

AN EFFICIENT DOCSIS UPSTREAM EQUALIZER

A Thesis Submitted
to the College of Graduate Studies and Research
in Partial Fulfillment of the Requirements
for the Degree of Master of Science
in the Department of Electrical and Computer Engineering
University of Saskatchewan

by
Md. Mushtafizur Rahman Choudhury

Saskatoon, Saskatchewan, Canada

© Copyright Md. Mushtafizur Rahman Choudhury, March, 2014. All rights reserved.

Permission to Use

In presenting this thesis in partial fulfillment of the requirements for a Postgraduate degree from the University of Saskatchewan, it is agreed that the Libraries of this University may make it freely available for inspection. Permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professors who supervised this thesis work or, in their absence, by the Head of the Department of Electrical and Computer Engineering or the Dean of the College of Graduate Studies and Research at the University of Saskatchewan. Any copying, publication, or use of this thesis, or parts thereof, for financial gain without the written permission of the author is strictly prohibited. Proper recognition shall be given to the author and to the University of Saskatchewan in any scholarly use which may be made of any material in this thesis.

Request for permission to copy or to make any other use of material in this thesis in whole or in part should be addressed to:

Head of the Department of Electrical and Computer Engineering
57 Campus Drive
University of Saskatchewan
Saskatoon, Saskatchewan, Canada
S7N 5A9

Acknowledgments

I would like to express my heartiest gratitude to my supervisor, Professor J. Eric Salt for his guidance and teaching throughout my pursuit of this Industrial Oriented Masters degree. His guidance and encouragement from the very beginning of this IOM program, always kept me on the right track and increased my confidence in doing research independently. My co-supervisor, Professor Ha H. Nguyen has also played an important role throughout the IOM program. I am very happy to have had the opportunity to work with such a wonderful supervisors.

I would also like to thank NSERC and TRILabs for financially supporting in the pursuit of completing the IOM program successfully. Special thanks to Mr. Andrew Kostiuk, head of interactive application services in TRTech, Saskatoon for his guidance, questions and comments in the continuation of research studies for two successive summer terms at TRILabs.

I would like to thank all my colleagues in the IOM program. This includes Rory Gowen, Suresh Kalle and Andy Fontaine, with whom I have enjoyed many constructive discussions and collaborations. I am also grateful to the Department of Electrical and Computer Engineering at University of Saskatchewan for providing high quality education and research environment.

A special thanks to Professor Aryan Saadat Mehr and Dr. Brian Berscheid for being in my M.Sc. thesis advisory committee and Professor Alexandre V. Koustov for being an external examiner. Their constructive criticism, comments and suggestions will be an asset to achieve my future goal.

Finally, I would like to express my gratitude to my parents, Md. Shazidur Rahman Choudhury and Kamrunnesa Choudhury, and my wife Iffath Fathima Chowdhury who have provided tremendous support, encouragement and inspiration over the years.

Abstract

The advancement in the CATV industry has been remarkable. In the beginning, CATV provided a few television channels. Now it provides a variety of advanced services such as video on demand (VOD), Internet access, Pay-Per-View on demand and interactive TV. These advances have increased the popularity of CATV manyfold. Current improvements focus on interactive services with high quality. These interactive services require more upstream (transmission from customer premises to cable operator premises) channel bandwidth.

The flow of data through the CATV network in both the upstream and downstream directions is governed by a standard referred to as the Data Over Cable Service Interface Specification (DOCSIS) standard. The latest version is DOCSIS 3.1, which was released in January 2014. The previous version, DOCSIS 3.0, was released in 2006.

One component of the upstream communication link is the QAM demodulator. An important component in the QAM demodulator is the equalizer, whose purpose is to remove distortion caused by the imperfect upstream channel as well as the residual timing offset and frequency offset. Most of the timing and frequency offset are corrected by timing and frequency recovery circuits; what remains is referred to as offset.

A DOCSIS receiver, and hence the equalizer within, can be implemented with ASIC or FPGA technology. Implementing an equalizer in an ASIC has a large non-recurring engineering cost, but relatively small per chip production cost. Implementing equalizer in an FPGA has very low non-recurring cost, but a relatively high per chip cost. If the choice technology was based on cost, one would think it would depend only on the volume, but in practice that is not the case. The dominant factor when it comes to profit, is the time-to-market, which makes FPGA technology the only choice.

The goal of this thesis is to design a cost optimized equalizer for DOCSIS upstream demodulator and implement in an FPGA. With this in mind, an important objective is to establish a relationship between the equalizer's critical parameters and its performance. The parameter-performance relationship that has been established in this study revealed that equalizer step size and length parameters should be $1/64$ and approximately 20 to yield a

near optimum equalizer when considering the MER-convergence time trade-off.

In the pursuit of the objective another relationship was established that is useful in determining the accuracy of the timing recovery circuit. That relationship establishes the sensitivity both of the MER and convergence time to timing offset.

The equalizer algorithm was implemented in a cost effective manner using DSP Builder. The effort to minimize cost was focused on minimizing the number of multipliers. It is shown that the equalizer can be constructed with 8 multipliers when the proposed time sharing algorithm is implemented.

Table of Contents

Permission to Use	i
Acknowledgments	ii
Abstract	iii
Table of Contents	v
List of Tables	viii
List of Figures	ix
List of Abbreviations	xiii
1 Introduction	1
1.1 Evolution of Cable Television	1
1.2 The Architecture of Cable Networks	7
1.3 DOCSIS Upstream Demodulator	9
1.4 DOCSIS Upstream Channel	11
1.5 Statement of Research Problems	13
1.6 Organization of the Thesis	14
2 CATV Upstream Channel	15
2.1 Overview	15
2.2 Physical Layer of DOCSIS Upstream Channel	15
2.2.1 Basic Modulation and Demodulation Technique	16
2.2.2 Channel Impairments	18

2.3	MAC layer of DOCSIS Upstream Channel	21
3	Theory of Equalization	24
3.1	Least Mean Square Error Equalizer	24
3.1.1	System Model	24
3.1.2	Mean Squared Error (MSE) Criterion	29
4	DOCSIS Equalizer	47
4.1	Coefficient Update Algorithm	48
4.2	LMS Algorithm	48
4.3	Structure of the DOCSIS Equalizer in Upstream Demodulator	55
4.3.1	The operating Mode of the Equalizer	57
4.4	Performance Criterion	58
5	Implementation of the DOCSIS Equalizer	61
5.1	Implementation Roadmap	62
5.2	Implementation Issues to Find a Cost Effective Equalizer	63
5.3	Effect of Timing Error on Equalizer Performance	72
5.3.1	Establishing the Timing Reference	73
5.4	Implementation Structure of the Equalizer	75
6	Simulation Results	80
6.1	MATLAB Simulation Results	80
6.2	DSP Builder Simulation Results	89
7	FPGA Based Results	93

7.1	Equipment Description	93
7.2	Performance of FPGA Implementation	97
8	Summary and Conclusions	101
8.1	Contribution towards establishing parameters-performance relationship for the equalizer	101
8.2	Contributions towards critical issues that affects equalizer performance . . .	102
8.3	Contributions towards cost optimization and implementation	104
	References	105

List of Tables

1.1	Versions of DOCSIS standard with the released year and features	9
2.1	DOCSIS specified echoes with their parameters	20
5.1	Performance measures of a 20 tap equalizer with different step sizes for a threshold of 125% of mean squared error in steady state.	70

List of Figures

1.1	Bidirectional frequency band in a DOCSIS network	8
1.2	High-level block diagram of DOCSIS network	10
1.3	High-level block diagram for a DOCSIS 3.0 Upstream Demodulator	10
1.4	Process of micro-reflection generation in a cable plant	12
2.1	High-level block diagram of DOCSIS upstream communication system	15
2.2	Basic block diagram of QAM modulator	16
2.3	Constellation diagram for 4-QAM data	17
2.4	Basic block diagram of QAM demodulator	18
2.5	Impulse response of channel model to illustrate the effect of echo	20
2.6	Constellation diagram showing the effect of echo	21
2.7	Process of ranging mode operation	22
3.1	The Digital Communication System	25
3.2	The Modified Digital Communication System	27
3.3	Generating $h[l]$	28
3.4	Somewhat Refined System Model	28
3.5	Symbol rate System Model	29
3.6	Real part of four sample functions in stochastic process $\mathbf{I}[k]$	40
3.7	Imaginary part of four sample functions in stochastic process $\mathbf{I}[k]$	41
3.8	Real part of four sample functions in stochastic process $\mathbf{I}_{special}[k]$	42

3.9	Real part of equalizer output for input $\mathbf{I}_{special}[k]$	43
3.10	Vectors used in MER	44
4.1	Block diagram of 24 tap symbol-spaced complex adaptive equalizer	56
4.2	A typical DOCSIS packet burst	57
4.3	Block diagram for MER measurement	59
5.1	A high-level structure of implementation roadmap.	63
5.2	MER comparison in a DOCSIS single echo channel with echo strength 0.3162 of main path and a step size of $\mu = \frac{1}{64}$ in a 20 and 24 tap equalizers.	65
5.3	Equalizer error in a 20 tap equalizer with step size, $\mu = \frac{1}{64}$	66
5.4	Equalizer error in a 24 tap equalizer with step size, $\mu = \frac{1}{64}$	67
5.5	Equalizer error with step size, $\mu = \frac{1}{32}$	68
5.6	Equalizer error with step size, $\mu = \frac{1}{64}$	69
5.7	Equalizer error with step size, $\mu = \frac{1}{128}$	69
5.8	MER in a DOCSIS single echo channel with echo strength 0.3162 of main path and a step size of $\mu = \frac{1}{64}$ in a 20 tap equalizer.	71
5.9	Convergence time in a DOCSIS single echo channel with echo strength 0.3162 of main path and a step size of $\mu = \frac{1}{64}$ in a 20 tap equalizer.	72
5.10	Magnitude of Matched Filter Output in Eye Diagram Format.	73
5.11	Minimum of Magnitudes of Matched Filter Output.	74
5.12	Canonical structure of equalizer.	78
6.1	Normalized theoretical MER for a SNR of 30 dB and an infinite length equalizer	81

6.2	Ensemble average of normalized theoretical MERs (infinite length equalizer, SNR = 30 dB)	82
6.3	Normalized practical MER (20 tap equalizer, SNR = 30 dB)	83
6.4	Ensemble average of normalized practical MERs (20 tap equalizer, SNR = 30 dB)	84
6.5	Relative convergence time ($\mu\sigma_I = \sqrt{2}/64$) (20 tap equalizer)	85
6.6	Ensemble average of normalized convergence times ($\mu\sigma_I = \sqrt{2}/64$) (20 tap equalizer)	86
6.7	Histogram of theoretical MER (infinite length equalizer)	87
6.8	Histogram of practical MER (20 tap equalizer)	87
6.9	Histogram of convergence time ($\mu\sigma_I = \sqrt{2}/64$) (20 tap equalizer)	88
6.10	Constellation plot of the noise corrupted QPSK data at the input of the equalizer	88
6.11	Constellation plot of the noise corrupted QPSK data at the output of the equalizer	89
6.12	Equalizer coefficients at steady state	90
6.13	Equalizer error in DSP Builder	91
6.14	Constellation plot of the output of the equalizer	92
7.1	Hardware and measurement equipment	94
7.2	High level block diagram illustrating connections between the DE4 Board and the daughter card	96
7.3	Comparison between the real part of equalizer output and the respective real part of the training data in the oscilloscope	98

7.4	Comparison between the imaginary part of the equalizer output and the respective imaginary part of the training data in the oscilloscope	99
7.5	Comparison between the real part of equalizer output and the corresponding training data in Signal Tap	100
7.6	Equalizer error displayed in the oscilloscope	100

List of Abbreviations

AGC	Automatic Gain Control
AWGN	Additive White Gaussian Noise
ASIC	Application Specific Integrated Circuit
ADC	Analog to Digital Converter
CATV	Community Antenna Television
CMTS	Cable Modem Termination System
CM	Cable Modem
DOCSIS	Data Over Cable Service Interface Specification
DSP	Digital Signal Processing
DAC	Digital to Analog Converter
FCC	Federal Communication Commissions
FPGA	Field Programmable Gated Array
HBO	Home Box Office
HFC	Hybrid Fibre Coax
ISI	Inter Symbol Interference
LUT	Look Up Table
LO	Local Oscillator
MATV	Master Antenna Television
MAC	Media Access Control
MSE	Mean Square Error
MER	Modulation Error Ratio
QAM	Quadrature Amplitude Modulation
SRRC	Square Root Raised Cosine Filter
TDMA	Time Division Multiple Axis

1. Introduction

1.1 Evolution of Cable Television

In a communication system, the link between the transmitter and receiver can be wireless, optical fibre, a twisted pair of wires or a co-axial cable. Among these, co-axial cable networks are prominently used in communication to homes. A variety of services such as high speed internet, analog television, high definition digital television, digital audio and IP telephony are provided through co-axial cable networks. Modern cable networks are able to provide two way communication between customer and a central point, which is called the headend. Each home is linked to the headend by a single cable that is divided into many channels using frequency division. On top of that each channel is broken into time slots. In the end for all intents and purposes many homes can communicate with the headend simultaneously.

Although in the early year cable television complemented the regulated television broadcasting, later it emerged as a major force in an increasingly demanding telecommunication and entertainment sector. Cable television services then rapidly started to reach the subscribers, making the cable industry an attractive investment. The local cable operator took the lead to develop this promising sector a link to the information superhighway. The development of cable industry was not easy at all. The evolution of cable industry has been hindered by restrictions imposed by regulation authorities.

In the 1940s and 1950s cable television was developed in USA [1]. The modern day cable plant started as a Community Access Television or Community Antenna Television network [1], [2]. The community antenna television network was called CATV. In the beginning cable plant was used to connect a common antenna to TV receivers that were not within line of sight of the broadcast antenna. In its infancy, which was in the 1940's, CATV systems were

deployed in rural areas where over-the-air reception was limited by mountainous terrain or by the distance from the broadcast transmitters. The reception of radio wave signal also was weak in the cities where high rise buildings blocked radio waves. Cable television system operators located antennas on mountains tops and other high points where reception of broadcasting signals was good. They then distributed the signals by a co-axial cable to the subscribers. The networks consists of a community receive antennas and co-axial cable with uni-direction power flow. Later the cable operators piped in TV channels from other cities and used the CATV network to provide their customers with more television channels. Having many channels made CATV systems popular in urban areas and the number of CATV subscribers increased rapidly.

In the early 1950's cable television was still fairly new and expensive. The restrictions imposed by Federal Communication Commissions (FCC) on new TV stations made it difficult for the cable operators to move forward. After a short time the FCC released a nationwide television broadcasting plan which prompted many new stations. With the rapid increase in TV stations cause an increase in the sale of TV sets and antennas. Manufacturers made many models of TVs available, but supply was ahead of demand. Those who had own TV sets also owned an antenna. It was common to see arrays of TV antennas on the roofs of a high rise apartment. Later, Milton Jerrold Shapp developed a system where one master antenna (MATV) [2] could be used for all televisions in a building. His idea was to connect an active signal booster to broadband antenna and use it to drive the co-axial cable with the multiple channel broadband signals. At about the same time Robert (Bob) Tarlton thought that this technique would also work for an entire city. A structure like today's cable television was born when he wired an entire city using co-axial cable and commercially manufactured signal boosters.

By 1952, cable systems were operated in 70 communities and served 14,000 subscribers nationwide in USA [2]. In the late 1950s, cable operators were able to take advantage of microwave and other technologies to pick up broadcast signals from distant broadcasting stations. The ability to access signals from broadcasting stations started a new era for the cable television industry. Cable system moved from transmitting local broadcast signals

to providing many programming choices. Once cable operators increased to the number of channels, CATV became much more attractive and the cable system became increasingly popular. The popularity provided an economic opportunity for the cable operator who quickly moved into the large cities. By 1962, almost 800 cable systems were operating and served approximately 850,000 subscribers in USA.

Many local broadcasters felt cable system was in an unfair competition. They asked the government to stop cable operator from distributing TV channels originating from distant broadcasters. The FCC responded by placing restrictions on cable systems disallowing them to import signals from distant broadcasting stations. As a result the growth of the cable industry slowed for next few years.

In the early 1970s, the FCC continued its restrictive policies by enacting regulations that limited the ability of cable operators to offer movies, sporting events, and syndicated programming. The freeze on cable's development lasted until 1972, then FCC started relaxing restrictions on importing distant signals. The continuous concerted effort by the cable industry at federal and local levels resulted in gradual lessening of restrictions on cable industry throughout the decade. These changes along with the development of satellite communications technology and pay TV programming were believed by many to be the main driving force behind the success of the cable industry during the last two decades. The offer of more services led to a substantial increase in subscribers. In 1972 the first pay TV network was launched with Home Box Office (HBO) service. This represented the first successful pay cable service in history. Though very few subscribed to HBO in the beginning, its subscriber base steadily grew and it became one of the most viewed TV service throughout the world. The success of HBO inspired the cable operators to offer many other programming services. With the help of satellite technology HBO was the first programming service conveyed to the subscribers. The second service that used satellite technology was a local TV station that broadcast sports and classic movies. The distribution of station signals by satellite made it available to cable operators to provide services throughout North America. Today almost all cable broadcasting is distributed by satellite. By the end of the 1970's cables growth had resumed and nearly 16 million households were cable subscribers.

The rules and regulations were becoming favourable for the cable industry to explore other opportunities. The 1980 cable act established a friendly regulatory framework for the cable industry. This cable act inspired many to invest in cable plant and television programming. This friendly environment had a strong impact on the rapid growth of cable services. One of the largest private construction projects started after World War II in the cable industry from 1984 through 1992. In this period industry spent more than \$15 billion on the wiring throughout USA, and billions more on program development. The relaxation of FCC rules with the combination of satellite technology paved the way for the cable industry to become a major force in providing high quality video entertainment and information services to subscribers. By the end of the decade, nearly 53 million households subscribed to cable, and cable program networks had increased from 28 in 1980 to 79 by 1989. At the same time cable operators started increasing the price for the services they provide to subscribers.

In the late 80's and early 90's the prices charged for cable services grew to a point where it concerned the policy makers. In 1992, Congress responded by opening the door for other wireless cable and direct to home services. This led to competition from direct satellite broadcast (DBS) service providers and telephone companies. These new service providers captured much of the growth market stalling the expansion of CATV networks.

Despite of the tight rules of Congress, many satellite networks formed an alliance with cable networks with the view of fulfilling an alternate plan to provide services to a specific sector of the market. Aided by these alliances, by the end of 1995, the number of cable networks had grown to 139 nationwide. This number steadily increased and by the end of 1998, the number of cable networks had elevated to 171. As a result, subscribers had the opportunity to choose from a wide selection of quality services. About 57% of all subscribers were receiving at least 54 channels, which was only 47 channels two years earlier. At the end of the decade, there were more than 55 million cable subscribers.

Starting in the mid 90's the cable operating companies started a major upgrade of their distribution networks. They invested \$65 billion from 1996 through 2002 to build higher capacity hybrid networks consisting of fiber optic and co-axial cable. These broadband

hybrid networks opened the gateway for the cable company to provide high speed internet access, multichannel video, two way voice and high definition and advanced digital video services to the subscribers. Later in the 2000's decade other competitive services such as telephone and digital cable services were added.

In 1996 the relaxation of telecommunication act changed the regulation and policies in favour of the cable operator services. The 1996 act was not only a blessing for the cable operators but also it attracted many investors and many new projects were launched throughout North America. This friendly regulatory environment helped the cable industries accelerate deployment of broadband services in urban, suburban and rural areas. As a result subscribers from those areas had more choices in information, communications and entertainment services.

Historically, the cable television industry was established mainly to provide television programming services. Initially, only one way communication was possible, but the demand for two way services and improved quality forced the cable operators to incorporate two way communication. Two way communication was introduced early in the 1970s with the intention to provide medical alerts, meter readings, etc. However, with progress two way communication services increased. The important aspects of two way communication is to provide data transmission and internet services to the subscribers. Cable quickly became the technology of choice for such services, dominating rival technologies, such as digital subscriber line (DSL) service, offered by telephone companies by about 2:1. Upgraded cable plant that allowed two way communication opened tremendous business opportunity and took the cable industry to an unprecedented level.

The new millennium started with a rigorous plan to convey advanced services to the subscribers over the cable broadband networks in a minimum possible time. Cable companies started testing video services that could change the way people watch television. These services include video on demand (VOD), Internet access, Pay-Per-View on demand and interactive TV. The addition of these advanced services challenged the equipment on customer premises and forced the cable operators to upgrade it. The cost was substantial, so the cable companies moved very carefully. The solution was a sophisticated digital set top box which

was initially very expensive. In the mid 1990's the cost of the digital set top boxes dropped and it started to become widely used. By 2000, digital set-top boxes could be remotely activated, programmed and monitored which allowed the cable operators to incorporate sophisticated encryption technology. Furthermore, this upgraded digital set top boxes paved the way for Pay-per-view and Video-on-Demand services to name a few. In addition to that, the inclusion of digital video compression technology (MPEG 2) greatly increased the capacity of system by packing 10 digitally encoded TV channels into one standard analog channel. The cable industry then had the capacity to provide many speciality channels like HBO, Discovery, ESPN, the shopping channels etc. These advancements in cable industry steadily increased the number of cable subscribers. By the end of 2000, about 40 percent of North Americans had access to high speed internet services. At the same time, growth in telephone services over cable was also notable in the limited market areas. By 2002, more than 2 million subscribers were using cable for their phone service provided by about 280 cable networks. The number of cable networks has steadily increased ever since.

The main competitor of CATV network is the telephone plant, which is primarily a wire line network. Both networks provide service via wire, but they are very different in their technology. Telephone system started two way communication long before the CATV system. The telephone service providers are connected to each other where as cable operators are isolated. One of the important differences is that in a telephone network each subscriber is connected to the central point via a dedicated line. On the other hand in a CATV networks subscribers are connected to headend by a shared co-axial plant network. This is possible in CATV system by using frequency division of its larger bandwidth to provide many channels.

Telephone services over the cable network evolved through two technologies. At first the service required a dedicated bandwidth for an entire conversation. It also required dedicated signal processing equipment that was designed to provide high reliability. This technology was used until about 2003. At time the technology changed to voice-over-IP (VOIP). This is packet based time sharing system that piggybacks the terminal equipment used for data services. While the quality of service for the VOIP technology is not as quite as good, it could be provided with much less cost.

The number of cable operators declined slightly in 2001 and 2002 due to unforeseen upgrading costs and tough competition. However, the cable industry overcame the challenges and in 2005 according to a FCC report, the use of cable modems had climbed 30%.

The CATV is still a very young industry and still in the growth phase. According to the cable television advertising bureau's report, about 92 million households are currently subscribe to cable. Cable systems are operating in every state of the United States and in many other countries, such as Canada and Australia, and throughout Europe and much of East Asia.

1.2 The Architecture of Cable Networks

Modern day cable networks must support two-way communication. In simpler words, there must be two communication links between the headend and end users. The link that supports communication from the headend to the end users is called the downstream communication link [3]. The link that supports communication in the other direction, which is from the end users to the headend, is called the upstream communication link [3]. For two-way communication, the frequency spectrum on the cable is divided. Upstream communication links are assigned the lower frequency region (typically 5-85 MHz) [3], while downstream communication links are assigned the upper frequency region (typically 100 MHz-1000 MHz) [3], as depicted in Figure. 1.1

The bidirectional frequency spectrum are standardized by Data Over Cable Service Interface Specification (DOCSIS) [3]. DOCSIS is an international telecommunication standard that came into existence as a result of lack of interoperability among various cable networks. DOCSIS was developed by a non-profit research and development consortium of cable operators and equipment manufacturers called CableLabs with the aim to promote interoperability. The first version of the DOCSIS standard is DOCSIS 1.0, originally developed in March 1997 [4]. In April 1999, it was revised to become DOCSIS 1.1. The primary upgrade was the addition of Quality of Service (QoS) capabilities. Since the beginning of DOCSIS, continuous demand for increased data throughput forced CableLabs to release further upgraded versions of the DOCSIS standard. All versions of the DOCSIS standard

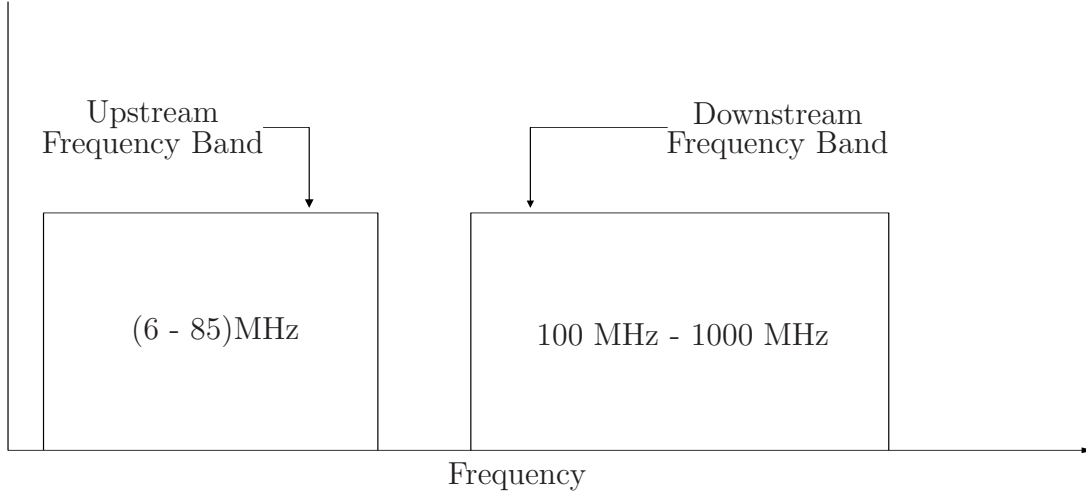


Figure 1.1 Bidirectional frequency band in a DOCSIS network

are described chronologically in Table. 1.1. At the time of writing, the latest version of the DOCSIS standard was DOCSIS 3.0, which was released in 2006. The next version, which is DOCSIS 3.1, is expected to be released in 2014.

A simplified block diagram of a DOCSIS network [3] is shown in Figure. 1.2. The DOCSIS network consists of a Cable Modem Termination System (CMTS), a Cable Network and many Cable Modem (CMs). The CMTS resides in the headend whereas the CM resides in the subscriber premises. The cable networks can be all co-axial cable or a hybrid of optical fiber and co-axial cable. The heart of the DOCSIS network is the CMTS, which communicates with the cable modem through the cable network in a bidirectional way. The upstream communication system changed considerably with each new version of the DOCSIS standard. A few of these changes are increased symbol rate, the addition of an error control coding scheme, modulation scheme and the packet format. The critical components of both the CMTS and CM are a modulator and a demodulator. The link between CMTS modulator and CM demodulator is called the downstream channel and the link between CMTS demodulator and CM modulator is called the upstream channel.

Table 1.1 Versions of DOCSIS standard with the released year and features

DOCSIS Version	Year Released	Features
DOCSIS 1.0	1997	Basic broadband Internet connectivity
DOCSIS 1.1	1999	Quality of Service, Dynamic Services, Concatenation, Fragmentation, IP Multicast, Fault Management, Secure Software etc. and includes all the features of DOCSIS 1.0
DOCSIS 2.0	2001	Significantly enhanced upstream capacity, 6.4 MHz maximum upstream channel width, 27 Mbps maximum upstream channel capacity, Increased robustness to upstream noise and channel impairments etc. and includes all the features of DOCSIS 1.1.
DOCSIS 3.0	2006	Channel bonding to increase possible upstream and downstream data rates by a factor of 4, Support for IPv6, Enhanced security features, Support for IPTV etc. and includes all the features of DOCSIS 2.0

1.3 DOCSIS Upstream Demodulator

The CM modulator, Quadrature Amplitude Modulates (QAM) [5] a carrier in preparation for transmission through the cable plant. This signal is received at the CMTS where it is demodulated. The structural block diagram of DOCSIS upstream demodulator is shown in Figure 1.3. The received signal is passed through an analog to digital converter, which is denoted by ADC. The output of the ADC is down converted to a complex baseband signal and then down sampled. The output of the down sampler is passed through a Square Root Raised Cosine (SRRC) filter [6], which is matched to the pulse shaping filter [6] in the transmitter. The matched filter output is passed through a timing synchronizer [7], [8]. The purpose of the timing synchronizer is to recover timing so that the sample points in the

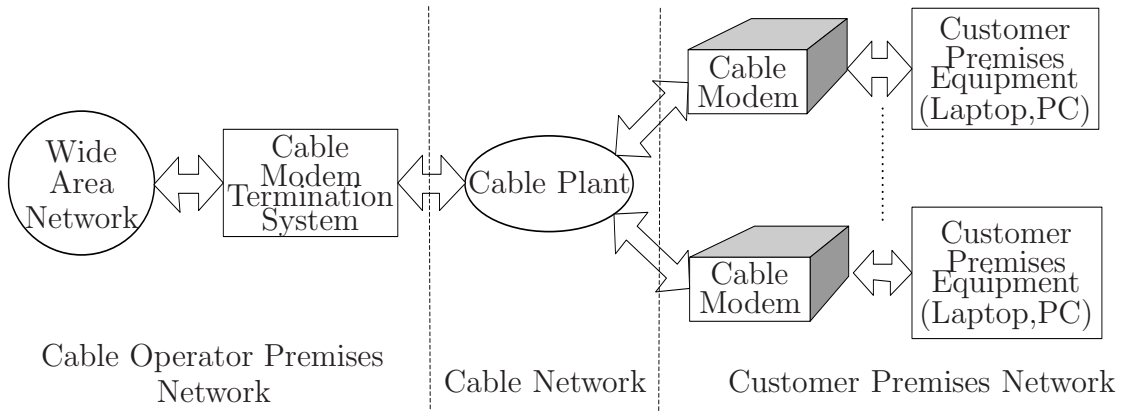


Figure 1.2 High-level block diagram of DOCSIS network

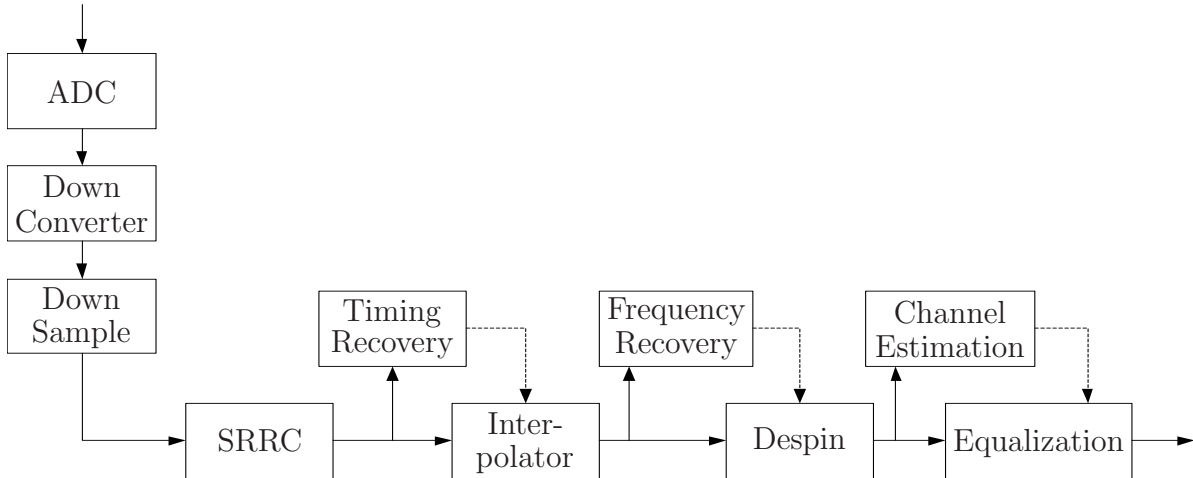


Figure 1.3 High-level block diagram for a DOCSIS 3.0 Upstream Demodulator

receiver coincide with the sample points in the transmitter. Timing synchronization is done in two stages [7], as a timing recovery circuit [8] and an interpolation circuit [7]. Timing recovery circuit measures the time offset in sample time between the sample time in the transmitter and the sample time in the receiver. The interpolation circuit calculates the values of the samples with the sample times delayed by the timing offset.

Frequency synchronization [9] is also necessary. It is done in two stages as well, as a frequency recovery circuit and the despinner circuit. The frequency recovery circuit measures the frequency offset due to an imprecise local oscillator in the down-converter. The despinner

corrects for error in the down-conversion.

The last block of upstream demodulator is an equalizer. The purpose of the equalizer is to remove signal distortion and residual timing and frequency offset that escaped the synchronizer.

1.4 DOCSIS Upstream Channel

The DOCSIS upstream channel is an one way communication link between the CM modulator in the subscriber premises and the CMTS demodulator in the cable operator's headend.

The presence of micro-reflections or echoes [3], [9] in DOCSIS upstream channels is one of the main impairments and yet somehow has not been well explained in literature. Impedance mismatches at the terminal component, poor return and isolation losses of HFC components, the corrosion of center conductor, damaged cable and using older equipments in a HFC network are a few factors that contribute to micro-reflections. One type of micro-reflection is caused when the upstream signal encounters impedance mismatches on its way to the CMTS. The mismatch causes a fraction of the signal's energy to be reflected back towards the CM. If this reflected signal encounters an impedance mismatch in downstream direction, the reflected signal is re-reflect back toward the CMTS. The re-reflection shows up at the CMTS as an echo. This phenomenon is illustrated in Figure 1.4.

The top part of Figure 1.4, shows a segment of a cable plant that connects 4 CM to a CMTS. The plant is a co-axial cable with 4 cable taps which are directional couplers divert a small portion of the downstream bound power to cable modem and add upstream bound power from the CM to the cable. The purpose of explaining echoes it is assumed the impedances of *Tap_2* and *Tap_4* do not match the impedance of the cable. The cable modem connected to *Tap_2* initiates an upstream transmission at time t_0 . This is shown in the distance vs time plot at the bottom of Figure 1.4. The burst transmitted from the house connected to *Tap_2* passes through *Tap_3* unaffected as *Tap_3* is perfectly matched and reaches *Tap_4* at time t_1 . *Tap_4* has an impedance mismatch that causes a reflection.

