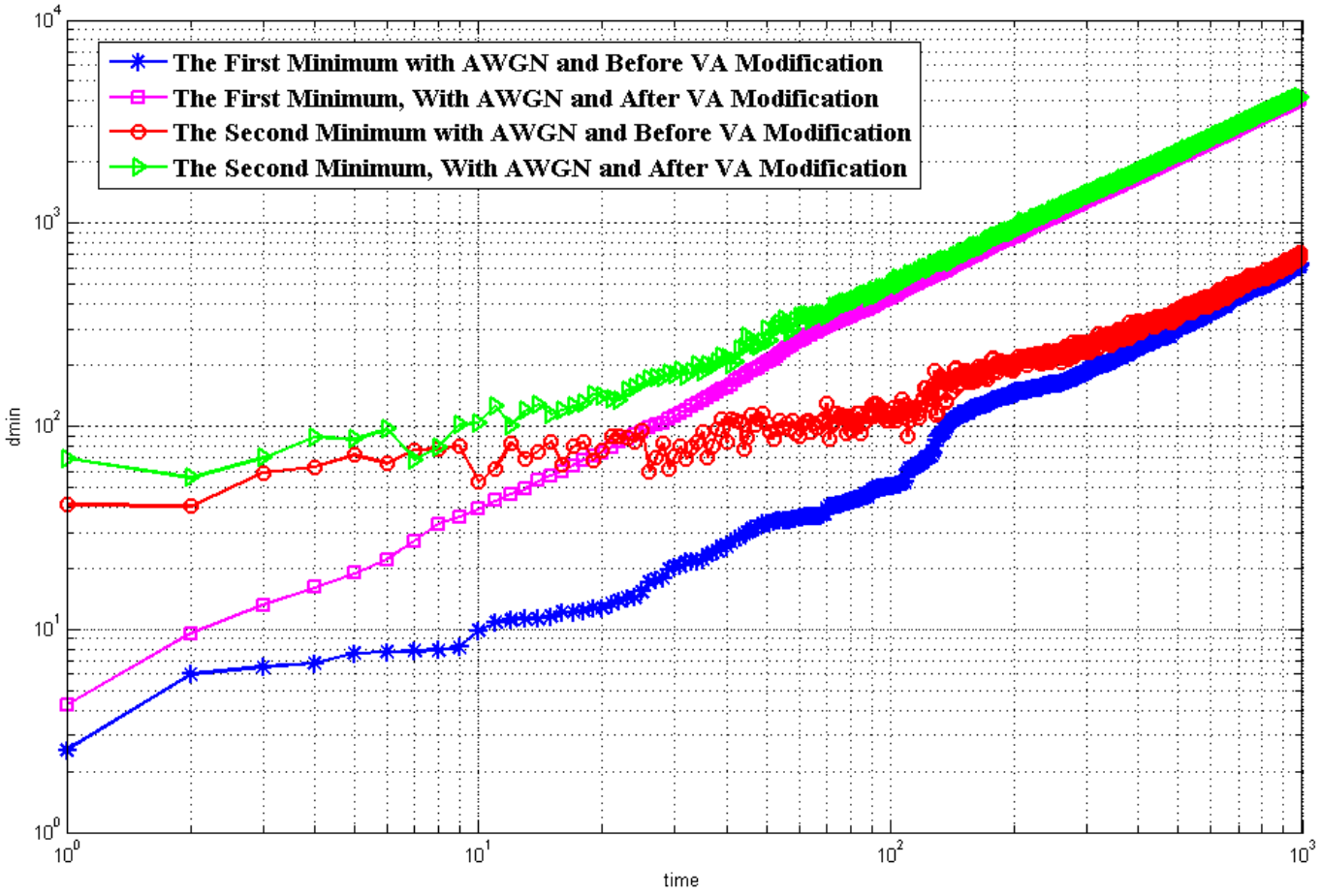

Figure 3.11: The First and the Second Minimums of VA and Modified VA Trellises

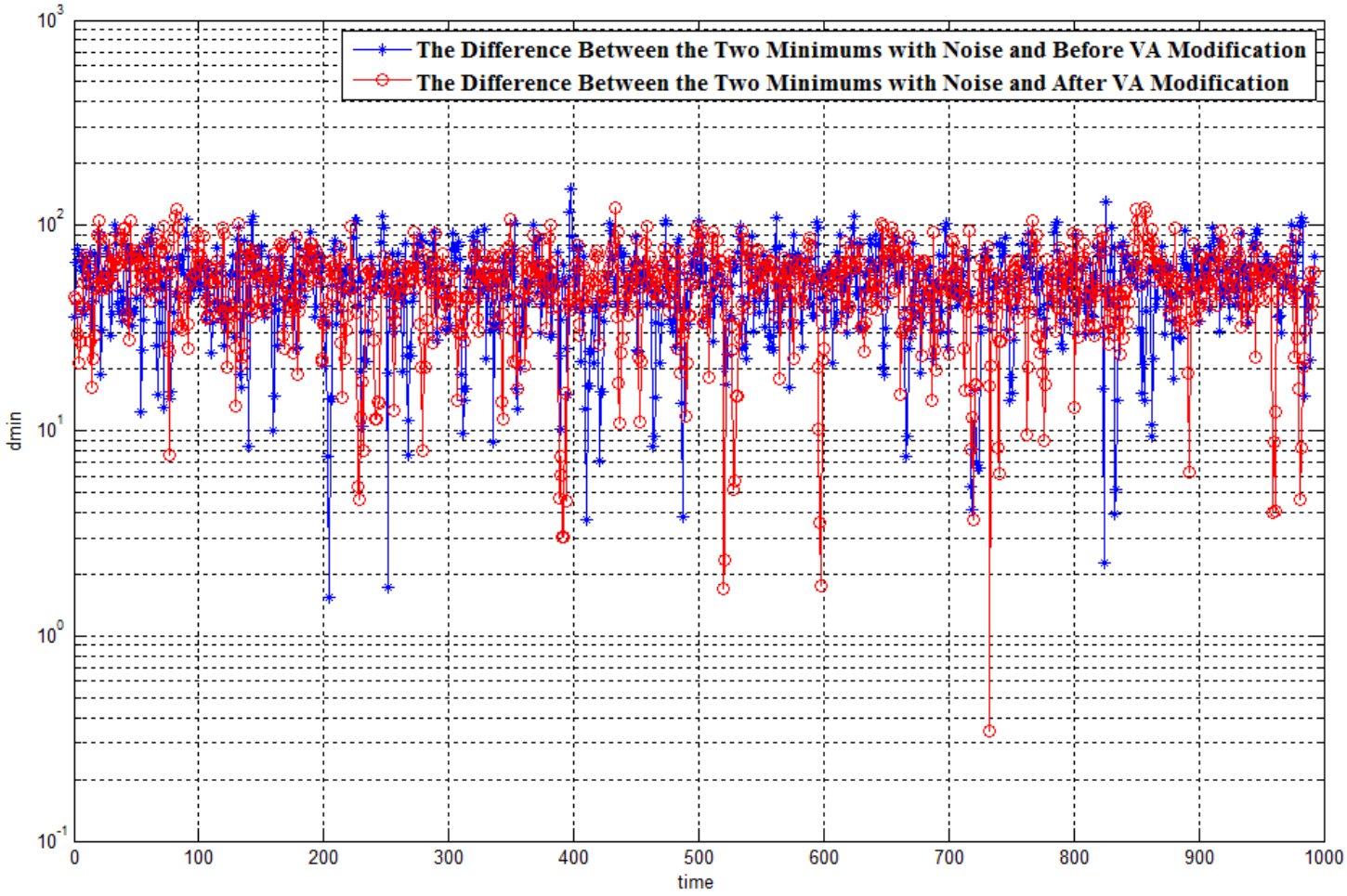

Figure 3.12: The Difference Between the First and the Second Minimums of VA and the Modified VA Trellises

3.10.2 PRML System Performance

The performance of PRML channel is measured by identifying the number of errors between the recorded and the recovered sequences. The BER and FER graphs have been obtained by computer simulations for GPR channel with AWGN at 0.5 user density.

Figure 3.13 shows the BER and FER performances for PRML system before and after VA modification. This figure shows that from 0 dB to 6 dB the two graphs are the same with no improvements but after 6 dB the improvements are arisen to 8 dB and then after 10 dB no improvements could be obtained.

However for MAP decoding, Figure 3.14 shows a tiny improvements in the BER and FER performances between 6 and 8 dB. The main reason for this fluctuations is that the dominant error events are changing as the SNR changes.

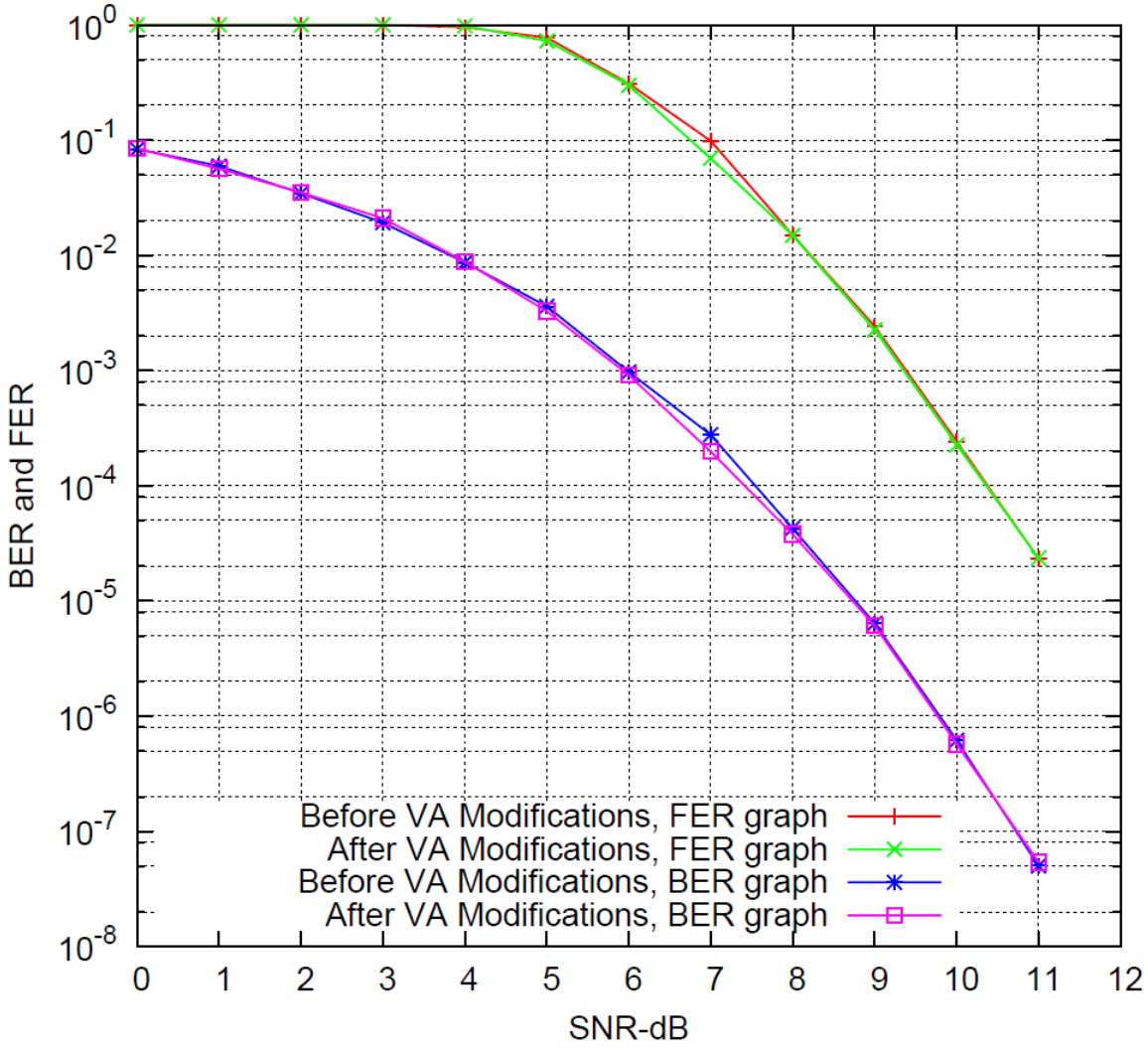

Figure 3.13: BER and FER Performances for PRML Before and After VA Modifications

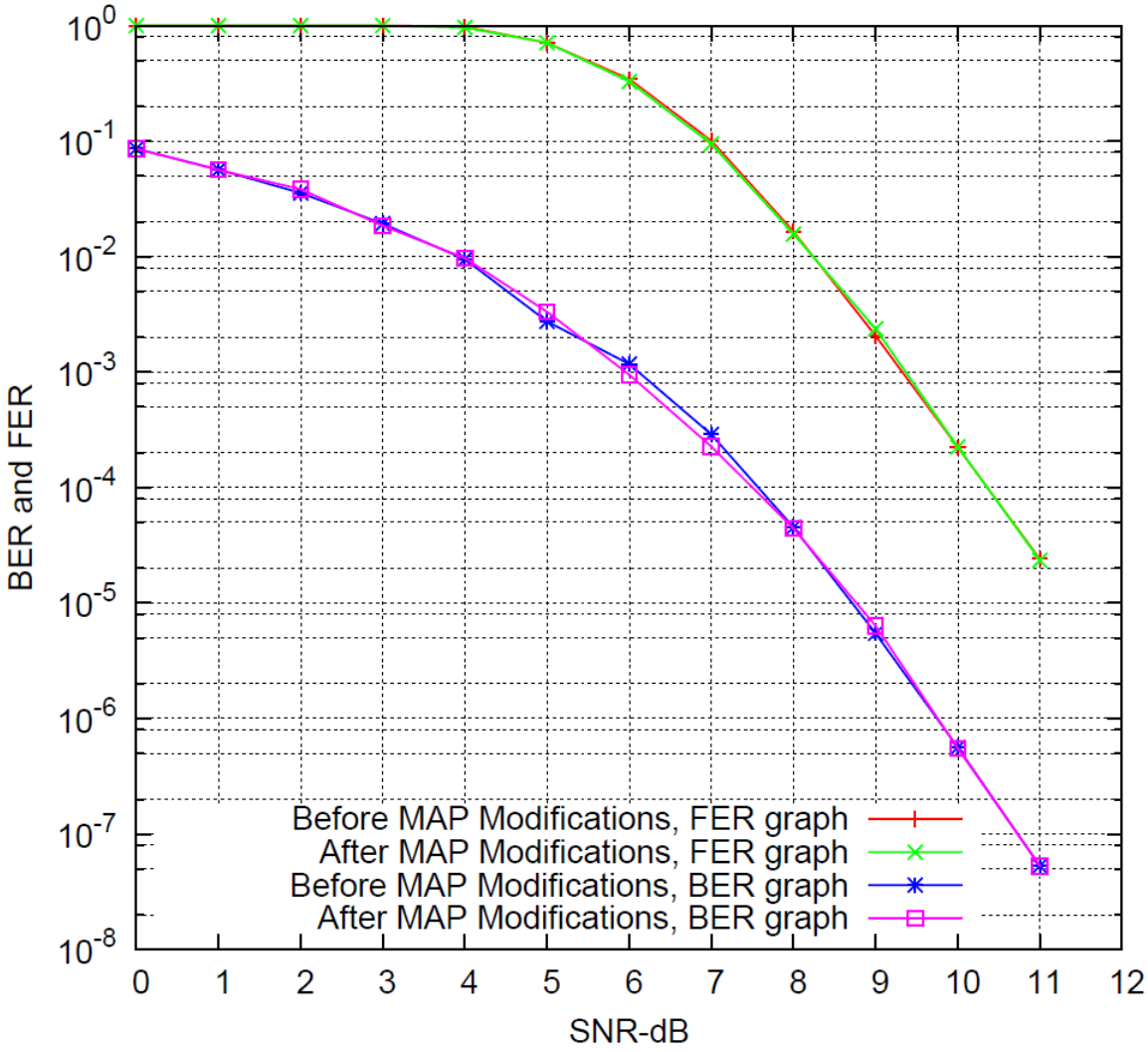

Figure 3.14: BER and FER Performances Before and After MAP Modification

3.10.3 Effective SNR

Table 3.4 compares the BER and the FER performances for the GPRML channel with AWGN at 12 dB. It shows the number of erroneous bits and frames that are obtained before and after VA modifications. Number of frames are 10000 with each frame having 1000 bits. The table depicts that in the third, forth and the sixth runs, the erroneous frames are zero that means all the frames that have been received are correct while before modifying VA the number of erroneous frames were very big.

Run No.	No. of Erroneous Bits Before VA Modifications	No. of Erroneous Bits After VA Modifications	No. of Erroneous Frames Before VA Modifications	No. of Erroneous Frames After VA Modifications
1	3	5	1460	9886
2	4	7	8773	5895
3	4	0	6620	0
4	8	0	7921	0
5	3	3	9945	2047
6	6	0	8123	0
7	2	2	2733	3369
8	2	9	6986	8694
9	6	3	7834	2539
10	3	2	4842	1009

Table 3.4: VA Performances Comparison

3.11 Summary

This chapter could be summarized as follows:

- The PMR channel shown in Figure 2.4 has been simulated.
- The optimal PRML system with integer coefficients GPR target polynomial has been described and implemented to find the performance of the read channel. The PRML implementation includes investigation of PR equaliser and ML detector.
- PR equaliser digital filter is designed in order to equalise the dibit response of PMR channel.
- The appropriate target for the PMR channel system is selected based on MMSE criterion.
- The derivation of maximum likelihood sequence for ML decoder is detailed.
- VA is applied in order to find the potential path sequence in the trellis of the channel. VA has been modified according to the pdfs of the ideal channel (target). The modification involves one additional sum in the path metric computation. All the derivations of VA before and after the modifications have been depicted. The performance comparison between before and after modification graphs is performed by computer simulation. The results show that there is an improvement in the BER and FER graphs but the changing of the dominant error events has an effect on its performance.
- As a part of the PRML system investigation, the optimum bit based decoder, MAP algorithm, is also applied and then modified based on the pdfs of GPR target.

- The GPR target used in this research work has a good performance where Figure 3.11 shows the distance metrics as a function of time for the trellis of the PMR channel. The divergence between the first and the second minimums is almost zero especially after many time intervals which indicates that the trellis tracks the statistics efficiently.

Chapter 4

Shingled System Implementation

4.1 Introduction

In this chapter, the decoding for shingled magnetic recording system is investigated. Decoding in the presence of ITI in PMR channel, which was not covered in Chapter 3, is studied in this chapter. Two models are described and simulated for PMR channel in this thesis. The first one is the PMR system with no ITI (one-track model) which was discussed in Chapter 3. The second one is with ITI (two-track model) that will be implemented in this chapter. The two-track equalisation and detection algorithms are investigated and implemented. However, for an ITI system, a complicated 16-state trellis is constructed for a known ITI amount ranging from zero to 0.5. Different techniques are listed to find the maximum likelihood solution for the ITI system. The complexity is evaluated for each technique and the results show that the complexity is reduced by more than three times with 0.5 dB loss in performance. BER graphs show a trade off can be achieved between the BER and complexity for shingled recording decoder.

Parts of this chapter appear in the Conferences under the paper titled:

- ***Nadia Awad, Mohammed Zaki Ahmed, and Paul Davey "On Reducing the Complexity of Shingled Decoding", IEEE International Magnetics Conference, INTERMAG 2012, Vancouver, Canada, May 2012***.

- ***Nadia Awad, Mohammed Zaki Ahmed, and Paul Davey "New Technique on Finding the Path Metrics of the Maximum Likelihood Sequence Decoder", 12th Joint***

MMM/Intermag Conference, Chicago, USA, January 2013.

4.2 Complexity

Complexity is the measure of algorithm efficiency which is defined as the number of arithmetic operations per detected symbol, branch or frame [10]. In this thesis, the definition of complexity is not just the number of arithmetic operations, but the number of arithmetic and comparison operations after obtaining the sampled data. In this research work, the way of finding this complexity is achieved by tracking the number of computations per decoded bit. However, finding the complexity of the 16-state trellis will be discussed in detail in Section 4.6. The complexity could be reduced further if some operations were completed off-line. Off-line means some operations would be performed while the received sequence is being equalized. By taking the off-line operation advantage, the overall complexity is reduced by more than three times compared to the complexity of the optimum performance method.

The following example explains finding the on-line and off-line complexities of the equation below:

$$min(a,b)+c+d+(e-z_{k_i})^2 \tag{4.1}$$

Where a,b,c,d, and e are constants and z_{k_i} is the received sample. The complexity of this equation would be 2 as $[min(a,b)+c+d]$ can be computed independently before/during obtaining the sample z_{k_i}. Therefore, the only computation that is required after obtaining z_{k_i} is $(e-z_{k_i})^2$.

4.3 What is ITI?

In magnetic recording systems, the areal density could be increased in two directions: radial direction and axial direction. In radial direction, the track width is decreased which causes ITI while in axial direction the pulse width is decreased which leads to ISI. Intertrack interference or the cross-talk is a magnetic transition picked up by the read head from the adjacent track. It has been considered as one of the factors that degrades the performance of some detectors like the detectors that are employed in narrow-track systems [10]. However even for wide width tracks, ITI does exist due to the recording head misalignment [82]. This chapter is concerned with the equalization and detection of the perpendicular recording channel subject to both ISI and ITI (shingled system). The following list summarises the work on ITI:

- 1988- ITI model was developed by Abbott, *et al.* [83] considering Off-Track Interference (OTI)/ITI as the dominant noise source in the magnetic disk storage channel. The model found the ITI distribution at the output of different types of equalisers such as PR and decision feedback equalisers. The results showed that the ITI distribution had a significant effect on the error probability.

- 1990- Barbosa believed that the cross-talk is a source of errors and described a maximum likelihood detector technique which included array heads detect the readback signals from interfering tracks simultaneously [84].

- 1991- A multichannel scheme for magnetic recording channel model was proposed. This scheme included old information in the guard band and ITI from the adjacent track [85].

- 1994- Lee, *et al.* [86] introduced new approach for multitrack RLL codes by finding

minimum and maximum run-length constraints. However, this work showed that the maximum run-length constraint does not have a significant effect.

- 1994- Davey, *et al.* [87] investigated two-dimensional codes to combat ITI for a multi-track recording system. The results showed that this code could introduce an improvement in terms of immunity to ITI while maintaining a higher linear density.

- 1998- Soljanin extended Barbosa work to multiple (more than two) heads that simultaneously read data from multiple track [88].

- 2002- Roh, *et al.* investigated two different equalisation/detection methods for longitudinal channel. The first method was considering OTI/ITI, as a noise and only the main track data being equalised. Then OTI was considered as a signal coming from an independent data sequence as a second method [89].

- 2003- New class of two-dimensional RLL recording codes was discovered and it showed significant improvements over the conventional RLL codes [22].

- 2005- Tan, *et al.* discussed single-track and joint-track equalisation and detection methods for different targets. For joint-track equalisation, the simulation results showed that the short targets had acceptable performance however 0.25 dB loss in performance at BER= 10^{-5} for long targets [90].

- 2005- Three-track decoding system of longitudinal magnetic recording channel subject to both ISI and ITI was used in [91]. This scheme allows a significant amount of ITI from interfering each track with its two neighboring tracks.

- 2009- Wood, *et al.* [9] proposed an approach based on shingled writing and 2-D readback techniques. Theoretically, this approach together with powerful signal processing tools might achieve an areal-densities of the order of 10 Tbit/in^2.

- 2009- Jermey *et al.* [92] investigated three adjacent tracks decoding system where the tracks had a narrow track-width and were not separated by guard bands.

- 2009- Greaves *et al.* [93] investigated shingled recording scheme and designed a write head, assuming the track pitch is 30nm where an areal density of 2 Tbit/in^2 was the target [93]. It was confirmed that the difficulty of reducing the written track width below 50 nm which limited the areal density of the conventional method to around 1 Tbit/in^2.

  From 1994 to 2002, the track densities have increased from 160 tracks/mm to 3540 tracks/mm for longitudinal recording method.
  In 2007, the track densities were about 590 tracks/mm with a track pitch of 1.7μm for perpendicular recording technique [2].
  However, Krishnan *et al.* and Chan *et al.* [94] [95] [96] outlined the TDMR technique as a novel storage architecture which could achieve densities toward 10 Tbit/in^2.
  In terms of cost, TDMR is a more suitable candidate technique among the other technologies as the signal processing algorithms could be embedded on a chip compared with the other systems that would require new heads and media [96].

- 2010- Chooruang [63] concluded that for SMR, the areal density can increase at least by factor of 1.85 (114.36 Gb/in^2) using the same conventional heads and media.

- 2012- A Shingle Write Disk (SWD) emulator was implemented which was deployed as a linux driver on top of the existing physical disks [58]. The Emulator used a PMR hard disk and emulated a SWD on a top of it.

- 2013- SMR simulation software was designed and developed to simulate the performance of a SMR disk [97].

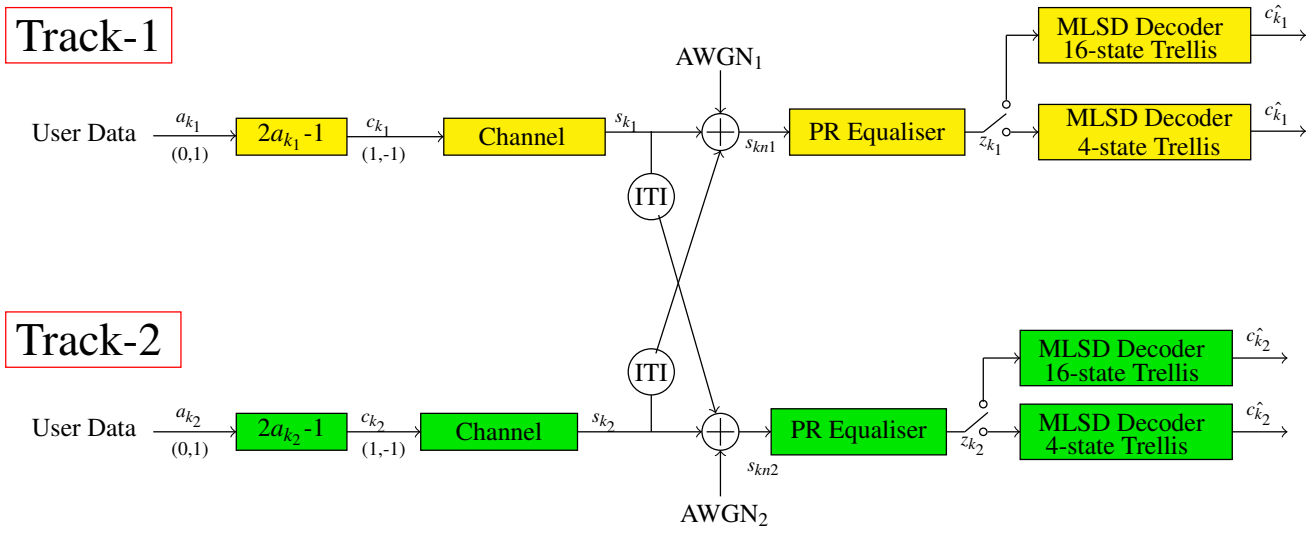

Figure 4.1: Shingled Magnetic Recording System Model

- 2013- Shingled recording scheme with a single read head was investigated with focus on reducing the read latency degradation that was caused by ITI in that scheme [98].

4.4 Implementation of Shingled System

Equalisation and detection techniques of a single data track for a single head model were considered in Chapter 3. This model is extended to have two interfering tracks supposing that the signal is picked up from the track which is under the read head and some fraction from the adjacent track. Two different equalisation/detection techniques are investigated. The first one considers ITI as a noise summed with AWGN. The PR equaliser would equalise the main data track, ITI noise and AWGN. The second technique views ITI as a signal coming from the adjacent track. PR equaliser now shapes the signal arising from the main track plus the signal coming from the adjacent track. New trellis is required to include the new signal, consequently, 16-state trellis is built up with four incoming paths and four outgoing paths at each state. The following two sections present the implementation of 2-D equalisation and detection techniques in order to implement the ITI system model.

4.4.1 2-D Equaliser Implementation

It is assumed that the read head picks up the signal from the track under it and a fraction from the adjacent track. Two-track one-head model is simulated assuming that ITI exists between the two adjacent tracks with no interaction between the groups of tracks. The implementation procedure of 2-D equaliser is in the same way that followed in Chapter 3. Figure 4.1 depicts the schematic of shingled system model (two-track one-head model) that is implemented in MATLAB written by the author.

ITI as a Noise

The input sequences to the two equalisers are

$$s_{kn_1} = s_{k_1} + \underbrace{\left(\alpha s_{k_2} + n_1\right)}_{\text{noise}}$$

$$s_{kn_2} = s_{k_2} + \underbrace{\left(\alpha s_{k_1} + n_2\right)}_{\text{noise}}$$

where,

s_{kn_1} = Readback signal from the first track

s_{kn_2} = Readback signal from the second track

s_{k_1} = Ideal ITI-free signal from the first track

s_{k_2} = Ideal ITI-free signal from the secondtrack

$n_1 = AWGN_1$ sample

$n_2 = AWGN_2$ sample

α = amount of ITI

The same target of one-track model is used for different ITI amounts, i.e. α= 0, 0.02, 0.04,...0.5.

ITI as a Signal

The same equaliser is applied to consider ITI as a signal except the sequence z_k now includes some crosstalk that comes from the adjacent track as shown in the following equations:

$$s_{kn_1} = \underbrace{\left(s_{k_1} + \alpha s_{k_2}\right)}_{\text{signal}} + n_1$$

$$s_{kn_2} = \underbrace{\left(s_{k_2} + \alpha s_{k_1}\right)}_{\text{signal}} + n_2$$

4.4.2 2-D Detection

MLSD is used to find the most likely path sequence in the trellis of the shingled system model. For detection, VA is applied to find that optimum sequence. The minimum Euclidean distance is used to find the minimum values between the two sequences [12] z_k and d_k, where this distance determines the probability of error detection. For example,

$$st = (z_k - d_k)^2 + x \tag{4.2}$$

Where st is the state metric of the next state and x is the state metric of the previous state. For each state at each time, the minimum value would be found by finding that state metric. The state metric is obtained by adding the squared difference between the received sequence and the possible transmitted sequence to the previous path metric. The possible transmitted sequence of one-track model is [-14 -6 -2 -6...]. Similar procedure is followed to find the MLSD for the equaliser that considers the ITI as a noise and as a signal. For the two-track model, a new trellis is constructed based on the signal that comes from the second track which is called ITI. The new trellis has 16 states with four outgoing paths from every state and four incoming paths to every trellis state as shown in Figure 4.2. For

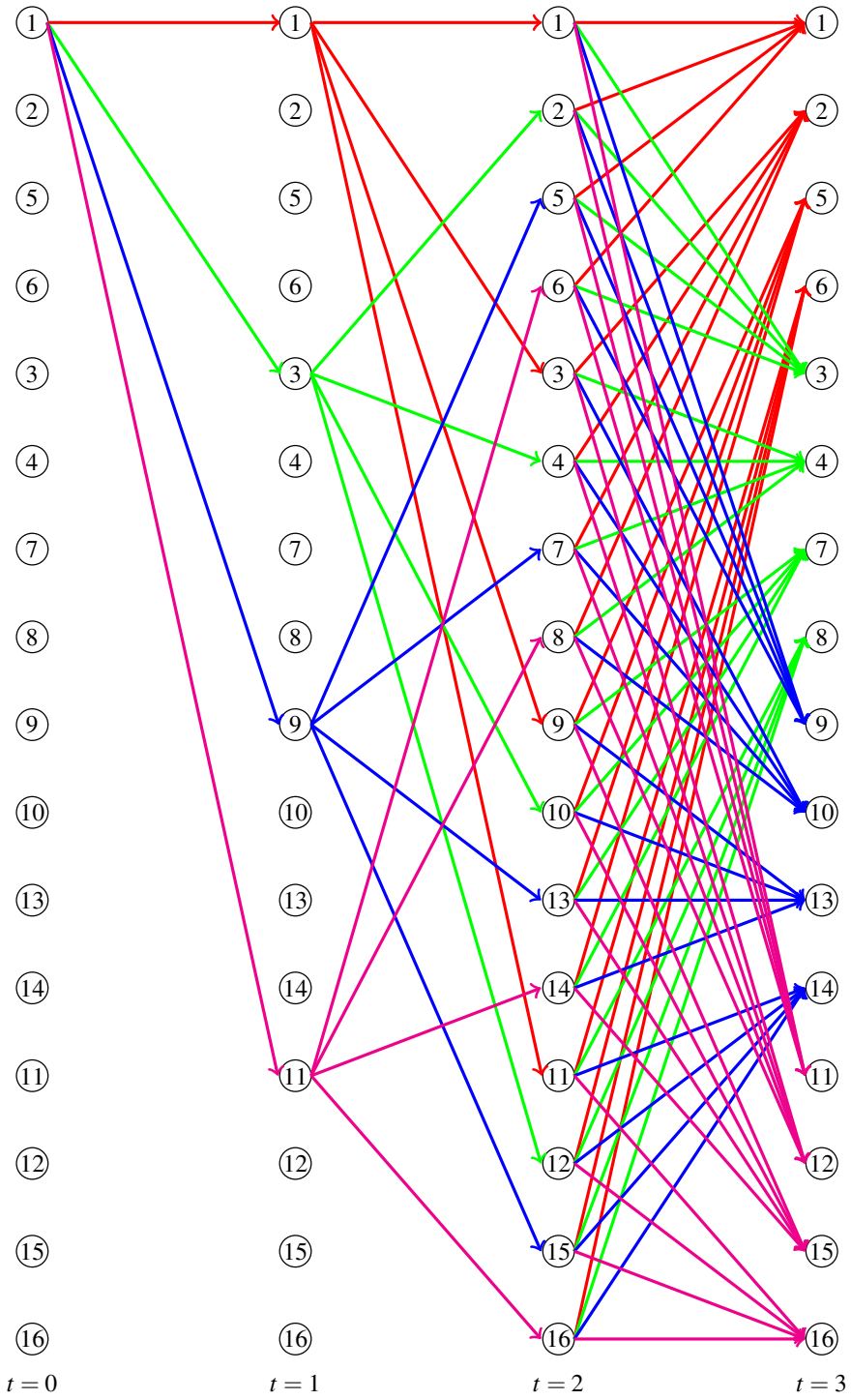

Figure 4.2: The 16-state Trellis of the First Track

each state, the minimum value should be found by adding ITI to Equation 4.2. The possible transmitted sequence, after adding ITI, is [-14-14α -14-6α -14+6α -6+6α ....]. Table 4.1 shows all the possibilities of the first track for the 16-state trellis.

