DESIGN AND IMPLEMENTATION OF MAC PROTOCOL FOR ROTATING POLARIZATION BASED WIRELESS MODEM FOR M2M WIRELESS COMMUNICATION SYSTEMS

A Project Report

Submitted by

Susheela Singh

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

MASTER OF TECHNOLOGY

Department Of Electrical Engineering

Indian Institute Of Technology, Madras

MAY 2017

THESIS CERTIFICATE

This is to certify that the project report titled ***Rotating Polarization Wave Based Wireless Modem for M2M wireless communication systems***, submitted by **susheela singh**, 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. Venkatesh Ramaiyan
Guide
Assistant Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai

Date: May 2017

ACKNOWLEDGEMENTS

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organization. I would like to extend my sincere thanks to all of them.

I am grateful to my guide, Dr. Devendra Jalihal, for having faith in my skills and providing me the platform to demonstrate the same and want to thank him for his continuous support. It would not have been possible without the kind of support and help he provided to me throughout my project.

I am highly indebted to Dr. Venkatesh ramiyan for his guidance and constant supervision as well as for providing necessary information regarding the project & for his support in devising and executing the MAC layer algorithms. He has been very kind and inquisitive at the same time in giving valuable inputs and also helped me in completing the project.

I am also fortunate to get in company of Praveen Kumar (Project associate) who has been a very interesting co-passenger in this eventful journey. His prompt inspirations, timely suggestions with kindness, enthusiasm and dynamism have enabled me to complete my project.

I would especially thank my project partner samriddhi majumdar for her continuous help and support and making my stay more memorable.

ABSTRACT

A wireless M2M (machine to machine) communication in the process industry plants is severely affected by slow fading nature of propagation channel and multipath 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. So the idea of introducing fast fading through polarization angle diversity in such environment is proposed by Dr. Takei of HITACHI. Rotating wave polarization (RPW) is one such diversity technique.

Basic structure of transmitter and receiver for generating RPW, physical layer algorithm at transmitter and receiver such as frame synchronization, frequency offset estimation & correction and channel estimation and equalization has already been developed and implemented by previous teams.

This thesis focuses on design and implementation of medium access control (MAC) layer protocol that makes it possible for several terminals or network nodes to communicate within a multiple access network that incorporates a shared medium. Two types of protocol has been implemented and tested in detail: one is star network (1: N) and other is peer to peer network (N: N). Experiments has been performed in the machines lab to mimic industrial environments and performance of these protocol has been evaluated in terms of frame error rate (FER) and retransmission rate which confirms reliability of channel sensing mechanism.

TABLE OF CONTENTS Page

LIST OF FIGURES

Page

CHAPER 1

INTRODUCTION

The signal transmission and reception via a wireless link is very unpredictable in nature, that the successful reception of the signal can only be speculated by means of probability measures. The situation is often worse when there is no Line of Sight (LOS) path between the transmitter and the receiver. A typical example of this is an industry/factory set up where the heavy metallic obstructions between any given transmitter and receiver are present and environment is of static nature. These two problems can together result in a deep fade scenario, i.e., the wireless channel between the transmitter and the receiver can make the signals that reach the receiver after multiple reflections and scattering to destructively interfere, thus making the received signal power to be very low. Moreover, the static nature of the environment ensures that the deep fade scenario can occur for a prolonged duration of time making the decoding of the signal almost impossible during this time interval. Furthermore, the communication link can easily be tampered with, thus rendering the path between the transmitter and receiver useless for communication. Security issues have been addressed at the PHY layer mainly using spread spectrum form of communication. The problem of deep fade scenario is also an area that has been seriously dealt with by academicians and researchers. To overcome deep fades, a variety of diversity techniques have been proposed in the literature. The rotating polarization technique addresses both the security and deep fade issues in one shot.

