A New Line Code for a Digital Communication System

by

Adib Shahabi

Submitted in partial fulfilment of the requirements
for the degree of Master of Applied Science

at

Dalhousie University
Halifax, Nova Scotia
August 2019

To my wife Shahideh

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Acoustic sensor arrays can be wired to a central processor, responsible for logging the payload. To avoid several lines and connectors, a single transmission line is preferred to transfer all the data between the multiple sensors. In this work, a low-complexity communication link is developed to enable communication between multiple sensors, and one master controller.

The objective of this project is to optimize the point-to-point link performance using specific front-end requirements. For this purpose, data recovery of the multiple nodes will be required, and the distortion of the signal due to cable bandwidth limitations will be mitigated using line coding.

In digital communications systems, digital data represented as ’0’ and ‘1’, are abstract quantities and need to be converted into electrical waveforms for effective transmission or storage. How to perform such a conversion is generally governed by many factors, of which the most important one is the available transmission bandwidth of the communication channel. In digital communication systems, a problem that is often faced is of the limited transmission bandwidth which can take the form of different components including narrowband front-ends, coupling transformers, as well as long cables.

Various line codes have been developed to shape the spectrum of the transmit signal. This thesis proposes a new line code, the Quadrature Code (or *Quad* Code) that increases the transmission rate and maintains reliability. Its *PSD*, *SNR*, eye pattern and bit error rate are compared to other standard line codes including the fundamental Non-Return-to-Zero (*NRZ*), *Manchester*, delay modulation, and *8B/10B*.

LIST OF ABBREVIATIONS USED

AC	Alternating Current
BER	Bit Error Rate
dB	decibel
DC	direct current
DM	Delay Modulation
DSP	Digital Signal Processing
FPGA	Field-Programmable Gate Array
ISI	Inter-Symbol Interference
LFSR	Linear Feedback Shift Register
NRZ	Non-return-to-zero
NRZ-L	Non-return-to-zero-Level
PN	Pseudo-Noise
PSD	Power Spectral Density
RC	Resistance-Capacitance
RD	Running Disparity
RL	Run-Length
SNR	Signal to Noise Ratio
S-P	Scattering Parameter
VHDL	VHSIC Hardware Description Language

ACKNOWLEDGEMENTS

I would like to express my special appreciation and thanks to my advisor, Dr. Jean-François Bousquet, you have been a tremendous mentor for me. I would like to thank you for encouraging my research and for allowing me to grow as a research scientist. Your advice on both research as well as on my career has been priceless.

I would like to thank Jay Abel to Ocean Sonics for his invaluable insights.

At the end, I would like to express my love to my dear wife, Shahideh.

CHAPTER 1 INTRODUCTION

1.1 BACKGROUND

This proposed project is intended to develop a custom point-to-point link between a master hub, and multiple slave sensors for the deployment of underwater acoustic sensor arrays with multiple remote nodes. A low-complexity communication link will be defined to minimize overhead and power consumption.

The primary constraint in this project is the power consumption at the remote nodes. To limit the power consumption at the front-end, switching mode buffer amplifiers are used to regenerate the signal. As such, binary waveforms with a constant envelope are transmitted to represent the digital information. The overall system throughput is evaluated to increase the number of remote nodes present on the network and constrained on the channel limited bandwidth.

A differential signal transmission technique has been proposed, particularly to reduce sensitivity to the ground noise. The transmit signal is *AC* coupled to the line using a transformer at the output of the power amplifier and the *AC* coupling will allow the transmission of the power signal on the line.

In this project, a digital communication link physical layer will be modelled and implemented on low power consumption, low complexity embedded processors.

This project is intended to enable a cabled network between multiple acoustic sensor nodes, the nodes being separated by approximately 35 meters along a tether to a central hub. The data is routed on a low-cost differential twisted copper line. During the forward link, the

hub is used to reconfigure and synchronize the sensor clocks, while on the forward link, the sensor payload is routed to the hub.

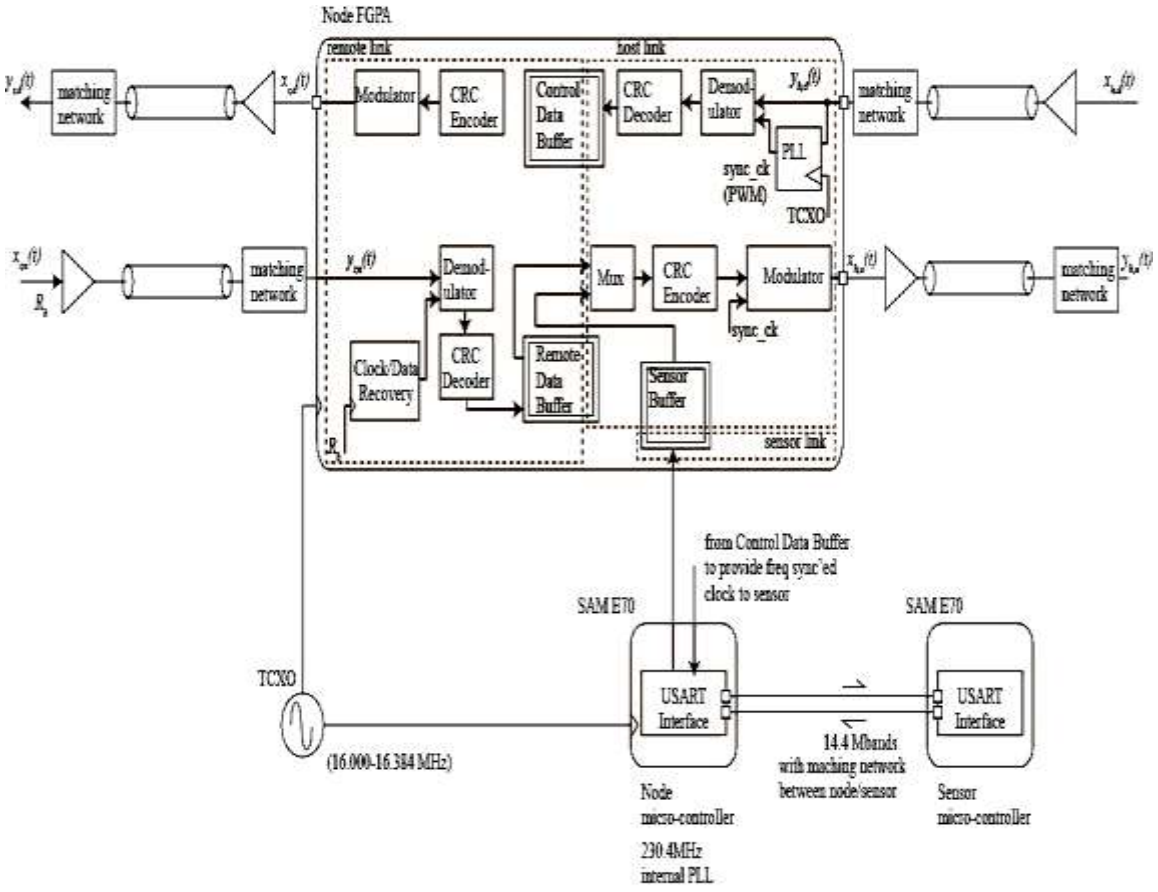


Figure 1.1 Block diagram of Bi-directional Link.

There a few alternative serial communication systems commercially available, including the transmission control protocol (TCP). In [1], the use of TCP has been recognized to be inefficient to gather sensor data from multiple remote nodes due to the necessity to re-collect the entire set of data in case of packet loss. As an alternative, a USB hub can be used to interface smart sensor nodes to an Ethernet gateway as described in [2]. Although USB 2.0 describes a 480 Mbps transfer rate, it cannot be expected to provide a reliable 35-

meter link. The serial protocols RS-422 and RS-485 can provide Mbps communication between a master a multiple slave over a range close to 100 meters, but the short frame format reduces the spectral efficiency of the communication link.

A custom TDMA scheme developed in a previous NSERC ENGAGE grant will be adopted here to allow continuous streaming of remote sensor data. The packet duration and handshaking mechanism between the hub and the sensors has been customized to stream using a 14.4-MHz clock. The objective of this project is to focus on the physical layer implementation to meet the network requirements.

To synchronize all the nodes, the reference clock will be embedded in the payload, instead of being sent on a separate line. As such, a clock recovery circuit is required at the receiver. To enable a highly accurate clock and carrier recovery, a decision directed digital phase detector will be used [3]. Instead of using a phased locked loop [4], the detector will be applied in a feedforward synchronization configuration to improve clock recovery accuracy. Note that in [5], a clock and data recovery is implemented for a TDMA network on fiber optic. Also, in [6], the system is enhanced with coding techniques to improve the performance. Instead in our proposed work, joint data and clock recovery algorithm will be proposed to optimize reliability such as described in [7].

In this project, the high-bandwidth streaming from multiple competing nodes to a central hub imposes a significant constraint on the bandwidth of the cable. As such, the objective of this project is to focus on the physical layer implementation to meet the network requirements.

To reduce the *DC* offset at the transmitter, *8b/10b* coders exist, but increase overhead. Alternative signatures with lower coding ratios as proposed in [8] will be considered.

To synchronize all the nodes, the reference clock is embedded in the payload, instead of being sent on a separate line.

Because the high-speed signal will travel a significant distance over a cable that is band limited, it is important to develop a novel coding technique to preserve its integrity. In this work, a high voltage signal is *AC* coupled into the line over a broad frequency range and is transmitted differentially. A behavioral model of the communication link includes the representation of the transmit waveform at passband. The model includes impairments due to 1) the transmit front-end electronics, 2) the inter-symbol interference as it travels along the transmission line.

To demonstrate the performance of the modelled link, it will be implemented on a field programmable gate array (*FPGA*) digital signal processing platform. Standard figures of merit, such as the eye opening, signal to noise ratio (*SNR*) and bit error rate (*BER*) will be used to quantify the system performance.

1.2 THESIS SYNOPSIS

In a wired communication system, the maximum link throughput depends on the available link bandwidth. The limited bandwidth of the link distorts the signal. In this project we model the cable using an RC lumped equivalent, where the capacitor and range of the cable are controlled to determine the bandwidth of the cable and its effect on the eye pattern, Additionally, modeling the front-end transformer coupling introduces a zero at low frequency and the effect on the eye pattern will be analyzed.

In this work, different line code techniques are studied to can shape the spectrum of the signal. The transmission medium must have a bandwidth wide enough to pass signals with a finite bandwidth. In order to select the coding technique for data communication in limited bandwidth, five basic line codes techniques are compared, including Non-return-to-zero (*NRZ*), delay modulation (*Miller*), *Manchester* (split phase), *8B\10B,* and the proposed *Quad* code.

Specifically, the codes are developed to eliminate long sequences of continuous bits with the same value, and thus to reduce the spectrum content at low frequency. An analysis of the codes will provide a relationship between the length of the sequence and the spectrum of the signal.

In this project, a primary objective is to analyze and compare different codes for a band limited link and to select the one which optimizes the link reliability.

For implementation and measurements, three codes have chosen and programed using vhdl on *FPGA* board which is interfaced to the transmission media, including the transformer at the transmitter, as well as the cable.

At the receiver, for data recovery quadrant phase shifters of clocks are implemented to detect the binary information and to compare the data at the receiver by analyzing the of eye pattern, as well as the *PSD*, and by establishing different figures of merit, including the *SNR*, and the *BER*.

1.3 ORGANIZATION OF THE THESIS

This thesis is organized as follows:

Chapter 1 provides the research background, the problem statement and current state-of-the-art.

Chapter 2 presents the modeling and simulation of an *RC* cable and a transformer as communication link. Then comparisons are used to validate the cable model.

Chapter 3 describes and analyzes different line codes. Then, an algorithm to implement the new Quad code is developed. Furthermore, the code characteristics, such as PSD and SNR at the receiver are compared to that of other codes.

Chapter 4 depicts the implementation of codes discussed in this work using *vhdl* on a *Xilinx Artix 7* platform.

Chapter 5 provides a conclusion and proposes insights on future work.

CHAPTER 2 MODELLING THE LINE CHANNEL

This chapter describes the model of the link between the transmitter and receiver and how it impacts the transmission of an uncoded signal at the receiver. Specifically, in Section 2.1 the cable model is described; then in Section 2.2, the model of the transformer at the transmitter front-end is described; finally, in Section 2.5, measurements of the cable are used to validate the cable model.

2.1 Cable Modelling

The objective of this section is to represent the cable using a transmission line. First, the cable is modelled as a simple *RC* cable, and its effect on a Pseudo-Random Bit Sequence (*PRBS*) is evaluated as shown in Figure 2.1. Note that the *PRBS* is implemented using an *LFSR* [9].

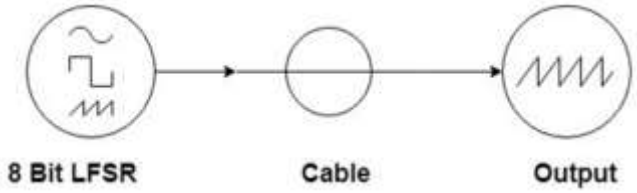


Figure 2.1 Transmission Line.

The effect of the transmission line on the signal at the receiver is analyzed by evaluating the spectrum content in the frequency domain, as well as by evaluating the eye diagram. In a preliminary simulation, the cable is represented using a distributed *RC* network. The parameters for this model are summarized in Table 2.

Table 2.1 Cable parameters

Cable parameters	**Unit**
Resistor (R)	series resistance/ meter
Capacitor (C)	shunt capacitance/ meter
Length (L)	Length

The total series resistance is $R_{tot} = R \cdot L$ while the total shunt capacitance $C_{tot} = C \cdot L$. Note that in this work, the distortion of the signal is limited to the effect of the bandlimited channel. The effect of additive white gaussian noise is not included.

The voltage source is a *PRBS* with a maximum run length R_L of 8 for two level of voltage as shown on Table 2.2. The run length is defined as the maximum number of consecutive binary symbols (either "1"s or "0"s). Using an *LFSR* with N registers, the output sequence repeats itself after 2^{N-1} symbols and the run length $R_L = N$.

Table 2.2 Pseudo-Random Bit Sequence Signal Source corresponding values

Voltage Source parameters	**Values**
Maximum length LFSR	8
Bit Rate	18 Mbps

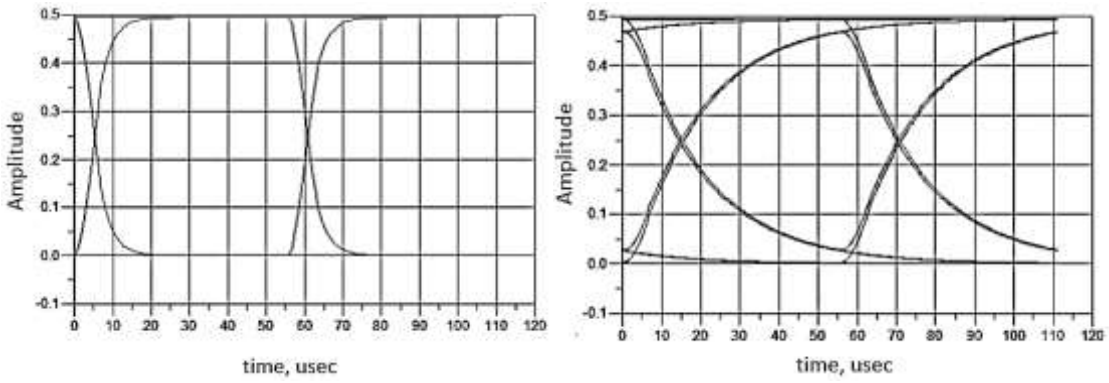
Using the *RC* model of the cable, we have chosen the two value of capacitors as a two different specification of cable for the range of 30 meter by clients as shown on Table 2.3.