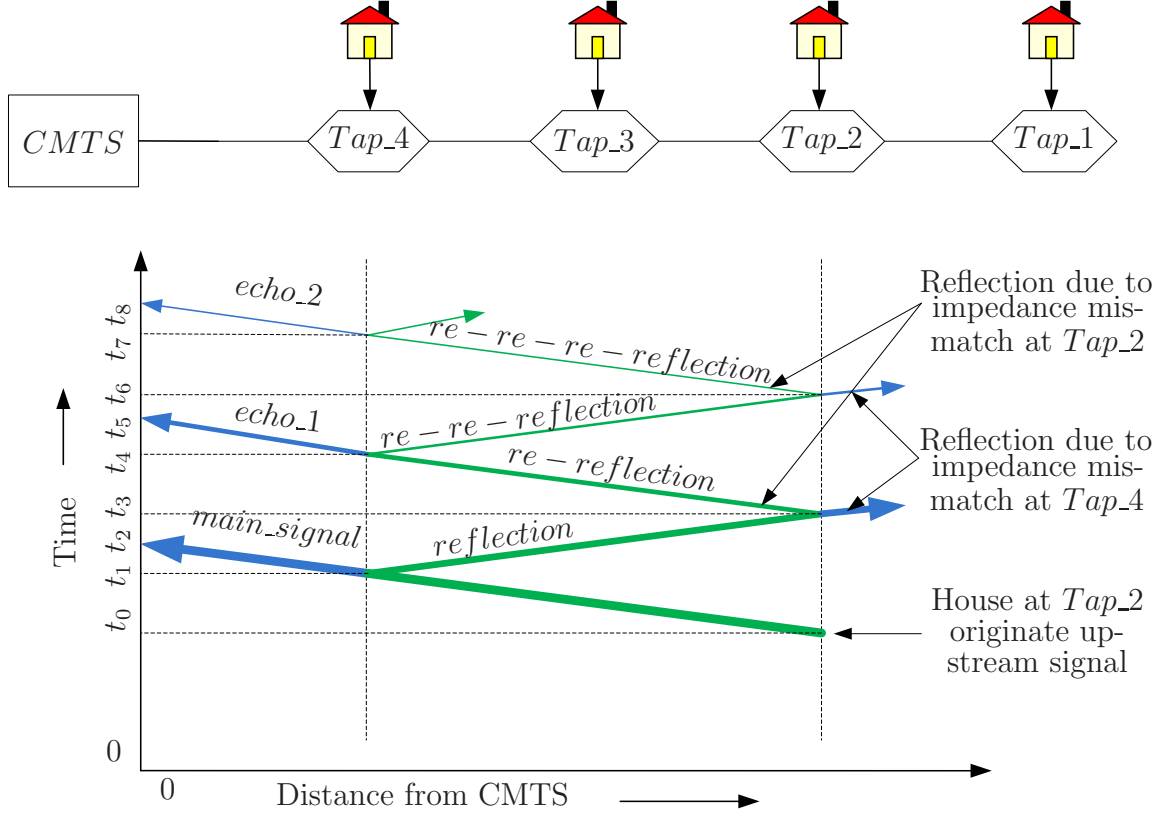


Figure 1.4 Process of micro-reflection generation in a cable plant

Subsequent to time t_1 , some of the burst power is propagating in downstream direction. At time t_2 the main part of the burst, which is denoted as *main_signal* reaches the CMTS. The signal reflected from *Tap_4* passes through *Tap_3* in the downstream direction and reaches *Tap_2* at time t_3 . Since *Tap_2* has an impedance mismatch it re-reflects a portion of the reflection towards the CMTS. At time t_5 the re-reflected burst reaches CMTS as an echo, which is denoted as *echo_1*. Through similar process a weaker echo, denoted as *echo_2* reaches CMTS at time t_8 .

The end effect of the mismatches is to create echoes. The echo delays, which are $(t_5 - t_2)$ and $(t_8 - t_2)$, are normally much shorter than the burst length and cause interference.

1.5 Statement of Research Problems

In a CATV system, upstream modulator transmits QAM data, which pass through the DOCSIS upstream channel. The upstream demodulator demodulates the received QAM data. The modulated QAM signal on its way to the demodulator encounters impairments, which include timing error, carrier frequency error, phase error, micro-reflections and thermal noise [9], [10]. These impairments need to be corrected before the signals can be successfully demodulated.

One of the major challenges in demodulating a signal received over the DOCSIS upstream channel is to mitigate the effect of the micro-reflections, which causes intersymbol interference (ISI) [11]. According to the DOCSIS standard, the micro-reflections can be as large as 10 dB below the carrier and can arrive up to seven symbol durations later. Another challenge is the correction of error in time recovery circuit. This error also causes ISI.

In order to mitigate ISI the upstream demodulator requires a timing synchronizer, a frequency synchronizer and an equalizer [9], [12], [13], [14]. Although timing and frequency synchronizer blocks correct most of timing and frequency offset, the equalizer block must suppress the error even further to minimize the ISI.

The equalizer in an upstream demodulator must adapt on a packet-by-packet basis. The received successive packet bursts could come from different CMs over different channels and require very different equalization. To facilitate the training of an equalizer, each packet is prefixed with a known preamble. The received preamble is used to estimate the impairment in the signal caused by the channel and synchronizer. By the time the end of preamble is reached, the equalizer must have converged and removed most of the ISI from the signal [15], [16], [17].

The problem is to construct an equalizer that converges quickly so the preamble length can be small and the data throughput is maximized.

The goal of this thesis is to implement the equalizer in a Stratix IV Field Programmable Gated Array (FPGA) board using the fewest possible logic elements.

1.6 Organization of the Thesis

The organization of remainder of the thesis is structured chapter by chapter as follows. In chapter 2, background study about the DOCSIS upstream channel is included. This chapter also explain in details channel impairments and different channel parameters which are specified by the DOCSIS standard.

Chapter 3 and 4, include details about equalization technique and performance criteria. Chapter 3, contains the mathematics behind equalization for an infinite length equalizer. It also sets the equation for measuring different performance criteria of equalizers. Specific equalization type, which is mentioned by DOCSIS is explained in chapter 4. This chapter also talks about the structure and performance criteria of the equalizer in details.

The implementation of DOCSIS upstream equalizer is included in chapter 5. Implementation of the equalizer is solely dependent on the different parameters of the equalizer. This chapter explains how different parameters of equalizer can have an impact on the performance of the equalizer.

Chapter 6, portrays different simulation results. This chapter is partitioned in two major sections. They are MATLAB based simulation results and DSP Builder based simulation results. The MATLAB simulation results includes for both theoretical and practical equalizer. It also compares the results and explain differences. The DSP Builder simulation results includes simulation results for practical equalizer and compares with the MATLAB results. Chapter 7 is based on hardware related results.

The thesis concludes with the conclusion chapter. All the successful outcomes of the thesis are included here. This chapter also adds some indications about the future work that can be done.

2. CATV Upstream Channel

2.1 Overview

A basic communication system consists of a transmitter, a receiver and a communication link between the transmitter and receiver. The same structure is applicable for a CATV system as well. A high-level structure of a DOCSIS upstream communication system is shown in Figure. 2.1. A modulator resides in the transmitter and a demodulator resides in the receiver. The modulator and the demodulator are connected by a cable. As this thesis is based on implementing an efficient equalizer, the purpose of which is to overcome the distortion caused by channel impairments, it is important to understand all the sources of impairments.

The next two sections of this chapter includes details about specific issues related to DOCSIS upstream channel with respect to physical layer and Media Access Control (MAC) layer.

2.2 Physical Layer of DOCSIS Upstream Channel

This section explains in details about the origin of major channel impairments with some background material related to the upstream cable channel.

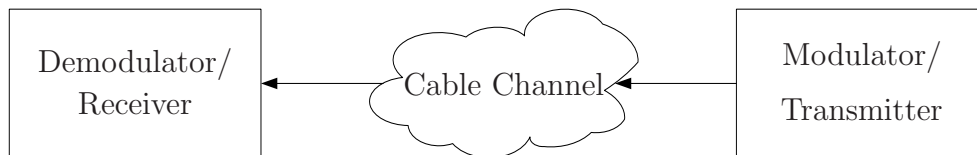


Figure 2.1 High-level block diagram of DOCSIS upstream communication system

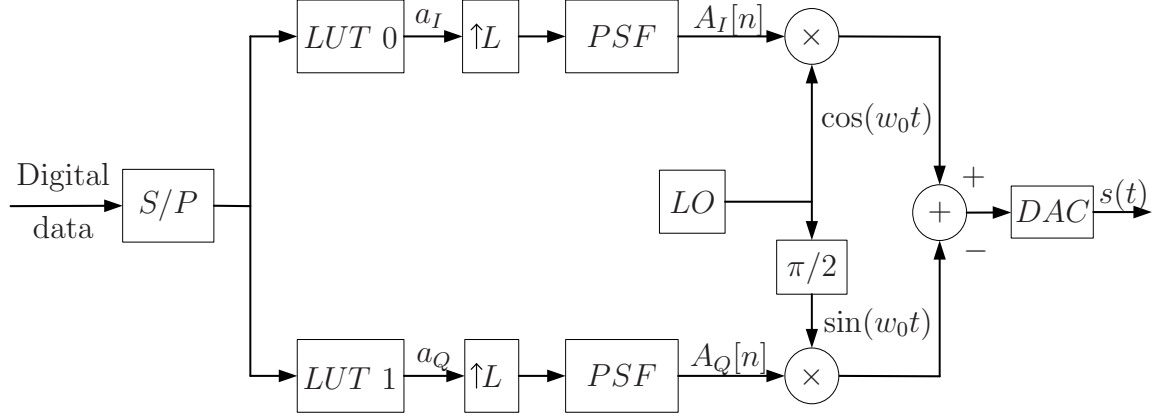


Figure 2.2 Basic block diagram of QAM modulator

2.2.1 Basic Modulation and Demodulation Technique

According to the DOCSIS standard the modulator shown in Figure 2.1 must be Quadrature Amplitude Modulation (QAM) in the upstream direction. The structure of a QAM modulator, which is well known [5], is illustrated in Figure 2.2.

The input to the serial-to-parallel (S/P) converter is a serial binary data stream. The bit to symbol mapping is done by this serial-to-parallel converter. For an M -ary QAM technique, the input bits are partitioned into non-overlapping segments of $\log_2 M$ bits by the serial-to-parallel converter. The rate at which the serial-to-parallel converter converts blocks of $\log_2 M$ bits into symbols is referred to as the symbol rate. The output of the serial-to-parallel converter becomes two $\log_2 M$ bit addresses for two look-up tables (LUT 0 and LUT 1). The look-up tables map the symbol values into numbers specified by the encoding rule. The output of the two look-up tables are denoted as a_I and a_Q , where a_I and a_Q are weighted impulses to be filtered by the Pulse Shaping Filter (PSF). Prior to filtering the impulses are up-sampled by zero stuffing by a factor of L . The outputs of the pulse shaping filter are used to amplitude modulate two sinusoidal carriers in quadrature. The modulated carriers are summed and then converted to an analog signal, denoted as $s(t)$, by a Digital-to-Analog Converter (DAC). The Local Oscillator, which is denoted as LO, generates a sinusoidal at the carrier frequency.

The symbol values that are stored in the look-up tables can be plotted in a 2-dimensional

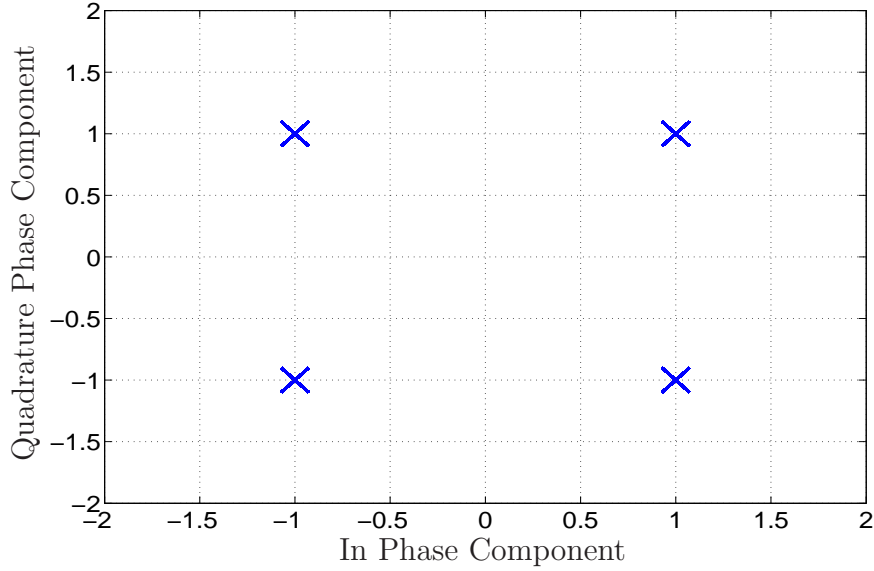


Figure 2.3 Constellation diagram for 4-QAM data

plane, which gives birth to a diagram called a constellation plot. An example of a constellation diagram for 4-QAM is illustrated in Figure 2.3. It should be mentioned here that each symbol is represented by a complex number. The in phase and quadrature phase component (a_I and a_Q) form the real and imaginary part of a symbol respectively.

The structure of a basic QAM demodulator is shown in Figure. 2.4. The received signal $r(t)$ is a noise corrupted version of transmitted signal $s(t)$. The first step in demodulation is to downconvert the received signal $r(t)$ to a baseband signal by multiplying it with two quadrature carriers generated by the Local Oscillator (LO) in the receiver. Prior to downconversion the received signal is passed through an analog to digital converter (ADC). It should be mentioned here that this local oscillator behaves in the same way it does in the transmitter. The downconverted in phase and quadrature phase components of the signal are denoted by $u_I[n]$ and $u_Q[n]$ respectively.

The downconverted baseband signal is passed through a low pass filter to remove any high frequency component that was generated during the downconversion. This low pass filter has an impulse response identical to the pulse shaping filter and is referred to as the matched filter. The matched filter outputs are denoted as $z_I[n]$ and $z_Q[n]$; they are down-

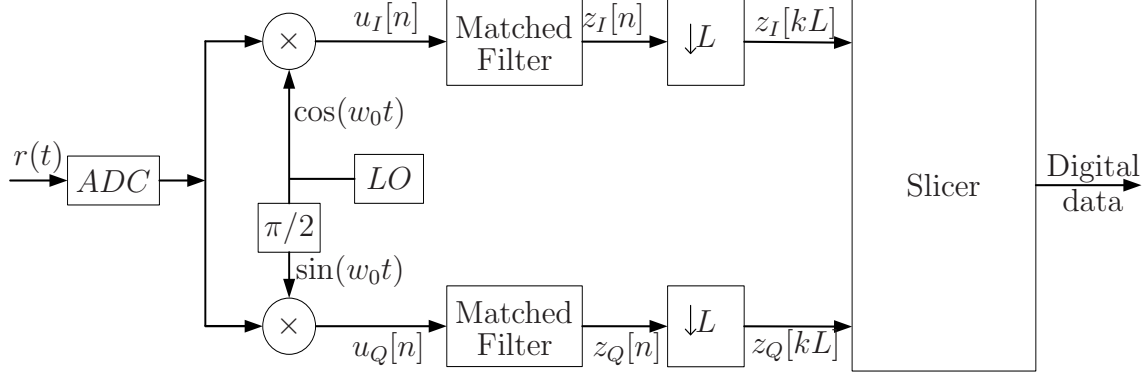


Figure 2.4 Basic block diagram of QAM demodulator

sampled by a factor of L to produce in phase and quadrature phase components of each symbol successively. The output of the down-sampler is passed to the decision device, which is referred to as slicer. The purpose of the slicer is to determine what symbol was transmitted.

Having explained modulation and demodulation technique used in DOCSIS upstream communication system, in the following subsection the channel impairments are explained in details.

2.2.2 Channel Impairments

As mentioned in section 1.5, there are numerous impairments that distort the transmitted upstream signal. The sources of most critical impairments together with their effect on distorting signals are explained in this sub-section.

Timing Error

One of the major channel impairments is timing error, which is the error created by sampling the received signal at the incorrect time. This type of error results from two phenomenon. One is that the oscillators that form the timing bases in transmitter and receiver are not synchronized. The other is the time to propagate through the cable plant is unlikely to be an integer number of samples.

This error, which is referred to as timing offset, causes ISI, which increase the probability

of the slicer making a wrong decision. Normally the upstream demodulator includes a circuit called the timing synchronizer that removes most of the timing error.

Frequency Error

Another important impairment that need to overcome is frequency conversion error, which is caused by the inaccuracy in the local oscillators in the transmitter and receiver. The error causes a frequency offset in the baseband signal which causes the constellation to spin and make the task of recovering the symbols very difficult.

Normally a circuitry called frequency synchronizer, whose purpose is to de-spin the constellation, is included in the upstream demodulator. The frequency synchronizer first estimates the frequency error and then passes this information to the despinner, which does not completely eliminate the spinning but slows the rate considerably.

Micro-reflections

One of the most commonly faced impairment is the presence of micro-reflection or echoes in the cable plant where many CM and the CMTS are connected to the same physical cable. Communication in the upstream direction is initiated by a CM. As the transmitted burst propagates through the large cable network to reach CMTS, it traverses taps, which directionally couple other CM's to the cable. Some of the CM's may not be perfectly impedance matched to the cable network. As a result, a portion of the transmitted burst get reflected from those impedance mismatches and eventually reaches to CMTS as echoes. The reflected copies of the main burst are called micro-reflections or echoes. The echoes are delayed and attenuated copies of the main transmitted burst.

Figure 2.5 shows three curves. One curve (solid line connecting markers 'o') is the output of the matched filter for the transmission of a single symbol that encounter a channel with an echo of magnitude 0.3162 with respect to the main signal and a delay of 1.5 symbols. The second curve (dashed line connecting marker '◇') is that same transmission in the absence of echo. The third curve (dashed line connecting marker '□') is the echo. It is clear from the combined response that it is not symmetric and does not cross zero at symbol times, which

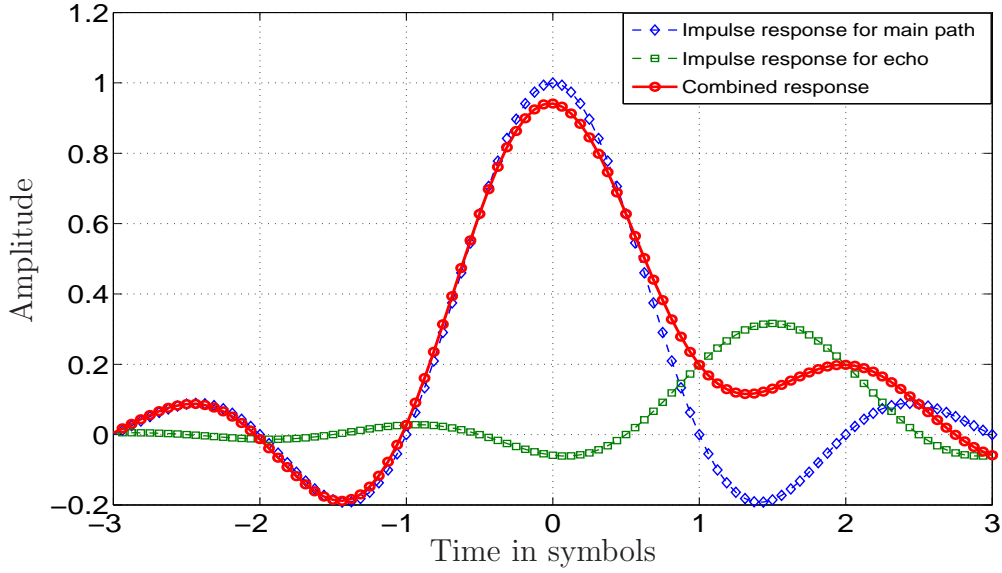


Figure 2.5 Impulse response of channel model to illustrate the effect of echo

causes ISI.

An example constellation diagram is shown in Figure 2.6 to visualize the effect of channel echoes. The decision variables is plotted for a burst of transmission of 1000 QPSK symbols that pass through the multipath channel. It is clear from Figure. 2.6 that the constellation points are scattered around the correct points described above. This scattering caused by the echo is by definition inter symbol interference.

As this thesis is based on a DOCSIS channel, it is important to know the characteristics of each echo. According to the DOCSIS standard the upstream channel consists of one main path and a maximum three echoes. The worst case parameters for each echo are specified in the DOCSIS standard as tabulated in Table 2.1.

Table 2.1 DOCSIS specified echoes with their parameters

	Amplitude (dBc)	Echo Delay (symbol)	Echo Phase (Rad)
Echo_1	-10	0-2.5	$0-2\pi$
Echo_2	-20	0-5	$0-2\pi$
Echo_3	-30	0-7.5	$0-2\pi$

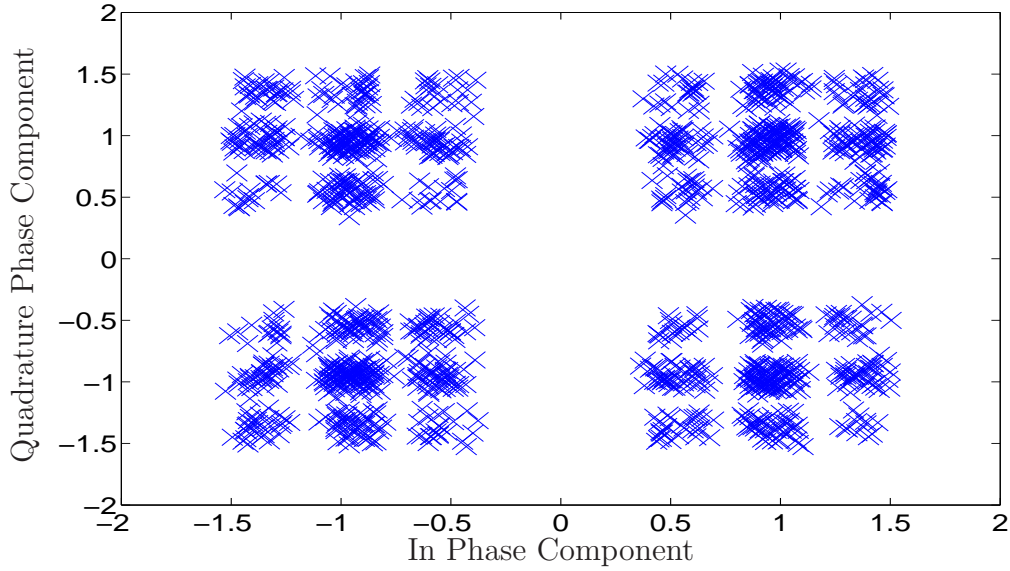


Figure 2.6 Constellation diagram showing the effect of echo

Having covered most critical impairments, it is understandable now how each impairment distorts the transmitted signal. It is the task of this thesis to mitigate the impairments so the decision variable can be reliably translated into the symbol that was transmitted. This task falls to an equalizer which must be included in the upstream demodulator.

2.3 MAC layer of DOCSIS Upstream Channel

The media access control (MAC) layer of the CMTS performs the network management tasks for the DOCSIS upstream channels. To explain all the tasks of MAC layer is beyond the scope of this thesis, but the issues relevant to this thesis are explained briefly in this section.

With the view to allow multiple users to transmit data to the CMTS in the headend via a single upstream channel, the DOCSIS standard incorporated two multiple access schemes, namely Time Division Multiple Access (TDMA) [3] and Synchronous Code Division Multiple Access (S-CDMA) [3]. Although both of these multiple access modes can be used in an upstream channel, this thesis only considers TDMA mode. In TDMA mode, each user is allowed to access a channel for a specific period of time.

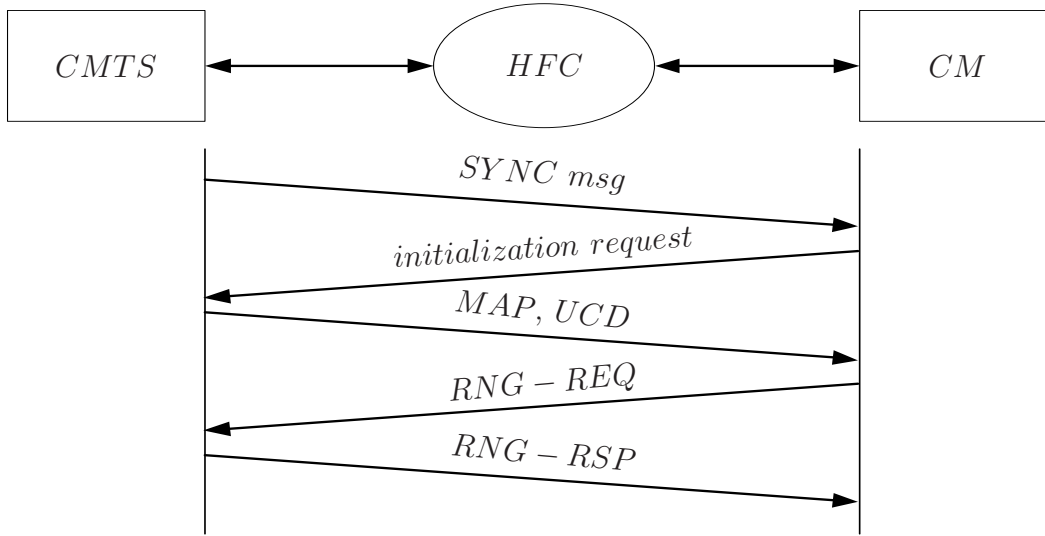


Figure 2.7 Process of ranging mode operation

The CMTS MAC partitions each TDMA upstream channel into a sequence of timeslots, which are called minislots by the DOCSIS standard. The MAC allocates each minislot to a CM upon request. In the allocated minislots there are two types of packet burst transmitted over the DOCSIS channel. They are called traffic mode packets and ranging mode packets [9].

Ranging Mode

As discussed in section 2.2.2, the presence of a large number of impairments in the upstream channel makes the task symbol recovery very difficult. The primary purpose of ranging mode is to configure circuitry in the CM, whose purpose is to mitigate the effects of the channel impairments. The MAC transmits the configuration information to the CM's via the down stream channels. The MAC calculates the configuration from measurements made at the CMTS while in ranging mode. A bidirectional communication process between a CM and a CMTS is shown in Figure. 2.7.

In a CATV network, at the time a cable modem is turned on, it monitors a downstream DOCSIS channel for purpose of synchronization. To facilitate synchronization the CMTS periodically sends SYNC messages on every downstream channel. After synchronization the

CM sends an initialization request on an ALOHA time slot reserved for this purpose. Upon receiving an initialization request the CMTS sends Upstream Channel Descriptor (UCD), which tells CM's the upstream channel frequency, symbol rate, modulation profile, and other parameters necessary to communicate on the network. Finally, the CMTS sends out Media Access Protocol (MAP) messages to allocate talk time to each cable modem. Since many cable modems are assigned to one upstream channel frequency, the cable modems must time-share the upstream channel using TDMA.

The cable modem is now ready to begin ranging with the CMTS. The ranging process begins with Initial Ranging, which is a process in which the cable modem begins by sending a Range-Request (RNG-REQ). If it does not receive a Range-Response (RNG-RSP) from the CMTS within 200 msec, the cable modem increases its transmit power and retransmits the RNG-REQ. This process is repeated until a RNG-RSP is received. Once the modem has received its first RNG-RSP from the CMTS it will be moved from Initial Ranging to Station Maintenance. The cable modem will also be instructed by the CMTS to make adjustments to its transmitting frequency, amplitude, timing offset and pre-equalization [3].

After successful reception of the ranging packet by the CM's, it is then registered with the CMTS. The CMTS then supervises the CM's to exit from the ranging mode and enter into the traffic mode.

Traffic Mode

In traffic mode, real datas are transmitted by a CM. Once CM's are ranged then they are configured to mitigate most of channel impairments and the transmission are relatively free of impairments. This implies that in traffic mode, the CMTS do not need to correct transmitter specific inaccuracies.

This thesis concerns ranging the CM's. In particular it focuses on the process of configuring the pre-equalizer in the CM's. So the rest of this thesis is centred on ranging mode packet only.

3. Theory of Equalization

There is much literature [16], [18], [19] available for channel equalization. One of the approaches uses linear transversal filter structure, which is commonly used for equalization, is the focus of this thesis. Research has been carried out to find the optimum filter coefficients for the equalizer. Research has delivered an algorithm for coefficient that produces a decision variable with minimum mean square error [18]. It is the Mean Square Error (MSE) criterion that will be used in this thesis.

3.1 Least Mean Square Error Equalizer

The Least Mean Square Error (LMSE) equalizer is based on the MSE criterion [18]. This section includes the theory that supports the MSE equalization criterion. The theory of equalization is explained in the next sub-section.

3.1.1 System Model

The digital system that will be used to derive a theoretical limit on the residual mean square error at the output of a mean square error equalizer is shown in Figure 3.1.

The input, $I[n]$, is an independent, zero mean, random complex data sequence with a variance of σ_I^2 [20]. That is

$$\begin{aligned} E[I[n]] &= 0 ; -\infty < n < \infty \\ E[I[n]I^*[k]] &= \begin{cases} \sigma_I^2 & \text{for } n = k \\ 0 & \text{for } n \neq k \end{cases} \end{aligned} \quad (3.1)$$

The data are up-sampled by a factor of L with zero stuffing and shaped with a Square

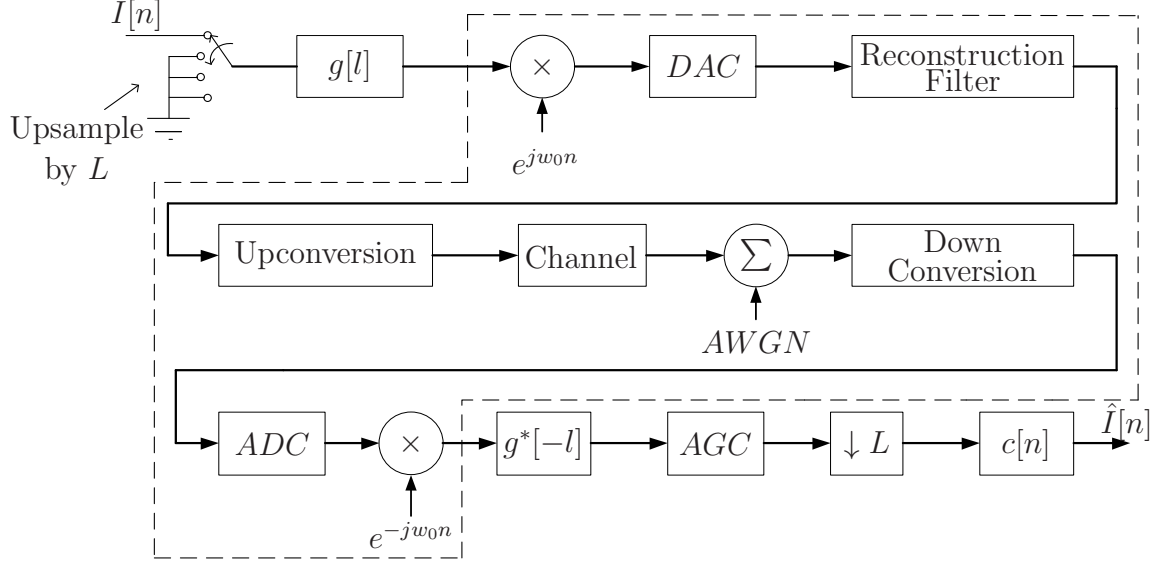


Figure 3.1 The Digital Communication System

Root Raised Cosine (SRRC) filter with a roll-off-factor, $r = 0.25$. The filter is labelled by its impulse response $g[l]$ and it is referred to in the literature as the pulse shaping filter. It operates at L times the symbol rate and has the following properties.

1. It has Discrete Time Fourier Transform (DTFT) denoted $G(e^{jw})$. The magnitude of $G(e^{jw})$ is given by

$$G(e^{jw}) = \begin{cases} \sqrt{L} ; & |w| \leq (1 - \beta)\frac{\pi}{L} \\ \sqrt{L} \sqrt{\frac{1}{2} + \frac{1}{2} \sin(\frac{Lw}{2\beta} - \frac{\pi}{2\beta})} ; & (1 - \beta)\frac{\pi}{L} < |w| < (1 + \beta)\frac{\pi}{L} \\ 0 ; & (1 + \beta)\frac{\pi}{L} \leq |w| \leq \pi \end{cases}$$

2. The energy in $g(l)$ is 1. That is $\sum_{l=-\infty}^{\infty} |g(l)|^2 = 1$. From Parseval's theorem [5] this also implies

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} |G(e^{jw})|^2 dw = 1.$$

3. The peak value of the impulse response of the cascade of the pulse shaping filter and the matched filter is 1. That is to say $\max (g[l] * g^*[-l]) = 1$. To be specific

$$\sum_{l=-\infty}^{\infty} g[m]g^*[-(l-m)]|_{l=0} = 1.$$

This is a consequence of the energy in $g[l]$ being 1.

4. Normally $g[l]$ is real, but it does not need to be.

The output of the pulse shaping filter is up converted in two stages: a digital stage and an analog stage. The up converted signal is transmitted over an analog medium (CATV cable) that has a real impulse response $h_a(t)$.

The receiver in Figure 3.1 begins with the additive noise. It first down converts the signal to an IF, then quantizes it and then digitally down-converts it to complex baseband. The down conversion to complex baseband generates a complex signal which is the complex low-pass equivalent of the real band-pass modulated signal.

The complex baseband signal is filtered with matched filter, that is actually a pair of filter that acts on the real and imaginary part separately. The matched filter is denoted by its impulse response, which is $g^*[-l]$. Automatic Gain Control (AGC) follows the matched filter. The gain of automatic gain control block is AGC . It is assumed the AGC is controlled to keep power of its output to the constant '1'. The output of the AGC is down-sampled by a factor of L to the symbol rate and then equalized by the block labelled as equalizer, $c[n]$.

Simplified System Model

A simplified model is obtained by rearranging the natural order of the blocks in Figure 3.1 and then consolidating them. The first step is to move the AGC to the input side of filter $g^*[-l]$. The next step is to relocate the AWGN to the output of the down-sampler and represent it with its complex low-pass equivalent. In the relocation the noise source passes through a real down-converter, ADC, a complex down-converter, an AGC, $g^*[-l]$ and the down-sampler. The system with AGC repositioned and noise relocated and represented by its complex low-pass equivalent, is shown in Figure 3.2. The low-pass equivalent of the noise, which is complex noise after the complex down-conversion, is denoted as $\eta'[l]$. The complex baseband noise is filtered with $g^*[-l]$ and down-sampled to get $\eta[n]$, which is added to the signal component output of the channel.

The blocks in Figure 3.2 enclosed by the dashed line can be modelled by a filter with

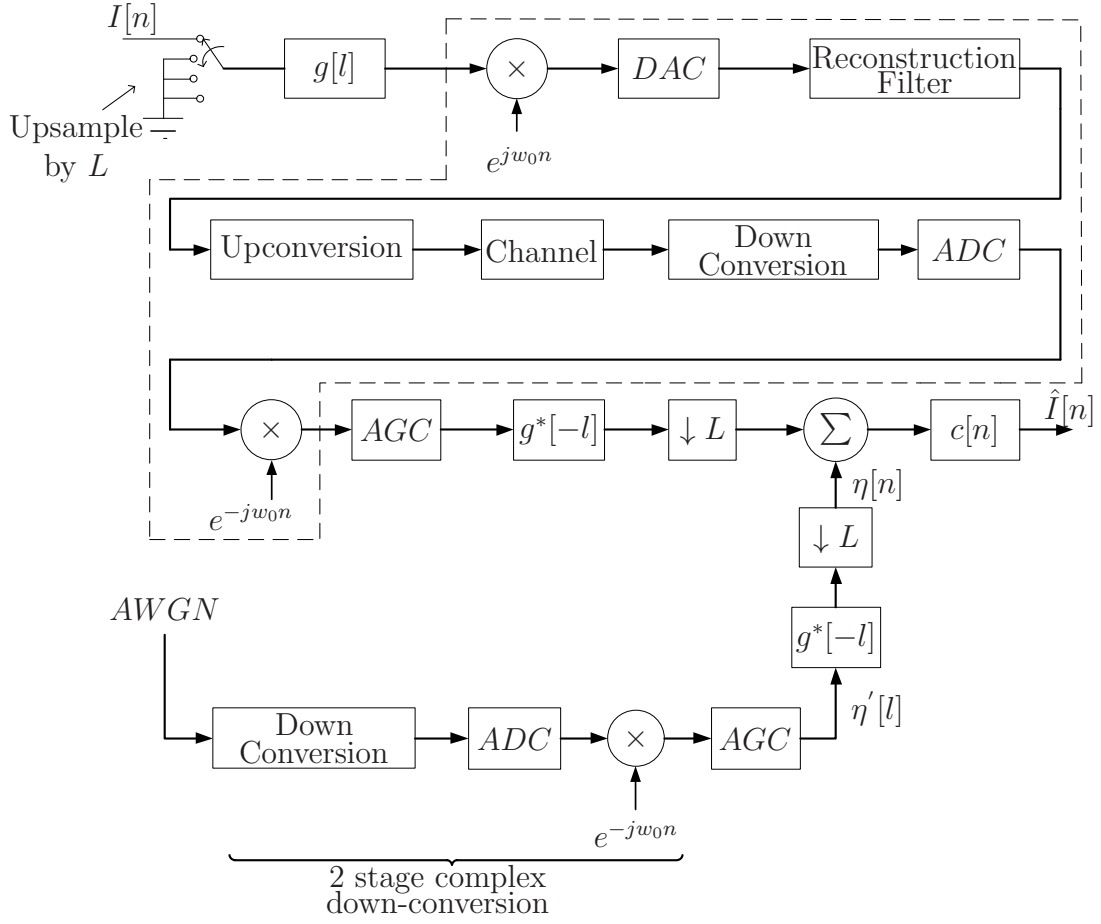


Figure 3.2 The Modified Digital Communication System

impulse response $e^{j\Delta w l}h[l]$, where Δw is the difference in the up conversion and down conversion frequencies and $h[l]$ is a time invariant complex low pass filter. Without loss of significant generality, the time expansion/contraction incurred from the small differences in frequency in the LOs that establish the time bases in the transmitter and receiver, is not included in the model.