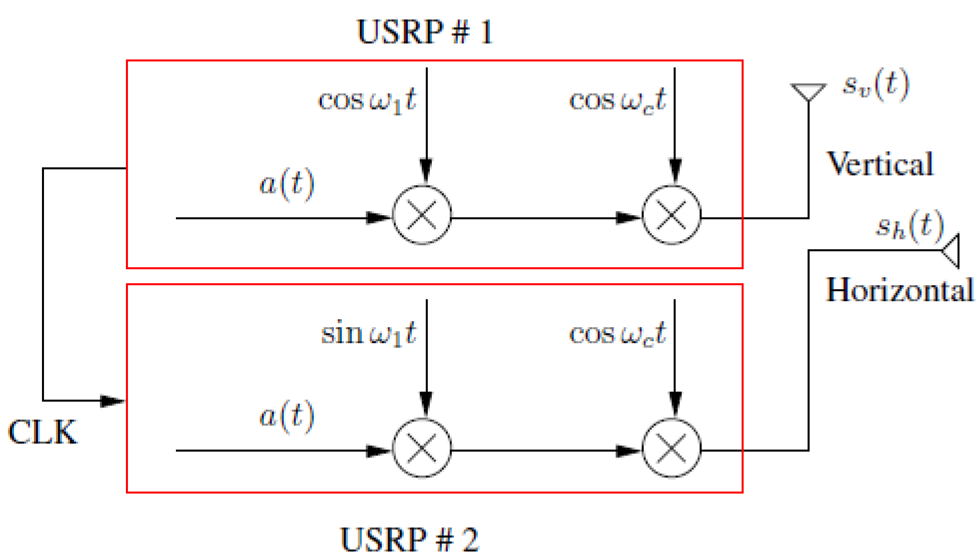
1.1 Objective

In an infrastructural environment, where there is no line of sight (NLOS) path between the transmitter and receiver, the signal is transmitted after multiple reflections and scattering due to heavy objects/machineries present in the environment. The different paths via which the signal gets reflected add up at the receiver and constitute the received signal. If the signals via multiple paths add destructively, the received signal will have a very low SNR and decoding the signal reliably might not be possible. A polarized wave after undergoing reflection changes its state of polarization. The transmission path can be regarded as a polarization slit that allows only certain kinds of polarizations to pass through. Thus, the angle of the incident wave’s polarization along with the scatterers present in the path is very important in deciding the polarization with which the signal reaches the receiver. Hence, by controlling the polarization angle of transmission, we can indirectly control the multipath fading that the signal undergoes. By adding diversity to the polarization angle of transmission, fading is artificially induced. Dr. Takei has proposed a technique known as polarization angle diversity by generating a rotating polarization wave and hence the name polarization angle diversity. This technique is very different from the traditional circular or elliptical polarization in the sense that a circularly polarized wave has a frequency of propagation equal to the frequency of rotation of the wave.

The frequency with which an RF wave propagates is generally in the order of MHz or GHz and so the rate at which the wave rotates is also of the same order. Resolving the multi paths undergone by each polarization angle at this rate is extremely difficult. In order to control the rate at which the wave rotates, we generate a wave called rotating polarization wave. The frequency of rotation of the wave is much smaller compared to the frequency of propagation of the wave. This is of great significance because, the multipath fading that the signal undergoes can easily be resolved at this smaller rate. This technique is very powerful in the sense that without the need for multiple antennas at the transmitter, we achieve diversity using the multiple polarization angles. Moreover, this technique is digital, that the number of polarization angles that we intend to resolve, in other words, the order of polarization angle diversity can be digitally controlled at the baseband by using an appropriate spreading code of required length and suitable frequency of rotation.

1.2 Generation and reception of Rotating Polarization Wave (RPW)

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 and other signal processing details are given in the previous reports. 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(*wc t*) before transmission. This is depicted in figure below.



Figue 1.1 Block Diagram of Transmitter

The signal at two antennae is given by

$$sv\,(t)=a(t)\cos(w1t)\cos(wc\,t)$$

$$sh(t)=a(t)\sin(w1t)\cos(wc\,t)$$

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 900. 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.

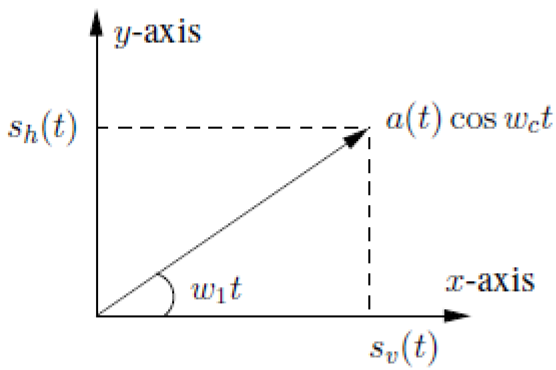


Figure 1.2 Projection of transmitted signal in the plane of 2 antennae

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

$$r(t)=h(t)a(t)\cos(w1\,t)\cos(wc\,t+\theta(t))$$

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 below.