The two values of capacitor are chosen to observe different extreme distortions on the signal.

Table 2.3 Cable parameters and their corresponding values

Cable parameters	**Values**
Resistor (R)	32 mΩ
Capacitor (C)	5, 30 pF
Length (L)	30 m

The result from Figure 2.2 shows eye diagram for two different value of capacitances as two different cable specification. As the value of capacitor is increasing, we can observe the eye is getting becomes much more closed. This is caused by the equivalent low pass filter on the high frequency of signals generated by the *LFSR*.



(a) (b)

Figure 2.2 Eye diagram for Different Values of Capacitance in RC cable: (a) C= 5pF, (b) C= 30pF.

2.2 TRANSFORMER COUPLING

In this section, we include the transformer that is used at the transmitter and observe the effect on the end-to-end distortion of the channel. The transformer introduces a zero at low frequency. A representation of the model is shown in Figure 2.3.

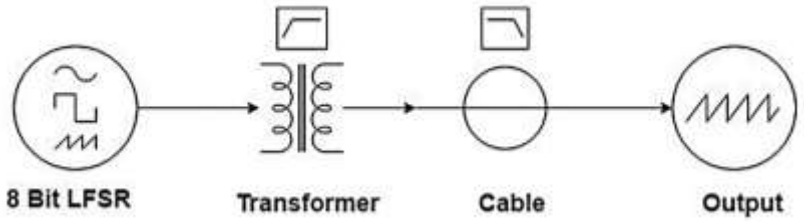


Figure 2.3 Transmission line

In this model, the transformer has an ideal coupling factor $k = 1$ in our link and the self inductor values are representative of the transformer used by the client as shown in Table 2.4.

Table 2.4 Transformer parameters and their corresponding values

Transformer parameters	**Values**
Coupling factor (k)	1
Inductor 1 (L_1)	10 μH
Inductor 2 (L_2)	10 μH

The equivalent frequency response of the channel is simulated in the frequency domain by applying a sinewave at the input. The load at the input and output is assumed to be matched to the transmission line. The effect of the line is evaluated for different capacitor values.

From Figure 2.4, we can observe the simulation results using S-parameters, for different values of capacitance in *RC* cable specification which simulated by *S-P* analysis. The S_{21} parameter provides the transduction gain of the channel, while the S_{11} is the reflection coefficient between the cable and the source as can be observed, the insertion of the transformer also introduces a zero in the transfer function, such that the gain at *DC* is zero. The slope of the gain is 20 dB/dec at low frequency, such that it can be modelled with an equivalent first order high pass filter.

Also, at higher frequencies, it can be observed that a low-pass filter effect is observed with a 40 dB/decade slope, indicating that a second order filter can be used to model the effect of the cable. As can be observed, as the capacitance of the cable increases, the cutoff frequency of the equivalent low-pass filter is reduced, thus limiting the bandwidth of the equivalent channel.

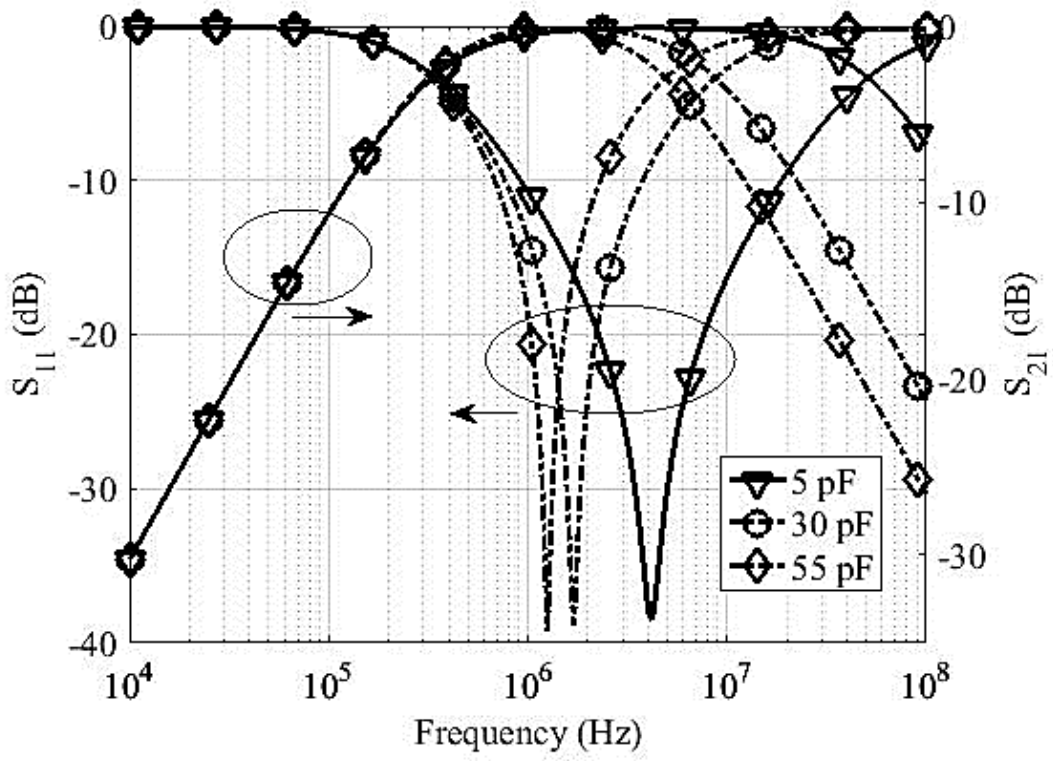
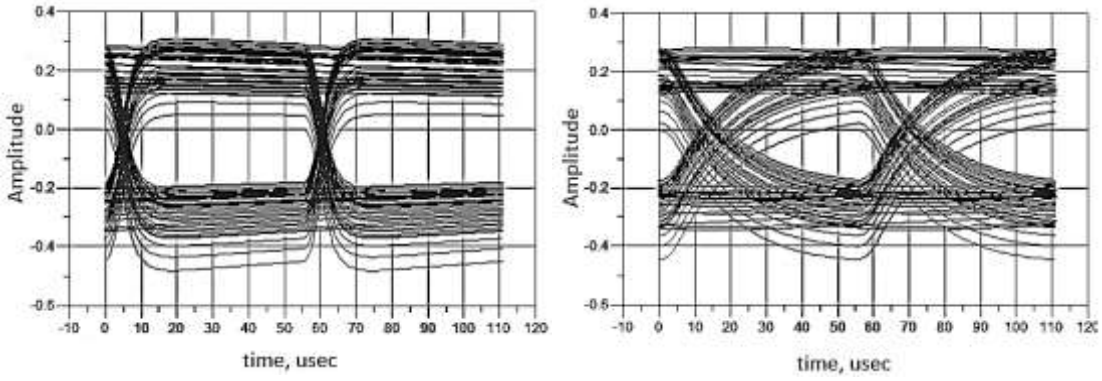


Figure 2.4 S-Parameters of different values of capacitance in RC cable.

The simulation result in Figure 2.4 indicates that the channel is frequency flat for a limited bandwidth. The transmit signal will need to be shaped such that its spectrum fits within this bandwidth, with minimal energy outside the allowable spectrum of the channel.

The simulation results in the time domain, for different capacitor values representing two different cable specifications. A broadband voltage source has the same specifications as described in Section 2.1.

As shown in Figure 2.5, the additional transformer introduces additional distortion to the broadband signal. The transformer effecting on low signals which is generated by *LFSR*. the two capacitors value are increasing by the factor of cable's length.



(a) C = 5 pF (b) C = 30 pF

Figure 2.5 Simulation results of transformer and RC cable of digital source for different capacitor values

2.3 TEST AND MEASUREMENTS

In this project, a low-cost cable with a nominal length of 100 meters is considered. A category *3 UTP (CAT3 UTP)* cable is chosen because it is cheap and has a suitable theoretical bandwidth for the system.

Electric characteristics of *CAT.3* cable as retrieved form the website of *HitekNOFAL* (*2011*) and shown in Table 2.5.

Table 2.5 Electric Characteristics of CAT.3 Cable

Mutual	Capacitance 6.6	Insulation Resistance	9:38 =100m
Characteristics Impedance	100	Conductor DC Resistance	9:5 =100m
Delay Skew (Max.)	35ns/100m	Insulation DC Resistance	5000 M=km
Propagation Delay (Max.)	5.7 ns/100m	Normal Velocity of Propagation	70%

The cable output is loaded with a 50 resistor to the model in order to present the resistance of a typical receiver input resistance.

The cable S-Parameters were measured using a digital vector network analyzer and specified to operate between 1 kHz, and 1.3 GHz. The measurement system was calibrated using a 50 Ω calibration kit.

The measurement was realized using a single-ended configuration. The S_{11} and S_{21} parameters were extracted. the S_{22} and S_{12} parameters were confirmed to be equal to S_{11} and S_{21} respectively.

The measured results using 100 Omega loads at both the transmitter and receiver showing the transmission gain and reflection coefficient are shown in Figure 2.7. As can be observed, the system is relatively well matched (below -10 dB) for frequencies between 100 kHz and 7 MHz. The cable gain in the testing conditions varies with frequency, and at 8 MHz, the gain is on the order of -10 dB.

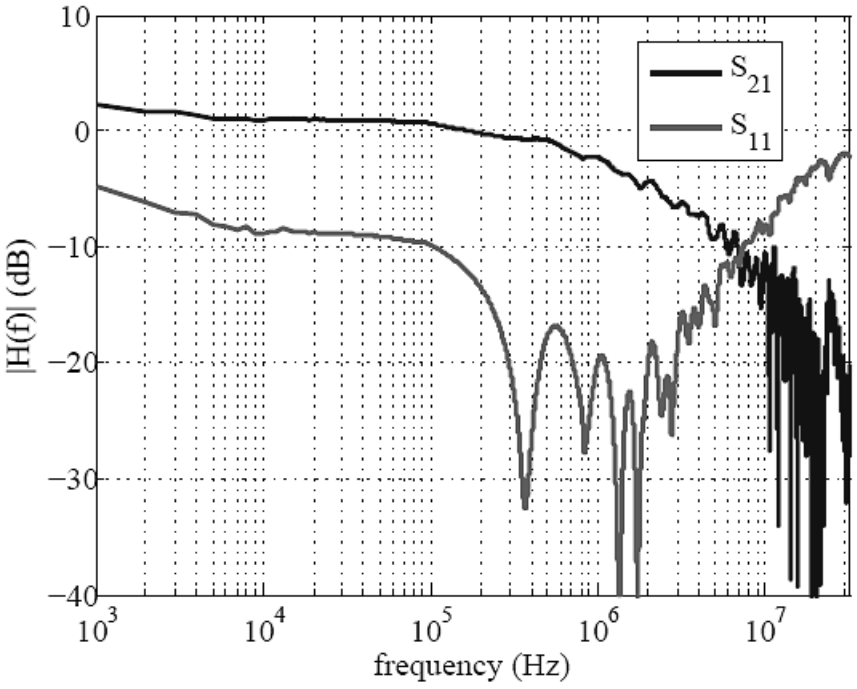


Figure 2.6 Measured S-Parameters.

CHAPTER 3 DIGITAL LINE CODING

In order to select the most applicable signaling technique for data transmission, five basic baseband signaling techniques are analyzed and compared in this chapter. The following baseband signaling techniques are considered: Non-return-to-zero *(NRZ)*, the *Miller* code (of delay modulation), *Manchester* (sometimes called split phase), *8B/10B* as well as a new proposed *Quad* Code.

In this Chapter, a new digital line code will be proposed. Specifically, in Section 3.1, the properties and choice of digital codes has described for the transmission line; then in section 3.2 the various technique of line codes such as *NRZ*, *Manchester*, *Miller*, and *8B/10B* has described by behaviour of *PSD*; in Section 3.3 the new technique of coding algorithm has described with it's *PSD* and Trellis diagram, the new code has named *Quad* code; finally, section 3.4 broadly deals with the different techniques of line codes [10] for represent the digital sequences transmission of rectangular pulses through band-limited channels.

3.1 PROPERTIES AND CHOICE OF DIGITAL CODES

In comparison to wireless communication, line communication relies on broadband data transmission, which is generally transmitted without carrier modulation.

Immunity to the channel distortion broadband signal can be assured on our system in two different ways [11]:

1. Implementing line coding techniques (such as *Manchester*, *8B/10B*, and *Miller*).

2. Developing signal processing techniques at the receiver, such as the matched filter and equalizer.

In this thesis, I focus on line coding.

The main constraints that we define for the design of the baseband signals are as follows [9]:

a. There must be a limitation on the run length to high spectral energy at low frequency.
b. The probability of data must show an appreciable *PSD* amount for our applications.

The choice of form of digital codes depends on the application of interest. In this research of line code technique constrained the following properties of the spectrum [9]:

a. *The DC* component: The channel eliminates the *DC* energy from the signal's *PSD*. Thus, low frequency information could be lost.
b. Favorable power spectral density: The spectral occupancy of signals must be contained within the characteristic of desire channel. There are several criteria that need to be taken into account when selecting a specific baseband signaling scheme for an electronic system. However, the one method which we chosen for more research in the digital communication is Signal spectral characteristics.
c. Signal spectral characteristics: The individual significance of these criteria and that of one signal's advantage over another when signals are used for comparison depends upon the applications. In view of each of these criteria, the spectral characteristics of the signal dictate both the transmission bandwidth required for the signal and the efficiency of the bandwidth which are two of the most important factors in the analysis of any communication system. Another important feature of

the signal that can be obtained from spectral information is whether a signal has zero frequency or information of very low frequency. This is important as these baseband signals are frequently transmitted via ac-coupled networks and thus the information on zero and low frequency is lost. Signals with dc energy must be restored before detection with lost average value information [12]. A few parts of the spectrum are significant. An absent of high-frequency components implies that transmission requires less bandwidth. On the other hand, it is also desirable to lack a direct current component [13-15].

3.2 VARIOUS TECHNIQUES OF LINE CODES

In this research, the digital data is transmitted as digital signals. The purpose of this section is to examine and compare various encoding techniques with the proposed *Quad* Code.

3.2.1 Nonreturn to Zero

The simplest format for a baseband signal is to represent each piece of data by a binary (on-off) pulse or a binary pulse group as shown in Table 3.1.