The Path Metrics of The First State of The 16-state Trellis

The first state has four incoming paths which are:

$$st_1 = \left(z_{k1} - (-14 - 14\alpha)\right)^2 + x_1 \tag{4.3}$$

$$st_2 = \left(z_{k1} - (-14 - 6\alpha)\right)^2 + x_2 \tag{4.4}$$

$$st_3 = \left(z_{k1} - (-6 - 14\alpha)\right)^2 + x_3 \tag{4.5}$$

$$st_4 = \left(z_{k1} - (-6 - 6\alpha)\right)^2 + x_4 \tag{4.6}$$

Where x_1, x_2, x_3 and x_4 are the previous minimums. The path metric will be one of Equations (4.3, 4.4, 4.5, 4.6). The same procedure should be followed for the rest of the trellis. However, before adding the amount of ITI, the first state had two inputs (two possibilities) which are:

$$st_1 = \left(z_{k1} - (-14)\right)^2 + x_1 \tag{4.7}$$

$$st_2 = \left(z_{k1} - (-6)\right)^2 + x_2 \tag{4.8}$$

Table 4.1: First Track States Table

4-state Trellis				**16-state Trellis**			
Input	**Initial State**	**Final State**	**Output**	**Input**	**Initial State**	**Final State**	**Output**
-1	-1 -1	-1 -1	-14	(-1-α)	(-1-α) (-1-α)	(-1-α) (-1-α)	(-14-14α)
				(-1-α)	(-1-α) (-1+α)	(-1-α) (-1-α)	(-14-6α)
				(-1-α)	(-1-α) (1-α)	(-1-α) (-1-α)	(-6-14α)
				(-1-α)	(-1-α) (1+α)	(-1-α) (-1-α)	(-6-6α)
				(-1-α)	(-1+α) (-1-α)	(-1-α) (-1+α)	(-14-2α)
				(-1-α)	(-1+α) (-1+α)	(-1-α) (-1+α)	(-14+6α)
				(-1-α)	(-1+α) (1-α)	(-1-α) (-1+α)	(-6-2α)
				(-1-α)	(-1+α) (1+α)	(-1-α) (-1+α)	(-6+6α)
+1	-1 -1	+1 -1	-6	(-1-α)	(1-α) (-1-α)	(-1-α) (1-α)	(-2-14α)
				(-1-α)	(1-α) (-1+α)	(-1-α) (1-α)	(-2-6α)
				(-1-α)	(1-α) (1-α)	(-1-α) (1-α)	(6-14α)
				(-1-α)	(1-α) (1+α)	(-1-α) (1-α)	(6-6α)
				(-1-α)	(1+α) (-1-α)	(-1-α) (1+α)	(-2-2α)
				(-1-α)	(1+α) (-1+α)	(-1-α) (1+α)	(-2+6α)
				(-1-α)	(1+α) (1-α)	(-1-α) (1+α)	(6-2α)
				(-1-α)	(1+α) (1+α)	(-1-α) (1+α)	(6+6α)
-1	+1 -1	-1 +1	-2	(-1+α)	(-1-α) (-1-α)	(-1+α) (-1-α)	(-14-6α)
				(-1+α)	(-1-α) (-1+α)	(-1+α) (-1-α)	(-14+2α)
				(-1+α)	(-1-α) (1-α)	(-1+α) (-1-α)	(-6-6α)
				(-1+α)	(-1-α) (1+α)	(-1+α) (-1-α)	(-6+2α)
				(-1+α)	(-1+α) (-1-α)	(-1+α) (-1+α)	(-14+6α)
				(-1+α)	(-1+α) (-1+α)	(-1+α) (-1+α)	(-14+14α)
				(-1+α)	(-1+α) (1-α)	(-1+α) (-1+α)	(-6+6α)
				(-1+α)	(-1+α) (1+α)	(-1+α) (-1+α)	(-6+14α)
+1	+1 -1	+1 +1	+6	(-1+α)	(1-α) (-1-α)	(-1+α) (1-α)	(-2-6α)
				(-1+α)	(1-α) (-1+α)	(-1+α) (1-α)	(-2+2α)
				(-1+α)	(1-α) (1-α)	(-1+α) (1-α)	(6-6α)
				(-1+α)	(1-α) (1+α)	(-1+α) (1-α)	(6+2α)
				(-1+α)	(1+α) (-1-α)	(-1+α) (1+α)	(-2+6α)
				(-1+α)	(1+α) (-1+α)	(-1+α) (1+α)	(-2+14α)
				(-1+α)	(1+α) (1-α)	(-1+α) (1+α)	(6+6α)
				(-1+α)	(1+α) (1+α)	(-1+α) (1+α)	(6+14α)
-1	-1 +1	-1 -1	-6	(1-α)	(-1-α) (-1-α)	(1-α) (-1-α)	(-6-14α)
				(1-α)	(-1-α) (-1+α)	(1-α) (-1-α)	(-6-6α)
				(1-α)	(-1-α) (1-α)	(1-α) (-1-α)	(2-14α)
				(1-α)	(-1-α) (1+α)	(1-α) (-1-α)	(2-6α)
				(1-α)	(-1+α) (-1-α)	(1-α) (-1+α)	(-6-2α)
				(1-α)	(-1+α) (-1+α)	(1-α) (-1+α)	(-6+6α)
				(1-α)	(-1+α) (1-α)	(1-α) (-1+α)	(2-2α)
				(1-α)	(-1+α) (1+α)	(1-α) (-1+α)	(2+6α)
+1	-1 +1	+1 -1	+2	(1-α)	(1-α) (-1-α)	(1-α) (1-α)	(6-14α)
				(1-α)	(1-α) (-1+α)	(1-α) (1-α)	(6-6α)
				(1-α)	(1-α) (1-α)	(1-α) (1-α)	(14-14α)
				(1-α)	(1-α) (1+α)	(1-α) (1-α)	(14-6α)
				(1-α)	(1+α) (-1-α)	(1-α) (1+α)	(6-2α)
				(1-α)	(1+α) (-1+α)	(1-α) (1+α)	(6+6α)
				(1-α)	(1+α) (1-α)	(1-α) (1+α)	(14-2α)
				(1-α)	(1+α) (1+α)	(1-α) (1+α)	(14+6α)
-1	+1 +1	-1 +1	+6	(1+α)	(-1-α) (-1-α)	(1+α) (-1-α)	(-6-6α)
				(1+α)	(-1-α) (-1+α)	(1+α) (-1-α)	(-6+2α)
				(1+α)	(-1-α) (1-α)	(1+α) (-1-α)	(2-6α)
				(1+α)	(-1-α) (1+α)	(1+α) (-1-α)	(2+2α)
				(1+α)	(-1+α) (-1-α)	(1+α) (-1+α)	(-6+6α)
				(1+α)	(-1+α) (-1+α)	(1+α) (-1+α)	(-6+14α)
				(1+α)	(-1+α) (1-α)	(1+α) (-1+α)	(2+6α)
				(1+α)	(-1+α) (1+α)	(1+α) (-1+α)	(2+14α)
+1	+1 +1	+1 +1	+14	(1+α)	(1-α) (-1-α)	(1+α) (1-α)	(6-6α)
				(1+α)	(1-α) (-1+α)	(1+α) (1-α)	(6+2α)
				(1+α)	(1-α) (1-α)	(1+α) (1-α)	(14-6α)
				(1+α)	(1-α) (1+α)	(1+α) (1-α)	(14+2α)
				(1+α)	(1+α) (-1-α)	(1+α) (1+α)	(6+6α)
				(1+α)	(1+α) (-1+α)	(1+α) (1+α)	(6+14α)
				(1+α)	(1+α) (1-α)	(1+α) (1+α)	(14+6α)
				(1+α)	(1+α) (1+α)	(1+α) (1+α)	(14+14α)

4.5 Decoding Techniques

Different decoding techniques have been found in order to calculate the path metrics of 16-state trellis which are:

1- Without Simplification

Finding the path metric by leaving Equations (4.3, 4.4, 4.5) and (4.6) as they are without simplification so that the minimum will be one of the following:

$$min\Big((z_{k_1}-(-14-14\alpha))^2+x_1,(z_{k_1}-(-14-6\alpha))^2+x_2, \\ (z_{k_1}-(-6-14\alpha))^2+x_3,(z_{k_1}-(-6-6\alpha))^2+x_4\Big) \tag{4.9}$$

2- Maximum Values of αz_k and α Terms

Simplifying Equations (4.3, 4.4, 4.5) and (4.6) and group them into two groups and then taking the maximum values of αz_k and α terms of each group as follows:

$$\Big(z_{k1}-(-14-14\alpha)\Big)^2+x_1=28z_{k1}+28\alpha z_{k1}+196+392\alpha+x_1 \tag{4.10}$$

$$\Big(z_{k1}-(-14-6\alpha)\Big)^2+x_2=28z_{k1}+12\alpha z_{k1}+196+168\alpha+x_2 \tag{4.11}$$

$$\Big(z_{k1}-(-6-14\alpha)\Big)^2+x_3=12z_{k1}+28\alpha z_{k1}+36+168\alpha+x_3 \tag{4.12}$$

$$\Big(z_{k1}-(-6-6\alpha)\Big)^2+x_4=12z_{k1}+12\alpha z_{k1}+36+72\alpha+x_4 \tag{4.13}$$

The first group is Equations (4.10) and (4.11). Thus, the minimum of these equations is approximated to:

$$\min(x_1, x_2) + 28z_{k1} + 28\alpha z_{k1} + 196 + 392\alpha \tag{4.14}$$

And the second group is the Equations of (4.12) and (4.13). The approximation of the minimum is:

$$\min(x_3, x_4) + 12z_{k1} + 28\alpha z_{k1} + 36 + 168\alpha \tag{4.15}$$

3- Minimum Values of αz_k and α Terms

Instead of using the maximum values of αz_k and α terms, the minimums of those terms are taken and then the same procedure would be followed to find the state metric of the first state:

$$\min(x_1, x_2) + 28z_{k1} + 12\alpha z_{k1} + 196 + 168\alpha \tag{4.16}$$

$$\min(x_3, x_4) + 12z_{k1} + 12\alpha z_{k1} + 36 + 72\alpha \tag{4.17}$$

4- Averaging αz_k and α Terms

Finding the approximated minimum by taking the average of αz_k and α terms. The equation of the first group is:

$$\min(x_1, x_2) + 28z_{k1} + 20\alpha z_{k1} + 196 + 280\alpha \tag{4.18}$$

And of the second group is:

$$\min(x_3, x_4) + 12z_{k1} + 20\alpha z_{k1} + 36 + 120\alpha \tag{4.19}$$

So that the approximation value of the minimum is one of them.

5- One Common Factor

By taking $(20\alpha z_{k1})$ term out as a common factor, the approximated minimum will be one of the following two equations plus that common factor.

$$\min(x_1, x_2) + 28z_{k1} + 196 + 280\alpha \tag{4.20}$$

$$\min(x_3, x_4) + 12z_{k1} + 36 + 120\alpha \tag{4.21}$$

6- Two Common Factors

Finally, by averaging α terms

$$\min(x_1, x_2) + 28z_{k1} + 20\alpha z_{k1} + 196 + 200\alpha \tag{4.22}$$

$$\min(x_3, x_4) + 12z_{k1} + 20\alpha z_{k1} + 36 + 200\alpha \tag{4.23}$$

Then taking two common factors out which are $(20\alpha z_{k1})$ and (200α) to yield

$$\min(x_1, x_2) + 28z_{k1} + 196 \tag{4.24}$$

$$\min(x_3, x_4) + 12z_{k1} + 36 \tag{4.25}$$

The approximated minimum is one of the above equations plus the two common factors.

4.6 Finding the Complexity of the Decoding Techniques

The previous section has discussed different techniques of grouping the state metric equations for each state of 16-state trellis, where adding an amount of crosstalk would require construction of a new trellis to accomplish all the possibilities of the channel. Several

performance metrics have been obtained with different complexities. The performances graphs of those techniques will be discussed in the next section. The following is finding the complexity of each decoding technique per decoded bit which equals the complexity of each state times sixteen:

- The complexity of *without simplification* technique, Equation (4.9), is 320 additions , 192 multiplications and 48 comparisons.
- The complexity of the second technique which is *maximum values of* z_k *and* αs *terms*, *minimum values of* z_k *and* αs *terms* and *averaging* $z_k s$ *and* αs *terms* techniques, Equations (4.14), (4.15), (4.16), (4.17), (4.18) and (4.19), is 128 additions, 128 multiplications and 16 comparisons.
- The complexity of *one common factor* technique, Equation (4.20) and Equation (4.21), is 112 additions, 96 multiplications and 16 comparisons. However this complexity will be re-calculated in detail in the next section.
- The full complexity of *two common factors* technique, Equation (4.24) and Equation (4.25), is 96 additions, 80 multiplications and 16 comparisons.

Table 4.2 summarizes the complexity of the six decoding techniques.

The minimum operations of all those equations which are $min(x_1, x_2)$ and $min(x_3, x_4)$ involve the previous state metric and does not involve the current sample of the received sequence z_k. Which means that before getting the new sample of z_k, that minimum operation can be completed off-line as it is independent of the current sample of z_k sequence. In other words, while the received sequence is being equalised this comparison can be completed. Consequently, the minimum operations will not be a part of this complexity. Also the α

Table 4.2: Complexity of The Decoding Techniques

Technique	Addition	Multiplication	Comparison
Without Simplification	320	192	48
The Second Technique	128	128	16
One Common Factor	112	96	16
Two Common Factors	96	80	16

terms will not be a part of this complexity as the amount of ITI is known. Therefore this complexity can be reduced further.

4.7 ITI System Performance Discussion

This section discusses the simulation results of ITI system model.

Figure 4.3 shows the BER performance for the PRML system (track 1 and track 2 MLSD New Scheme graphs) when compared with one-track model graph with zero ITI (MLSD Old Scheme). No ITI means that the noise that is added to that system is pure AWGN. At 6 dB SNR, the BER is almost 10^{-3}.

Figure 4.4 shows the BER performance when ITI is treated as a noise at 6 dB SNR. In this graph the noise is increased by adding an amount of ITI which has a significant effect on the BER, BER exceeded 10^{-1} for the higher amount of ITI (beyond 0.4) that used in this simulation.

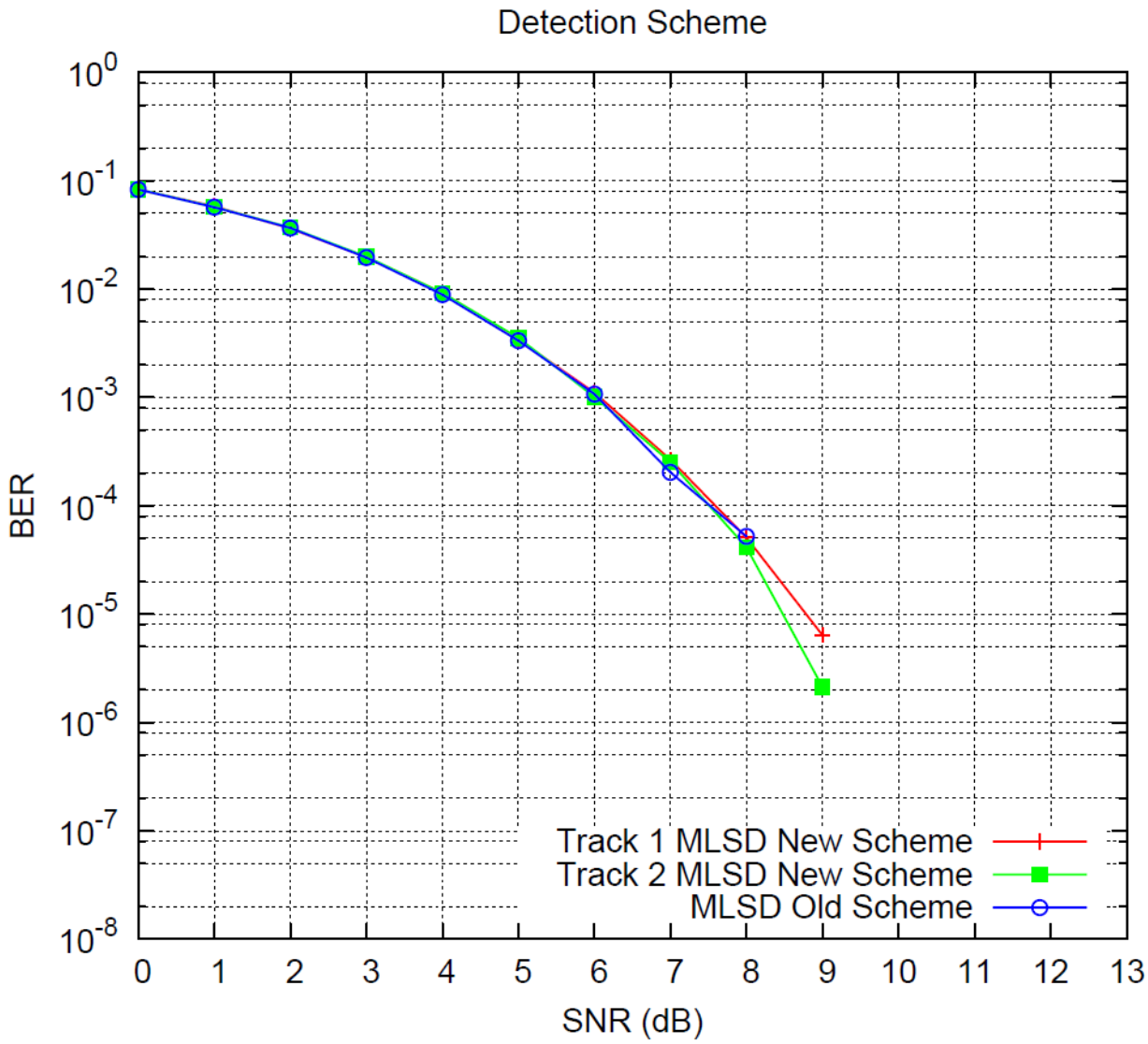

Figure 4.3: BER Performance for PRML System With no ITI

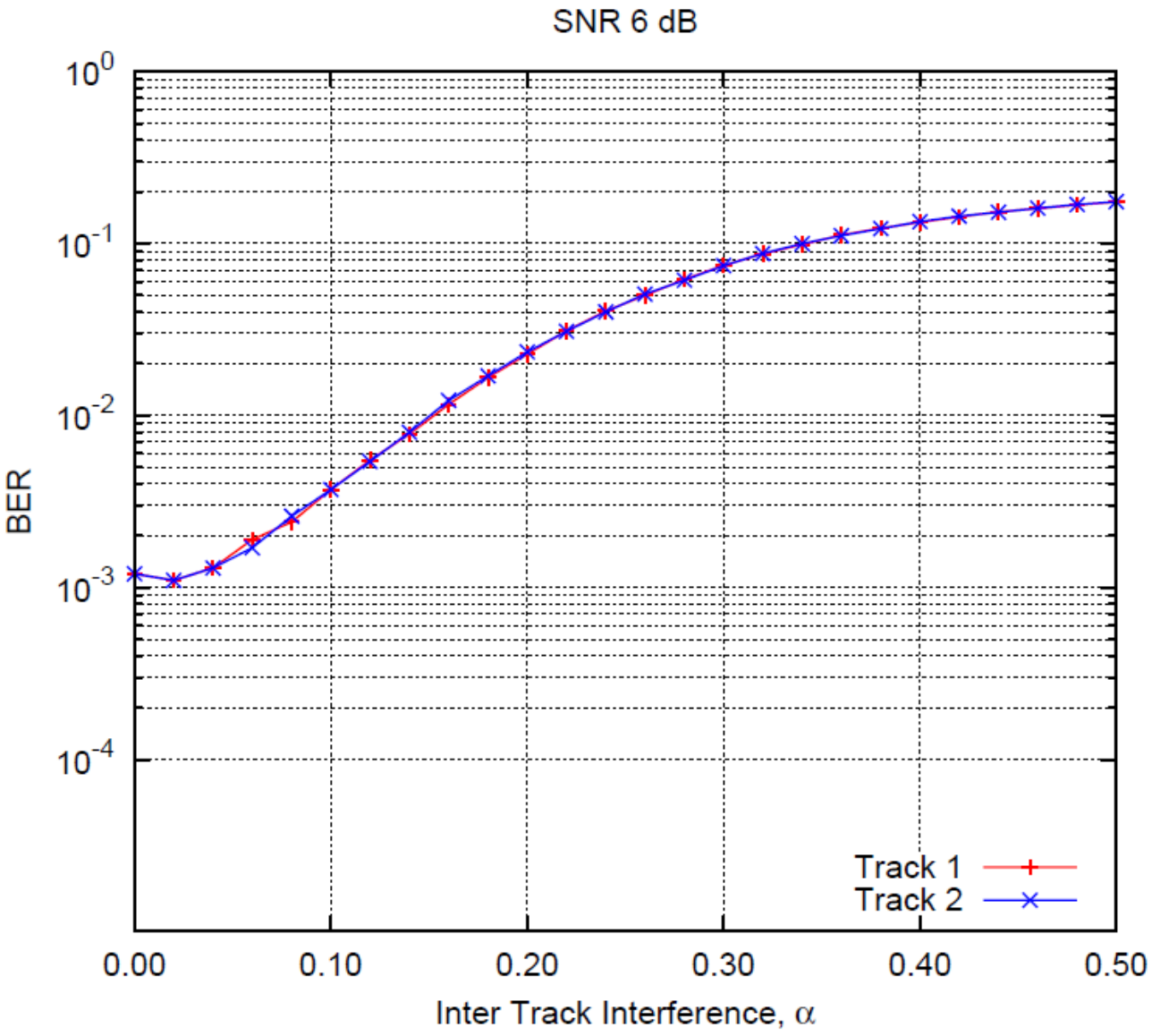

Figure 4.4: BER Performance For PRML System Where ITI is Treated as a Noise

BER performance for the decoding techniques are shown in Figure 4.5 at 6 dB SNR and in Figure 4.6 for 7.5 dB SNR. The performances could be grouped into three bands. The first band is the maximum and minimum values of z_k and αs terms techniques. The second band is two common factors technique. While the third band includes averaging of z_k and αs terms and one common factor techniques which they had the same performance. However, the optimal performance is the performance of without simplification technique. The graphs show that a huge difference in the performance of the first and the second bands compared with the optimal one. The third band has the closest performance to the optimal one where the difference between the two performances (conventional and one common factor methods) becomes larger at higher amount of ITI (more than 30%).

In terms of complexity, the six decoding techniques can be grouped into four bands as shown in Table 4.2. The first band has a huge complexity compared with the other techniques complexity. The second band which is the second technique approximation has less complexity compared with the first band. The one common factor approximation has the lower complexity. However, the last band (two common factor approximation) has the least complexity but its performance is very far from the optimal one.

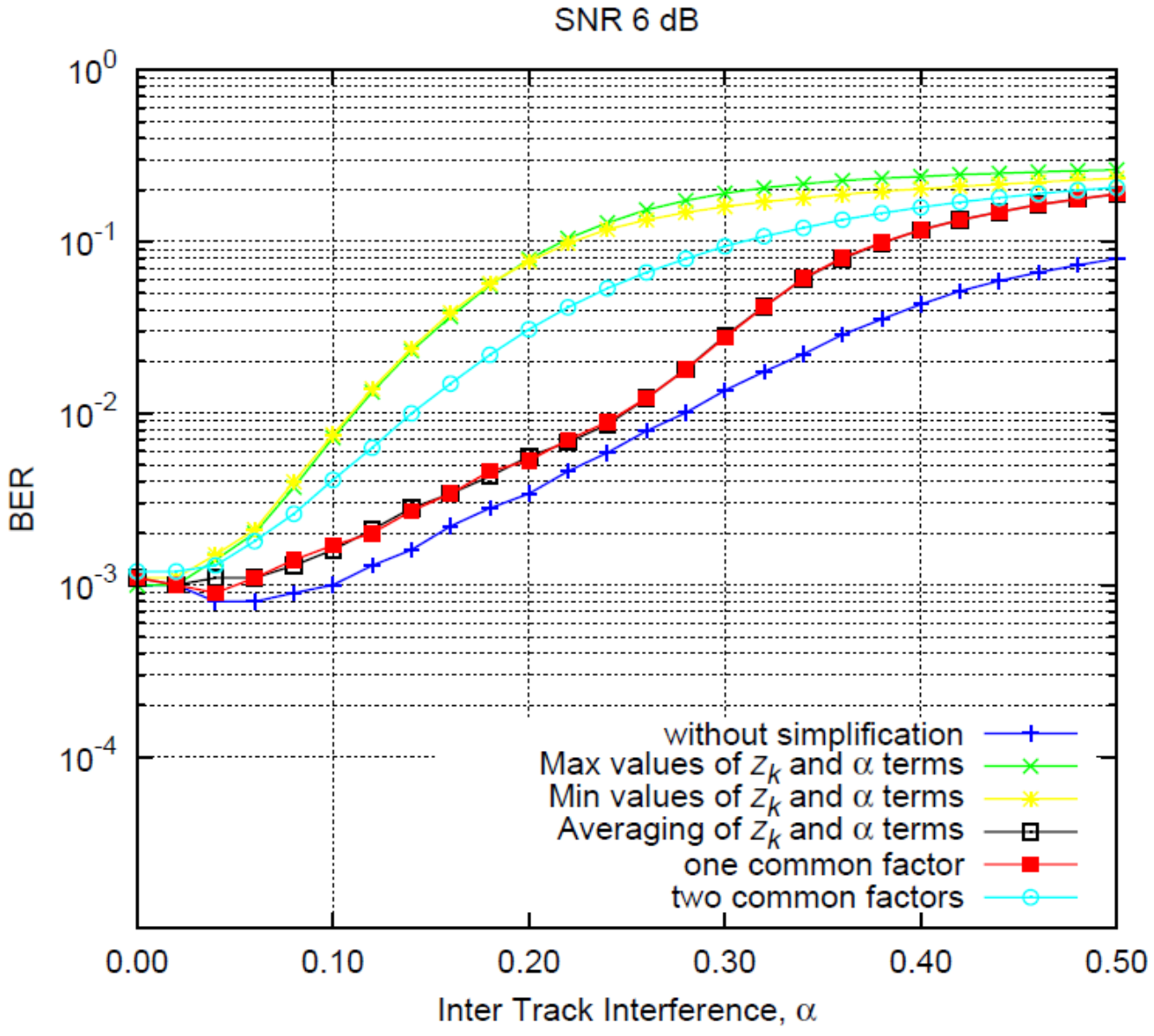

Figure 4.5: BER Performance for Shingled System at 6 dB SNR

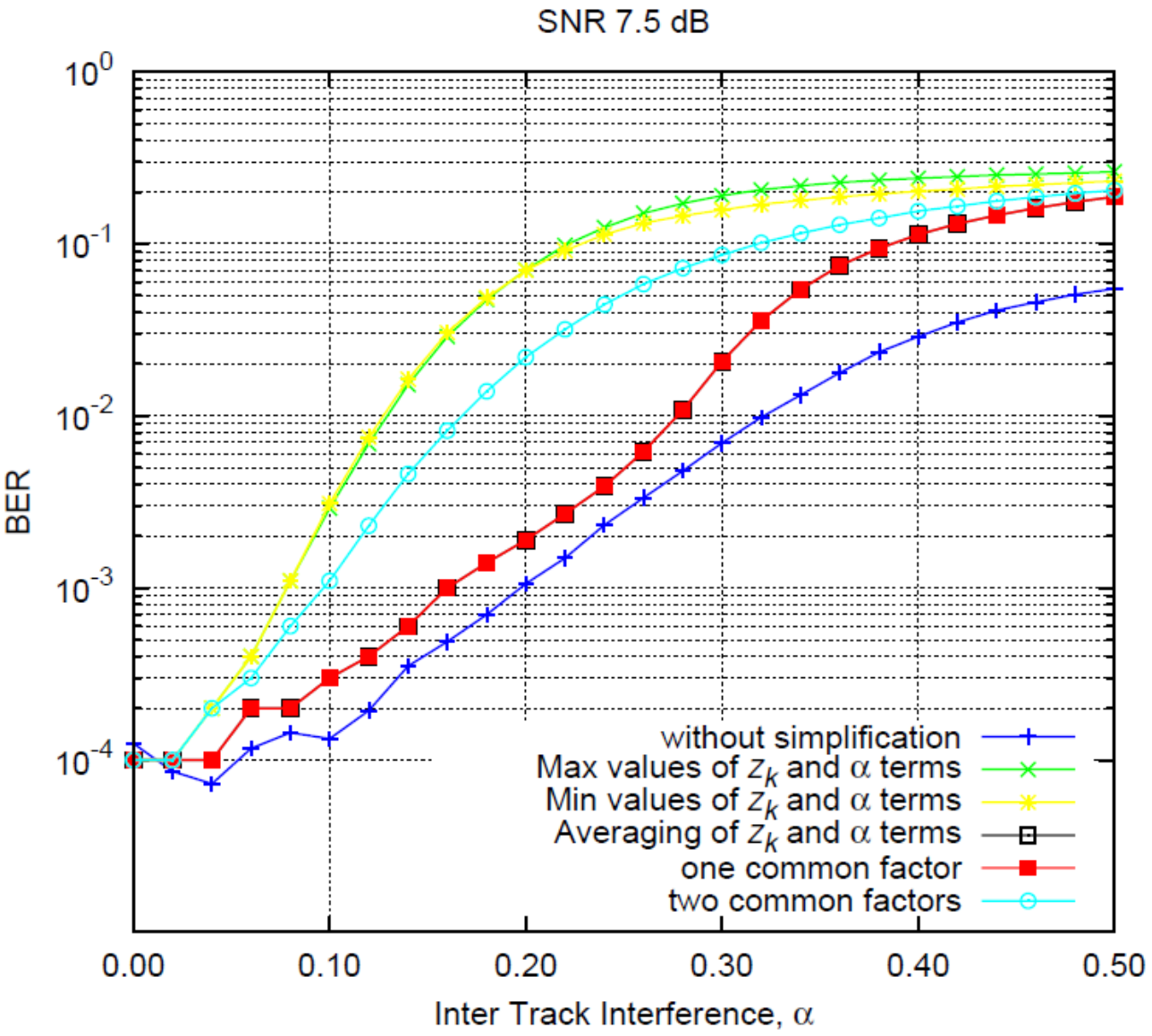

Figure 4.6: BER Performance for Shingled System at 7.5 dB SNR

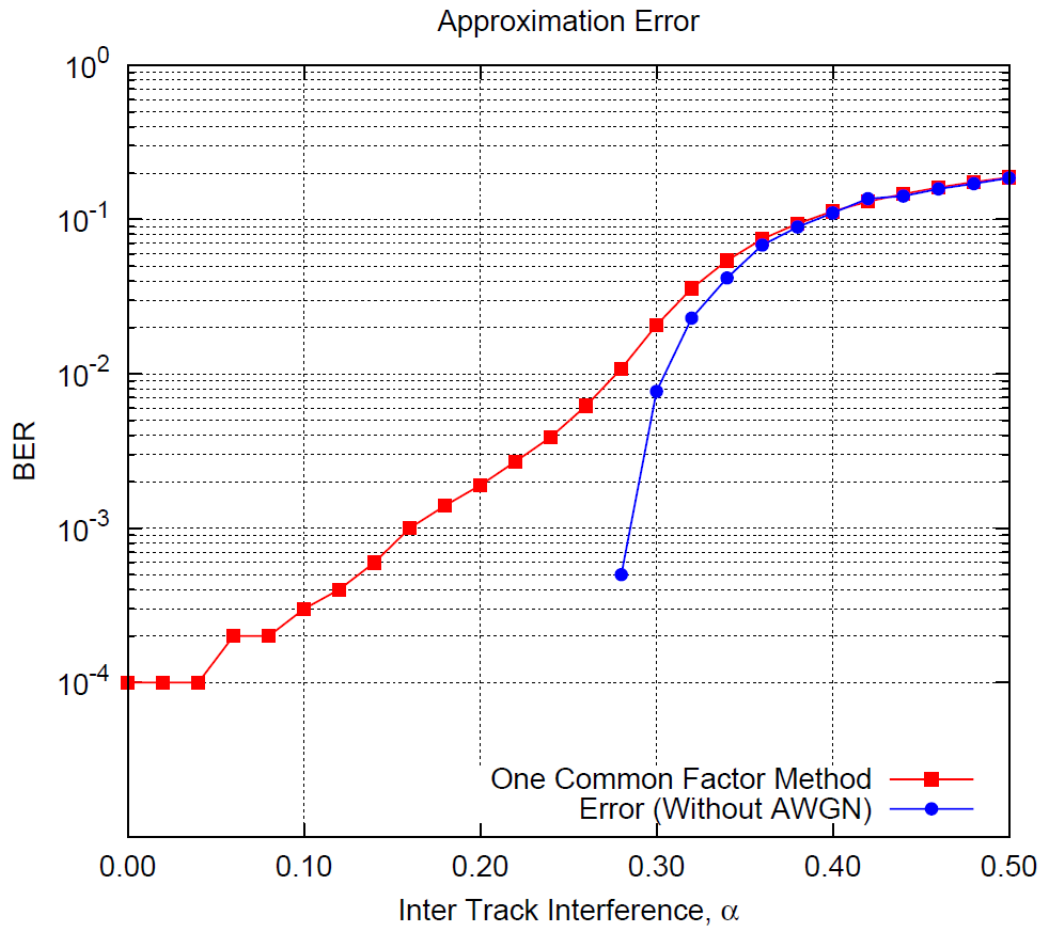

Figure 4.7: Performance of One Common Factor Technique With and Without AWGN

Figure 4.7 shows the performance of the approximation (one common factor) that had a closest performance to the optimal one with AWGN at SNR 7.5 dB and without AWGN. The error graph shows the performance of this approximation without AWGN i.e. the noise is just ITI. It shows that the ITI has a significant effect as a dominant noise for an amount less than 30%, however for higher amount than this the ITI has no significant effect.

