A Project Report

on

DESIGN AND FIELD-TRIALS OF ROTATING POLARIZATION WAVE BASED WIRELESS MODEM

submitted by

Sandeepkumar S Surkute

in partial fulfilment of the requirements for the award of the degree of

MASTER OF TECHNOLOGY

Department Of Electrical Engineering

Indian Institute Of Technology, Madras

JUNE 2016

PROJECT CERTIFICATE

This is to certify that the project report titled ***Design And Field-Trials Of Rotating Polarization Wave Based Wireless Modem***, submitted by **Sandeepkumar S Surkute**, to the *Indian Institute of Technology, Madras*, for the award of the degree of **Master Of Technology**, is bonafide record of the research work done by him under our supervision. The contents of this report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof. Devendra Jalihal
Guide
Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Prof. Radha Krishna Ganti
Guide
Assistant Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Prof. Venkatesh Ramaiyan
Guide
Assistant Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai
Date: Jun 2016

ACKNOWLEDGEMENTS

At the outset, I deem it to be a privilege to have guided by Dr. Devendra Jalihal, Dr. RK Ganti and Dr. Venkatesh Ramaiyan for this project.

I sincerely wish to thank Dr. Devendra Jalihal for the faith and confidence, he reposed in me. He had given a lot of freedom in exploring ideas despite the inherent pressure of deadlines and various setbacks. He has also encouraged me and other team mates at the crucial times of crisis when the experiments have not been so much fruitful and the time at hand has gone in vein. Owing to clear vision and professional expertise, Dr. Devendra Jalihal has helped us in course corrections and redefining the plan of action.

I wish to thank Dr. Radha Krishna Ganti for stepping in to help us interpret results correctly and devise future plans. On many occasions, when we encountered dead ends, he had taken initiative and paved the way for future progress. Real Time Implementation is one such example which has now become a success with his guidance and direct involvement.

I wish to thank Dr. Venkatesh Ramaiyan for his support in devising and executing the plan of half duplex communication. He has been very kind and inquisitive at the same time in giving valuable inputs on kick starting the next stage of the project i.e. star topology based network. As the project moves into further stages, I am sure the next team of students will vouch for his guidance and expertise on the subject.

I wish to thank Dr. Takei and Dr. Ritesh of Hitachi for their patience during the various presentations. Despite being highly experienced professionals in their respective domain, Dr. Takei and Dr. Ritesh have been more than candid in recognising our efforts thereby encouraging us to attain future goals. Together, they have been a great source of inspiration for the team.

I am also fortunate to get in company of Ms. Theeksha (MS Scholar) who has been a very interesting co-passenger in this eventful journey. She has been tirelessly leading not only us but the previous MTech students also. I am grateful to her for being patient and enthusiastic in answering all our queries.

I am also thankful to Mr. Arjun Nadh, Ph.D Scholar of the EE dept., who has been very forthcoming to help us through arbitrary but very crucial issues of USRP devices and Linux platform.

I also wish to take this opportunity to thank all the professors who have taught me various courses at this prestigious institute. They have not only taught me a particular subject but also imbibed me with engineering vision to analyse and solve challenging problems in ways I never knew before.

I would be forever indebted to the prestigious Indian Institute of Technology, Madras, its faculty and staff.

ABSTRACT

The project work is a collaboration of Hitachi and IITM. A wireless M2M (machine to machine) communication in the process industry plants is severely affected by slow fading nature of propagation channel and mutipath reflections due to presence of metallic obstructions that interact with the Electromagnetic Waves. Slow fading means if the signal is in deep fade it will continue to remain in deep fade for longer time with high probability.

An idea of introducing fast fading through polarisation angle diversity is presented by Dr. Takei of HITACHI. The polarisation angle diversity is discussed in great details in his paper[1]. At IITM, the team has worked on the basic transmitter and receiver structure needed for introducing Rotating Polarisation. The idea is to send the same data through different polarisation angles at the transmitter and receive it in those many polarisations. The signal processing techniques are used at the Receiver to decode the signal and achieve diversity.

In the early phase, we have used GNURadio along with USRP (Universal Software Radio Peripheral) to analyse and compare performances of erstwhile Fixed Polarisation scheme, Rotating Polarisation Scheme and the existing standard radio 'ZigBee' which is also a Fixed Polarisation Wave scheme. The experiments are conducted in the Machines Lab which resembles to an industry plant like environment due to heavy metallic electrical equipment present. The results have shown that the performance of Fixed Polarisation schemes FPW and ZigBee is heavily dependent on the antenna orientation. Whereas , the RPW performance is consistent with any of the antenna orientation. Here, the antenna orientation is used as an equivalent for changing environment.

On completion of the above experiments at frequencies 430MHz and 860MHz, It has been envisaged to build Real Time Transmitter and Receiver so that a Star Topology based network could be designed for extensive field trials as well as practical deployment of the RPW Scheme. In case of erstwhile experiments, the data was received in file and it was then processed offline separately. A Real Time system will receive and process data in the receiver itself. The receiver will also be able to ascertain the integrity of data using CRC Checksum.

Towards, this a half duplex point to point communication system is developed using USRPs and C/C++ environment. Each node (USRP) in the link is made capable of switching its role between transmitter and receiver. This is the basic requirement to proceed with design of a protocol for the Star Topology. In the Half Duplex link, the master node transmits continuously at suitable Frame repetition rate. It switches to receiver mode in after each frame transmission to receive acknowledgements from the receiver if any. The Slave node is always in the receive mode and acknowledges only if the frame has certain id field corresponding to this receiver. The half duplex link is an interim stage to ascertain the USRP capabilities. As the half duplex is successfully implemented. All the features of the half duplex can be tuned to suit our requirements to design a Star Topology based Rotating Polarisation Wave Modem.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Project Background

The project is a joint work of HITACHI and IITM. The scope includes design and field trials of Rotating Polarisation Wave Based Modem. The project is in the interim stage. The concept of polarisation angle diversity has been introduced by Dr. Takei of HITACHI through his papers on the subject. In the previous stages of this project the team at IITM has formulated the problem statement and designed basic transmitter and receiver architecture to implement rotating polarisation using GNURadio and USRP (Universal Software Radio Peripheral) and Matlab for signal processing. The core architecture has adopted some changes with respect to frame size, pulse shaping etc. over time however, the method of introducing rotating polarisation remains same. The background is however given here once again for the sake of completeness.

Many complex control systems are deployed in the heavy industries like cement plants, oil refinery, heavy manufacturing units. The control systems also use a plenty of sensor devices for data acquisition. A robust and reliable M2M (machine to machine) communication is highly essential for reliable operation of the whole plant. Many of the critical systems still rely on wired communication for this purpose. Reliability of wireless communication is doubtful given the Non Line of Sight and multipath propagation channel in presence of all the metallic and other scatterers which tend to reflect or interact with EM energy. On the contrary, the wired communication is susceptible to physical damages and breach of information security. At times, the fault finding could be very difficult in case of wired networks as the size of network increases.

The main drawback of the wireless communication is slow fading (static) nature of the propagation channel. Slow fading means if the signal is in deep fade it will continue to remain in deep fade for longer time. An idea of introducing fast fading through polarisation angle diversity is presented by Dr. Takei of HITACHI. The polarisation angle diversity is discussed in great details in his paper[1]. Also the students have worked on the project in previous phases of this project. Their reports have been mentioned in the references [2],[3]. In the previous reports, the students have talked about how the rotation is introduced to transmit same data in different polarisation angles to achieve diversity. However, some of the basics of the signal processing done at transmitter and receiver are reproduced here briefly for the sake of completeness. It is recommended that this report may be read in conjunction with the previous reports.

1.2 Rotating Polarisation Wave (RPW) Transmitter and Receiver

The data(1s & -1s) is spread using a 64 length P-N sequence of which the second half 32 bits are same as first half 32 bits. It is then pulse shaped using SRRC(Square Root Raised Cosine) pulse before giving it for rotation and up conversion by USRP. The details of spreading and rotation are given in the previous reports. The other signal processing details are given in subsequent chapters. The transmitter of this modem is based on the following method. The signal is represented by $a(t)$ is first multiplied with $\exp(jw_1 t)$ i.e. with $\cos(w_1 t)$ in other arm and $\sin(w_1 t)$ in other then upconverted using $\cos(w_c t)$ before transmission. This is depicted in figure below.

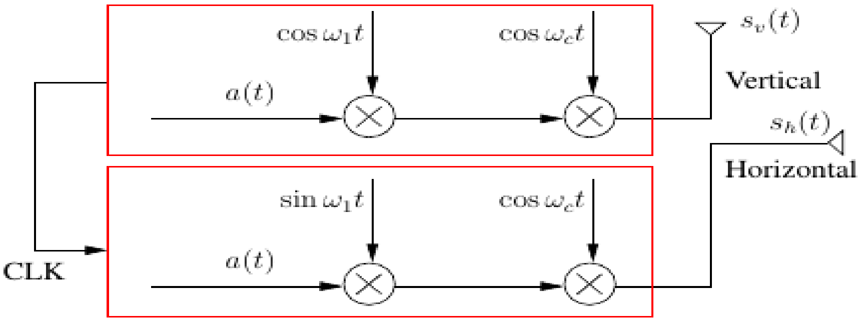


Fig.1.1 Transmitter Structure

The signal at two antennae is given by

$$s_v(t) = a(t)\cos(w_1 t)\cos(w_c t)$$
$$s_h(t) = a(t)\sin(w_1 t)\cos(w_c t)$$

These signals are fed to two antennae which are mutually perpendicular to each other so that a vector sum of these two signals is created in the air. The signal from each antenna is a vector rotating at w_1 frequency. These two vectors are perpendicular to each other in space due to crossed dipole antennae and have a phase difference of 90^0. The resultant EM wave is a circularly polarised wave with rotation frequency of w_1 . The projection of the transmitted signal on x and y axis is depicted in the figure below.

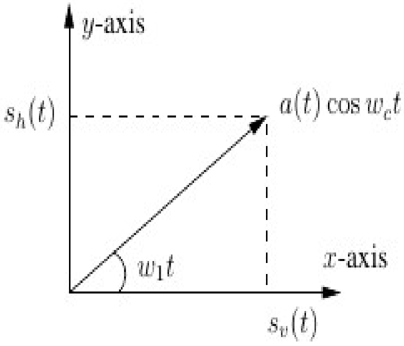


Fig. 1.2 Projection of the transmitted signal in the plane of two antennae

The propagating signal experiences narrowband fading due to scattering in the presence of obstruction and the received signal $r(t)$ can be written as :

$$r(t) = h(t)a(t)\cos(w_1 t)\cos(w_c t + \theta(t))$$

where $h(t)$ represents the narrow-band time-varying channel coefficient and $\theta(t)$ is the phase in the received signal due to the propagation time delay. This is received by a combination of crossed dipole antennas and a splitter as shown in fig 1.3.