Table 3.1 Definition of digital signal encoding formats

Nonreturn to Zero	
Data	**Transition**

1	High level
0	Low level

The baseband signaling scheme represented a binary “one” by one signal level and the “zero” by a second level which is the basic non-return-to-zero *(NRZ)* [16] scheme and has been given the designation *NRZ-level (NRZ-L)* in the *IRIG* Telemetry Standards Document [17]. When the "one" symbol is represented by a signal level and "zero" by a zero-level signal, the wave is referred to as unipolar *NRZ-L* or simply as *NRZ-L* to distinguish it from the polar *NRZ-L* signal in which the two binary symbols correspond to a certain positive and negative signal amplitudes. In polar signaling, one logic state is represented by a positive voltage level, and the other by a negative voltage level. The data signaling rate (or data rate) of a signal is the rate, in bits per second (b/s), at which that data is transmitted [18].

Figure 3.1 shows a typical polar *NRZ* and unipolar *NRZ* wave.

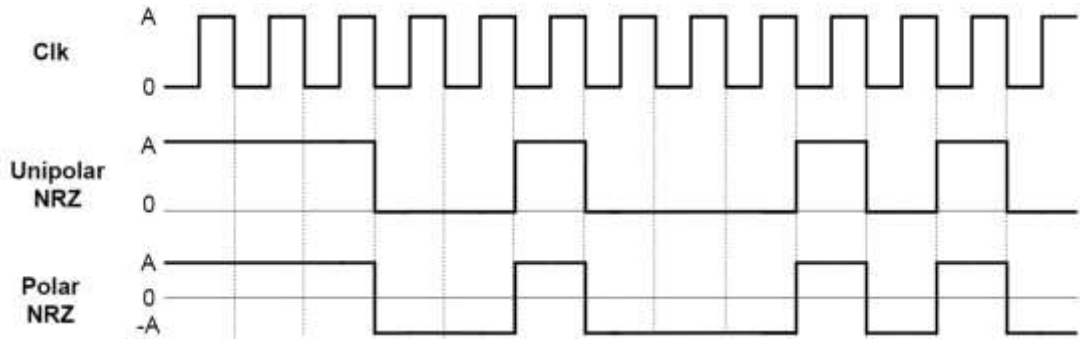


Figure 3.1 Waveform representations of typical Unipolar NRZ and Polar NRZ waveforms.

The *NRZ* codes share the property that the voltage provides constant level during a bit interval, there is no change (no return to a zero-voltage level). These are the least complex codes to implement, and the simplest of these is the *NRZ-L*. *NRZ-L* is a common code used to create or interpret digital data by data processing terminals and different devices. If a different code is to be used for transmission it is commonly produced from an *NRZ-L* signal by the transmission system [18].

The *NRZ* codes are simplest to engineer and make efficient use of bandwidth. This latter property is illustrated in Figure 3.2 (based on [14 and 19]).

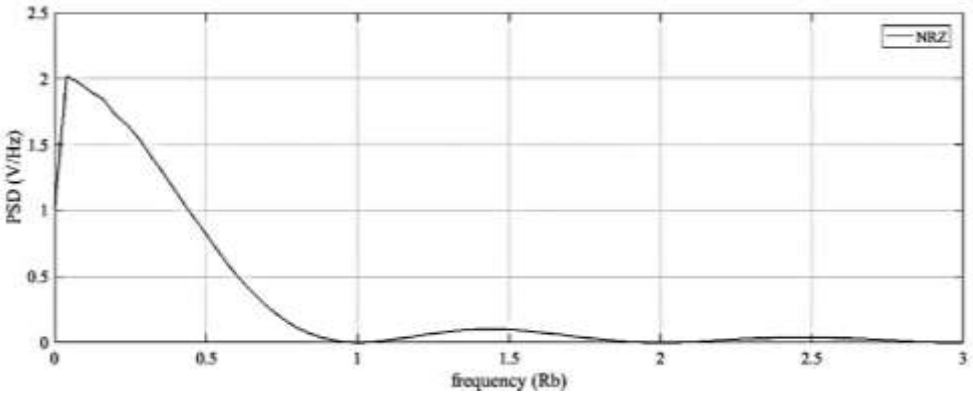


Figure 3.2 PSD of NRZ.

Frequency is normalized to the data rate in Figure 3.2 As can be seen, most of the energy is between the direct component (*DC*) and half the bit rate in *NRZ* signals.

The main limitations of *NRZ* signals are the presence of the *DC* component and the lack of ability to synchronize. To explain the latter issue, consider that when a long string of 1's is received for *NRZ,* the output is a constant voltage, and there is no transition that can help align the received clock to the middle of the symbol. As such, a drift between transmitter and receiver can not be corrected based on the signal alone.

NRZ codes are commonly used for digital magnetic recording due to their simplicity and relatively low frequency response characteristics [20]. However, the limitations of *NRZ* make these codes unattractive for data transmission applications.

3.2.2 Manchester

Both "ones" and "zeros" are represented by a bilevel signal in the Bi-phase signaling arrangement the “ones” by a signal which has the highest of the two possible signal levels during the first half of a bit period and the lowest level during the last half of the bit period, and the “zeros” by a signal that is the inverse of the signal the “ones”.

Table 3.2 Definition of digital signal encoding formats

Manchester	
Data	**Transition**
1	Transition from high to low in middle of interval
0	Transition from low to high in middle of interval

A representation of a typical signal is given in Figure 3.3

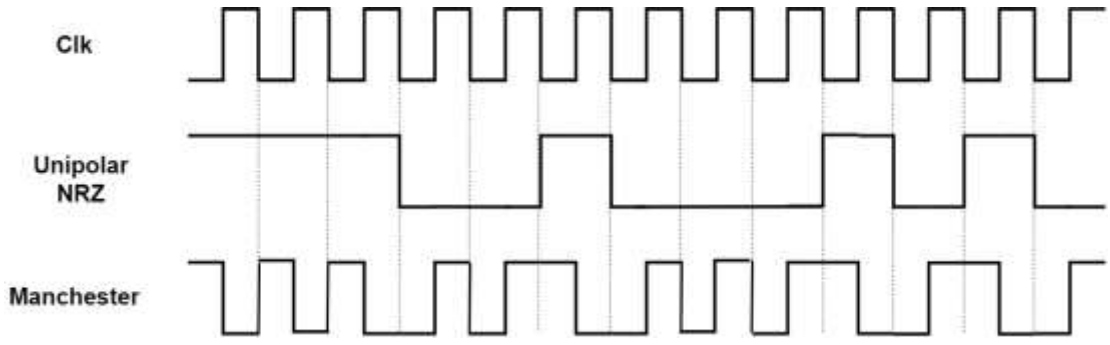


Figure 3.3 Waveform representations of typical *Manchester* and NRZ waveforms

Manchester codes are expected to overcome the disadvantages of *NRZ* and signal-encoding techniques. The *Manchester* schemes require at least one transition per bit time (Table 3.2) and may have two transitions. In this way, the maximum pulse rate is twice that of *NRZ* and the bandwidth is correspondingly greater. To compensate with *NRZ*, the *Manchester* scheme has several advantages [21]:

a. Synchronization: Since there is a predictable transition during each bit time, the receiver can synchronize on the transition. For *Manchester*, there is always a transition in the middle of each bit interval.
b. No *DC* component: *Manchester* code has no *DC* component, yielding the advantages depicted earlier.

From Figure 3.4 as can be seen, the bulk of energy (above 0.5 the peak value) of the *Manchester* code is between 0.4 Rb and 1.25 Rb. Thus, the bandwidth is narrow and contains no *DC* component.

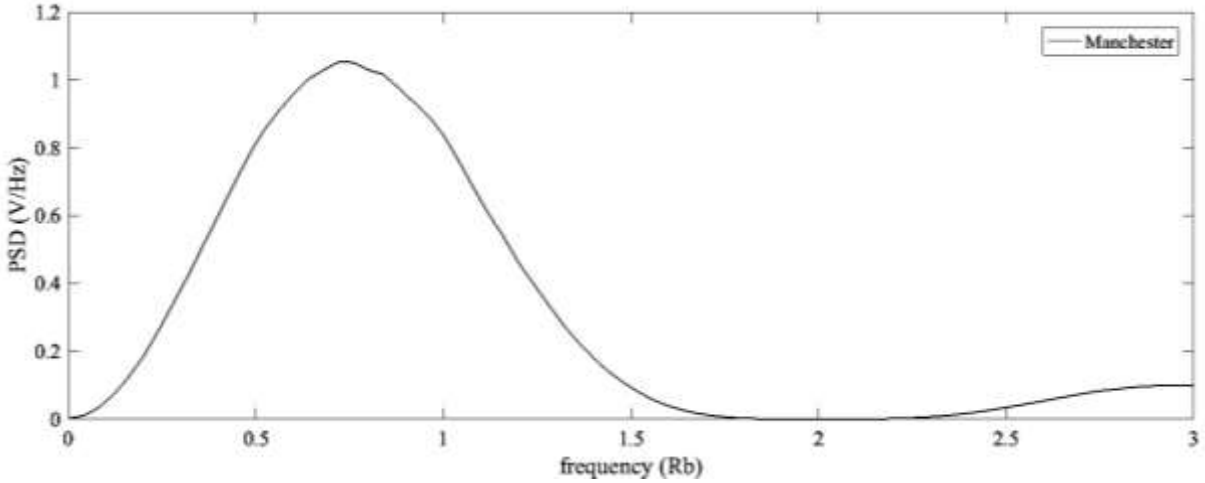


Figure 3.4 PSD of Manchester

Manchester codes are popular techniques for data transmission. Although *NRZ* is still the most broadly utilized in data communications systems, the *Manchester* code is gaining

ground rapidly [22]. The *Manchester* code appear in local network standards. The more common *Manchester* code has been specified for the *IEEE 802.3* standard for baseband coaxial cable using *CSMA/CD* access [23]. It has also been used in *MIL-STD-l553B*, which is a shielded twisted-pair bus system designed for high-noise environments [24].

3.2.3 Miller

Delay modulation, also known as *Miller* coding is an interesting alternative to the *Manchester*. There's at least one transition per two-bit times with *Miller*, and there's always more than one transition per bit.

Table 3.3 Definition of digital signal encoding formats

Delay Modulation (Miller)	
Data	**Transition**
1	Transition in middle of interval
0	No transition if followed by 1
Transition at end of interval if followed by 0 [18]	

Miller coding therefore has some ability to synchronize but requires less bandwidth than *Manchester*. Figure 3.5 shows that the bandwidth for *Miller* is less than either *Manchester* or *NRZ*. for worst-case bit patterns, *Miller* can have a significant *DC* component and greater bandwidth than *NRZ* [20].

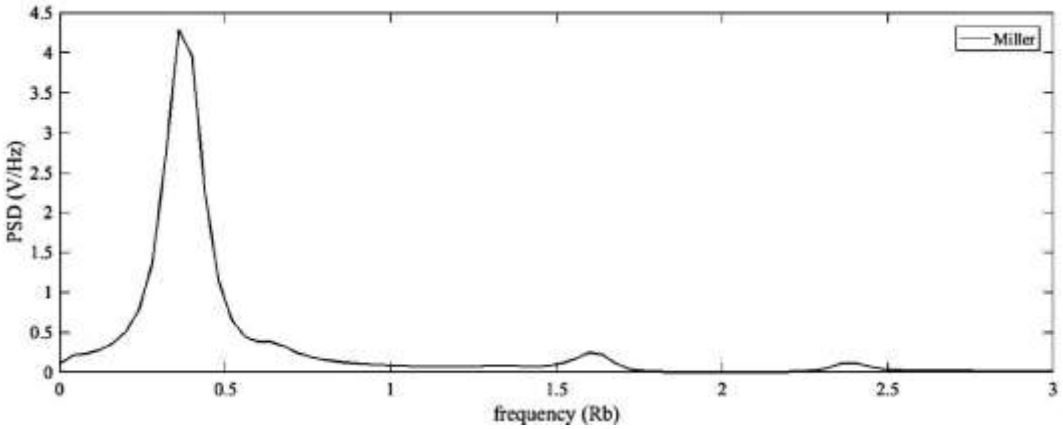


Figure 3.5 PSD of Miller

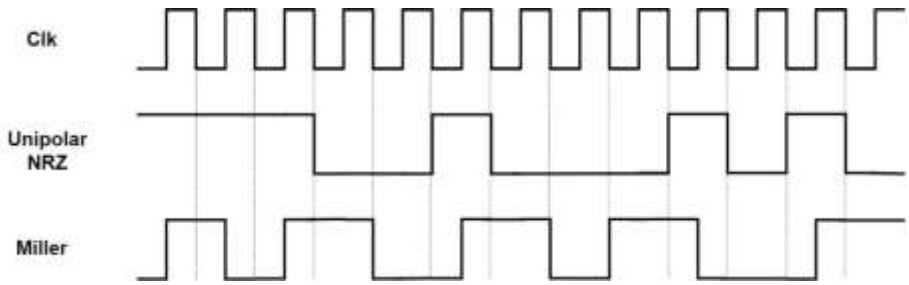


Figure 3.6 Waveform representations of typical Miller and NRZ waveforms.

The delay modulation *(DM)* [25 and 26] or *Miller* code [27] is a scheme that has benefits in some applications. This code's format is shown in Figure 3.6 at the midpoint of the bit interval, a binary "one" is represented by a signal transition. No transition is a "zero" unless another zero follows. A transition is placed at the end of the first zero-bit period in this latter instance.

3.2.4 8B/10B

Due to uprising demands of technology development with high speed and large capacity transmission, the *8B/10B* code is also analyzed in this work. The *8B/10B* code has merit of high precision percent in transmission and a *DC* compensation feature for high-speed transmission.

The *8B/10B* encoder basically transfers the 8 bits of input into10 bits data per the mapping principle. It initially splits the 8 bits data into two groups of bits data, the first group consists of 3 bits data and the second one consists of 5 bits data. Then, to make a 10 bit data, the encoder transfers the first group into 4 bits data and the second one into 6 bits data, as shown in Figure 3.7.

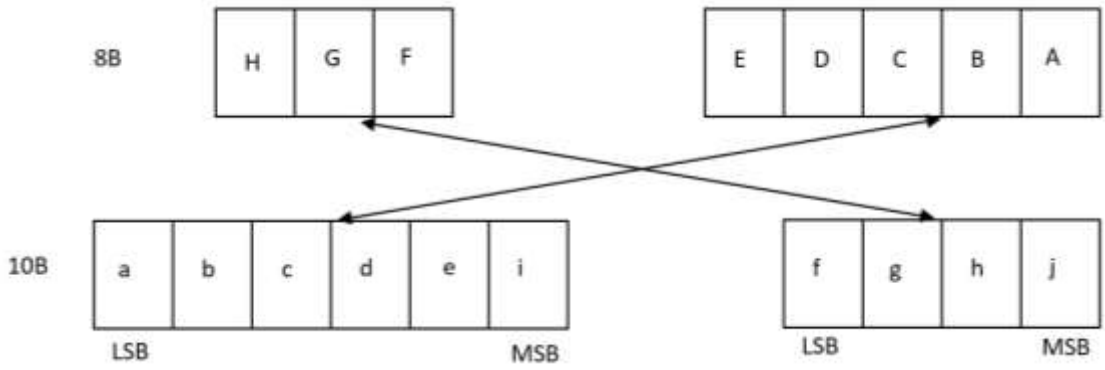


Figure 3.7 8B/10B.

There are three potential arrangements to obtain a sequence of continuous 0s or 1s. There are

- 5 successive 0s followed by 5 consecutive 1s;
- 4 successive 0 followed by 6 successive 1;

- 6 successive 0 followed by 4 successive 1.