It can be shown that a frequency offset does not affect the theoretical performance of a Least Mean Square (LMS) equalizer [18]. For that reason Δw is set to zero in the analysis that follows.

The critical block is the replacement filter $h[l]$, is the channel, which has a wide band

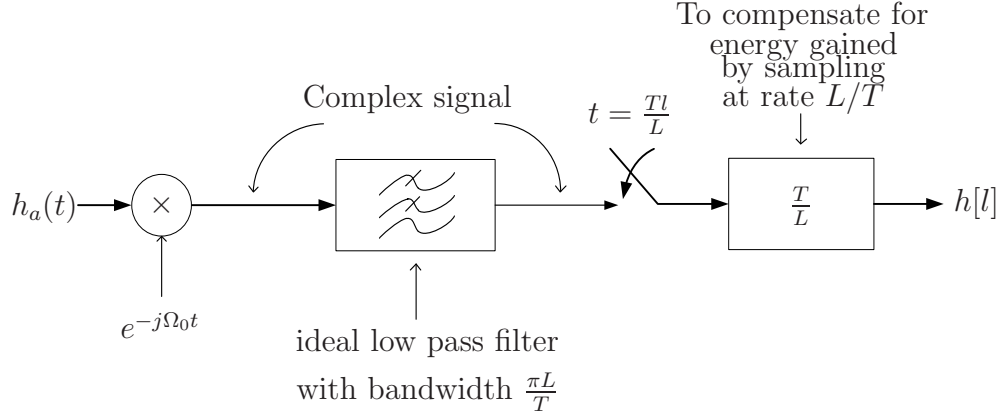


Figure 3.3 Generating $h[l]$

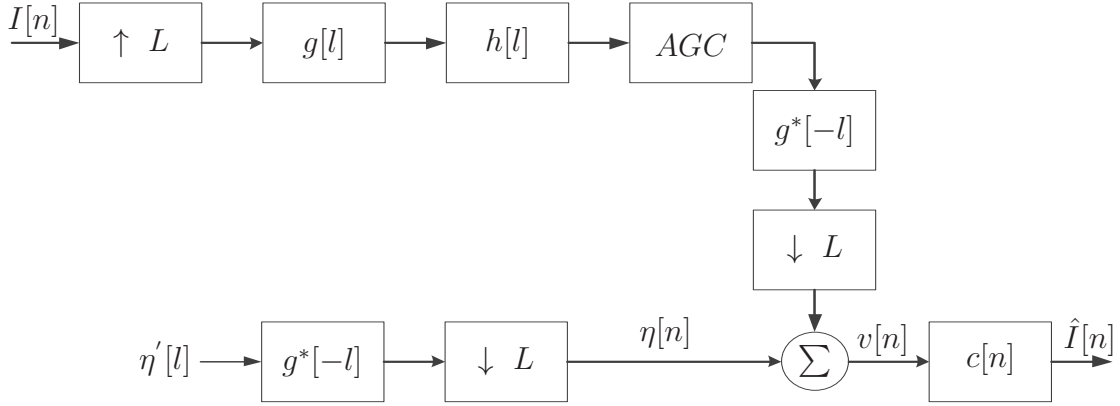


Figure 3.4 Somewhat Refined System Model

impulse response denoted as $h_a(t)$. While $h_a(t)$ is wide band, the real down-conversion contains a band-pass filter with a bandwidth equal to the nyquist zone of the analog to digital converter (ADC). That is, the band-pass filter has a bandwidth of $\frac{L}{2T}$ Hz or $\frac{\pi L}{T} \frac{\text{rad}}{\text{sec}}$, where $\frac{1}{T}$ is the symbol rate and $\frac{L}{T}$ is the sampling rate of ADC. That being the case, $h[l]$ is obtained from $h_a(t)$ as shown in Figure 3.3.

After replacing the blocks inside the dashed line with a filter having impulse response $h[l]$, the modified digital communication system becomes the one shown in Figure 3.4.

The next step is to represent the path from the system input to the adder input as the

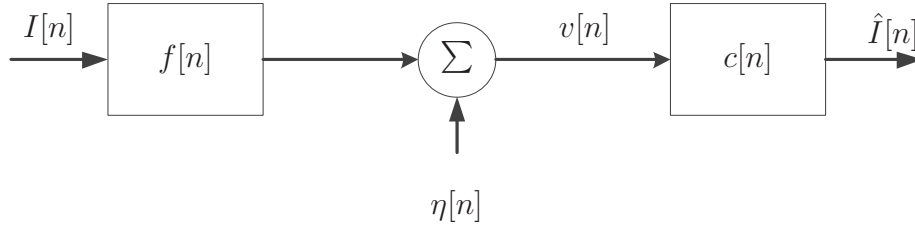


Figure 3.5 Symbol rate System Model

low-rate filter with impulse response $f[n]$. This allows the system be modelled at the symbol rate as shown in Figure 3.5, where:

1. $f[n] = AGC * g[l] \star h[l] \star g^*[-l]|_{l=L_n}$. Note that $f[n]$ is complex.
2. $c[n]$ is the impulse response of the equalizer.
3. $v[n]$ is the input to the equalizer.
4. $\eta[n]$ is an independent gaussian noise sequence with $E[\eta[n]\eta^*[n]] = \sigma_n^2$
5. The output of the equalizer is $\hat{I}[n]$ which is an estimate of $I[n]$.

3.1.2 Mean Squared Error (MSE) Criterion

Let $\hat{I}[n]$ is an estimated symbol at the output of the equalizer. Now, the error in estimate $\hat{I}[n]$ is

$$\varepsilon[n] = \hat{I}[n] - I[n] \quad (3.2)$$

Both $I[n]$ and $\hat{I}[n]$ are random variables which means $\varepsilon[n]$ is also a random variable.

The mean of the square of the magnitude of the error is referred to as mean squared error. It is given by [18]

$$\begin{aligned} J &= E[|\varepsilon[n]|^2] \\ J &= E[|\hat{I}[n] - I[n]|^2] \end{aligned} \quad (3.3)$$

From the working system model shown in Figure. 3.5, $\hat{I}[n]$ is given by

$$\hat{I}[n] = \sum_{k=-\infty}^{\infty} c[k]v[n-k] \quad (3.4)$$

Substituting equation(3.4) into equation(3.3) yields

$$\begin{aligned} J &= E[|(\sum_{k=-\infty}^{\infty} c[k]v[n-k]) - I[n]|^2] \\ J &= E[(\sum_{k=-\infty}^{\infty} c[k]v[n-k]) - I[n])(\sum_{k=-\infty}^{\infty} c[k]v[n-k]) - I[n]]^*] \end{aligned} \quad (3.5)$$

Finding the Equalizer Coefficients to Minimizing MSE

There is an infinite number of complex equalizer coefficients that can be represented by

$$c[k] = c_R[k] + jc_I[k]; \quad -\infty < k < \infty \quad (3.6)$$

where $c_R[k]$ and $c_I[k]$ are real coefficients.

The problem is to find the sets of $c_R[n]$ and $c_I[n]$ that yield the minimum J . These are the set of coefficients that satisfy

$$\frac{\delta J}{\delta c_R[n]} = 0 \text{ and } \frac{\delta J}{\delta c_I[n]} = 0 \text{ for } -\infty < n < \infty \quad (3.7)$$

A useful expression for $\frac{\delta J}{\delta c_R[n]}$ is found using equation (3.5) by first exchanging the variables k and n so that the dummy variable in the summation is n and the coefficients of the equalizer are indexed by n . Second the derivative is moved inside the expectation. Doing so yields

$$\begin{aligned} \frac{\delta J}{\delta c_R[n]} &= E[\{\frac{\delta}{\delta c_R[n]}(\sum_{n=-\infty}^{\infty} c[n]v[k-n] - I[k])\} \times (\sum_{n=-\infty}^{\infty} c[n]v[k-n] - I[k])^* \\ &+ (\sum_{n=-\infty}^{\infty} c[n]v[k-n] - I[k]) \times \{\frac{\delta}{\delta c_R[n]}(\sum_{n=-\infty}^{\infty} c[n]v[k-n] - I[k])^*\}]. \end{aligned} \quad (3.8)$$

Taking the partial derivatives yields

$$\begin{aligned} \frac{\delta J}{\delta c_R[n]} &= E[v[k-n](\sum_{n=-\infty}^{\infty} c[n]v[k-n] - I[k])^* \\ &+ v^*[k-n](\sum_{n=-\infty}^{\infty} c[n]v[k-n] - I[k])] \end{aligned} \quad (3.9)$$

Changing the dummy index variable from n to m in (3.9) allows $v[k-n]$ and $v^*[k-n]$ that are outside the summation to be moved inside the summation. Doing so and distributing the conjugation produces:

$$\begin{aligned} \frac{\delta J}{\delta c_R[n]} = E[& \sum_{m=-\infty}^{\infty} c^*[m]v^*[k-m]v[k-n] - v[k-n]I^*[k] \\ & + \sum_{m=-\infty}^{\infty} c[m]v[k-m]v^*[k-n] - v^*[k-n]I[k]] \end{aligned} \quad (3.10)$$

Taking the expectation of the four terms separately and then moving expectations inside the summations yields

$$\begin{aligned} \frac{\delta J}{\delta c_R[n]} = & \sum_{m=-\infty}^{\infty} c^*[m]E[v^*[k-m]v[k-n]] \\ & - E[v[k-n]I^*[k]] \\ & + \sum_{m=-\infty}^{\infty} c[m]E[v[k-m]v^*[k-n]] \\ & - E[v^*[k-n]I[k]] \end{aligned} \quad (3.11)$$

Inspection of equation (3.11) reveals that the four terms are two pairs of complex conjugates, which makes the right hand side real. Setting $\frac{\delta J}{\delta c_R[n]} = 0$ and moving the negative terms to the left hand side yields

$$\mathbb{R}\{E[v^*[k-n]I[k]]\} = \mathbb{R}\left\{\sum_{m=-\infty}^{\infty} c[m]E[v[k-m]v^*[k-n]]\right\}. \quad (3.12)$$

Solving $\frac{\delta J}{\delta c_I[n]}$ in a similar manner yields

$$\mathbb{I}\{E[v^*[k-n]I[k]]\} = \mathbb{I}\left\{\sum_{m=-\infty}^{\infty} c[m]E[v[k-m]v^*[k-n]]\right\}. \quad (3.13)$$

Equations (3.12) and (3.13) can be combined into a single equation.

$$E[v^*[k-n]I[k]] = \sum_{m=-\infty}^{\infty} c[m]E[v[k-m]v^*[k-n]]. \quad (3.14)$$

The solution to equation (3.14) for $c[n]$, $-\infty < n < \infty$, provides an equalizer whose output is an estimate of $I[k]$ that minimizes $E[|\hat{I}[k] - I[k]|^2]$.

An alternate form of equation (3.14) that will be useful later on is

$$E \left[\sum_{m=-\infty}^{\infty} c[m] v[k-m] v^*[k-n] - v^*[k-n] I[k] \right] = 0$$

$$E \left[v^*[k-n] (\hat{I}[k] - I[k]) \right] = 0. \quad (3.15)$$

Solving For $c[n]$

The problem at hand is to find an equation that gives the optimum set of $c[n]$. The derivation starts by getting expressions for $v[k-m]$ and $v^*[k-n]$. From the symbol rate system model shown in Figure 3.5

$$v[n] = I[n] \star f[n] + \eta[n]$$

Therefore,

$$v[k-m] = \sum_{i=-\infty}^{\infty} I[i] f[k-m-i] + \eta[k-m]$$

and

$$v^*[k-n] = \sum_{q=-\infty}^{\infty} I^*[q] f^*[k-n-q] + \eta^*[k-n]. \quad (3.16)$$

Using equation (3.16), the product $v[k-m]v^*[k-n]$ can be expressed as the sum of four terms:

$$\begin{aligned} v[k-m]v^*[k-n] &= \sum_{q=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} I[i] I^*[q] f[k-m-i] f^*[k-n-q] \\ &\quad + \sum_{q=-\infty}^{\infty} \eta[k-m] I^*[q] f^*[k-n-q] \\ &\quad + \sum_{i=-\infty}^{\infty} I[i] \eta^*[k-n] f[k-m-i] \\ &\quad + \eta[k-m] \eta^*[k-n]. \end{aligned} \quad (3.17)$$

Using the statistical properties:

$$E[I[i]I[q]] = \begin{cases} \sigma_I^2 & \text{for } i = q \\ 0 & \text{for } i \neq q, \end{cases}$$

$$E[I[i]\eta[q]] = E[I^*[i]\eta[q]] = \begin{cases} 0 & \text{for all } i \text{ \& } q, \end{cases}$$

and

$$E[\eta[i]\eta^*[q]] = \begin{cases} \sigma_\eta^2 & \text{for } i = q \\ 0 & \text{for } i \neq q, \end{cases}$$

in taking the expectation of equation (3.17) yields

$$\begin{aligned} E[v[k-m]v^*[k-n]] &= \sigma_I^2 \sum_{i=-\infty}^{\infty} f[-m-i]f^*[-n-i] \\ &\quad + \sigma_\eta^2 \delta[n-m] \end{aligned} \quad (3.18)$$

where $\delta[\cdot]$ is Kronecker delta [21] and

$$\delta[n-m] = \begin{cases} 1 & \text{for } m = n \\ 0 & \text{for } m \neq n. \end{cases}$$

Finally computing the sum over m has

$$\begin{aligned} \sum_{m=-\infty}^{\infty} c[m]E[v[k-m]v^*[k-n]] &= \sigma_I^2 \sum_{i=-\infty}^{\infty} \left\{ \sum_{m=-\infty}^{\infty} c[m]f[-m-i] \right\} f^*[-n-i] \\ &\quad + \sigma_\eta^2 \sum_{m=-\infty}^{\infty} c[m]\delta[n-m] \end{aligned} \quad (3.19)$$

and making a change of index variables $q = -i$, then evaluating $\sum_{m=-\infty}^{\infty} c[m]\delta[n-m] = c[n]$, results in

$$\begin{aligned} \sum_{m=-\infty}^{\infty} c[m]E[v[k-m]v^*[k-n]] &= \sigma_I^2 \sum_{q=-\infty}^{\infty} \left\{ \sum_{m=-\infty}^{\infty} c[m]f[q-m] \right\} f^*[q-n] \\ &\quad + \sigma_\eta^2 c[n]. \end{aligned} \quad (3.20)$$

The sum over m enclosed in $\{\}$ is the convolution of $c[q]$ and $f[q]$. Defining $f_{rev}[n]$ to be the time reversal of $f[n]$, i.e. $f_{rev}[n] = f[-n]$ has

$$\begin{aligned} \sum_{m=-\infty}^{\infty} c[m]E[v[k-m]v^*[k-n]] &= \sigma_I^2 \sum_{q=-\infty}^{\infty} \left\{ c[q] \star f[q] \right\} f_{rev}^*[n-q] \\ &\quad + \sigma_\eta^2 c[n]. \end{aligned} \quad (3.21)$$

Clearly the sum over q is a convolution, which allows equation (3.21) to be expressed as

$$\begin{aligned} \sum_{m=-\infty}^{\infty} c[m]E\left[v[k-m]v^*[k-n]\right] &= \sigma_I^2 c[n] \star f[n] \star f_{rev}^*[n] \\ &+ \sigma_\eta^2 c[n]. \end{aligned} \quad (3.22)$$

The next step in the derivation is to find a suitable expression for the left hand side of equation (3.14):

$$v^*[k-n]I[k] = \sum_{q=-\infty}^{\infty} I[k]I^*[q]f^*[k-n-q] + \eta^*[k-n]I[k]. \quad (3.23)$$

Taking the expected value yields

$$\begin{aligned} E\left[v^*[k-n]I[k]\right] &= \sigma_I^2 f^*[-n] \\ &= \sigma_I^2 f_{rev}^*[n]. \end{aligned} \quad (3.24)$$

Substituting equation (3.22) and equation (3.24) into equation (3.14) has

$$\sigma_I^2 f_{rev}^*[n] = \sigma_I^2 c[n] \star f[n] \star f_{rev}^*[n] + \sigma_\eta^2 c[n]. \quad (3.25)$$

The next step is to take the Z-transform [21] of both sides of the equation. To do that the time reversal and conjugation properties of the Z-transform must be utilized. The Z-transform of $f_{rev}^*[n]$ is equal to the Z-transform of $f^*[-n]$, is by definition

$$\therefore Z\left\{f_{rev}^*[n]\right\} = \sum_{n=-\infty}^{\infty} f^*[-n]z^{-n}.$$

Let $m = -n$ then

$$Z\left\{f_{rev}^*[n]\right\} = \sum_{m=-\infty}^{\infty} f^*[m](z^{-1})^{-m}.$$

Moving the conjugation of $f[m]$ outside the summation has

$$\begin{aligned} Z\left\{f_{rev}^*[n]\right\} &= \left[\sum_{m=-\infty}^{\infty} f[m]((z^*)^{-1})^{-m} \right]^* \\ &= \left[F((z^*)^{-1}) \right]^* \\ &= F^*((z^*)^{-1}). \end{aligned} \quad (3.26)$$

Using equation (3.26) in transforming equation (3.25) has

$$\sigma_I^2 F^*((z^*)^{-1}) = \sigma_I^2 C(z) F(z) F^*((z^*)^{-1}) + \sigma_\eta^2 C(z). \quad (3.27)$$

Solving equation (3.27) for $C(z)$ yields the Z-transform of $c[n]$, which is

$$C[z] = \frac{\sigma_I^2 F^*((z^*)^{-1})}{\sigma_I^2 F(z) F^*((z^*)^{-1}) + \sigma_\eta^2} \quad (3.28)$$

The expression for the Z-transform of $\hat{I}[n]$ is denoted $\hat{I}[z]$ and is given by

$$\begin{aligned} \hat{I}[z] &= I[z] \left(\frac{\sigma_I^2 F^*((z^*)^{-1})}{\sigma_I^2 F(z) F^*((z^*)^{-1}) + \sigma_\eta^2} \right) F(z) \\ &\quad + \eta[z] \left(\frac{\sigma_I^2 F^*((z^*)^{-1})}{\sigma_I^2 F(z) F^*((z^*)^{-1}) + \sigma_\eta^2} \right), \end{aligned} \quad (3.29)$$

where $I[z]$ is the Z-transform of $I[n]$.

If $\eta[z] = 0$ (i.e. no noise), then $\sigma_\eta^2 = 0$, and the Z-transform of the equalizer given by equation (3.28) becomes $1/F(z)$. This completely compensates for “channel” in Figure 3.5, which has system function $F(z)$, and there is no ISI.

However, if $\sigma_\eta^2 \neq 0$, then the equalizer will not have Z-transform $1/F(z)$ and there will be ISI, i.e. some of the noise on $\hat{I}[k]$ will come from the sequence $I[n]$.

The mean square error is minimized. It is just that some of the error is due to ISI and some is due to the AWGN.

The Mean Squared Error After Equalization

The mean squared error is minimized by using the optimum equalizer coefficients. The coefficients will be denoted $C_{opt}[n]$. The minimum error is

$$\begin{aligned} J_{min} &= E[|\varepsilon[k]|^2] \Big|_{C_{opt}[n]} \\ &= E[(\hat{I}[k] - I[k])(\hat{I}[k] - I[k])^*] \\ &= E[\hat{I}^*[k](\hat{I}[k] - I[k])] - E[I^*[k](\hat{I}[k] - I[k])]. \end{aligned} \quad (3.30)$$

Assuming $E[\hat{I}^*[k](\hat{I}[k] - I[k])] = 0$ (it will shown to be true shortly), J_{min} becomes

$$J_{min} = E[-I^*[k](\hat{I}[k] - I[k])]. \quad (3.31)$$

The assumption $E\left[\hat{I}^*[k](\hat{I}[k] - I[k])\right] = 0$ is now proven to be true. The optimum coefficients were chosen to satisfy equation (3.15), which is

$$E\left[v^*[n-k](\hat{I}[k] - I[k])\right] = 0$$

Multiplying both sides of the above equation by $c^*[n]$ and summing both sides over n produces

$$\sum_{n=-\infty}^{\infty} c^*[n]E\left[v^*[n-k](\hat{I}[k] - I[k])\right] = 0. \quad (3.32)$$

Since $c^*[n]$ is deterministic it can be moved inside the expectation. After doing that, the summation can be moved inside the expectation as well. The result is

$$E\left[\left(\sum_{n=-\infty}^{\infty} c^*[n]v^*[n-k]\right)(\hat{I}[k] - I[k])\right] = 0 \quad (3.33)$$

since $\sum_n c^*[n]v^*[n-k] = \hat{I}^*[k]$, which has

$$E\left[\hat{I}^*[k](\hat{I}[k] - I[k])\right]_{c[n]=C_{opt}[n]} = 0 \quad (3.34)$$

Evaluating J_{min}

The last step is to find a useful expression for J_{min} . Such an expression is obtained by manipulating the right hand side of equation (3.31) with $c[n] = C_{opt}[n]$. Expanding the product in the expectation gives

$$J_{min} = E\left[\left|I[k]\right|^2\right] - E\left[I^*[k]\hat{I}[k]\right] \quad (3.35)$$

Since $E\left[\left|I[k]\right|^2\right]$ is known to be σ_I^2 , only $E\left[I^*[k]\hat{I}[k]\right]$ needs to be evaluated. This starts by expressing $\hat{I}[k]$ as the convolution of $c[k]$ and $v[k]$. The equation that results is

$$\begin{aligned} E\left[I^*[k]\hat{I}[k]\right] &= E\left[I^*[k]\sum_{m=-\infty}^{\infty} c[m]v[k-m]\right] \\ &= \sum_{m=-\infty}^{\infty} c[m]E\left[I^*[k]v[k-m]\right] \\ &= \sum_{m=-\infty}^{\infty} c[m]\left(E\left[I[k]v^*[k-m]\right]\right)^*. \end{aligned} \quad (3.36)$$

From equation (3.24)

$$E\left[v^*[k-n]I[k]\right] = \sigma_I^2 f_{rev}^*[n].$$

Therefore

$$\begin{aligned} E\left[I^*[k]\hat{I}[k]\right] &= \sum_{m=-\infty}^{\infty} c[m]\left(\sigma_I^2 f_{rev}^*[m]\right)^* \\ &= \sigma_I^2 \sum_{m=-\infty}^{\infty} c[m]f[-m]. \end{aligned} \quad (3.37)$$

The sum over m can be evaluated using a property of the DTFT, that states

$$\sum_{n=-\infty}^{\infty} x[n] = X(c^{jw})\Big|_{w=0} \quad (3.38)$$

Proof:

$$\begin{aligned} X(e^{jw})\Big|_{w=0} &= \sum_{n=-\infty}^{\infty} x[n]e^{-jwn}\Big|_{w=0} \\ &= \sum_{n=-\infty}^{\infty} x[n]e^{-j0} = \sum_{n=-\infty}^{\infty} x[n] \end{aligned}$$

The DTFT of the product of two sequences is the circular convolution of the DTFTs for each sequence. Since $C(z)\Big|_{z=e^{jw}}$ is the DTFT of $c[m]$, and $F(z^{-1})\Big|_{z=e^{jw}}$ is the DTFT of $f[-m]$, the DTFT of the product is

$$DTFT\{c[m]f[-m]\} = \frac{1}{2\pi} \int_{-\pi}^{\pi} F(e^{-j\lambda})C(e^{j(w-\lambda)}) d\lambda. \quad (3.39)$$

Evaluating at $w = 0$ yields

$$\sum_{m=-\infty}^{\infty} c[m]f[-m] = \frac{1}{2\pi} \int_{-\pi}^{\pi} F(e^{-j\lambda})C(e^{-j\lambda}) d\lambda. \quad (3.40)$$

Using equation (3.28) evaluated at $z = e^{-j\lambda}$

$$C(e^{-j\lambda}) = \frac{\sigma_I^2 F^*(e^{-j\lambda})}{\sigma_I^2 F(e^{-j\lambda})F^*(e^{-j\lambda}) + \sigma_\eta^2} \quad (3.41)$$

Substituting equation (3.41) into equation 3.40 yields

$$\sum_{m=-\infty}^{\infty} c[m]f[-m] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{\sigma_I^2 F(e^{-j\lambda})F^*(e^{-j\lambda})}{\sigma_I^2 F(e^{-j\lambda})F^*(e^{-j\lambda}) + \sigma_\eta^2} d\lambda. \quad (3.42)$$

An observation that will be helpful later is that the integrand on the right hand side of equation (3.42) is real. This means the sum $\sum_{m=-\infty}^{\infty} c[m]f[-m] = \sum_{m=-\infty}^{\infty} c[-m]f[m]$ is real.

The expression for J_{min} is obtained by substituting equation (3.42) into equation (3.37) and then substituting that result into equation (3.35). This results in

$$J_{min} = \sigma_I^2 \left[1 - \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{\sigma_I^2 F(e^{-j\lambda}) F^*(e^{-j\lambda})}{\sigma_I^2 F(e^{-j\lambda}) F^*(e^{-j\lambda}) + \sigma_\eta^2} d\lambda \right]. \quad (3.43)$$

To get the final expression for J_{min} , the constant ‘1’ is expressed in a way it can be absorbed into the integrand of equation (3.43). The constant ‘1’ can be expressed as

$$\begin{aligned} 1 &= \frac{1}{2\pi} \int_{-\pi}^{\pi} d\lambda \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{\sigma_I^2 F(e^{-j\lambda}) F^*(e^{-j\lambda}) + \sigma_\eta^2}{\sigma_I^2 F(e^{-j\lambda}) F^*(e^{-j\lambda}) + \sigma_\eta^2} d\lambda. \end{aligned} \quad (3.44)$$

Substituting equation (3.44) for the constant ‘1’ and merging with the integral in equation (3.43) yields

$$J_{min} = \frac{\sigma_I^2}{2\pi} \int_{-\pi}^{\pi} \frac{\sigma_\eta^2}{\sigma_I^2 F(e^{-j\omega}) F^*(e^{-j\omega}) + \sigma_\eta^2} d\omega. \quad (3.45)$$

Performance Measures

Signal to Noise Ratio (SNR): The SNR is the ratio of signal power to noise power and since it is a ratio of two powers, hence it is unitless. The SNR at the input to a receiver is often used as a performance measure. The SNR is not a precise measure of signal quality, but being a single real number makes it easy to understand.

The power at the output of filter $f[n]$ is

$$\begin{aligned} P_f &= E [(I[k] \star f[k])(I[k] \star f[k])^*] \\ &= \sigma_I^2 \sum_{n=-\infty}^{\infty} f[n] f^*[n]. \end{aligned} \quad (3.46)$$

The power in noise $\eta[n]$ depends on the power in $\eta'[l]$ since $\eta[n]$ is $\eta'[l]$ filtered by $g^*[-l]$ and down-sampled by L .

The noise $\eta'[l]$ is an independent sequence. After passing through $g^*[-l]$ it becomes correlated. However, it can be shown that after down-sampling, which produces $\eta[n]$, the sequence is again independent and gaussian with a variance of that of $\eta'[l]$. I.e.

$$E[\eta[n]\eta^*[m]] = \begin{cases} \sigma_\eta^2 & \text{for } n = m \\ 0 & \text{for } n \neq m. \end{cases} \quad (3.47)$$

The SNR at the input to the equalizer in terms of the symbol rate system model of Figure 3.5 on (page 29) is the ratio of the power at the output of filter $f[n]$ to the power in the *AWGN*, $\eta[n]$. The SNR is therefore given by

$$SNR_{input} = \frac{\sigma_I^2}{\sigma_\eta^2} \sum_{n=-\infty}^{\infty} f[n]f^*[n]$$

It can also be expressed in the frequency domain using Parseval's theorem. This has

$$SNR_{input} = \frac{\sigma_I^2}{\sigma_\eta^2} \frac{1}{2\pi} \int_{-\pi}^{\pi} F(e^{j\omega})F^*(e^{j\omega}) d\omega$$

or

$$SNR_{input} = \frac{\sigma_I^2}{\sigma_\eta^2} \frac{1}{2\pi} \int_{-\pi}^{\pi} F(e^{-j\omega})F^*(e^{-j\omega}) d\omega. \quad (3.48)$$

One reason that the SNR at the input to a system is not a perfect measure of signal quality is that not all of the signal power is useful. Often some of the signal power is harmful in that has the same effect as noise. The harmful portion of the signal power is referred to as Inter Symbol Interference (ISI).

Modulation Error Ratio (MER): A figure of merit often used to measure the quality of an equalizer is the Modulation Error Ratio (MER). It is defined as the ratio of signal power to noise-plus-interference power. It can be calculated statistically using ensemble averages (i.e. using expectations) or it can be calculated using time averages.

To calculate it statistically requires viewing the symbol sequence as stochastic process [20]. With this view $\mathbf{I}[k]$ is the stochastic process from which a particular symbol sequence is drawn. The stochastic process $\mathbf{I}[k]$ is the set of all possible sample functions which are denoted $I_1[k], I_2[k], \dots$. A small segment of $\mathbf{I}[k]$ is shown in Figure 3.6 and 3.7. The real

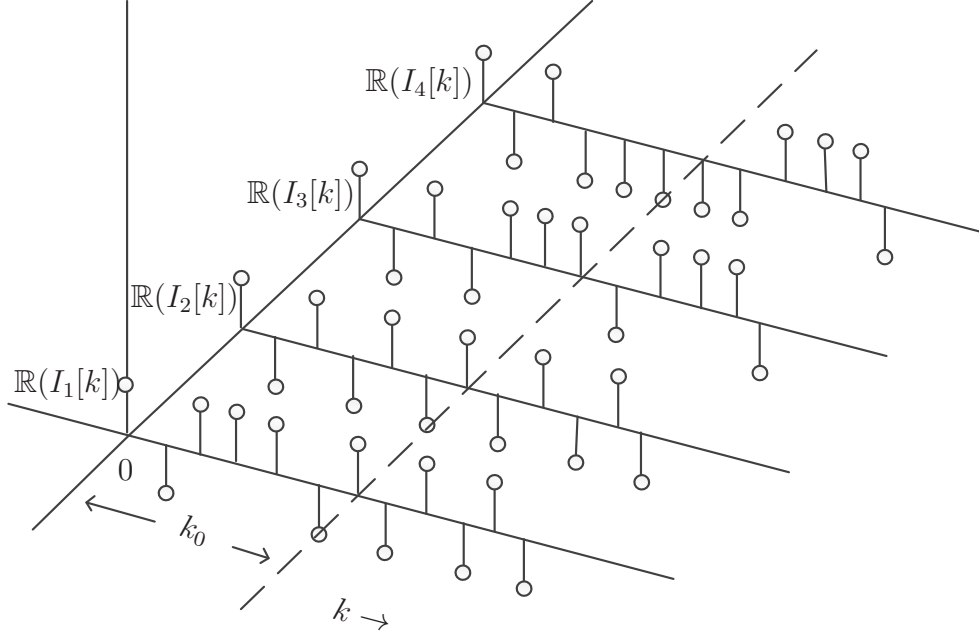


Figure 3.6 Real part of four sample functions in stochastic process $\mathbf{I}[k]$

part is shown in Figure 3.6 and the imaginary part is shown in Figure 3.7. The symbol $I[k]$ is used to denote a generic sample function in $\mathbf{I}[k]$.

In statistical terms MER is the ratio of two expectations:

$$MER = \frac{E \left[\tilde{\mathbf{I}}[k] \tilde{\mathbf{I}}^*[k] \right]}{E \left[(\hat{\mathbf{I}}[k] - \tilde{\mathbf{I}}[k]) (\hat{\mathbf{I}}[k] - \tilde{\mathbf{I}}[k]) \right]} \quad (3.49)$$

where $\hat{\mathbf{I}}[k]$ is the stochastic process representing the output of the equalizer and $\tilde{\mathbf{I}}[k]$ is the stochastic process representing the signal component of $\hat{\mathbf{I}}[k]$. The expectation is the ensemble average given by

$$E \left[\tilde{\mathbf{I}}[k] \tilde{\mathbf{I}}^*[k] \right] = \sum_{i=0}^{\infty} p_i I_i[k] I_i^*[k],$$

where p_i is the probability that sample function $I_i[k]$ will occur.

At first glance it appears that the MER is a function of k . However, the stochastic processes $\hat{\mathbf{I}}[k]$ and $\tilde{\mathbf{I}}[k]$ are stationary. In such cases the ensemble average for each and every

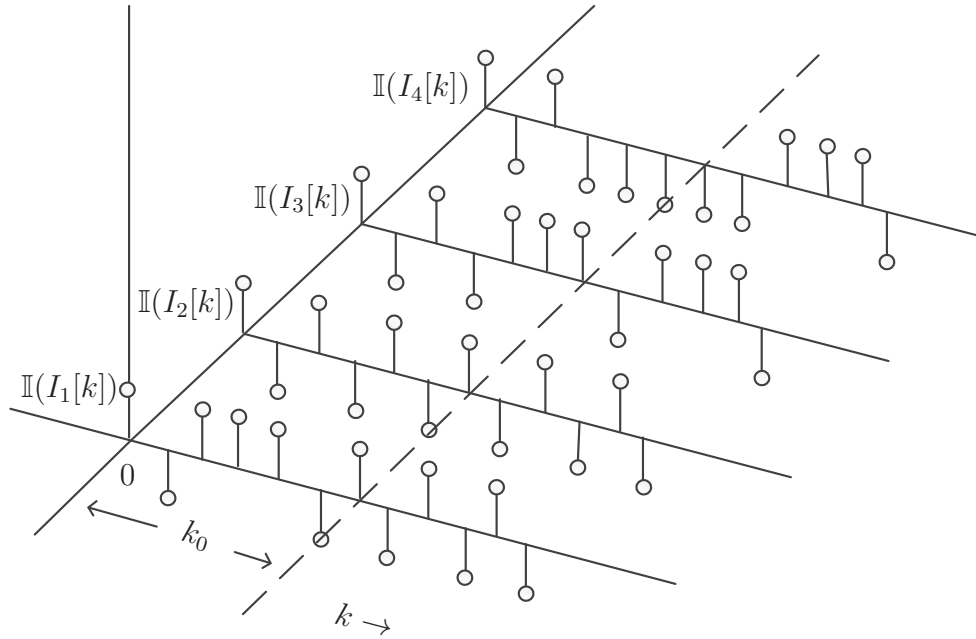


Figure 3.7 Imaginary part of four sample functions in stochastic process $\mathbf{I}[k]$

k is the same and the expectations can be taken for any value of k with the same result.