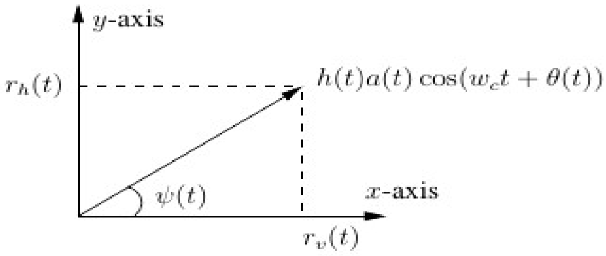


Fig.1.3 Received Signal Vector and its projection on two axes

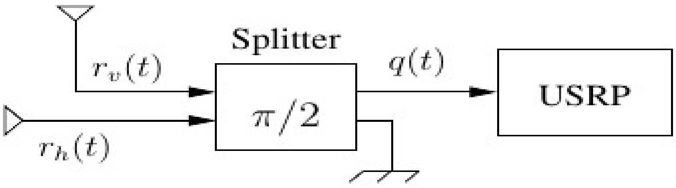


Fig 1.4 Receiver with Two antennae and Phase Splitter/ Combiner

Where the various signals are given by

$$r_v(t)=h(t)a(t)\cos\psi(t)\cos(w_c t+\theta(t))$$
$$r_h(t)=h(t)a(t)\sin\psi(t)\cos(w_c t+\theta(t))$$

and $\psi(t)=w_1 t+\varphi(t)$

Narrow band fading and hence single tap channel coefficient is assumed in the project, as the distances of propagation being considered(< $100m$) are so small that all the significant multipaths have approximately the same delay as that of the Line-of-Sight (LOS) path. These are the very essential details to understand the concept at first hand. The details of signal extraction from this are given in the previous reports. The signal processing part pertaining to frame synchronisation, channel estimation and frequency offset correction is given in chapter 3 on signal processing.

1.3 Organisation of Report

In Chapter 2, the performance of two other radio schemes i.e. IITM designed Fixed Polarisation Wave (FPW) scheme and existing standard Zigbee radio (another fixed polarisation radio) is compared with IITM designed Rotating Polarisation Wave (RPW) radio under various conditions.

In Chapter 3, the various signal processing aspects are discussed along with the transmitter and receiver building blocks. The building blocks illustrate the general flow of operations that are performed in the transmitter and receiver.

In Chapter 4, the existing program were modified to implement the transmitter and receiver in Real Time to are towards Half Duplex point to point link and a star topology based communication system. The UHD interface, C/C++ environment and libraries, USRP related aspects of real time implementation are also discussed in depth.

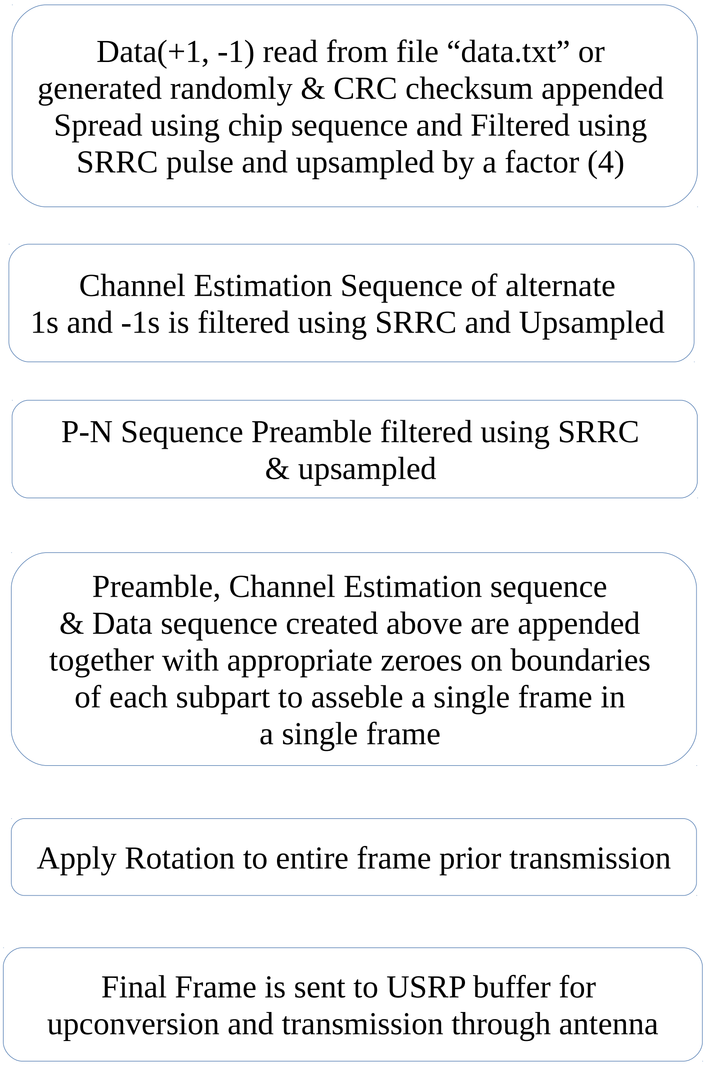
CHAPTER 2

SIGNAL PROCESSING

The main signal processing operations at the transmitter and receiver are signal spreading, pulse shaping, upsampling / oversampling, downsampling, channel estimation etc. It is considered important that the sequence of operations done at transmitter and receiver be discussed at first and later the individual signal processing operation can be discussed in required depth.

2.1 Transmitter and Receiver Operations Sequence

The sequence of operations at the transmitter and receiver is summarised using the Fig. 2.1 and Fig. 2.2.



Data(+1, -1) read from file “data.txt” or generated randomly & CRC checksum appended Spread using chip sequence and Filtered using SRRC pulse and upsampled by a factor (4)

Channel Estimation Sequence of alternate 1s and -1s is filtered using SRRC and Upsampled

P-N Sequence Preamble filtered using SRRC & upsampled

Preamble, Channel Estimation sequence & Data sequence created above are appended together with appropriate zeroes on boundaries of each subpart to asseble a single frame in a single frame

Apply Rotation to entire frame prior transmission

Final Frame is sent to USRP buffer for upconversion and transmission through antenna

Fig. 2.1 Operations at Transmitter

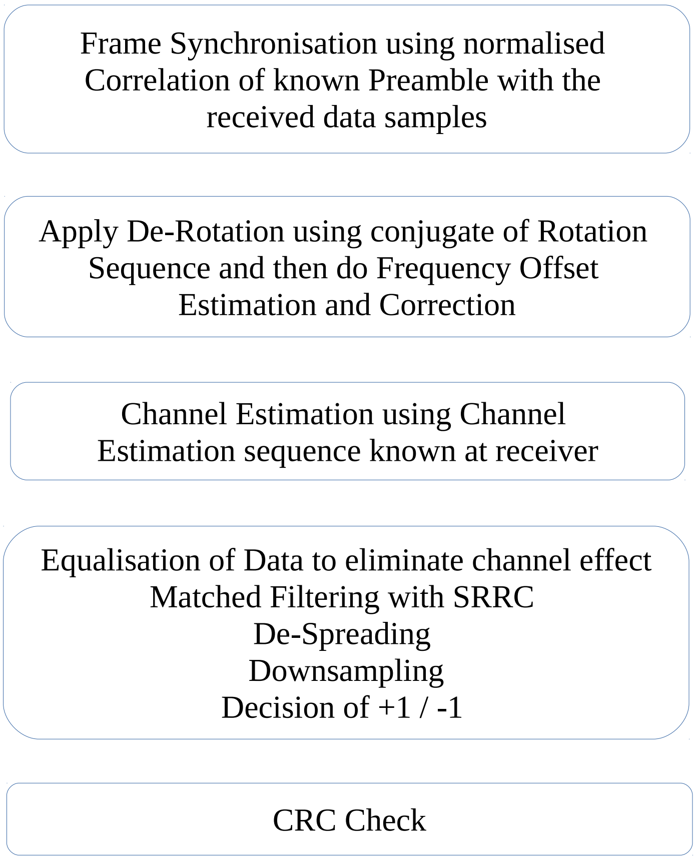


Fig. 2.2 Operations at Receiver

2.2 Pulse Shaping

In order to avoid inter-symbol interference, ideal Nyquist Pulse is recommended theoretically. However, the ideal Nyquist pulse (Sinc pulse), though band limited, has a long tail. If the receiver is not synchronised in time, the filter taps with value considerably greater than zero can create a significant inter-symbol interference. Also the ideal filter has long length due to long tail which is not very desirable for practical implementation. A SRRC (Square Root Raised Cosine) Pulse is used for in typical practical implementations. The SRRC pulse satisfies the Nyquist criterion and still has close to zero filter taps beyond 2T(T-symbol duration) thereby avoiding inter-symbol interference due to timing synchronisation problem at receiver. The tail of SRRC decays faster and therefore, the filter length is also small. Given that the raised cosine filtering simplifies the practical implementation (by making the receiver more robust to timing synchronization errors), the increase in transmission bandwidth may be a small price to pay. For simplicity the detailed equations of Nyquist Pulse and SRRC pulse are not given as they are found in any standard literature on Communication Systems. A standard “*rcosdesign*” function of Matlab is used for generating this SRRC Pulse.

2.3 Generation of Various Sequences

Preamble. A P-N sequence (length 256) is filtered using SRRC and oversampled by a factor of 4. It is generated using Matlab and stored in a file for use in transmitter and receiver software.

Channel Estimation Sequence. A channel estimation sequence of alternate +1s and -1s is used. The sequence is filtered through SRRC and oversampled. This final sequence is also stored in a file for use in transmitter and receiver. The details of the channel estimation are discussed in subsequently.

Data Sequence. A predetermined data sequence could be read from file or generated randomly at the transmitter. The data sequence of +1s and -1s is spread with a chip sequence of 64 length. It is then oversampled and filtered using SRRC. The CRC checksum is calculated for the data and appended before the spreading stage. The data which contains checksum is then spread, oversampled and filtered.

Rotation Sequence. The sequence for rotation frequency of w_1 is samples of $\exp(j w_1 \frac{n}{f_s})$. The channel estimation sequence and data portions are multiplied with the rotation sequence. Only Preamble is not being rotated as the frame synchronisation algorithm will need some modifications if preamble is rotated. The rotation of preamble and the modification of frame synchronisation is envisaged in the next stage of the project. The de-rotation sequence is the complex conjugate of above. The de-rotation is required at the receiver to cancel the effect of rotation.

2.4 Frame Synchronisation

It is very important to synchronise with the frame start at the receiver to decode the data correctly under reasonably good SNR. If the frame starts are wrong the decoding will definitely be incorrect. Automatic Gain Control (AGC) feature of USRP Receiver is used effectively to arrive at a robust frame synchronisation algorithm. The steps that are followed in the frame synchronisation are enumerated below.

(a) In order to determine the threshold, the receiver alone is switched on and the noise samples are correlated with the preamble with the following method. Thus, the correlation output levels are known only when noise is present. As a safety margin, a suitable level above this noise – preamble correlation level is chosen as threshold. The data samples will have to cross this threshold to be detected as Frame.

(b) The received data samples ($\vec{Y}$ of length 4088) are correlated with the preamble ($\vec{X}$ of length 1030) and their magnitude stored in $\vec{A} = |(\sum \vec{X}\vec{Y})|$ of length 1030+4088-1. The correlation is done by sliding the preamble vector (1030) over the received signal vector (4088) by one step at a time. After sliding, overlapping samples of the two vectors are multiplied and added to compute the one sample of the correlation output. The different instances of overlap are shown in the diagram below. The normalisation should result in a highest correlation output at an instance when preamble in the received signal vector fully overlaps with the known preamble that is sliding. If each such output sample has to be normalised, it needs to be divided by the power of preamble and received samples which participated in the particular instance of sliding. It is to note that we calculate the preamble power at once by calculating its norm and use the whole preamble power to normalise in each output as we want output to be maximum at only one instance when entire preamble enters the window and matched perfectly with preamble present in the received signal vector. However, the power of received samples is calculated only for the received sample that overlapped in the particular sliding window. To calculate the power of received signal like this the received signal magnitude is convolved with a vector of ones (1s) of length equal to preamble. The convolution output has the power of so many received samples as those participated in the respective correlation output.