These arrangements leads the system to three unbalance scenarios that add up to 0, -2, +2 accordingly which are referred to as a running disparity (*RD)*. A running disparity *RD* that is negative (this is called *RD*-), means that there are additional 1s, and a positive running disparity means that there are additional 0s (this is *RD*+).

As an example, the 8B input data is split into two groups: 3B (front bits) and 5B (back bits). The 3B front bits are converted into 4B and the 5B back bits is converted into 6B, as shown in Table 3.4 The original 8B input data is *HGFEDCBA*. After the bits are divided, the fist group is *HGF* which is encoded into *fghj* and the second group *EDCBA* is mapped into *abcdei*. The final transferred 10B is *abcdeifghj*, as show in Figure 3.7.

Table 3.4 3B/4B Encoder

3B Decimal	0	1	2	3	4	5	6	7
3B Binary (HGF)	000	001	010	011	100	101	110	111
4B Binary (fghi)	0100/1011	1001	0101	0011/1100	0010/1101	1010	0110	0001/1110/ 1000/0111

Ensuring the *DC* balance of data flow during the transmission process is important. Therefore, each 8B data, after encoding will be translated into one kind of *RD*- or *RD*+, as shown in Table 3.5. The next 10B data will be decided based on the previous *RD* to assure the *DC* data flow balance. The first status of the encoder is considered as *RD*-, then it

checks the data flow, if the quantity of number 1 and number 0 remains the same, the polarity of the next 10B data will be *RD*-, otherwise, it will turn to *RD*+. In situation where the previous 10B polarity is *RD*+, and the quantity of number 1 and number 0 of previous 10B data remains the same, the polarity of the next 10B data will be *RD*+, otherwise, it will turn to be *RD*- [21].

The *PSD* of *8B/10B* has non negligible energy up to 1 *Rb* as shown as Figure 3.8.

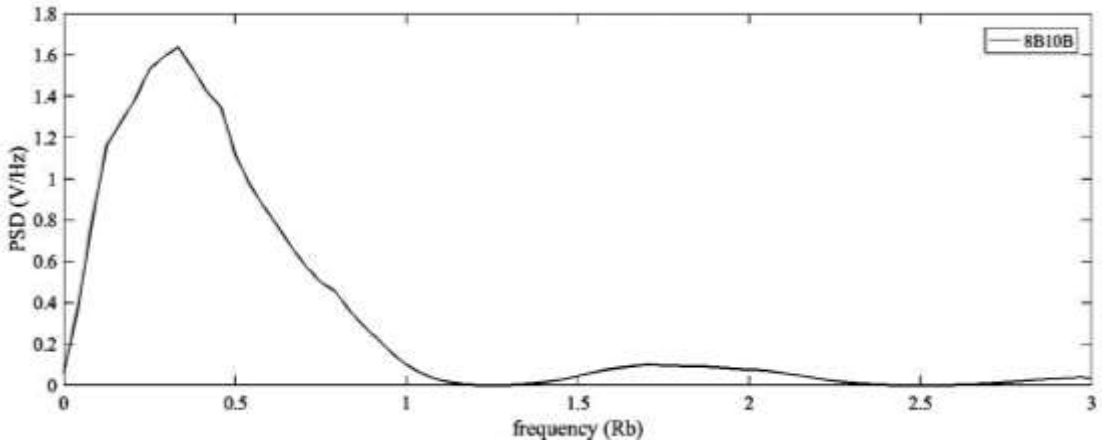


Figure 3.8 PSD of 8B/10B

Table 3.5 Special code encoding list

S.C. Byte Name	S . C. Code Name		HGF	EDCBA	abcdei	fghj	abcdei	fghj
K28.0	C0.0	(c00)	000	00000	001111	0100	110000	1011
K28.1	C1.0	(c01)	000	00001	001111	1001	110000	0110
K28.2	C2.0	(c02)	000	00010	001111	0101	110000	1010
K28.3	C3.0	(c03)	000	00011	001111	0011	110000	1100
K28.4	C4.0	(c04)	000	00100	001111	0010	110000	1101
K28.5	C5.0	(c05)	000	00101	001111	1010	110000	0101
K28.6	C6.0	(c06)	000	00110	001111	0110	110000	1001
K28.7	C7.0	(c07)	000	00111	001111	1000	110000	0111
K23.7	C8.0	(c08)	000	01000	111010	1000	000101	0111
K27.7	C9.0	(c09)	000	01001	110110	1000	001001	0111
K29.7	C10.0	(c0A)	000	01010	101110	1000	010001	0111
K30.7	C11.0	(c0B)	000	01011	011110	1000	100001	0111

3.3 THE QUADRATURE CODE

In this section, a new code is described to transmit through a bandlimited channel. The code name is chosen because for a given bit value, the output code is taken by choosing between one of 4 phases of the transmit clock.

3.3.1 The Quad Code Algorithm

Principle: The *Quad* code that is designed in this work has a rate 2/1. It has at least one transition for every two-bit interval and there are never more than two transitions every two-bit interval. It thus provides relatively good synchronization capabilities, while requiring less bandwidth than the bi-phase signal. The encoding rules are:

• A bit "1" is encoded by a transition in the middle of the bit interval. Hence the code can take on two different values: "01" or "10". On the first transmission of a "1", the output is "01". Else, every time a bit "1" is encoded, the middle transition edge toggles from rising to falling. For example, if the previous bit "1" changed to ''01'', then then the next will change to ''10''. Similarly, if the previous bit "1" was "10", then then the next will change to "01".

• A bit "0" is encoded by a transition at the beginning of the bit interval. On the first transmission of a "0", a "11" is output. Else, every time a bit "0" is received, the code toggles between "00" to "11". Specifically, if the previous bit is "0" then the next changed to ''1''. If the previous bit is "1", then the next changed to ''0''.

The waveforms for *Quad* code illustrated in Figure 3.9.

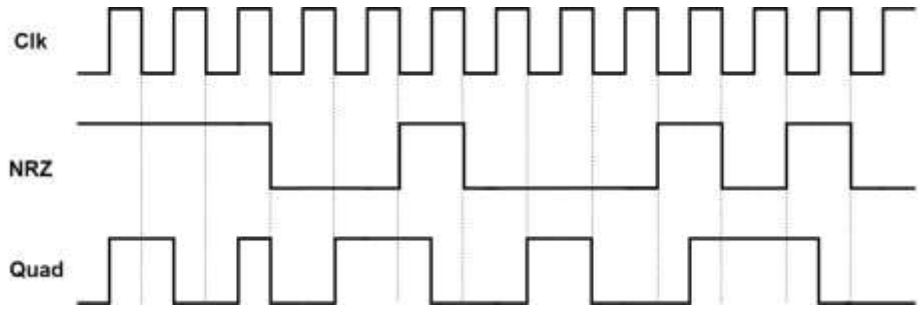


Figure 3.9 Signaling format of NRZ and Quad code

In general, the encoder system can be described using its previous state, and the current input to determine the next encoder output. In this work, encoder is also used to modulate the broadband signal.

Figure 3.10 shows the *PSD* for the *Quad* code, in comparison with that of the *Manchester* code, *NRZ* as well as *Miller*. It is normalized to the symbol rate. As can be observed the peak of the spectrum is at approximately 0.5Rb. This is slightly higher than for the *Miller* code, and smaller than for the *Manchester* code. Also, the energy is spread over a greater bandwidth than the *Manchester*, but not as large as the *Manchester*. With these properties, the *Quad* code will be advantageous in different situations.

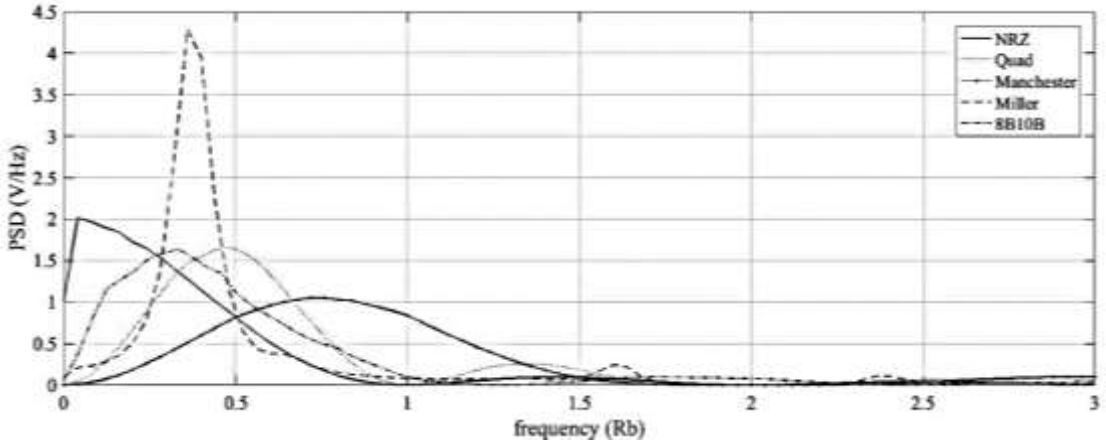


Figure 3.10 PSD for different codes technique

3.3.2 State Diagram of Quad Code

Recall that using the *Quad* code, the transmitted signal depends on the bit to be transmitted in the present interval and the signal, or bit, transmitted in the previous interval. There are

four possible combinations of the present bit and previous signal and therefore there are four states.

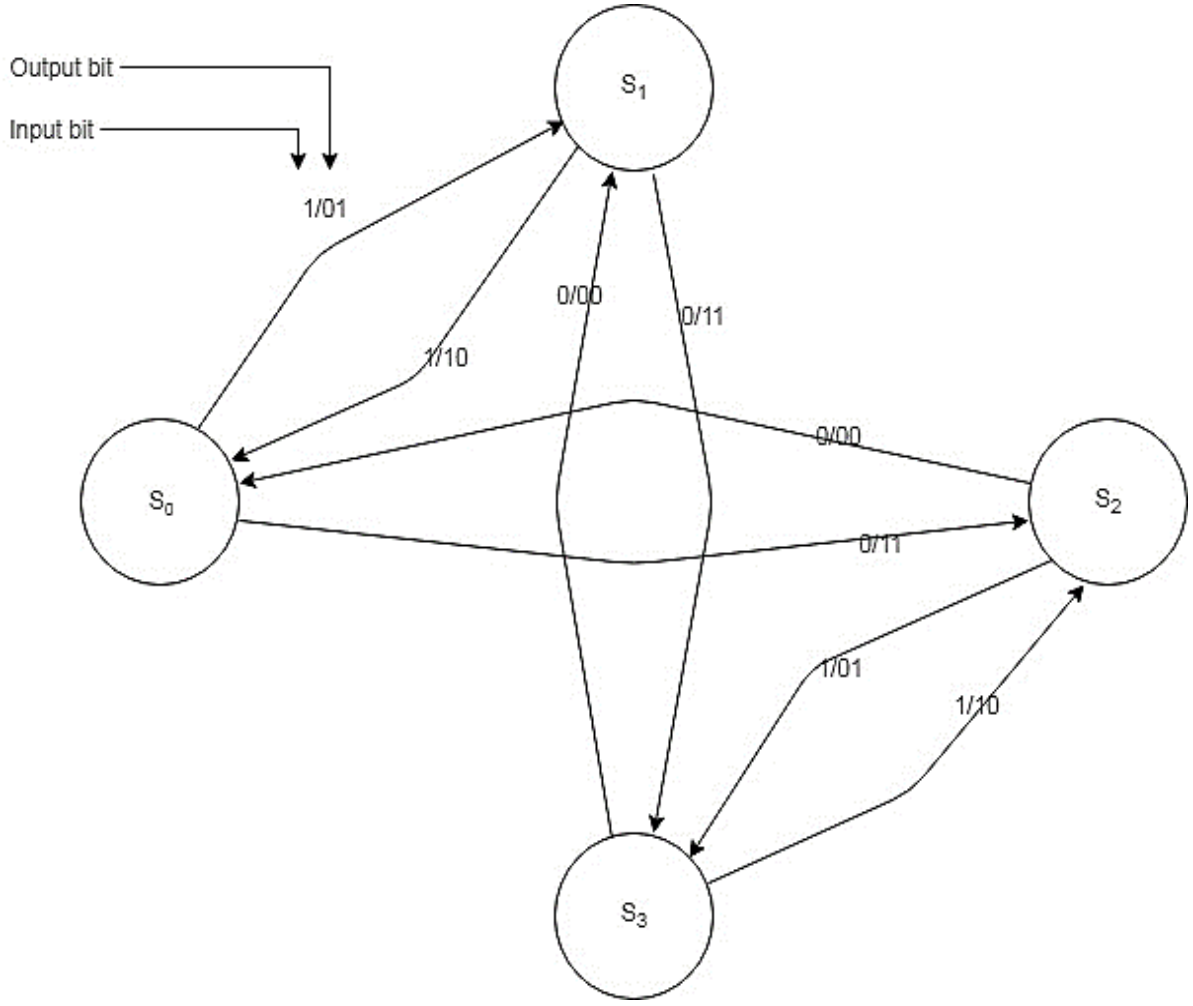


Figure 3.11 State diagram of the Quad code. The state is defined as the signal transmitted in the previous bit interval.

Table 3.6 State table

States	Definition
S_0	Last time we output a zero, we used "00", and Last time we output a one, we used "10"
S_1	Last time we output a zero, we used "00", and Last time we output a one, we used “01”
S_2	Last time we output a zero, we used "11", and Last time we output a one, we used "10"
S_3	Last time we output a zero, we used "11", and Last time we output a one, we used “01”

An example set of transitions is:

$0/S_0 \rightarrow$ "11"/S_2

$1/S_0 \rightarrow$ “01”/S_1

$0/S_1 \rightarrow$ “11”/S_3

$1/S_1 \rightarrow$ “10”/S_0

$0/S_2 \rightarrow$ “00”/S_0

$1/S_2 \rightarrow$ “01”/S_3

$0/S_3 \rightarrow$ “00”/S_1

$1/S_3 \rightarrow$ “10”/S_2

The state diagram that represents the *Quad* code encoding rule is shown in Figure 3.11. As can be observed it includes 4 states. At the beginning of the transmission, the encoder starts in state S_1. Each new input bit forces a transition between the states, and the output code $S_1(t)$, … to $S_4(t)$ are described in Table 3.7.

Table 3.7 Representation of the different codes

Code mnemonic	**Code value**
$S_1(t)$	00
$S_2(t)$	11
$S_3(t)$	10
$S_4(t)$	01

3.3.3 Trellis Diagram of the Quad Code

Although the state diagram in Figure 3.12 clearly and concisely describes the encoding rule, it only describes the transitions between each bit interval. To illustrate the modulator's output for any possible input sequence, one can follow the path dictated by the input bits and produce the output signal. However, a more informative approach is to use a Trellis diagram. In essence, a trellis diagram is simply an unfolded state diagram that represents the output code as a function of time. Figure 3 shows the trellis diagram for *Quad* code modulation over the interval (0, 4T).