The signal component $\tilde{\mathbf{I}}[k]$ can be obtained by applying a noise free special input, given by

$$\mathbf{I}_{special}[k] = \begin{cases} 0; & k \neq k_0 \\ \mathbf{I}[k_0]; & k = k_0, \end{cases}$$

to the equalizer and observing the output at time $k = k_0$. Whatever sample function is chosen from $\mathbf{I}[k]$, only symbol $I[k_0]$ is sent through the system. Since all other symbols are zeroed out, i.e. $I[k] = 0$ for $k \neq k_0$, there can be no ISI on output $\hat{I}[k_0]$.

Of course the symbol at $k = k_0$, i.e. $I[k_0]$, will introduce ISI on other symbols, but those symbols do not enter into the analysis. The special input and associated equalizer output is illustrated in Figure 3.8 and 3.9. If the sequence $I[k]$ is applied to the input then $\hat{I}[k]$ will have an ISI component. This ISI component at $k = k_0$ will depend on symbols $I[k - k_0]$ for

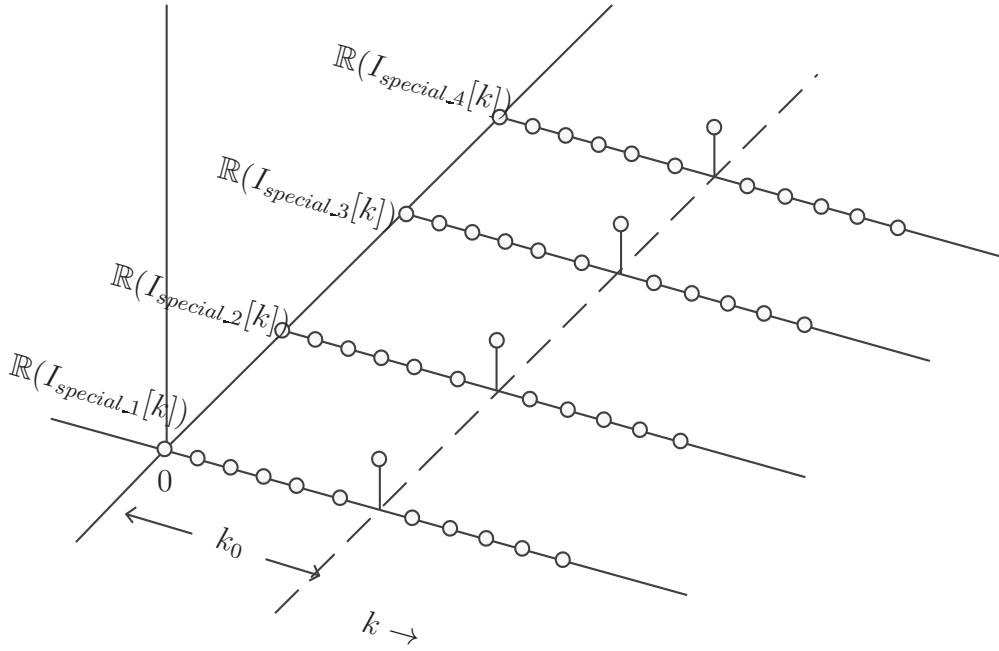


Figure 3.8 Real part of four sample functions in stochastic process $\mathbf{I}_{special}[k]$

$k \neq k_0$. Since symbols are random, the ISI is as well. The output $\hat{I}[k_0]$ has three components

$$\hat{I}[k_0] = \tilde{I}[k_0] + ISI + AWGN$$

where the *ISI* and *AWGN* both are random.

If $I[k_0]$ is known, then $\hat{I}[k_0]$ will contain a component due to $I[k_0]$ plus some random *ISI* and *AWGN*. This means the value for $\tilde{I}[k_0]$ can be obtained when the input sequence is random by taking the conditional expectation

$$\tilde{I}[k] = E \left[\hat{\mathbf{I}}[k] \middle| \mathbf{I}[k_0] = I[k_0] \right]$$

The given condition indicates that the k_0^{th} symbol transmitted is known, but the others are not.

One might think that $E \left[\hat{\mathbf{I}}[k] \middle| \mathbf{I}[k_0] = I[k_0] \right] = I[k_0]$, but that is not true for minimum mean squared equalizers.

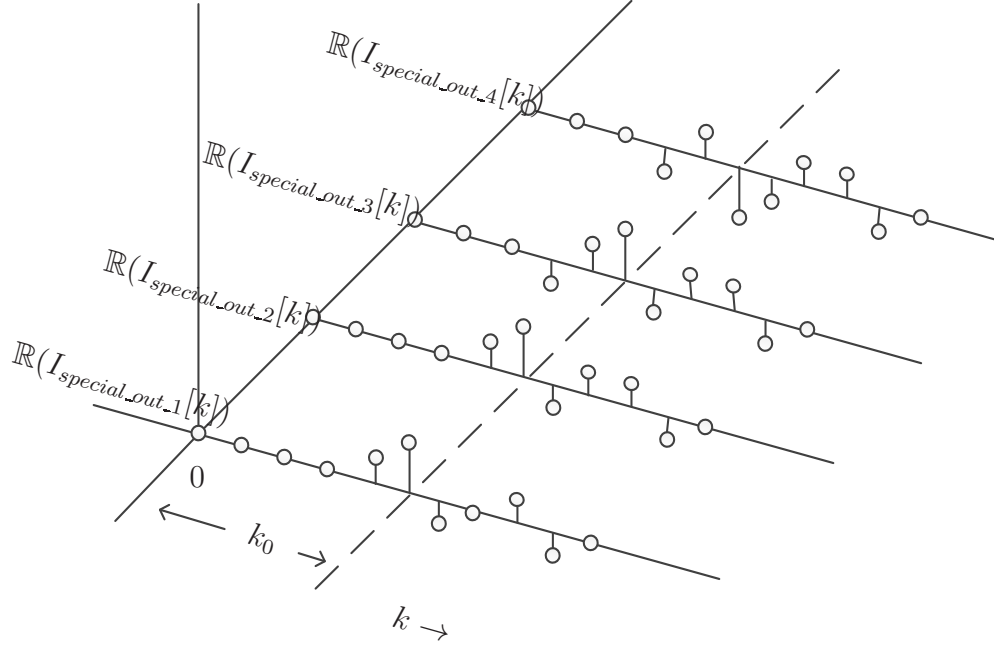


Figure 3.9 Real part of equalizer output for input $\mathbf{I}_{special}[k]$ given in Figure 3.8.

The MER is the ratio of the expectations of the magnitudes of two vectors. These vectors are illustrated in Figure 3.10. Vector $\tilde{I}[k]$ is used in the numerator and error vector $\hat{I}[k] - \tilde{I}[k]$ is used in the denominator.

The MER given by (3.49) can also be calculated from (instead of ensemble averages) on sample function $\tilde{I}[k]$ and $\hat{I}[k]$. This is not the usual method of analysis, but expressing MER in this way helps in the understanding of ensemble average analysis. The MER expressed as a time average is

$$MER = \frac{\overline{|\tilde{I}[k]|^2}}{\overline{|\hat{I}[k] - \tilde{I}[k]|^2}}, \quad (3.50)$$

where the overline indicates a time average.

The MER is sometimes referred to as the SNR of the output. To view the SNR of the output as the MER, the definition of SNR at the output has to be different than the SNR

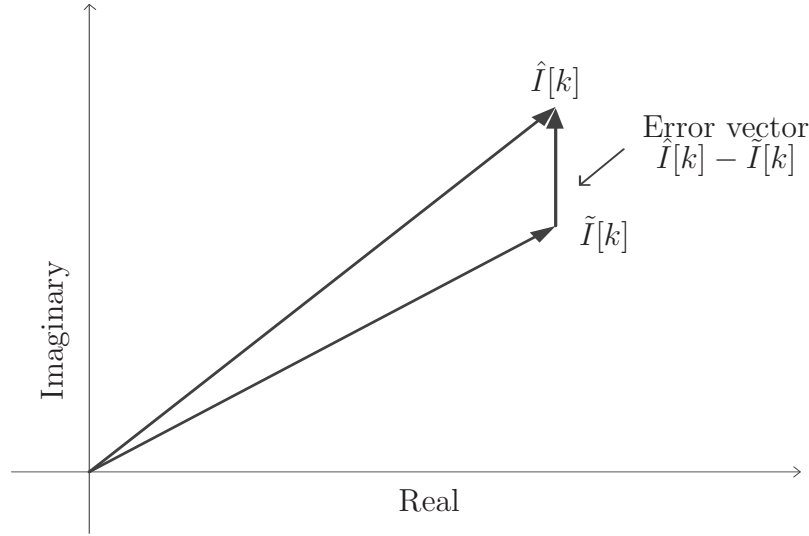


Figure 3.10 Vectors used in MER

defined for the input.

$$SNR_{input} = \frac{\text{Total power at output of } f[n]}{\text{Power in AWGN}}.$$

$$MER = \frac{\text{Useful power in decision variable}}{\text{Harmful power corrupting the decision variable}}.$$

That is to say MER is the power in signal at output of $f[n] \star c[n]$ that has not been dispersed (i.e. the power that remains in the symbol time in which it was transmitted) divided by power in signal at output of $f[n] \star c[n]$ that has been dispersed plus the power in the *AWGN*.

The useful power in the decision variable is $E[|\tilde{\mathbf{I}}[k_0]|^2]$. Clearly

$$E[|\tilde{\mathbf{I}}[k_0]|^2] \leq \text{Total signal power}.$$

$\tilde{I}[k_0]$ is the output of the equalizer at symbol time k_0 for an input that is an impulse applied at $k = k_0$, i.e. input is equal to $I[k]\delta[k - k_0]$. Since the system $f[k] \star c[k]$ is time invariant the output at time $k = k_0$ is the same as the output at time 0 for an impulse

applied at time 0. Therefore $\tilde{I}[k_0]$ is the output at $k = 0$ for input $I[k]\delta[k]$ which is given by

$$\begin{aligned}\tilde{I}[k_0] &= I[k_0] \left(c[n] \star f[n] \Big|_{n=0} \right) \\ &= I[k_0] \sum_{m=-\infty}^{-\infty} c[m] f[n-m] \Big|_{n=0} \\ &= I[k_0] \sum_{m=-\infty}^{-\infty} c[m] f[-m].\end{aligned}\tag{3.51}$$

The sum $\sum_{m=-\infty}^{-\infty} c[m] f[-m]$ can be expressed in terms of J_{min} . This is done by substituting equation (3.42) into equation (3.43) and then solving for $\sum_{m=-\infty}^{-\infty} c[m] f[-m]$. Doing this yields

$$\sum_{m=-\infty}^{-\infty} c[m] f[-m] = 1 - \frac{J_{min}}{\sigma_I^2}.\tag{3.52}$$

Substituting equation (3.52) into equation (3.51) has

$$\tilde{I}[k] = \left(1 - \frac{J_{min}}{\sigma_I^2} \right) I[k].\tag{3.53}$$

and it follows

$$E \left[|\tilde{I}[k]|^2 \right] = \left(1 - \frac{J_{min}}{\sigma_I^2} \right)^2 \sigma_I^2.\tag{3.54}$$

The error on the decision variable is $\hat{I}[k] - \tilde{I}[k]$, which is $\hat{I}[k] - \left(1 - \frac{J_{min}}{\sigma_I^2} \right) I[k]$. The expected value of the square of the error on the decision variable is

$$\begin{aligned}E \left[\left| \hat{I}[k] - \tilde{I}[k] \right|^2 \right] &= E \left[\left| \hat{I}[k] - \left(1 - \frac{J_{min}}{\sigma_I^2} \right) I[k] \right|^2 \right] \\ &= E \left[\left| \hat{I}[k] - I[k] + \frac{J_{min}}{\sigma_I^2} I[k] \right|^2 \right] \\ &= E \left[\left| \left(\hat{I}[k] - I[k] \right) + \frac{J_{min}}{\sigma_I^2} I[k] \right|^2 \right] \\ &= E \left[\left(\hat{I}[k] - I[k] \right) \left(\hat{I}[k] - I[k] \right)^* \right] \\ &\quad + E \left[\frac{J_{min}}{\sigma_I^2} I[k] \left(\hat{I}[k] - I[k] \right)^* \right] \\ &\quad + E \left[\left(\hat{I}[k] - I[k] \right) \frac{J_{min}}{\sigma_I^2} I^*[k] \right] \\ &\quad + E \left[\frac{J_{min}}{\sigma_I^2} I[k] \frac{J_{min}}{\sigma_I^2} I^*[k] \right]\end{aligned}$$

Since J_{min} is real so $\frac{J_{min}^*}{\sigma_I^2} = \frac{J_{min}}{\sigma_I^2}$. Therefore

$$E \left[\left| \hat{I}[k] - \tilde{I}[k] \right|^2 \right] = J_{min} + \frac{J_{min}}{\sigma_I^2}(-J_{min}) + \frac{J_{min}}{\sigma_I^2}(-J_{min}) + \frac{J_{min}^2}{\sigma_I^2}. \quad (3.55)$$

Collecting the terms in equation (3.55) simplifies the expression to

$$E \left[\left| \hat{I}[k] - \tilde{I}[k] \right|^2 \right] = J_{min} \left(1 - \frac{J_{min}}{\sigma_I^2} \right). \quad (3.56)$$

Substituting equation (3.54) and equation (3.56) into equation (3.49) yields

$$MER = \frac{\sigma_I^2 - J_{min}}{J_{min}}. \quad (3.57)$$

It is important to note that the MER is maximum when J_{min} is minimum. Therefore the coefficients $c[n]$ that minimize $E \left[\left| \hat{I}[k] - \tilde{I}[k] \right|^2 \right]$, i.e. optimum MSE coefficients, are the ones that maximize the MER.

4. DOCSIS Equalizer

Equalizer are used in digital QAM communication receivers to mitigate the effects of distortion introduced by the channel. Equalizer are not the only circuits in the receiver that mitigate effects of distortion. Timing and frequency correction circuits, which are located upstream of the equalizer, correct most of the error in sampling time and in frequency offset. The equalizer corrects all of the remaining distortion, which includes residual timing and frequency offset.

There are many equalization techniques available in literature. These techniques can be broken down to two main categories: static equalizers and adaptive equalizers. In static equalization, the equalizer coefficients are not updated, whereas in adaptive equalization, the equalizer coefficients are continually updated to track channel changes.

In a DOCSIS system, the impulse response of the upstream channel is unknown due to micro-reflections with unknown delay and amplitude. The incoming signal carries information about the channel that can be processed by the equalizer to get the information needed to correct the distortion. A micro-reflection can appear and disappear over time, causing the channel impulse response to vary with time. For such channels an adaptive equalization technique is necessary [22], [23], [24], [25].

The adaptive equalizer in the DOCSIS upstream demodulator is a filter with complex coefficients, which is usually linked to output of the matched filter by a down-sampler. The down-sampler reduces the sampling rate to be commensurate with the symbol rate. Such equalizers are said to be symbol-spaced [26]. Equalizers that operate at a multiple of the symbol rate are said to be fractionally-spaced [26]. The adaptive equalizer that is used in the DOCSIS upstream demodulator is a symbol-spaced equalizer.

Symbol-spaced equalizers have an economic advantage over fractionally-spaced equalizers, but are sensitive to the sampling times, i.e. the phase of the down-sampler. The economic advantage comes from a shorter filter and a less complex coefficient-update algorithm. The disadvantage is that a timing recovery circuit is required to resample in order to adjust the phase for the down-sampler.

The remainder of this chapter discusses the algorithm that updates the coefficients, the adaptive structure and the criterion used to measure the performance of the equalizer.

4.1 Coefficient Update Algorithm

The equalizer coefficients are updated by an adaptive algorithm. The most commonly encountered adaptive algorithms in the literature are the Least Mean Square (LMS) and the Recursive Least Square (RLS) algorithms [15], [17], [18].

There are advantages and disadvantages of each. The best choice depends on the application. For example, if an application requires the equalizer to converge quickly, then RLS algorithm is preferred, but it is much more costly [27], [28], [29]. On the other hand, if an application requires a low cost equalizer, then the LMS algorithm is preferred. The LMS algorithm is much less costly, but it has a larger convergence time. The larger convergence time means a longer training sequence, which, in effect, reduces the length of the payload data and decreases the bandwidth efficiency of the system.

This thesis pursues a low cost equalizer based on the LMS update algorithm. However, emphasis is placed on maximizing performance within the constraint of using the LMS update algorithm.

4.2 LMS Algorithm

The LMS algorithm continually adjusts the coefficients of the equalizer to seek the minimum mean squared error in the decision variable. The majority of adjustment is made during the preamble where the values for the decision variable are known in advance.

The analysis that follows concerns the sequence of adjustment made to the coefficients.

Specifically the analysis determines the adjustment to the coefficients from those used to compute $\hat{I}[n]$ to those that will be used to compute $\hat{I}[n+1]$, where n could be any integer. The computation for $\hat{I}[n]$ has samples $v[n + \frac{M}{2} - 1], \dots, v[n+1], v[n], v[n-1], \dots, v[n - \frac{M}{2}]$ in the equalizer whereas the computation for $\hat{I}[n+1]$ has samples $v[n + \frac{M}{2}], \dots, v[n+2], v[n+1], v[n], \dots, v[n - \frac{M}{2} + 1]$ in the equalizer. For notational convenience the M samples in the equalizer used to calculate $\hat{I}[n]$ and $\hat{I}[n+1]$ are expressed as the length M row vectors.

$$\vec{v}[n] = \left[v[n + \frac{M}{2} - 1], \dots, v[n+1], v[n], v[n-1], \dots, v[n - \frac{M}{2}] \right]$$

and

$$\vec{v}[n+1] = \left[v[n + \frac{M}{2}], \dots, v[n+2], v[n+1], v[n], \dots, v[n - \frac{M}{2} + 1] \right].$$

Vector $\vec{v}[n]$ is referred to as the input vector for output $\hat{I}[n]$.

The set of complex equalizer coefficients used on $\vec{v}[n]$ to compute $\hat{I}[n]$ are $c_0[n], c_1[n], \dots, c_{M-1}[n]$ and represented in vector form by the length M row vector:

$$\vec{c}[n] = [c_0[n], c_1[n], \dots, c_{M-1}[n]],$$

The vector $\vec{c}[n]$ is referred to as tap weight vector.

To make the mathematics that follows more compact and readable, the set of complex vectors $\vec{c}[n]$ and $\vec{c}[n+1]$ are represented

$$\vec{c}[n] = \vec{\alpha} + j\vec{\beta}, \tag{4.1}$$

$$\begin{aligned} \vec{c}[n+1] &= \vec{\alpha} + \Delta\vec{\alpha} + j(\vec{\beta} + \Delta\vec{\beta}) \\ &= \vec{\alpha} + j\vec{\beta} + \Delta\vec{\alpha} + j\Delta\vec{\beta} \end{aligned} \tag{4.2}$$

where $\vec{\alpha} = [\alpha_0, \alpha_1, \dots, \alpha_{M-1}]$ and $\vec{\beta} = [\beta_0, \beta_1, \dots, \beta_{M-1}]$ are the real and imaginary part of $\vec{c}[n]$ and $\Delta\vec{\alpha} = [\Delta\alpha_0, \Delta\alpha_1, \dots, \Delta\alpha_{M-1}]$ and $\Delta\vec{\beta} = [\Delta\beta_0, \Delta\beta_1, \dots, \Delta\beta_{M-1}]$ are incremental changes made to $\vec{c}[n]$ to obtain $\vec{c}[n+1]$.

The equalizer output, $\hat{I}[n]$, which is the vector product of $\vec{v}[n]$ and $\vec{c}[n]$, is given by

$$\hat{I}[n] = \vec{v}[n] \vec{c}^T[n]. \tag{4.3}$$

The analysis begins with the equation for the mean of the square of the magnitude of the error (MSE) on the decision variable. The mean of the square of the magnitude of the error at sample n is given by (3.3) (page 29) and is repeated here, which is shown here as a function of n :

$$J[n] = E [|\varepsilon[n]|^2] = E [\varepsilon[n]\varepsilon^*[n]], \quad (4.4)$$

where $J[n]$ is the mean squared error and $\varepsilon[n]$ is the error in the decision variable (output of the equalizer) given by

$$\varepsilon[n] = \hat{I}[n] - I[n], \quad (4.5)$$

where $I[n]$ is the n^{th} symbol in the training sequence, which is known in advance.

Of course the conjugate of the error is

$$\varepsilon^*[n] = \hat{I}^*[n] - I^*[n]. \quad (4.6)$$

Substituting (4.3) into (4.5) yields

$$\varepsilon[n] = \vec{v}[n]\vec{c}^T[n] - I[n]. \quad (4.7)$$

Substituting (4.1) into (4.7) has

$$\begin{aligned} \varepsilon[n] &= \vec{v}[n]\vec{\alpha}^T + j\vec{v}[n]\vec{\beta}^T - I[n] \\ \varepsilon[n] &= \sum_{i=1}^M v_i[n]\alpha_i + j \sum_{i=1}^M v_i[n]\beta_i - I[n]. \end{aligned} \quad (4.8)$$

Similarly

$$\begin{aligned} \varepsilon^*[n] &= \left(\vec{v}[n]\vec{c}^T[n]\right)^* - I^*[n] \\ \varepsilon^*[n] &= \vec{v}^*[n] \left(\vec{\alpha}^T + j\vec{\beta}^T\right)^* - I^*[n] \\ \varepsilon^*[n] &= \vec{v}^*[n]\vec{\alpha}^T - j\vec{v}^*[n]\vec{\beta}^T - I^*[n] \\ \varepsilon^*[n] &= \sum_{i=1}^M v_i^*[n]\alpha_i - j \sum_{i=1}^M v_i^*[n]\beta_i - I^*[n], \end{aligned} \quad (4.9)$$

where M is the number of elements in each vector.

$J[n]$ is the cost function to be minimized. The objective of the algorithm is to adjust the coefficients to decrease $J[n]$ with increasing n , i.e. to make changes to the coefficients so that $\vec{c}[n+1]$ makes $J[n+1]$ less than or equal to $J[n]$.

Each coefficient in $\vec{c}[n]$ is adjusted independently to make $J[n+1] \leq J[n]$. More precisely, the real and imaginary parts of each coefficient are adjusted independently to make $J[n+1]$ smaller than $J[n]$.

To that end the adjustment for the real part of coefficient $c_i[n]$ to get coefficient $c_i[n+1]$ is considered. The first step in the adjustment is to find the derivative of the cost function $J[n]$ with respect to α_i . Taking the derivative of (4.4) yields

$$\begin{aligned}\frac{\partial}{\partial \alpha_i} J[n] &= \frac{\partial}{\partial \alpha_i} E[\varepsilon[n] \varepsilon^*[n]] = E \left[\frac{\partial}{\partial \alpha_i} [\varepsilon[n] \varepsilon^*[n]] \right] \\ &= E \left[\varepsilon^*[n] \frac{\partial}{\partial \alpha_i} \varepsilon[n] + \varepsilon[n] \frac{\partial}{\partial \alpha_i} \varepsilon^*[n] \right].\end{aligned}\quad (4.10)$$

Since the partial derivatives of (4.8) with respect to α_i has

$$\begin{aligned}\frac{\partial}{\partial \alpha_i} \varepsilon[n] &= \frac{\partial}{\partial \alpha_i} \left(\sum_{i=1}^M v_i[n] \alpha_i + j \sum_{i=1}^M v_i[n] \beta_i - I[n] \right) \\ &= \frac{\partial}{\partial \alpha_i} \sum_{i=1}^M v_i[n] \alpha_i + j \frac{\partial}{\partial \alpha_i} \sum_{i=1}^M v_i[n] \beta_i - \frac{\partial}{\partial \alpha_i} I[n] \\ &= v_i[n]\end{aligned}\quad (4.11)$$

and the partial derivative of (4.9) with respect to α_i has

$$\begin{aligned}\frac{\partial}{\partial \alpha_i} \varepsilon^*[n] &= \frac{\partial}{\partial \alpha_i} \left(\sum_{i=1}^M v_i^*[n] \alpha_i - j \sum_{i=1}^M v_i^*[n] \beta_i - I^*[n] \right) \\ &= \frac{\partial}{\partial \alpha_i} \sum_{i=1}^M v_i^*[n] \alpha_i - j \frac{\partial}{\partial \alpha_i} \sum_{i=1}^M v_i^*[n] \beta_i - \frac{\partial}{\partial \alpha_i} I^*[n] \\ &= v_i^*[n],\end{aligned}\quad (4.12)$$

$\frac{\partial}{\partial \alpha_i} J[n]$ becomes

$$\begin{aligned}\frac{\partial}{\partial \alpha_i} J[n] &= E \left[\varepsilon^*[n] v_i[n] + \varepsilon[n] v_i^*[n] \right] \\ &= E \left[2\Re \left(\varepsilon[n] v_i^*[n] \right) \right].\end{aligned}\quad (4.13)$$

For $J[n+1]$ to be less than $J[n]$ the increment $\Delta\alpha_i$ must have sign opposite of that of $\frac{\partial}{\partial\alpha_i}J[n]$. As with steepest descent method for the algorithm to be stable the increment must be sufficiently small. To that end the slope is scaled by a positive constant denoted, $\mu/2$, where μ is referred to as step size. Doing this has

$$\begin{aligned}\Delta\alpha_i &= E \left[-(\text{slope}) \left(\frac{\text{step size}}{2} \right) \right] \\ &= E \left[-2\Re(\varepsilon[n]v_i^*[n]) \frac{\mu}{2} \right] \\ \Delta\alpha_i &= E [-\mu\Re(\varepsilon[n]v_i^*[n])] \end{aligned} \tag{4.14}$$

The adjustment for the imaginary part of coefficient $c_i[n]$ to get coefficient $c_i[n+1]$ is calculated in a similar manner. However, there is a difference as conjugation does not effect the real part of $\varepsilon[n]$, but does effect the sign of the imaginary part. Adjustment $\Delta\beta_i$ is found by differentiating the cost function $J[n]$ with respect to β_i and then multiplying it by $\mu/2$.

Taking the derivative of (4.4) yield

$$\begin{aligned}\frac{\partial}{\partial\beta_i}J[n] &= \frac{\partial}{\partial\beta_i}E[\varepsilon[n]\varepsilon^*[n]] = E \left[\frac{\partial}{\partial\beta_i}[\varepsilon[n]\varepsilon^*[n]] \right] \\ &= E \left[\varepsilon^*[n] \frac{\partial}{\partial\beta_i}\varepsilon[n] + \varepsilon[n] \frac{\partial}{\partial\beta_i}\varepsilon^*[n] \right]. \end{aligned} \tag{4.15}$$

Taking the partial derivative of (4.8) with respect to β_i yields

$$\begin{aligned}\frac{\partial}{\partial\beta_i}\varepsilon[n] &= \frac{\partial}{\partial\beta_i} \left(\sum_{i=1}^M v_i[n]\alpha_i + j \sum_{i=1}^M v_i[n]\beta_i - I[n] \right) \\ &= \frac{\partial}{\partial\beta_i} \sum_{i=1}^M v_i[n]\alpha_i + j \frac{\partial}{\partial\beta_i} \sum_{i=1}^M v_i[n]\beta_i - \frac{\partial}{\partial\beta_i}I[n] \\ &= jv_i[n] \end{aligned} \tag{4.16}$$

and taking the partial derivative of (4.9) with respect to β_i yields

$$\begin{aligned}\frac{\partial}{\partial\beta_i}\varepsilon^*[n] &= \frac{\partial}{\partial\beta_i} \left(\sum_{i=1}^M v_i^*[n]\alpha_i - j \sum_{i=1}^M v_i^*[n]\beta_i - I^*[n] \right) \\ &= \frac{\partial}{\partial\beta_i} \sum_{i=1}^M v_i^*[n]\alpha_i - j \frac{\partial}{\partial\beta_i} \sum_{i=1}^M v_i^*[n]\beta_i - \frac{\partial}{\partial\beta_i}I^*[n] \\ &= -jv_i^*[n]. \end{aligned} \tag{4.17}$$

Substituting (4.16) and (4.17) into (4.15) yields

$$\begin{aligned}
\frac{\partial}{\partial \beta_i} J[n] &= E \left[j \varepsilon^*[n] v_i[n] - j \varepsilon[n] v_i^*[n] \right] \\
&= E \left[-j \left(\varepsilon[n] v_i^*[n] - \varepsilon^*[n] v_i[n] \right) \right] \\
&= E \left[-j 2 \Im \left(\varepsilon[n] v_i^*[n] \right) \right] \\
&= E \left[2 \Im \left(\varepsilon[n] v_i^*[n] \right) \right].
\end{aligned} \tag{4.18}$$

The adjustment, $\Delta \beta_i$ is given by

$$\begin{aligned}
\Delta \beta_i &= E \left[-(\text{slope}) \left(\frac{\text{step size}}{2} \right) \right] \\
\Delta \beta_i &= E \left[-2 \Im \left(\varepsilon[n] v_i^*[n] \right) \frac{\mu}{2} \right] \\
\Delta \beta_i &= E \left[-\mu \Im \left(\varepsilon[n] v_i^*[n] \right) \right].
\end{aligned} \tag{4.19}$$

Therefore, the adjustment made to coefficient $c_i[n]$ to get coefficient $c_i[n+1]$ is given by

$$\begin{aligned}
\Delta \alpha_i + j \Delta \beta_i &= E \left[-\mu \Re \left(\varepsilon[n] v_i^*[n] \right) \right] + j E \left[-\mu \Im \left(\varepsilon[n] v_i^*[n] \right) \right] \\
&= -\mu E \left[\varepsilon[n] v_i^*[n] \right]
\end{aligned} \tag{4.20}$$

The above analysis is valid for all coefficients $c_i[n]$, $i = 0, 1, \dots, M-1$, in $\vec{c}[n]$. Therefore, the adjustment in vector form is given by

$$\Delta \vec{\alpha} + j \Delta \vec{\beta} = -\mu E \left[\varepsilon[n] \vec{v}^*[n] \right]. \tag{4.21}$$

The expectation in (4.21) is a problem. Classically the expectation of $\varepsilon[n] v_i^*[n]$ is computed by

$$E \left[\varepsilon[n] v_i^*[n] \right] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \varepsilon[n] v_i^*[n] f_{\alpha_i, \beta_i, \varepsilon}(\alpha_i, \beta_i, \varepsilon[n]) d\alpha_i d\beta_i d\varepsilon[n], \tag{4.22}$$

where $f_{\alpha_i, \beta_i, \varepsilon}(\alpha_i, \beta_i, \varepsilon[n])$ is the joint density function for variables α_i, β_i and $\varepsilon[n]$. Unfortunately this joint density function is unknown so the expectation can not be computed this way.

The expectation can also be computed with the time average. In this scenario the coefficients are not updated upon the arrival of each new symbol, but remain constant at $\vec{c}[n]$ while the time average is calculated. The time average is given by

$$E[\varepsilon[n]v_i^*[n]] = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=0}^{N-1} \varepsilon[n+k]v_i^*[n+k]. \quad (4.23)$$

While computing this time average, $\hat{I}[n]$ is estimated N times by $\vec{v}[n+k]\vec{c}^T[n]$, for $k = 0, 1, \dots, N-1$.

The expectation can be approximated with a finite time average of length N . Using the finite length time average to approximate the expectation has

$$\Delta\vec{\alpha} + j\Delta\vec{\beta} \simeq -\mu \frac{1}{N} \sum_{k=0}^{N-1} \varepsilon[n+k]\vec{v}^*[n+k]. \quad (4.24)$$

Holding $\vec{c}[n]$ constant for the duration of the calculation means, $\Delta\vec{\alpha} + j\Delta\vec{\beta}$ is applied to $\vec{\alpha} + j\vec{\beta}$ at symbol $n+N$ and become the adjustment from $\vec{c}[n]$ to $\vec{c}[n+N]$.

The implementation cost can be reduced under the assumption $\Delta\vec{\alpha}$ and $\Delta\vec{\beta}$ are small by updating the coefficients after every symbol so that

$$\vec{c}[n+1] - \vec{c}[n] = \Delta\vec{\alpha} + j\Delta\vec{\beta} = -\frac{\mu}{N} \varepsilon[n]\vec{v}^*[n], \quad (4.25)$$

where the step size is now μ/N . Using this “update every new symbol” algorithm has the difference in the coefficients after N symbols as being

$$\vec{c}[n+N-1] - \vec{c}[n] = \sum_{k=0}^{N-1} -\frac{\mu}{N} \varepsilon[n+k]\vec{v}^*[n+k]. \quad (4.26)$$

If the difference $\vec{c}[n+N-1] - \vec{c}[n]$ is small then $E[\varepsilon[n+k]\vec{v}^*[n+k]] \simeq E[\varepsilon[n]\vec{v}^*[n]]$ and the difference $\vec{c}[n+N-1] - \vec{c}[n]$ will be essentially the same as if it was calculated by the time average. This justifies the use of the “update every new symbol” algorithm with the sequence of coefficient vectors given by

$$\vec{c}[n+1] = \vec{c}[n] - \mu\varepsilon[n]\vec{v}^*[n]. \quad (4.27)$$

Equation (4.27) is expressed in matrix form by

$$\begin{bmatrix} c_0[n+1] \\ \cdot \\ \cdot \\ c_i[n+1] \\ \cdot \\ \cdot \\ c_{M-1}[n+1] \end{bmatrix} = \begin{bmatrix} c_0[n] \\ \cdot \\ \cdot \\ c_i[n] \\ \cdot \\ \cdot \\ c_{M-1}[n] \end{bmatrix} - \begin{bmatrix} v^*[n + \frac{M}{2} - 1] \\ \cdot \\ \cdot \\ v^*[n] \\ \cdot \\ \cdot \\ v^*[n - \frac{M}{2}] \end{bmatrix} \varepsilon[n]\mu.$$

4.3 Structure of the DOCSIS Equalizer in Upstream Demodulator

The structure of a symbol-spaced adaptive equalizer is comprised of a complex coefficient linear transversal filter. According to the DOCSIS standard the adaptive equalizer must run at the symbol rate and can have up to 24 complex coefficients. Hence the equalizer is a symbol-spaced equalizer of length up to 24 taps. The structure of this symbol-spaced complex adaptive equalizer is illustrated in Figure 4.1.

The number of delay elements in transversal filter in Figure 4.1 is denoted M . For a DOCSIS equalizer M can be up to 24. The top part in the figure is a transversal filter. The taps of the filter are denoted $v[n + \frac{M}{2} - 1], \dots, v[n + 1], v[n], v[n - 1], \dots, v[n - \frac{M}{2}]$, which are the elements of the tap input vector $\vec{v}[n]$ defined earlier. The corresponding tap weights are denoted by $c_0[n], c_1[n], \dots, c_{M-1}[n]$, which are the elements of the tap weight vector $\vec{c}[n]$ defined earlier. The output of the equalizer, $\hat{I}[n]$, is vector product $\vec{v}[n] \cdot \vec{c}^T[n]$. $\hat{I}[n]$ goes to the slicer as well as to the summer that calculates the error.