Fig. 2.3 Correlation Procedure

(c) The norm of Preamble is calculated as $P=\sum \vec{P}.(\vec{P})^{H}$

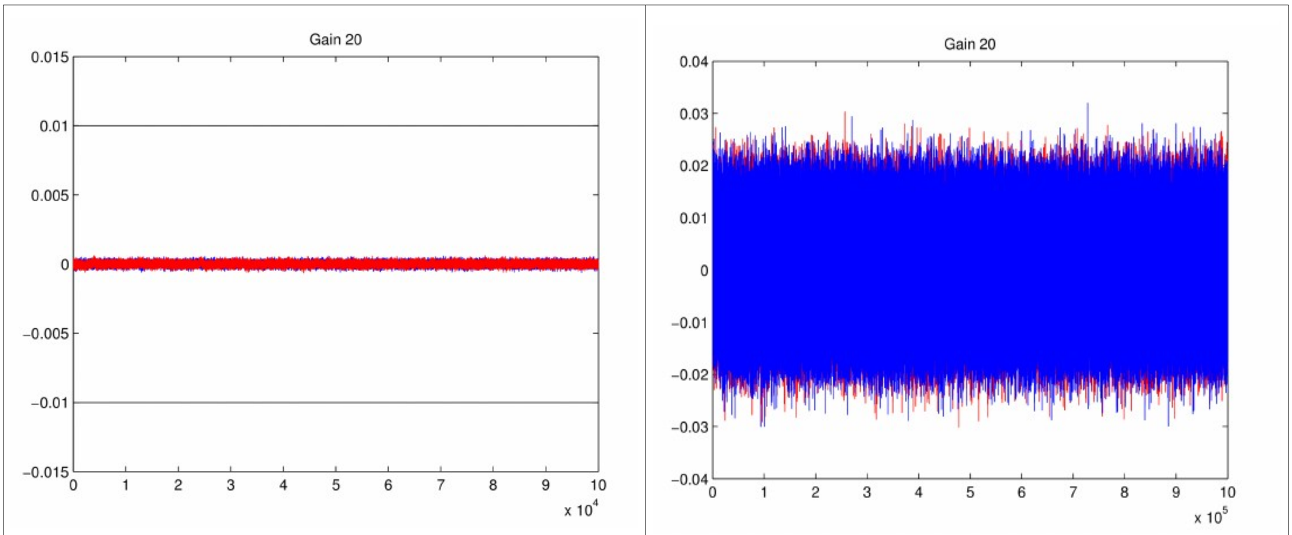
(d) The square of magnitude of the received signal samples $\vec{Y}.(\vec{Y})^{H}$ is convolved with a vector of ones $\vec{V}$ of size equal to that of preamble to get power of received signal at each instance of correlation done in step (b) $\vec{R}=\vec{V}*(\vec{Y}.\vec{Y}^{H})$ (* denotes convolution).

(e) Normalisation is done as follows:

$$\vec{F}=\frac{\vec{A}}{\sqrt{(P.\vec{R})}}$$

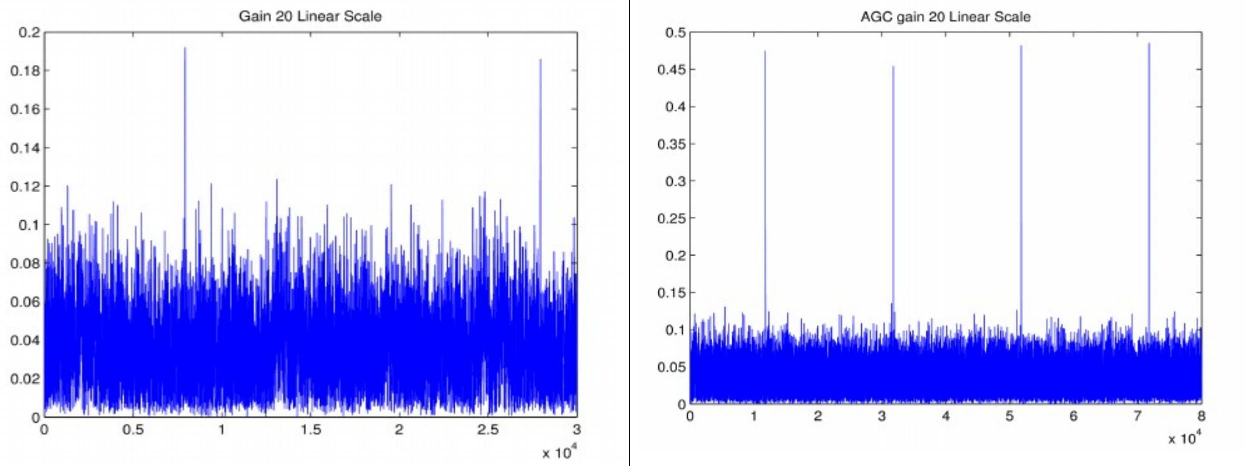
(f) The maximum value present in the resultant vector $\vec{F}$ is compared with the threshold. If this maximum value is greater than the threshold then its index marks the end of the preamble of the received frame.

Thus, the frame is located correctly with the index of this and further processing can be undertaken on the remaining of the frame to decode. The Fig. 2.4 explains the effect of AGC on the received samples as well as the correlation output. In Fig.2.4 (a), (b) it can be seen that the signal is boosted significantly. In Fig.2.4(c), (d) it can be seen that the Correlation output peaks have increased from 0.18 to 0.45 when preamble is present in the received signal. However, the correlation output values with other irrelevant samples have remained around same levels. This is the result of normalised correlation and AGC together. The same correlation outputs are plotted in dB scale in Fig.2.4 (e), (f) to mark the stark differences between the peak values and other adjacent values.



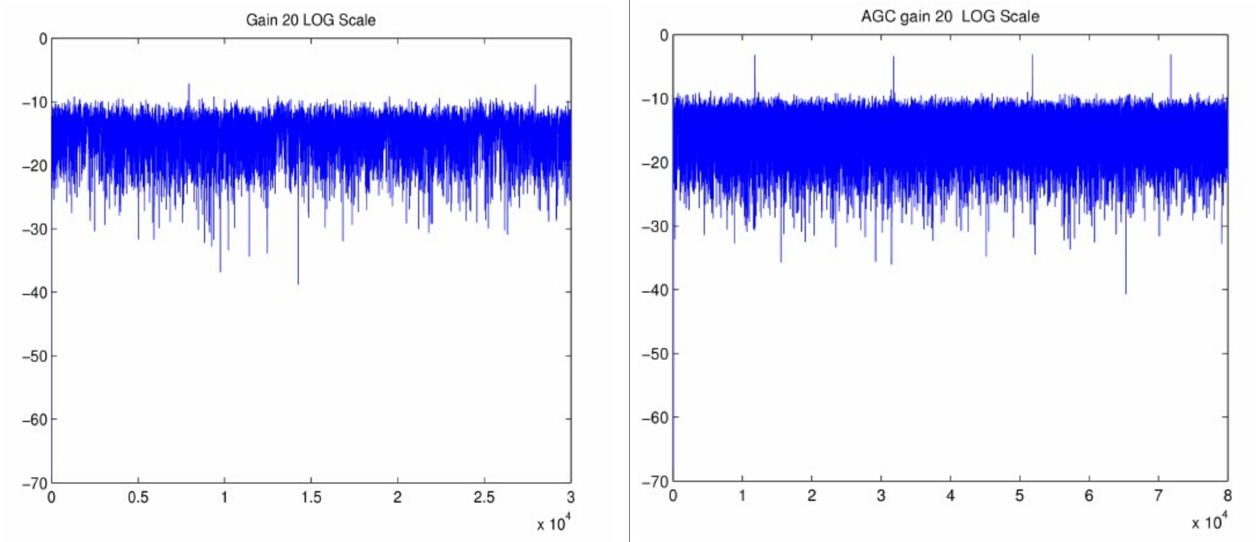
(a) Received Samples with AGC OFF

(b) Received Samples AGC ON



(c) Correlation Output with AGC OFF (Linear Scale)

(d) Correlation Output with AGC ON (Linear Scale)



(e) Correlation Output with AGC OFF (Log Scale)

(f) Correlation Output with AGC ON (Log Scale)

Fig. 2.4 AGC Effects on Received Signal

2.5 Frequency Offset Estimation and Correction

The channel estimation sequence known to the receiver is used for frequency offset correction as well. The Schmidl – Cox algorithm is being used for frequency offset estimation and correction. The sequence $a(n)$ has 5 identical blocks of 200 samples each. Each 200 sample block has 100 plus ones (+1s) and 100 minus ones (-1s).

$$a(n)=\underbrace{\underbrace{\underbrace{(+1+1\cdots)}_{100}\underbrace{(-1-1\cdots)}_{100}}_{200}\cdots\cdots\cdots\cdots\cdots\cdots\underbrace{\underbrace{(+1+1\cdots)}_{100}\underbrace{(-1-1\cdots)}_{100}}_{200}}_{1000}$$

The $a(n)$ after filtering with SRRC and other operations becomes $x(n)$. The structure of $x(n)$, that every sample is identical to 200th sample ahead of it in ideal situation, can be used to estimate the offset.

Let's say $$x(n)=\exp(j2\pi f_c\frac{n}{f_s})$$

is received as $$\hat{x}(n)=\exp(j2\pi(f_c+f_{off})\frac{n}{f_s})$$ with frequency offset f_{off}

The complex conjugate multiplication of received sample with ideal sample is extended for the entire vector for better averaging.

$$\sum_{n=1}^{800} x(n).\hat{x}(n+200)^H=\exp(-j2\pi f_{off}\frac{200}{f_s})$$

the angle ($\sphericalangle$) of which should be multiplied by $\frac{-f_s}{(2.\pi\times200)}$ to get f_{off} .

Angle of the sum of the resulting vector elements is calculated and converted appropriately to know the frequency offset present as shown above. Once the frequency offset is known the remaining data samples including channel estimation sequence are compensated for this offset.

In absence of the Frequency Offset Estimation and correction mechanism, the TX and RX USRP would need to be synchronised with the help of external clock from same source.

2.6 Channel Estimation and Equalisation

Once the Frame is detected, the immediate task is to estimate the channel coefficients so that the data can be equalised prior to decoding stages. Lets say the received signal is given by

$$\bar{y}=\bar{h}\bar{x}+\bar{w}$$

where,
$\bar{x}$ – transmitted channel estimation sequence
$\bar{h}$ – channel coefficients
$\bar{w}$ – Noise at receiver

To estimate the channel coefficients we divide the received signal with the $\bar{x}$ we get the estimate as follows:

$$\hat{\bar{h}}=\bar{h}+\frac{\bar{w}}{\bar{x}}$$

The Mean Squared Error (MSE) using the above method would be,

$\frac{\sigma_z^2}{E_x}$ where σ_z^2 - noise variance and

E_x - signal power

The data sequence is equalised by dividing the it with the estimates $\hat{\bar{h}}$ obtained above. The rotating polarisation wave scheme has rotation frequency of 100KHz. It is important to note that that 100KHz frequency of rotation is used in the advanced stages of the project i.e. Real Time implementation onwards. However, for the machines laboratory experiments which are explained in chapter 3. The rotation Frequency used is 3906.25Hz. The explanation of calculation of rotation frequency is given in that chapter.