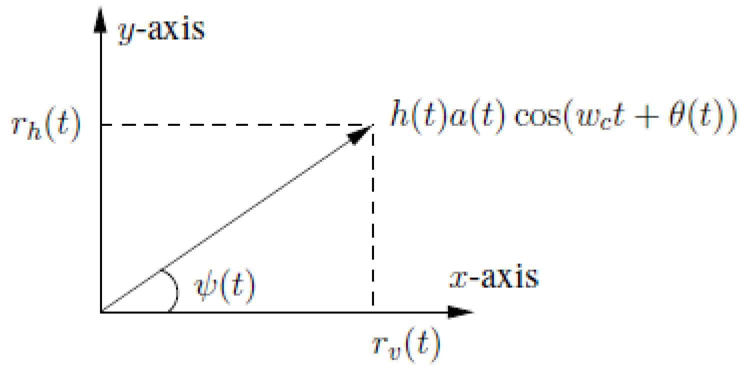


Figure 1.3 Received signal vector & its projection on 2 axes

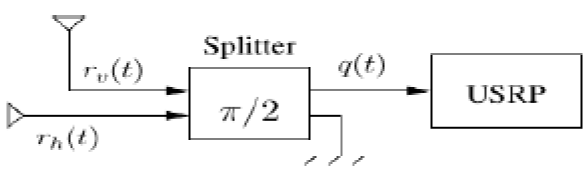


Fig 1.4 Receiver with Two antennae and Phase Splitter/ Combiner

Where the various signals are given by

$$rv(t)=h(t)a(t)\cos\psi(t)\cos(wc\,t+\theta(t))$$
$$rh(t)=h(t)a(t)\sin\psi(t)\cos(wc\,t+\theta(t))$$

and $\psi(t)=w1\ t+\varphi(t)$

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 multipath 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 synchronization, channel estimation and frequency Offset correction is given in chapter 2.

1.3 Project Background

The project is a joint work of HITACHI and IITM. The scope includes design and field trials of Rotating Polarization Wave Based Modem. The concept of polarization 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 GNU Radio and USRP (Universal Software Radio Peripheral) and Matlab for signal processing.
Physical layer algorithm for rotating polarization wave (RPW) has been designed and implemented by previous team. Also, Preliminary work for higher layers such as Medium access control layer(MAC) has been initiated which includes designing a protocol based communication system that follows certain topology. Setup for half duplex node has been built.

1.4 Organization of thesis

Chapter 2, focuses on physical layer algorithms and their implementation which has been done by the previous team of this project. It includes time synchronization to detect start of the frame, frequency synchronization to detect and correct frequency offset and channel estimation to know about channel conditions.

In Chapter 3, we present design and implementation of medium access layer (MAC) protocols for two types of network topologies. One is star topology based network and second is peer to peer network where there is no central coordinator.

In chapter 4, we evaluate the performance of two proposed MAC layer protocols in terms of frame error rate, throughput and retransmission rate and collision rate.

CHAPTER 2

PHYSICAL LAYER ALGORITHM AND IMPLEMENTATION

2.1 PHY Frame Format

PHY frame format for RPW is shown in figure2.1. Synchronization header (SHR) of PHY frame contains PN sequence as preamble bits in order to perform frame synchronization. It also contains channel estimation sequence which is used for frequency offset detection and correction as well as to estimate channel. Appropriate zeros are padded in between preamble, channel estimation sequence and data sequence. The size of zeros vector is variable and can be changed depending upon frequency of rotation of RPW. Length of PHY header (PHR) is 1 octet in which starting 7 bits are used to store frame length in bytes and last 1 bit is reserved. For Beacon frame, Data frame and ACK frame, frame lengths are 7, 14 and 6 bytes respectively. Payload for RPW is variable and can vary from 0 to 14 octets depending upon the application.