The error is calculated by taking the difference between the output of the equalizer, $\hat{I}[n]$, and either the output of the slicer denoted $\hat{I}_{sli}[n]$ or training data, $I[n]$. Whether $\hat{I}_{sli}[n]$ or $I[n]$ is used to calculate the error depends on the operating mode of the equalizer.

There are two possible operating modes: the training mode and the data mode. Both modes will be explained after a brief description of Figure 4.1.

For the purpose of description, the structure of the complex adaptive equalizer is broken

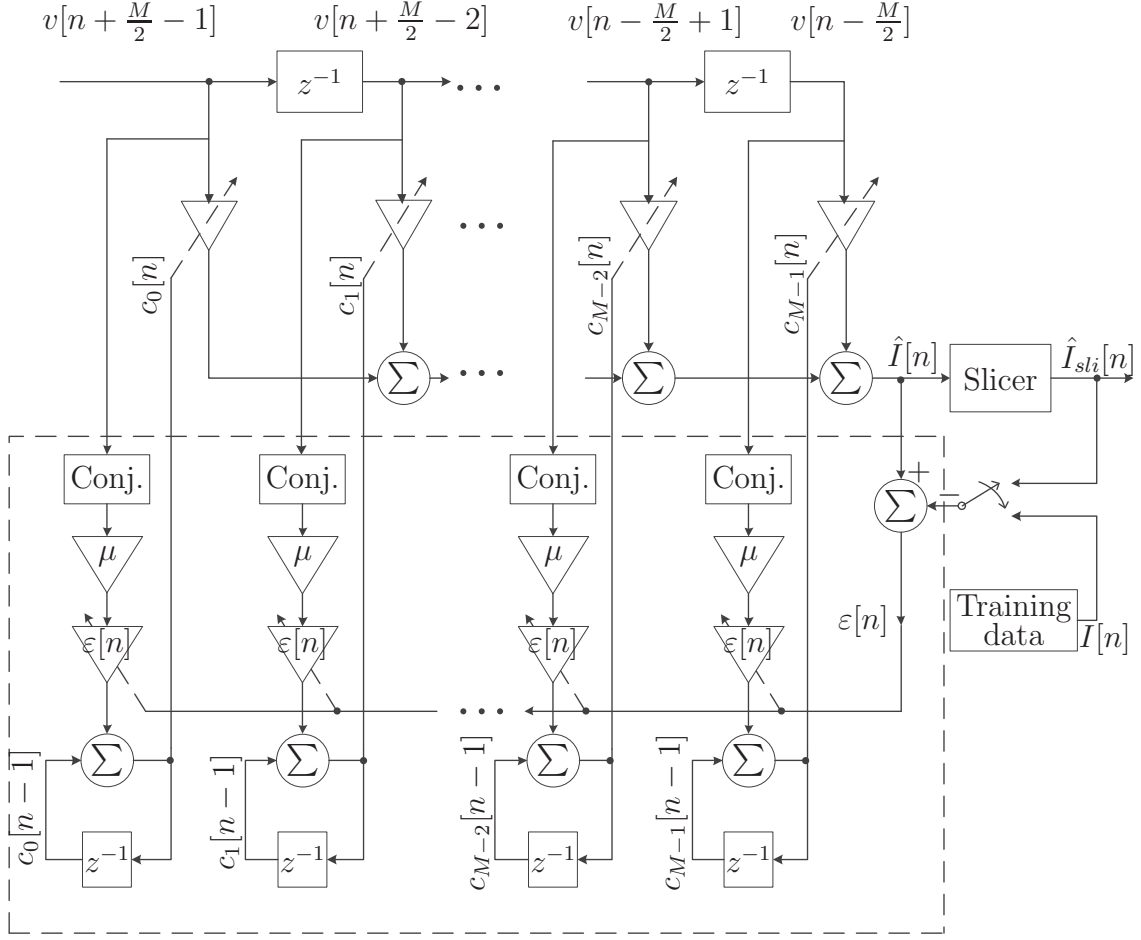


Figure 4.1 Block diagram of 24 tap symbol-spaced complex adaptive equalizer

down into two basic processes: the filtering process and the adaptive process.

The filtering process involves computing the output of a linear transversal filter (i.e. computing $\hat{I}[n]$ as shown in Figure 4.1).

The adaptive process is basically the automatic adjustment of the tap weights of the linear transversal filter. The adaptive process is shown in the enclosed dashed box in Figure 4.1. The coefficients are updated each time a symbol enters the transversal filter. This is done by first conjugating and weighting the data in the equalizer, $\vec{v}[n]$, and then updating the coefficients by adding the weighted data to the coefficients. The weighting is the product of the error, $\epsilon[n]$ and the step size, μ . The error, $\epsilon[n]$, is obtained by subtracting either of

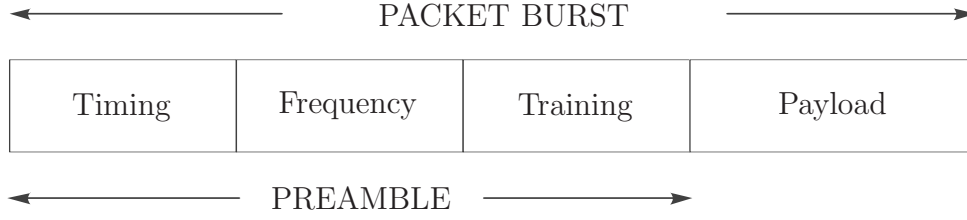


Figure 4.2 A typical DOCSIS packet burst

the training data (training mode) or slicer output (data mode) from the equalizer output. The correction to the coefficients is usually, but not always, in the direction of improvement, however the size of the correction is always proportional to $\varepsilon[n]$.

The iterative process of tap weight adaptation starts with an initial guess of $\vec{c}[0] = \vec{c}_{init}$. The vector, \vec{c}_{init} has just one non-zero element and that element is 1. The non-zero element is located near the center of the vector.

4.3.1 The operating Mode of the Equalizer

The operation of the equalizer depends on its operating mode. During the preamble of the packet the equalizer operates in training mode and switches to data when the payload part of packet is reached. A packet burst is illustrated in Figure 4.2. The preamble, which consists of three fields, precedes the payload field, which is the data part. The three fields in the preamble are, in order of sequence, timing, frequency, and training. The flow order of each field in the packet burst is shown in Figure 4.2 sequentially.

The purpose of the first two fields is to provide information suitable for estimating timing and frequency offset. The next two field in the packet burst are training and payload. The training field is used to obtain the coefficients for the equalizer. The payload field, which contains the data, is also used to update the coefficients to track changes in the channel.

Training Mode

The equalizer is in training mode when the training field of preamble reaches the equalizer. In this mode the error is calculated by subtracting the training data, $I[n]$ from the equalizer output $\hat{I}[n]$. For the equalizer in Figure 4.1 to be in training mode, the switch located between the slicer and training data blocks must select the training data. The training data is programmed to the same as the data transmitted in the training field, which is known.

The equalizer is trained for the duration of the training field and hence the name of this mode is training mode. At the beginning of training field the equalizer starts with an initial set of coefficients. As data from the training field enters the equalizer, the coefficients are updated in a way that on average continually improves the decision variable. By the time the end of the training sequence is reached, the equalizer coefficients have converged. As the coefficients converge the error in the output reduces to a steady state rms level.

Data Mode

By the time the end of the training sequence is reached the equalizer output, $\hat{I}[n]$, is a very good estimate of $I[n]$ and the slicer quantizes $\hat{I}[n]$ to $I[n]$ with a high degree of certainty (with an occasional error). Immediately after the training sequence is exhausted the switch is flipped to select the slicer output. With the flip of this switch, the equalizer enters data mode. With the trained equalizer the slicer produces $I[n]$ most of the time, which keeps the coefficients of the equalizer tracking the changes in the channel.

In data mode, the error is calculated by subtracting the slicer output, $\hat{I}_{sli}[n]$, which is $I[n]$ most of the time from the equalizer output, $\hat{I}[n]$. Prior to entering the data mode, the equalizer should be in a state where probability of error is small (near steady state). Using the slicer output in data mode, keeps equalizer in steady state.

4.4 Performance Criterion

The performance for a QAM receiver can not be reduced to a single metric. Commonly two metrics are used: convergence time and the quality of the decision variable at the output

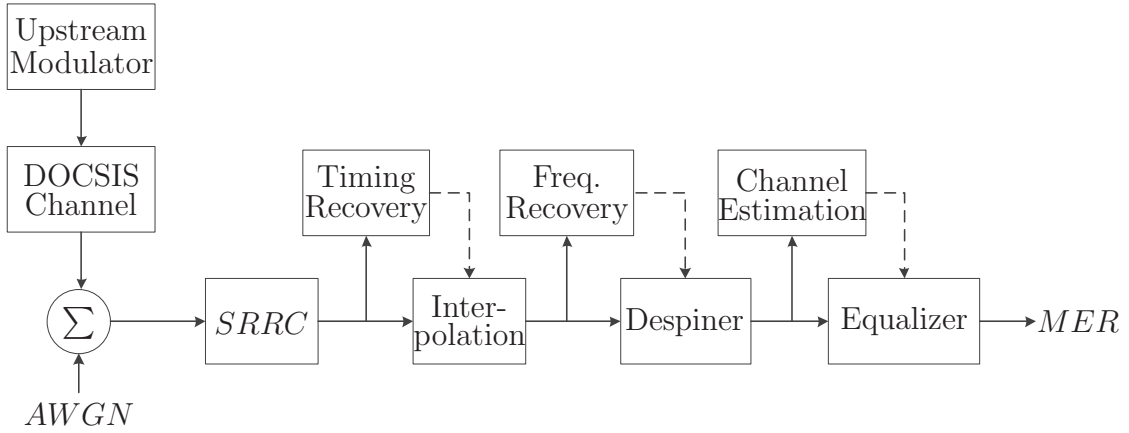


Figure 4.3 Block diagram for MER measurement

of the equalizer. The convergence time is measured in symbols¹. The quality of the decision variable can be measured in a variety of ways. Modulation error ratio (MER) is the measure used in this thesis.

MER Measurement

The MER is one of the most commonly used performance measures in a communication system. It gauges the error on the decision variable. The MER should not be measured until the adaptive algorithm has the coefficient at steady state. Customarily it is used to gauge the steady state error at the output of the equalizer. It can also be used to gauge how close the equalizer is to steady state when the end of the preamble is reached. To do this, the update algorithm is turned off at the end of the preamble and MER is calculated using the data in the payload. A high-level simulation setup for measuring MER is shown in Figure 4.3.

Convergence Time

Another performance criteria for the LMS equalizer is convergence time, which is measured in symbols. The convergence time for an LMS equalizer can be measured from equal-

¹In this context a time of 1 symbol is the time between transmission of successive symbols.

izer error. The nature of adaptive algorithm has the expected value of the magnitude of the equalizer error decaying exponentially from an initial value to its steady state value. This can be expressed by

$$E\left[\left|\varepsilon[n]\right|\right] = \varepsilon_{ss} + \left(\left|\varepsilon[0]\right| - \varepsilon_{ss}\right)\varepsilon^{-n/\tau} \quad (4.28)$$

where $\varepsilon_{ss} = E\left[\left|\varepsilon[\infty]\right|\right]$ and τ is the time constant in units symbols.

The definition of convergence used in this thesis is: The equalizer is said to converge at symbol n_0 , where n_0 is the smallest value of n satisfying $\left|\varepsilon[n]\right| \leq (1 + 0.25)\varepsilon_{ss}$.

The convergence time not only depends on the channel, but also on the data. Some of the data dependency can be removed by fitting an exponential curve to $|\varepsilon[n]|$ and then using the smooth curve to find the convergence time. This results in

$$t_0 = \tau \ln \left(\frac{|\varepsilon[0]| - \varepsilon_{ss}}{0.25\varepsilon_{ss}} \right) \quad (4.29)$$

where t_0 is the convergence time in symbols, ε_{ss} and τ are the parameters that make $\varepsilon_{ss} + \left(\left|\varepsilon[0]\right| - \varepsilon_{ss}\right)\varepsilon^{-n/\tau}$ a least mean squared error fit to $\left|\varepsilon[n]\right|$.

5. Implementation of the DOCSIS Equalizer

The practical performance of an equalizer will not achieve the theoretical performance. The primary reason for this is that the arithmetic in a practical equalizer is based on finite word length. There is cost associated with the multipliers and adders used in the arithmetic depend on the length of the words. The relationship between the cost of the algorithm and the word length depends on the technology used to implement the equalizer.

An equalizer can be implemented in a variety of technologies. The preferred technologies in the cable industry are application specific integrated circuit (ASIC) for the high volume user-end demodulators and field programmable gated array (FPGA) for the more modest volume head-end demodulators.

Commercially the implementation of the DOCSIS upstream demodulator is very popular in implementing with ASIC. Though ASICs are popular, they have some disadvantages too.

The fabrication process to manufacture an ASIC is very costly. For a modest volume applications, it is too expensive to work with ASIC technology. In addition, ASICs are not programmable and can not be upgraded [30]. This means they can not be integrated into a product until the standard is mature and steady. Furthermore, the fabrication of an ASIC integrated circuits is a lengthy process, which delays entry into the market and cost market share. And not to be marginalized is the added cost of testing ASICs for fabrication malfunction. These testing procedures cause further delays in market entry.

In order to shorten the time to market and eliminate the re-occurring fabrication cost of ASICs, manufacturing industries are turning to FPGA technology. Today, cable equipment manufacturers are implementing the DOCSIS upstream demodulators in FPGAs.

FPGAs are basically a programmable ASICs. The difference is that the tracks required programming consume a large amount of silicon making the chip larger and more expensive. This extra per unit cost is offset by the absence of non re-occurring engineering cost. FPGA also have a great advantages when time to market is important.

Each FPGA contains finite number of logic elements, which are used to implement an application. In the 1980's, when the size of FPGAs was smaller, the complex functions like equalizers, could not be implemented in FPGA due to the limited number of logic elements. Now, the number of logic elements in FPGA has been increased many fold making it possible to implement complex functions. Now FPGA have taken over many applications that were once the sole domain of ASICs. As a result the use of FPGAs throughout the world has increased tremendously.

The modern day FPGAs has some specialized blocks such as DSP blocks, built in RAM etc. Some FPGAs have the option to interface with other devices. These extra features making FPGAs increasingly more popular.

The major limitation of an FPGA is the limited number of multipliers. If an application algorithm requires a very large number of multiplications, ASICs may be needed. Often such algorithms, at expense of performance, are modified to reduce the number of multipliers so they will fit in an FPGA. Furthermore, the algorithm must be arranged so that all multipliers can be time shared to run at their maximum rate.

This thesis implements the DOCSIS equalizer with the economy of FPGA in mind. Therefore, it must be implemented with a minimum number of multipliers that are time shared to the extent possible.

5.1 Implementation Roadmap

The implementation of the equalizer is basically conducted in two stages: the simulation stage and the implementation stage. The simulation stage involves working with MATLAB and DSP builder whereas the implementation stage involves synthesizing the equalizer in the FPGA using the Quartus II Design Software, which is produced by Altera corporation.

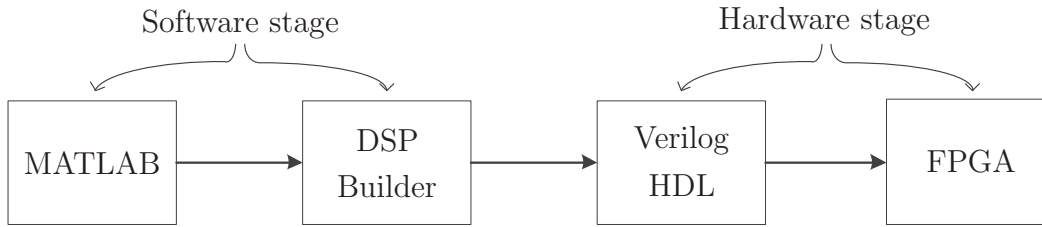


Figure 5.1 A high-level structure of implementation roadmap.

A high level structure of the implementation is shown in Figure 5.1.

MATLAB is used as the simulation tool that models a cost effective equalizer with carefully designed simulations. Later the same model is developed in DSP builder, which utilize MATLAB simulink. DSP builder is nothing but a suite of MATLAB simulink blocks. These blocks are different than the normal MATLAB simulink blocks in that they have extra features. DSP builder, in addition to generating a simulink model of the equalizer, generates the VHDL code that implements the simulink model of the equalizer. The VHDL code is then rapped in a verilog shell and used in Quartus as a Verilog HDL module.

The formation of the remainder of this chapter explores the effect of practical implementation issues using simulation. Later the results will be analysed to find cost effective equalizer. At the close of the chapter the implementation in hardware is discussed in detail.

5.2 Implementation Issues to Find a Cost Effective Equalizer

The implementation of DOCSIS equalizer is not straight forward. There are many factors such as length of the equalizer, step size, convergence time, MER etc., most of which are related to each other. This complicates the task of finding the best equalizer in terms of cost-performance trade off. To find a suitable model for the equalizer, a detail analysis need to be carried out and this analysis involves carefully designed simulations.

Length of the Equalizer

The DOCSIS standard allows the length of the equalizer to be up to 24. On the surface it may appear that the longer the length of equalizer, the better the performance. It is simply not true. At some point increasing the length of an equalizer degrades the performance. The question that needs to be answered is: is length 24 the optimum length for a DOCSIS equalizer or should it be shorter?

To find a suitable equalizer length an analysis is necessary. To start the analysis, a MATLAB simulation that estimates the MER is necessary. The analysis was performed for equalizer of lengths 20 and 24.

The results of the analysis are plotted in Figure 5.2 as MER versus echo delay. Each point on the MER curves is obtained by processing the first 5,000 QPSK data symbols in payload of the packet when the equalizer is in data mode. A channel with single echo of strength 0.3162 of the direct path was constructed with a parameter that controls the delay of the echo. The MERs plotted in Figure 5.2 are a function of the delay of echo with respect to the main path. The delay axis has unit in symbols.

There are 2 MER curves in Figure 5.2. One is for a 20 tap equalizer and the other is for 24 tap equalizer. The top curve, with the points of measurement marked with a ‘□’ is for 20 tap equalizer. The bottom curve, with the points of measurement marked with a ‘△’ is for 24 tap equalizer. Both equalizers use a step size of $\mu = 1/64$. It is clear that the 20 tap equalizer outperforms the 24 tap equalizer for all echo delay between 0 and 3 symbol.

The reason for that is, in case of 24 tap equalizer, once the equalizer is converged, the extra 4 taps has a weight very close to zero and are contributing to the error measured at the output of the equalizer. The result is a reduced MER. On the other hand, in case of 20 tap equalizer, there is no chance for any contribution by the extra 4 taps to the estimated error. As a result MER is better in this case.

The MER is only one of two performance measures. The second performance measure is the convergence time, which is the time required for the mean squared error at the output

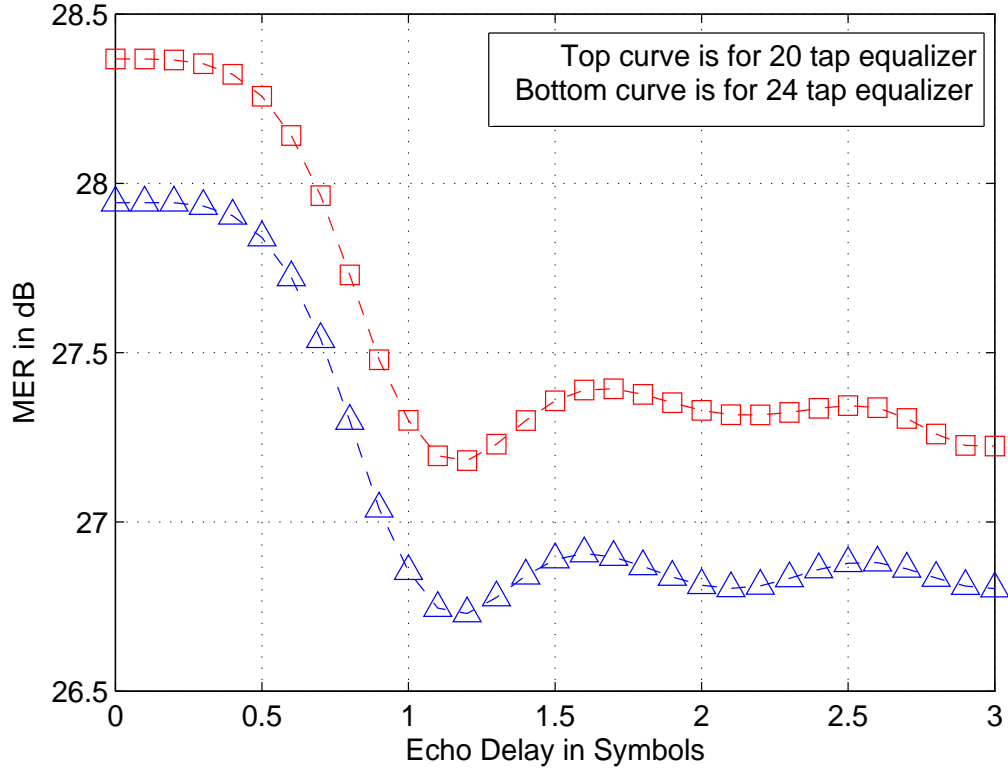


Figure 5.2 MER comparison in a DOCSIS single echo channel with echo strength 0.3162 of main path and a step size of $\mu = \frac{1}{64}$ in a 20 and 24 tap equalizers.

of the equalizer to drop below some threshold. The mean squared error at the output of the equalizer is a function of time. The instantaneous error for one packet is plotted in Figures 5.3 and 5.4 for equalizer lengths 20 and 24 respectively. While some judgement is required to ascertain the mean squared error from plots of instantaneous error. It appears from these figures, it takes approximately 200 symbols for the mean squared error in both equalizers to drop below 0.1 of the expected decision variable voltage.

From above analysis it is clear that a 20 tap equalizer outperforms a 24 tap equalizer for a single delay channel of strength 0.3162 for echo delays up to 3 symbol. Neither have an advantage in convergence time. Both require 200 symbols to converge. The 20 tap equalizer does however outperform the 24 tap equalizer in terms of MER. Perhaps most importantly

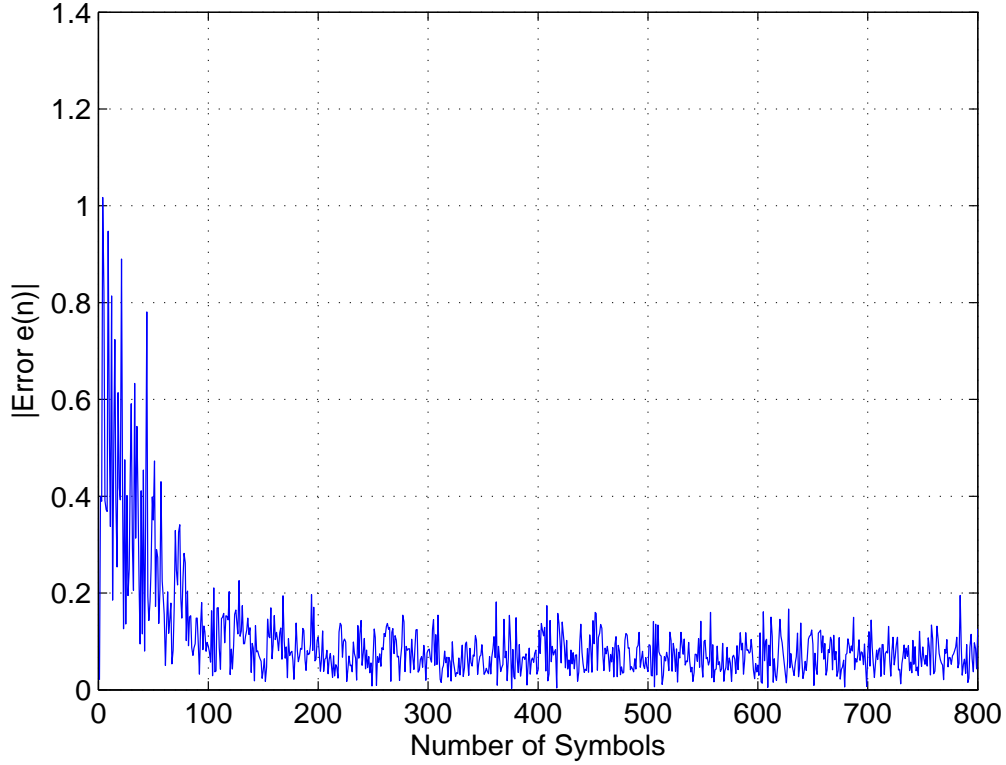


Figure 5.3 Equalizer error in a 20 tap equalizer with step size, $\mu = \frac{1}{64}$.

the 20 tap equalizer requires fewer multipliers, giving it a great cost advantage. Since delays for 0 to 3 symbols cover the range of possible DOCSIS channels and echo strength of 0.3162 is the maximum possible, it can be concluded that the 20 tap equalizer is the better choice. The remainder of the thesis is concentrated on 20 tap equalizer.

Step Size

It is difficult to determine the optimum step size. Reducing the step size has a negative impact on the convergence time but a positive impact on the MER. The reason being, the correctional increments to the equalizer coefficients is proportional to the step size. A smaller increment means, a longer time to achieve the steady state coefficients, which means a longer convergence time. On the other hand, small increments means, the coefficients remain close to their steady state values making the MER better. It is clear that, the selection of a suitable step size requires a trade off between the convergence time and the MER.

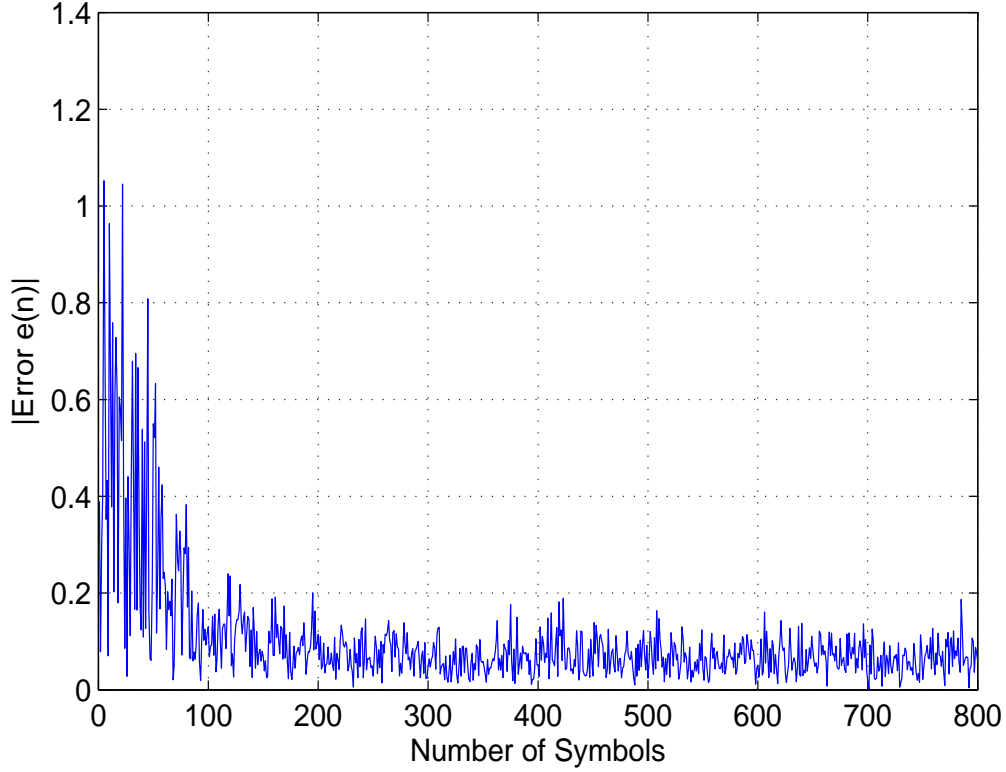


Figure 5.4 Equalizer error in a 24 tap equalizer with step size, $\mu = \frac{1}{64}$.

The selection of the step size value is not just random. If a step size is chosen to be a negative power of 2, it can be easily implemented in FPGA just by using a shift operator. Most importantly, this way of selecting step size saves a multiplier. With this view in mind, for comparison purpose, 3 step sizes of $1/32$, $1/64$ and $1/128$, which are negative power of 2 (2^{-5} , 2^{-6} and 2^{-7}) is chosen here.

The convergence time and the MER, both are measured from equalizer error. A MATLAB simulation is conducted to plot the instantaneous equalizer error for a 20 tap equalizer. The instantaneous error is plotted for 3 step sizes, $1/32$, $1/64$ and $1/128$ for single packet transmission over the same DOCSIS channel. The instantaneous equalizer error, or more precisely, the magnitude of the instantaneous error is shown in Figures 5.5, 5.6 and 5.7 for step sizes, $1/32$, $1/64$ and $1/128$ respectively. The error is normalized to the expected value of the decision variable and is shown in linear scale. The time axis is normalized to the time

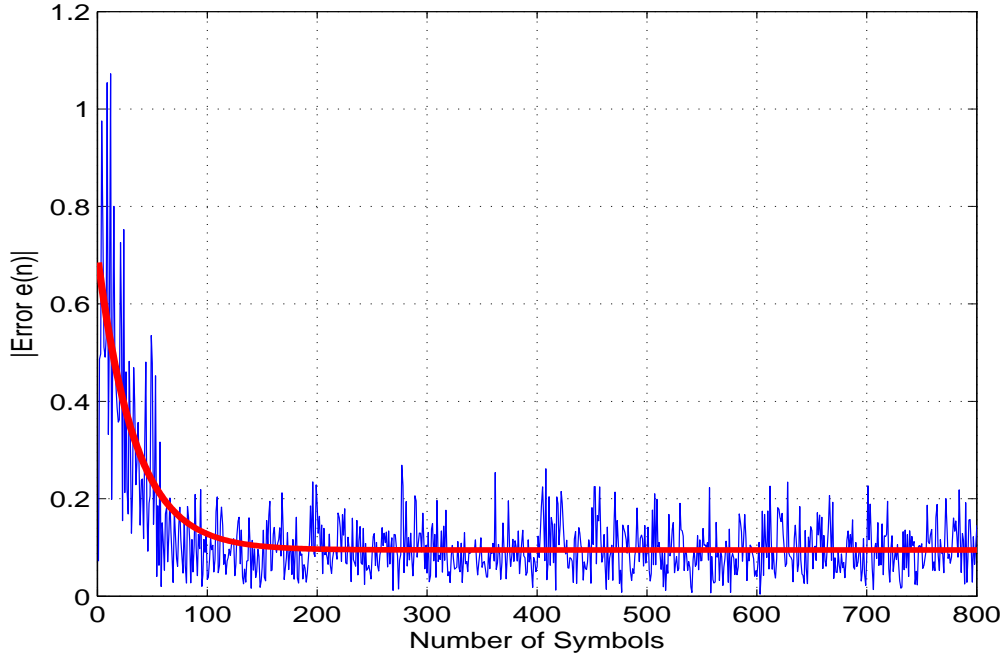


Figure 5.5 Equalizer error with step size, $\mu = \frac{1}{32}$.

between symbols.

The convergence time is measured using mean squared error, whereas the error curves in Figures 5.5, 5.6 and 5.7, plot instantaneous error. To properly reveal the effects of step size on convergence time, a curve (thick solid line), that represent the magnitude of mean squared error, has been fitted to the instantaneous data.

A close look at Figures 5.5, 5.6 and 5.7 reveals that decrease in the step size (i.e. $\frac{1}{32} > \frac{1}{64} > \frac{1}{128}$) cause the convergence time to increase. It is also clear from the figures that with decreasing step size (i.e. $\frac{1}{32} > \frac{1}{64} > \frac{1}{128}$), the magnitude of mean squared equalizer error decreases in steady state, which confirms an increasing MER.

The smooth fitted curves in Figures 5.5, 5.6 and 5.7 shows the steady state mean squared error, which are 0.1, 0.075 and 0.05 respectively to each error curve is required to measure the convergence time. The convergence times and MERs measured using the three fitted curves are tabulated in Table 5.1.

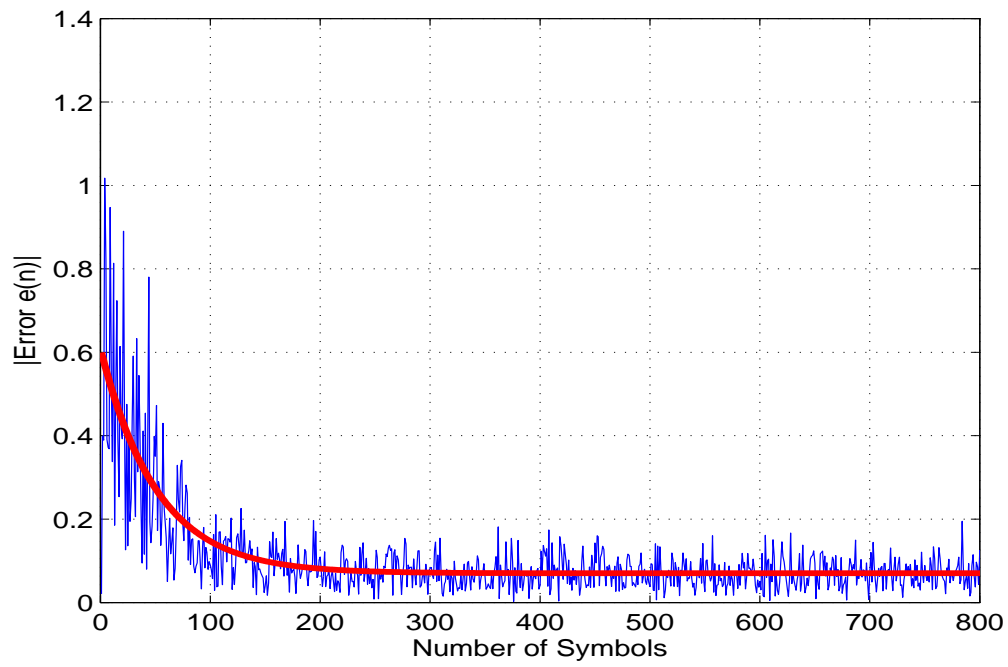


Figure 5.6 Equalizer error with step size, $\mu = \frac{1}{64}$.

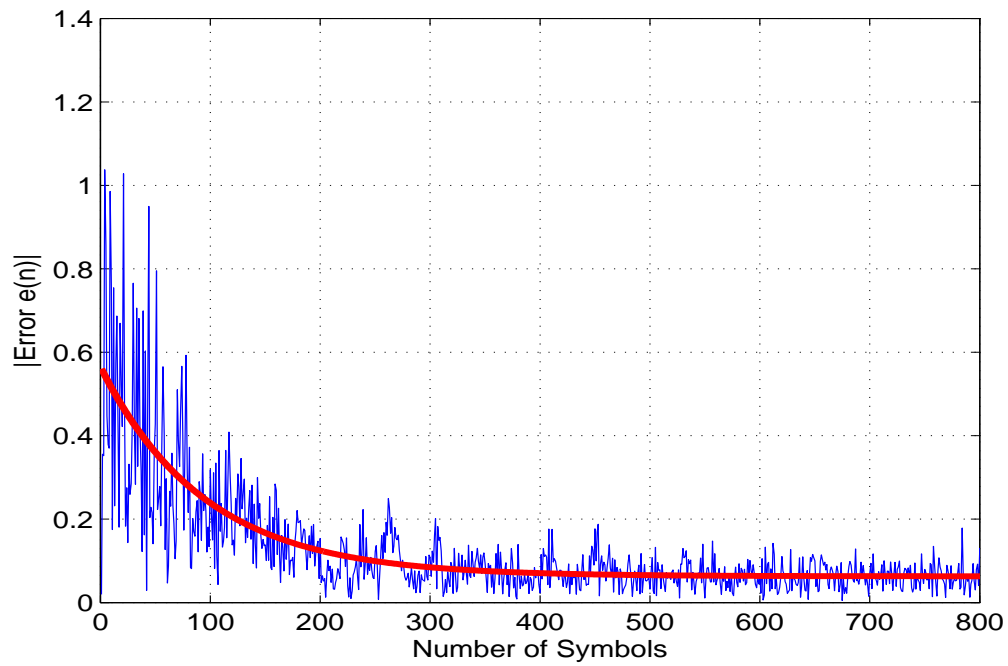


Figure 5.7 Equalizer error with step size, $\mu = \frac{1}{128}$.