For the rotation frequency of 100Khz we expect only 10 different channel coefficients to be present in one rotation. However, the data sequence is of large length. Therefore, in one iteration only 10 data samples are equalised by these channel coefficients. In next iteration next 10 data samples are equalised and so on.

2.7 Matched Filtering and Decoding

The equalised data is filtered using SRRC pulse for matched filtering purpose. The data is downsampled by a factor of 4 because it was oversampled by same factor at the transmitter. The downsampled data is despread by multiplying it with same PN sequence and summing all 64 elements to get that one data bit.

2.8 Decision and CRC Check

If the real part of final data samples available after de-spreading is greater than 0 then it is taken as +1 otherwise as -1. CRC check sum is calculated for this final decoded data and if the check sum is true it indicates that correct data has been received.

CHAPTER 3

COMPARISON OF FPW, RPW AND ZIGBEE RADIO

The computation of rotation frequency and its effect on the channel coefficients are explained in details before proceeding with the other experiments set-up and results.

3.1 Determining the Correct Frequency of Rotation

In order to carry out more experiments to analyse the performance of FPW vis-a-vis RPW, the calculation of rotation frequency is reconsidered. The Given parameters are as follows:

$$Length\,of\ spreading\,sequence\,(L)=64$$
$$sampling\ frequency\,(f_s)=1\,Msps$$
$$Oversampling\,factor\,(osf)=4$$

The rotation frequency is previously calculated by the formula

$$Rotation\,Frequency=\frac{f_s}{L}=156.25\,KHz$$

It is to ensure that 64 samples are taken in one rotation to see all 64 channels. Or in other words signal has to be sampled 64 times and one sample is of 1 micro seconds. However, the final sequence is oversampled by a factor of 4 which has not been considered in above equation. Therefore, the rotating frequency has to be recalculated using

$$Rotation\,Frequency=\frac{f_s}{(L\times osf)}=3906.25\,Hz$$

The experiments have been conducted with these two rotation frequencies and the channels observed in these two cases are given in the figures below. The y-axis represents channel coefficient value. The x-axis represents the particular channel coefficient number from the 64 channel coefficients being estimated.

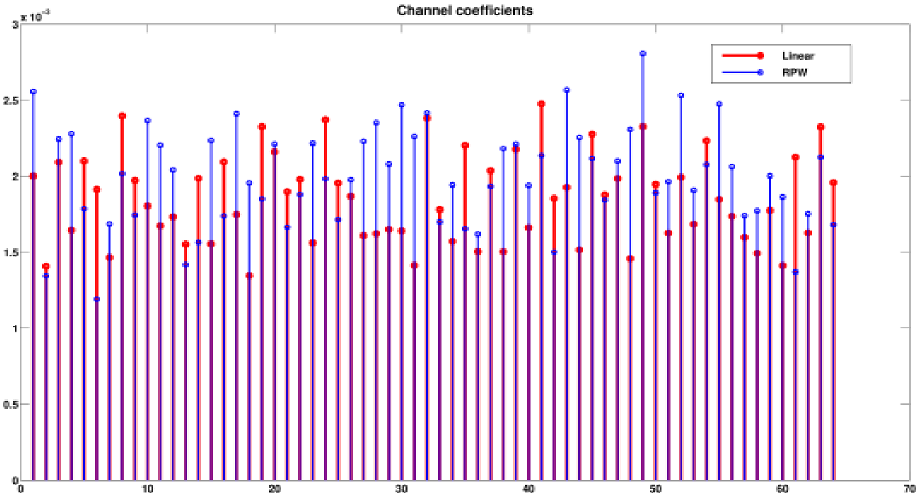


Fig 3.1 Channels coefficients with rotation frequency of 156.25Khz

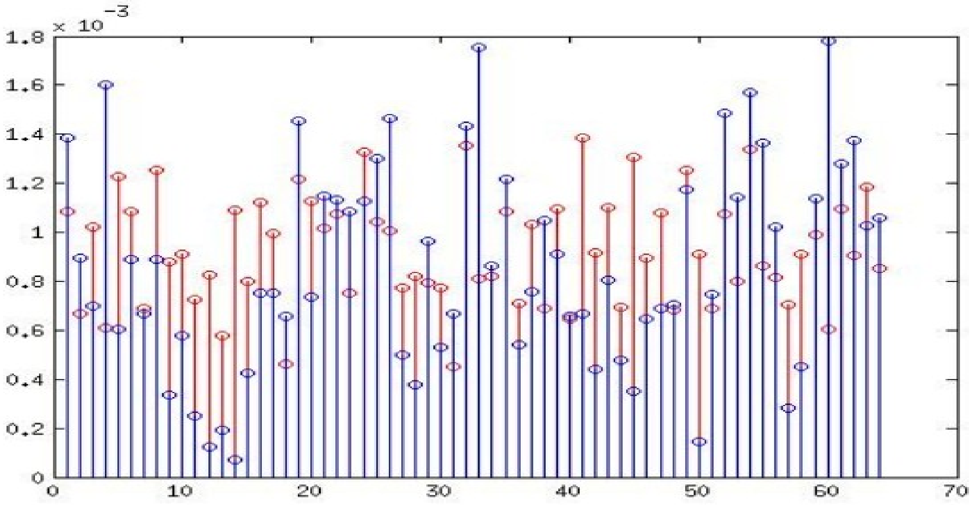


Fig 3.2 Channels coefficients with rotating frequency of 3906.25Hz

It is seen from the channel plots that the channel variations are more which is justified as only 3906.25Hz as rotation frequency allows the 64 samples to be taken in one rotation. Therefore, rotating frequency is chosen as 3906.25hz for future experiments given the parameters above.

3.2 Experiment Set-up

In order to analyse performance, a series of experiments are conducted. The general set up is explained in this section. This set up with minor changes is used in all experiments depending on whether the experiment is wired or wireless. However, the basic structure remains same. The Fig 3.1 and Fig 3.2 show the transmitter and receiver for the wireless experiment respectively. The set up is implemented using the USRP devices and GNU Radio application on Linux platform. The frame structure used by transmitter and receiver is as follows. However, this frame structure has been changed later on to suit the requirements of real time implementation.

Channel Pilots (bits)	Preamble (bits)	Data (bits)
10	4	20

Each bit is spread by a 64 length sequence and oversampled by factor of 4 to create actual frame. This frame is stored in a file and repeatedly transmitted for continuous transmission through USRP using GNU Radio application file.

The transmitter of fig 3.3 using GNU radio application works as follows:

(a) Fetches the source files
(b) Gives it to USRP for upconversion
(c) Output of TX port(s) of USRP is given to antenna.

In case of FPW only one transmitter is connected to only one antenna. In case of RPW, the data is being transmitted on two transmitter ports through two antennas. The figures show the RPW schematic. In case of reception, the output of two antennas is given to a phase shifter / combiner where it provides 90^0 phase shift to one of the antenna signal connected to its port 1 and adds it with other antenna signal to give one combined output. In case of FPW reception, only one antenna signal is connected to the USRP RX through phase shifter with all other unused ports terminated.

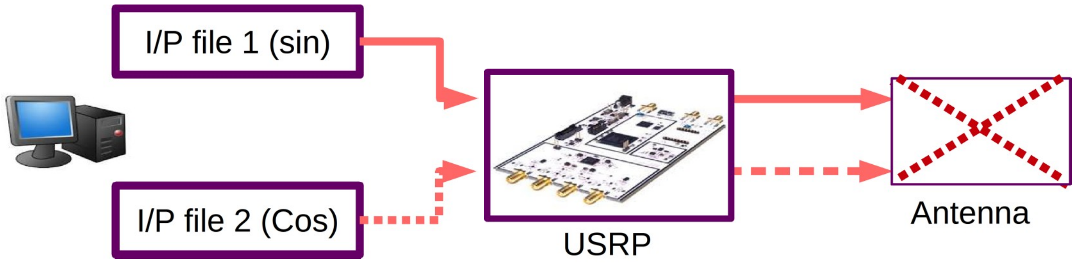


Fig 3.3 Transmitter

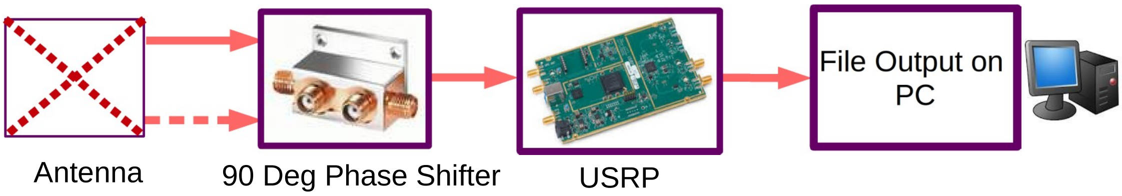


Fig 3.4 Receiver

3.3 Understanding The Effect of Antenna Angle

A basic experiment was done to understand the effect of antenna orientation on the received signal in case of FPW scheme. The channel coefficients observed at various angles are given in the the Fig. 3.5. Antenna one and two are used to cover angles of 180^0 with steps of 15^0 each. The y-axis gives channel magnitude in dB. The x-axis gives channel coefficient number out of total 64 channels being estimated.

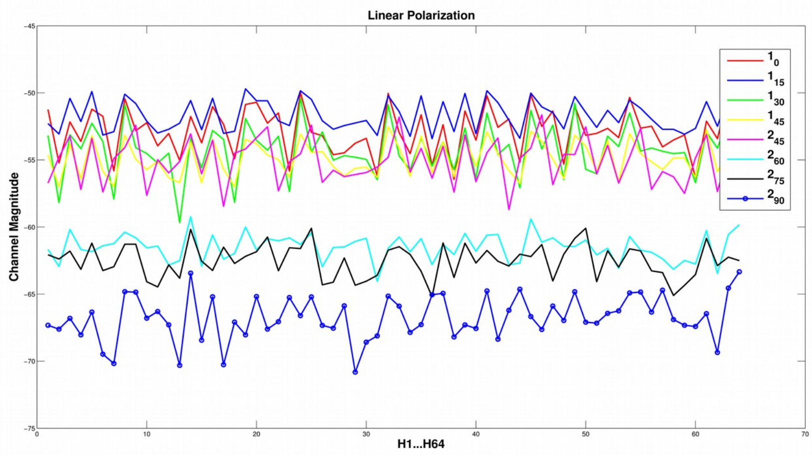


Fig. 3.5 Channel Coefficients at different antenna orientations

The fact that signal is heavily dependent on antennae rotation, is now established from the these channel plots as well as the signal spectrum seen on the GNURadio application. Therefore, it is decided to further undertake preliminary experiments to know Bit Error Rates offered by Fixed Polarisation scheme and Rotating Polarisation scheme in

Machines Lab as well as ESB 218 at an antenna orientation where the signal received in Fixed Polarisation is weak relative to other orientation offering better signal. The procedure is described below.

(a) FPW transmitter is set and the RX antenna is aligned such that it receives low signal.

(b) This position of antenna is fixed and data is collected for both Fixed polarisation and RPW at three TX gains i.e. 36dB, 42dB, 48dB.

(c) The experiment is repeated in the same manner and non line of sight condition in the ESB 218 Lab to revalidate the results.

(d) The data files, thus obtained, are processed to assess the Bit Error Rate for individual gains and the plots are illustrated below.