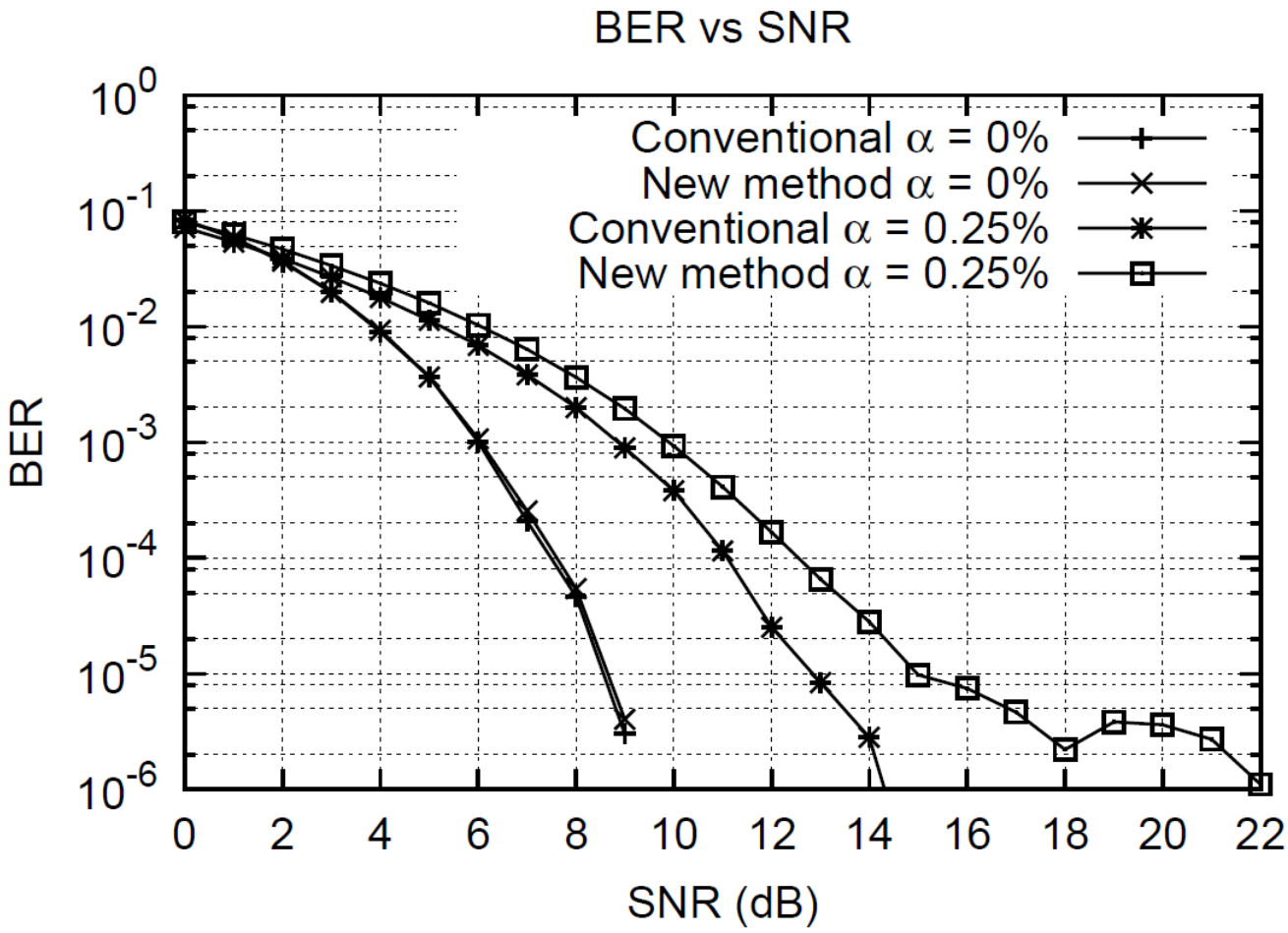

Figure 4.8: BER vs SNR for Different Amount of ITI

Figure 4.8 discusses the effect of the ITI for the conventional method and one common factor approximation. The purpose of this simulation is to demonstrate the effect of adding ITI as signal coming from the adjacent track. It shows the BER performance of these two techniques with no ITI and at 25% ITI. There is no difference in performance with no ITI. However, with 25% ITI, there is a 0.5 dB SNR loss. For high SNR, the new method has an error floor around a BER of 10^{-6}.

Figure 4.9 compares BER performance of two techniques which are conventional and

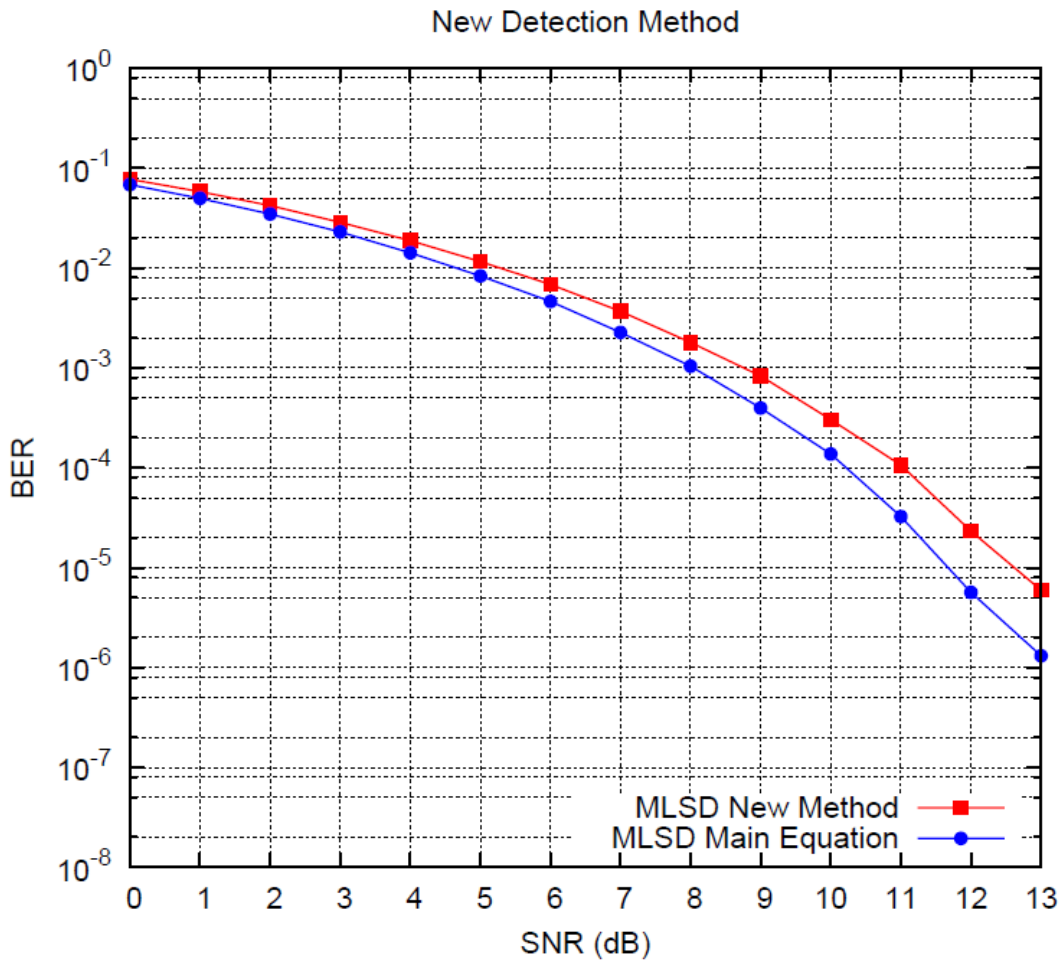

Figure 4.9: BER Performance Comparison

new method techniques at 22% ITI. The figure shows that the loss in performance is about 0.5 dB.

4.8 Performance Evaluation

Because the performance of one common factor technique is the closest to the optimal one, this section explains in detail finding the minimum Euclidean distances and the complexity of this technique. Then, a comparison will be made with the conventional technique.

Finding The State Metrics

The four state metrics of the first node of the trellis at ($t = 3$) are:

The state metric of the first path is

$$st_1 = \left(z_{k1} - (-14 - 14\alpha) \right)^2 + x_1 \tag{4.26}$$

$$= \left(z_{k1} + (14 + 14\alpha) \right)^2 + x_1 \tag{4.27}$$

$$= z_{k1}(z_{k1} + (14 + 14\alpha)) + (14 + 14\alpha)(z_{k1} + (14 + 14\alpha)) + x_1 \tag{4.28}$$

$$= z_{k1}^2 + 14z_{k1} + 14\alpha z_{k1} + 14z_{k1} + 14\alpha z_{k1} + ((14 + 14\alpha))^2 + x_1 \tag{4.29}$$

$$= z_{k1}^2 + 28z_{k1} + 28\alpha z_{k1} + 196\alpha^2 + 392\alpha + 196 + x_1 \tag{4.30}$$

Where, z_{k1}^2 and $196\alpha^2$ have been ignored for the simplification purposes. Therefore the equation of the first path is

$$st_1 = 28z_{k1} + 28\alpha z_{k1} + 196 + 392\alpha + x_1 \tag{4.31}$$

The same simplifications are made for the other paths of the first state of the trellis.

For the second path, the state metric is

$$st_2 = 28z_{k1} + 12\alpha z_{k1} + 196 + 168\alpha + x_2 \tag{4.32}$$

The state metric of the third path is

$$st_3 = 12z_{k1} + 28\alpha z_{k1} + 36 + 168\alpha + x_3 \tag{4.33}$$

And the state metric of the fourth path is

$$st_4 = 12z_{k1} + 12\alpha z_{k1} + 36 + 72\alpha + x_4 \tag{4.34}$$

Now to find the minimum of the this node, the four equations above are grouped into two groups. Therefore, the minimum of the first group st_1 and st_2 is

$$\min\left(\begin{array}{l} 28z_{k1} + 28\alpha z_{k1} + 196 + 329\alpha + x_1, \\ \qquad 28z_{k1} + 12\alpha z_{k1} + 196 + 168\alpha + x_2 \end{array}\right) \tag{4.35}$$

And then the average of αz_k and α terms are taken as follows:

The average of $28\alpha z_{k1}$ and $12\alpha z_{k1}$ is $20\alpha z_{k1}$.

The average of 392α and 168α is 280α.

Thus,

$$\min\left(\begin{array}{l} 28z_{k1} + 20\alpha z_{k1} + 196 + 280\alpha + x_1, \\ \qquad 28z_{k1} + 20\alpha z_{k1} + 196 + 280\alpha + x_2 \end{array}\right) \tag{4.36}$$

Both equations have common terms therefore the minimum could be approximated into

$$\min(x_1, x_2) + 28z_{k1} + 20\alpha z_{k1} + 196 + 280\alpha \tag{4.37}$$

And for the second group st_3 and st_4, the minimum is

$$\min\Big(12z_{k1} + 28\alpha z_{k1} + 36 + 168\alpha + x_3, \; 12z_{k1} + 12\alpha z_{k1} + 36 + 72\alpha + x_4\Big) \tag{4.38}$$

$$\min(x_3, x_4) + 12z_{k1} + 20\alpha z_{k1} + 36 + 120\alpha \tag{4.39}$$

So that the minimum of the first node is the minimum value of the following two equations plus a common factor which is in this case $(20\alpha z_{k1})$:

$$\min(x_1, x_2) + 28z_{k1} + 196 + 280\alpha \tag{4.40}$$

$$\min(x_3, x_4) + 12z_{k1} + 36 + 120\alpha \tag{4.41}$$

Finding The Complexity

The complexity of this technique is found as follows:

- The complexity of Equation (4.40) and Equation (4.41) is three additions and two multiplications for each equation. However, $\min(x_1, x_2)$ and $\min(x_3, x_4)$ would be pre computed as they are the minimum values of the previous states.
- One comparison operation to find the minimum value of Equations (4.40) and (4.41).
- The common factor $(20\alpha z_{k1})$ adds one addition operation and two multiplications for

Table 4.3: Complexity Comparison of Two Techniques

	The Conventional Technique	**The New Technique**
Addition	320	112
Multiplication	192	96
Comparison	48	16

each state. Therefore per node the complexity is six additions, four multiplications and one comparison.

- Per decoded bit the complexity will be the complexity per node times sixteen. It is 112 additions, 96 multiplications and 16 comparisons.

However, the complexity of the conventional technique is:

- Each path has a complexity of five additions, three multiplications. Therefore, the complexity is twenty additions and twelve multiplications for the four paths.
- Three comparison operations to find the minimum value of the state.
- Per each state, the complexity would be twenty additions, twelve multiplications and three comparison.
- Per decoded bit, the complexity is the complexity per node times 16 which is 320 additions, 192 multiplications and 48 comparisons.

Table 4.3 shows the complexity of the two techniques per decoded bit.

Off-line and On-line Complexities

Many terms of Equations (4.40) and (4.41) can be completed off-line such as the comparison operations and the constant values. The minimisation function $\min(x_1, x_2)$ and $\min(x_3, x_4)$ can also be completed off-line as it depends on the previous value of z_k rather

than the present value. The other values such as 280α and the constant (196) can be pre computed as the ITI amount has been assumed known. So that, Equation (4.40) and Equation (4.41) can be reduced into

$$\min(x_1, x_2) + 196 + 280\alpha \tag{4.42}$$

$$\min(x_3, x_4) + 36 + 120\alpha \tag{4.43}$$

The off-line (after sampling) complexity is:

- Four additions, two multiplications and two comparisons for each state.
- 64 additions, 32 multiplications and 32 comparisons for each time interval.

The On-line complexity is:

- One addition and one multiplication times two for each node which is two additions and two multiplications plus one addition and two multiplications of the common factor and one comparison to find the minimum value.

  Therefore, per node the complexity is three additions and four multiplications and one comparison.

- In terms of the decoded bit, the complexity that can be completed on-line is three additions and four multiplications times 16.

  Per bit, the complexity is 48 additions, 64 multiplications and 16 comparisons.

On-line and off-line complexities are depicted in Table 4.4.

Table 4.4: Complexity of One Common Factor Technique

	On-line	**Off-line**
Addition	48	64
Multiplication	64	32
Comparison	16	32

Efficiency of One Common Factor Approximation

In order to compare the performance of the new technique to the optimum one, the penalty of this method is found as

$$P = \frac{BER_1}{BER_2} \tag{4.44}$$

where, BER_1 is the BER of the new method and BER_2 is the BER value of the optimum method. Figure 4.10 shows the penalty of the new technique that the system would incur. At higher SNR, the penalty becomes a significant value.

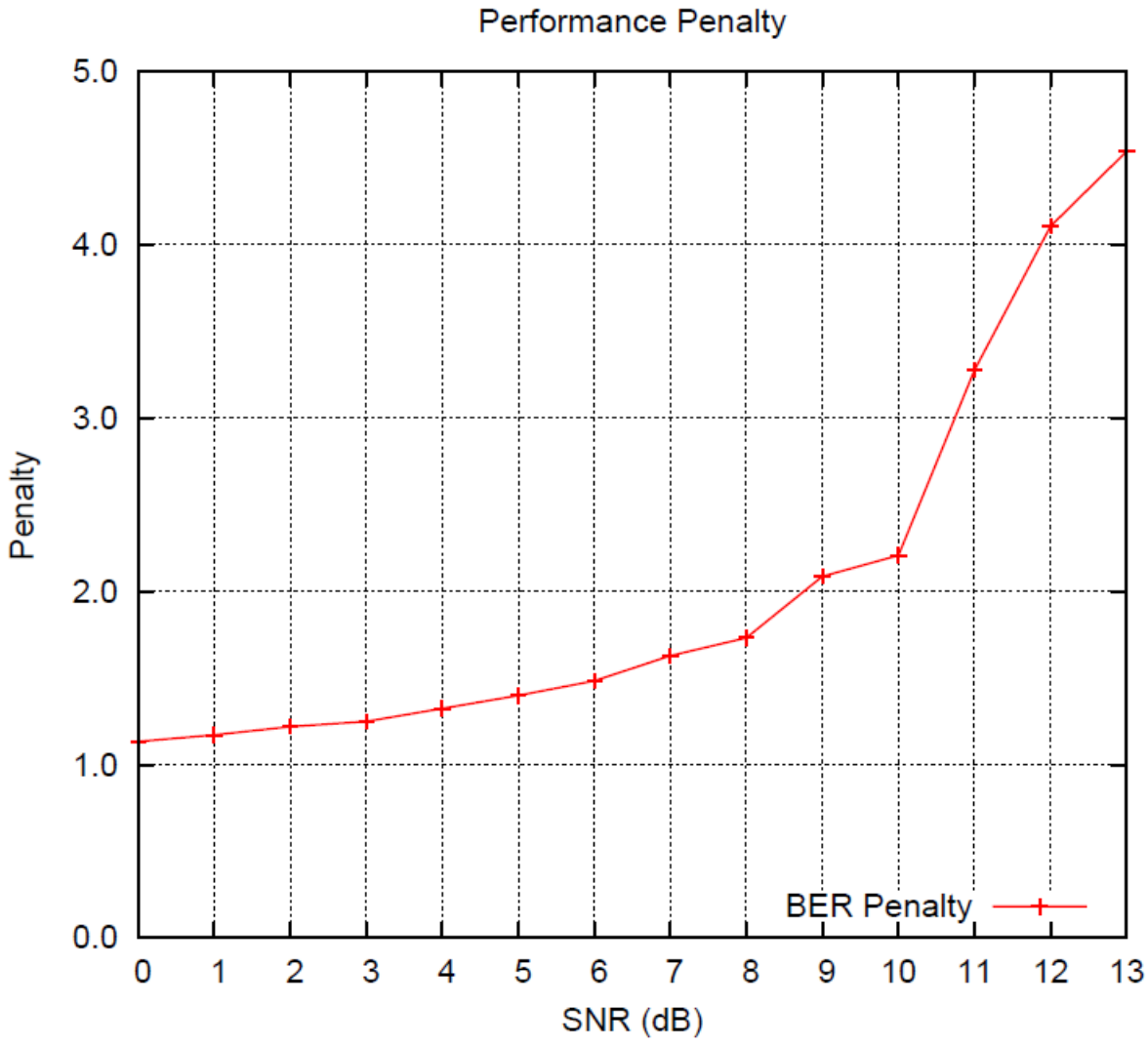

Figure 4.10: The Performance Penalty of One Common Factor Approximation

4.9 Summary

This chapter can be summarized as follows:

- One-track one-head model system has been extended to include two tracks with known amount of ITI and then is implemented in MATLAB.
- The ITI is treated as a noise then as a signal.
- For detection purpose, a 16-state trellis has been constructed to meet the equaliser output sequences.
- Six techniques are developed to find the Euclidean distances of the system trellis.
- The complexity has been calculated for each technique.
- The first one, without simplification technique, has better performance but its complexity is very high.
- The one common factor technique has a very close performance to the optimal one with low complexity.
- The complexity of the two techniques have been compared where the tables and the graphs show that the complexity could be reduced greater than three times with 0.5 dB penalty.
- Because some computations of the new method technique can be completed off-line, its complexity has been reduced further.
- The shingled system performance graphs show that a trade off can be made between the BER and complexity as shown in Figure 4.9.

Chapter 5

Introduction to FPGA Design Implementation

5.1 Introduction

This chapter presents an introduction to the simulation software's that are used to implement PRML system in hardware. The used softwares are Simulink of MathWorks, ModelSim and Altera Quartus II. Then Altera DE1 development and education board is used in order to verify the design model. Also, some features of STM32F4 High-Performance Discovery board which will be used to convert the digital output data of FPGA chip into analog signal.

5.2 FPGAs and DSP Processors

Recently, several applications have been developed very rapidly as a result of expansion of growing the DSP market among them are: 3G wireless, radar and satellite systems, image-processing applications and medical systems [99]. Most of these applications are implemented by DSP processors. DSP processors are programmable devices through software. However, their hardware architectures are not flexible i.e. fixed hardware. Fixed hardware architectures such as fixed number of multiply accumulate (MAC) blocks, fixed memory and fixed data width could limit the DSP processors usage. Therefore, some applications

cannot be obtained by DSP processor which in turn requires customized DSP function implementations. An eligible solution for the DSP processors limitations is FPGAs. FPGAs could be reconfigurable hardware with higher DSP throughput and data processing than DSP processors. In addition, FPGAs offer hardware customization for DSP applications implementations. Therefore, DSP systems that are implemented in FPGAs can have a customized architecture, customized memory, customized bus structure, and a variable number of MAC blocks.

Altera Cyclone II device is used in this work to implement the PRML system model. This device has the following features [1]:

1. High-density architecture with 4608 to 68416 Logic Elements (LEs).
2. Embedded multipliers: up to 150 18 $\times$ 18-bit multipliers each configurable as two independent 9 $\times$ 9-bit multipliers with up to 250-MHz performance.
3. Flexible clock management circuitry: up to four phase-locked loops (PLLs) per device provide clock multiplication and division, phase shifting, programmable duty cycle, and external clock outputs, allowing system-level clock management and skew control
4. Device Configuration: Fast serial configuration allows configuration times less than 100 ms and it supports multiple voltages (either 3.3, 2.5, or 1.8 V)

The Cyclone II EP2C20 device features are listed in Table 5.1 [1].

Cyclone II device family have embedded multiplier blocks combined with programmable logic devices (PLDs) which help the designer to implement some varied cost DSP functions effectively [1]. In addition to that, the embedded multiplier blocks are optimized for multiplier-intensive DSP functions such as FIR filters, fast Fourier transform functions and

Table 5.1: Cyclone II EP2C20 Device Features [1]

LEs	M4K RAM Blocks	Total RAM Bits	PLLs	Maximum User I/O Pins
18,752	52	239,616	4	315

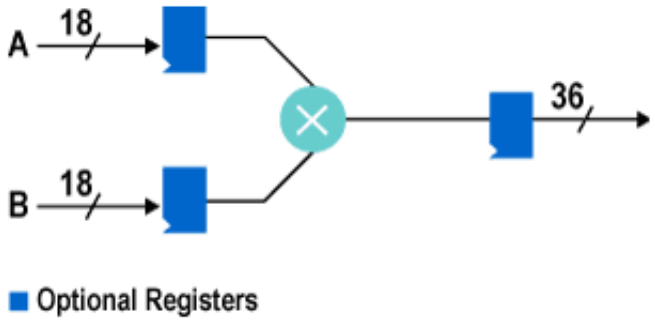

Figure 5.1: Embedded Multipliers in Cyclone II Devices [18]

discrete cosine transform functions. The embedded multiplier is configured as one 18-bit $\times$ 18-bit multiplier or two 9-bit $\times$ 9-bit multipliers as shown in Figure 5.1. For other multiplier implementations, Cyclone II devices also have M4K memory blocks. The combination of two blocks, embedded multiplier blocks and M4K memory blocks, will increase number of multipliers in Cyclone II chips and that would enable the user to implement the design with a wide range of implementation options and more flexibility [1]. Each Cyclone II device has one to three columns of embedded multipliers for functions implementation purposes. Table 5.2 depicts the number of embedded multipliers for different Cyclone II devices families. However, Cyclone II devices also have soft multipliers which are implemented by using Cyclone II M4K memory blocks. The soft multipliers would increase the number of multipliers within the device itself which in turn help the user to implement a wide range of designs. The total number of multipliers including embedded multipliers and soft multipliers is shown in Table 5.3.

Table 5.2: Number of Embedded Multipliers in Cyclone II Devices [1]

Device	**Embedded Multipliers**	**9×9 Multipliers**	**18×18 Multipliers**
EP2C5	13	26	13
EP2C8	18	36	18
EP2C20	26	52	26
EP2C35	35	70	35
EP2C50	86	172	86
EP2C70	150	300	150

Table 5.3: Number of Multipliers in Cyclone II Devices [1]

Device	**Embedded Multipliers (18×18)**	**Soft Multipliers (16×16)**	**Total Multipliers**
EP2C5	13	26	39
EP2C8	18	36	54
EP2C20	26	52	78
EP2C35	35	105	140
EP2C50	86	129	215
EP2C70	150	250	400

In this work, EP2C20 device family is used which has 26 embedded multipliers and 52 soft multipliers so that the total number of multipliers is 78 as indicated in Table 5.2 and Table 5.3 [1].

5.3 STM32F4 High-Performance Discovery Board

The low-cost STM32F4DISCOVERY board is used in this work to visualise the output signal of the designed model by connecting STM32F4 board into the oscilloscope. STM32F4 high-performance chip includes an ST-LINK/V2 embedded debug tool interface, ST Micro Electro-Mechanical Systems (MEMS) digital accelerometer, ST MEMS digital microphone, audio DAC with integrated class D speaker driver, LEDs, push-buttons and a USB OTG micro-AB connector as shown in Figure 5.2. The STM32F4 Digital-to-Analog Converter (DAC) is used in this work to convert the digital output signal of the implemented model into analog signal. The chip could be powered up by the PC through USB cable or an external 5V power supply. However, the chip has 5V and 3V which can be used as output power supplies for another application board [19].

5.4 DSP Builder Software

DSP Builder is a DSP development tool that links MathWorks MATLAB (Signal Processing Toolbox and Filter Design Toolbox) and Simulink software with Altera Quartus II software [100]. DSP Builder generates the hardware as a Very High Speed Integrated Circuits (VHSIC) Hardware Description Language (VHDL) script that integrates with ModelSim and Quartus II softwares [101]. After installing DSP Builder software, two libraries will be added into Simulink Library Browser of Simulink software which are Altera DSP Builder Advanced Blockset and Altera DSP Builder Blockset. DSP Builder software offers some features including:

- VHDL script is generated automatically by using *testbench* block from Altera DSP Builder Blockset library.
- It is ideal for project prototyping along with Altera development boards.

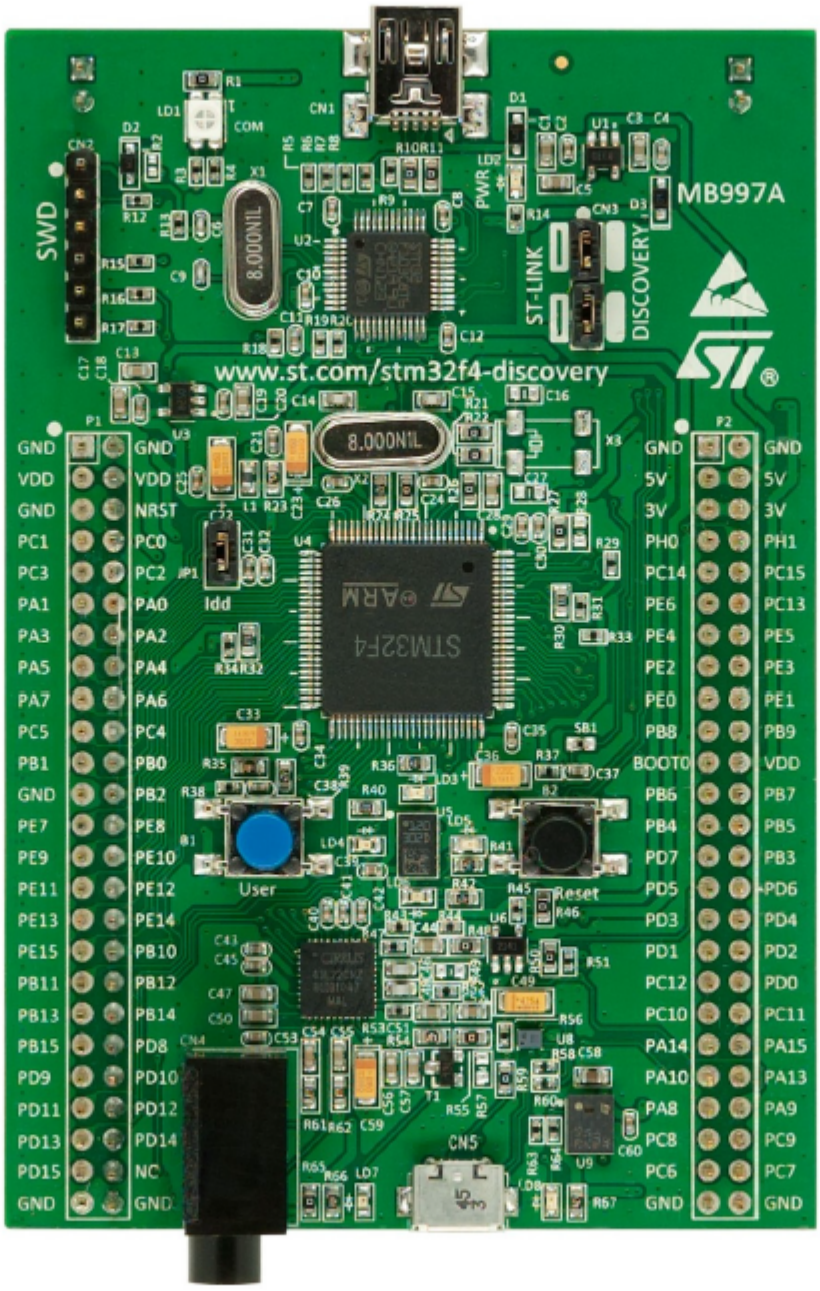

Figure 5.2: STM32F4DISCOVERY Board [19]

- DSP Builder supports tabular and graphical state machine editing.
- DSP Builder would increase the arithmetic and logical operators of Simulink software [100].
- It supports *SignalTap II logic analyser* to get fast functional verification of the design that runs on a development board [102].

5.5 ModelSim Software

The ModelSim software is a Mentor Graphics simulation tool for logic circuits [103]. In this thesis, ModelSim-Altera10.0c is used to produce the graphics simulation results of the implemented PMR channel and PR equaliser block diagrams. This software comes with Quartus II 11.1 software.

5.6 Altera Quartus II Software

Altera Quartus II software is a multiplatform design environment that suits the design into specific design requirements. Altera Quartus II offers some features that would help performing the design easily, quickly and highly performance such as:

Incremental Compilation and Rapid Recompile features- Only the partitions of the design that change between compilations could be incrementally compiled through Quartus II Analysis and Synthesis and Quartus II Fitter tools. The compilation time could be reduced by up to 70 percent. However for small engineering change orders, the Rapid Recompile feature reduces the compilation time by 65 percent [104].

Timing and Power Analysis- The TimeQuest timing analyzer tool analyzes the timing characteristics of the design. It offers Graphical User Interface (GUI) and scripting environment for timing constraints and reports. It also analyzes the timing paths in the design,

calculates the propagation delay along each path, checks for timing constraint violations, and produces report so that the designer can customize this report to find some timing information about particular paths [104].

PowerPlay Power Analyzer Flow- This feature is used for power consumption estimation purposes of the design. Quartus II PowerPlay power analysis and optimization tool provides an estimation of the design power consumed and then produces the estimated information as a Microsoft Excel-based spreadsheet.

SignalTap II Logic Analyzer and System Console- SignalTap II logic analyzer tool captures the signal flow while the design model runs under the FPGA chip clock. Moreover, DSP Builder offers SignalTap II Analysis block to visualize the output signal. This block allows testing the designed circuit and monitoring the inputs/outputs results under the FPGA chip clock [105] [106]. The results could be imported into MATLAB Workspace for further investigation [106]. System Console tool performs with Tool Command Language (Tcl) scripts and GUI low-level hardware debugging of the design [104].

Quartus II Block Editor- Quartus II Block Editor tool reads and edits block design files (.bdf) which allows entering and editing graphic design information of the schematic form designs.

MegaWizard Plug-In Manager- The MegaWizard Plug-In Manager helps the designer creating or modifying design files that contain custom megafunction variations [104]. For instance to use one of the internal oscillators that Altera FPGA board has, the schematic diagram of that oscillator would be created using MegaWizard Plug-In Manager tool.