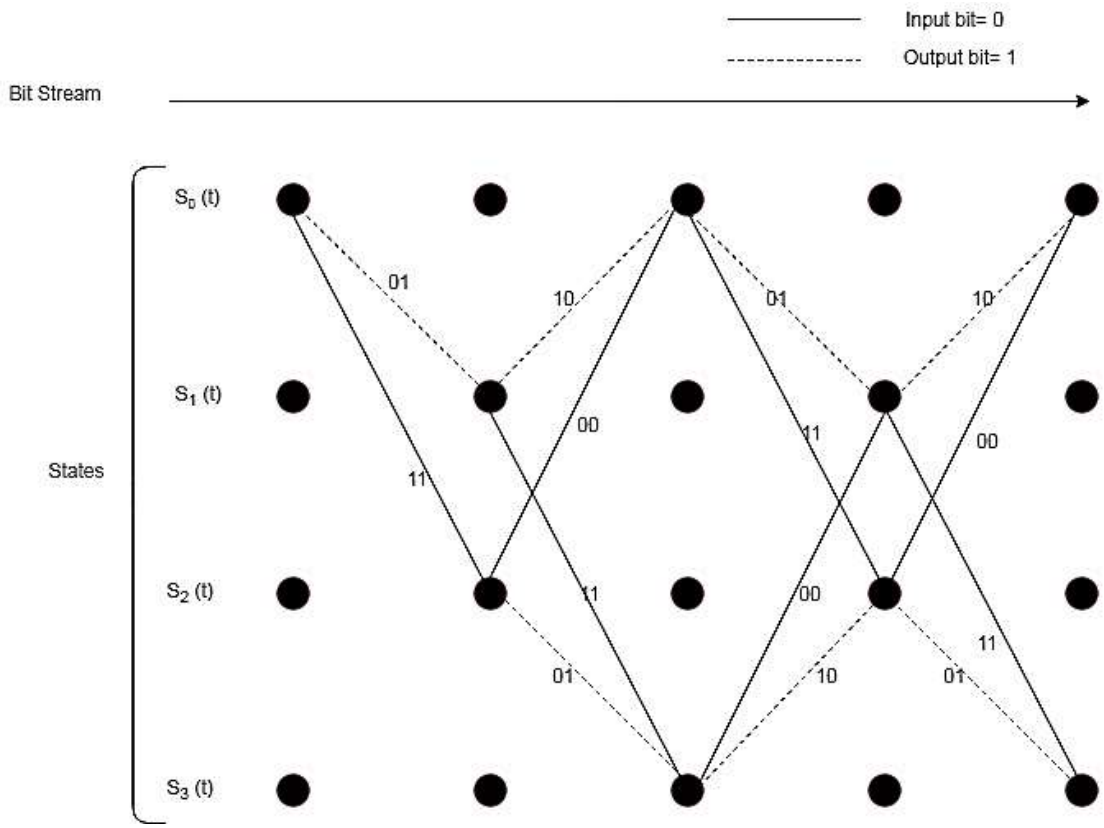


Figure 3.12 Trellis diagram of Quad code

Some important remarks regarding Figure 3.12 are as follows:

• It is assumed that the initial (or starting) state is S_0. In practice, this can always be guaranteed by an agreed protocol.

• A transmitted signal in a given bit interval is represented by a branch connecting two states. The solid line corresponds to bit "0", whereas the dashed line corresponds to bit "1".

• Each possible output sequence is represented by a path through the trellis. Conversely, each path in the trellis represents a valid (or allowable) *Quad* code signal.

Working with the trellis diagram, the main steps in the Viterbi algorithm to recover the sequence (i.e., the path through the trellis) that is closest to the received signal are as follows:

Step 1: Start from the initial state (s_0 (t) in our case).

Step 2: In each bit interval, calculate the branch metric, which is the distance squared between the received signal in that interval with the signal corresponding to each possible branch. Add this branch metric to the previous metrics to get the partial path metric for each partial path up to that bit interval.

Step 3: If there are two partial paths entering the same state, discard the one that has a larger partial path metric and call the remaining path the survivor.

Step 4: Extend only the survivor paths to the next interval. Repeat Steps 2 to 4 till the end of the sequence.

While the Viterbi algorithm provides a structured solution to the implementation of the receiver, in practice, the decoder can be implemented using a simple state machine. In a follow up technical note, the implementation of the receiver, including the synchronization algorithm will be described.

3.4 EFFECT OF THE CHANNEL ON THE DIGITAL LINE CODES

In this chapter, the different techniques of line code such *NRZ*, *Miller*, *Manchester*, and *Quad* through has simulated in band-limited channels.

Specifically, in Section 3.4.1, the different number of run length in *LFSR* has chosen to observe the *SNR* in band limitation of transmission line; then in section 3.4.2 the effect of

low pass filter can be observe in various technique of line codes in *SNR* and eye pattern; finally; then in section 3.4.2 the effect of high pass filter can be observe in various technique of line codes in *SNR* and eye pattern.

3.4.1 Effect of Long Sequences of Pulses

During the design of the code spectral shaping, it is also important to consider the capacity of the code. For example, the *Manchester* code has a 1/2 code rate, and as such the rate of the code pulse is twice the rate of the bit. An interesting alternative to the *Manchester* code, is the *mBnB* (e.g. *8B/10B*) code since it also limits the run length, and the rate of the code pulse is 10/8 times the rate of the bit.

The high-pass effect of the channel distorts a signal that have a low-frequency component. As such, to assess the importance of coding to compensate for the high-pass effect of the channel, the *SNR* can be evaluated as a function of the run length, where the run length is defined as the maximum number of consecutive "1"s or "0"s. In this experience all the results of signal-to-noise ratio *(SNR)* are calculated in dB and are evaluated in the middle of the eye diagram.

The effect of the transformer on a transmit *LFSR* sequence is simulated using a first order high-pass filter in Matlab. A linear feedback shift register *(LFSR)* sequence is encoded using differential *NRZ*. Specifically, in a first instance the effect of the high-pass filter is evaluated using a first high-pass filter. The *LFSR* sequence is transmitted at 16 Mbps and is oversampled by a factor of 20. The output of the channel model is analyzed both in the frequency domain and using the eye diagram.

In Figure 3.13, the *SNR* produced by the Matlab simulator is shown as a function of normalized cutoff frequency (relative to the transmission rate) of the high pass filter. As can be observed, as the run-length *(RL)* increases, the *SNR* decreases. The *SNR* will also deteriorate with an increase normalized frequency. For example, assuming a transmit frequency of 14 Mbps, and a transformer cutoff frequency of 200 kHz, the normalized frequency is 0.014, and the maximum run length allowed to obtain an SNR of 20 dB is $RL = 3$. This highlights the importance of the code design.

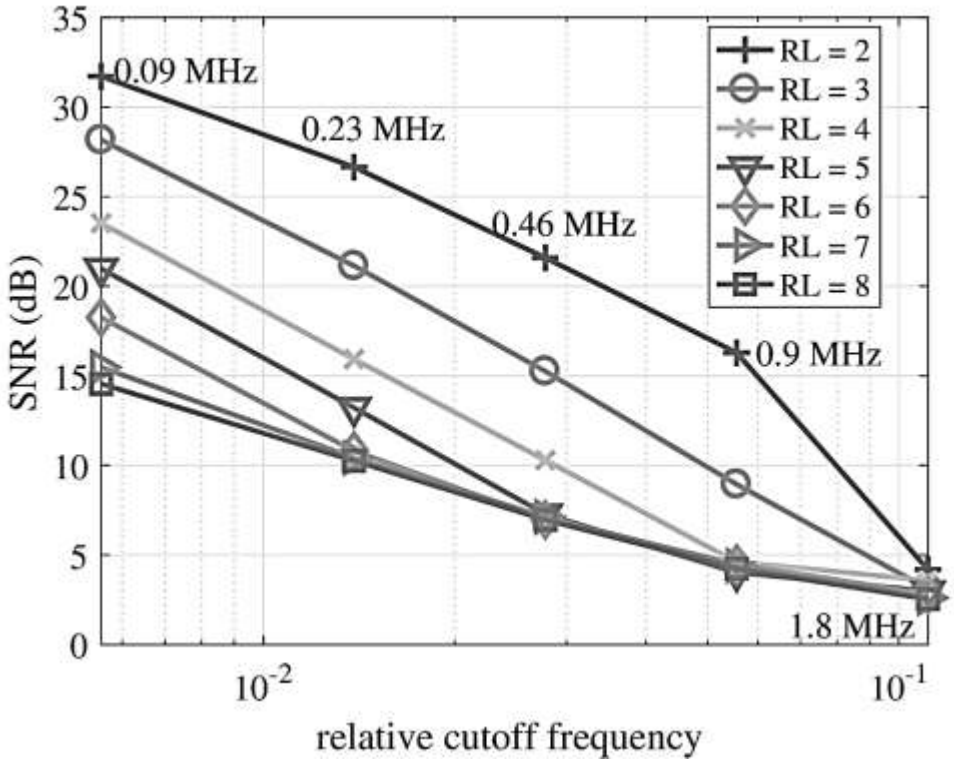


Figure 3.13 Effect of high pass filter on LFSR

3.4.2 Effect of Low Pass Filter in Line Codes

In this Section, the effect of the cable is evaluated using a lowpass filter for simplification on the coded data. The filter response is normalized to the symbol rate R_b and the type of

filter is Butterworth second order. Specifically, the uncoded signal bitrate is 1, and the run-length of the data is $RL = 2^{12}-1 = 2047$. The transient response of the signals for different codes with comparison of *NRZ* are shown in Figure 3.14. As can be observed the high frequency signals are distorted more than low frequencies due to low pass filter.

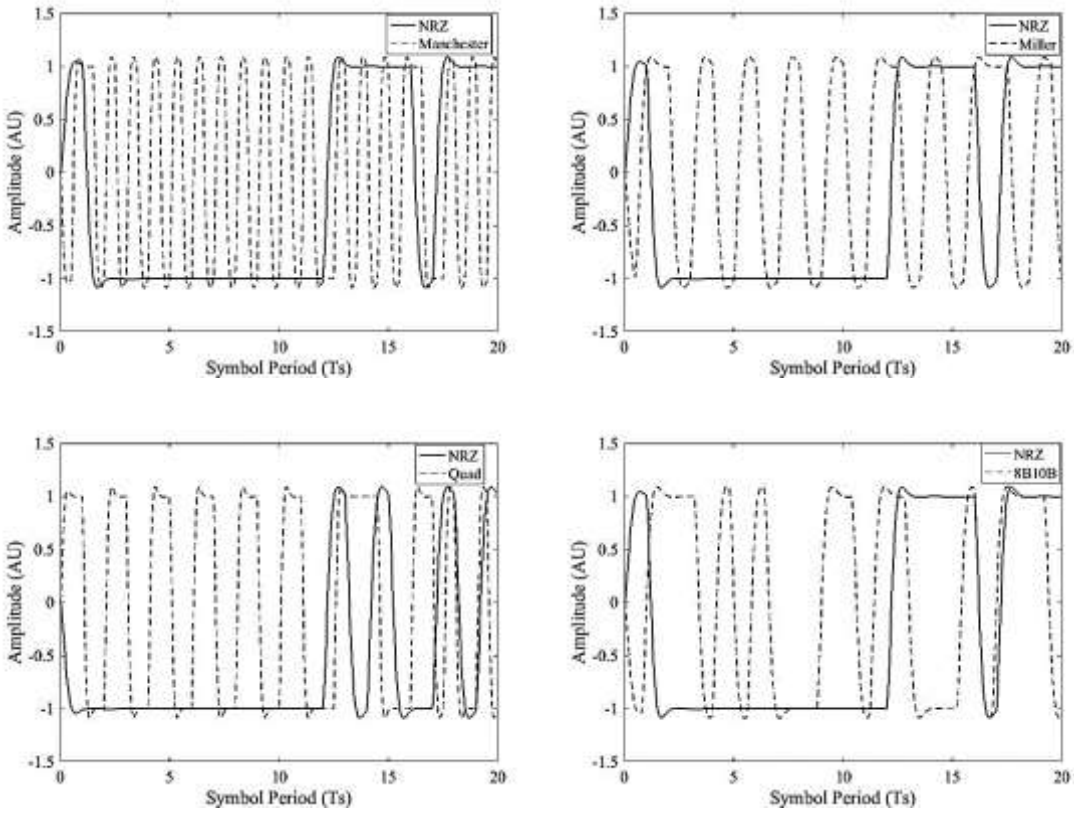


Figure 3.14 transient response of the signals for Lowpass filter at cut-off= 1.

The eye diagram at the output of the low pass filter for different codes is shown in Figure 3.15. As can be observed the eye is getting closer in *Miller* and *8B/10B* cause the effect of low pass filter in the power of high frequencies.

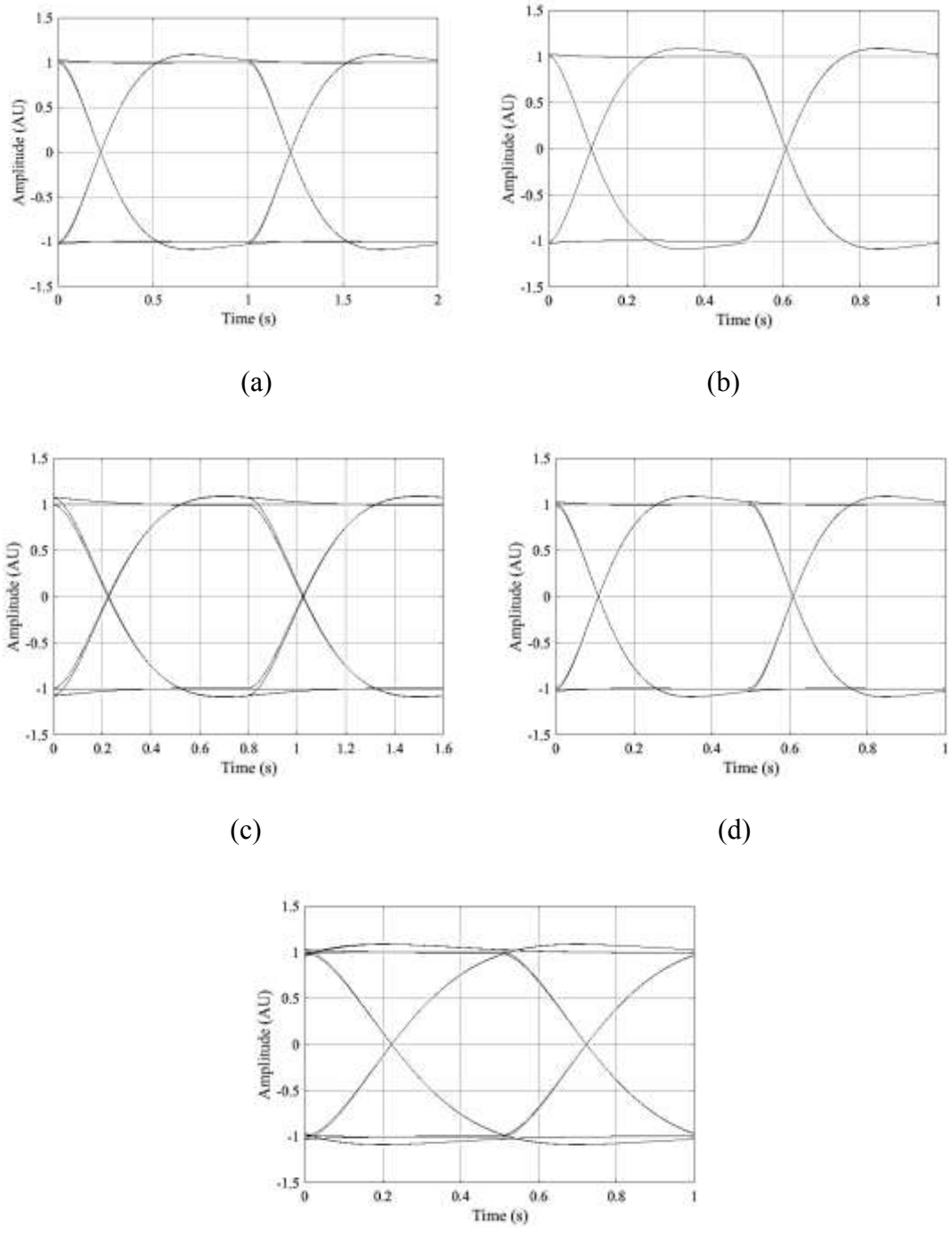


Figure 3.15 Eye diagram of codes for Lowpass cut-off =1: (a) NRZ, (b) Manchester, (c) 8B/10B, (d) Quad code, (e) Miller.