3.4 BER Plots of Simple Experiment

BER plot of the simple experiment at machines lab is given below:

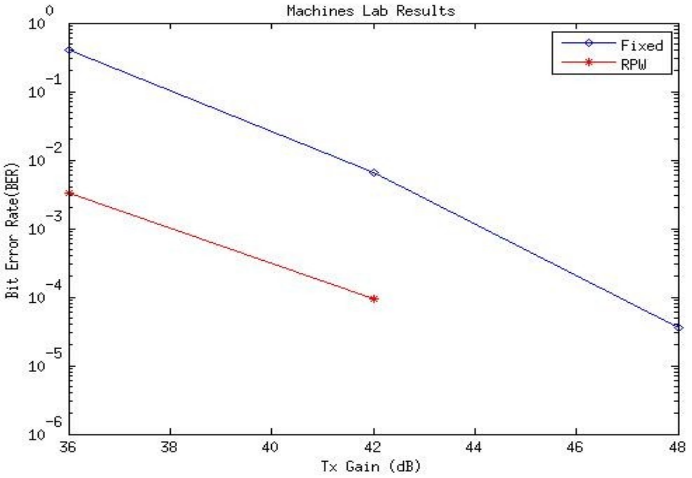


Fig 3.6 Bit Error Rate of FPW and RPW in Machines Lab Non Line of Sight with 1m distance

(Note: If BER goes to zero that point can not be shown in plot by Matlab)

The BER curve of the similar non line of sight (NLOS) simple experiment in ESB 218 lab is given below:

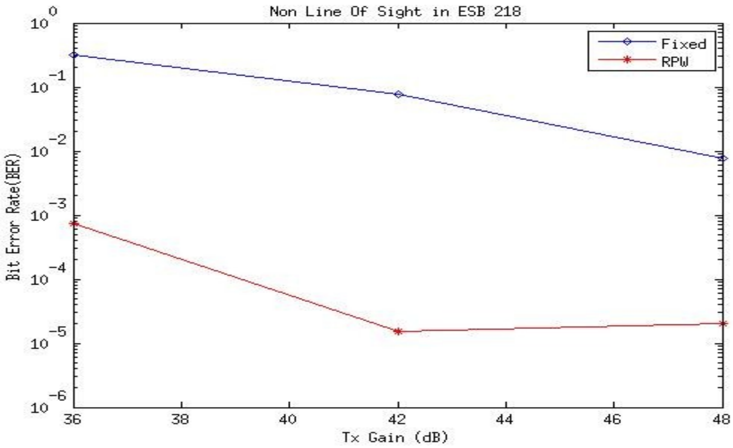


Fig 3.7 Bit Error Rate of FPW and RPW in ESB 218 Non Line of Sight with 1m distance

It is seen that if FPW receiver receives poor signal due to antenna orientation, it has offered higher Bit error rates BERs than the RPW receiver which is also receiving with the same antenna orientation but in RPW configuration. This experiment is preliminary in nature to demonstrate the effects of antenna orientation over the performances of various radio schemes in question. However, the scope of experiments is increased further in order to establish some conclusions related to effects of antenna orientation on the Fixed Polarisation Wave radio, Rotating Polarisation Wave radio and another standard radio based on Fixed Polarisation scheme i.e. Zigbee radio.

The enhanced scope includes increased number of locations in machines lab which would also be non line of sight (NLOS). The antenna orientation would be set in two positions as explained subsequently. The layout of machines lab depicting the transmitter and receiver locations is given in the figure below:

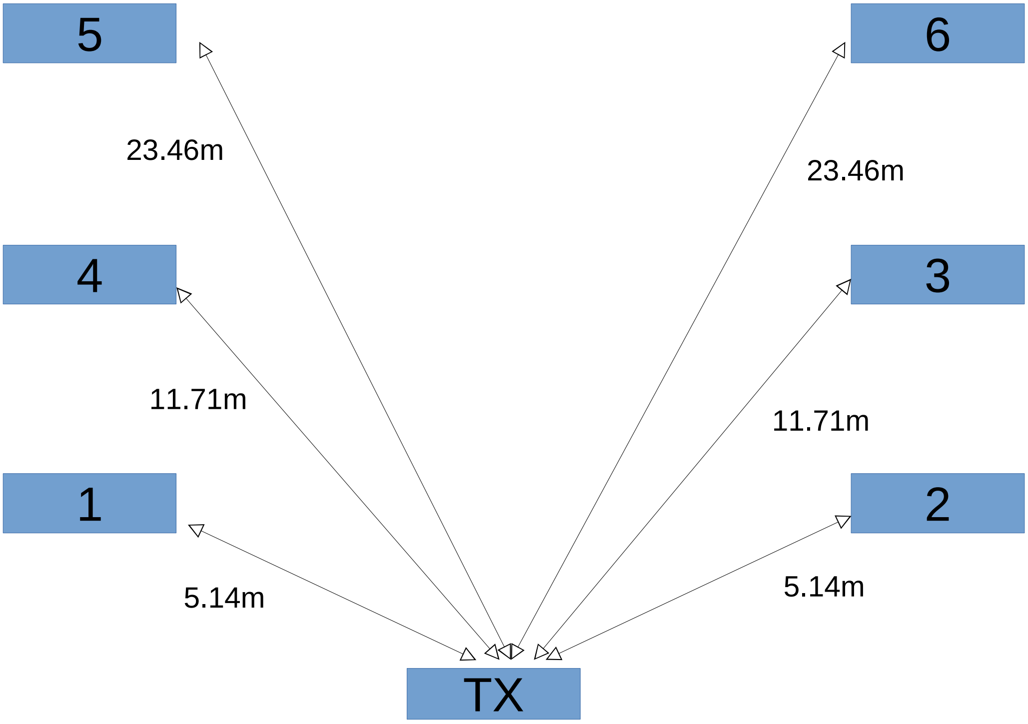


Fig 3.8 Layout of Machines Lab with significant obstructions (NLOS paths)

3.5 Major Experiment at 430 MHz

In case of Fixed Polarisation Wave (FPW), the received signal strength has varied with the orientation of the antenna as shown earlier. The antenna orientation is changed manually and the signal strength is observed on GNURadio application at the receiver USRP. Among the possible antenna orientations, an orientation with strong signal reception of FPW is identified and called *Good* and an orientation with weak signal reception of FPW is identified and called *Poor* orientation.

At each location in lab, the antenna is rotated to achieve the *Good* FPW orientation and the data is collected for the three radio schemes i.e. FPW, RPW and Zigbee for comparison. The same experiment is repeated at the same location with antenna orientation changed in *Poor* FPW orientation. The transmitter gain is varied in a specified range in each experiment to analyse the Bit Error Rate (BER). As the distance between the transmitter and receiver is increased beyond locations 1 and 2, the received signal strength would decrease significantly. The resulting BER is observed to be very high in case of *Poor* at location 4 for the transmitter gain range of 36 to 48 dB as seen in the BER table 2. Accordingly, the transmitter gain range is extended further to observe the decrease in BER at higher gains at location 3 (distance same as Location 4). From the BER readings of location 3 it is considered that the transmitter gain range of 51 to 66 dB should suffice to demonstrate a BER trend in the location 5 and 6. Therefore, the transmitter gains were chosen as given in the Table 3.1.

Range of TX gains (in dB)	**Positions**
36 to 48 (steps of 3)	1, 2
36 to 60 (steps of 3)	3
36 to 48 (steps of 3)	4
51 to 66 (steps of 3)	5, 6

Table 3.1 TX Gain Range vis-a-vis RX Location

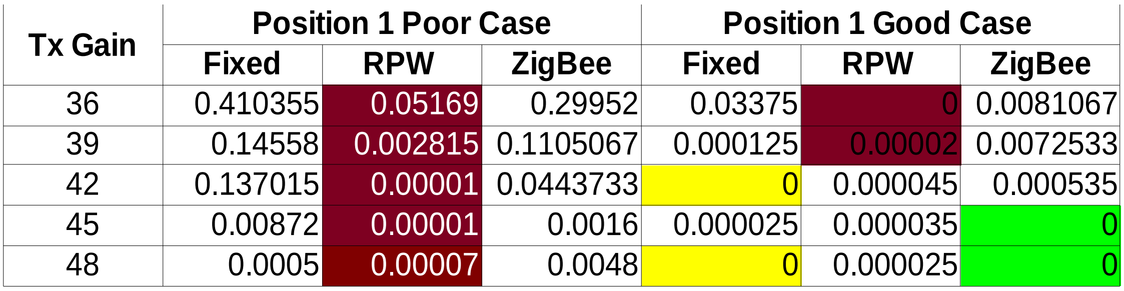
Receiver Gain was kept at 35dB throughout the experiment with a centre frequency at 430 MHz

3.6 Comparison of BER Results of Major Experiment at 430 MHz

All the data files obtained are processed to obtain the Bit Error Rates in all cases. The Bit Error Rates are given in the tables 3.3 (a) to (f) for various locations. The different colours are used in the table to highlight the scheme which has given least BER i.e. best performance for particular location, Tx Gain and antenna orientation (Good or Poor case). The colours and corresponding scheme are given in Table 3.2.

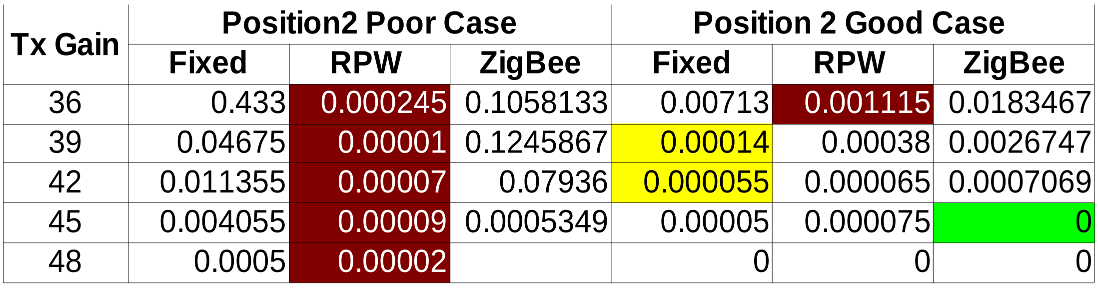
XXXXXXX	**RPW**
XXXXXX	**FPW**
XXXXXXX	**Zigbee**

Table 3.2: Colours used for various schemes



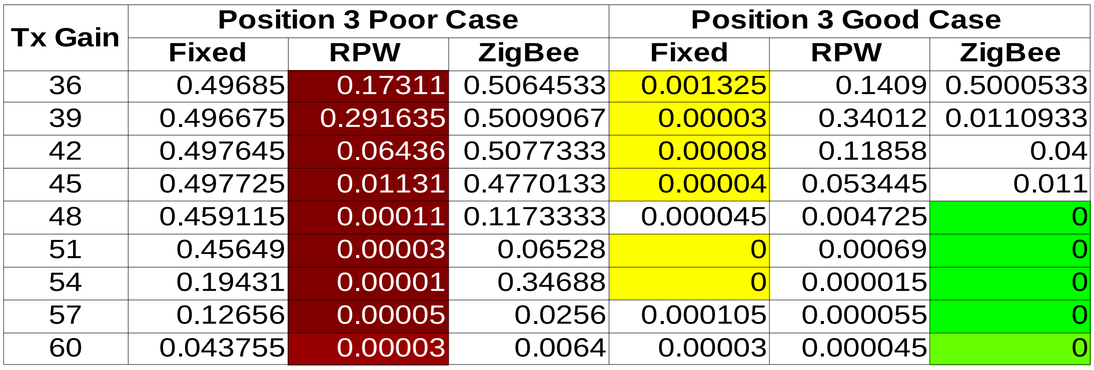
Tx Gain	Position 1 Poor Case			Position 1 Good Case		
	Fixed	RPW	ZigBee	Fixed	RPW	ZigBee
36	0.410355	0.05169	0.29952	0.03375	0	0.0081067
39	0.14558	0.002815	0.1105067	0.000125	0.00002	0.0072533
42	0.137015	0.00001	0.0443733	0	0.000045	0.000535
45	0.00872	0.00001	0.0016	0.000025	0.000035	0
48	0.0005	0.00007	0.0048	0	0.000025	0