5.7 What is an FPGA?

In 1985, the first FPGA was invented by Ross Freeman, co-founder of Xilinx, for implementation and verification of a wide rang of algorithms and digital circuitry. FPGA is a

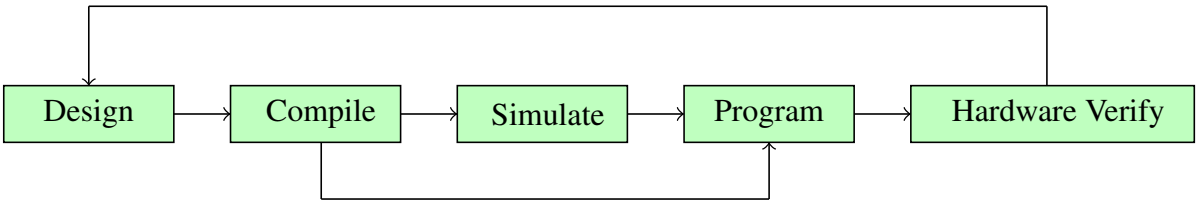

Figure 5.3: FPGA Standard Design Flow [20]

reprogrammable semiconductor device containing a matrix of logic cells. FPGA chip can be configured to implement any hardware design system using VHDL or Verilog [99]. The two major FPGA vendors are Altera and Xilinx. Unlike PC processors that are required run a software application, programming an FPGA means rewiring the chip itself to implement the design. In this work, DE1 development and education board is used to verify the design. The board has FPGA chip, I/O interfaces, switches and LEDs, memories, displays and clocks.

5.8 How to Create an Altera FPGA Design?

Creating an FPGA design means creating a digital circuit and then implement it inside the FPGA chip [20]. Thus, the design requirements are Altera Quartus II software, Altera FPGA development board and imagination of the designer. Figure 5.3 explains the standard design flow of any FPGA design. The design could be implemented using either schematic diagram or HDL. A schematic diagram is a graphical representation of a digital circuit consisting of modules, like logic gates, and connections between the modules [107]. It is an ideal method for the simple designs but with increasing the complexity of the digital circuits, the schematic diagram is no longer practical [107]. Therefore, the alternative method is HDL.

5.8.1 Schematic Diagram

In 1980s, the schematic capture was introduced into the VLSI design field [108]. The schematic capture is a hand-drawing of a digital circuit by using Computer-Aided Design (CAD) tool as a first step of design construction. Then a database is generated from the completed hand-drawn schematic. After that a simulation is performed to check the functionality of the designed model. The design may require to be simulated many times until the designer is satisfied that the design is functioning correctly [108]. With increasing the complexity of the design to hundreds of thousands of the logic components, the schematics capture is no longer a design tool. Therefore, since that time (1990s) HDL has become the alternative way of design. The most common types of HDL is VHDL and Verilog.

5.8.2 VHDL

VHDL is a language that describes digital electronic components (hardware). In 1980, the United States Government initiated VHSIC program. The outcome of this program was a need for a language to describe the structure and the function of the integrated circuits. Therefore, VHDL was developed and later became as a standard by the Institute of Electrical and Electronic Engineers (IEEE) in the US [109]. VHDL is a powerful language that meets the design requirements like a description of the structure of the design where VHDL allows that by decomposing the design into sub-designs and then recombine them. The structure of VHDL is divided into three sections. The first section is the *entity* where it declares input/output ports names of the design. Every VHDL design should have only one *entity* declaration [108]. The second section is the *architecture* which describes the implementation of the behaviour of the design model and it is called the body of the design. There may be multiple *architectures* that describe one *entity* [108] [109]. Finally, *configuration* section. This part of the design structure declares the *entity* and *architecture*

of the design. However, the *configuration* is not necessary especially if the design has one *architecture* with no component instantiations [108].

The final VHDL design could be simulated before being manufactured so that the design would be tested and corrected without doing hardware prototyping [109].

5.9 Implementation a Model (Block diagram) in Hardware

This section introduces the procedure of using DSP Builder software for any design on Altera FPGAs. This procedure will be used in the next chapter in order to implement PR equaliser system in hardware. The design flow starts by creating a DSP Builder (including Simulink software) design model, adding it into Quartus II software and then downloading it into the FPGA chip. The output results of MATLAB (if the design model was implemented in MATLAB), Simulink and hardware implementation should be matched. Therefore in order to understand the design flow of any model or block diagram, the design flow of each software will be individually explained in the following sections.

5.9.1 Design Flow of Simulink Models

The design flow of any Simulink design model involves the following steps:

- Creating a model in a combination of Simulink of MathWorks software and DSP Builder software from Altera.

  For any design model, the DSP Builder blocks should be separated from the Simulink blocks by *Input* and *Output* ports from DSP Builder IO and Bus library. *Input* port defines the input boundary of a hardware system and casts floating-point Simulink signals to signed binary fractional format (input to the DSP Builder blocks). On the contrary, the *Output* port defines the output boundary of a hardware system and casts signed binary fractional format (from DSP Builder blocks) to floating-point Simulink

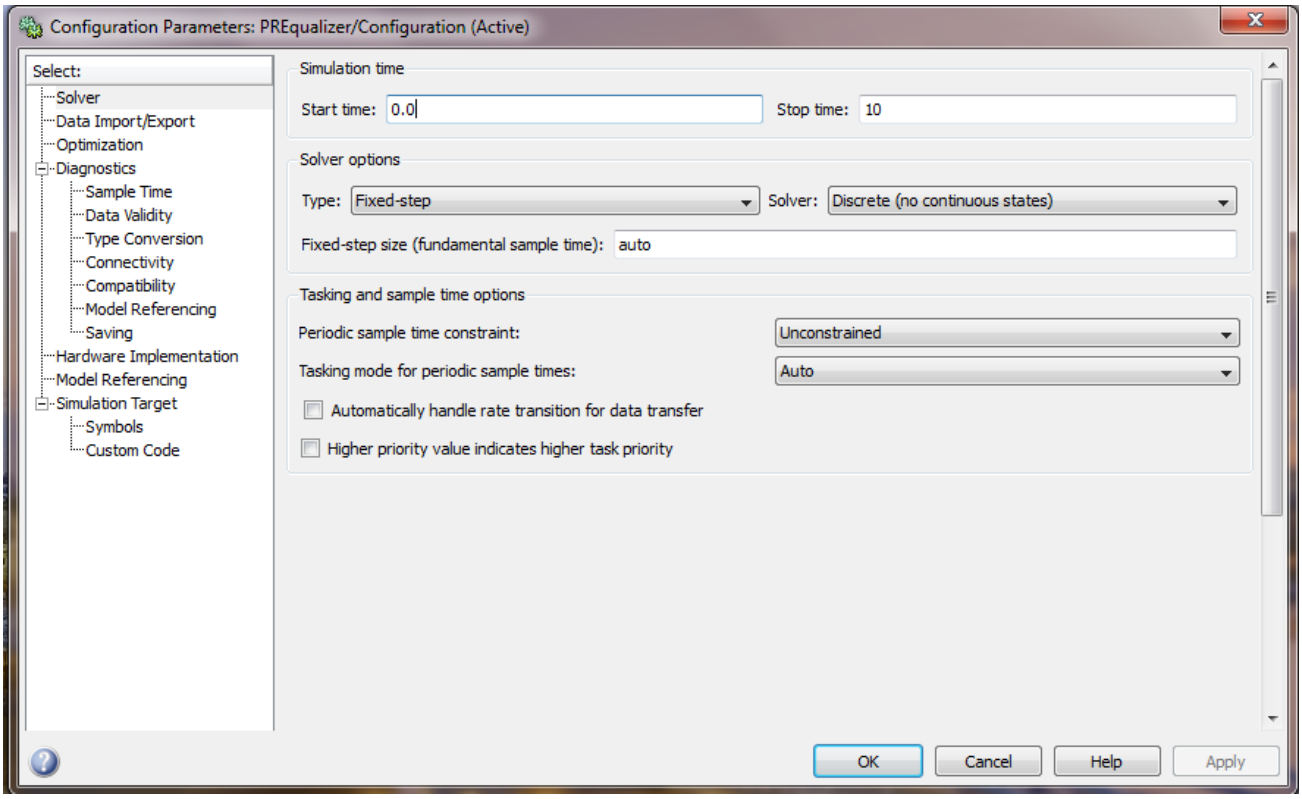

Figure 5.4: Solver Pane of Simulink Software

signals (to Simulink blocks). However, *Input* and *Output* blocks can be parameterized into other signal format like signed integer and unsigned integer formats.

- Choosing the right *solver* for that design using the Configuration Parameters dialog box in Simulink that is shown in Figure 5.4.

  Solver is a component of the Simulink software which determines the time of the next simulation step of a design model and applies a numerical method to solve a set of ordinary differential equations of that design. There are two major types of solver: fixed-step and variable-step which could be further divided into discrete or continuous. Next simulation time of both solvers, fixed-step and variable-step, is calculated as the sum of the current simulation time plus a quantity called step size. The step size is constant amount throughout the simulation for fixed-step and it varies

from step to step for variable-step solver depending on the dynamics of the model design. In addition, discrete and continuous solvers depend on the model design blocks to calculate the values of any discrete states [110]. In this work, fixed-step discrete solver is chosen because the step size is constant.

- Simulating the model in Simulink using *start simulation* button and then using *Scope* block to visualize the results. Simulating a model in Simulink means computing its inputs, outputs, and states at intervals starting from the simulation start time that specified at the Solver pane to the stop time. Default simulation time starts with 0.0 seconds and ends at 10.0 seconds. Simulation time differs from the actual clock time. For instance, if the simulation time is specified for 10 seconds this does not take 10 seconds. However, the amount of time that is taken to simulate any design model depends on model complexity, solver's step size and computer speed.

- Adding *Signal Compiler* block from Altera DSP Builder Blockset library and then running it to perform Register-Transfer Level (RTL) simulation and synthesis. RTL simulation is used in HDLs to create high-level representations of the model. *Signal Compiler* block reads Simulink file (.mdl) which has a mixed blocks of Simulink and DSP Builder softwares and then generates VHDL files and Tcl scripts for synthesis, hardware implementation and simulation [100].

- Adding *TestBench* block from Altera DSP Builder library to perform RTL simulation with ModelSim software. ModelSim gives behavioral simulation of a number of languages like VHDL, Verilog and SystemC [111]. The waveforms that are obtained by ModelSim should match the *Scope* output in Simulink.

5.9.2 Quartus II Implementation of Schematic Diagram and VHDL Designs

This section illustrates how to use Quartus II CAD software to implement the logic circuits in an Altera FPGA device whether the design entry is schematic design or VHDL design. The typical FPGA CAD design flow is explained below:

- ***Design Entry-*** The design can start as a schematic diagram or HDL such as VHDL or Verilog [104][112] [113].
- ***Synthesis-*** The desired circuit (entered design) is synthesized into a specific LEs that consists the FPGA chip.
- ***Functional Simulation-*** This is the first and easiest simulation that the designer should perform in order to make sure the functionality of that design is correct. This simulation is accomplished without timing issues.
- ***Fitting-*** At this stage, the CAD Fitter tool maps the location of LEs specified in the netlist into the LEs in an FPGA board.
- ***Timing Analysis-*** It puts delays to verify the performance of the fitted circuit.
- ***Timing Simulation-*** It tests the fitted circuit to verify the timing and the functionality are correct.
- ***Programming and Configuration-*** It implements the designed circuit in FPGA chip [112] [113].

5.9.3 Adding a Design into Quartus II Environment

The following procedure clarifies the way of adding any Simulink-DSP Builder design model into Quartus II Environment (creating a Quartus II project) and then running the design on FPGA board:

1. Create a new Quartus II project in order to create a set of files which are Quartus II Settings File (.qsf) and Quartus II Project File (.qpf) to maintain the information of that design.

2. Assign the FPGA device family, EP2C20 is used in this work, that corresponds to the device family on the FPGA board that will be used for that design.

3. Create a schematic or (.bdf) file which is the top-level design by choosing: File> New> Block Diagram/Schematic File on Quartus File menu and save it as top-level design.

4. Add VHDL code that has been generated by running *Testbench* block of the designed model in Simulink: Choose File> New> VHDL File on Quartus File menu and save it as (.vhd) or (.vhdl) and then copy this code from ModelSim vhdl folder and paste it into the blank file and then save it.

5. Create symbol files for the current file by choosing File> Create/Update> Create Symbol Files for Current File on Quartus File menu. Double-click the blank area of the bdf to open the Symbol dialog box, expand the Project directory to get the created symbol design.

6. Add input and output pins for the created symbol above by double clicking, again, the blank area of the bdf to open the Symbol dialog box Change the name of input and output pins to exactly match the names on the created symbol.

7. Compile the design by using *Start Compilation* button on Quartus Processing menu to make sure no errors like bad pin connections error.

8. Prepare the design for assigning pin locations by choosing Processing > Start > Start Analysis and Elaboration.

9. Assign the pins by selecting Assignments > Pin Planner.

10. Now, compile the design once again to convert the design into a bitstream which can be downloaded into the development board. The compilation process generates (.sof) file that will be using to program the device.

11. Program the device. Now the device is ready to be programmed but before that a hardware set up is required, perform that using the following steps:

    - Connect the USB cable to that DE1 development board.
    - Power up the board by pressing the ON/OFF switch on that board.
    - Turn RUN/PROG Switch to the RUN position [114].

12. Select Tools >Programmer. Click the Hardware Setup tab and add USB-Blaster [USB-0] under Currently selected hardware then close this window. Add (.sof) file from the project directory if this file does not show otherwise click Start tap to start downloading (.sof) file to the DE1 board. The Progress bar should tell the successful downloading by showing that 100% status when the downloading the design is completed. The flashing should stop as an indication the design is now completely downloaded into the development FPGA board and it should be running.

5.10 Summary

In this chapter:

- Different implementation softwares have been explained. Namely, Simulink, DSP Builder, ModelSim and Quartus II softwares.
- The process of verifying any system model in hardware is also explained in detail.
- The verification process includes the implementation of model using DSP Builder and MATLAB/Simulink and then in Quartus II softwares.
- Design a system model in DSP Builder software generates HDL scripts for the model. Those scripts are integrated with ModelSim and Quartus II softwares which they can be used to add that design into Quartus II environment and then download it into a FPGA chip.
- The softwares of implementation that mentioned above will be used in the next chapter in order to implement PRML system model in hardware.

Chapter 6

Hardware Implementation of PRML System

6.1 Introduction

This chapter presents the hardware implementation of PRML system. Hardware implementation procedure includes four stages which are designing, implementation, creating FPGA design and testing or running the design. The first stage, designing, includes creating a model using Simulink blocks from MathWorks in combination with Altera DSP Builder blocksets. Each block should be parameterized according to the design requirements. The second stage is the implementation of the model in Simulink. At this stage, DSP Builder software compiles the design model and generates its HDL scripts. The third stage is creating FPGA design using either schematic diagram or HDL script. Finally, testing the design using ModelSim software or running the design by downloading it into FPGA development board. The output results of Simulink, ModelSim and FPGA board should be matched.

Parts of this chapter appear in the Conference under the paper titled:

Nadia Awad, Mohammed Zaki Ahmed, and Paul Davey "Hardware Implementation of Partial Response Equaliser Channel", 3rd International Symposium on Advanced Magnetic Materials And Applications 2013, Taichung, Taiwan, July 2013.

6.2 Hardware Implementation of PRML Block Diagram

The block diagram of PRML system shown in Figure 2.4 is implemented in hardware. The design starts by opening a new model menu in Simulink and then following this block diagram step by step. The solver must be a discrete as no continuous states for this design. The Simulink blocks can not be converted into HDL. In other words, *Signal Compiler* block does not convert Simulink blocks into HDL during the compilation process. For instance, *Gain* block from DSP Builder must be used instead of *Gain* block of Simulink software [100]. Also, the DSP Builder blocks should be separated from the Simulink blocks by *Input* and *Output* blocks from DSP Builder IO and Bus library [100].

Because the PRML system is a complex system, therefore the design will be broken down into smaller parts. Every part of the main design would be implemented and tested separately which would help finding out the errors and the problems that might be there and fix them easily. Thus, the design flow of PRML block diagram will include the design of the following parts: PMR channel, PR Equaliser, Target and VA part.

6.3 Hardware Implementation of PMR Channel

This section presents implementation of PMR channel in a combination of Simulink and DSP Builder softwares, ModelSim simulator and Quartus II software.

6.3.1 Simulink/DSP Builder Softwares Implementation of PMR Channel

Implementation of PMR channel in MATLAB has been performed in Section 3.2. The Simulink implementation will be broken down into parts as follows.

Dibit Response Implementation

Equation (2.12) and Equation (2.13) define the impulse and the dibit responses of PMR channel respectively. In order to avoid the hardware complexity, Equation (2.12) is simplified into

$$h_t = \tanh(2.1972 \times (t - 0.5)) \tag{6.1}$$

The vector t has been imported from MATLAB workspace using *From Workspace* block from Simulink. The implementation of PMR channel in Simulink is shown in Figures 6.1, the upper part is for transition response implementation and lower part is for dibit response implementation. Figure 6.2 shows the impulse and dibit responses of the channel which have been obtained by *Scope* block.

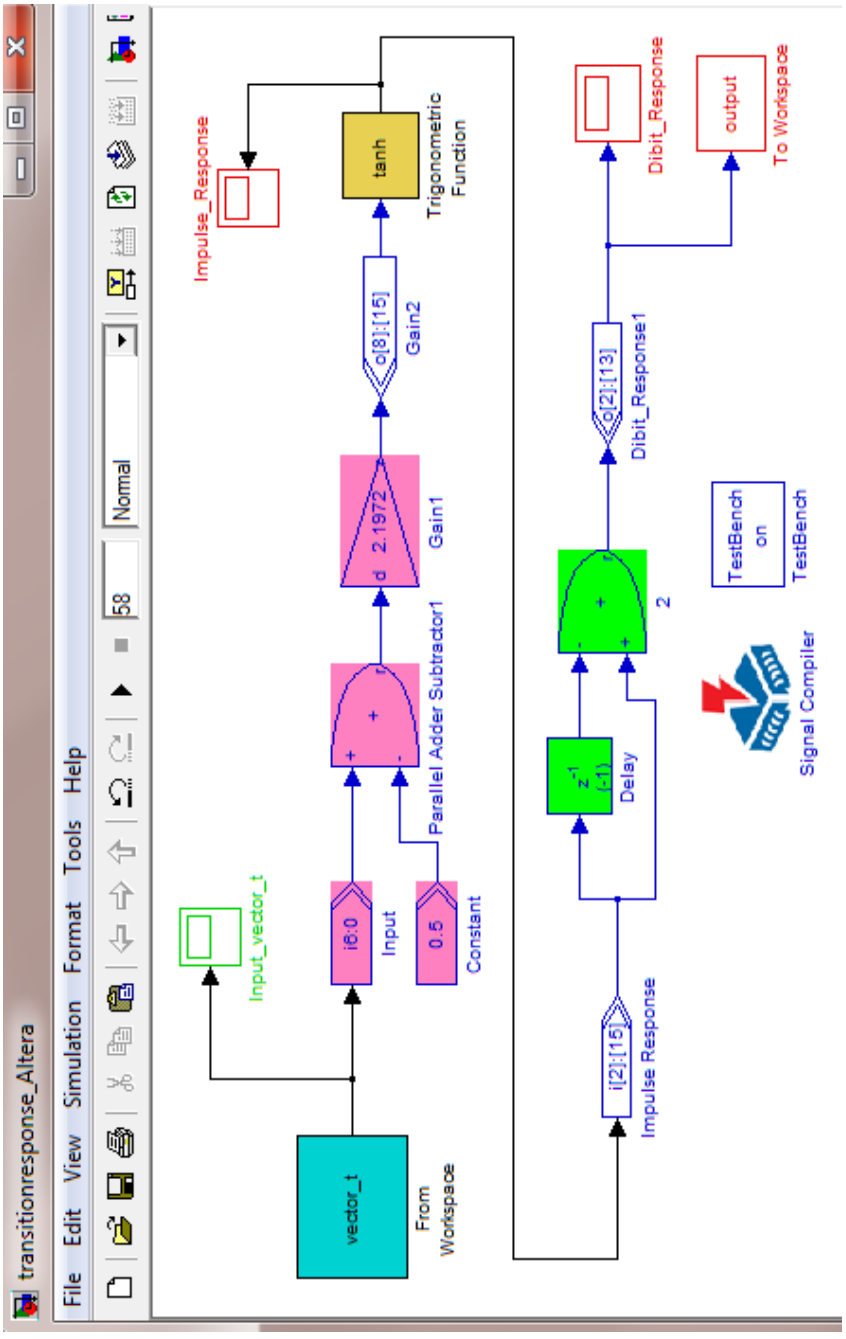

Figure 6.1: Simulink Implementation of The Dibit Response of PMR Channel

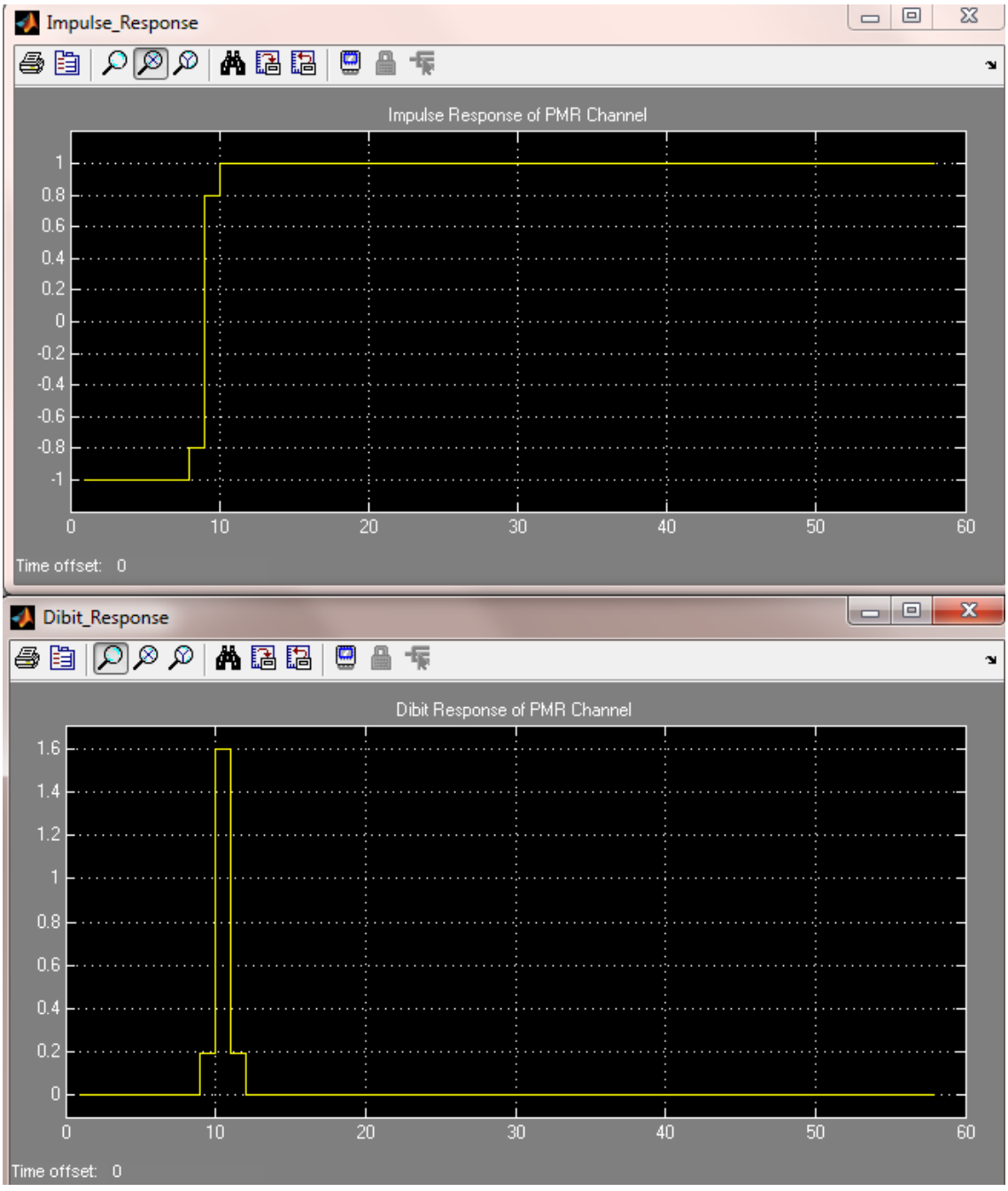

Figure 6.2: Impulse and Transition Responses of PMR Channel

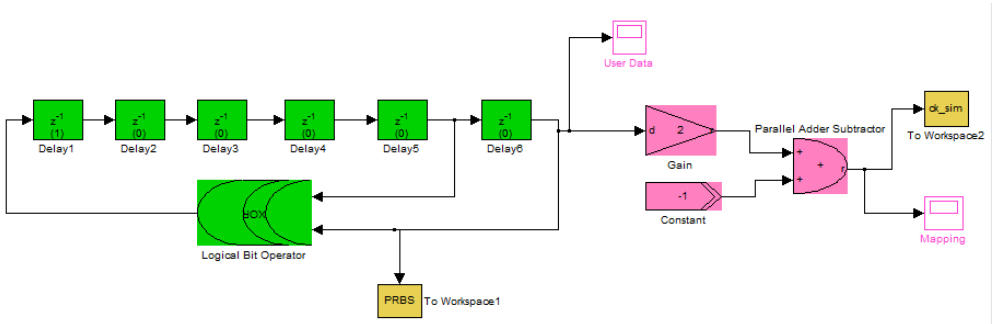

Figure 6.3: Simulink Implementation of The Mapping

User Data Implementation

The user data, a_k, is a randomly generated binary sequence of 1's and 0's. In Simulink, Pseudo Random Binary Sequence (PRBS) is generated to represent a_k sequence. Shift registers with feedback are used to generate a PRBS pattern. PRBS pattern is very common to use in serial interconnect technology [115]. Thus, *Logical Bit Operator* block and six *Delay* blocks from Altera DSP Builder Blockset library are used to generate the PRBS pattern. The output signal of any part of the design could be viewed by *Scope* block after clicking the *Start* button on the Simulation menu.

Mapping Implementation

In data storage system, the magnetic recording head writes the input data by magnetizing the storage media into two opposite directions north and south i.e. +1 and -1. Therefore, mapping is required to convert the binary sequence of the user data into a sequence of +1's and -1's. The $(2a_k - 1)$ equation is used to perform the mapping. In Simulink this can be accomplished by *Gain*, *Constant* and *Parallel Adder Subtractor* blocks from Altera DSP Builder Blockset as shown in Figure 6.3. The user data sequence and the mapped sequence are visualized by *Scope* block as indicated in Figure 6.4.

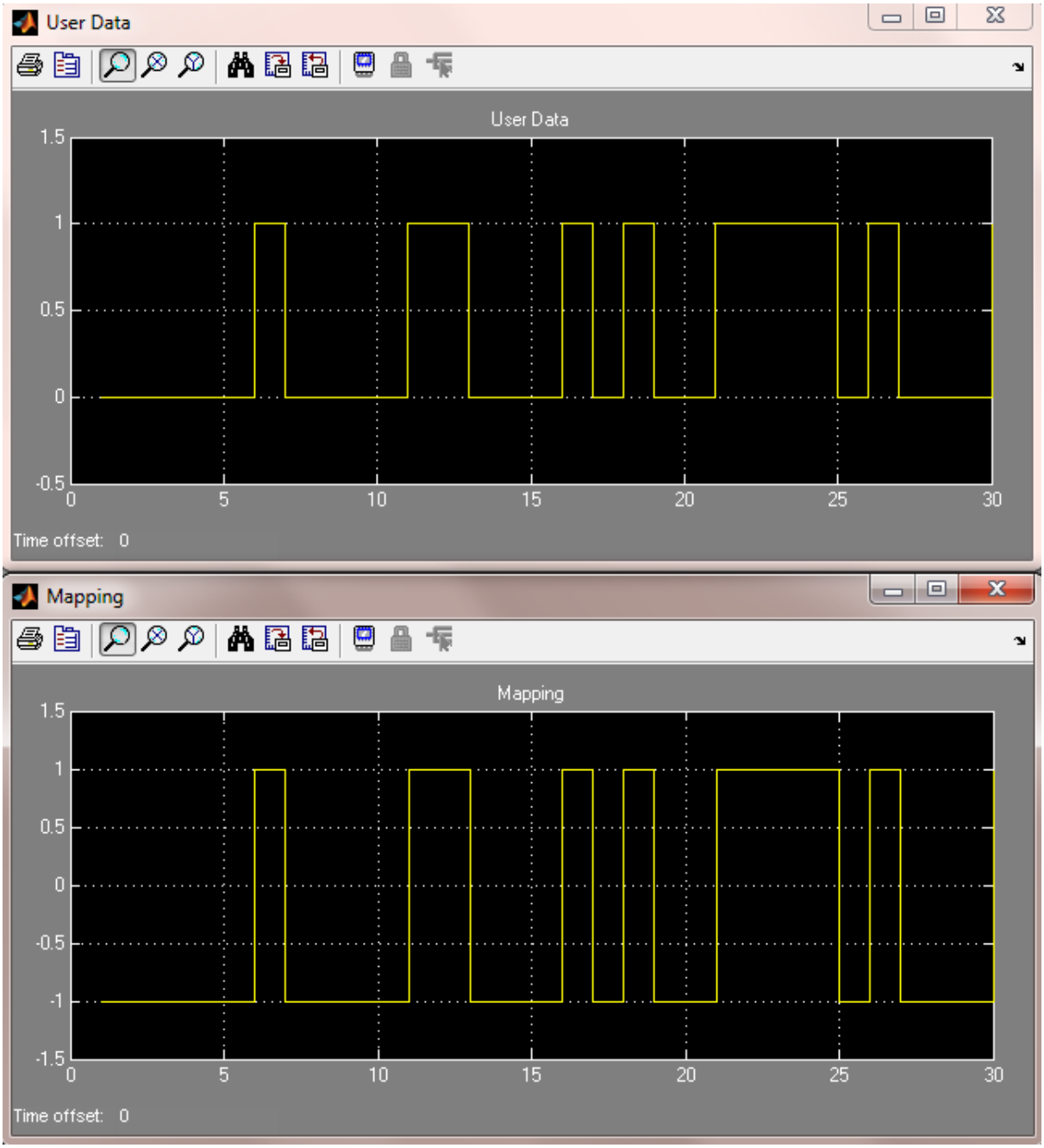

Figure 6.4: User Data and Mapped Input Sequences

Channel Implementation

The channel output s_k is a convolution of c_k and the dibit response of the channel.

Discrete convolution of two sequences such as $f(j)$ for $j = 0,1,2,...,a-1$ and $h(j)$ for $j = 0,1,2,...,b-1$, is defined as [116]

$$g(k) = \sum_{j=0}^{k} h(k-j)f(j) \tag{6.2}$$

for $k = 0,1,2,...,a+b-2$. Thus, the convolution is multiplication, shifting and adding.

To implement the convolution process in Simulink, the channel is represented as a 3-tap FIR filter. The filter coefficients are [0.1973 1.6 0.1973]. The other values of dibit response have been ignored for the complexity sake as adding many blocks would increase the design complexity. Thus three *Gain* blocks, two *Delay* blocks and *Parallel Adder Subtractor* block from the same library above are used to implement the channel. The initial value of *Delay* blocks is (-1) and the Bus Type for *Gain* blocks is Signed Fractional. Figure 6.5 shows the Simulink design of PMR channel and Figure 6.6 shows the channel output sequence. *Output* port is added at the end of each design. The setting of the parameters of the *Output* port is: The Bus Type is Signed Fractional, Number of Bits for the sign and magnitude is 3 and 15 bits for the fractional part. The 15-bit is chosen in order to get the required precision.