Table 5.1 Performance measures of a 20 tap equalizer with different step sizes for a threshold of 125% of mean squared error in steady state.

Step Size (μ)	Convergence Time (in symbols)	MER (in dB)
1/32	104	22.56
1/64	167	25.18
1/128	316	26.13

From the tabulated results, it is clear that, the step size has an impact on both performance measures. Both increase inversely to the step size. Since a low convergence time and high MER are required, the selection of a suitable step size requires a trade off between convergence time and MER. In order to limit the training sequence to 200 symbols, the step size must be greater than 1/128. Clearly the MER is minimized by choosing the largest step size within this constraint. Therefore, step size, $\mu = 1/64$ is a best choice.

Performance Measures

The MER and the convergence time are effected by numerous factors. The effect of step size on these performance measures has already been addressed. Besides step size, another important factor that effects performance, is the channel response. In order to see the effect of channel response on performance, the same single echo channel used to plot Figure 5.2 (page 65), is used here. The simulation is conducted for a 20 tap equalizer with step size of 1/64.

The MERs and the convergence times are plotted in Figure 5.8 and 5.9 respectively. Both of them are plotted as a functions of the delay in echo with respect to the direct path. The first 6 points in Figure 5.9 indicate the convergence time is zero. This happens when the initial mean squared error is within the 125% of steady state threshold.

The MER values in the MER curve start falling from its maximum (28.4 dB) with the increase in echo delay and reaches to its minimum (27.2 dB), when the echo delay is approximately 1.2 symbols with respect to the direct path of the channel. For higher echo

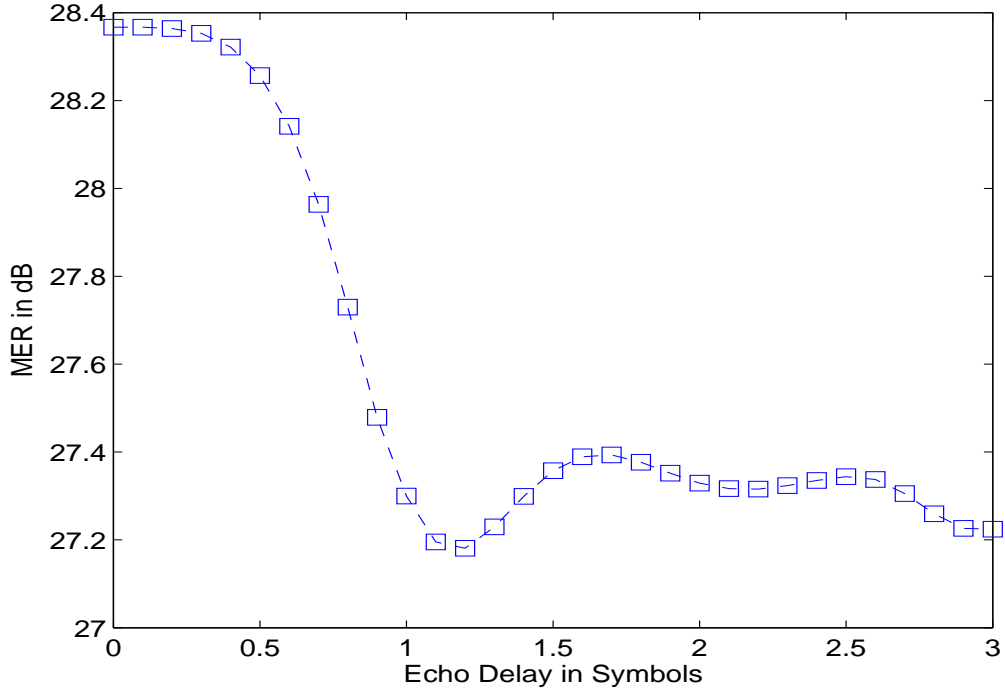


Figure 5.8 MER in a DOCSIS single echo channel with echo strength 0.3162 of main path and a step size of $\mu = \frac{1}{64}$ in a 20 tap equalizer.

delays, the MER deviates a little bit but stays close to the minimum MER. The maximum deviation is 1.2 dB.

The convergence time plot in Figure 5.9 starts increasing from 0 at an echo delay of 0.5 symbols and reaches to its maximum 150 symbols at an echo delay of 1.2 symbols. For higher echo delays, the convergence time remains just below the maximum. It is clear that in terms of convergence time, the worst case single echo channel has an echo delay of 1.2 symbols. At this echo delay, the equalizer requires 150 symbols to converge.

From the above analysis, it can be concluded that, a cost effective equalizer of length 20 should have a step size of $1/64$. Moving forward, the studies will be centred around the 20 tap equalizer with a step size of $1/64$.

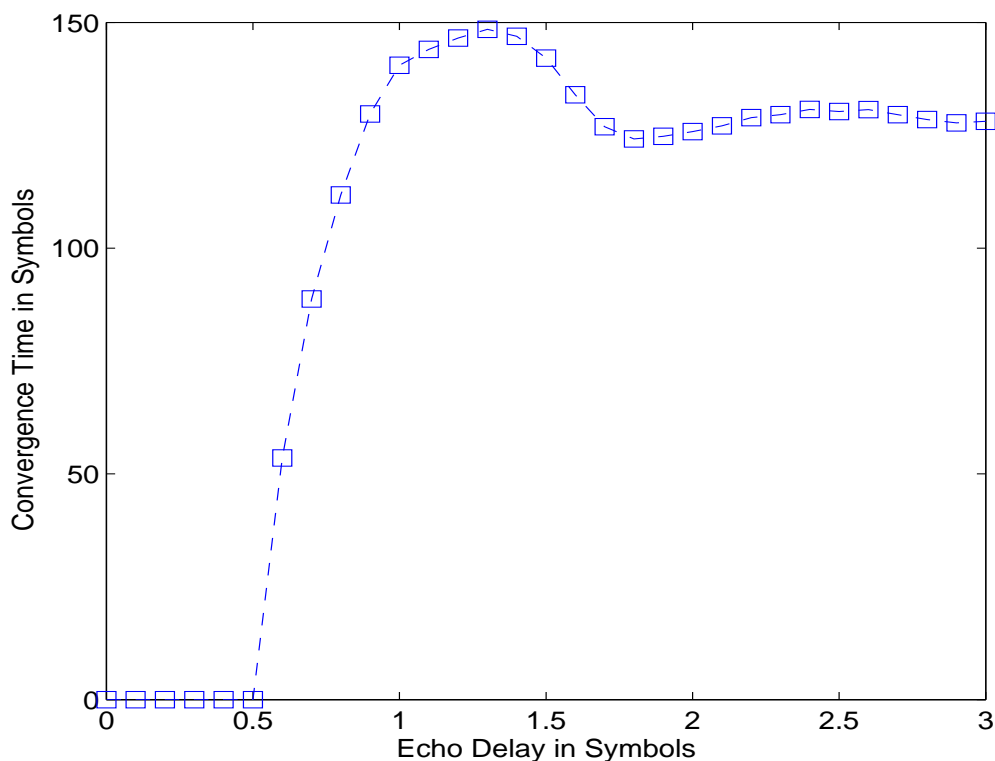


Figure 5.9 Convergence time in a DOCSIS single echo channel with echo strength 0.3162 of main path and a step size of $\mu = \frac{1}{64}$ in a 20 tap equalizer.

5.3 Effect of Timing Error on Equalizer Performance

The analysis of the equalizer is not complete yet. The primary analysis in section 5.2 (page 63), paved the way for modelling a cost effective equalizer. To fine tune the efficiency of this cost effective equalizer the analysis must be extended to include the effect of the timing recovery circuitry on the performance.

Unfortunately, the performance of the equalizer depends strongly on the sampling times. The performance is maximized if the sampling time at the output of the matched filter coincides with the time where the eye in the eye diagram [31] is maximally open, is called the correct sampling time [32].

It is likely, in the absence of correction, the sampling time will be offset from the optimum time. A time offset can be a positive or negative and is measured with respect to the correct

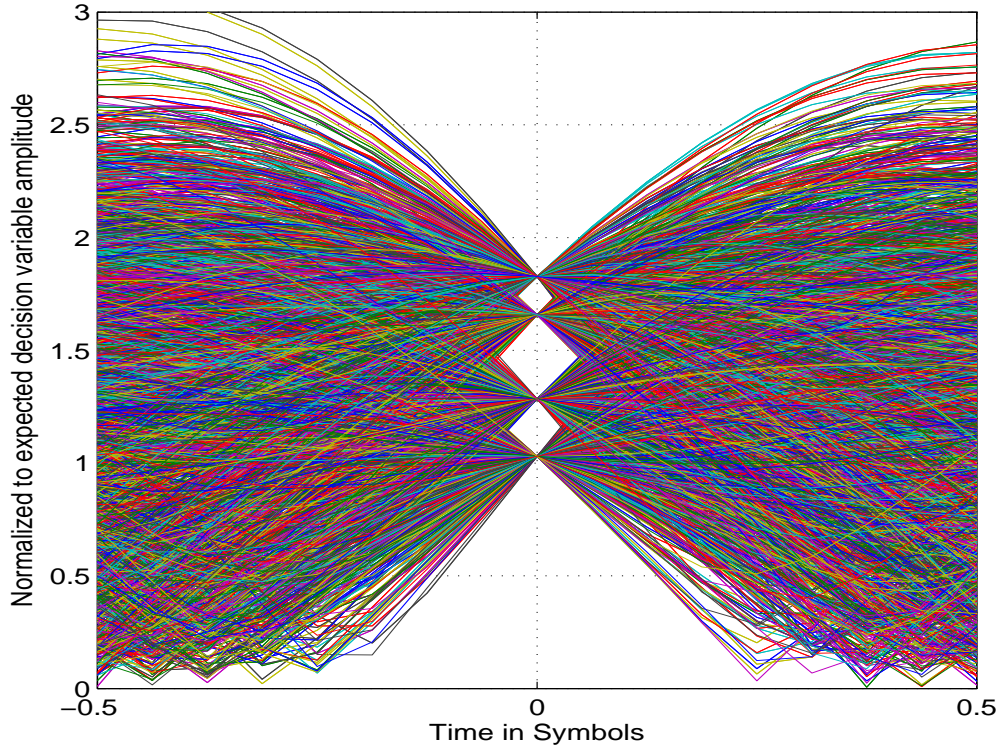


Figure 5.10 Magnitude of Matched Filter Output in Eye Diagram Format.

sampling time in units of symbols.

5.3.1 Establishing the Timing Reference

The origin for “time offset” is where the eye is maximally open, which is the point in time where the distance between the constellation points and the output of matched filter is statistically smallest. This point in time is found in MATLAB by processing 10,000 QPSK symbols with a matched filter. The absolute value of the output is taken and then it is segmented into 5000 vectors each two symbol in length. A truncated plot of the 5000 vectors showing only the first of the two symbols shown in Figure 5.10. The plot in Figure 5.10 is for a single echo channel where the echo has maximum strength (0.3162) and delay (3 symbols) with respect to the main path.

Another vector, referred to as the *min_of_magnitude* vector, is obtained by taking the

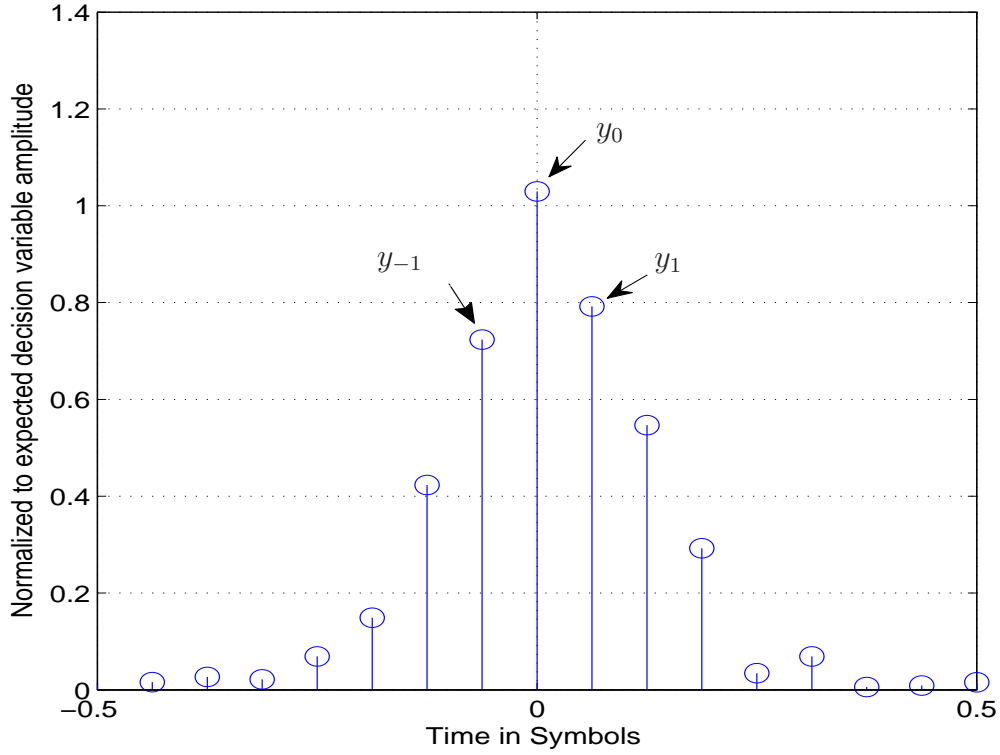


Figure 5.11 Minimum of Magnitudes of Matched Filter Output.

minimum across the 5000 vectors. The *min_of_magnitude* vector for the vectors plotted in Figure 5.10 is plotted in Figure 5.11. The *min_of_magnitude* vector represents a sampled analog waveform whose maximum value occurs at the point in time where the eye is maximally open. It is this point in time that will serve as the time origin.

The time at which the eye is maximally open is found from the *min_of_magnitude* vector using a parabolic approximation to the curve given by [33]

$$t_p = \frac{|y_{-1}| - |y_1|}{2(|y_{-1}| - 2|y_0| + |y_1|)} \quad (5.1)$$

where y_0 is the maximum in the *min_of_magnitude* vector and y_{-1} and y_1 are the neighbouring samples and t_p is the time corresponding to peak of the parabolic curve which is the time the eye is maximally open. The unit of t_p is the time between samples measured with respect to the time of y_0 . t_p can be expressed in unit of symbols by multiplying it by ratio of symbol rate to sample rate.

Once t_p is found, the impulse response of $g[l]$ (page 25) can be modified to introduce a delay that aligns sampling points with t_p . A time offset is introduced by increasing or decreasing this delay.

The MATLAB simulations to see the effect of the timing offset on MER and convergence time is included in section 6.1 of chapter 6.

5.4 Implementation Structure of the Equalizer

The problem at hand is to build a complex coefficient equalizer in an FPGA that supports only real mathematics. This forces to develop and expand the required complex mathematics to build a detail structure of the complex coefficient equalizer.

The mathematics that follows to develop a detail structure of the complex equalizer involves complex mathematical operation such as multiplication and addition. The property of the complex number is such that, the product of two complex number produces 4 terms, two of which form the real part of the resultant complex number and the other two form the imaginary part. As the inputs as well as the coefficients of the equalizer are complex, the product of these two complex number also produces a 4 term complex number.

The primary requirement for building a detail structure of the equalizer is to view the input data, output data and the tap weights as complex numbers. The inputs (the tap input vector $\vec{v}[n]$ and the training data $I[n]$), the outputs (the equalizer output $\hat{I}[n]$ and the equalizer error $\varepsilon[n]$) and the tap weight vector $\vec{c}[n]$ shown in Figure 4.1 (page 56) can be expressed with their real and imaginary part as follows:

$$\vec{v}[n] = \vec{v}_I[n] + j\vec{v}_Q[n]. \quad (5.2)$$

$$I[n] = I_I[n] + jI_Q[n]. \quad (5.3)$$

$$\hat{I}[n] = \hat{I}_I[n] + j\hat{I}_Q[n]. \quad (5.4)$$

$$\varepsilon[n] = \varepsilon_I[n] + j\varepsilon_Q[n]. \quad (5.5)$$

$$\vec{c}[n] = \vec{\alpha}_I[n] + j\vec{\beta}_Q[n]. \quad (5.6)$$

where the subscripts I and Q stands for in-phase and quadrature phase components respectively. These definitions are required to view the complex equalizer output $\hat{I}[n]$, equalizer error $\varepsilon[n]$ and the updated complex tap weight vector $\vec{c}[n+1]$ as a combination of real and imaginary numbers.

Substituting (5.2), (5.4) and (5.6) into (4.3) (page 49) has

$$\begin{aligned}
\hat{I}[n] &= \hat{I}_I[n] + j\hat{I}_Q[n] = \vec{v}[n]\vec{c}^T[n] \\
&= \left(\vec{v}_I[n] + j\vec{v}_Q[n]\right)\left(\vec{\alpha}_I[n] + j\vec{\beta}_Q[n]\right)^T \\
&= \left(\vec{v}_I[n] + j\vec{v}_Q[n]\right)\left(\vec{\alpha}_I^T[n] + j\vec{\beta}_Q^T[n]\right) \\
&= \vec{v}_I[n]\vec{\alpha}_I^T[n] + j\vec{v}_I[n]\vec{\beta}_Q^T[n] + j\vec{v}_Q[n]\vec{\alpha}_I^T[n] - \vec{v}_Q[n]\vec{\beta}_Q^T[n] \\
&= \vec{v}_I[n]\vec{\alpha}_I^T[n] - \vec{v}_Q[n]\vec{\beta}_Q^T[n] + j\left(\vec{v}_I[n]\vec{\beta}_Q^T[n] + \vec{v}_Q[n]\vec{\alpha}_I^T[n]\right). \quad (5.7)
\end{aligned}$$

Expressing the real and imaginary parts separately has

$$\hat{I}_I[n] = \vec{v}_I[n]\vec{\alpha}_I^T[n] - \vec{v}_Q[n]\vec{\beta}_Q^T[n] \quad (5.8)$$

and

$$\hat{I}_Q[n] = \vec{v}_I[n]\vec{\beta}_Q^T[n] + \vec{v}_Q[n]\vec{\alpha}_I^T[n]. \quad (5.9)$$

Substituting (5.3) and (5.4) into (4.5) (page 50) has

$$\begin{aligned}
\varepsilon[n] &= \varepsilon_I[n] + j\varepsilon_Q[n] = \hat{I}[n] - I[n] \\
&= \left(\hat{I}_I[n] + j\hat{I}_Q[n]\right) - \left(I_I[n] + jI_Q[n]\right) \\
&= \hat{I}_I[n] - I_I[n] + j\left(\hat{I}_Q[n] - I_Q[n]\right). \quad (5.10)
\end{aligned}$$

Expressing the real and imaginary parts separately has

$$\varepsilon_I[n] = \hat{I}_I[n] - I_I[n] \quad (5.11)$$

and

$$\varepsilon_Q[n] = \hat{I}_Q[n] - I_Q[n]. \quad (5.12)$$

Substituting (5.2), (5.5) and (5.6) into (4.27) (page 54) has

$$\begin{aligned}
\vec{c}[n+1] &= \vec{\alpha}_I[n+1] + j\vec{\beta}_Q[n+1] \\
&= \vec{c}[n] - \mu\varepsilon[n]\vec{v}^*[n] \\
&= \vec{\alpha}_I[n] + j\vec{\beta}_Q[n] - \mu\left(\varepsilon_I[n] + j\varepsilon_Q[n]\right)\left(\vec{v}_I[n] + j\vec{v}_Q[n]\right)^* \\
&= \vec{\alpha}_I[n] + j\vec{\beta}_Q[n] - \mu\left(\varepsilon_I[n] + j\varepsilon_Q[n]\right)\left(\vec{v}_I[n] - j\vec{v}_Q[n]\right) \\
&= \vec{\alpha}_I[n] + j\vec{\beta}_Q[n] - \mu\left(\varepsilon_I[n]\vec{v}_I[n] - j\varepsilon_I[n]\vec{v}_Q[n] + j\varepsilon_Q[n]\vec{v}_I[n] + \varepsilon_Q[n]\vec{v}_Q[n]\right) \\
&= \vec{\alpha}_I[n] + j\vec{\beta}_Q[n] - \mu\varepsilon_I[n]\vec{v}_I[n] + j\mu\varepsilon_I[n]\vec{v}_Q[n] - j\mu\varepsilon_Q[n]\vec{v}_I[n] - \mu\varepsilon_Q[n]\vec{v}_Q[n] \\
&= \vec{\alpha}_I[n] - \mu\left(\varepsilon_I[n]\vec{v}_I[n] + \varepsilon_Q[n]\vec{v}_Q[n]\right) + j\left(\vec{\beta}_Q[n] + \mu\left(\varepsilon_I[n]\vec{v}_Q[n] - \varepsilon_Q[n]\vec{v}_I[n]\right)\right).
\end{aligned} \tag{5.13}$$

Expressing the real and imaginary parts separately has

$$\vec{\alpha}_I[n+1] = \vec{\alpha}_I[n] - \mu\left(\varepsilon_I[n]\vec{v}_I[n] + \varepsilon_Q[n]\vec{v}_Q[n]\right) \tag{5.14}$$

and

$$\begin{aligned}
\vec{\beta}_Q[n+1] &= \vec{\beta}_Q[n] + \mu\left(\varepsilon_I[n]\vec{v}_Q[n] - \varepsilon_Q[n]\vec{v}_I[n]\right) \\
&= \vec{\beta}_Q[n] - \mu\left(\varepsilon_Q[n]\vec{v}_I[n] - \varepsilon_I[n]\vec{v}_Q[n]\right).
\end{aligned} \tag{5.15}$$

Transforming (5.8), (5.9), (5.11), (5.12), (5.14) and (5.15) to an equivalent schematic has a canonical structure of the complex equalizer, which is shown in Figure 5.12.

It is obvious from Figure 5.12 that the complex LMS equalizer is comprised of 4 real cross-coupled LMS equalizers. Although this structure gives a clear picture of the complex adaptive equalizer, it does not necessarily make optimum use of the multipliers.

It is clear from Figure 5.12 that to achieve $\vec{\alpha}_I[n+1]$, four vector products need to be calculated. These vector products are: $\vec{v}_I[n]\vec{\alpha}_I[n]$, $\vec{v}_Q[n]\vec{\beta}_Q[n]$, $\varepsilon_I[n]\vec{v}_I[n]$ and $\varepsilon_Q[n]\vec{v}_Q[n]$. Since each vector has 20 elements, each vector product requires 20 multiplies. This means 80 multiplies are required to achieve $\vec{\alpha}_I[n+1]$ and 80 more multiplies to achieve $\vec{\beta}_Q[n+1]$. Clearly, a total of 160 multiplies required to achieve $\vec{c}[n+1]$. Without optimization, the implementation cost would be excessive. The cost is greatly reduced by time sharing multipliers.

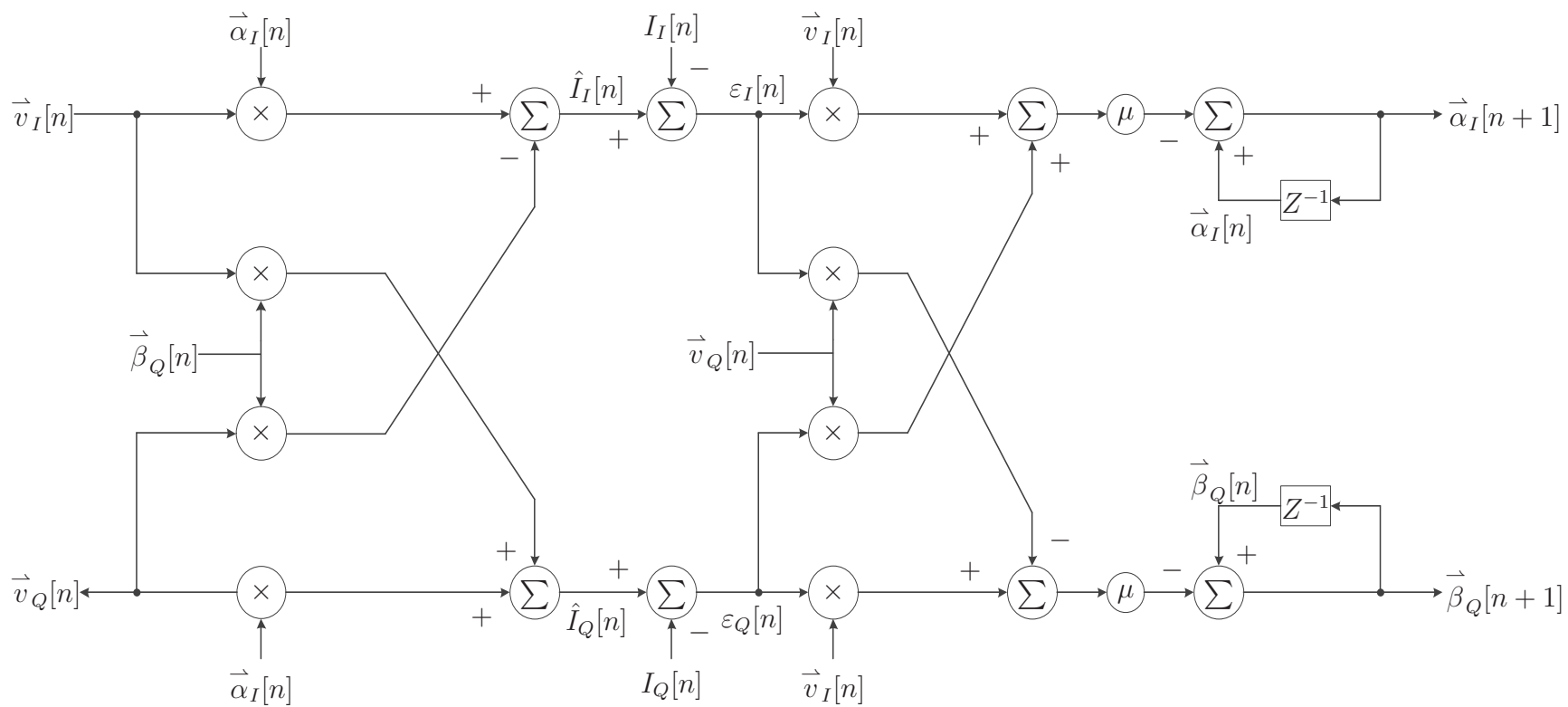


Figure 5.12 Canonical structure of equalizer.

Multipliers can be only time shared if the system clock is running at a multiple of the symbol rate. For example, if the clock rate is 20 times the symbol rate then one multiplier can be used to compute 20 multiplies. Assuming a clock rate of 20 times the symbol rate, the total 160 multiplies in a 20 tap equalizer can be done with 8 multipliers.

Based on the information in this chapter, it is clear that a 20 tap equalizer outperforms a 24 tap equalizer. For this reason, a 20 tap equalizer with a step size of $1/64$ will be used going forward.

It should be mentioned that the decision to use a 20 tap equalizer is based on a single echo with maximum delay of 3 symbols and a strength of 0.3162 with respect to the main path. However, if multiple echoes or a single echo with a longer delay was considered, a 24 tap equalizer may rival the 20 tap equalizer and the decision to use a 20 tap equalizer may not have been made.

The next step is to model the cost optimized equalizer in DSP builder, which is in effect is in MATLAB simulink. Once the cost optimized equalizer is modelled in DSP Builder then the performance of the equalizer need to be verified. A detail analysis and verification of the length 20 equalizer, with a step size of $1/64$, is included in the next chapter with MATLAB and DSP Builder based simulation results.

6. Simulation Results

The theoretical and analytical studies in the previous chapters have already confirmed an optimized model of the cost effective equalizer that is to be implemented in FPGA. The decision came after analysing the carefully made simulations and after proposing a scheme to utilize minimum possible multipliers for the 20 tap complex adaptive equalizer.

The simulations conducted in the last chapter covered a corner case for the channels and were designed to find the best equalizer. In this chapter, simulations conducted in MATLAB and DSP builder are designed to span all possible channels.

6.1 MATLAB Simulation Results

In the previous chapters some simulation results for the MER and the convergence time has already been analysed. Some of the simulation results showed the effect of the presence of micro-reflections in the DOCSIS channel and others showed the effect of step size and equalizer length. One other important issue, which is the effect of the timing error on equalizer performance, was addressed to fine tune the efficiency of the equalizer. Although the theory was explained in detail, there was no simulation for analysing MER and convergence time in support of the theory.

To be in consistent with the theory, it is now necessary to have some simulation results to see the effect of timing error on MER and convergence time. The MER is simulated for both finite (20 tap) and infinite lengths whereas the convergence time is simulated only for the finite length (20 tap) equalizer.

The theoretical maximum for the MER for an infinite length equalizer with optimum

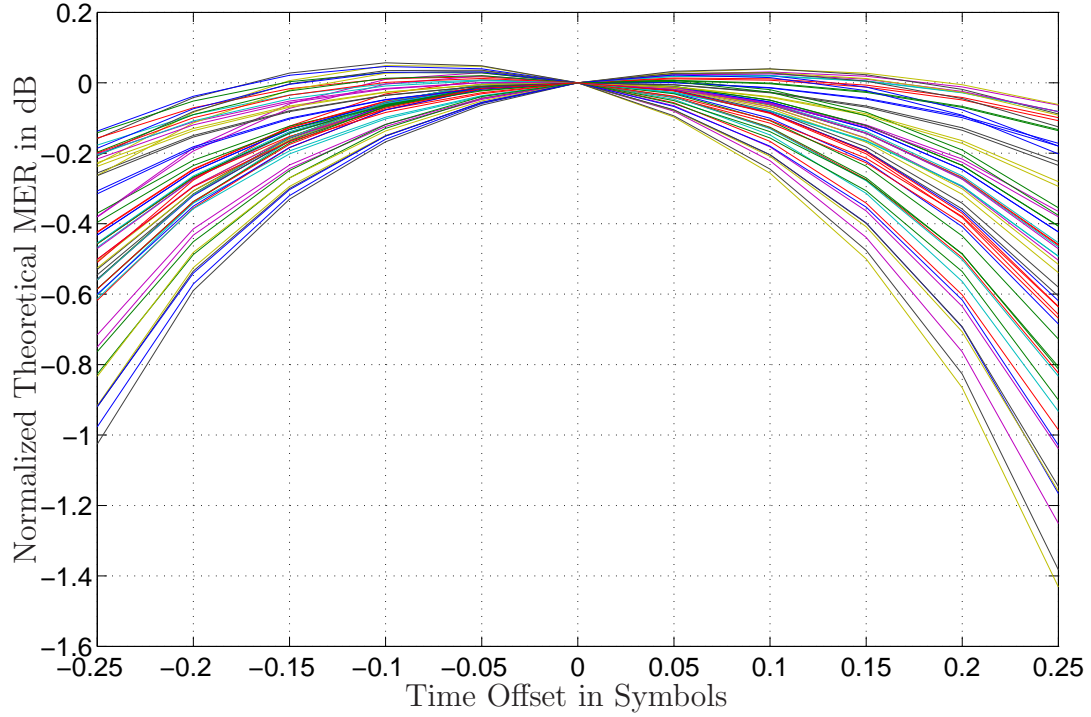


Figure 6.1 Normalized theoretical MER for a SNR of 30 dB and an infinite length equalizer

coefficients is given by (3.57) (page 46) and repeated here for convenience:

$$MER = \frac{\sigma_I^2 - J_{min}}{J_{min}}, \quad (6.1)$$

where σ_I^2 is the variance of the input sequence and J_{min} is the minimum mean square error given by (3.45) (page 38).

A MATLAB simulations were used to generate 62 MER versus timing offset curves, which is shown in Figure 6.1. The SNR at the input of the receiver was set at 30 dB for all the curves. A different channel was constructed for each curve by randomly selecting the parameters for the DOCSIS channel model. After the parameters were selected the channel impulse response, $h[l]$, was calculated from the model. Each curve consists of 11 simply connected equally spaced points. The timing offset for each point was created by modifying $g[l]$ to introduce a delay.

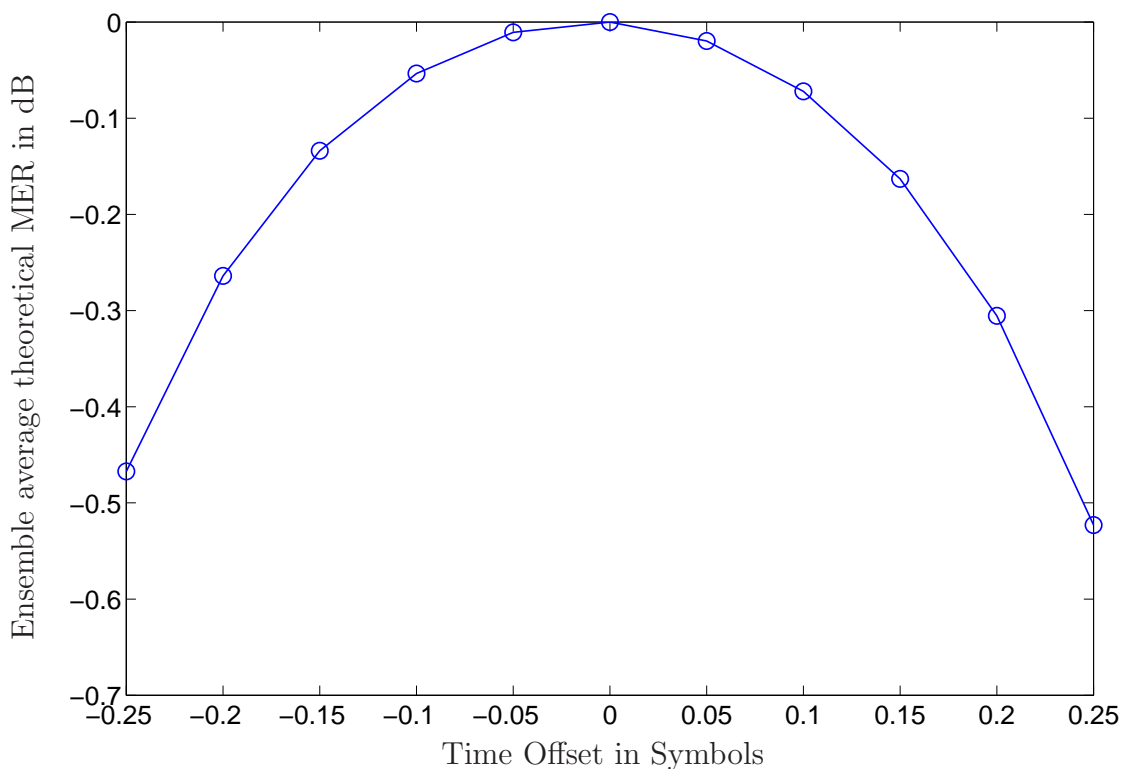


Figure 6.2 Ensemble average of normalized theoretical MERs (infinite length equalizer, SNR = 30 dB)

To better illustrate the effects of timing offset, the effect of the channel on the MER was removed by normalizing the curves with the values at a timing offset of 0. The normalized MER is defined as follows:

$$\text{Normalized MER} = \text{MER for specific timing offset} - \text{MER for timing offset 0}.$$

It appears from Figure 6.1 that the MER is sensitive to timing offset.