(a)



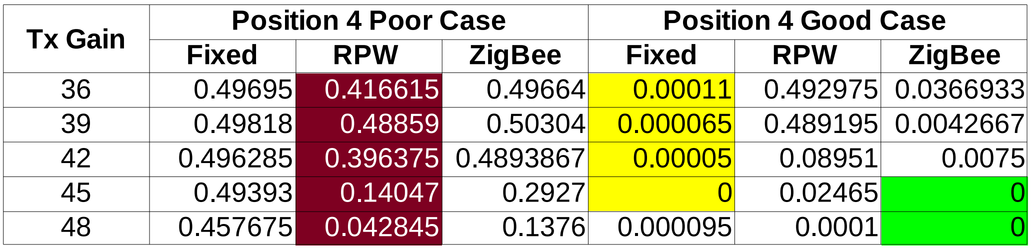
Tx Gain	Position2 Poor Case			Position 2 Good Case		
	Fixed	RPW	ZigBee	Fixed	RPW	ZigBee
36	0.433	0.000245	0.1058133	0.00713	0.001115	0.0183467
39	0.04675	0.00001	0.1245867	0.00014	0.00038	0.0026747
42	0.011355	0.00007	0.07936	0.000055	0.000065	0.0007069
45	0.004055	0.00009	0.0005349	0.00005	0.000075	0
48	0.0005	0.00002		0	0	0

(b)



Tx Gain	Position 3 Poor Case			Position 3 Good Case		
	Fixed	RPW	ZigBee	Fixed	RPW	ZigBee
36	0.49685	0.17311	0.5064533	0.001325	0.1409	0.5000533
39	0.496675	0.291635	0.5009067	0.00003	0.34012	0.0110933
42	0.497645	0.06436	0.5077333	0.00008	0.11858	0.04
45	0.497725	0.01131	0.4770133	0.00004	0.053445	0.011
48	0.459115	0.00011	0.1173333	0.000045	0.004725	0
51	0.45649	0.00003	0.06528	0	0.00069	0
54	0.19431	0.00001	0.34688	0	0.000015	0
57	0.12656	0.00005	0.0256	0.000105	0.000055	0
60	0.043755	0.00003	0.0064	0.00003	0.000045	0

(c)



Tx Gain	Position 4 Poor Case			Position 4 Good Case		
	Fixed	RPW	ZigBee	Fixed	RPW	ZigBee
36	0.49695	0.416615	0.49664	0.00011	0.492975	0.0366933
39	0.49818	0.48859	0.50304	0.000065	0.489195	0.0042667
42	0.496285	0.396375	0.4893867	0.00005	0.08951	0.0075
45	0.49393	0.14047	0.2927	0	0.02465	0
48	0.457675	0.042845	0.1376	0.000095	0.0001	0

(d)

Tx Gain	Position 5 Poor Case			Position 5 Good Case		
	Fixed	RPW	ZigBee	Fixed	RPW	ZigBee
51	0.49436	0.485186	0.5085867	0.337605	0.227635	0.3771733
54	0.460925	0.261275	0.35584	0.078395	0.397255	0.02176
57	0.233105	0.16037	0.16896	0.012185	0.17408	0.00896
60	0.078765	0.057385	0.2973867	0.002345	0.074845	0
63	0.012955	0.01406	0.0648533	0.00004	0.00866	0
66	0.002235	0.00002	0.0183467	0.000045	0.00017	0

(e)

Tx Gain	Position 6 Poor Case			Position 6 Good Case		
	Fixed	RPW	ZigBee	Fixed	RPW	ZigBee
51	0.34278	0.03191	0.1913	0.06467	0.06702	0.1762133
54	0.13201	0.001495	0.1228	0.002055	0.00379	0.0379733
57	0.007535	0.008725	0.0348	0.000575	0.001905	0.0038783
60	0.002215	0.000825	0.0203	0.000055	0.00002	0.0038783
63	0.00064	0.000015	0.0016	0.000045	0.00007	0
66	0.00006	0.000015	0.0005349	0.0013	0.00078	0

(f)

Table 3.3 : (a) Location 1 BER readings (b) Location 2 BER readings (c) Lcoation 3 BER readings (d) Location 4 BER readings (e) Location 5 BER readings (f) Location 6 BER readings

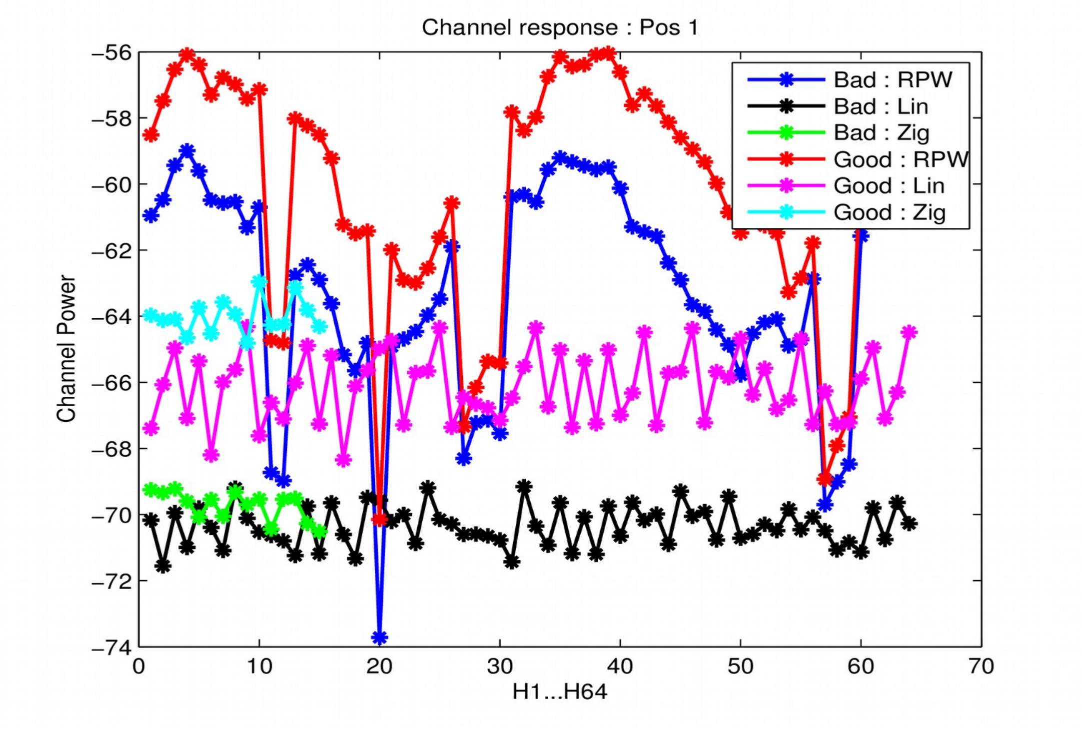
It is seen from the readings of Table 3.3 (a) to (f) that RPW has performed better than the other two schemes at all locations when the antenna orientation results in *poor* FPW reception. The performance of FPW and ZigBee is heavily dependent on the antenna orientation. Before FPW or ZigBee is installed, it is required to carry out measurements and fix the antenna orientation to achieve satisfactory performance. It also means that if the scattering environment changes, the same antenna orientation may no longer be optimal. However, it is seen that the RPW performance is not drastically affected by change in antenna rotation and it continues to perform satisfactorily irrespective of the antenna orientation.

3.7 Channel Coefficients' Plots

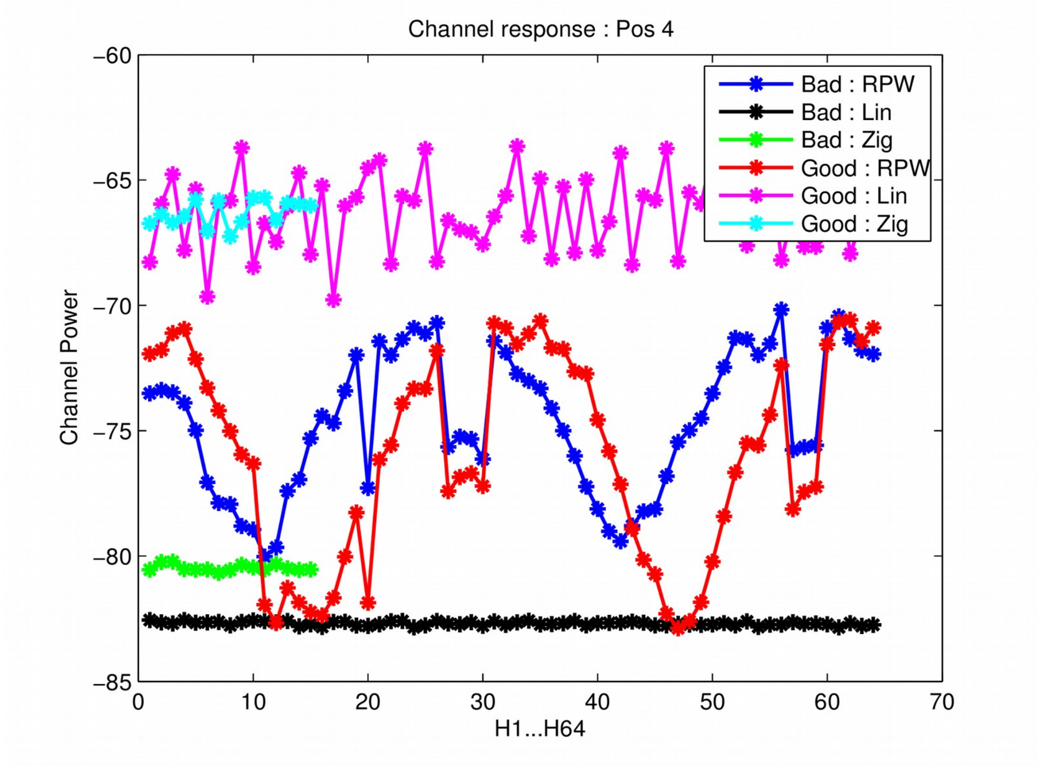
The figures Fig 3.9 (a) to (c) illustrate the channel powers observed at locations 1, 4 and 6 for the three schemes RPW, Fixed and Zigbee when antenna orientation is at *Good* and *Poor* positions. RPW is able to estimate the channel coefficients under both Good and Poor orientations and the gap between them is small. Whereas the channel estimation for FPW is even better when the reception is *Good*. However, the performance of FPW suffers greatly when the reception is *Poor*. It is to be noted that the Y-axis represents the channel powers in dB scale and X-axis shows the number of channel coefficient. In case of FPW and RPW the channel coefficients are 64, however, it is 15 for Zigbee.

The same experiments are repeated in machines lab with no change in set-up and locations except for the carrier frequency changed to 860MHz. The results and conclusions thereof have been similar at carrier frequency of 860MHz also. Therefore, these results are not reproduced here to maintain the brevity of the report. However, those results are available in the interim reports on the lab computer.