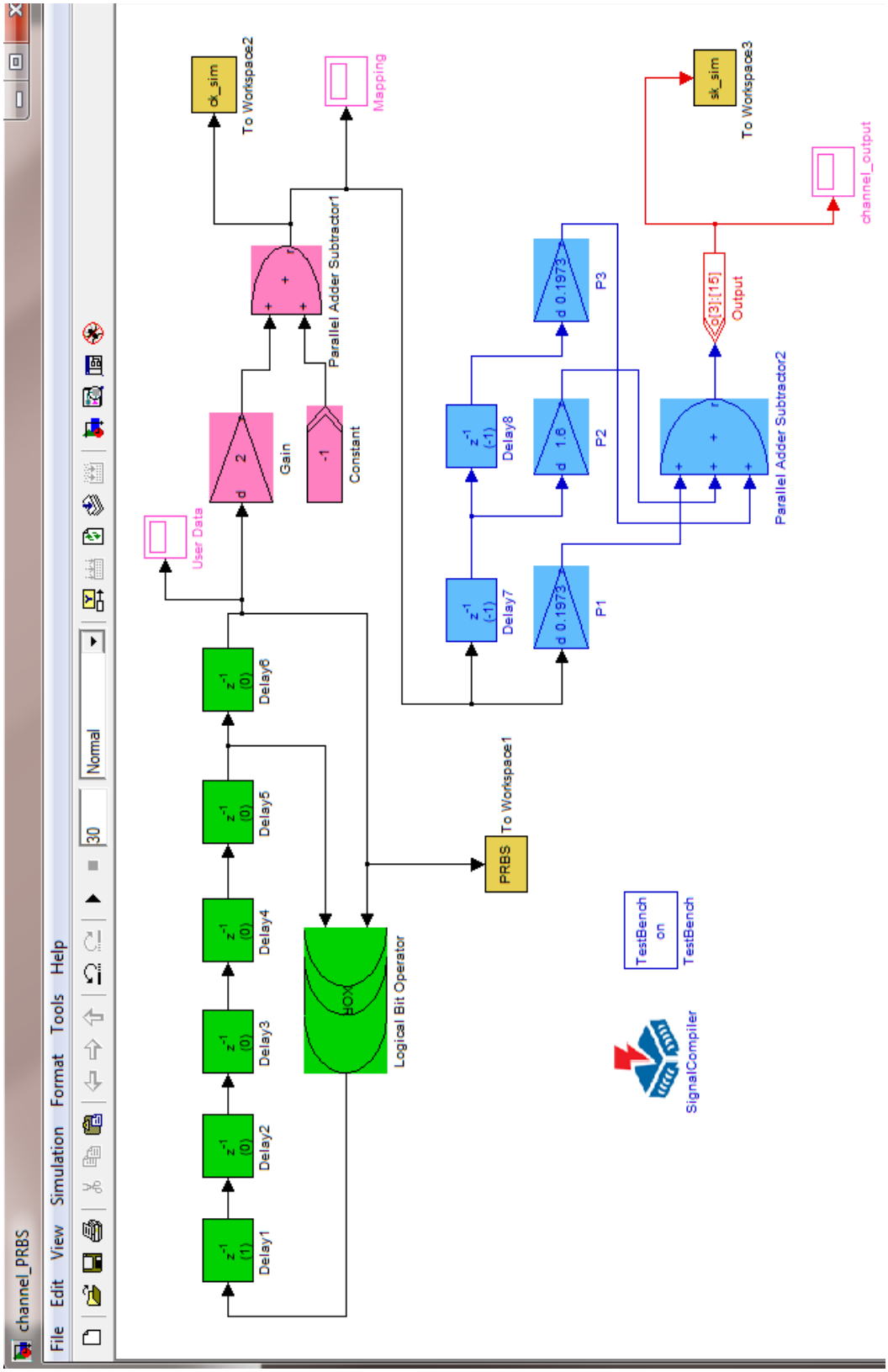

Figure 6.5: Simulink Implementation of PMR Channel

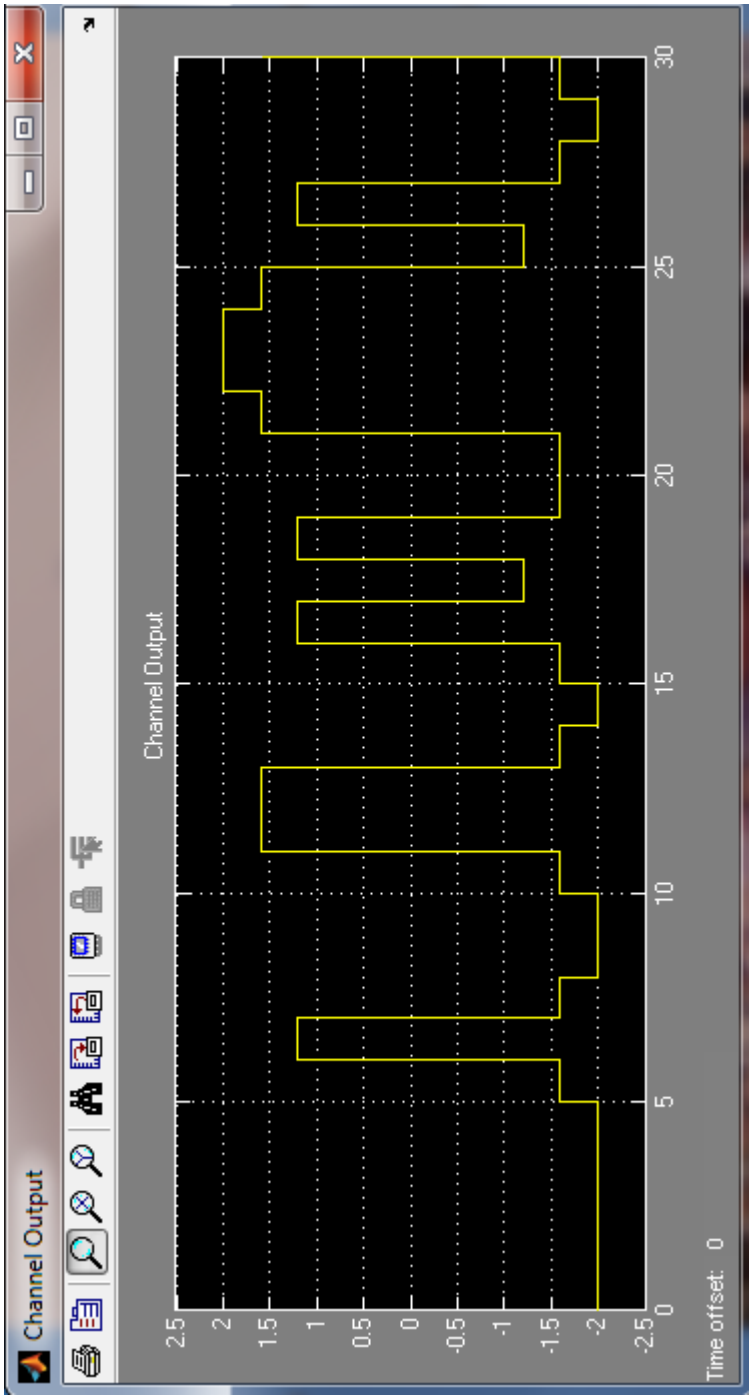

Figure 6.6: Simulink Design Output of PMR Channel

6.3.2 Verifying Channel Simulink Design in ModelSim Software

ModelSim software is used to edit, compile, simulate and then to verify the results of Simulink as following:

1. Simulation in Simulink. Simulate the design model using *Start* on the Simulation menu.

2. Compilation the design. The compilation of the DSP Builder design is accomplished by *Signal Compiler* block from Altera DSP Builder Blockset library. The device family for this design is Cyclone II and it is specified by double-clicking the Signal Compiler block and selecting that device from the list.

3. Performing RTL Simulation by adding *Testbench* block from the same library of *Scope* block. Simulation confirms the design behavior before programming the FPGA device [102]. The *Testbench* block offers many advantages such as:

    - Launch GUI. Loading the testbench into ModelSim would allow the designer to compare the results with Simulink. This can be executed by launching the ModelSim GUI of the Advanced tap of the main menu of *Testbench* block.
    - Generate a VHDL-based testbench from the model design by using *Generate HDL* tap.
    - Generate Simulink simulation results for the testbench by clicking *Run Simulink* tap.
    - Running ModelSim by using *Run ModelSim* tap.
    - Finally, *Compare Results* tap compares the results of ModelSim and Simulink.

The ModelSim output result of PMR channel design model is shown in Figure 6.7. The results of Simulink design and ModelSim are matched.

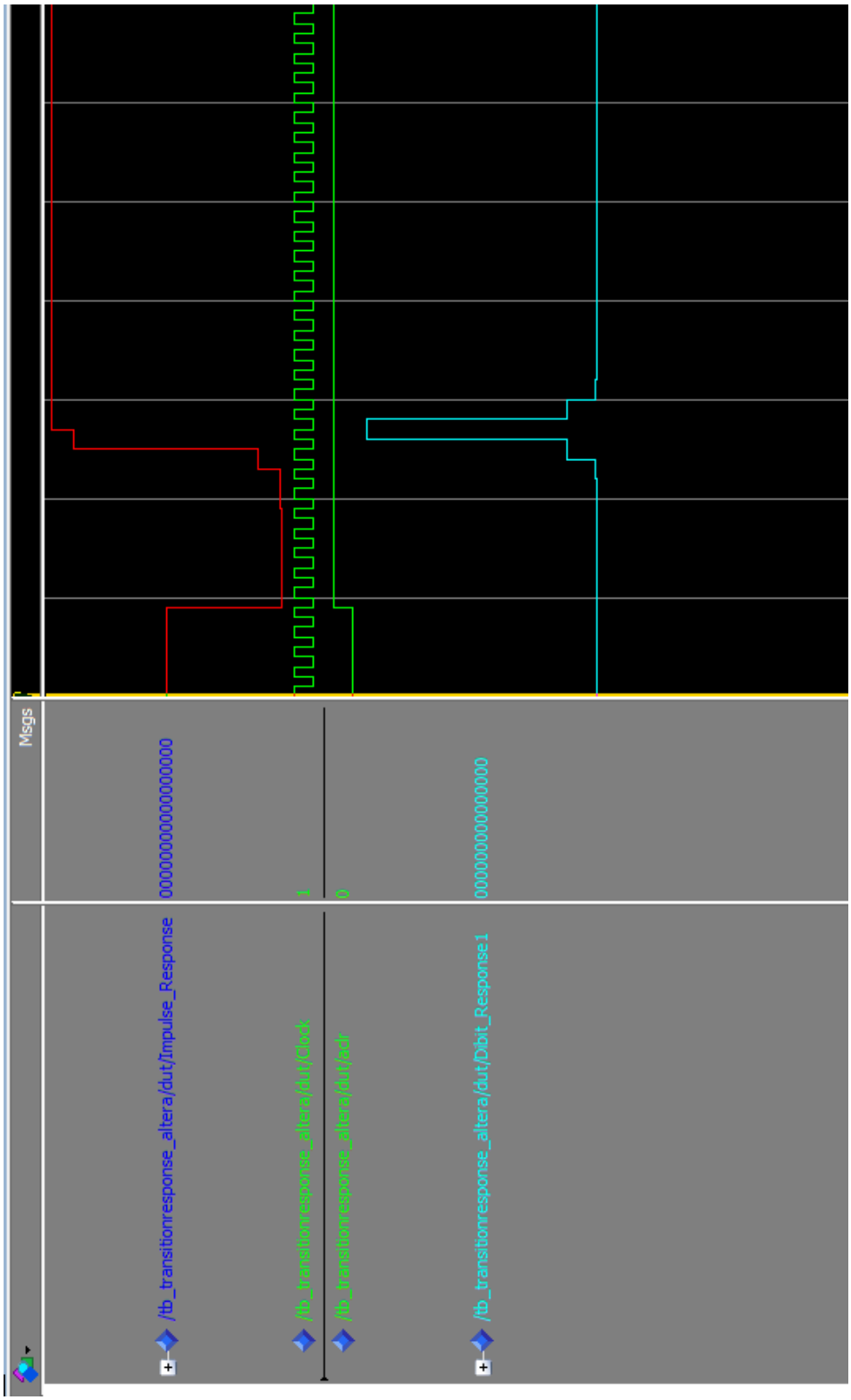

Figure 6.7: ModelSim Output of the Implemented PMR Channel

6.3.3 Verifying Channel Simulink Design in Hardware

In order to verify the Simulink model of the channel in hardware, a Quartus II design model is required as shown in Figure 6.8. Quartus II design model requires 24 input pins and 17 output pins to produce the transition response of the channel in addition to two pins for the *aclr* signal and the clock signal. The design registers are initialised by the default active-low *aclr* signal [117]. So that 43 pins of FPGA chip should be assigned. For dibit response output result, the design requires 24 input pins and 35 output pins and also the *aclr* and the clock signals pins so that 61 pins of FPGA device should be assigned. DE1 board provides two 40-pin expansion headers each one connects to 36 pins on FPGA chip. Each header provides DC +5V, DC +3.3V and two GND pins. For both responses the two expansion headers of DE1 board should be used and then send the digital data into DAC to produce the output results. Figure 6.9 shows the compilation report of PMR channel Quartus II design. The report shows that 1% has been used from the total logic elements of FPGA chip for that design.

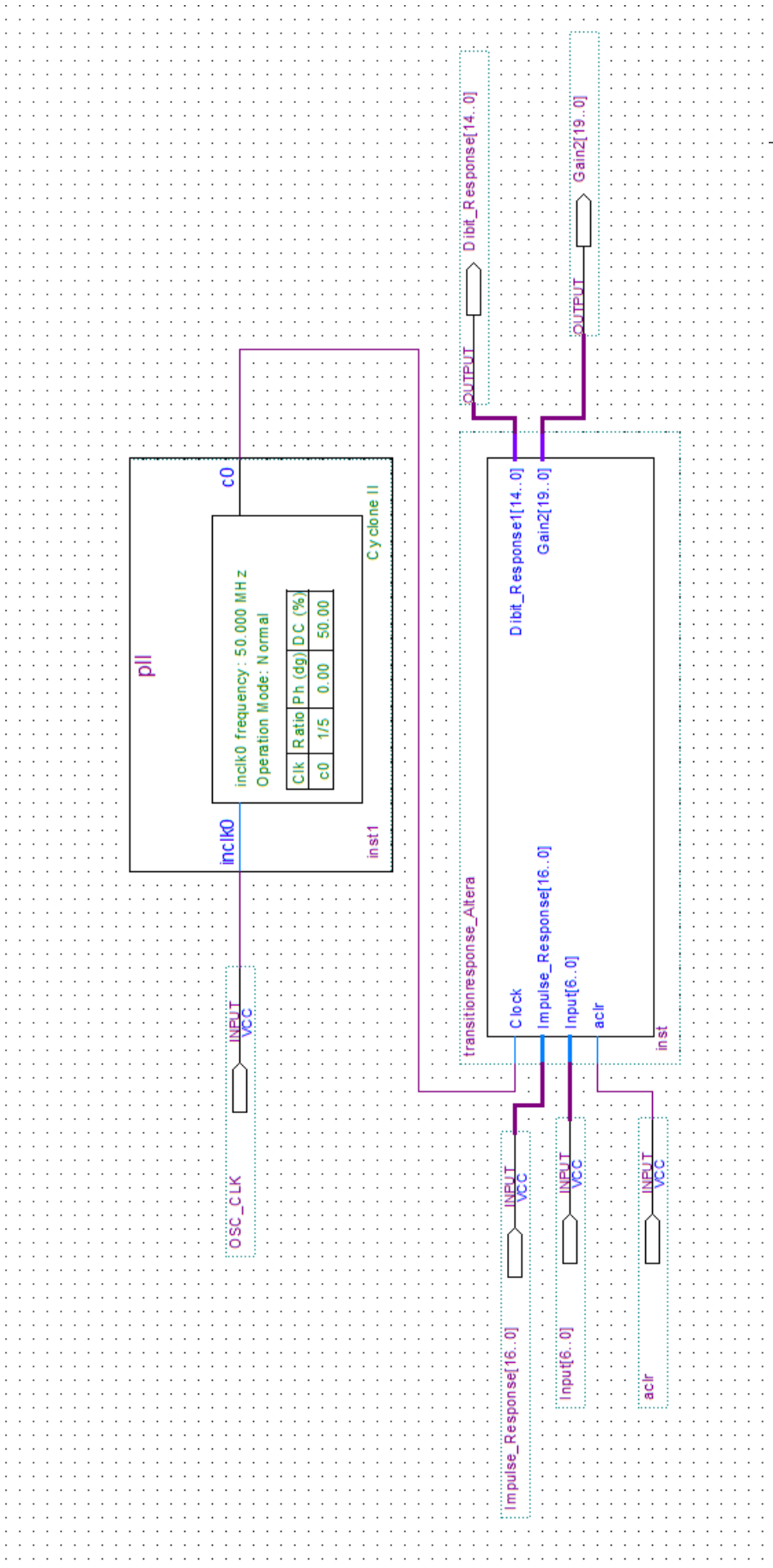

Figure 6.8: Quartus II Design of PMR Channel

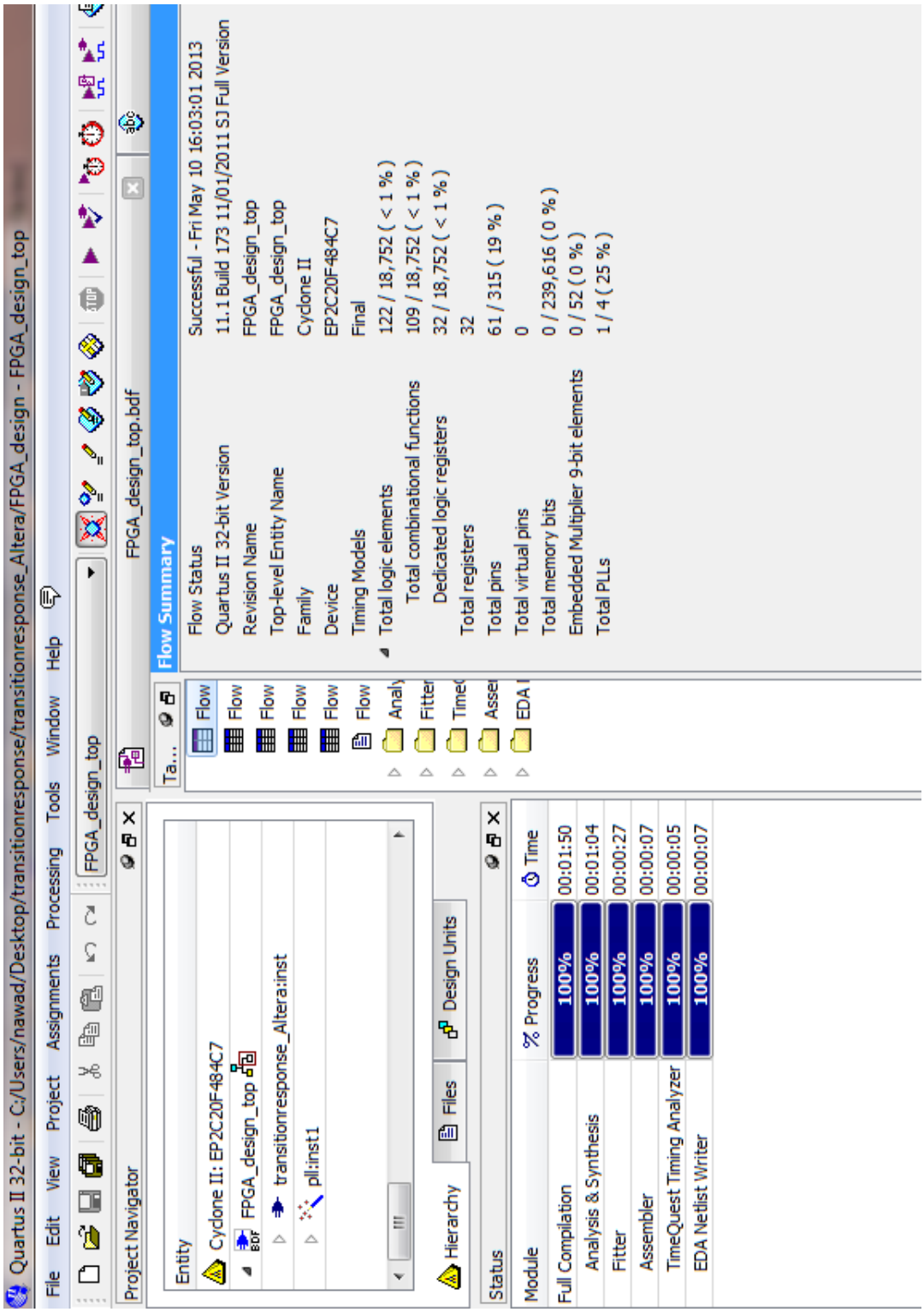

Figure 6.9: Compilation Report of PMR Channel Quartus II Design

6.4 Hardware Implementation of PR Equaliser

This section illustrates implementation of PR equaliser in Simulink/DSP Builder softwares, ModelSim and FPGA device.

6.4.1 Simulink/DSP Builder Softwares Implementation of PR Equaliser

Simulink implementation of PR equaliser before adding AWGN and after adding AWGN using a combination of Simulink and DSP Builder Softwares is introduced in this section.

1- Without AWGN

The PR equaliser is a 7-tap FIR digital filter. Seven *Gain* blocks, six *Delay* blocks and *Parallel Adder Subtractor* block from Altera DSP Builder Blockset library are used to implement the filter. The initial value of *Delay* blocks is (-2) and the Bus Type for *Gain* blocks is Signed Fractional. The input sequence of this design is PRBS pattern. From the channel, 18-bit signed fractional data is received and then is applied to the equaliser. The setting of the parameters of the *Output* port is: The Bus Type is Signed Fractional, Number of Bits for the sign and magnitude is 5 and 17 bits for the fractional part. Therefore, the output data signal is a 22-bit signed fractional bus will be sent to the decoder.

In order to reshape the channel output into a particular response, 3-tap FIR digital filter (GPR target) is applied. The filter coefficients are [4 6 4]. The initial value of *Delay* blocks is (-1) and the Bus Type for *Gain* blocks is Signed Integer. Therefore to implement it in DSP Builder software, three *Gain* blocks, two *Delay* blocks and *Parallel Adder Subtractor* block are used.

Figure 6.10 shows the complete system design without AWGN. The equaliser and target output results are represented in Figure 6.11.

Figure 6.11 shows that the two sequences are not the same. Therefore, three *Delay*, *Parallel*

Adder Subtractor, *Product*, *Mean* blocks are added at the output of the target in order to make the two sequences identical. Figure 6.12 shows the difference between the sequences is zero.

The PR equaliser design model is verified in ModelSim after following the same procedure of verifying the channel design model. The ModelSim output result of the this design model is shown in Figure 6.13. The results of Simulink and ModelSim are exactly matched as shown in Figure 6.14.

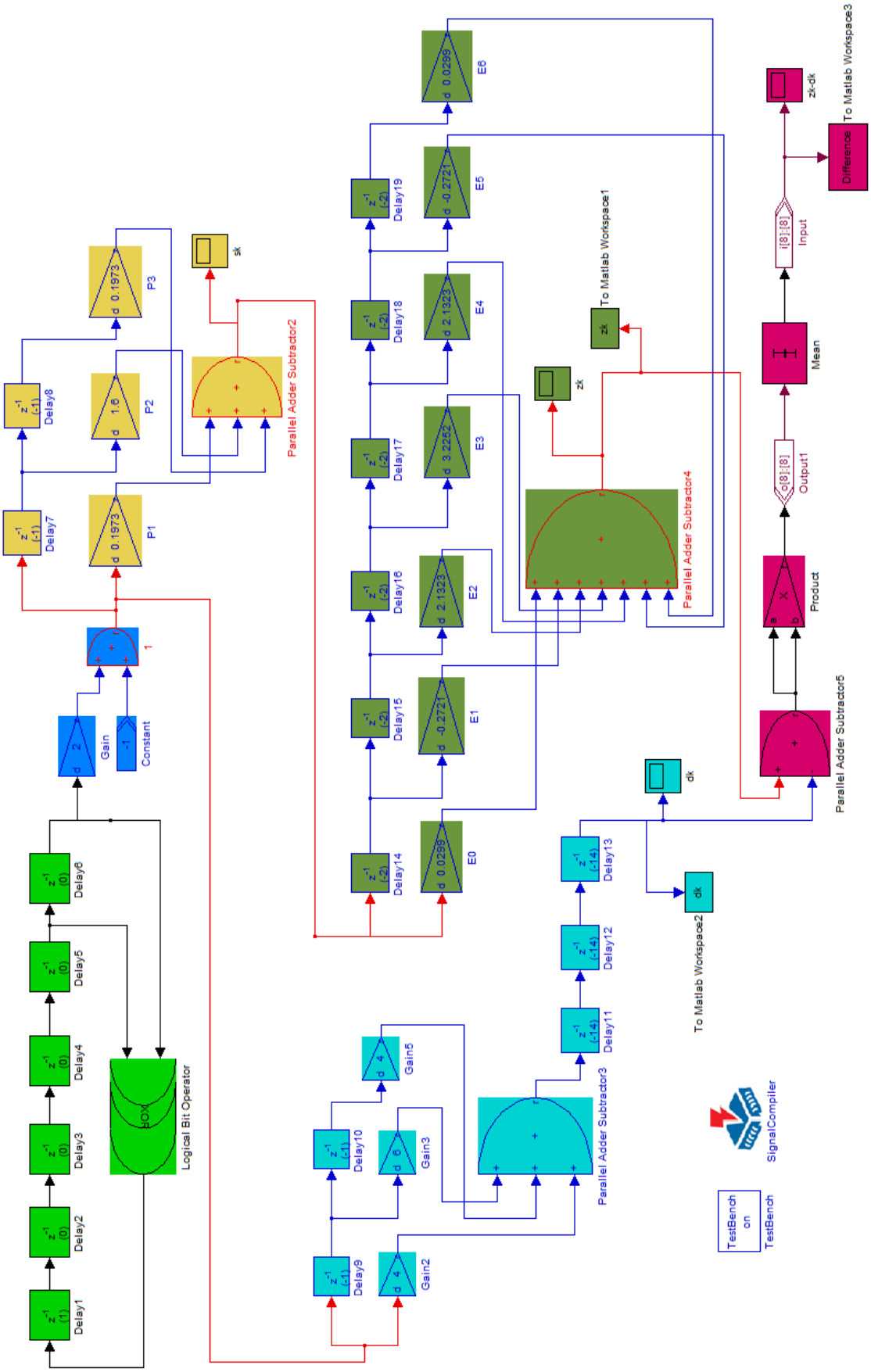

Figure 6.10: DSP Builder Design of PR System Without AWGN

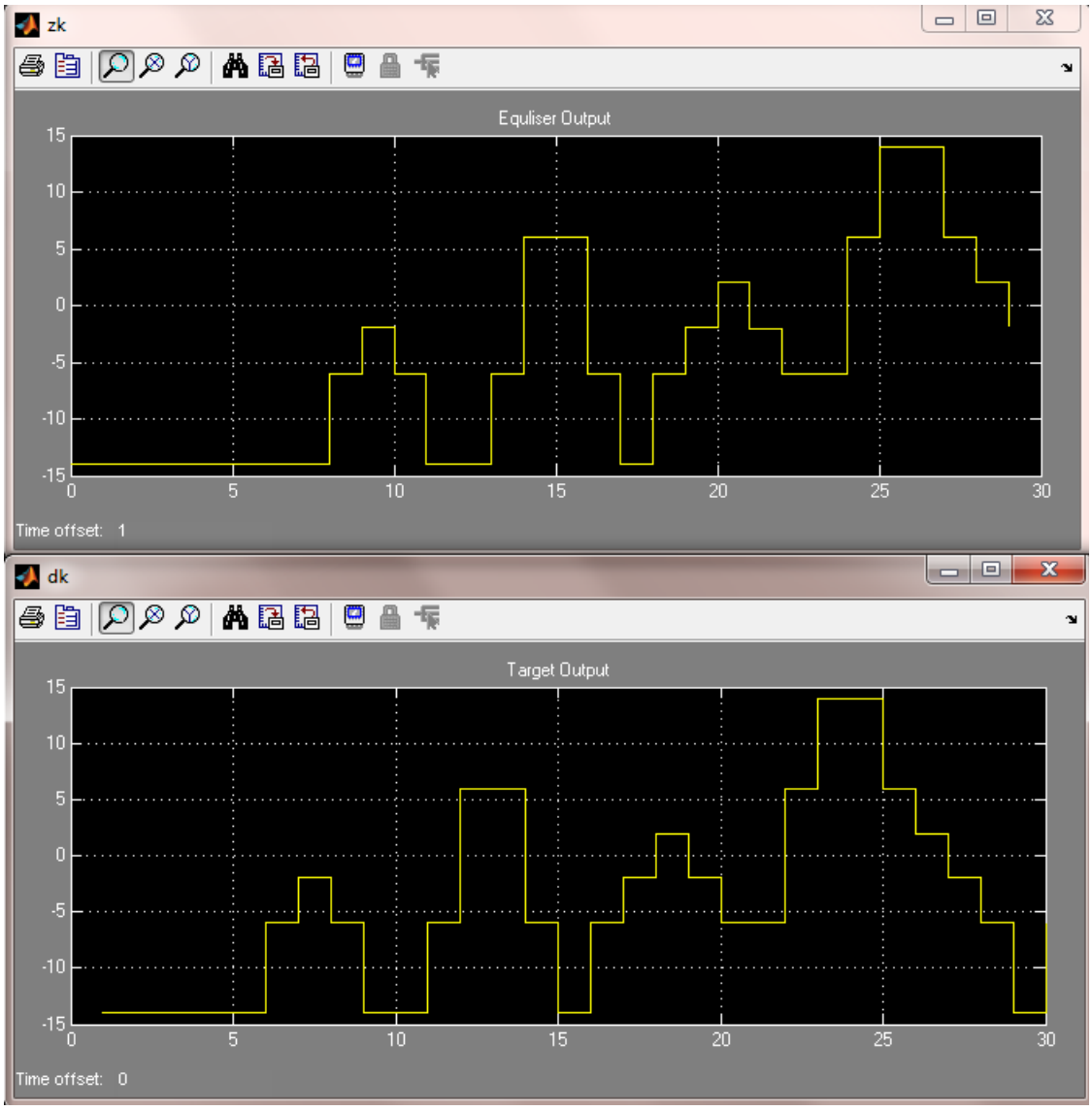

Figure 6.11: Equaliser and Target Output Results of Simulink implementation of PR System Without AWGN

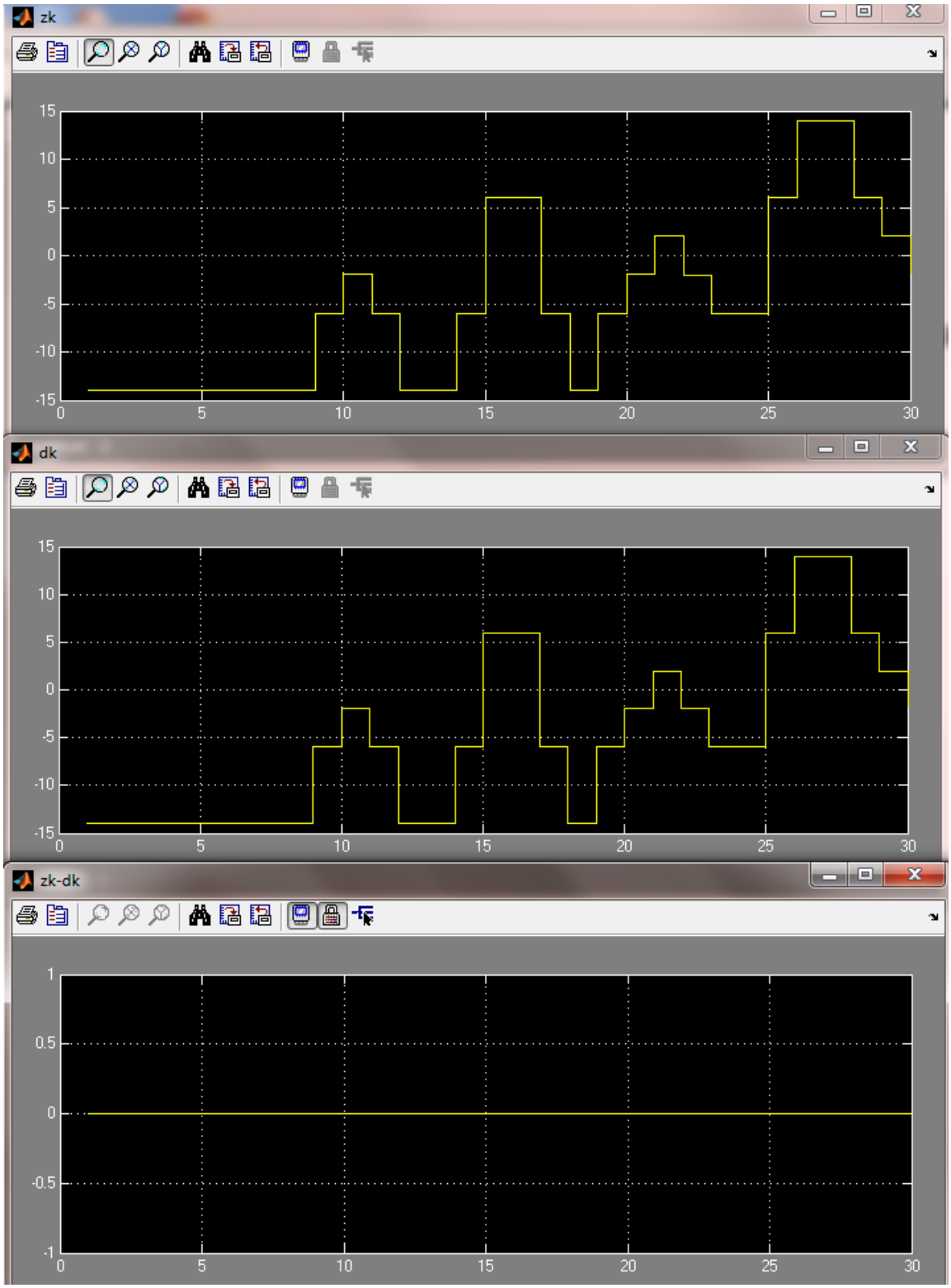

Figure 6.12: The Difference Between Equaliser and Target Sequences of Simulink Design of PR System Without AWGN