The ensemble average of the theoretical MERs plotted in Figure 6.1, is plotted in Figure 6.2. The best MER is obtained when the sampling times correspond to the times the eye is maximally open (i.e. when the timing offset is 0).

A MATLAB model of Figure 5.12 (page 78) (i.e. 20 tap equalizer) was constructed. The equalizer was initialized to zero except for the seventh tap, which was initialized to $1 + 0j$. The training sequence was 3500 symbol long, which is more than sufficient for the equalizer

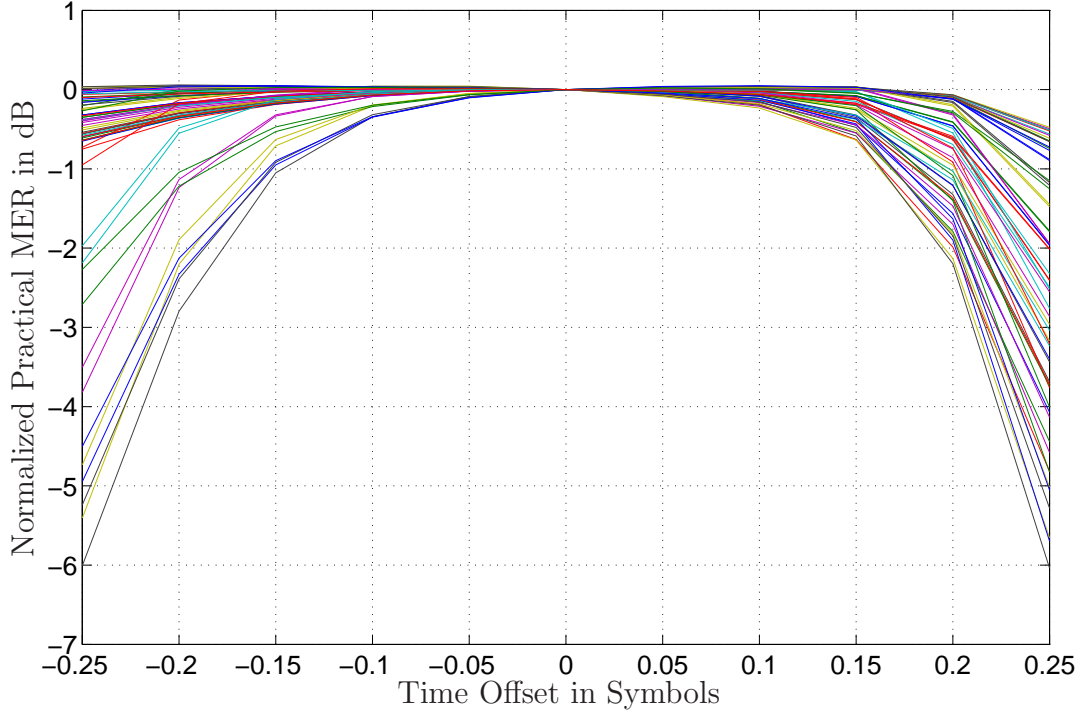


Figure 6.3 Normalized practical MER (20 tap equalizer, SNR = 30 dB)

to converge to its steady state values. The training data was QPSK with a variance of σ_I^2 . The step size, μ , was $1/64$. The MER was measured in data mode using time averages taken over 5000 symbols and is given by

$$\text{MER} = 10 \log_{10} \left(\frac{\overline{I[n]I^*[n]}}{\overline{\varepsilon[n]\varepsilon^*[n]}} \right).$$

The normalized MERs for a 20 tap equalizer are plotted in Figure 6.3 for the same 62 channels. It is interesting to note that this 20 tap (practical) equalizer is considerably more sensitive to timing offset. A closer look to the Figure 6.3 reveals that for timing offset of less than 0.1 symbols the worst case sensitivity of MER to timing offset is about 3dB/symbol.

The ensemble average of the practical MERs plotted in Figure 6.3 is plotted in Figure 6.4. This ensemble average based curve of practical MERs reinforces that the best MER performance is obtained when the matched filter is sampled at the time the eye is maximally open.

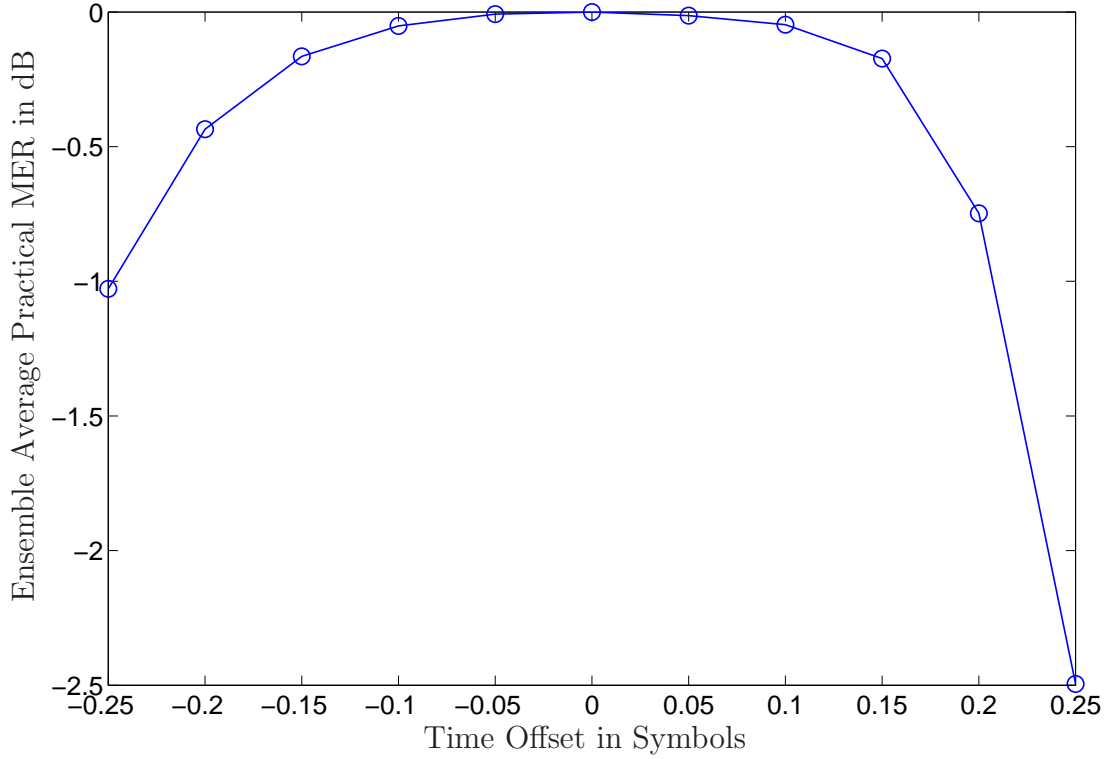


Figure 6.4 Ensemble average of normalized practical MERs (20 tap equalizer, SNR = 30 dB)

The convergence times for timing offsets ranging between -0.25 symbols and 0.25 symbols were found for the same 62 channels by way of a MATLAB simulation using (4.29) (page 60). These convergence times were taken with respect to the convergence time for a timing offset of 0 and plotted in Figure 6.5 as normalized convergence time. It is clear from Figure 6.5 that convergence time is sensitive to timing offset. A timing offset of 0.25 symbols can add up to 140 symbols to the convergence time for timing offset of zero. A closer look to the Figure 6.5 reveals that for timing offset of less than 0.1 symbols, the worst case sensitivity of convergence time to timing offset is about 400 symbols/symbol.

The ensemble average of the convergence times plotted in Figure 6.5, is plotted in Figure 6.6. The minimum ensemble average convergence time is obtained when the matched filter is sampled at the time the eye is maximally open. This again reinforces that sampling

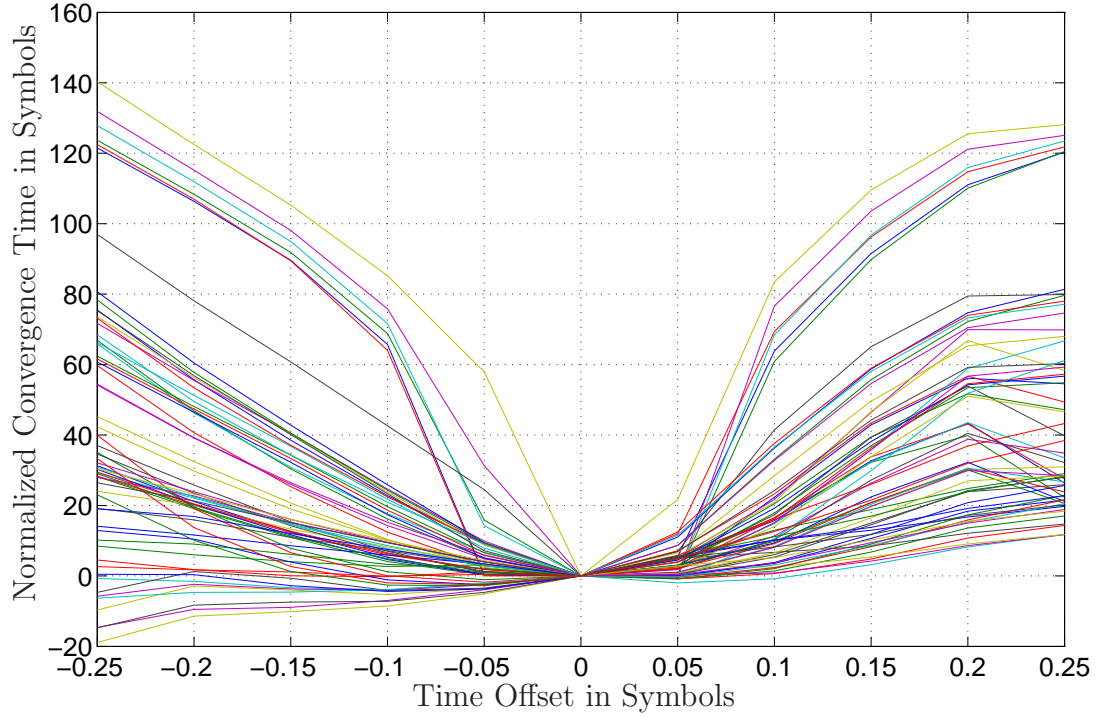


Figure 6.5 Relative convergence time ($\mu\sigma_I = \sqrt{2}/64$) (20 tap equalizer)

the matched filter at the time the eye is maximally open gives the best convergence time (minimum) in terms of symbols.

For more interest histograms of the theoretical MERs (infinite length equalizer), practical MERs (20 tap equalizer) and convergence times (20 tap equalizer) for a timing offset of 0 for the same 62 channels is shown in Figures 6.7, 6.8 and 6.9, respectively. It is clear from Figures 6.7 and 6.8 that the worst case theoretical MER for the infinite length equalizer is approximately 28.9 dB and the worst case practical MER for the finite length (20 tap) equalizer is approximately 27.2 dB. This means, the implementation loss, measured as a difference in the worst MERs, is about 1.7 dB.

Furthermore, from Figure 6.9, it can be noted that the largest convergence time in the histogram is just under 150 symbols. Since the relative (additional) convergence time for a timing offset of 0.25 symbols is about 140 symbols. This means, the convergence time is almost doubled when the timing offset is 0.25 symbols.

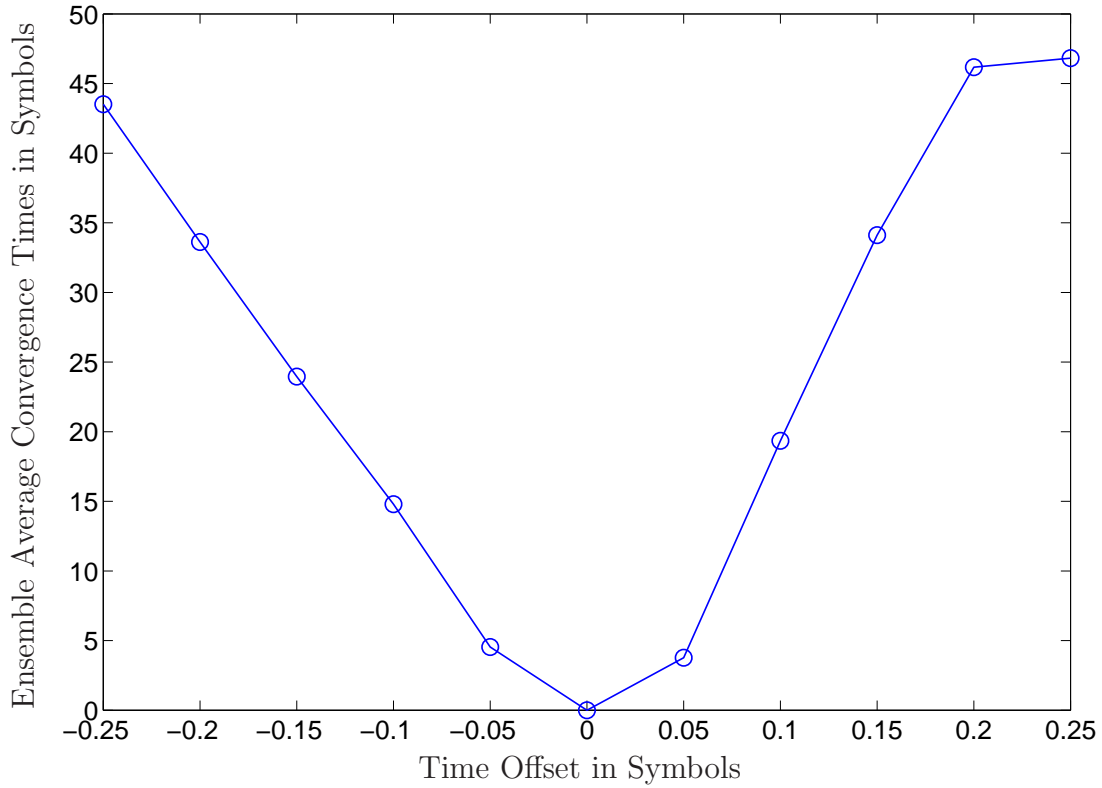


Figure 6.6 Ensemble average of normalized convergence times ($\mu\sigma_I = \sqrt{2}/64$) (20 tap equalizer)

To this point, the discussion of simulation results centered on the performance measures of the equalizer. Another useful measure, which is somewhat qualitative is a constellation plot. To assess the quality of the equalizer the constellation plot of the output is compared to the constellation plot of the input. To achieve a constellation plot of the decision variables at the output of the equalizer, the steady state value of the equalizer coefficients were used. The output is a result of convolving the data input sequence with the impulse response of the equalizer with its coefficients frozen when payload data begins.

Constellation plots for the equalizer input and output consisting of 8000 noise corrupted QPSK symbols were generated in MATLAB. These plots are shown in Figure 6.10 and 6.11.

The improvement in the constellation plot at output is evident from Figures 6.10 and

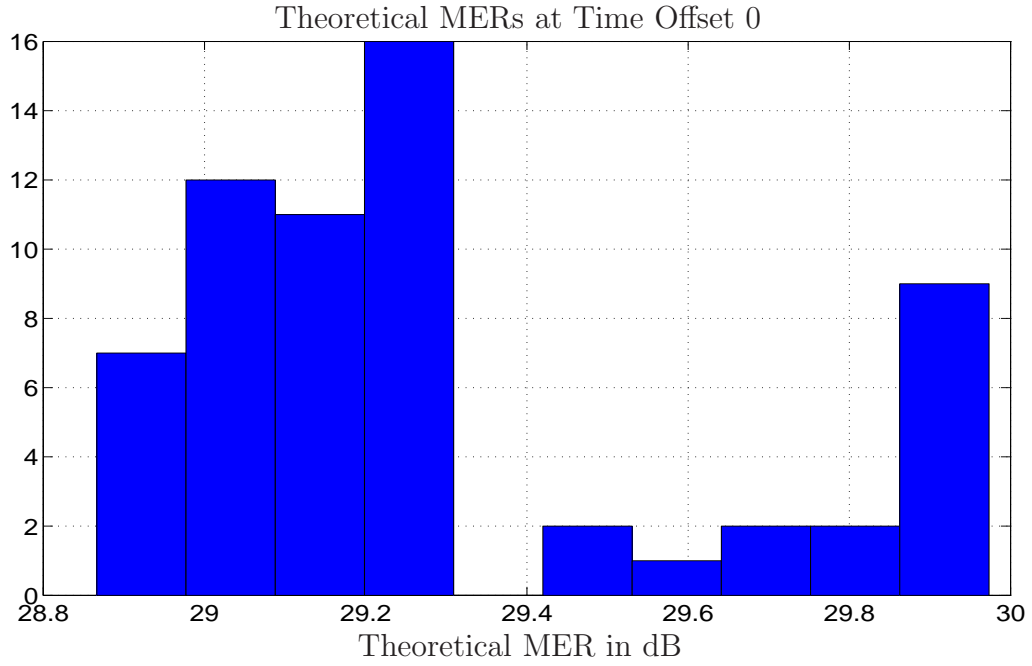


Figure 6.7 Histogram of theoretical MER (infinite length equalizer)

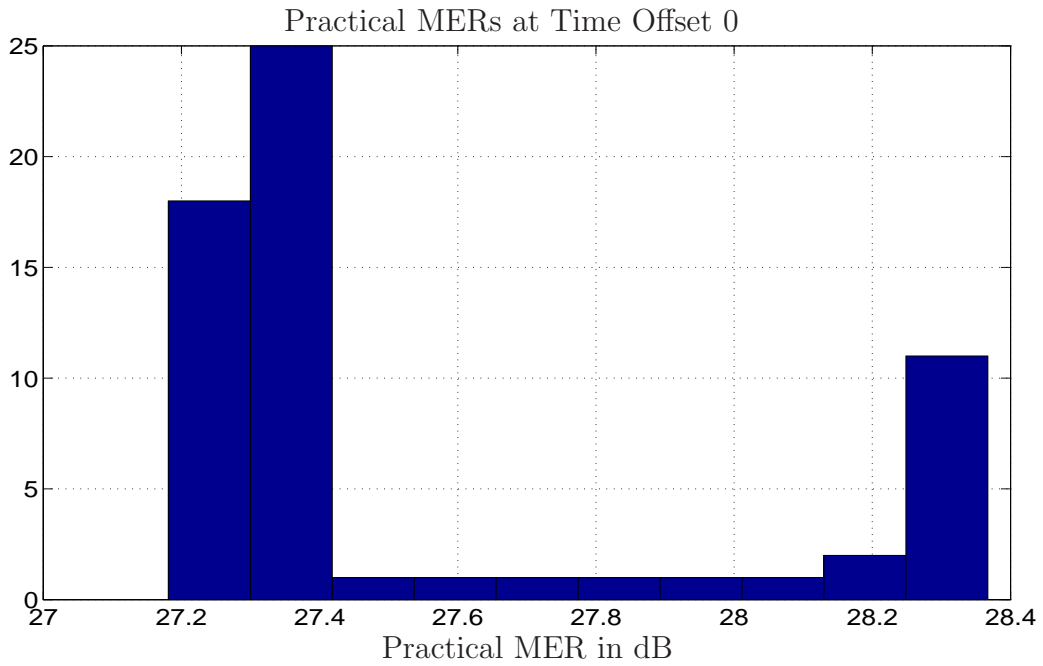


Figure 6.8 Histogram of practical MER (20 tap equalizer)

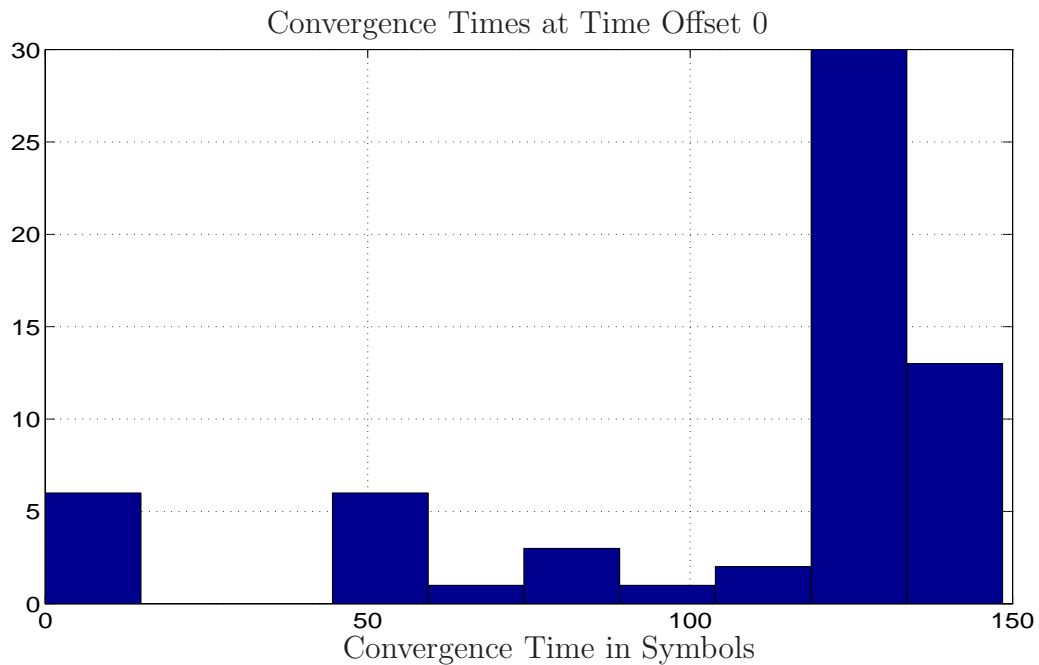


Figure 6.9 Histogram of convergence time ($\mu\sigma_I = \sqrt{2}/64$) (20 tap equalizer)

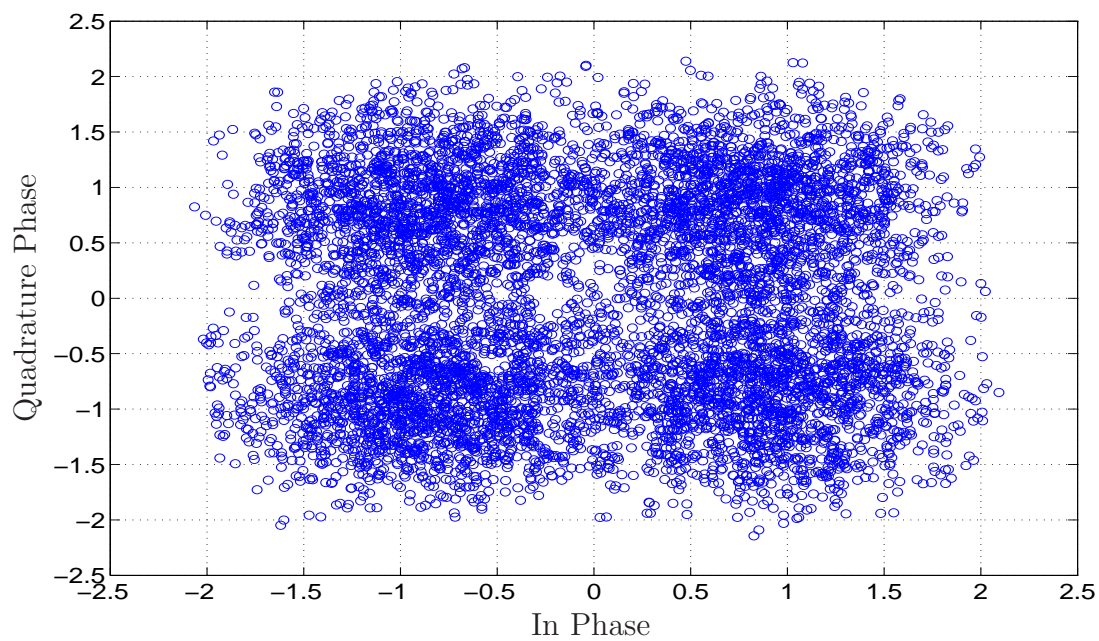


Figure 6.10 Constellation plot of the noise corrupted QPSK data at the input of the equalizer

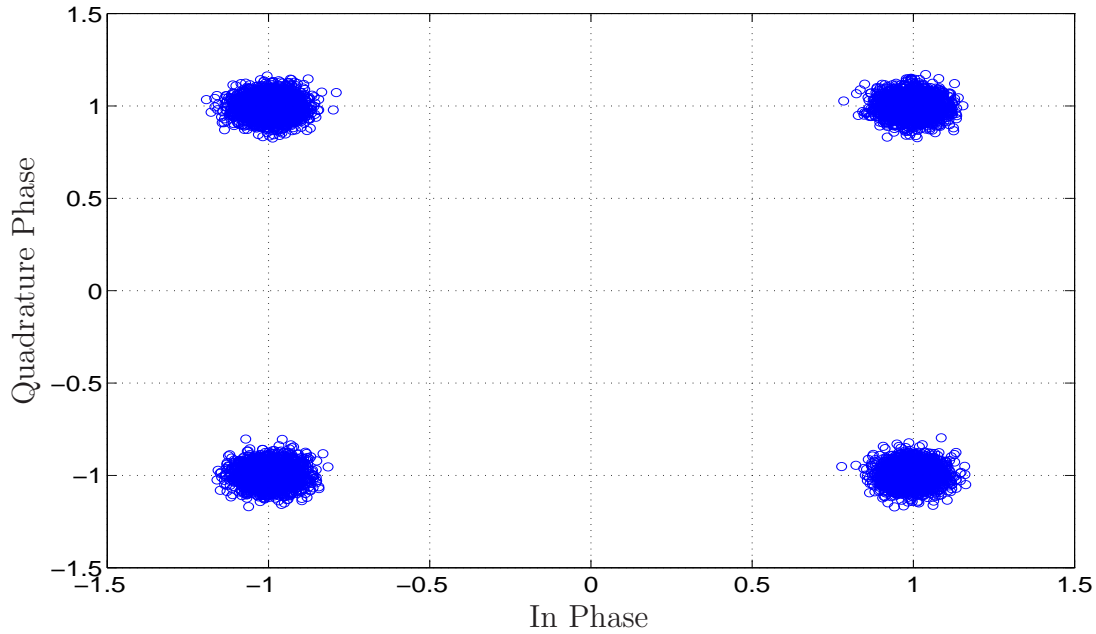


Figure 6.11 Constellation plot of the noise corrupted QPSK data at the output of the equalizer

6.11. For many of the points plotted in Figure 6.10, it is not obvious which point transmitted, whereas it is quite obvious in Figure 6.11.

For interest a MATLAB plot of the steady state value of the equalizer coefficients are shown in Figure 6.12. In this figure there are three curves. The solid line connecting markers 'o', is the magnitude of the complex steady state coefficients. The dashed line connecting markers ' Δ ', is the real part of the steady state coefficients. The dashed line connecting markers ' \star ', is the imaginary part of the steady state coefficients. Figure 6.12 shows that all 20 of the complex coefficients are non-zero in the steady state.

6.2 DSP Builder Simulation Results

Prior to implementing the equalizer in the FPGA, a simulink model of the cost optimized equalizer, which is in effect designed in DSP builder, needs to be cross checked. In order to make sure that the equalizer is working properly, several simulations were conducted.

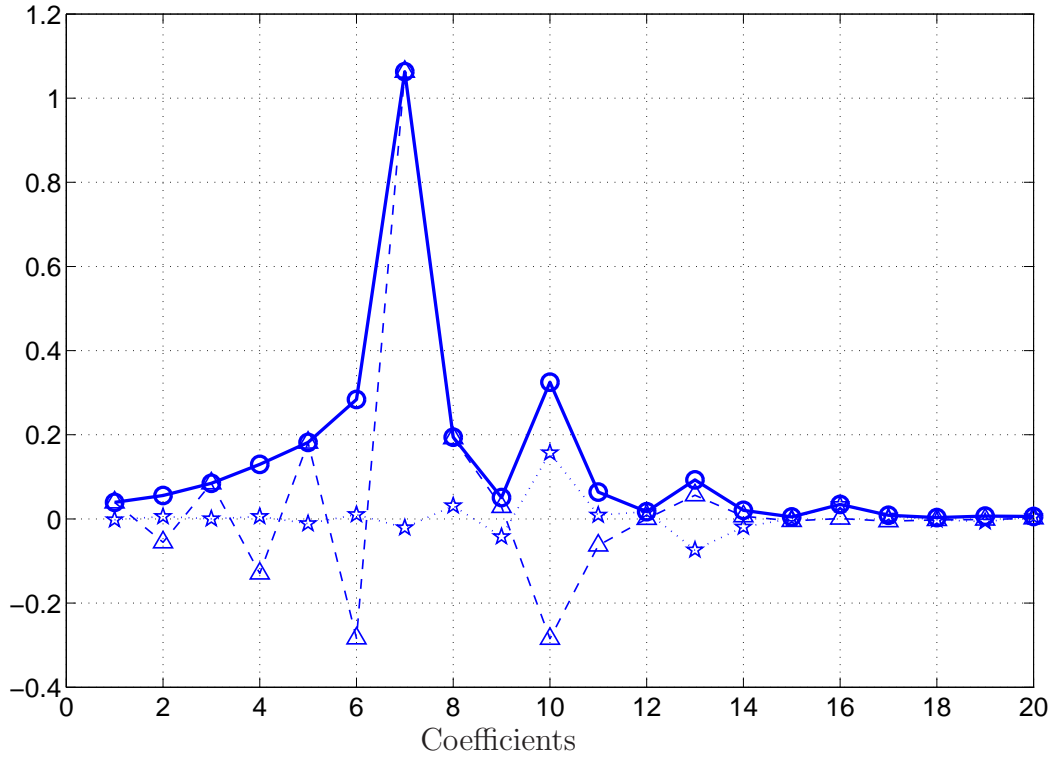


Figure 6.12 Equalizer coefficients at steady state

An efficient implementation requires two clocks. One clock, the symbol clock, runs at the symbol rate and the other, the system clock, runs at 20 times the symbol clock. The symbol clock is derived from the system clock. The input and output of the equalizer are updated at symbol rate, whereas the multiplying by the coefficient is accomplished using time shared multipliers that operate at system rate.

Technically the equalizer error should be measured with MSE. However, a feel for the equalizer error can be gained from a plot of the equalizer error as a function of symbols. Such a plot, from an equalizer constructed in DSP builder and plotted from MATLAB, is shown in Figure 6.13. Figure 6.13 shows the equalizer error trends downward until about 200 symbols at which time it appears to have reached steady state.

It is pointed out that Figure 6.13 must be plotted from MATLAB since DSP builder does not have block that computes the magnitude of the complex number. In order to

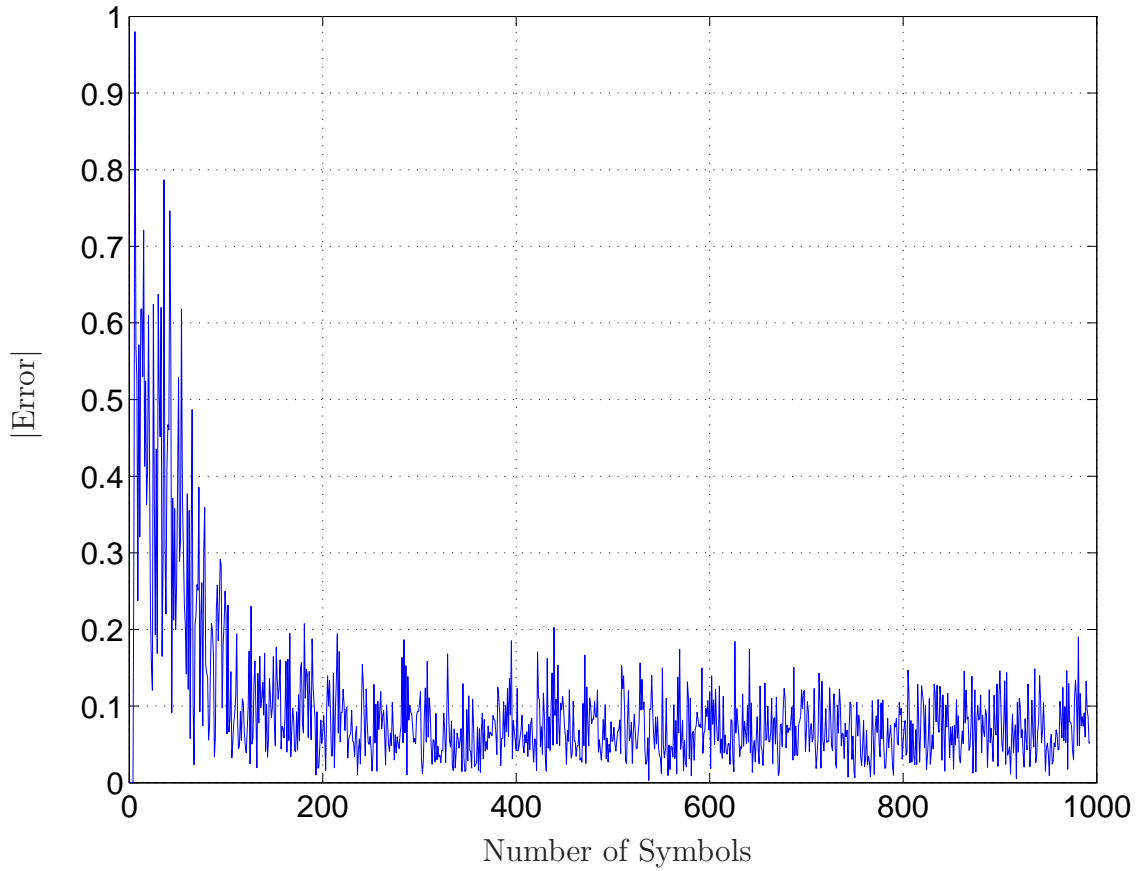


Figure 6.13 Equalizer error in DSP Builder

achieve Figure 6.13, first the cost optimized complex equalizer model was developed in DSP builder which is compatible with MATLAB simulink. The real and imaginary part of the complex error for a sequence of 1000 symbols is then extracted from the simulink model as a MATLAB vector. The magnitude of this vector was plotted in Figure 6.13.

Now, the quality of the equalizer will be demonstrated. The quality is demonstrated by comparing a constellation plot of the equalizer output with the equalizer input. The constellation of the equalizer input is given in Figure 6.10 and the constellation of equalizer output, which is plotted using the same procedure that was used to get Figure 6.13 is shown in Figure 6.14. A steady state plot of Figure 6.14 was generated for 800 symbols. Comparison of Figures 6.10 and 6.14 shows a marked improvement in grouping about the 4 QPSK points that were transmitted.