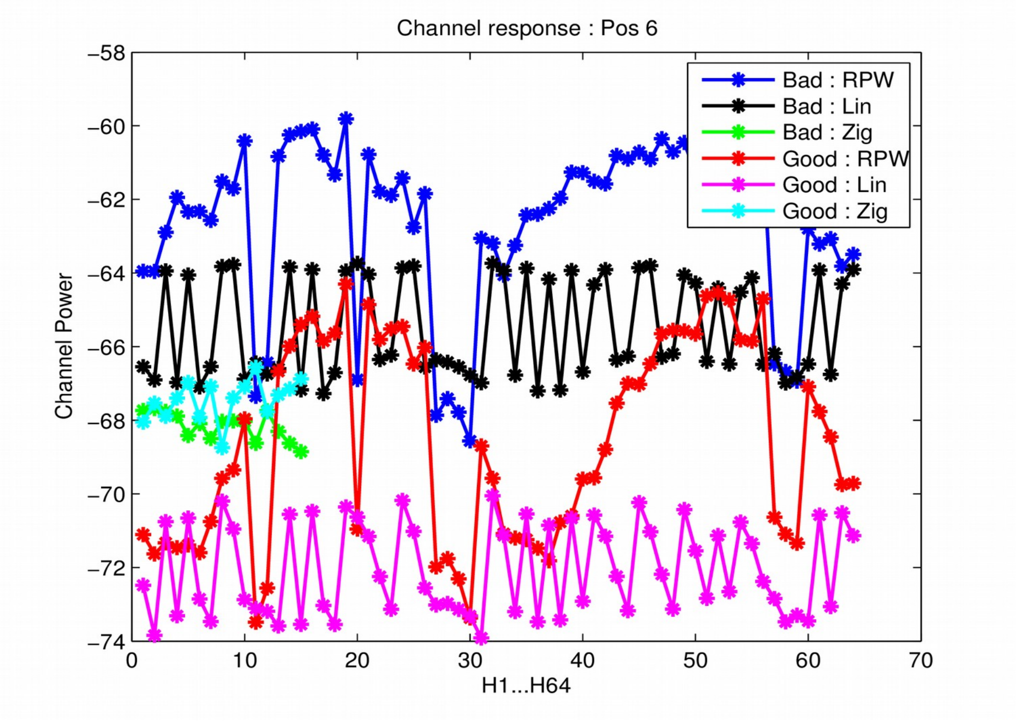
On completion of exhaustive experiments to analyse the performance of FPW, RPW and Zigbee, it is envisaged to carry out implementation of the transmitter and receiver in real time using UHD interface and C / C++ environment. Till this point, the transmitter and receiver are implemented using GNURadio application and USRPs. At the receiver, the data is collected in file. This file is input to a C/C++ program which then calculates BER after processing data in the file. However, in real time communication, the data is processed as and when it arrives continuously. The real time implementation is discussed in the following chapters.



(a).



(b)



(c)

Fig 3.9 Channel Powers for (a) Location 1 (b) Location 4 (c) Location 6

CHAPTER 4

REAL TIME IMPLEMENTATION

In order to realize the field trials of the radios, it is required that the radios be working in real time in a star topology with one node serving as hub and 4 other nodes as spokes. The test deployment will compare the performance of rotating polarization radio with ZigBee radio under the same condition. Steps to proceed with field trial could be listed down as

(a) Real-Time implementation of the radios and their testing

(b) Half-Duplex operation of point-to-point link, and

(c) Operation in star topology with one master and 4 client nodes.

This chapter describes the work carried out to achieve real-time implementation. In which, the details pertain mainly to the USRP and UHD interface issues, real time transmitter/receiver implementations, problems encountered and remedies thereof. The details of the modified frame synchronization, frequency synchronisation and other algorithms will be discussed in subsequent chapters. The experiment set up that has been used in earlier experiments prior to real time implementation is depicted in Fig. 3.3 and Fig 3.4 of Chapter 3.

The experiment set up is implemented using the GNURadio application. The data frames to be transmitted are generated and written into files. These files are read from a host PC and fed to the transmitter USRP. The data samples received using receiver USRP are stored in a binary file for further processing like frame decoding, Bit Error Rate (BER) calculation etc. However, it is required to generate and process data frames at the transmitter and receiver respectively in real time to implement any protocol based communication system. Thus, the real time implementation phase of the project has been undertaken subsequently. The development of real time implementation has happened in the following sequence:

(a) UHD(Universal Hardware Driver) interface between USRP and Host PC

(b) Real Time Transmitter

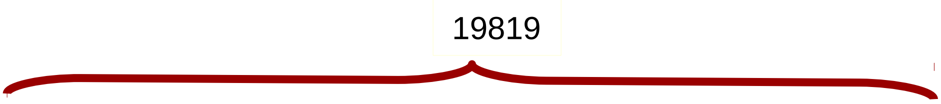
(c) Real Time Receiver

(d) Half Duplex communication

4.1 Real Time Transmitter

4. The frame structure to be followed by the communication system is as shown in figure 3. The subparts of the frame are separated by zeroes to demarcate boundaries. All the numbers indicate the number of data samples. However, it is to be noted that all the sizes of the subparts are variable and can be chosen to suit our requirements. The frame starts with Zeroes(40) followed by Preamble(1030), Zeroes(40), Channel Pilots (1004),

Zeroes (36), Data (17669), Idle time (181). The idle time is the extended time of the frame to maintain the frame repetition rate of 20 ms at the sampling rate of 1Msps (Mega samples per second). At the sampling rate of 1Msps, the frame size of 19819 corresponds to 19.819 ms and with addition of 181 idle time samples it corresponds to 20000 samples or 20 ms. The overhead i.e. preamble and channel pilots etc. occupy roughly 10% of the total frame length.



Zeros	Preamble	Zeros	Channel Pilots	Zeros	Data	Idle time	Next Frame
40	1030	40	1004	36	17669	181	40

Fig 4.1 Frame Structure

Configuring the USRP. The USRP is interfaced using UHD in C/C++ environment. The USRP is configured for various parameters like Tx frequency, Clock rate, Bandwidth, Sampling frequency, choice of transmission port etc. using various commands. A brief code snippet example related to USRP configuration is displayed below.

```
// setting up the clock rate and reference to be selected
tx_usrp->set_clock_source("external");//clock source should be
externaltx_usrp->set_master_clock_rate(10e6);
std::cout << boost::format("Using TX Device: %s") % tx_usrp-
>get_pp_string() << std::endl;

// setting up the Transmitter rate for both channels required by RPW
for ( size_t i=0; i < tx_usrp->get_tx_num_channels(); i++ )
{
 tx_usrp->set_tx_rate(tx_rate,i);
 tx_usrp->set_tx_bandwidth(tx_bw, i);
 cout<< "Actual TX Rate for channel: " << i << " is
"<<(tx_usrp>get_tx_rate(i)/1e6) << " (Msps) "<<endl<<endl;
}

//setting up the Transmitter frequency
tx_usrp->set_tx_freq ( tx_freq, i );
std::cout<< "Actual Frequency for channel " << i << " is " <<(tx_usrp-
>get_tx_freq(i)) << endl<<endl;

// Setting TX gain

 tx_usrp->set_tx_gain ( tx_gain, i );
std::cout<< "Actual gain for channel " << i << " is " <<(tx_usrp-
>get_tx_gain(i)) << endl<<endl;
```

Generation of Data Frame. The subparts of the frame i.e. preamble, channel pilots and data are generated separately and appended together before transmitting it through USRP. The particular data bits could be generated randomly for each frame transmission or it could be read from a .txt file. However, the CRC check sum algorithm implemented at transmitter and receiver allows to know if the data is received correctly. Therefore, it is preferred to generate random data at each frame transmission to check the efficacy of CRC algorithm in the process. The code snippet displaying the assembly of the frame is given below:

```
// Preamble
TX_Sig.subvec(Ngaurd_F,Ngaurd_F+LP-1 ) =Preamble_RRC;
// Channel pilots
TX_Sig.subvec(Ngaurd_F+LP+Ngaurd_CH, Ngaurd_F+LP+Ngaurd_CH+LC-1)
=Channel_RRC;
//Data
TX_Sig.subvec( Ngaurd_F+LP+Ngaurd_CH+LC+Ngaurd_CH2,Ngaurd_F+LP+Ngaurd_CH
+LC+Ngaurd_CH2+LD-1 ) =bits_filtered;
```

The USRP buffer can accommodate only 2044 complex float data type samples. Accordingly, the USRP buffer is fed with those samples in a loop from the complete frame of 19819 data samples in the code snippet below.

```
for ( k=0; k<total_num_samps; k++ )
{

buff0[k+40] = std::complex<double> ( real(*(TX_sig_rot_mem+k))*0.8 ,0);
buff1[k+40] = std::complex<double> ( imag(*(TX_sig_rot_mem+k))*0.8 ,0);

}
```

Presently, the frame repetition rate of 20ms is used. Once the real time transmission is done the waveform can be observed on the oscilloscope to notice the frame repetition time between two preambles. In order to visibly demarcate between frames, the data is kept as zero.

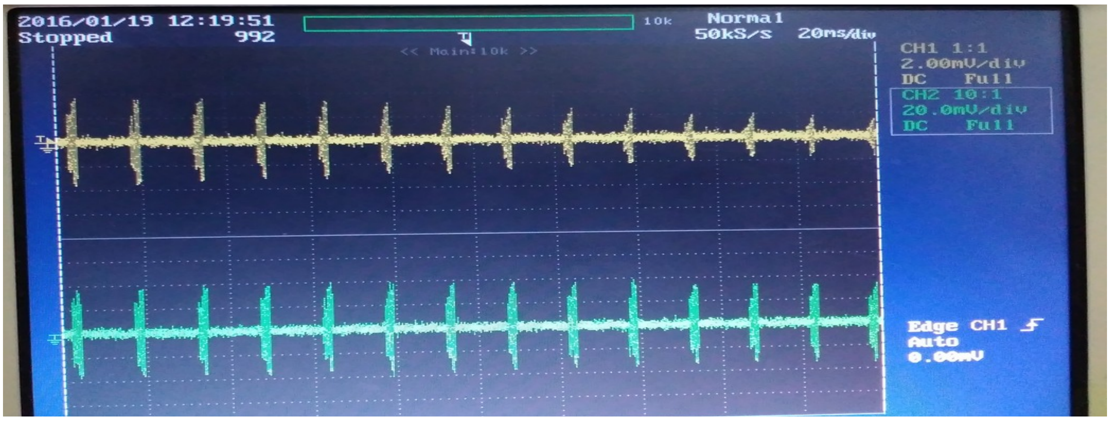


Fig 4.2 Real time transmission waveform

Need for C/C++ Libraries. The transmitter and receiver signal processing chain requires frequent generation of vectors and matrices. It is also requires to perform signal processing, arithmetic and logic operations on these vectors and matrices. The conventional C/C++ programming using arrays would not only become highly exhaustive but also make it difficult to debug the program for errors. In order to overcome these issues, the Armadillo library, which has MATLAB like functions for various operations, is used. At the receiver end, a combination of faster library i.e. Intel IPP and Armadillo is used to enhance the processing speed. The details are explained in the subsequent section.