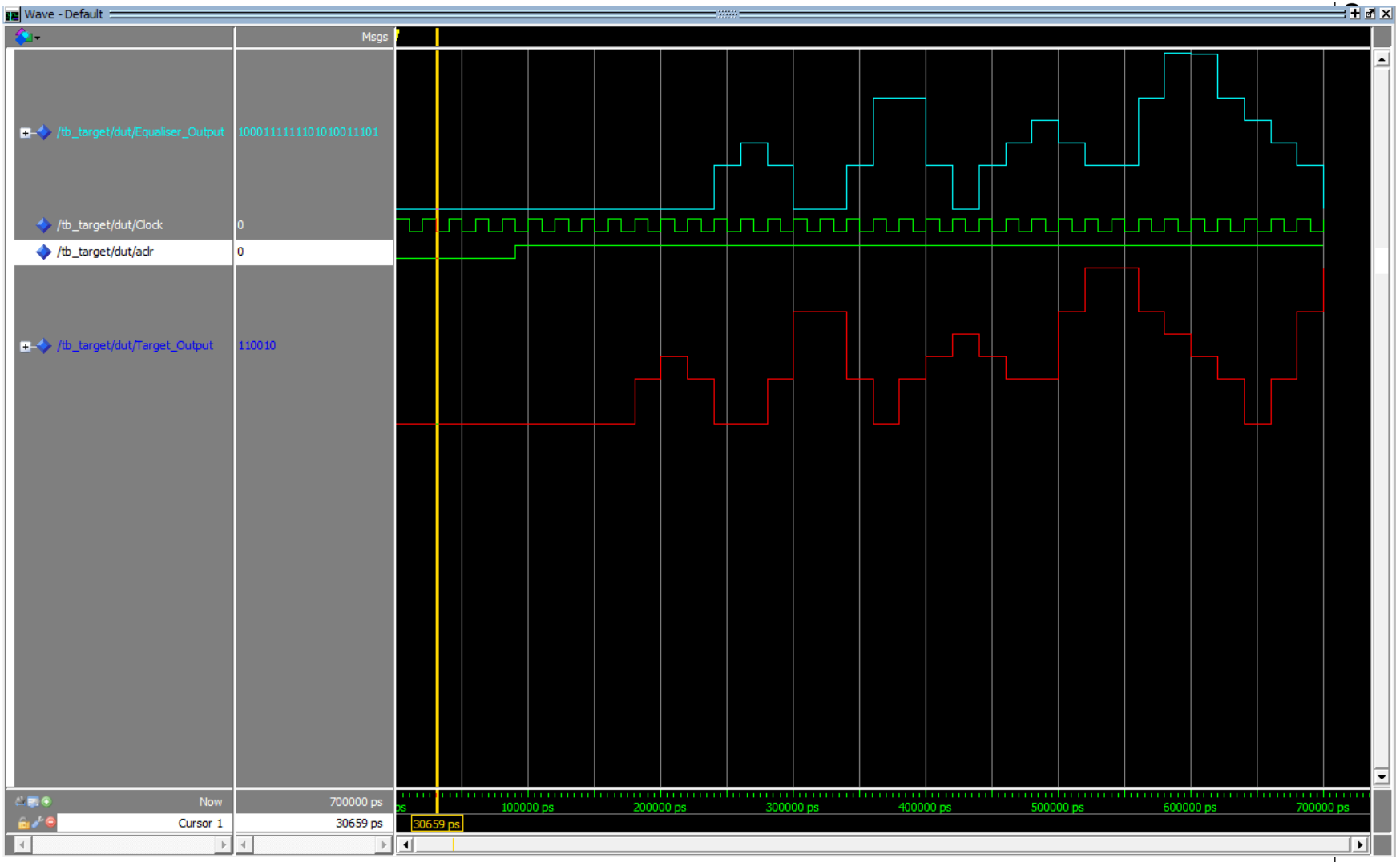

Figure 6.13: ModelSim Output of The Implemented PR Equaliser

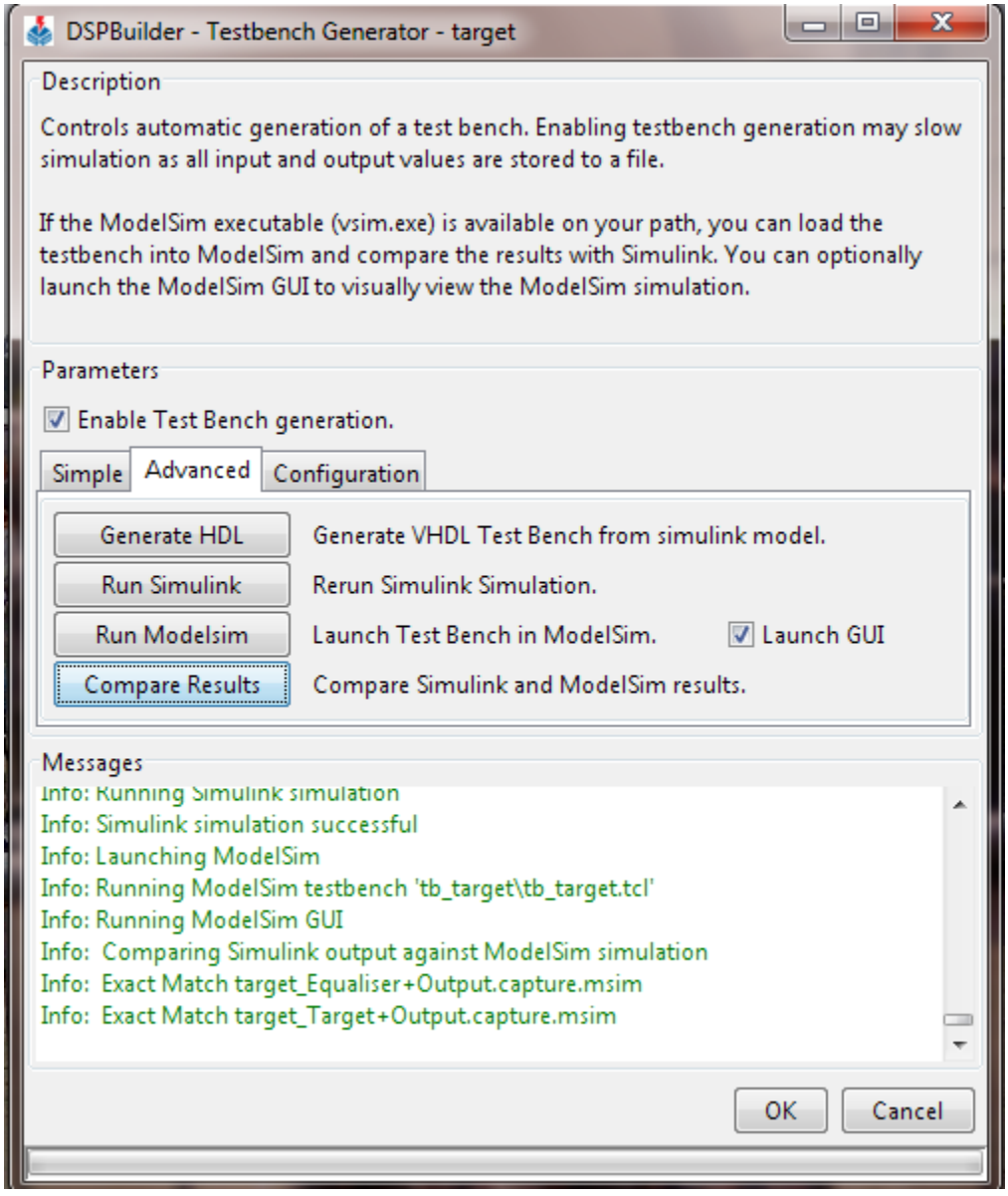

Figure 6.14: Screenshot of Running *Testbench* Block of PR Simulink Design

2- With AWGN

AWGN is added to the Simulink model of PR equaliser as shown in Figure 6.15. *Random Number* block from Simulink library is used to generate normally (Gaussian) distributed random signal. The mean is zero and the variance is 0.5006 (3-dB SRN). The channel output before adding AWGN and when AWGN is added is shown in Figure 6.16.

The Simulink design of PR equaliser with AWGN has been verified in ModelSim simulator. Figure 6.17 shows the output of ModelSim compered with Simulink output.

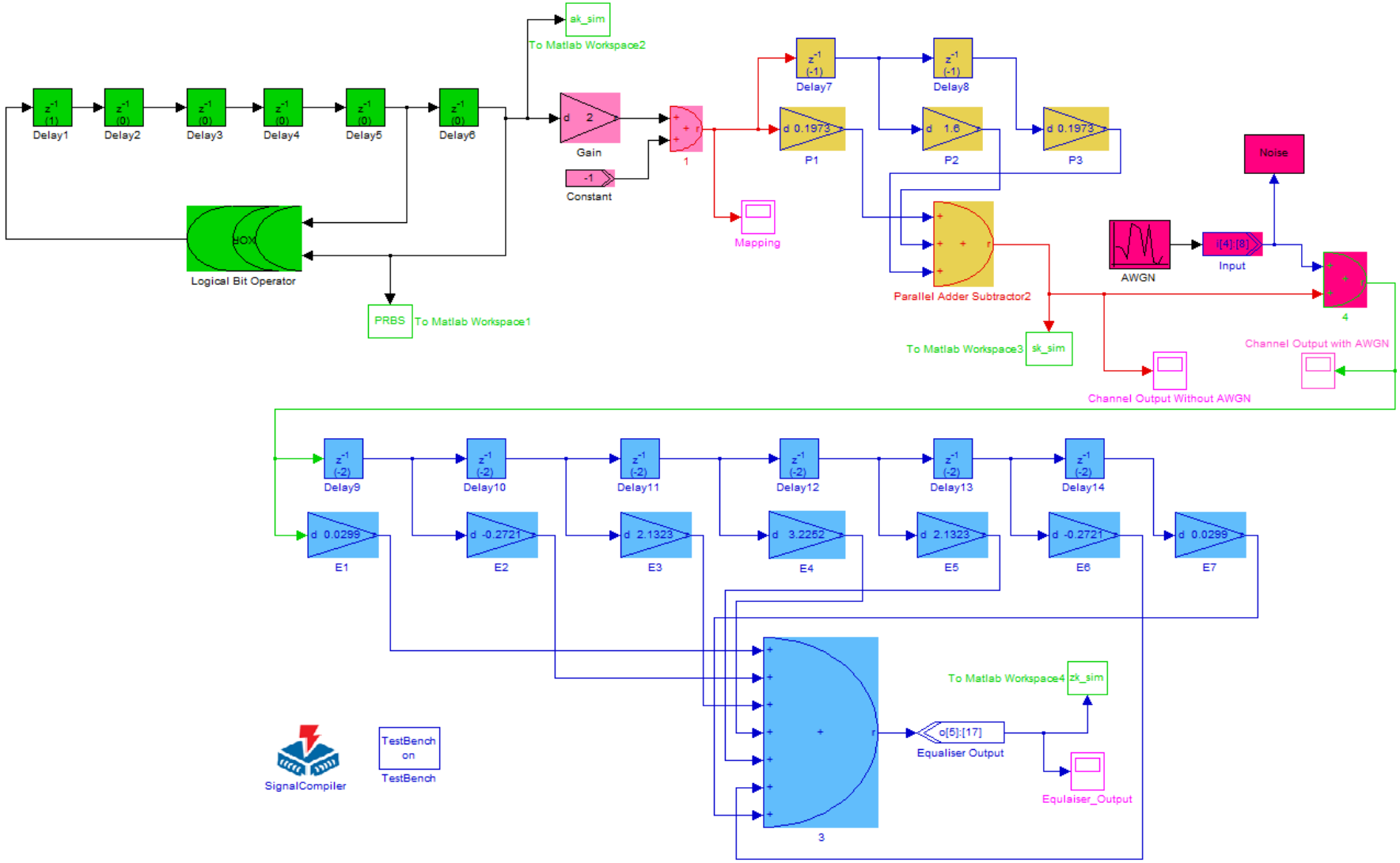

Figure 6.15: Complete DSP Builder Design of PR System With AWGN

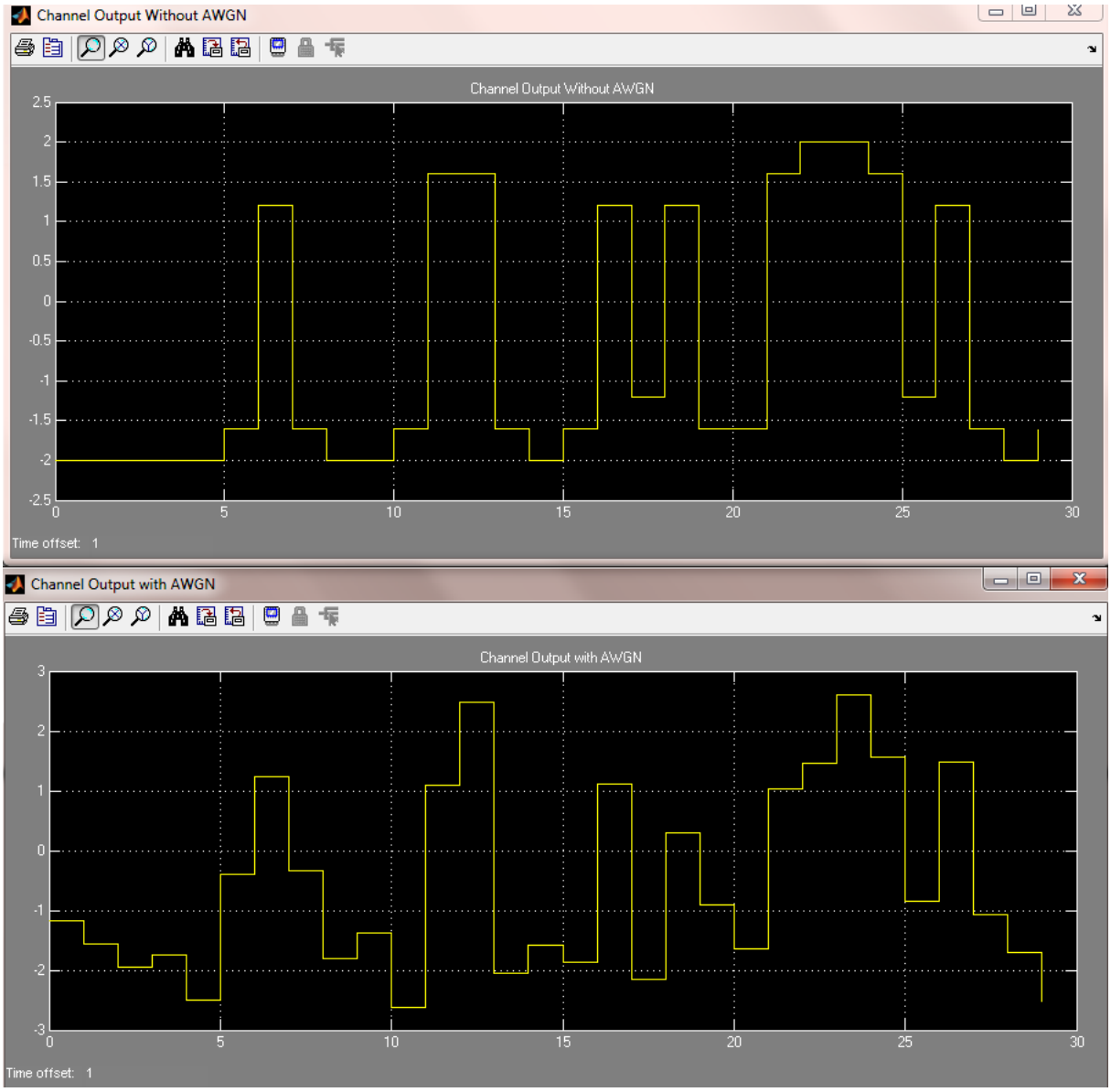

Figure 6.16: PMR Channel Output Without and With AWGN

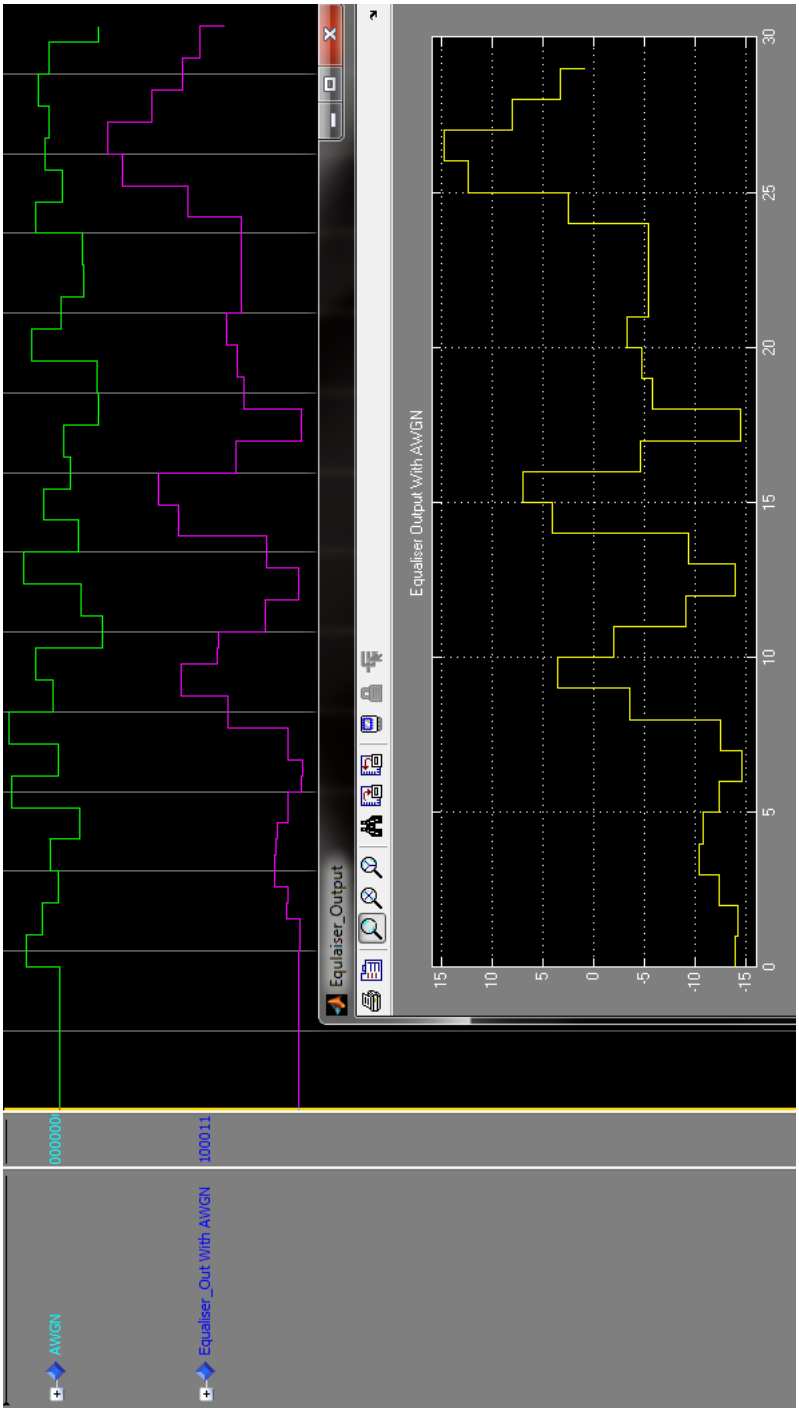

Figure 6.17: ModelSim and Simulink Output Sequences of PR System With AWGN

6.4.2 Verifying Equaliser Simulink Design in Hardware

The procedure of creating a Quartus II project for any Simulink/DSP Builder design model and then adding it into FPGA chip is explained in detail in Section 5.9.3. The PR Simulink design that is implemented in Quartus II software is shown in Figure 6.18. The necessary files associated with the VHDL code should be added into the Quartus II project of that design in order to compile the project correctly by Quartus II compiler.

Combination of schematic editor and VHDL code is used as a design entry for PR equaliser design. The design has two inputs which are *aclr* signal and clock signal as shown in Figure 6.19. A 50-MHz oscillator is used as a clock source to drive the design which is included in the DE1 board where (PLL) box is the clock signal implementation of the design. Therefore, two pins are needed to assign the input signals. Also, the design has 22-bit output signal which means 22 pins are required to assign the output signal. One bit for the sign and four bits for the magnitude and 17-bit for the fractional part to get the required accuracy.

The digital output signal of the equaliser is sent from FPGA chip into ST board through data cable. Some of the expansion header pins have been assigned to ST board pins for that purpose. The total pins required are 24, one pin for the clock signal and another one for *aclr* signal and 22 pins for the output signal. FPGA chip pin assignments of PR equaliser design model is shown in Figure 6.20.

Figure 6.21 shows the compilation report of the PR equaliser Quartus II design. The report shows that 3% has been used from the total logic elements of FPGA chip for that design.

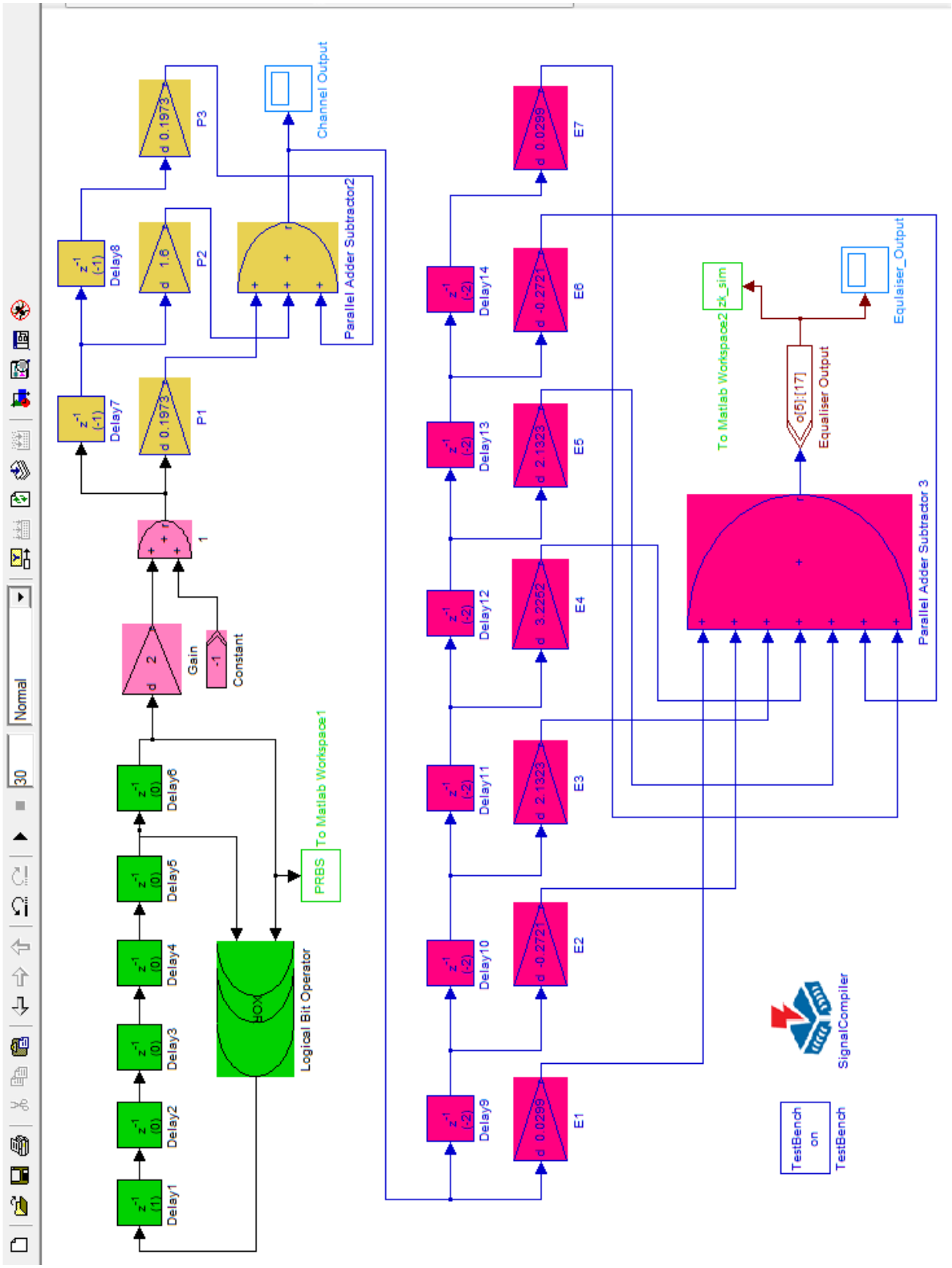

Figure 6.18: Simulink Design of the PR Equaliser

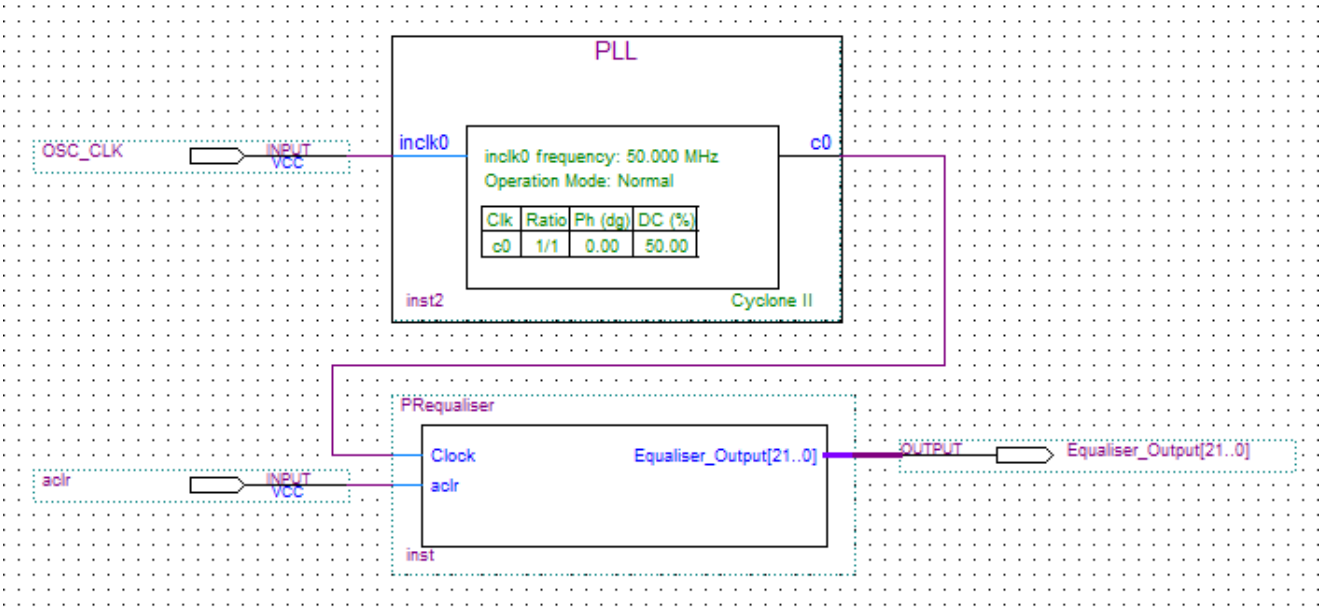

Figure 6.19: Quartus II Design of the PR Equaliser

Node Name	Direction	Location	I/O Standard	Reserved	Current Strength	I/O Bank	VREF Group
aclr	Input	PIN_L19	3.3-V LV...default)		24mA (default)	5	B5_N1
Equaliser_Output[21]	Output	PIN_E22	3.3-V LV...default)		24mA (default)	5	B5_N0
Equaliser_Output[20]	Output	PIN_E21	3.3-V LV...default)		24mA (default)	5	B5_N0
Equaliser_Output[19]	Output	PIN_D22	3.3-V LV...default)		24mA (default)	5	B5_N0
Equaliser_Output[18]	Output	PIN_D21	3.3-V LV...default)		24mA (default)	5	B5_N0
Equaliser_Output[17]	Output	PIN_C22	3.3-V LV...default)		24mA (default)	5	B5_N0
Equaliser_Output[16]	Output	PIN_C21	3.3-V LV...default)		24mA (default)	5	B5_N0
Equaliser_Output[15]	Output	PIN_B20	3.3-V LV...default)		24mA (default)	4	B4_N0
Equaliser_Output[14]	Output	PIN_A20	3.3-V LV...default)		24mA (default)	4	B4_N0
Equaliser_Output[13]	Output	PIN_B19	3.3-V LV...default)		24mA (default)	4	B4_N0
Equaliser_Output[12]	Output	PIN_A19	3.3-V LV...default)		24mA (default)	4	B4_N0
Equaliser_Output[11]	Output	PIN_B18	3.3-V LV...default)		24mA (default)	4	B4_N0
Equaliser_Output[10]	Output	PIN_A18	3.3-V LV...default)		24mA (default)	4	B4_N0
Equaliser_Output[9]	Output	PIN_B17	3.3-V LV...default)		24mA (default)	4	B4_N1
Equaliser_Output[8]	Output	PIN_A17	3.3-V LV...default)		24mA (default)	4	B4_N1
Equaliser_Output[7]	Output	PIN_B16	3.3-V LV...default)		24mA (default)	4	B4_N1
Equaliser_Output[6]	Output	PIN_A16	3.3-V LV...default)		24mA (default)	4	B4_N1
Equaliser_Output[5]	Output	PIN_B15	3.3-V LV...default)		24mA (default)	4	B4_N1
Equaliser_Output[4]	Output	PIN_A15	3.3-V LV...default)		24mA (default)	4	B4_N1
Equaliser_Output[3]	Output	PIN_B14	3.3-V LV...default)		24mA (default)	4	B4_N1
Equaliser_Output[2]	Output	PIN_A14	3.3-V LV...default)		24mA (default)	4	B4_N1
Equaliser_Output[1]	Output	PIN_B13	3.3-V LV...default)		24mA (default)	4	B4_N1
Equaliser_Output[0]	Output	PIN_A13	3.3-V LV...default)		24mA (default)	4	B4_N1
OSC_CLK	Input	PIN_L1	3.3-V LV...default)		24mA (default)	2	B2_N1

Figure 6.20: FPGA Chip Pin Assignments of PR Equaliser Design

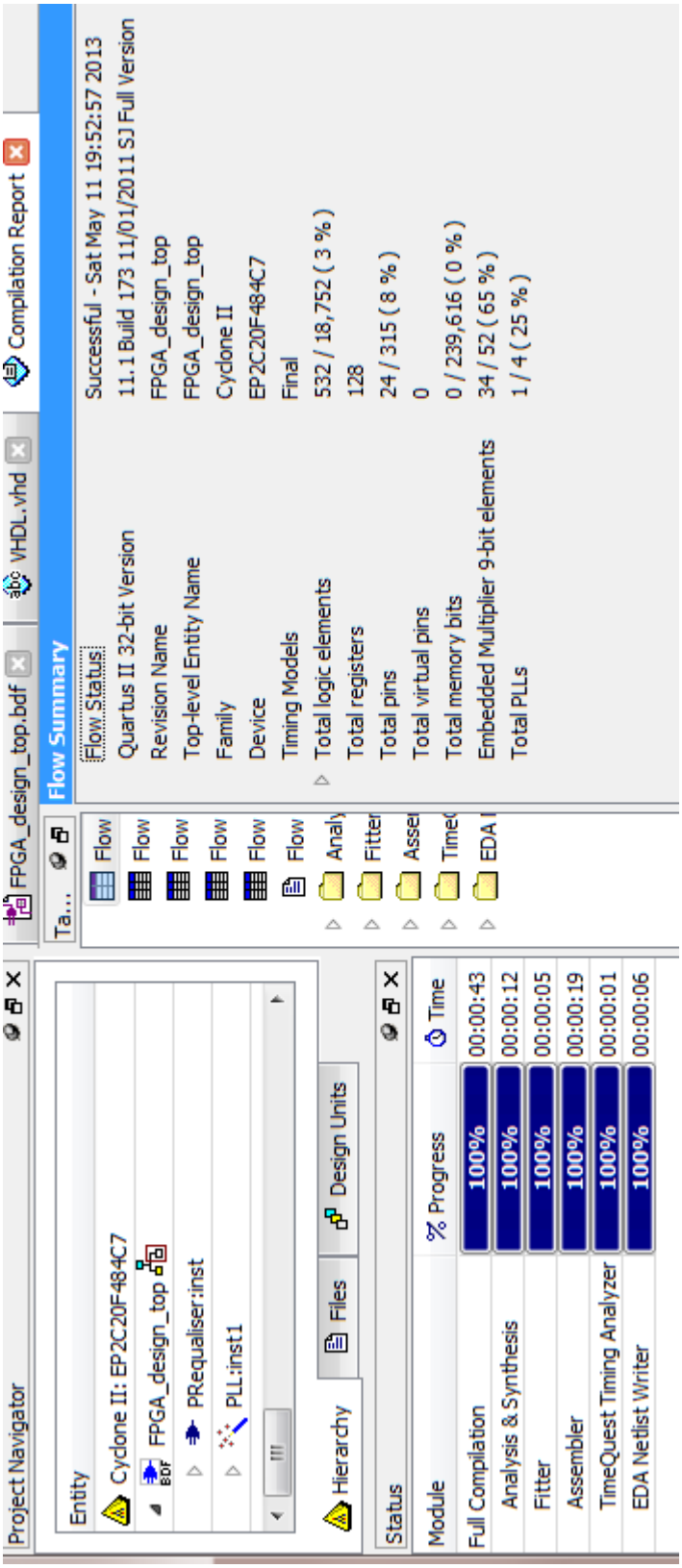

Figure 6.21: Compilation Report of PR Equaliser Quartus II Design

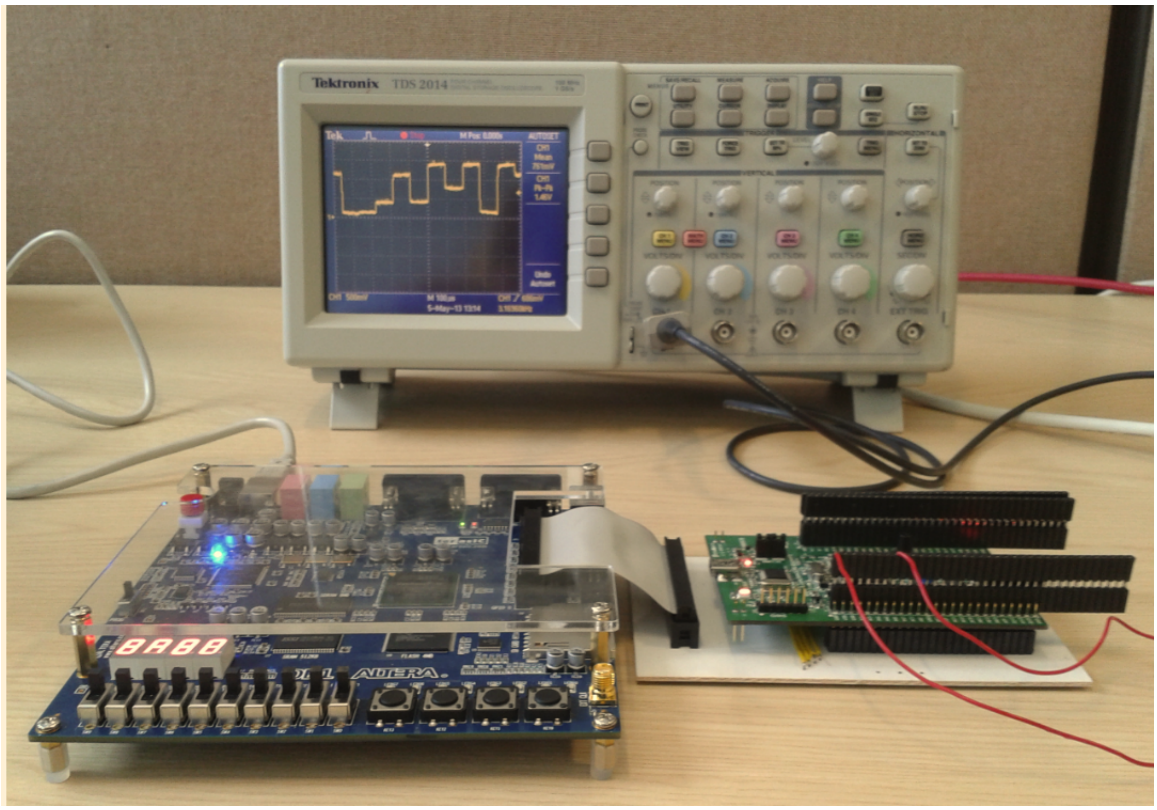

Figure 6.22: Hardware Verifying of PR Equaliser Design Model

6.4.3 Running Equaliser Design in FPGA Chip

The final step is downloading the PR equaliser design into FPGA board to verify the design. The 22-bit data digital output sequence of FPGA chip is sent to the ST board through 40-pin Expansion Header. The Expansion header of DE1 board is mapped into ST board through data bus as shown in Figure 6.22. Then this digital sequence is converted into analog sequence by 12-bit data DAC of ST board. FPGA board and ST board clocks have been set at 50MHz. Figure 6.22 shows the PR equaliser output result for a specific input sequence. Because the input sequence of the design model is randomly generated therefore a comparison between the Simulink output and FPGA output is unavailable.