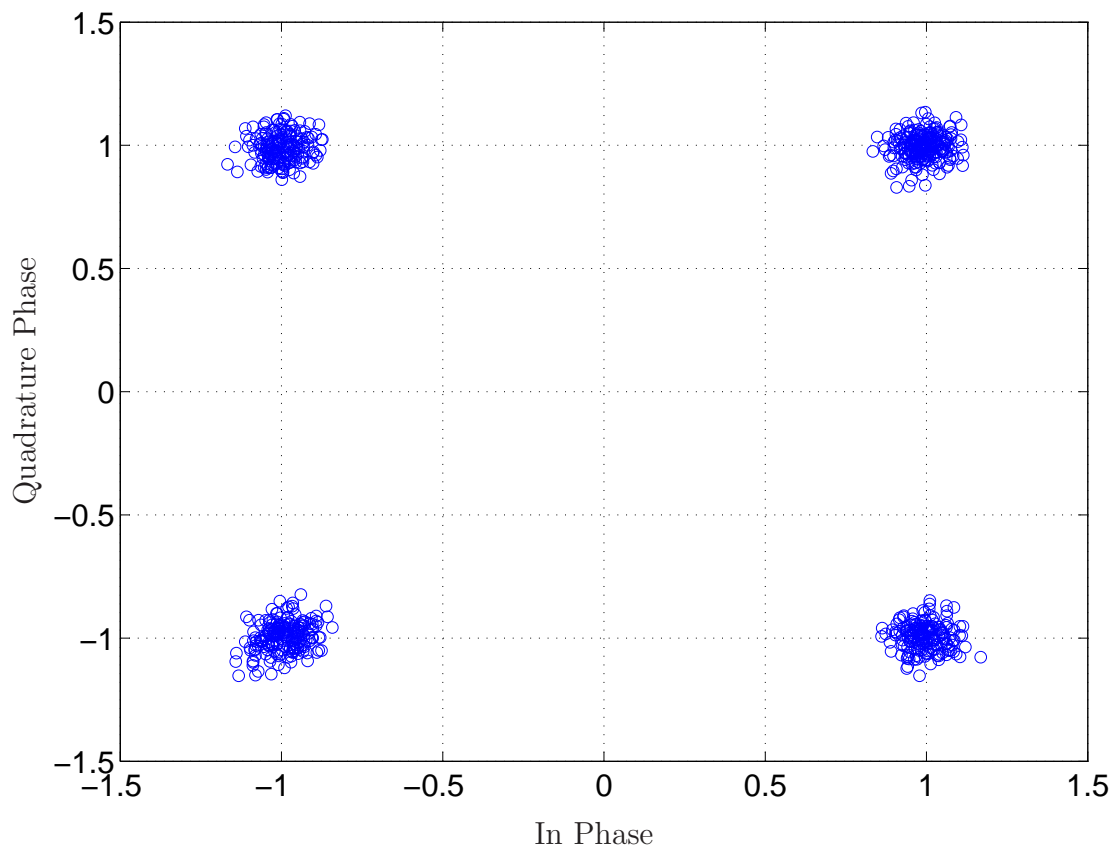


Figure 6.14 Constellation plot of the output of the equalizer

7. FPGA Based Results

The objective of the thesis is to implement a cost optimized equalizer in an FPGA. The equalizer of interest is for a DOCSIS upstream channel. At this point a detail analysis of a 20 tap cost optimized equalizer has been presented. The logical progression is to present the hardware implementation next.

Prior to analysing the performance of the equalizer implemented in hardware, a brief introduction to the equipment that was used to measure the performance needs to be added. In order to include all the equipments and an analysis of performance, this chapter is organized in two sections: equipment descriptions and equalizer performance.

7.1 Equipment Description

The equipment that was used to implement the equalizer in FPGA is listed below:

1. PC
2. DE4 Development and Education Board
3. AD/DA Data conversion Card

The equipment that was used to measure the performance of the equalizer as it was implemented in the FPGA is listed below:

1. Oscilloscope
2. Signal Analyzer

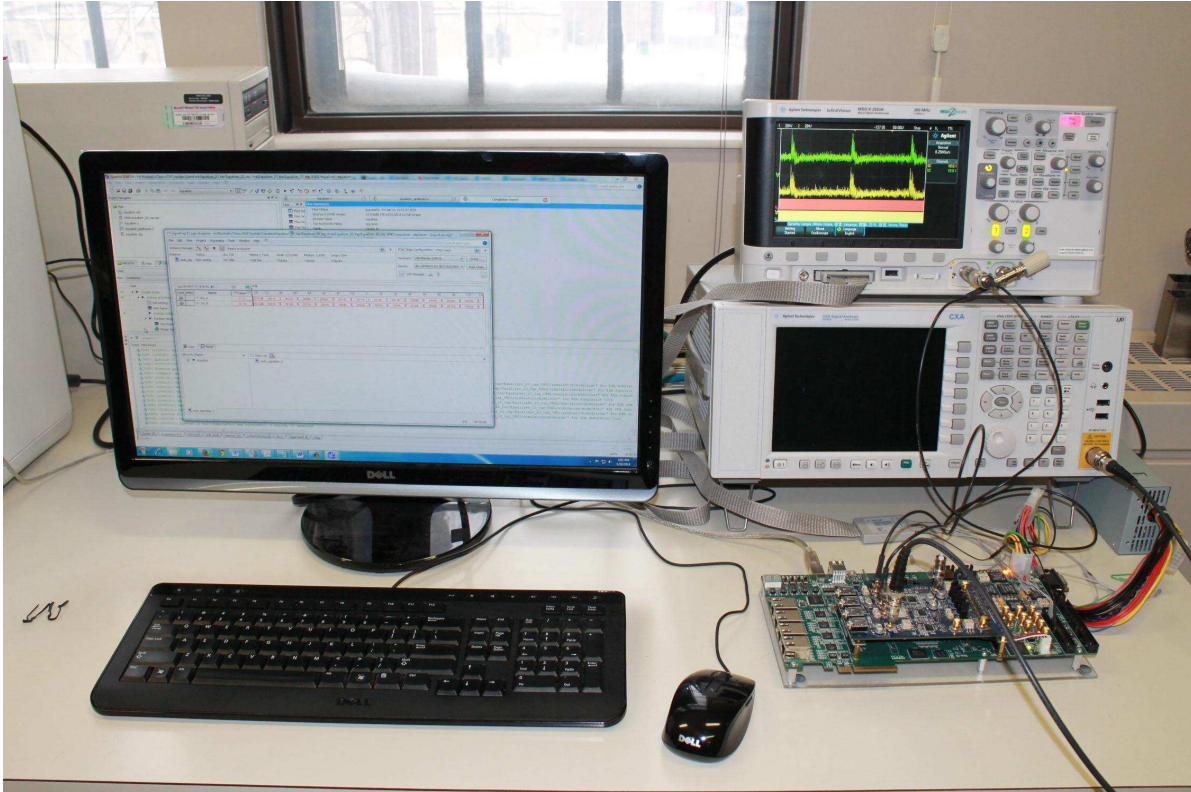


Figure 7.1 Hardware and measurement equipment

A picture that shows both the implementation and measurement equipment is presented in Figure 7.1.

The equipment needed to implement the equalizer is described below.

1. PC

The PC has the Windows 7 operating system. The PC is also equipped with other software, specifically, DSP Builder and Quartus 12.0 (64 bit), which was provided free of charge by Altera corporation.

2. DE4 Development and Education Board

“The DE4 Development Board provides the ideal hardware platform for system designs that demand high-performance, serial connectivity, and advanced memory interfacing. De-

veloped specifically to address the rapidly evolving requirements in many end markets for greater bandwidth, improved jitter performance, and lower power consumption. The DE4 is powered by the Stratix IV GX device and supported by industry-standard peripherals, connectors and interfaces that offer a rich set of features that is suitable for a wide range of compute-intensive applications”.¹

The hardware components on DE4 board that are key to this project are: Featured Device (FPGA) - Altera Stratix IV GX FPGA (EP4SGX230C2/EP4SGX530C2), Configuration Component - Built-in USB Blaster circuit for programming, Connectors - Two HSMC connectors and Two 40-pin expansion headers, General User input/output- 4 push-button and 4 slide switches and Clock System - On-board 100 MHz clock oscillator.

The FPGA that is used to implement the equalizer is Altera Stratix IV GX (EP4SGX530C2). The built-in USB Blaster circuit is used to program the FPGA from a GUI in Quartus II. The on-board 100 MHz oscillator is used for the system clock.

FPGA pins are directly connected to two HSMC connectors. Several daughter boards that connect to HSMC connectors are available from Terasic. One such daughter board provides high rate AD/DA data conversion. This card, which is called the mezzanine daughter card, was used in this project. The DAC output of the daughter card is connected to the measurement equipment.

3. AD/DA Data Conversion Card

The high speed mezzanine daughter card [34] is used for data conversion. For DSP applications, the daughter card can be used with the Altera DE4 boards that has HSMC connector. A high level block diagram showing the connections between DE4 board and daughter card is given in Figure 7.2.

In Figure 7.2, a block labelled FPGA Development Board (i.e. DE4 Board) is connected with a block labelled as Daughter Card via an HSMC connector. The daughter card has

¹The description inside the quotation mark has been taken from the DE4 user manual guide, which is available in Altera website.

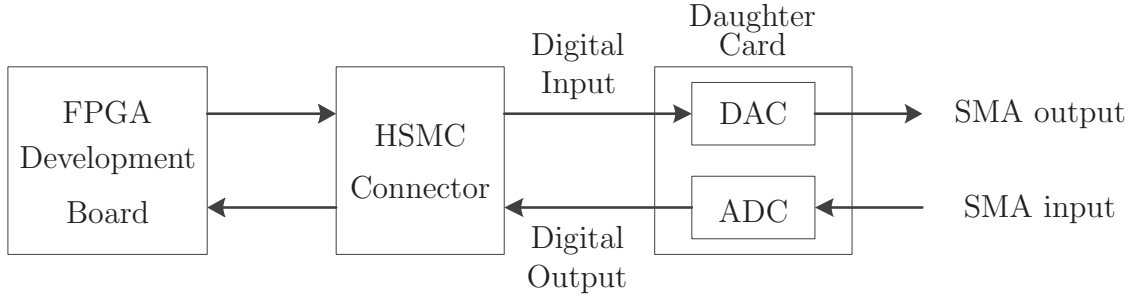


Figure 7.2 High level block diagram illustrating connections between the DE4 Board and the daughter card

two blocks labelled as DAC (Digital to Analog Converter) and ADC (Analog to Digital Converter). There are two inputs to the daughter card. One is a digital input, which is the digital output from the FPGA board. The other is the SMA input, which is an external analog input, connected to the ADC block. There are two outputs. One is a digital output. It is from the ADC block and goes to the FPGA board. The other is the SMA output. It is the analog output from the DAC block.

For this project only DAC function was used. The DAC chip, which has IC number DAC5672 [35], has two monolithic, dual-channel, 14 bit, high speed DACs.

The measurement equipments needed to display performance are described below.

1. Oscilloscope

The oscilloscope that is used for analysis is InfiniiVision MSO-X-2022A [36], which is a Mixed Signal Oscilloscope (MSO), produced by the Agilent technologies. The main features of this oscilloscope are: a bandwidth of 200 MHz, 8 digital channels, 2 analog channels, a sampling rate of 2 GSa/s and USB ports for printing and saving data.

2. Signal Analyzer

The signal analyzer that is used in this project is N9000A CXA Signal Analyzer, which is developed by Agilent technologies. It was used in swept spectrum mode, which makes it a swept spectrum analyzer. Some of the key features are: frequency range from 9 KHz to 3 GHz, Open Windows XP Operating System and a USB 2.0 port.

7.2 Performance of FPGA Implementation

This section presents results achieved with the Stratix IV FPGA on the Altera DE4 board. For the purpose of analysis, the equalizer implemented in the FPGA for the training mode, where only the preamble is processed. The training sequence was made sufficiently long to ensure the equalizer converged prior to reaching its end.

Performance of the Equalizer

The performance of the equalizer can be demonstrated in a variety of way. In this project, the performance is demonstrated by displaying equalizer output and equalizer error in the oscilloscope. For the purpose of comparison, the equalizer output is compared with the respective training data.

The equalizer output, training data and the equalizer error all are complex data. As mentioned previously, the sequence of complex data are processed in the FPGA as two sequences of real data. Therefore, two FPGA outputs are required to realize a complex output. The daughter card attached to DE4 board has two analog channels making both the real and imaginary output available to the oscilloscope.

The real part of the equalizer output and the real part of the training data are connected to the two DAC and displayed in the scope for comparison. The comparison between the real part of equalizer output and training data, for a sequence of approximately 13 symbols is shown in Figure 7.3.

The top waveform in Figure 7.3 is the real part of the training data and the bottom waveform is the real part of the equalizer output. Clearly, they look very much the same.

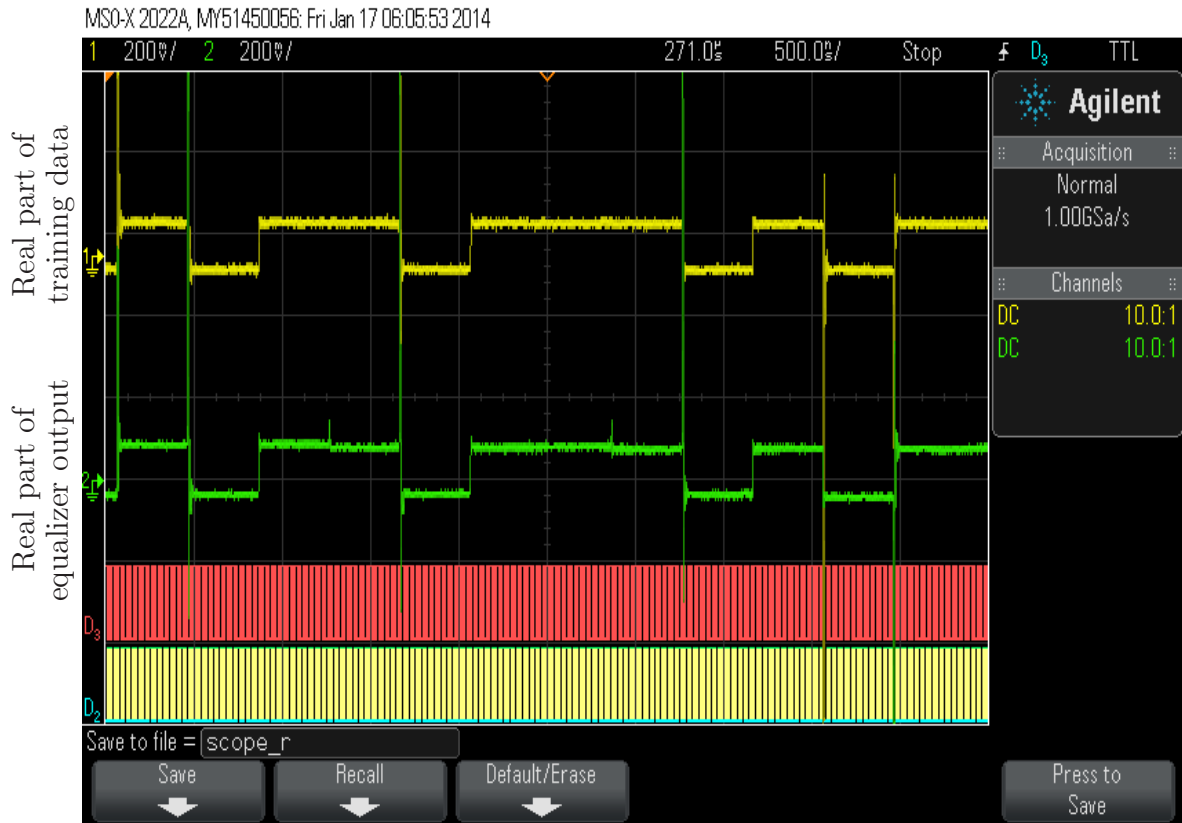


Figure 7.3 Comparison between the real part of equalizer output and the respective real part of the training data in the oscilloscope

Upon close examination of the real part of the equalizer output, the amplitude is seen to vary a bit.

Similarly, the comparison between the imaginary part of the equalizer output and the training data, for a sequence of approximately 13 symbols is shown in Figure 7.4.

The top waveform in Figure 7.4 is the imaginary part of the training data and the bottom waveform is the imaginary part of the equalizer output. The imaginary output of the equalizer has the variation as the real one.

For interest, the comparison between the real parts of the equalizer output and the training data are displayed in Quartus using data collected in Signal Tap. The Quartus

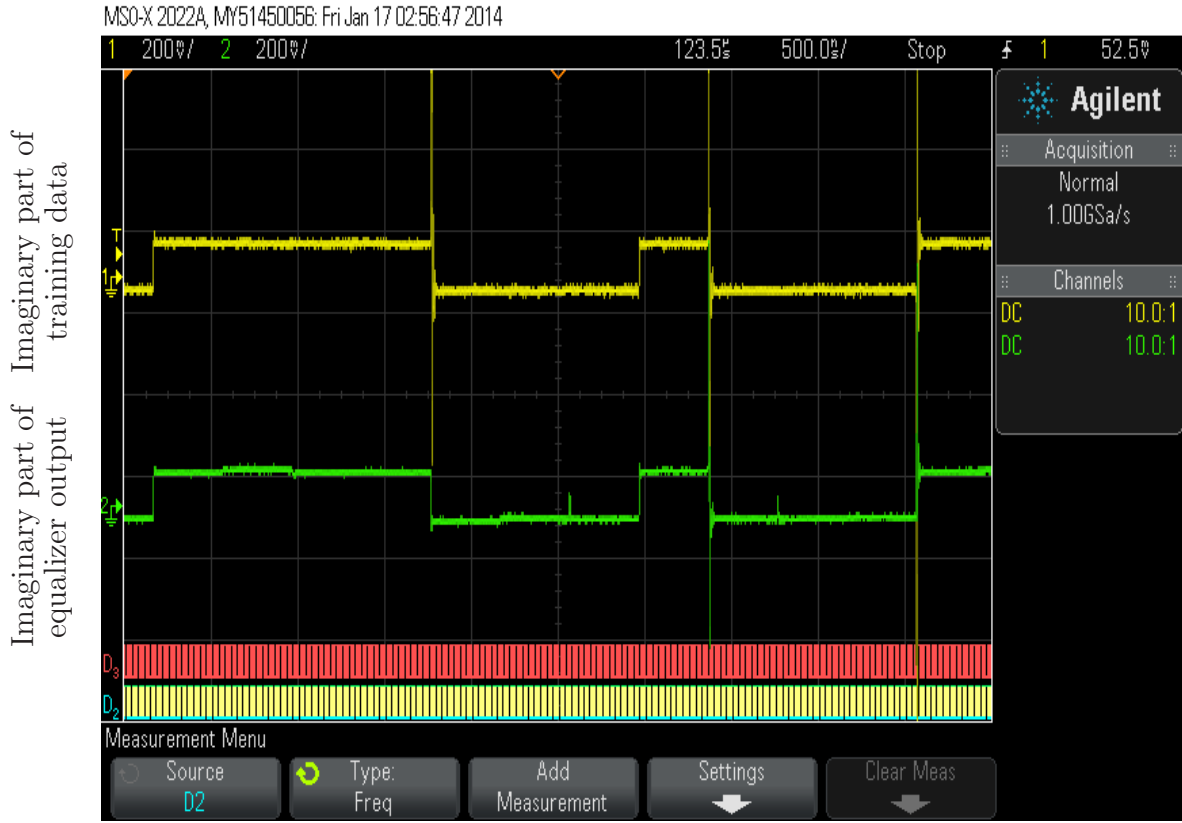


Figure 7.4 Comparison between the imaginary part of the equalizer output and the respective imaginary part of the training data in the oscilloscope

display is shown in Figure 7.5. This display, reconfirms that the equalizer output is very much the same as the transmitted training data.

The difference between the equalizer output and the training data (in training mode) is the equalizer error. For the purpose of demonstration, the absolute value of real and imaginary part of the complex equalizer error, for a sequence of 1000 symbols are shown in Figure 7.6.

The top error curve is the absolute value of the real part of the equalizer error and the bottom error curve is the absolute value of the imaginary part of the equalizer error. From both error curves, it appears that the equalizer error is exponentially decreasing until it reaches steady state.

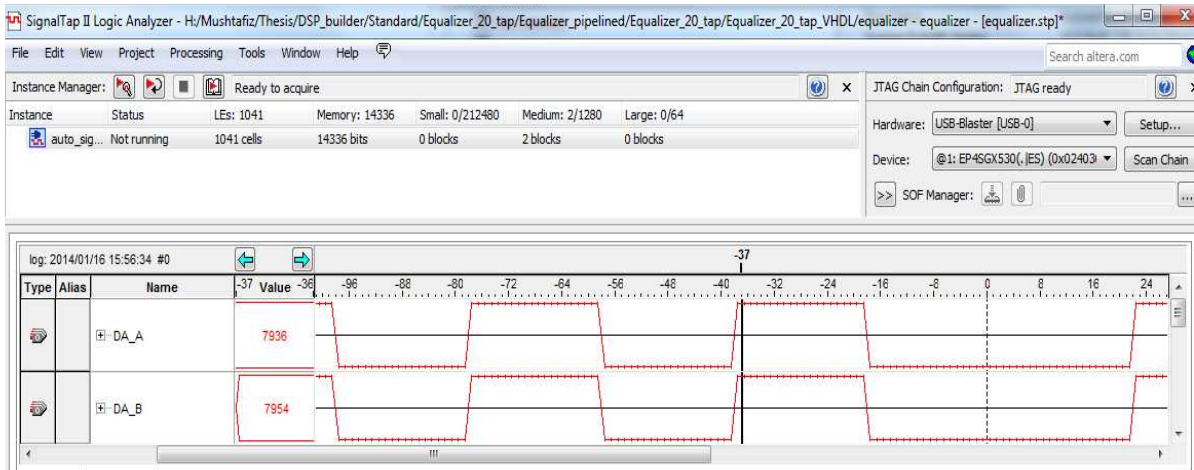


Figure 7.5 Comparison between the real part of equalizer output and the corresponding training data in Signal Tap

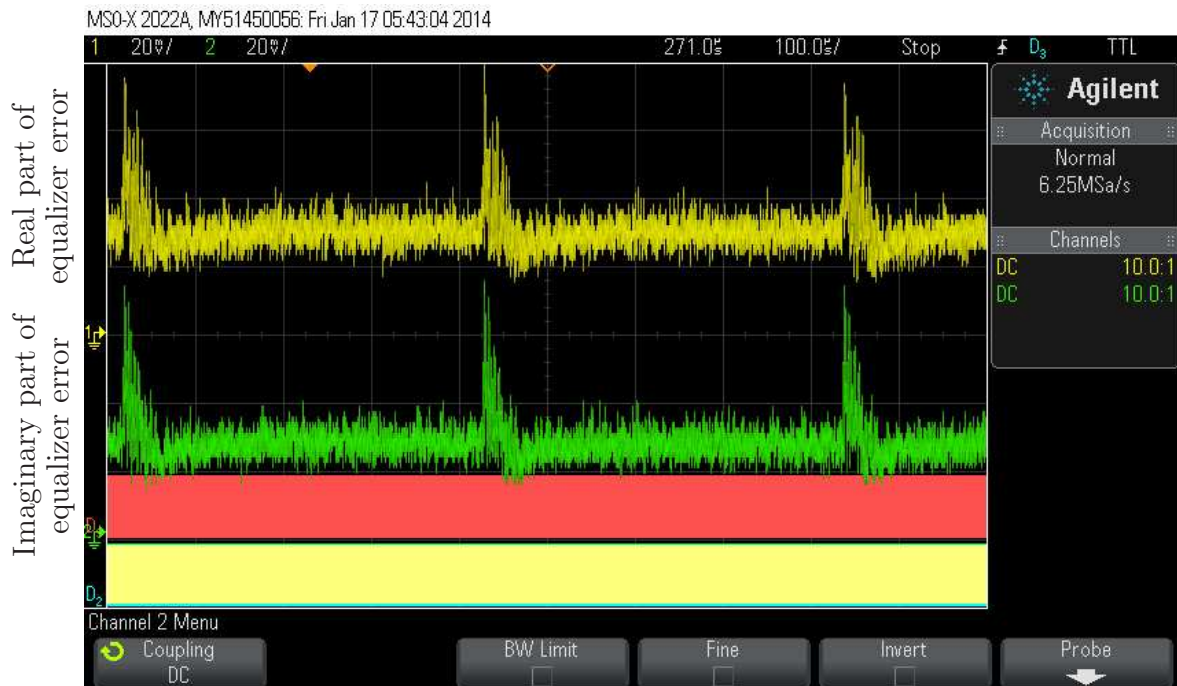


Figure 7.6 Equalizer error displayed in the oscilloscope

8. Summary and Conclusions

The design of a cost optimized equalizer for a DOCSIS upstream channel is challenging. Impairments in the channel and constraints imposed by the standards must be overcome. In addition, optimizing parameters like step size and equalizer length involves a trade-off between MER performance and convergence time.

Once a decision is made for the parameters of the equalizer, the equalizer must be implemented in the FPGA in a cost effective way. The largest cost component is that of the multipliers. The implementation of the equalizer requires an extra effort to reduce the number of multipliers and use a minimum amount of FPGA logic elements.

In this concluding chapter, a summary of the contribution and achievements of the research work is organized in three sections: contributions towards establishing parameter-performance relationship for the equalizer, contributions towards critical issues that affect equalizer performance and contribution towards optimal implementation in terms of cost.

8.1 Contribution towards establishing parameters-performance relationship for the equalizer

One of the contributions of this research is the finding the relationship between parameters and the performance of the equalizer. The main parameters are the length of the equalizer and the step size, and the performance criteria are MER and convergence time. This research determined the parameter-performance relationship through a detailed MATLAB simulation.

A MATLAB simulation showed that the performance of the equalizer depends on the

length of the equalizer. Prior to choosing a length for the equalizer, a comparative analysis through MATLAB simulation was conducted for the standard 24 length and proposed 20 length equalizers. After a carefully designed simulation for a DOCSIS upstream channel, it was found that the performance in terms of MER, for a length 20 equalizer outperforms a length 24 equalizer. It was also found that there was no difference in convergence time. Based on this finding a length 20 equalizer was used in subsequent performance analysis.

The step size of the equalizer also has an influence on the performance of the equalizer. The MATLAB simulation showed that with the decrease of step size, the MER performance gets better. On the other hand, the convergence time gets longer (time measured in symbol). Since shorter is better for convergence time and higher is better for MER, this set up a trade-off between MER and convergence time. To provide data to aid with a trade-off decision, a carefully designed MATLAB simulation was conducted for a worst case channel with a length 20 equalizer with 3 step sizes ($1/32$, $1/64$ and $1/128$)¹. It was found that a step size of $1/64$ provide a convergence time 167 symbols, which is below 200 symbols, and MER of 25.18 dB. This compares to the convergence time and MER for step sizes $1/32$ and $1/128$ are 104 symbols, 22.56 dB and 316 symbols, 26.13 dB, respectively. In order to limit the convergence time within 200 symbols, the step size should be either $1/32$ or $1/64$. With step size $1/32$, convergence time is shorter, which is better, but the MER goes down. Thus, the decision that a step size of $1/64$ is probably the best choice.

8.2 Contributions towards critical issues that affects equalizer performance

The critical issues that influence the performance of the equalizer are the characteristics of the channel and imperfect timing recovery. This research analysed both of the issues based on the MER and the convergence time of the equalizer.

The impulse response of the channel has a direct affect on the performance of the equalizer. The MATLAB simulation showed that the performance varies with the delays in the

¹Choosing a step size of negative power of two has a cost advantage in hardware as they do not require the use of a multiplier.

echoes with respect to the main path. It was found that, for a single echo channel, the MER reaches a minimum when the delay in the echo is about 1.2 symbols. Not surprisingly for this echo, the convergence time is at its maximum 150 symbols. This establishes that the worst case single echo channel has the maximum strength (0.3162 on main path) echo with a delay of 1.2 symbols.

The effect of timing error on equalizer performance is one of the useful finding in this research. The effect of timing error has been analysed on an infinite length equalizer as well as on a finite length (20 tap) equalizer.

In order to see the effect of timing error on an infinite length equalizer, an equation was set up to calculate MER. A MATLAB simulation was used to generate the MER versus time offset curves for 62 channels. Each curve consisted of 11 simply connected equally spaced points, where each point represents a value of timing offset ranging from -0.25 to 0.25 symbols. These 62 MER curves were normalized with respect to the MER for a timing offset of zero. At the end, an ensemble average of normalized MER curves for an infinite length equalizer was calculated and plotted in MATLAB. The ensemble average showed that the best MER performance is obtained when the matched filter is sampled at the time the eye is maximally open.

Similarly, for a length 20 equalizer, an ensemble average based MER curve was plotted in MATLAB for the same 62 channels. It was found that the best MER performance is obtained when the matched filter is sampled at the time the eye is maximally open. In addition, an ensemble average based convergence time curve for the same 62 channel was also plotted in MATLAB for the length 20 equalizer. This curve showed that the best convergence time is obtained when the matched filter is sampled at the time the eye is maximally open.

It was noted in a length 20 equalizer for a timing offset of less than 0.1 symbols, the worst case sensitivity of MER to timing offset is about 3dB/symbol. It was also noted for the same equalizer, for timing offset of less than 0.1 symbols, the worst case sensitivity of convergence time to timing offset is about 400 symbols/symbol.

8.3 Contributions towards cost optimization and implementation

One of the goals of this research was to minimize the implementation cost. Since the most expensive logic element in an FPGA is the multiplier, the goal became reducing the number of multipliers needed to implement the equalizer. That was accomplished in part by time sharing multiplier to accomplish 20 multipliers per symbol.

It was showed that, without time sharing, the 20 tap symbol-spaced complex equalizer would have taken 160 multipliers, but with time sharing the same equalizer was modelled by using only 8 multipliers.

Once the design of the cost optimized model was finalized, then it was developed in DSP Builder, which is in effect a special application of MATLAB simulink. The performance of the cost optimized equalizer was verified by comparing output of the circuit constructed with DSP Builder with the MATLAB results.

The final step of this research work was to implement the equalizer in FPGA and verify through demonstration. To do that, the cost optimized equalizer was successfully implemented in a Stratix IV FPGA board and for the purpose of verification, the equalizer output and equalizer error were verified in the oscilloscope.

References

- [1] E. Smith, “The emergence of catv: A look at the evolution of a revolution,” *Proceedings of the IEEE*, vol. 58, no. 7, pp. 967–982, 1970.
- [2] K. S. University, “History of cable tv,” July 2010.
- [3] CableLabs, *Data Over Cable Service Interface Specifications DOCSIS 3.0 - Physical Layer Specification*. cm-sp-phyv3.0-i11-130808 ed., Aug. 2013.
- [4] CableLabs, *Data Over Cable Service Interface Specifications DOCSIS 1.0 - Radio Frequency Specification*. sp-ossi-rfi-c01-011119 ed., 1997.
- [5] M. Rice, *Digital Communications: A Discrete - Time Approach*. Pearson Prentice Hall, 2009.
- [6] N. J. Fliege, *Multirate Digital Signal Processing*. John Wiley & Sons, first ed., 1994.
- [7] S. Kalle, M. Choudhury, F. Bui, and D. Dodds, “Performance of unique word timing offset estimator,” in *Electrical and Computer Engineering (CCECE), 2013 26th Annual IEEE Canadian Conference on*, (Regina, Canada), pp. 1–4, 2013.
- [8] E. Pelet and J. Salt, “On economical timing-error detectors for qam receivers,” *Communications, IET*, vol. 1, no. 4, pp. 618–622, 2007.
- [9] B. Berscheid, *FPGA-Based DOCSIS Upstream Demodulation*. PhD thesis, University of Saskatchewan, Saskatoon, Saskatchewan, Canada, July 2011.
- [10] S. O. Kasap, *Principles of electrical engineering materials and devices*. McGraw-Hill, 2000.
- [11] H. H. Nguyen and E. Shwedyk, *A First Course in Digital Communications*. Cambridge University Press, 2009.
- [12] U. Mengali and A. N. D. Andrea, *Synchronization Techniques for Digital Receivers*. Plenum Press, 1997.

- [13] E. Pelet, J. Salt, and D. Fast, "Timing and carrier acquisition in upstream catv channels," in *IEEE Industrial Electronics, IECON 2006 - 32nd Annual Conference on*, pp. 3267–3272, Nov 2006.
- [14] Y. Kim, H. Ha, J. Lee, W. Oh, W. Kim, E. Lee, and Y.-J. Song, "Upstream channel synchronization for cable modem system," in *Advanced Communication Technology, The 9th International Conference on*, vol. 3, pp. 1864–1867, Feb 2007.
- [15] S. Haykin, *Adaptive Filter Theory*. Prentice Hall, 2002.
- [16] S. Qureshi, "Adaptive equalization," *Proceedings of the IEEE*, vol. 73, pp. 1349–1387, Sept 1985.
- [17] A. H. Sayed, *Fundamentals of Adaptive Filtering*. Hoboken, N.J. : J. Wiley & Sons : IEEE Press, 2003.
- [18] J. G. Proakis, *Digital Communications*. McGraw Hill Higher Education, 2000.
- [19] B. Sklar, *Digital Communications: Fundamentals and Applications*. Prentice Hall, 2001.
- [20] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Process*. Tata McGraw Hill, 2002.
- [21] S. S. Soliman and M. D. Srinath, *Continuous and Discrete Signals and Systems*. Prentice Hall of India, 2007.
- [22] T. Wolf, C. Lemonds, and A. Gatherer, "Low complexity equalization for cable modems," in *Circuits and Systems, 1998. ISCAS '98. Proceedings of the 1998 IEEE International Symposium on*, vol. 5, pp. 522–524 vol.5, May 1998.
- [23] A. Gatherer, "The effect of microreflections on decision feedback equalization," *Signal Processing, IEEE Transactions on*, vol. 45, pp. 228–231, Jan 1997.
- [24] Y. Kim, Y. Kim, Y. Lee, W. Oh, and W. Kim, "Synchronization and channel equalization for upstream cable modem," in *Consumer Electronics, 2008. ISCE 2008. IEEE International Symposium on*, pp. 1–4, April 2008.

- [25] Z. Andalibi, B. Berscheid, E. Salt, and H. Nguyen, “A fast-converging equalizer for upstream dcofs channels,” *Broadcasting, IEEE Transactions on*, vol. 56, pp. 311–320, Sept 2010.
- [26] G. L. Stuber, *Principles of Mobile Communication*. Kluwer Academic Publishers, 2004.
- [27] R. Merched and A. Sayed, “Extended fast fixed-order rls adaptive filters,” *Signal Processing, IEEE Transactions on*, vol. 49, pp. 3015–3031, Dec 2001.
- [28] G. Carayannis, D. Manolakis, and N. Kalouptsidis, “A fast sequential algorithm for least-squares filtering and prediction,” *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 31, pp. 1394–1402, Dec 1983.
- [29] J. Cioffi and T. Kailath, “Fast, recursive-least-squares transversal filters for adaptive filtering,” *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 32, pp. 304–337, Apr 1984.
- [30] V. Betz, “Fpga challenges and opportunities at 40nm and beyond,” in *Field Programmable Logic and Applications, 2009. FPL 2009. International Conference on*, pp. 4–4, Aug 2009.
- [31] Z. D. B.P. Lathi, *Modern Digital and Analog Communication Systems*. Oxford University Press, fourth ed., 2009.
- [32] M. Choudhury, E. Salt, and D. Dodds, “Effect of timing error on equalizer performance,” in *Electrical and Computer Engineering (CCECE), 2013 26th Annual IEEE Canadian Conference on*, pp. 1–5, May 2013.
- [33] R. Boucher and J. Hassab, “Analysis of discrete implementation of generalized cross correlator,” *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 29, pp. 609 – 611, June 1981.
- [34] Terasic, *Data Conversion HSMC Reference Manual*. Terasic, Altera Corporation, Mar. 2008.
- [35] Texas Instruments, *DAC5672*, Dec. 2010.

- [36] Agilent Technologies, *Agilent InfiniiVision 2000 X-Series Oscilloscopes*, June 2011.