4.2 Real Time Receiver

The receiver is also configured with the same USRP parameters as that of the transmitter. Frame synchronization is done using correlation with the known preamble sequence at the receiver. The preamble portion of the frame does not contain the rotating polarization wave. It is linearly polarized. It is depicted below. The complete frame except for the preamble undergoes rotating polarization and reaches the receiver. The channel estimation is done with the channel estimation bits using zero forcing technique. The data portion of the frame is then equalized using the estimated channel coefficients and then the frame error rate is calculated using CRC checksum.

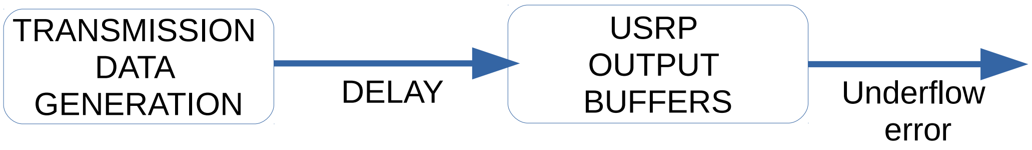
Preamble	Rest of the Frame

4.3 Challenges Faced And Remedies Adopted

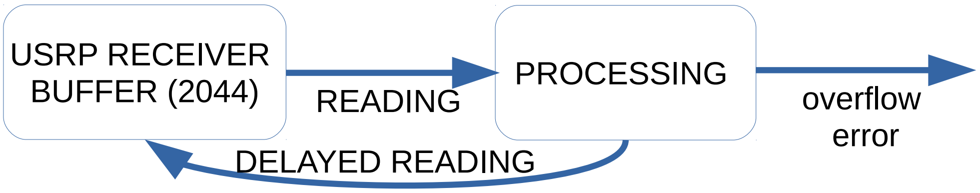
During the implementation, various problems are encountered in the USRP interface and C/C++ environment. A brief about these and problems and remedies thereof is discussed here.

(a) **<u>Processing Speed Errors.</u>** Any processing speed mismatch between the host PC and the USRP would result in overflow or underflow errors. These errors are illustrated in the following block diagram. Transmitter underflow errors arise mainly from the erratic configuration of transmission timing parameters. If the feed to USRP transmit buffer is delayed for more than acceptable limit, the USRP will return the underflow errors. Similarly, if there is an unacceptable delay between two consecutive reading calls to USRP buffer due to extended processing, the receiver USRP will return overflow errors. The transmitter underflow errors are eliminated using correct specification of USRP timing parameters.

At the transmitter:



At the receiver:



(b) **Library Migration (Armadillo To Intel IPP).** Intel provides Intel Integrated Performance Primitives (Intel IPP) C/C++ compiler libraries for signal processing and other applications. These libraries provide very fast processing speed on Intel architecture. Therefore, the Intel IPP libraries are used at the receiver for faster signal processing tasks. The entire receiver algorithm is changed from existing Armadillo library to Intel IPP library. However, the change of code has been an complicated task view the specific commands used by Intel IPP. A simple coding snippet shows the complexity involved in writing a correlation operation in Intel IPP program and Armadillo. In Armadillo, it is simply a one or two line task however, in Intel IPP it is a complicated approximately 10 line programme. Notwithstanding, the conversion from Armadillo to Intel IPP is completed and the receiver trials are successful with the new Intel IPP library. The difference between part of programme written in Armadillo and Intel IPP is brought out in the following code snippet.

Intel IPP Snippet

```
IppEnum funCfgNormNo = (IppEnum)(ippAlgAuto|ippsNormB);
int bufSizeNo=0;
Ipp8u *pBuffer;
status = ippsCrossCorrNormGetBufferSize(src1Len, src2Len,
    dstLen, lowLag, ipp32fc, funCfgNormNo, &bufSizeNo);
if ( status != ippStsNoErr ) return status;
pBuffer = ippsMalloc_8u( bufSizeNo);
status = ippsCrossCorrNorm_32fc(pSrc1, src1Len, pSrc2, src2Len,
         pDst,dstLen,lowLag,funCfgNormNo,pBuffer);
ippsFree( pBuffer );
```

Armadillo Code

```
out = conv(temp,filter);
```

Processing Speed Comparison. In order to illustrate the difference in processing speed using Armadillo and Intel IPP, a correlation operation is performed on two vectors of sizes 4088 x 1070 using Intel IPP and Armadillo libraries for 100 iterations. It turns out that Armadillo takes 26 micro seconds per sample whereas Intel IPP takes 72 nano seconds. It can be seen that per sample processing time using IPP is about 370 times faster than Armadillo.

4.4 Half Duplex Point To Point Link

The architecture used to achieve half duplex communication is shown below. Each transmitter and receiver uses the two antennae connected at the designated ports. The diagram pertains to RPW scheme. In case of Fixed polarisation only one crossed dipole will be connected to only one TX/RX port and while reception one of the ports of 90^0 will be terminated. The schematic is shown for two USRPs in the figure 4.3.

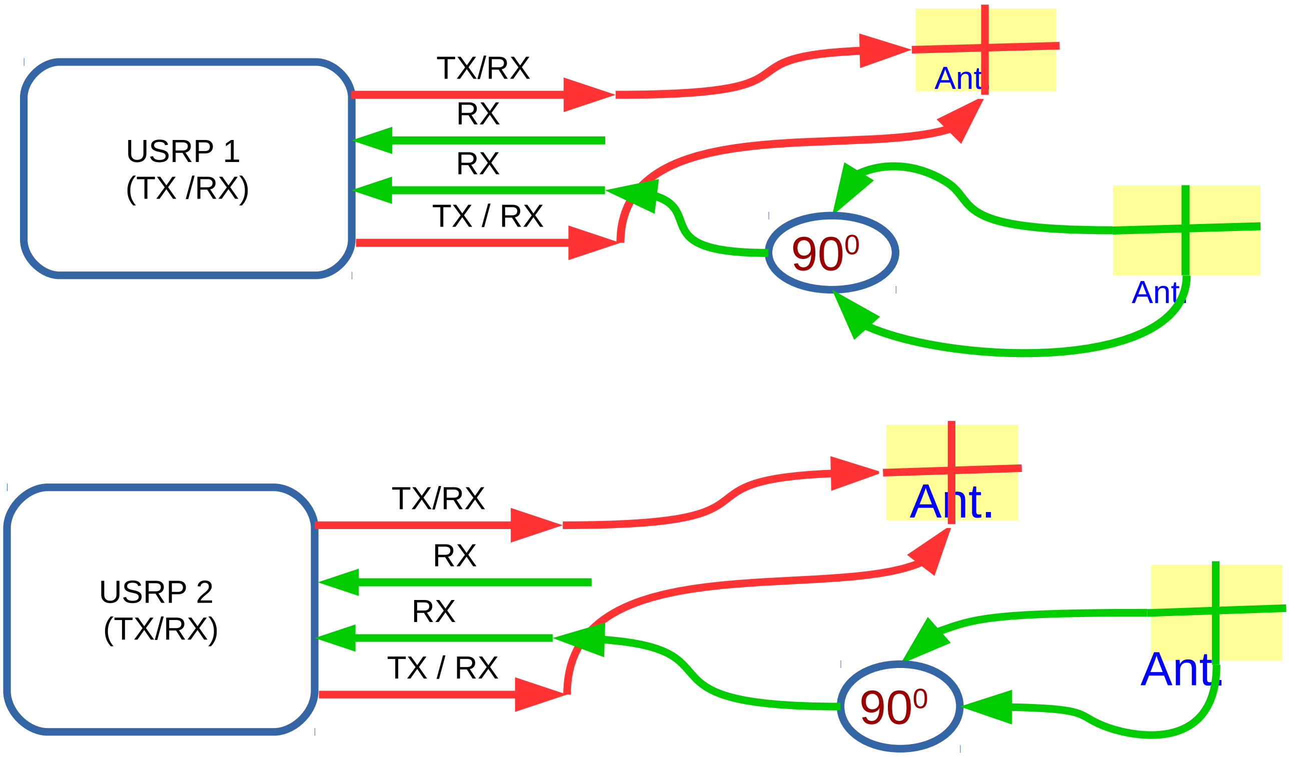


Fig 4.3 Schematic for Half Duplex Communication between Two USRPs

The USRP needs to be configured in TX as well as RX mode at once to work as a full fledged half duplex node, though either only transmission or reception would be performed at any given time. Accordingly, the independent transmitter and receiver programs are infused together to create a half duplex node. However, there have been significant changes that are made while configuring the various internal parameters of the USRP. An important observation about B210 USRP is that it could be enabled only in a 2x2 mode i.e. two transmitter ports and two receivers ports enabled. It does not allow the option of 2 Tx ports and 1 Rx port which is required for the set up. Therefore, only the Rx buffer of that RX port connected to receiving antenna is read by the program though two RX ports are enabled.

The Operation of the Half Duplex Link is understood with help of Fig.4.4. In which, it transmission and reception instances are demonstrated using the timing diagram. Any one fo the node in link is configured as master which transmits data frames at a fixed Frame Repetition Rate. In this example, the frame duration is 20ms and frame repetition rate is 100ms. The master node transmits a frame for 20ms and switches to receiver mode for balance 80ms i.e. until next instance of transmission. On the other hand, the slave node is predominantly in the reception mode. It receives a frame sent by master and sends an acknowledgement only if certain 'id field' is present in the received data. It is a case of conditional acknowledgement.

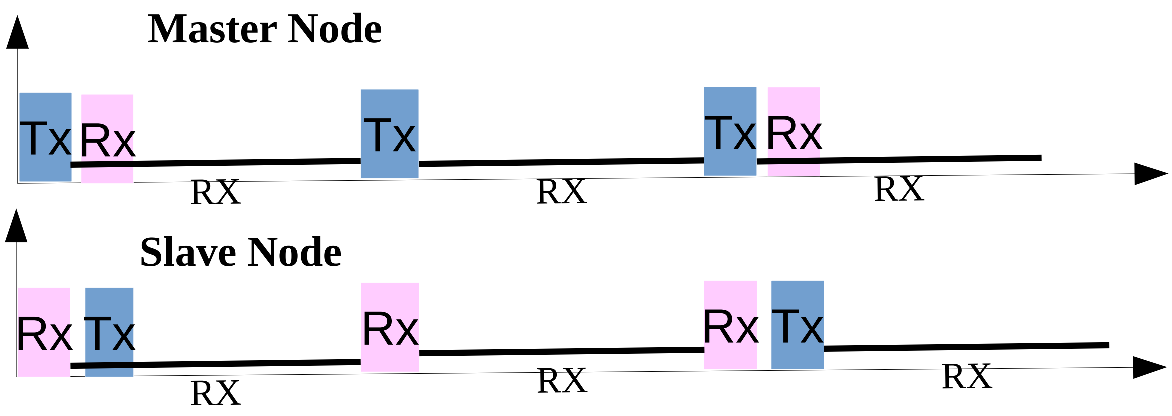


Fig. 4.4 Timing Diagram of Half Duplex Link

The TX denotes transmission instance and RX denotes a reception instance. It can be seen that slave node has received all three frames transmitted by the master. However, it has acknowledged first and third frame and not second as that frame may not have the certain id field as required by slave to send and acknowledgement. Whereas, the master is shown to receive every frame that is transmitted by slave as acknowledgement. This sequence is followed just for the demonstration purpose.

However, it is important to mention that this link is designed to preliminary test the working of single USRP in Transmission and Reception modes in same session. All the features of this half duplex link can be chosen and exploited to suit requirements of individual nodes of a star topology based network.