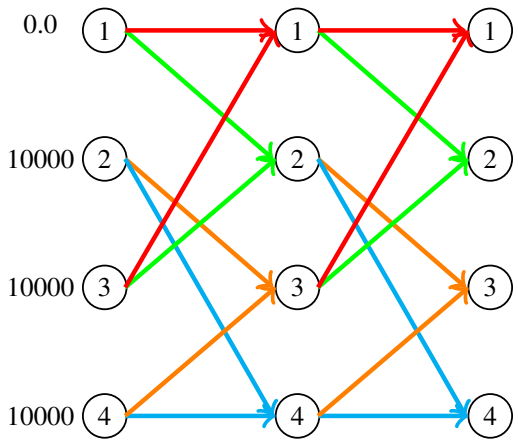

Figure 6.23: The Implemented GPR trellis in Simulink

6.5 Hardware Implementation of Viterbi Algorithm

Two segments of the PR system trellis are implemented in Simulink and then a Quartus II project is created in order to download the design into the FPGA chip. Simulink implementation and Quartus II project for that implementation are illustrated below.

6.5.1 Simulink/DSP Builder Implementation of VA

An assumption is made which is the initial value of the first state of the trellis is zero and the other states are set to some large number like 10000 [118]. Figure 6.23 shows the implemented trellis in a combination of Simulink and DSP Builder blocks.

The most important implementation in Simulink design of any trellis structure is the minimum operation. The *Min* block from Altera DSP Builder Advanced Blockset library and *MinMax* block from Simulink library would implement the minimum operation.

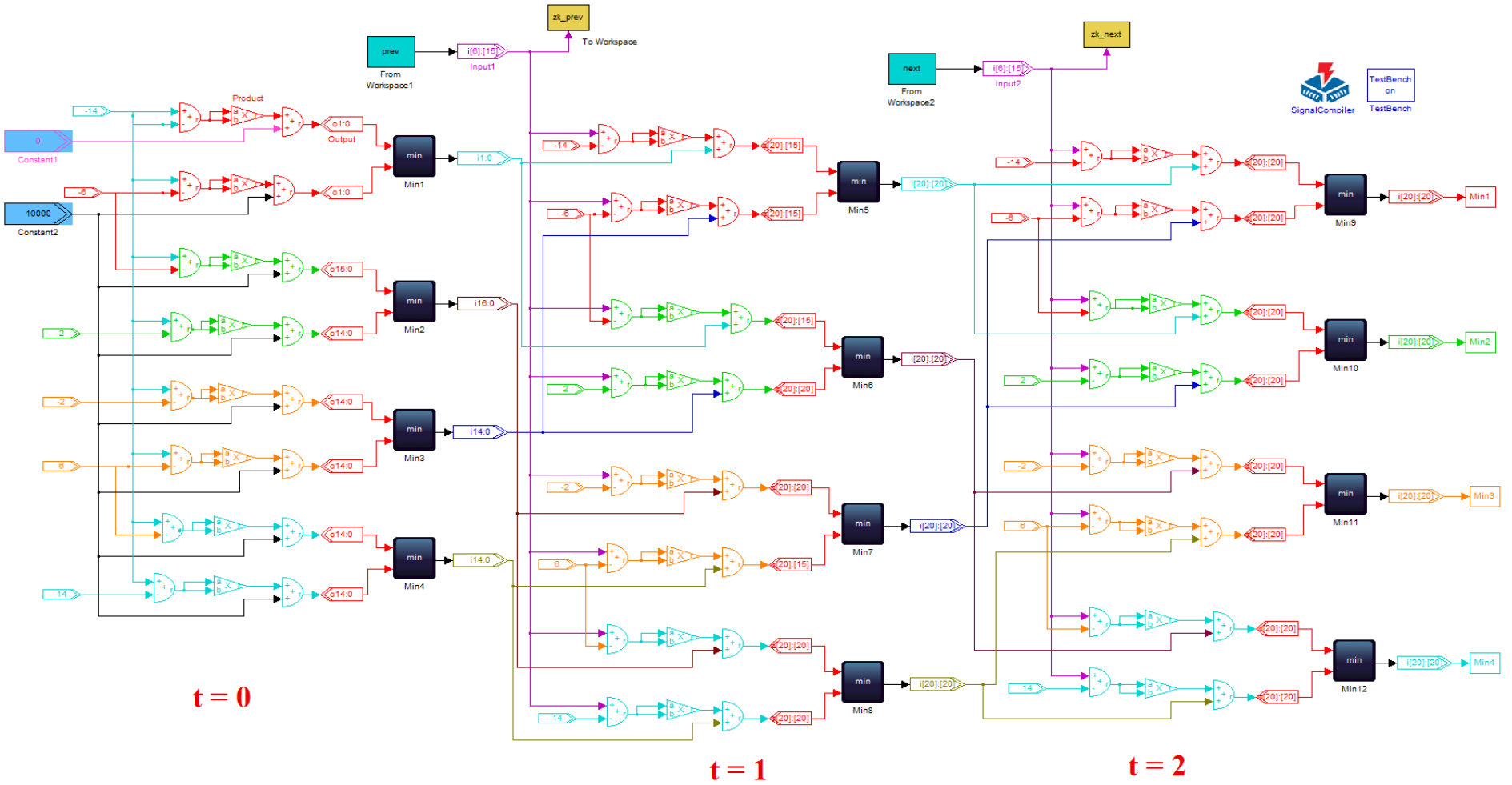

Figure 6.24: Simulink/DSP Builder Design of GPR Trellis Design Using *Min* Blocks

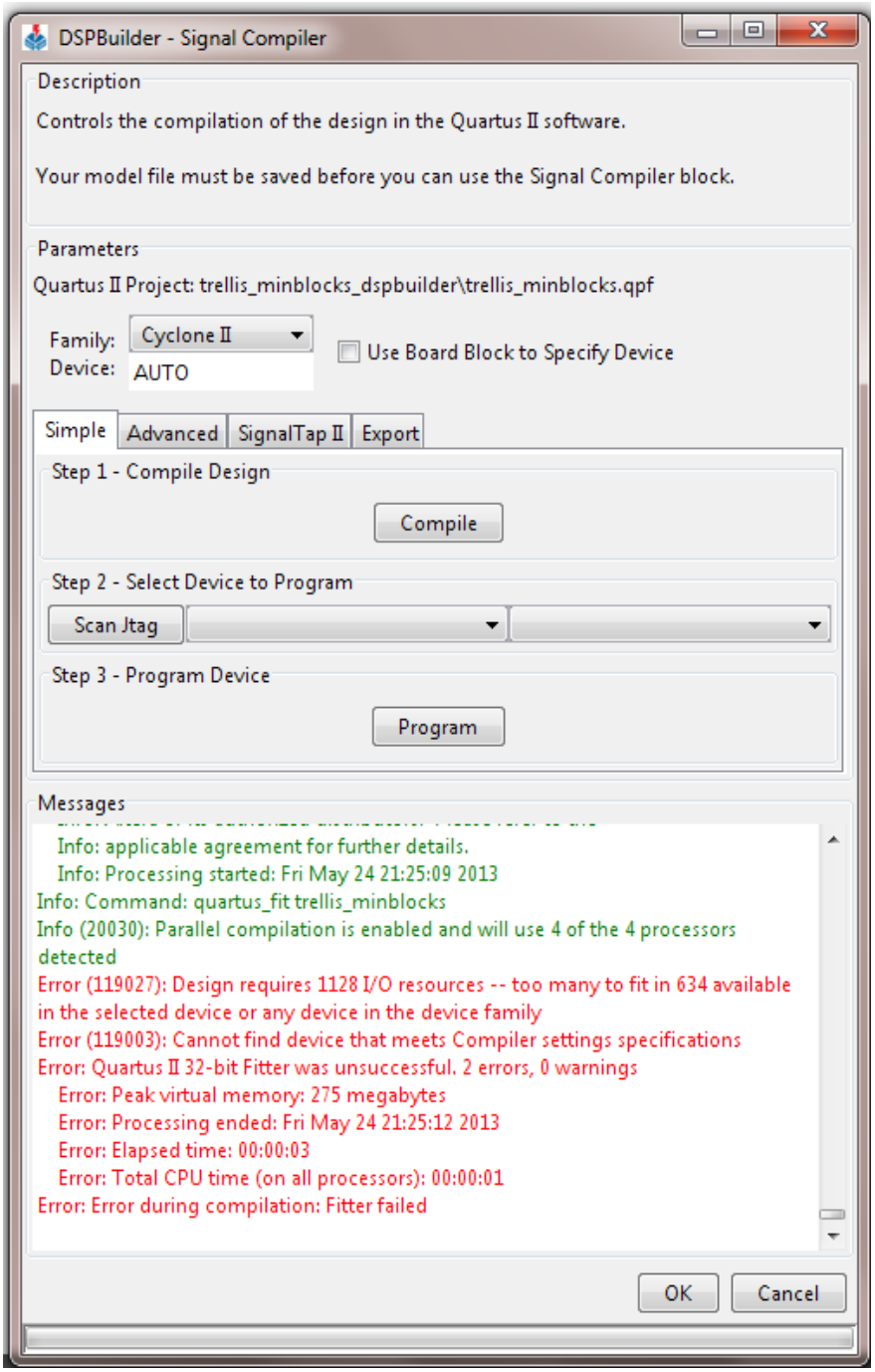

Figure 6.25: Compilation Report of VA Simulink Design Using *Min* Blocks

Implementation of VA by Using Min *Blocks*

Firstly, VA is implemented in Simulink using *Min* blocks as shown in Figure 6.24. Each *Min* block requires two *Output* ports and one *Input* port in order to cast the signal through it. The design consists of three stages. The first stage at ($t = 0$) initialises the trellis with the initial values that are mentioned above. The second stage at ($t = 1$) finds the previous minimums and at ($t = 2$) the next minimums are calculated. The 22-bit equaliser sequence of Simulink design is exported to MATLAB workspace and then is imported from MATLAB workspace as an input sequence to Viterbi detector. The equaliser sequence has been delayed by one sample in order to find the previous minimum of each state. In other words, the trellis has two equaliser sequences. The first sequence (*From Workspace1* block) is to calculate the pervious minimum values. The second sequence (*From Workspace2* block) is to find the minimums of the current states. Many *Input/Output* ports are required in order to cast the signal between Altera DSP Builder blockset and Altera DSP Builder Advanced blockset.

After running Altera *Compiler* block to compile the design, an error message has occurred which states that Quartus II Fitter was unsuccessful. According to that, the entire design will not fit in EP2C20 FPGA because the design requires 1128 I/O resources while the available I/O resources are 634. The screen capture from the *Compiler* block is shown in Figure 6.25.

However, implementation of VA in Simulink by using *MinMax* block from Simulink library would have the same compilation result because each *MinMax* block requires also two *Output* ports and one *Input* port.

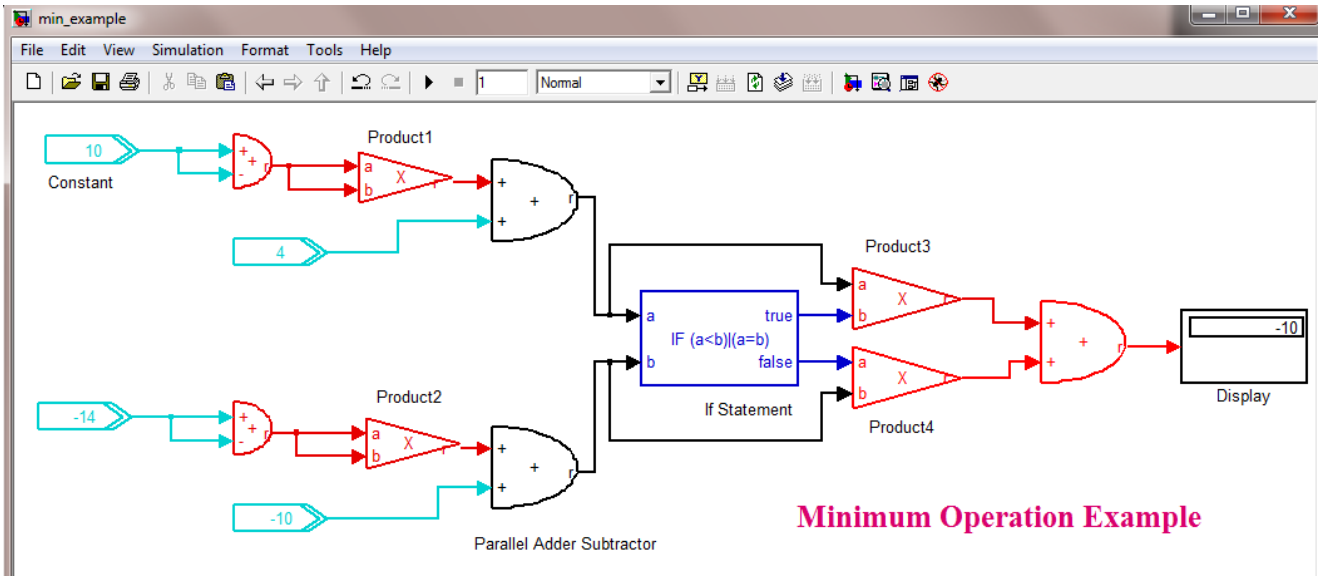

Figure 6.26: Example of Implementation One Minimum Operation of 4-state Trellis

Implementation of VA by Using IF Statement *Blocks*

In order to avoid the big numbers of I/O ports that used with *Min* blocks, *IF Statement*, two *Product* and *Parallel Adder Subtractor* blocks from Altera Blockset library are used to implement the minimum operation.

Figure 6.26 illustrates implementation of just one minimum operation of 4-state trellis using *IF Statement* block. The example shows that two *Product* blocks are needed in order to implement every minimum function.

Implementation of VA in Simulink using *IF Statement* blocks is shown in Figure 6.27. Also, the index of the state that has the minimum value at each time interval is implemented. Two *Product* blocks and two *Constant* blocks are used for index implementation. The *Constant* blocks have been parameterized into specific numbers. For instance, the index of the first and the second states is either 1 or 3. Therefore, the *Constant* blocks of those states have 1 and 3 values. However, the *constant* blocks of the third and the fourth states would have either 2 or 4 values.

Unfortunately, again the compilation process is failed because the design requires 408 Embedded multiplier block resources however the available blocks are 300 in EP2C20 chip as shown in Figure 6.28. Also, the compilation report states that this design will not fit any FPGA chip from Cyclone II family. Based on that to implement 4-state trellis in hardware, another device family with enough DSP blocks must be used. For example, 5CEA9 FPGA chip from Cyclone V family has 684 DSP blocks [119].

Implementation of VA in Simulink/DSP Builder

The VA has been described in Section 3.8 however the implementation of 4-state trellis in Simulink/DSP Builder is performed as follows:

- At $(t = 0)$, the four states are initialised with (0, 10000, 10000, 10000).
- At $(t = 1)$, the next stage of the trellis is constructed using the first sample of the equaliser output, z_k,. The initial values are added as the previous minimum values.
- Then, at $(t = 2)$ the second sample of z_k is received. New paths metric are calculated and the minimum values of the previous stage plus the initial values are added.

At each time interval, the initial values are added to the previous minimum values. This will effect the branch metric of each state and will lead to wrong computations. Consequently, there are two ways to implement the trellis. Either, implementation of 1000 stage for 1000 samples of z_k however this way is impractical because it requires a huge number of multiplier blocks or writing HDL code and then verifying it in hardware. Therefore, the only way to verify VA in hardware is by using the high level description language (HDL).

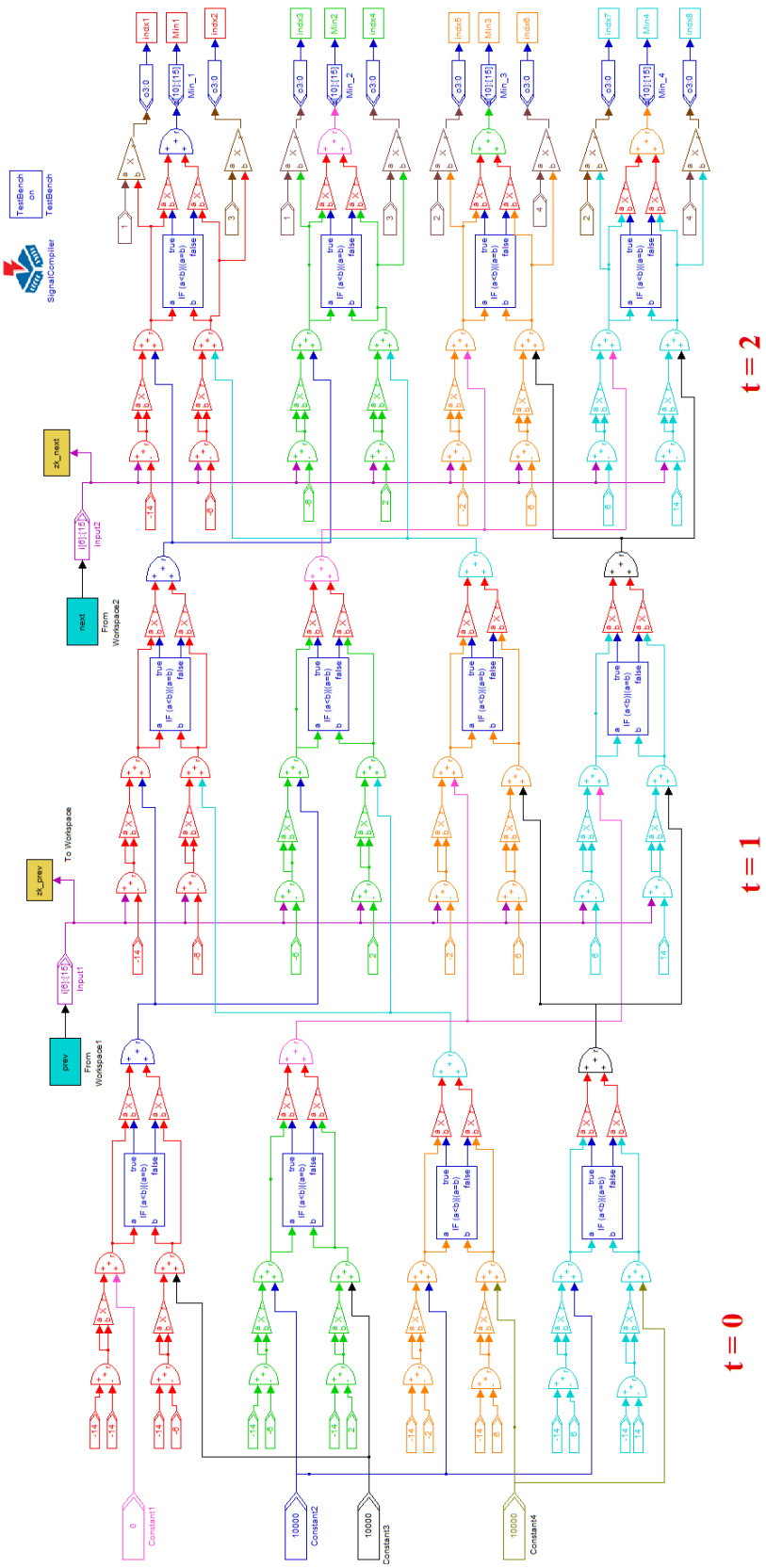

Figure 6.27: Simulink/DSP Builder Design of GPR Trellis Design Using IF Statement Blocks

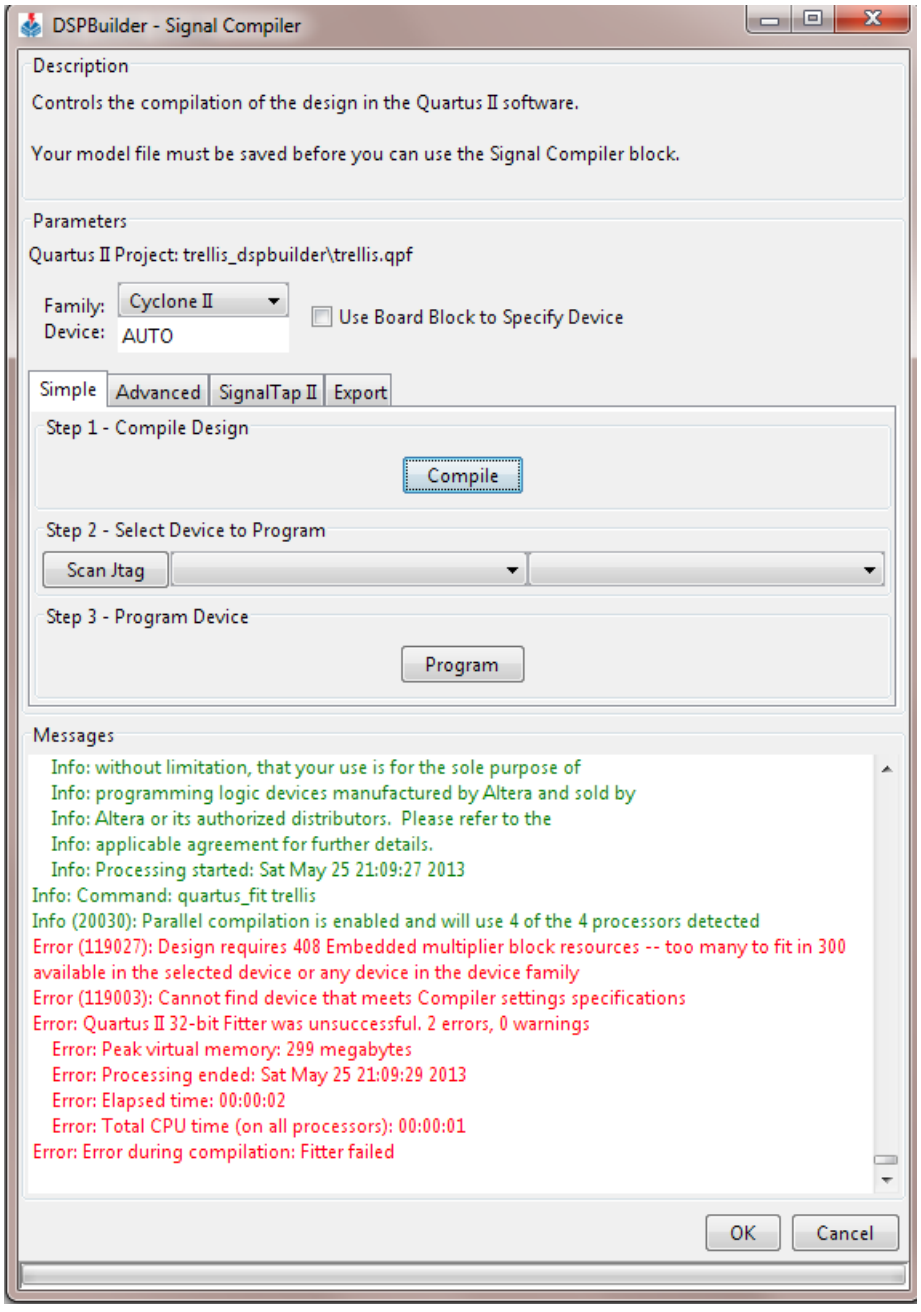

Figure 6.28: Compilation Report of VA Simulink Design Using IF Statement Blocks

6.5.2 Verifying VA Simulink Design in Hardware

In order to specify exactly the VA Simulink design requirements, Quartus II project is created as shown in Figure 6.29. However, this design has not been verified in hardware because the design uses 398 DSP blocks while the blocks available are only 26 DSP blocks. The compilation report shown in Figure 6.30 illustrates that and it also indicates that the total logic elements have been used for this design are 70% and the total used pins are 167 from 315 available pins.

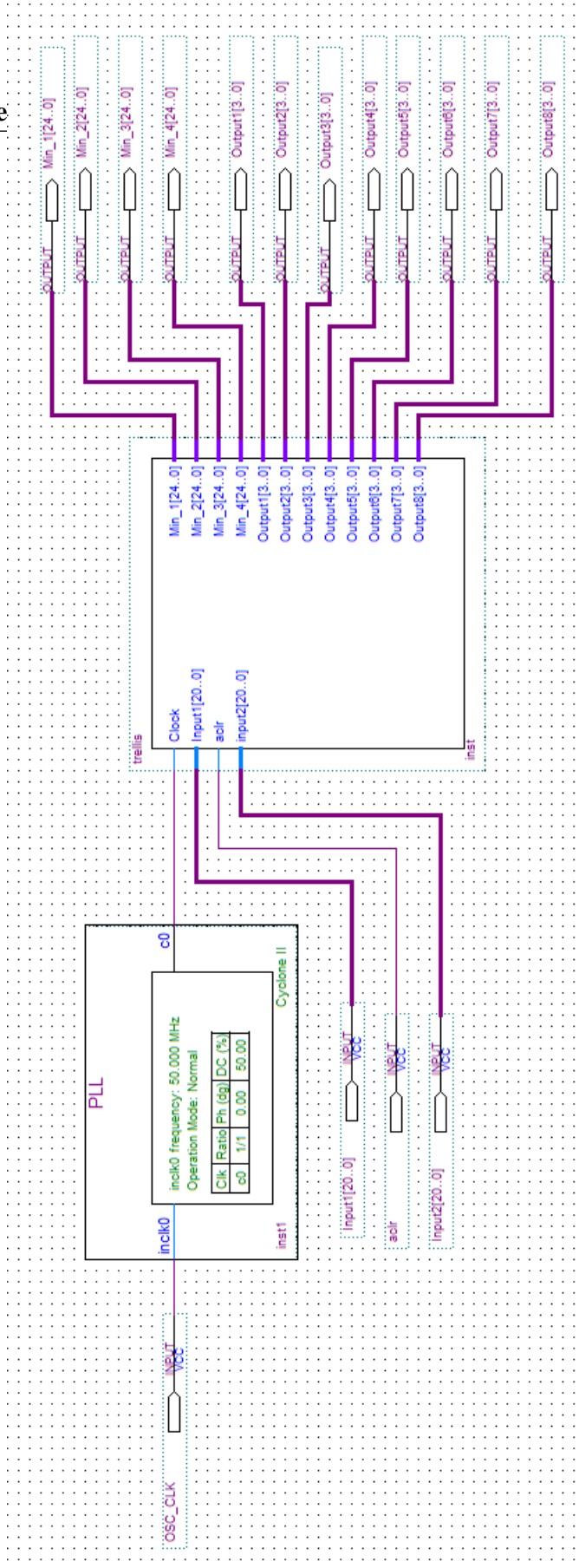

Figure 6.29: Quartus II Design of VA Simulink Design

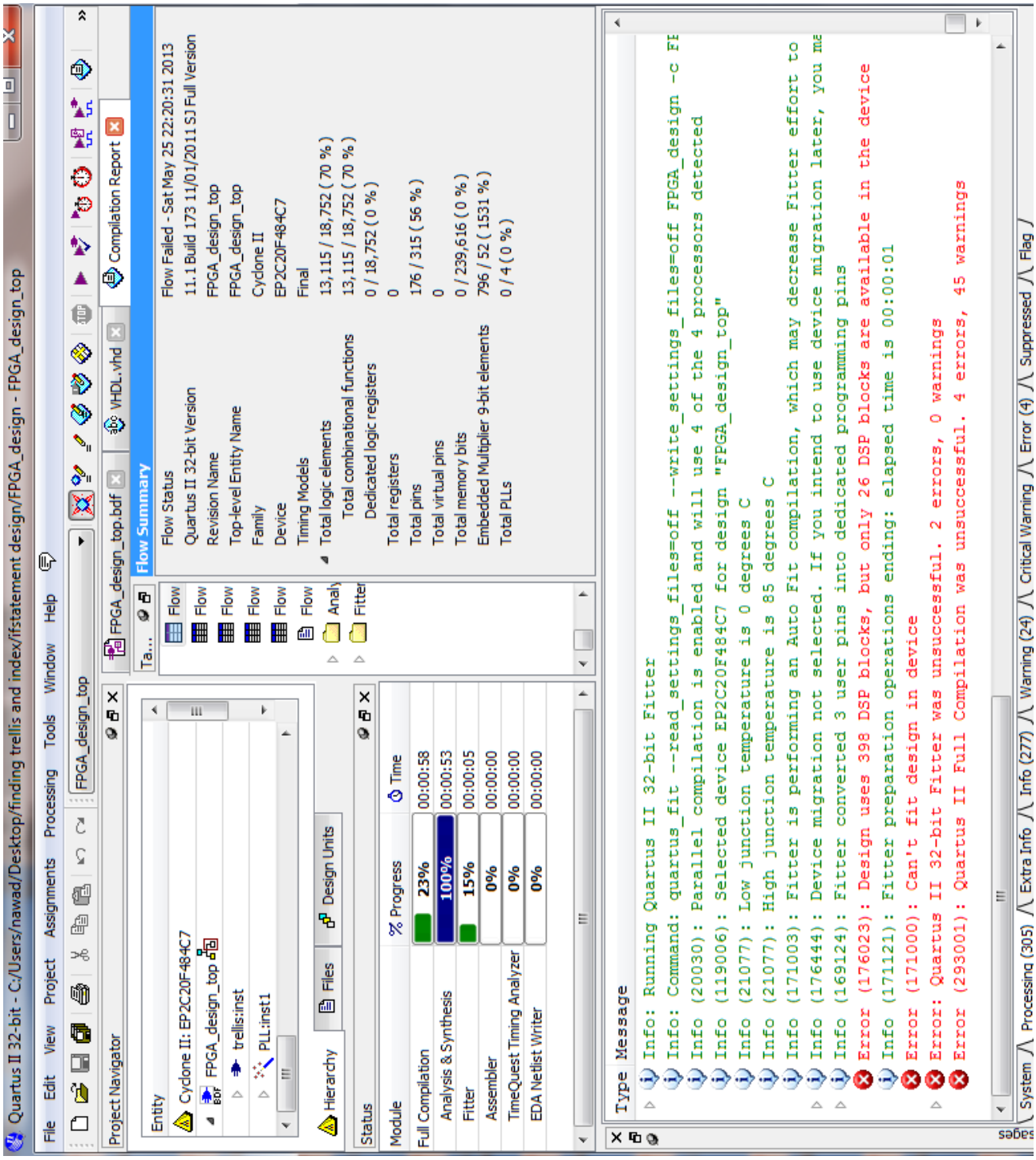

Figure 6.30: Compilation Report of VA Quartus II Design

6.6 Summary

- The PRML system components including PMR channel, PR equaliser without and with AWGN, the ideal channel (target) and VA have been implemented in Simulink/DSP Builder softwares and in Quartus II software.

- The PMR channel has been implemented in Simulink and then verified in hardware. The design fits comfortably in EP2C20 FPGA as it occupies only 1% from the total logic elements. The output results of Simulink, ModelSim and Quartus II softwares are matched.