In Table 3.9, the *SNR* at the receiver is simulated using Matlab is summarized for different low-pass filter normalized cutoff frequencies. As can be observed the *Quad* code has better performance for all lowpass filter conditions in comparison to other codes.

Table 3.8 SNR for different codes as a function of SNR

SNR in dB					
Cutoff Frequency (Rb)	**NRZ**	**Manchester**	**Miller**	**New Code**	**8B/10B**
1	28.13	28.13	25	28.26	28.81
0.5	22.98	21.86	11.64	22.34	15.74
0.3	9.77	8.29	7.12	11.45	6.06

3.4.3 Effect of High Pass Filter in Line Codes

In this Section, the effect of the *RF* front-end transformer is evaluated using a high pass filter for simplification, the filter response is normalized to the symbol rate R_b and the type of filter is Butterworth second order. Specifically, the bitrate for *NRZ* is1, and the run-length of the uncoded data is $RL = (2^{12}-1) = 2047$.The transient response of the signals for different codes with comparison of *NRZ* are shown in Figure 3.16. As can be observe the low frequency signals are distorted more in the high frequencies due to *DC* component and characteristics of coding technique, it can observe more in *NRZ* than other codes.

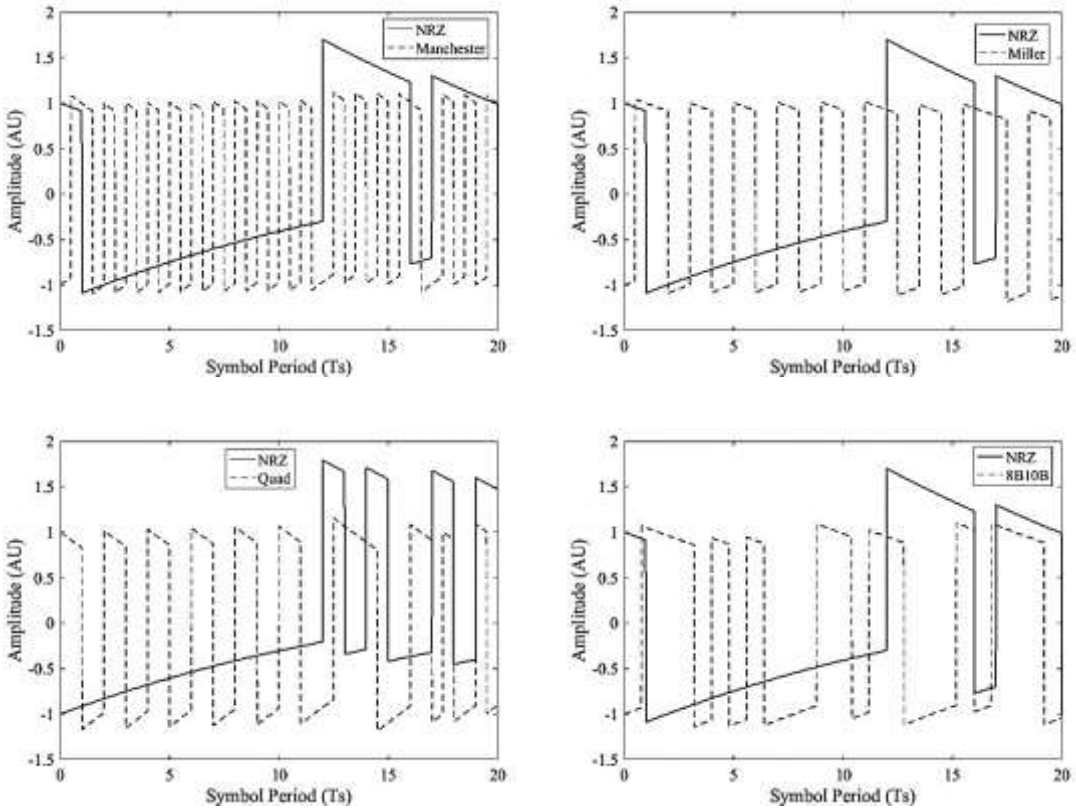


Figure 3.16 transient response of the signals for Highpass filter at cut-off= 0.01.

The eye diagram at the output of the high pass filter for different codes is shown in Figure 3.17. As can be observed the eye is getting closer firstly in *NRZ*, secondly *Miller* and the *8B/10B*, cause the effect of high pass filter in the power of low frequencies.

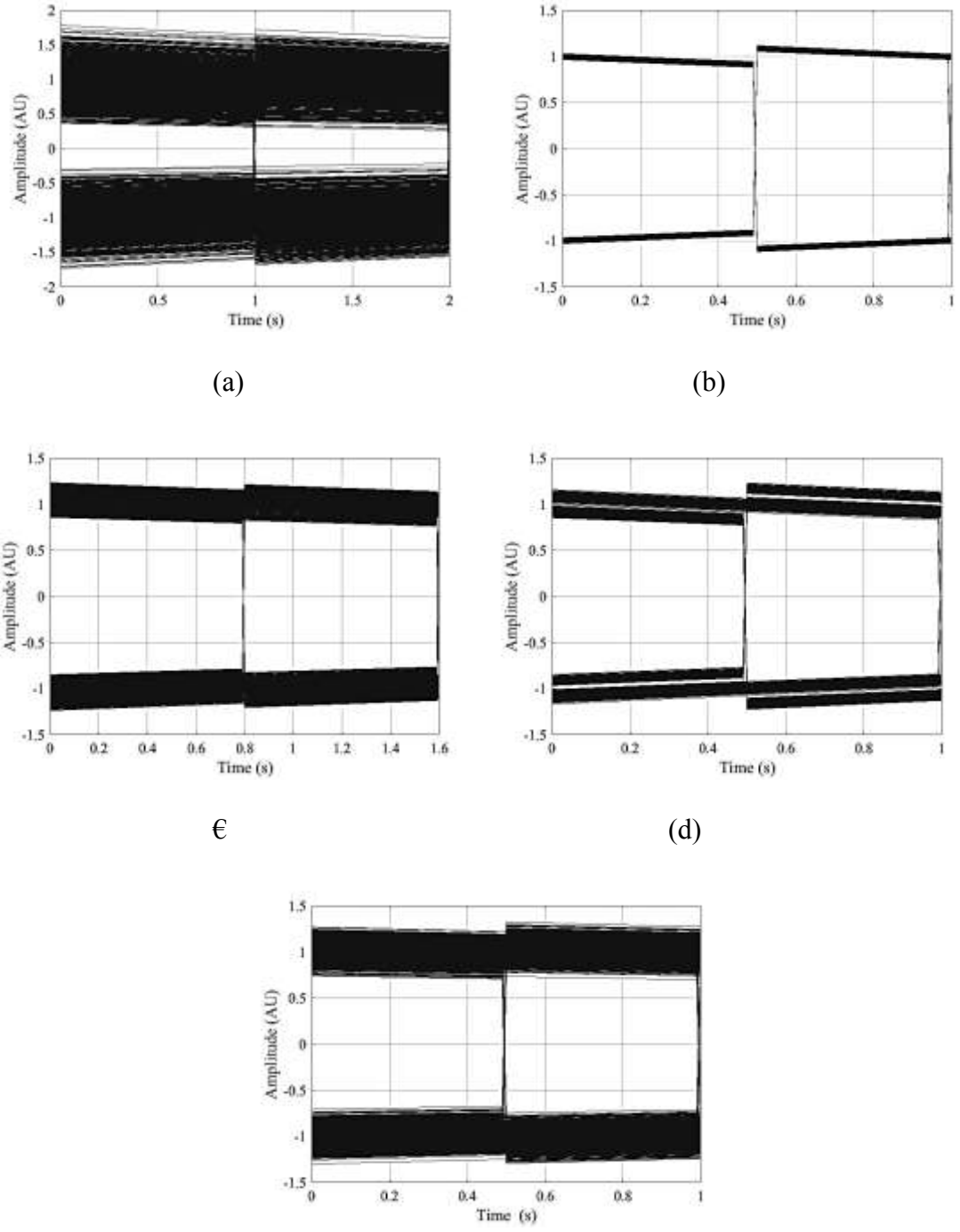


Figure 3.17 Eye diagram of codes for Highpass cut-off= 0.01 for: (a) NRZ, (b) Manchester, (c) 8B/10B, (d) Quad code, € Miller.

In Table 3.9, the *SNR* at the receiver is evaluated using Matlab. As can be observed the *Manchester* code has better performance for the different highpass filter conditions with respect to other codes. The *Quad* Code does get outperformed by both the Manchester Code and the *Miller* Code. Nonetheless, since it is expected that the high-pass filter normalized cutoff frequency will be on the order of 0.01, the *SNR* remains very high above 20 dB. As such, the high-pass filter will not limit the performance of the communication link.

Table 3.9 SNR for different codes as a function of SNR

SNR in dB					
Cutoff Frequency (R_b)	**NRZ**	**Manchester**	**Miller**	**Quad code**	**8B/10B**
0.01	11.77	38.15	21.21	22.41	22.79
0.05	4.67	20.98	12.44	8.69	10.13
0.08	3.22	16.91	9.33	4.6	6.88

CHAPTER 4 RESULTS AND MEASUREMENTS

In this Chapter the implementation and measurements are described. Specifically, in Section 4.1 the procedure of Encoding and decoding of the *Quad* code is described using flow chart diagrams; then in Section 4.2, the phase detection of the *Quad* code for four different clock phases at receiver is described by simulation; finally, in Section 4.3, the *FPGA* board is programmed and connected to the transmission line, and three different coding techniques are tested and compared using the eye pattern and *BER* for four different frequencies.

4.1 QUAD CODE ENCODER/ DECODER

In this Section, first the encoder for the *Quad* Code will be described, then the decoder. The prototype is implemented on an *Artix-7 FPGA* with significant resources and an input frequency of 100 MHz. However, in the final solution the input clock is equal to the data transmission rate and the processing platform contains a core with low computational complexity.

Encoder:

At the transmitter of the *Quad* Code, the encoder is implemented using a reference clock. The start of a frame can be initiated using a reset pulse. When the reset pulse is disabled, at the next rising edge of the clock, a reference signal will be sent. To implement the prototype, in this work a clock divider *IP* core is used to obtain a clock with frequency equal to the symbol frequency. For the implementation, three codes aer programmed: the *NRZ*, the *Quad* Code, and the *Manchester* Code are programmed in *VHDL*. The data

transmission rate is tested at four different frequency equal to 1 MHz, 5 MHz, 8 MHz, and 16 MHz.

The encoder/decoder is tested at four different frequency to observe the effect of the bandwidth limited channel on the different codes.

To encode the *Quad* Code, the following algorithm shown in Figure 4.1 is implemented:

1. The input data *(NRZ)* is initialized and assigned to the array *d*.
2. *i,m*, and *n* represent the input data index, number of zeros, and number of ones, respectively.
3. The algorithm checks the value of the current data input (*d(i)*). If it is zero, it adds the number of zeros (*n*) otherwise the number of ones (*m*).
4. Then the algorithm checks whether *n* or *m* is odd or even:

   4.1 When *d(i)* is zero and *n* is odd, zero is assigned to the output data (*e(i)*) otherwise 1 is assigned to *e(i)*.

   4.2 When *d(i)* is one and m is odd, *d(i)* is *XNOR'ed* with the Clock (*CLK*) and then assigned to the output data (*e(i)*). Otherwise *d(i)* is *XOR'ed* with the Clock (*CLK*) and then assigned to *e(i)*.

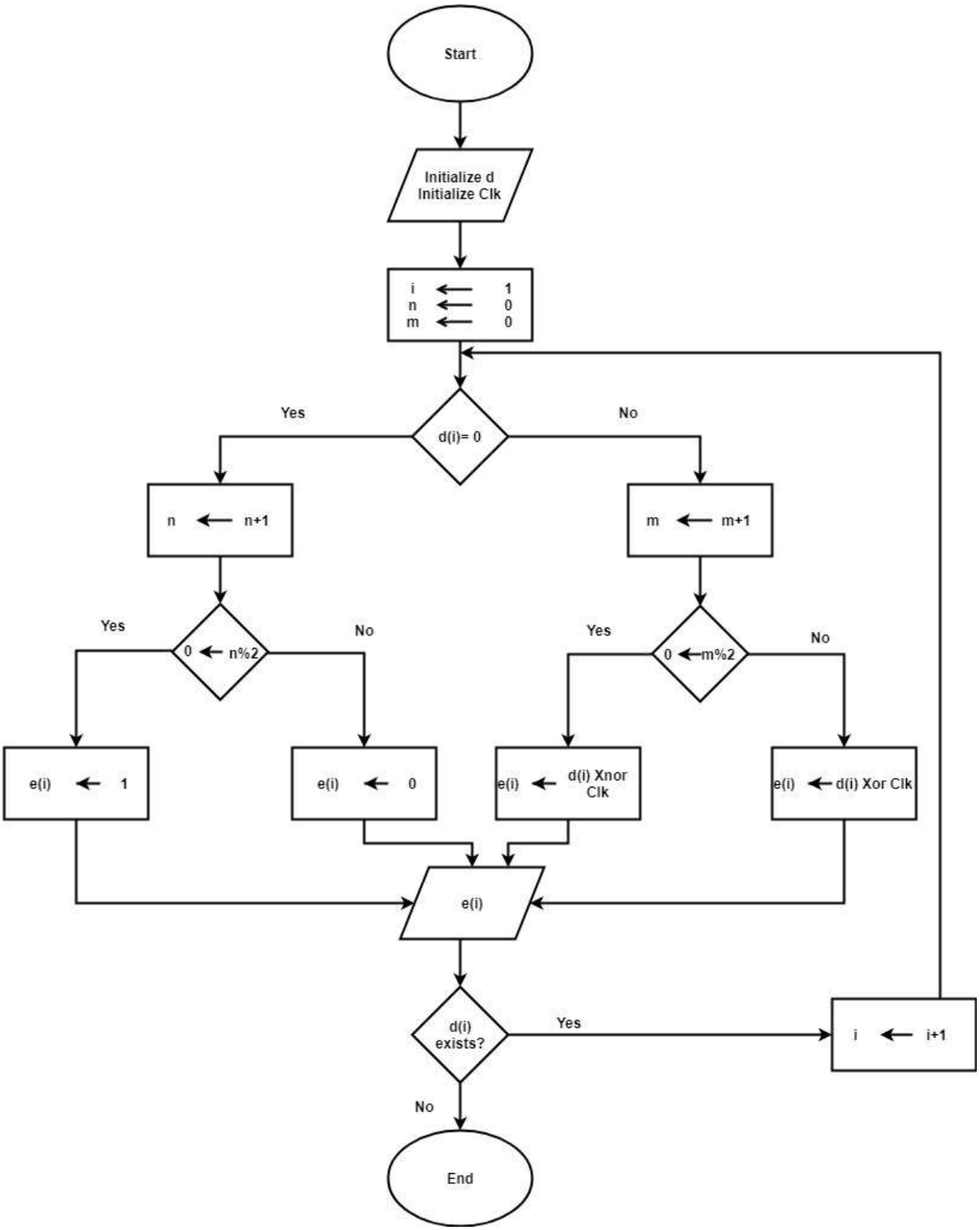


Figure 4.1 Flowchart of Quad encoder.