- The PR equaliser without AWGN and with AWGN are also implemented in Simulink and then in hardware. The entire design occupies 34 embedded multiplier 9-bit elements and 3% from the total logic elements which means the design fits in EP2C20 FPGA. The equaliser output is sent from FPGA chip into the ST board via data cable. The DAC of ST board is used to convert the digital sequence of the equaliser output into analog signal. The output results of PR equaliser from Simulink, ModelSim and FPGA board are matched.

- Different Simulink designs have been constructed for 4-state trellis structure.

- The first design is built using *Min* blocks to implement the minimum operations of the trellis. Unfortunately, the compilation process has been failed because the design had large number of I/O ports which exceeded the available limit of these ports.

- Then, *Min* blocks have been replaced with *IF Statement*, two *Product* and *Parallel Adder Subtractor* blocks. Again, the compilation process has been failed because the design requires a lot of multiplication blocks that are held within the DSP blocks. The compilation report states that the number of DSP blocks that are needed for that

design has exceed the number of DSP blocks of EP2C20 chip. The compilation of Quartus II design has been also failed for that reason.

- The VA Simulink design had some wrong computations because the initial values were added at each time interval which effected the computations of each branch metric. Consequently, there are two ways to implement VA with correct computations. The first way is implementation a trellis starting at $t = 0$ and ending at $t = 1000$ in order to recover the 1000-sample recorded sequence, z_k. However, this way is impractical as it requires thousands of DSP blocks. The second way is writing HDL code.

  Based on that, for the complicated designs the Simulink/DSP Builder softwares are not efficient however these softwares are very ideal for some applications like FIR filters designs. Therefore, the eligible solution for implementation such complex designs is using high level description languages such as Verilog or VHDL code.

- The Simulink blocks can be used only for simulation purposes because these blocks will not be converted into HDL scripts.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

The advances in recording materials, read/write heads, mechanical designs and signal processing tools have played a significant role in increasing the recording densities and the data rates of magnetic recording systems [10]. Signal processing aspects of magnetic recording systems include modulation and equalisation techniques, decoding algorithms, consideration of noise types like ISI and ITI and other variuos factors [54].
This research work investigates, discusses and analyses equalisation and decoding techniques for ISI and ITI based magnetic recording systems. Furthermore, it provides a background investigation into the hardware complexity of the perpendicular magnetic recording system.

In current conventional storage technology, perpendicular way recording, the data is written in concentric tracks where the width of each track is governed by the width of the write head [61]. The track pitch (center-to-center spacing of each track) is slightly larger than the track width. The tracks are separated by guard bands where the guard band width equals to the track pitch minus the track width [61]. However, the track densities are determined in terms of number of concentric tracks per radial length in the disk.

During retrieval the recorded data, the read back head is precisely located on a single track. Because the width of the reading head is equal or smaller than the width of the

written track, the read back head reads the recorded data from only that single track with no interfering from the adjacent tracks. The major advantage of perpendicular recording technology is that each track could be randomly written at any time without disturbing the data of the neighboring tracks.

However, a limitation of how much information can be stored on the HDD is presented by this storage technology. In 2000, a paper was published [50] predicating the limit of perpendicular recording technology which is around 1 Tbit/in^2.
The limiting factor is the onset of the superparamagnetic limit (thermal stability) [50] which is a trade-off between media SNR, and the thermal stability of small grain media and the write ability of a narrow track head [8]. Whereas, increasing the recording density requires shrinking the grains of the magnetic media further, this reduction in the grain size is governed by the superparamagnetic limit. The superparamagnetic effect causes erasing the recorded data spontaneously after a short period of time [10].

The alternative technique that can replace the conventional one is SMR technology. This technology can be built on the existing one with few changes to the write head [9]. The concept of SMR method is the overlapping [120] [121] where each written track partially overlaps an immediately adjacent track that is contiguous to it [61]. The write head can be larger than the track pitch (unlike the perpendicular method) while the width of read head is slightly less than the track pitch. The read back head picks up signals from the track under it and from the neighboring tracks [61]. Consequently, a sever ITI could result which degrades the system performance.

The primary research aim of the thesis is investigation, analysing, discussion of shingled recording system and then reducing the decoding complexity of this system.
The research was carried out by:

- Investigation and implementation of one-track one-head system model for PMR system.
- Extending the first model into two-track one-head system model (SMR system) and then investigation and implementation of that system.
- Implementation of the first system model in hardware in order to verify the obtained results, measure and understand the hardware complexity of the SMR system.

One-track one-head model consists of two parts: the first part is the perpendicular recording channel and the second part is PRML read channel which applies for recovering the recorded data. A sequence of random binary data which represents the digital data that intend to record on the magnetic media is fed to the system channel. Replay samples have been generated from the channel and then have been equalised by a digital filter equaliser to the ideal samples that are generated by a particular target. MMSE method is used in order to find the target that suits the channel of that system model. Finally, the equalised samples are applied to the decoder to recover the written data.
The significant part of any read channel model is the data recovery process. Therefore, the ability of the read channel to recover the recorded data reliably should be verified. The verification has been accomplished in two ways. The first way is by feeding the read channel system with samples that have no any type of noise where the recorded data has been recovered correctly without any error. The second way is by adding some types of noise to the replay samples of that system. AWGN is added to approximate the electronic noise in the head and preamp circuits which is assumed the dominate source noise in one-track

one-head model. The system performance has been verified through BER against SNR graph.

However, MAP algorithm is not valid for systems that have no noise (σ=0) because this algorithm is based on transition probabilities calculations which depend on noise power.

For perpendicular magnetic systems, the main signal energy from readback signal is at low frequencies. Therefore, the GPR targets are suitable more than PR targets because the later would filter the low-frequency energy which is the useful energy. Thus, a GPR target (ideal channel) is applied to the system model.

In order to investigate the ITI in shingled systems, two-track one-head system model has been constructed to have two interfering tracks. An assumption is made which is the head picks up the signal from the track under it and some signal from the adjacent track with no interaction between groups of tracks. Two equalisation/detection techniques have been investigated.

The first one has considered ITI as noise that is added to the AWGN of the system. The results showed that for less than 30%, the ITI has significant effect as a dominant noise however for higher amount than this the ITI has no significant effect.

The second equalisation/detection technique has considered the ITI amount as a signal coming from the adjacent track. Consequently, 16-state novel trellis with four outgoing paths from every state and four incoming paths to every state has been constructed for that system. The trellis of the decoder has very high complexity. Therefore, an intensive work was required in order to reduce this complexity. Six novel decoding techniques have been applied to the two-track one-head system in order to find the ML sequence. The four branches of each state have been reduced into two branches with further simplifications.

- The performances of the six decoding techniques have been found in terms of BER graphs. The graphs showed that, the first technique (without simplification) had a better performance in terms of BER values. However, the others approximations have different performances but the one common factor approximation had a very close performance to the optimal one.

- Also, the performances of the decoding techniques have been found in terms of the complexity where the complexity of each method has been measured and compared.

In this work, the complexity is defined as the number of arithmetic operations and the number of comparison operations. The tables of results showed that, one common factor approximation had a lower complexity compared to optimum technique. In addition, it has been found that some operations that do not depend on the equalised sample can be completed off-line (before sampling process) however the operations that require the equalised sample should be completed on-line. Consequently, the decoding complexity of one common factor approximation has been reduced by more than three times with 0.5 dB penalty. The results showed that a trade-off between the BER performance and the complexity of the system can be made.

In order to investigate the complexity of two-track one-head system further and measure it, one-track one-head system has been implemented in hardware. However hardware implementation of two-track one-head system has been proposed for future work.
Four hardware architectures have been designed in Simulink which are: PMR channel, digital filter equaliser without and with AWGN and digital filter target architectures. These four designs have been fitted comfortably in the FPGA chip which means they have been verified in hardware. The channel architecture has occupied only 1% of the total DSP

blocks of the FPGA chip while the equaliser architecture without and with AWGN have occupied 3%. The output results of MATLAB, Simulink, ModelSim softwares were matched.

The PR equaliser design have been run on DE1 board. The expansion header of DE1 board were mapped into ST board through data cable in order to convert the digital output sequence of FPGA chip into analog signal via DAC of ST board.

Two different architectures have been designed in order to implement VA. The first architecture was implemented by using *Min* blocks from Altera DSP Builder Advanced Block set library. Quartus II fitter for this design was unsuccessful because it requires 1128 I/O resources while the available I/O resources are 634.

Therefore, the *Min* blocks have been replaced with *IF Statement*, two *Product* and *Parallel Adder Subtractor* blocks where each minimum operation requires two *Product* blocks. The compilation process of this design was also unsuccessful because the design requires a lot of multipliers that are held within the DSP blocks. The design required 408 Embedded multiplier blocks however the available blocks are 300 which exceeded the capability of the FPGA chip that used in this research.

Furthermore, VA is built on path metric calculations where each path is calculated by adding the current state metric into the previous one. That means, at every time interval, the previous and the next minimum values are changing which makes the design very complicated.

Therefore, there are two different methods to implement VA in hardware which are:

- The first method is the construction of 1000 stages in order to recover the 1000-sample of the recorded sequence. However, this method would require a huge number of multipliers to accomplish the minimum operations which exceeds the capability of any FPGA chip.

- The second method is writing HDL code for that implementation.

Replacing some traditional HDL based designs by Simulink/DSP Builder based design flow, would allow over 1 million MATLAB users to design FPGAs and offer a significant reduction in development time. However, this research work has shown that Simulink/DSP Builder based design is not valid for the complex designs. Consequently, writing HDL code using either VHDL or Verilog languages would be the eligible solution for such designs.

7.2 Future Work

- To extend this research work, it would be interesting to write HDL code for 4-state trellis VA in order to verify the one-track one-head system in hardware which will allow finding the BER graph performance. Definitely, implementation of PRML system in hardware is much faster than MATLAB simulation. Where for 50-MHz clock signal, the FPGA chip would perform 50 million simulations per second.

- Investigation of the complexity of two-track one-head system and reducing it further as a first step. Then, implementing this system in hardware as a second step. However, verifying this system in hardware is exactly the same as verifying the one-track one-head system except the trellis will be more complicated than 4-state trellis.

- Some techniques have been proposed in order to ensure the continued rapid increases in the capacity of HDD such as BPM and HAMR and MAMR. These technologies

involve significant engineering challenges [8] and require the media to be radically redesigned [50].

However, there is a fourth technique which has an advantage of extending the use of the perpendicular method media and read/write head [8] [50]. The technique combines SMR with 2-D readback (TDMR) where signal processing is accomplished in two dimensions using information available across several adjacent tracks [9]. Therefore, it would also be interesting to investigate this technique as one of the promising technologies that would increase the areal density to the order of 10 Tbit/in^2.

Bibliography

[1] Altera Corporation. Cyclone II Device Handbook, Volume 1. *Chapter 12: Embedded Multipiers in Cyclone II Devices*, February 2007.

[2] Kevin Craig, Professor of Mechanical Engineering, Rensselaer Polytechnic Institute. A Mechatronic Marvel: The Computer Hard Disk Drive, Past-Present-Future, November 2007.

[3] tom'sHARDWARE. Hard Drive Technology @ONLINE. http://www.tomshardware.com/reviews/hitachi-7k1000-terabyte-hard-drive,1584-3.html, 17 April 2007.

[4] The Free Dictionary. Perpendicular Recording @ONLINE. http://encyclopedia2.thefreedictionary.com/Perpendicular+magnetic+recording.

[5] Hitachi. Hitachi Global Storage Technologies @ONLINE. https://www1.hgst.com/hdd/technolo/overview/chart13.html.

[6] Akira Kikitsu. Prospects for bit patterned media for high-density magnetic recording. *Journal of Magnetism and Magnetic Materials*, 321(6):526 – 530, 2009.

[7] Robin Harris. Engineering the 10 TB notebook drive @ONLINE. http://www.zdnet.com/blog/storage/engineering-the-10-tb-notebook-drive/194, 27 September 2007.

[8] Y. Shiroishi, K. Fukuda, I. Tagawa, H. Iwasaki, S. Takenoiri, H. Tanaka, H. Mutoh, and N. Yoshikawa. Future options for HDD storage. *Magnetics, IEEE Transactions on*, 45(10):3816 –3822, October 2009.

[9] R. Wood, M. Williams, A. Kavcic, and J. Miles. The feasibility of magnetic recording at 10 terabits per square inch on conventional media. *Magnetics, IEEE Transactions on*, 45(2):917 –923, Febuary 2009.

[10] Vasic, B., and Kurtas E. M. *Coding and Signal Processing for Magnetic Recording Systems*. CRC Press, United States of America, 2005.

[11] J. Moon. The role of SP in data-storage systems. *Signal Processing Magazine, IEEE*, 15(4):54–72, 1998.

[12] S. X. Wang and A. M. Taratorin. *Magnetic Information Storage Technology*. Academic Press, 1999.

[13] G.D. Forney. Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference. *Information Theory, IEEE Transactions on*, 18(3):363–378, May 1972.

[14] G.D. Fisher, W.L. Abbott, J.L. Sonntag, and R. Nesin. PRML detection boosts hard-disk drive capacity. *Spectrum, IEEE*, 33(11):70–76, 1996.

[15] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv. Optimal decoding of linear codes for minimizing symbol error rate (corresp.). *Information Theory, IEEE Transactions on*, 20(2):284 – 287, 1974.

[16] Jaekyun Moon and Weining Zeng. Equalization for maximum likelihood detectors. *Magnetics, IEEE Transactions on*, 31(2):1083 –1088, March 1995.

[17] Mathematica Inkscape Inductiveload, self-made. Normal distrbution probability density function @ONLINE. $https://en.wikipedia.org/wiki/Gaussian_function$, 2 April 2008.

[18] Altera. Low-Cost DSP Solutions in Cyclone II Devices.

[19] ST. STM32F4 High-Performance Discovery Board. *UM1472 User Manual, STM32F4DISCOVERY, Doc ID 022256 Rev 2*, January 2012.

[20] Altera Corporation. My First FPGA Design Tutorial. *Document*, July 2008.

[21] University of California Berkeley. How much information? *document*, 2003.

[22] M. Z. Ahmed. *Crosstalk-Resilient Coding For High Density Digital Recording*. PhD thesis, Department of Communication and Electrical Engineering, Faculty of Technology, University of Plymouth, UK, June 2003.

[23] S. Iwasaki. Lessons from research of perpendicular magnetic recording. *Magnetics, IEEE Transactions on*, 39(4):1868 – 1870, July 2003.

[24] H. Thapar and M. Marrow. Magnetic recording system - an overview. In *Globecom Workshops, 2007 IEEE*, pages 1 –3, November 2007.

[25] H Neal Bertram. *Theory of Magnetic Recording*. Cambridge University Press, Syndicate, University of Cambrige, 1994.

[26] G.D. Forney. Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference. *Magnetics, IEEE Transactions on*, 18(3):363–378, May 1972.

[27] S. Iwasaki. Perpendicular magnetic recording. *Magnetics, IEEE Transactions on*, 16(1):71 – 76, January 1980.

[28] IBM. IBM 3370 Direct Access Storage Device @ONLINE. $http://www-03.ibm.com/ibm/history/exhibits/storage/storage_3370.html$.

[29] M. W. Marcellin and H. J. Weber. Two-dimensional modulation codes. *Selected Areas in Communications, IEEE Journal on*, 10(1):254 –266, January 1992.

[30] R. E. Swanson and J. K. Wolf. A new class of two-dimensional RLL recording codes. *Magnetics, IEEE Transactions on*, 28(6):3407 –3416, November 1992.

[31] Hideki Sawaguchi, Yasutaka Nishida, Hisashi Takano, and Hajime Aoi. Performance analysis of modified PRML channels for perpendicular recording systems. *Journal of Magnetism and Magnetic Materials*, 235(13):265 – 272, 2001. Proceedings of the fifth Perpendicular Magnetic Recording Conference.

[32] Roger Wood, Yoshiaki Sonobe, Zhen Jin, and Bruce Wilson. Perpendicular recording: the promise and the problems. *Journal of Magnetism and Magnetic Materials*, 235(13):1 – 9, 2001. Proceedings of the fifth Perpendicular Magnetic Recording Conference.

[33] Seagate. Seagate Swings HAMR to Increase Disc Drive Densities by a Factor of 100 @ONLINE. $http://www.seagate.com/gb/en/about/newsroom/press-releases/seagate-swings-increase-disc-drive-densities-master-pr/$, 21 August 2002.

[34] Toshiba. Toshiba leads industry in bringing perpendicular data recording to hdd–sets new record for storage capacity with two new hdds @ONLINE. $http://www.toshiba.co.jp/about/press/2004_12/pr1401.htm$, 14 Dec 2004.

[35] Fujitus. Fujitsu Develops Optical Element for Thermal Assisted Magnetic Recording @ONLINE. $http://www.fujitsu.com/emea/news/pr/fel-en_20061128.html$, 28 Nov. 2006.

[36] Western Digital company HGST. Perpendicular Magnetic Recording Technology. *Whitepaper,Western Digital*, November 2007.

[37] Seagate. Seagate Powers Next Generation of Computing with Three New Hard Drives, Including World's First 1.5 Terabyte Desktop PC and Half Terabyte Notebook PC Hard Drives @ONLINE. $http:$

//www.seagate.com/gb/en/about/newsroom/press – releases/seagate – powers – next – gen – of – computing – with – 3 – new – drives – master – pr/, 10 July 2008.

[38] HTLounge.net. Western Digital, the first to ship an internal 3TB hard drive @ONLINE. *http://www.htlounge.net/art/13919/western – digital – the – first – to – ship – an – internal – 3tb – hard – drive.html*, 23 October 2010.

[39] ANANDTECH. Seagate Ships World's First 4TB External HDD @ONLINE. *http://www.anandtech.com/show/4738/seagate – ships – worlds – first – 4tb – external – hdd*, 7 September 2011.

[40] HARDWARE.INFO. Western Digital demonstrates 5mm thick hard disks @ONLINE. *http://uk.hardware.info/news/30068/idf – western – digital – demonstrates – 5mm – thick – hard – disks*, 15 September 2012.

[41] tom'sHARDWARE. TDK Finally Crams 2TB on One 3.5-inch HDD Platter @ONLINE. *http://www.tomshardware.com/news/HAMR – platters – heat – assisted – CREATEC – areal – density*,18126.*html*, 4 October 2012.

[42] M. Despotovic and B. Vasic. Hard disk drive recording and data detection. In *Telecommunications in Modern Satellite, Cable and Broadcasting Service, 2001. TELSIKS 2001. 5th International Conference on*, volume 2, pages 555 –561 vol.2, 2001.

[43] CITRIX. Introduction to storage technologies. *White Paper*.

[44] Gijoo Yang Joohyun Lee, Jaejin Lee. Channel model and GPRML detections for a single-layered perpendicular magnetic recording with a modified ring-head. *Journal of Magnetism and Magnetic Materials*, 272276(Part 3):2279–2281, May 2004.

[45] Yuanjing Shi. *Investigation of Island Geometry Variations in Bit Patterned Media Storage Systems*. PhD thesis, School of Computer Science, University of Manchester, UK, 2011.

[46] T. D. Leonhardt, R. J. M. van de Veerdonk, P. A. A. van der Heijden, T. W. Clinton, and T. M. Crawford. Comparison of perpendicular and longitudinal magnetic recording using a contact write/read tester. *Magnetics, IEEE Transactions on*, 37(4):1580 –1582, July 2001.

[47] D. Weller, Andreas Moser, Liesl Folks, Margaret E. Best, Wen Lee, M.F. Toney, M. Schwickert, J. U Thiele, and Mary F. Doerner. High Ku materials approach to 100 Gbits/in2. *Magnetics, IEEE Transactions on*, 36(1):10–15, 2000.

[48] E. Papagiannis, C. Tjhai, M. Z. Ahmed, M. Ambroze, and M. Tomlinson. Improved Iterative Decoding for Perpendicular Magnetic Recording. *ISCTA'05 Conference*, February 2005.

[49] S. Iwasaki and Y. Nakamura. An analysis for the magnetization mode for high density magnetic recording. *Magnetics, IEEE Transactions on*, 13(5):1272 – 1277, September 1977.

[50] R. Wood. The feasibility of magnetic recording at 1 terabit per square inch. *Magnetics, IEEE Transactions on*, 36(1):36 –42, January 2000.

[51] P.H. Siegel and J.K. Wolf. Modulation and coding for information storage. *Communications Magazine, IEEE*, 29(12):68 –86, December 1991.

[52] Yoshihiro Okamoto, Hisashi Osawa, Hidetoshi Saito, Hiroaki Muraoka, and Yoshihisa Nakamura. Performance of PRML systems in perpendicular magnetic recording channel with jitter-like noise. *Journal of Magnetism and Magnetic Materials*, 235(1-3):259 – 264, 2001.

[53] Western Digital. Perpendicular Magnetic Recording (PMR). *Technology Papers,Western Digital*, July 2006.

[54] Purav Shah. *Equalisation Techniques for Multi-Level Digital Magnetic Recording*. PhD thesis, School of Computing, Communications and Electronics, Faculty of Technology, University of Plymouth, UK, August 2008.

[55] The Register. Shingle Writing. $http$: $//www.theregister.co.uk/2010/07/12/adding_platters/page2.html$, July 2010.

[56] A. Amer, D. D. E. Long, E. L. Miller, J. F. Paris, and S. Thomas S.J. Design issues for a shingled write disk system. In *Mass Storage Systems and Technologies, 2010 IEEE 26th Symposium on*, pages 1 –12, May 2010.

[57] R. Wood. Perpendicular magnetic recording technology. *Witepaper, HGST, Westren Digital Company*, November 2007.

[58] R. Pitchumani, A. Hospodor, A. Amer, Yangwook Kang, E.L. Miller, and D.D.E. Long. Emulating a shingled write disk. *2012 IEEE 20th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pages 339 – 346, August 2012.

[59] R.D. Cideciyan, F. Dolivo, R. Hermann, W. Hirt, and W. Schott. A PRML system for digital magnetic recording. *Selected Areas in Communications, IEEE Journal on*, 10(1):38–56, January 1992.

[60] Sadik C. Esener, and Mark H. Kryder, and Panel Co-Chair, and William D. Doyle, and Marvin Keshner, and Masud Mansuripur, and D. A. Thompson. The Future of Data Storage Technologies. *International Technology Research Institute*, 1999.

[61] Kasiraj et al. System and method for writing hard disk drive depending on direction of head skew. *U.S. Patent 6967810*, 22 November 2005.

[62] E. F. Haratsch, G. Mathew, Jongseung Park, Ming Jin, K. J. Worrell, and Yuan Xing Lee. Intertrack interference cancellation for shingled magnetic recording. *Magnetics, IEEE Transactions on*, 47(10):3698 –3703, October 2011.

[63] Komkrit Chooruang. *Studies of Signal and Noise Properties of Perpendicular Recording Media*. PhD thesis, University of Exeter, UK, 2010.

[64] Hongwei Song and B.V.K. Vijaya Kumar. Low-density parity check codes for partial response channels. *Signal Processing Magazine, IEEE*, 21(1):56 – 66, January 2004.

[65] H. Kobayashi and D. T. Tang. Application of partial-response channel coding to magnetic recording systems. *IBM Journal of Research and Development*, 14(4):368–375, July 1970.

[66] Elizabeth Chesnutt. *Novel Turbo Equalization Methods for the Magnetic recording channel*. PhD thesis, Georgia Institute of Technology, 2005.

[67] M. Madden, M. Oberg, Zining Wu, and Runsheng He. Read channel for perpendicular magnetic recording. *Magnetics, IEEE Transactions on*, 40(1):241 – 246, January 2004.

[68] R. Wood, S. Ahlgrim, Kurt Hallamasek, and R. Stenerson. An experimental eight-inch disc drive with one-hundred megabytes per surface. *Magnetics, IEEE Transactions on*, 20(5):698–702, 1984.

[69] J.C. Coker, R.L. Galbraith, G.J. Kerwin, J.W. Rae, and P.A. Zipperovich. Integrating a partial response maximum likelihood data channel into the ibm 0681 disk drive. In *Signals, Systems and Computers, 1990 Conference Record Twenty-Fourth Asilomar Conference on*, volume 2, pages 674–, November.

[70] H.K. Thapar and A.M. Patel. A class of partial response systems for increasing storage density in magnetic recording. *Magnetics, IEEE Transactions on*, 23(5):3666–3668, September 1987.

[71] E. Kretzmer. Generalization of a techinque for binary data communication. *Communication Technology, IEEE Transactions on*, 14(1):67–68, 1966.

[72] Purav Shah, M. Z. Ahmed, and Y. Kurihara. New method for generalised PR target design for perpendicular magnetic recording. *The Eighth Perpendicular Magnetic Recording Conference, PMRC 2007*, October 15 - 17 2007.

[73] H. Ide. A modified PRML channel for perpendicular magnetic recording. *Magnetics, IEEE Transactions on*, 32(5):3965–3967, 1996.

[74] Jr. Forney, G.D. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973.

[75] A.E. Cetin, O.N. Gerek, and Y. Yardimci. Equiripple FIR filter design by the FFT algorithm. *Signal Processing Magazine, IEEE*, 14(2):60–64, March 1997.

[76] Edmund Lai. *Practical Digital Signal Processing for Engineers and Technicians*. Newnes, An imprint of Elsevier, Linacre House, Jordan Hill, Oxford OX2 8DP.

[77] Wang Yang, Jin Lin, and Liu Zhong. Utilization of chirp-z transform to improve the performance of target number detection of low resolution radar. In *Computational*

Engineering in Systems Applications, IMACS Multiconference on, volume 1, pages 403–407, 2006.

[78] Tapio Saramaki. *Chapter 4: Finite Impulse Response Filter*. Department of Electrical Engineering, Tampere University of Technology, Tampere, Finland.

[79] Sklar B. *Digital Communications: Fundamentals and Applications, Second Edition*. Prentice Hall, New Jersey, 2001.

[80] J. G. Proakis. *Digital communications, Third Edition*. McGraw-Hill, third edition, 1995.

[81] Marshall Brain. What is White Noise @ONLINE. http://www.howstuffworks.com/question47.htm.

[82] P.A. Voois and J.M. Cioffi. A decision feedback equalizer for multiple-head digital magnetic recording. In *Communications, 1991. ICC '91, Conference Record. IEEE International Conference on*, pages 815–819 vol.2, 1991.

[83] W. L. Abbott, J. M. Cioffi, and H. K. Thapar. Offtrack interference and equalization in magnetic recording. *Magnetics, IEEE Transactions on*, 24(6):2964 –2966, November 1988.

[84] L. C. Barbosa. Simultaneous detection of readback signals from interfering magnetic recording tracks using array heads. In *Magnetics Conference, 1990. Digests of INTERMAG1990. International*, pages EC–05, April 1990.

[85] M. P. Vea and J. M. F. Moura. Magnetic recording channel model with intertrack interference. *Magnetics, IEEE Transactions on*, 27(6):4834–4836, November 1991.

[86] J. Lee and V. K. Madisetti. Multitrack RLL codes for the storage channel with immunity to intertrack interference. In *Global Telecommunications Conference, 1994. GLOBECOM '94. Communications: The Global Bridge., IEEE*, volume 3, pages 1477 –1481 vol.3, November-2 December 1994.

[87] P. J. Davey, T. Donnelly, and D. J. Mapps. Two-dimensional coding for a multiple-track, maximum-likelihood digital magnetic storage system. *IEEE Transactions on Magnetics*, 30(6):4212 –14, November 1994.

[88] E. Soljanin and C. N. Georghiades. Multihead detection for multitrack recording channels. *Information Theory, IEEE Transactions on*, 44(7):2988 –2997, November 1998.

[89] Bong Gyun Roh, Sang Uk, Jaekyun Moon, and Ying Chen. Single-head/single-track detection in interfering tracks. *Magnetics, IEEE Transactions on*, 38(4):1830 –1838, July 2002.

[90] W. Tan and J. R. Cruz. Signal processing for perpendicular recording channels with intertrack interference. *Magnetics, IEEE Transactions on*, 41(2):730 – 735, Febuary 2005.

[91] N. Singla, J. A. O'Sullivan, C. T. Miller, and R. S. Indeck. Decoding for magnetic recording media with overlapping tracks. *Magnetics, IEEE Transactions on*, 41(10):2968 – 2970, October 2005.

[92] P.M. Jermey, H.A. Shute, M.Z. Ahmed, and D.T. Wilton. Exploitation of intertrack interference in a shingled replay system. *Journal of Magnetism and Magnetic Materials*, 321(19):2992 – 2998, 2009.

[93] S. Greaves, Y. Kanai, and H. Muraoka. Shingled recording for 2-3 Tbit/in^2. *Magnetics, IEEE Transactions on*, 45(10):3823 –3829, October 2009.

[94] A.R. Krishnan, R. Radhakrishnan, B. Vasic, A. Kavcic, W. Ryan, and F. Erden. 2-D magnetic recording: Read channel modeling and detection. *Magnetics, IEEE Transactions on*, 45(10):3830 –3836, October 2009.

[95] A.R. Krishnan, R. Radhakrishnan, and B. Vasic. Read channel modeling for detection in two-dimensional magnetic recording systems. *Magnetics, IEEE Transactions on*, 45(10):3679 –3682, October 2009.

[96] Kheong Sann Chan, J.J. Miles, Euiseok Hwang, B. Vijayakumar, Jian-Gang Zhu, Wen-Chin Lin, and R. Negi. TDMR platform simulations and experiments. *Magnetics, IEEE Transactions on*, 45(10):3837–3843, 2009.

[97] S. Tan, Weiya Xi, Zhi Yong Ching, Chao Jin, and Chun Teck Lim. Simulation for a shingled magnetic recording disk. In *APMRC, 2012 Digest*, pages 1–7, 2012.

[98] K. Venkataraman, N. Xie, G. Dong, and T. Zhang. Reducing read latency of shingled magnetic recording with severe inter-track interference using transparent lossless data compression, 2013.

[99] Altera. FPGAs Provide Reconfigurable DSP Solutions. *White Paper FPGA/DSP*, August 2002.

[100] Altera. DSP Builder Handbook Volume 2: DSP Builder Standard Blockset. *Document last updated for Altera Complete Design Suite version: 11.1*, November 2011.

[101] Altera. DSP Builder Handbook Volume 3: DSP Builder Advanced Blockset. *Document last updated for Altera Complete Design Suite version: 11.1*, November 2011.

[102] Altera. Quartus II Handbook Version 12.0 Volume 3: Verification. *Document last updated for Altera Complete Design Suite version: 12.0*, June 2012.

[103] Altera Corporation-University Program. Using ModelSim to Simulate Logic Circuits for Altera FPGA Devices. *Document*, January 2011.

[104] Altera. Introduction to Quartus II Software. *Document, Quartus II Design Software*, 2011.

[105] Mike Pridgen, TA and Dr. Eric M. Schwartz. Tutorial for Quartus SignalTap II Logic Analyzer. *Tutorial, University of Florida, Dept. of Elec. and Comp. Engr., EEL4712*, pages 1–7, February, 2008.

[106] Altera Corporation, AN-376-1.0. Cyclone II Filtering Lab. *Document, Application Note 376, ver. 1.0*, May, 2005.

[107] C.R. Lageweg. Designing an Automatic Schematic Generator for a Netlist Description. *Technical Report, Laboratory of Computer Architecture and Digital Techniques (CARDIT), Department of Electrical Engineering, Faculty of Information Technology and Systems, Delft University of Technology, The Netherlands*, pages 1–83, September 8, 1998.

[108] Weng Fook Lee. *VHDL Coding and Logic Synthesis with SYNOPSYS*. Academic Press, Harcourt Place, 32 Jamestown Road, London, NWl 7BY, UK, New Jersey, 2000.

[109] Peter J. Ashenden. The VHDL Cookbook. *First Edition, Dept. Computer Science, University of Adelaide, South Australia*, pages 1–111, July, 1990.

[110] MathWorks. Solver Pane R2013a @ONLINE. http://www.mathworks.co.uk/help/simulink/gui/solver-pane.html.

[111] University of Maryland ENEE 359a: Digital VLSI Circuits, Spring 2008. ModelSim Tutorial and Verilog Basics. *Project*, 2008.

[112] Altera Corporation-University Program. Quartus II Introduction Using Schematic Designs. *Document For Quartus II 11.1*, August, 2011.

[113] Altera Corporation-University Program. Quartus II Introduction Using VHDL Designs. *Document For Quartus II 11.1*, August, 2011.

[114] Altera Corporation. DE1 Development and Education Board. *User Manual, version 1.1*, 2006.

[115] Daniele Riccardi and Paolo Novelliniz. An Attribute-Programmable PRBS Generator and Checker. *XILINX, XAPP884(v1.0)*, January, 2011.

[116] P. Korohoda and A. Dqbrowski. Generalized convolution concept extension to multidimensional signals. In *Multidimensional Systems, 2005. NDS 2005. The Fourth International Workshop on*, pages 187 – 192, July 2005.

[117] Altera. Section I. DSP Builder Standard Blockset User Guide. *Document*, 1.1, April 2011.

[118] Moon Todd. *Error Correction Coding, Mathematical Methods and Algorithms*. John Wiley and Sons, Canada, 2005.

[119] Altera. Cyclone V FPGA family overview @ONLINE. *http* : *//www.altera.co.uk/devices/fpga/cyclone − v − fpgas/overview/cyv − overview.html*.

[120] Roger Wood. Shingled magnetic recording and two-dimensional magnetic recording. *Power Point Presenation*, 19 October 2010.

[121] Garth Gibson. Principles of operation for shingled disk devices. *Newsletter on Parallel Data Laboratory Activities and Events, Carnegie Mellon University*, Spring 2011.