Decoder:

At the receiver, it is expected that the incoming signal has been synchronized by recognizing an input pattern. To improve the performance, and optimize the sampling phase, it is highly desirable to upsample the signal. As such, a four-phase clock is generated and is applied to 4 different decoders.

The *Quad* code decoder has the following steps as shown in Figure 4.2:

1. The input data (Encoder) is initialized and assigned to the array d.
2. The variables *i,m*, and *n* represent the input data index, number of zeros, and number of ones, respectively.
3. The algorithm checks the value of the current data input (*d(i)*). If it is zero or one then it adds the number of zeros (*n*) otherwise the number of ones (*m*).
4. Then the algorithm checks whether *n* or *m* is odd or even:

4.1 When *d(i)* is zero and n is odd, zero is assigned to the output data (*e(i)*) otherwise 1 is assigned to e(i).

4.2 When *d(i)* is one and *m* is odd, *d(i)* is *XNOR'ed* with Clock (*CLK*) and then assigned to the output data (*e(i)*) otherwise *d(i)* is *XOR* with Clock (*CLK*) and then assigned to *e(i)*.

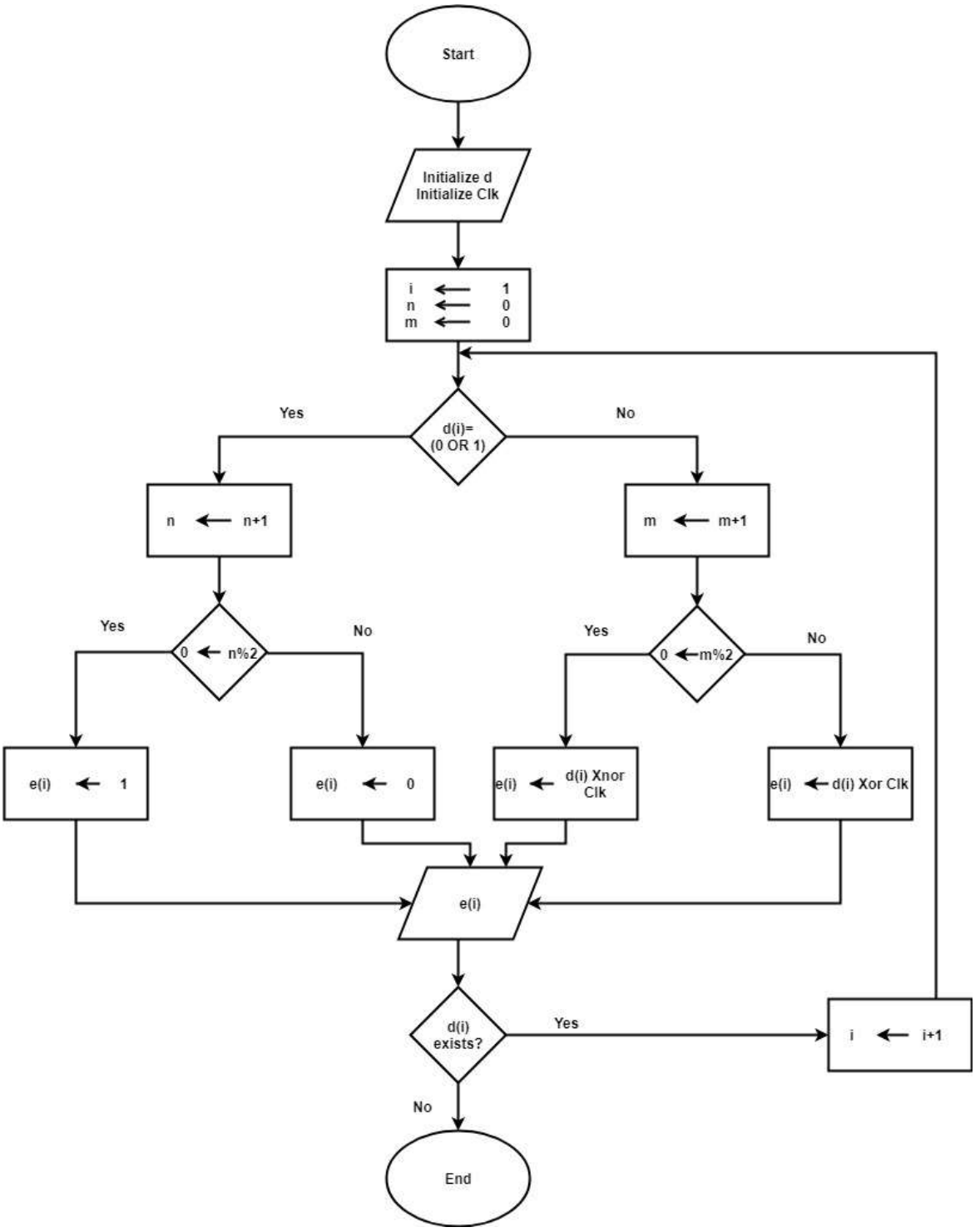


Figure 4.2 Flowchart of Quad decoder.

4.2 PHASE DETECTION

At the receiver, clock recovery is important to sample the data at the right phase, as well as at the right frequency. In this work, four decoders are implemented in parallel, the first, with the rising edge of the clock, a second with the falling edge of clock; a third with a 90-degree phase shift, and a fourth with a -90 degrees phase shift. We use a reference pattern to identify the start of a frame. Note that the clock phase and frequency recovery is not the focus of this work and is proposed as future activities.

Four decoders are controlled using four different phases of the clock. The four cases below identify the different possibilities:

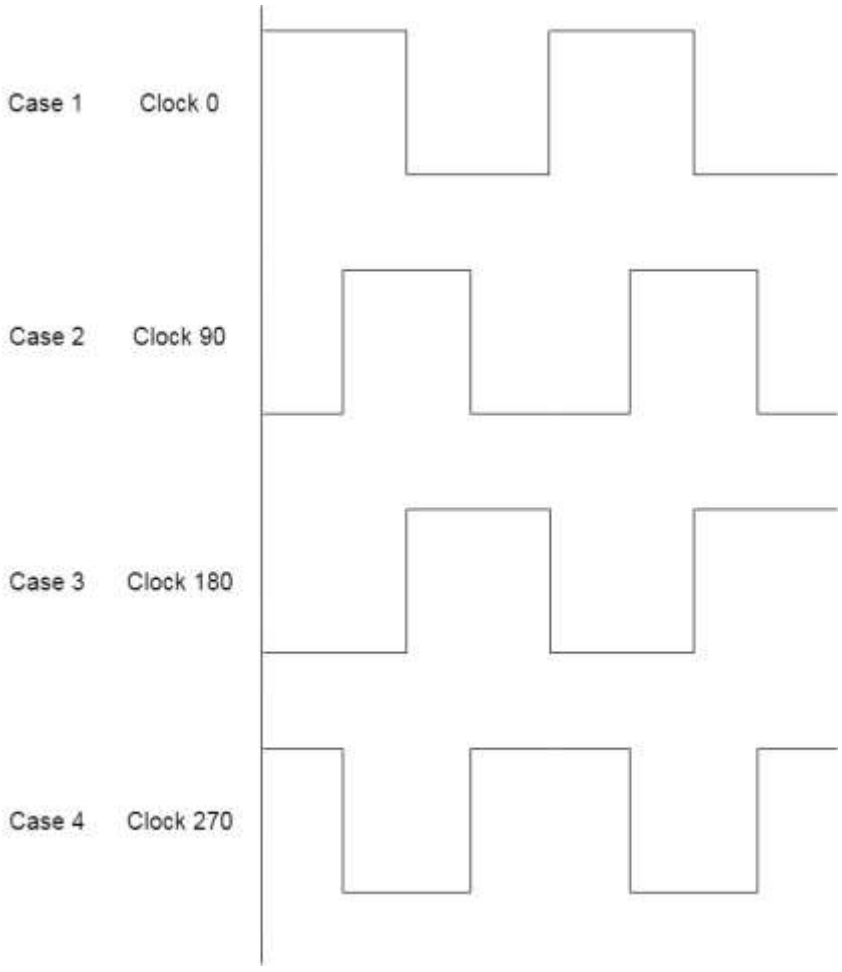


Figure 4.3 Quadrant phase shift of clocks

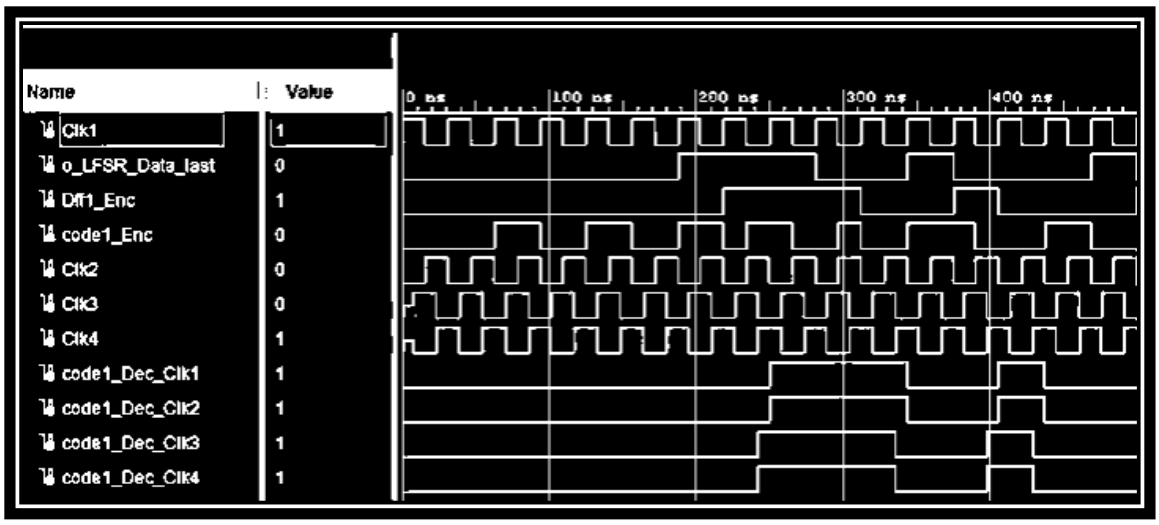
- Case 1: In this case, the reference pattern has detected that the clock with the 0 degree phase shift samples the data. The decoder is expected that the setup time is using the rising up edge of the clock.

- Case 2: In this case, the reference pattern has detected that the clock with the 0 degree phase shift samples the data. The decoder is expected that the setup time is using the rising up edge of the clock.

- Case 3: In this case, the reference pattern has detected that the clock with the 0 degree phase shift samples the data. The decoder is expected that the setup time is using the falling down edge of the clock.

- Case 4: In this case, the reference pattern has detected that the clock with the 0 degree phase shift samples the data. The decoder is expected that the setup time is using the falling down edge of the clock.

The four different clock phases and four relative decoders are shown in Figure 4.4.



4.4 Decoder of Quad code for four phases.

4.3 IMPLEMENTATION AND MEASUREMENT

From the simulation using the high pass filter, the Quad code is demonstrated to eliminate the low-frequency component of the signal., Also, at high frequency, it obtains a better performance than the Manchester code in terms of distortion and SNR. Therefore, the Quad and Manchester codes are programmed as well as the non coded NRZ as a reference code.

In this section the *Quad* Code, the *Manchester* and the *NRZ* are programmed and tested on the real communication link. The three codes are programed using *VHDL* on an *FPGA* board *Artix 7*.

The maximum run length of *LFSR* is 12 which is encoded in real time and implemented at four different frequencies: 1,8,12 and 16 MHz.

As shown in Figure 4.5, the output of the encoder is connected to a buffer amplifier that strengthens the binary signal before it is sent across the line. A second amplifier is used at

the receiver, to regenerate the binary signal at the output of the serial communication link to amplify and improve the signals.

At the transmitter a transformer is used to couple the signal into the differential line. A similar circuit is also used at the receiver. The combination of the transformers and cable produce a bandlimited channel between the transmitter and receiver.

The received signals after passing through the amplifier are decoded using four different phases of clocks that are applied to four decoders.

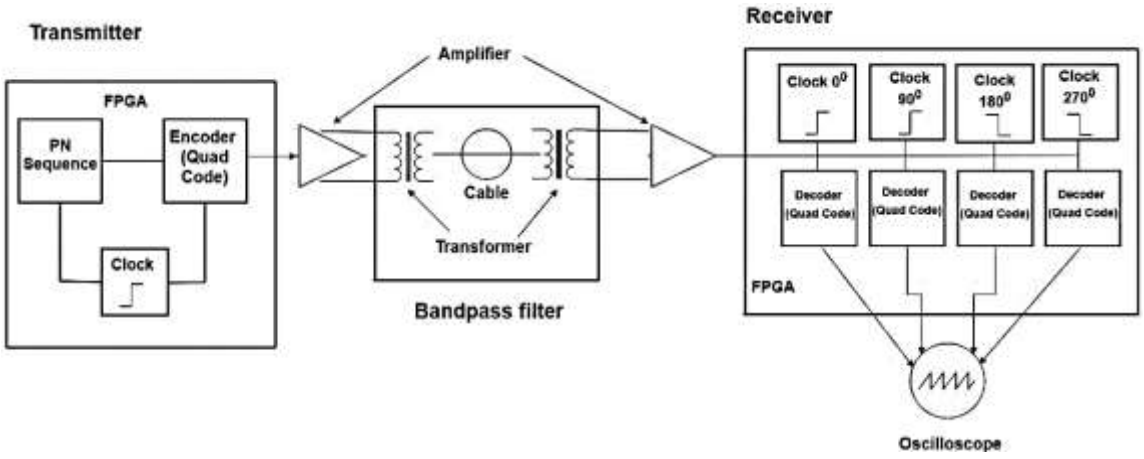


Figure 4.5 Block diagram of implementation.

At 8 MHz, the transmit *PSD* for the different codes is shown in Figure 4.6. As can be observed, the *NRZ* has a much larger *DC* component. The *Quad* Code has energy centered at two frequency, at approximately 0.5 Rb and 1 Rb. In contrast, the *Manchester*'s spectral content is spread over a larger bandwidth and seems to have more energy in the high frequency region. A similar conclusion can be extracted for a signal transmitted at 5 MHz.

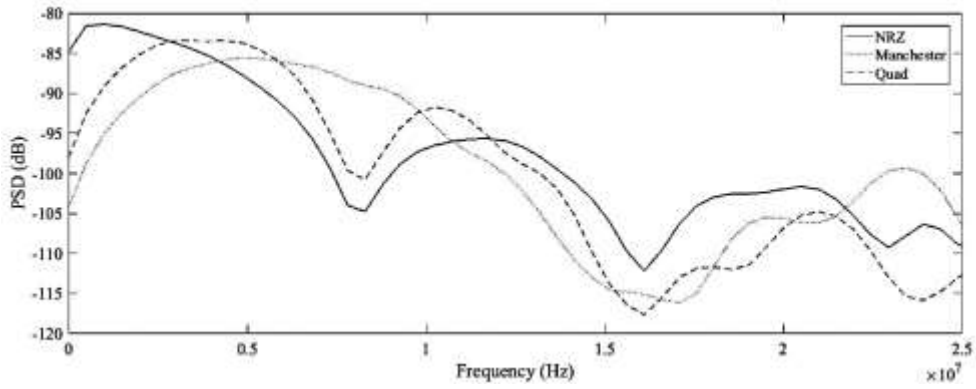
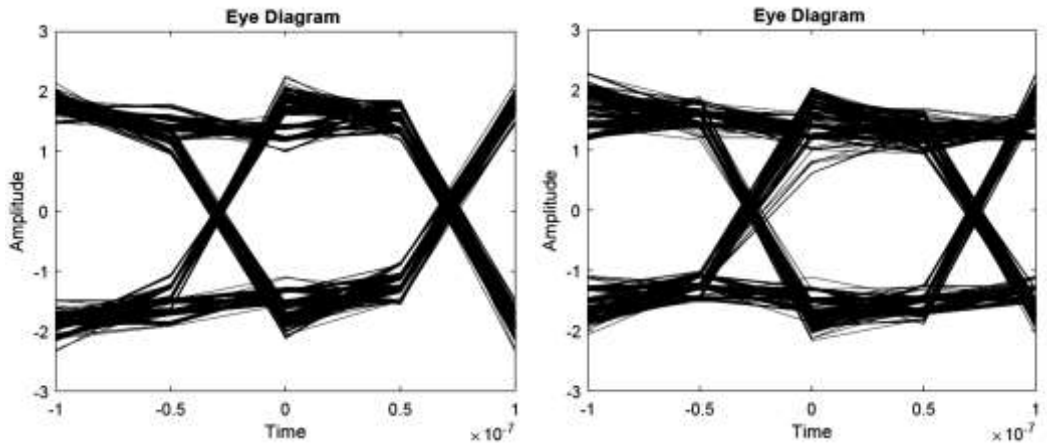


Figure 4.6 PSD of 8 MHz.

Also, at 5 MHz, the eye diagram for the different codes are shown in Figure 4.7. the eye diagram of the *Manchester* code has sharper edges since it is less sensitive to the low frequency zero. This can be confirmed by observing the power spectrum of the Manchester. Note that the distortion of the signal must also account for the effect of the oscilloscope probes, that have a limited bandwidth.



(a) Manchester (b) Quad Code

Figure 4.7 Eye diagram, at the transmitter.

The *Quad* Code was also transferred between the transmitter and receiver as shown in Figure 4.5, The transmit data is recorded using an oscilloscope at the output of the transmitter (producing the signal "input''); at the receiver, the linear output of the channel is measured differentially after the transformer (it is the signal "diff"); at the receiver, the signal is also recorded at the output of the nonlinear buffer amplifier (it is the signal "single ended"). These three signals are measured for bit rates of 8 MBps and 16 Mbps in Figures 4.8 and 4.9, respectfully. As can be observed from Figure 4.8, for the 8 MHz signal, the spectrum of the signal at the receiver is very similar to that of the input for this frequency. Specifically, the shape of the PSD for *Quad* code from the transmitter, differential line, and the output of second amplifier are very close for the first and second lobes. In comparison, it can be observed from Figure 4.9, at 16 MHz the spectrum of bandpass link is not covering the second lob of *Quad* spectrum properly, so we can observe distortion in second lob of *Quad* code.

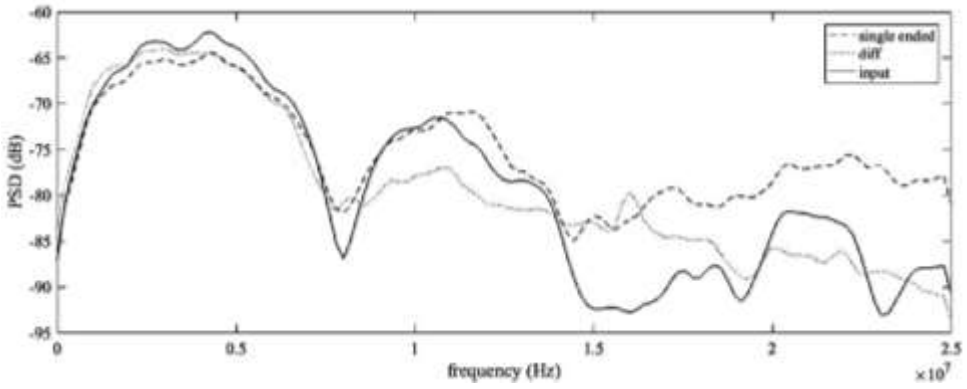


Figure 4.8 PSD of Quad code 8MHz.

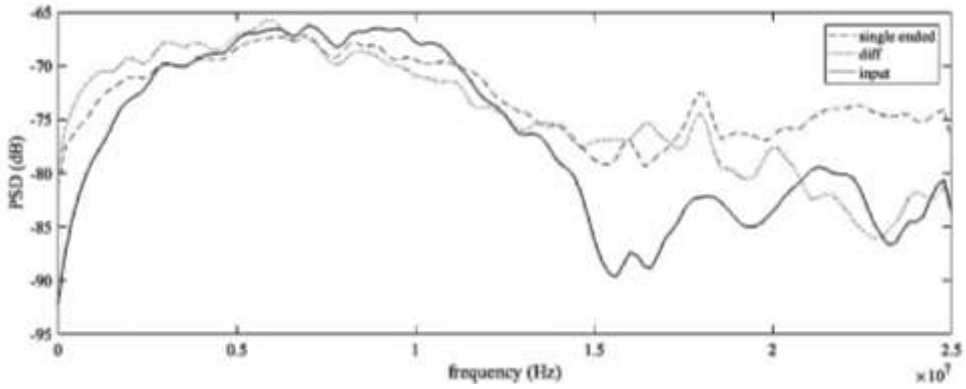


Figure 4.9 PSD of Quad code

To summarize the performance, the bit error rate for the 8 MHz transmission is shown in Table 4.1 for eight sampling phases. As can be observed, at high frequency, the *Quad* code outperforms the *Manchester* code and the probability of bit error is equal to 1.43%. In comparison at low frequency, the *NRZ* performs very poorly compared to the Manchester, and Quad Code. Note that for these measurements, the number of bits processed on oscilloscope is limited by the window size of 10 μsecond and the maximum number of sampling point was 8194. This limited the run length to only a few symbols. Finally, it should be observed that up to a frequency of 12 MHz, the Quad code provides a perfect probability of bit error (for the phase offset of 0.375). This shows its potential for outperforming the Manchester code. This seems to be at the expense of phase sensitivity. To further improve the performance for example at 16 Mbps, spectrally efficient error correction codes can be combined with the proposed line code.

Table 4.1 Bit error rate for different line codes as a

Freq	**Line codes**	**Phase offset (symbol period)**							
		0	0.125	0.25	0.375	0.5	0.625	0.75	0.875
1 MHz	NRZ	0.1231	0.0389	0.1199	0.1199	0.1199	0.1199	0.1199	0.1199
	Manchester	0.000	0.000	0.000	0.000	0.000	0.0343	0.000	0.000
	Quad	0.0265	0.0265	0.0265	0.0265	0.0265	0.0265	0.0265	0.0265
8 MHz	NRZ	0.0076	0.063	0.0611	0.0611	0.0611	0.0611	0.0611	0.0611
	Manchester	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	Quad	0.000	0.000	0.000	0.0458	0.000	0.000	0.000	0.000
12 MHz	NRZ	0.0076	0.0076	0.0076	0.0089	0.0089	0.0089	0.0089	0.0089
	Manchester	0.0229	0.0229	0.0229	0.0229	0.0229	0.0229	0.0229	0.0229
	Quad	0.0025	0.0025	0.0013	**0.000**	0.0025	0.0025	0.0025	0.0025
16 MHz	NRZ	0.0487	0.0448	0.0448	0.0448	0.0448	0.0448	0.041	0.0382
	Manchester	0.166	0.1565	0.0782	0.1555	0.1594	0.1718	0.1718	0.1718
	Quad	0.0592	0.0592	0.0592	0.0592	0.0592	0.0592	**0.0143**	0.0592

BER assuming 8 different sampling phases, for the 3 different codes, and at different frequencies.

CHAPTER 5 CONCLUSIONS

In this thesis, a physical layer communication system has been modeled for transferring digital data. A cable was modeled which provided a simple frequency selective channel that also included a model of the front-end transformer coupling.

In order to select the most reliable signaling technique for data transmission, five basic baseband signaling techniques were analyzed and compared. The following baseband signaling techniques were considered: *NRZ*, the *Miller* code, *Manchester*, *8B/10B*, and the *Quad* code.

The *Quad* code is a new line coding technique that provides a unique spectral content to the transmit signal. Basically, the energy of the signal is focused on a relatively narrow band, and its main lobe is lower than that of other high-speed transmission techniques such as *Manchester*. This provides communication engineers an additional design flexibility. The *Quad* code also is a good choice for data synchronization at the receiver since the maximum duration for a given pulse is two-bit periods.

The simulation of different line codes using a realistic model of the communication channel shows that the *Manchester* mitigates the high-pass effect of transformer, but the its drawback is that the spectrum contains significant energy at high frequency and its performance deteriorates when applied to a long cable in comparison, the *Quad* code provides superior performance as the transmission requirements are increased.

The three codes of *NRZ, Manchester*, and *Quad* were implemented in *vhdl* using an *FPGA* board. The *FPGA* board was interfaced to the communication channel and four quadrant clocks were implemented to detect the data at the receiver.

The implementation results showed a good agreement with the simulation results, which can be observed in *BER* table for four different frequencies.

The different coding techniques provide a balance for different band limitation requirements and while the Manchester is demonstrated to perform best at low transmission rates, the Quad code outperforms all other codes at transmission rates above 12 Mbps.

Finally, a clock recovery algorithm to combat frequency drift, and jitter and effectively chose the appropriate phase forms a very exciting and challenging task for future work.

BIBLIOGRAPHY

[1] T. Braun, T. Voigt, and A. Dunkels, "TCP support for sensor networks," in 2007 Fourth Annual Conference on Wireless on Demand Network Systems and Services, Jan 2007, pp. 162{169.

[2] A. Depari, A. Flammini, D. Marioli, and A. Taroni, "Usb sensor network for industrial applications," in Proceedings of the 21st IEEE Instrumentation and Measurement Technology Conference (IEEE Cat. No.04CH37510), May 2004, vol. 2, pp. 1203{1207 Vol.2.

[3] Moeneclaey Marc Meyr, H. and Stefan Fechtel, Digital communication receivers: Synchronization, channel estimation, and signal processing, Wiley, New York, USA, 1998.

[4] R. Best, Phase-locked loops design, simulation, and applications, McGraw-Hill, New York, USA, 2011.

[5] Julien Faucher, Burst-mode clock and data recovery circuits for optical multiaccess networks, Ph.D. thesis, 2006.

[6] J. Faucher, M. B. Venditti, and D. V. Plant, "Application of parallel forward-error correction in two-dimensional optical-data links," Journal of Lightwave Technology, vol. 21, no. 2, pp. 466{475, Feb 2003.

[7] S. Byun, "A 400 Mb/s 2.5 Gb/s Referenceless CDR IC Using Intrinsic Frequency Detection Capability of Half-Rate Linear Phase Detector," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 63, no. 10, pp. 1592{1604, Oct 2016.

[8] Vucetic B. and Yuan J., Turbo codes, principles and applications, Kluwer Academic Publishers, Massachussets, USA, 2001.

[9] Das, A., "Digital Communication: Principles and System Modelling", Springer-Verlag Berlin Heidelberg, 2010.

[10] Carlson, A.B., Crilly, P.B., Rutledge, J.C., "Communication Systems − An Introduction to Signals and Noise in Electrical Communication (4th Ed.)", McGraw-Hill International, 2002.

[11] Bhattacharya, Amitabha, "Digital Communication", TATA McGrawHill, 2006.

[12] Deffebach, H. L. and Frost, W.O.: A Survey of Digital Baseband Signaling Techniques. TH X-64615, NASA TECHNICAL MEMORANDUM, June 30, 1971.

[13] Deffeback H. L. and Frost W. 0., "A Survey of Digital Baseband Signalling Techniques," NASA Technical Memorandum TM X-64615, June 30, 1971.

[14] Lindsay W. C. and Simon M. K., Telecommunication Systems Engineering, Englewood Cliffs, NJ: Prentice-Hall, 1973.

[15] Lathi B. P., Modern Digital and Analog Communication Systems, New York: Holt, Rinehart, and Winston, 1983.

[16] Bennett, W. R.; and Davey, J. R.: Data Transmission. McGraw-Hill Book Co., Inc., New York, N. Y., 1965.

[17] Anon.: Telemetry Standards. (Revised February 1969), Document 106-69, Secretariat, Range Commander's Council, White Sands Missile Range, N. M. 88002.

[18] December 1984-VoI. 22, No. 1 2 IEEE Communications Magazine.

[19] Bell Telephone Laboratories, "Transmission Systems for Communications," 1982.

[20] Severt R. H., "Encoding schemes support high-density digital data recording," Computer Design, May 1980.

[21] Sanders L., "When to prefer Manchester coding," Nectronics, July 28, 1982.

[22] Sanders L., "Manchester coding gaining on NRZ," Nectronic Design, August 5, 1982.

[23] Stallings W., Local Networks: An Introduction, New York, Macmillan, 1984.

[24] Mandelkern D., "Rugged local network follows military aircraft standard," Nectronics, April 7, 1982.

[25] Hecht, M.; and Guida, A.: Delay Modulation. Proceedings of the IEEE, July 1969, pp. 1314-1316.

[26] Jacoby, G.: U. S. Patent 3,414,894, December 3, 1968.

[27] Booye, M. A.: An Engineering Evaluation of the Miller Coding in Direct PCM Recording and Reproducing. Prepared by the Custom Products Engineering Dept., Ampex Corp.

[28] Qiaoyu X., and Huijie L., 8b/10b Encoder Design, The 2nd international conference on computer application and system modeling, 2012.