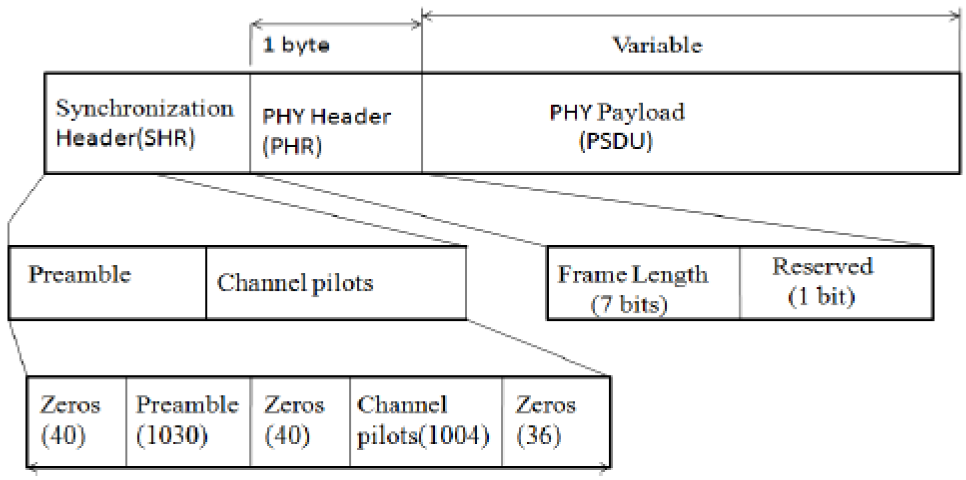


Figure 2.1 RPW PHY Frame structure

Now we discuss generation of various sequences which are used in PHY layer.
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.

Payload: 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 lengths. 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.

Before sending data to RF processing following sequence of operations need to be performed at the transmitter end which is shown in the figure 2.2.

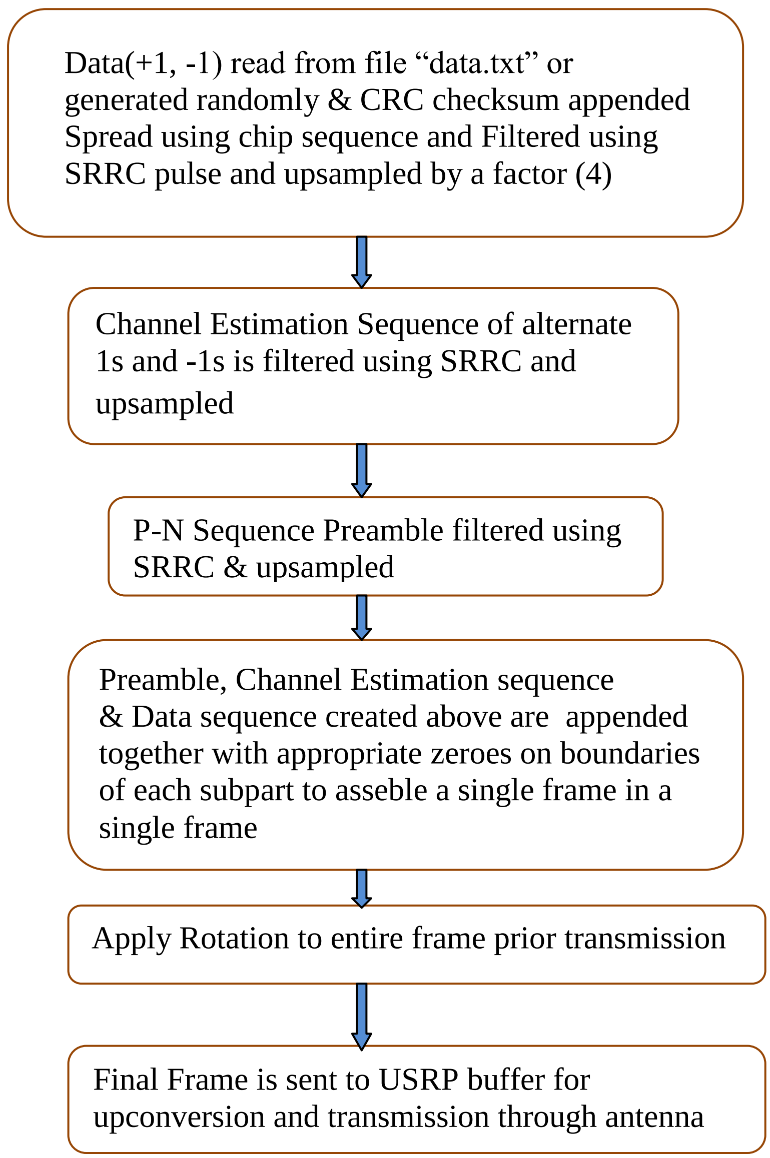


Figure 2.2 Transmitter operations sequence

Sequence of operations performed at receiver after receiving a frame is shown in the figure 2.3.

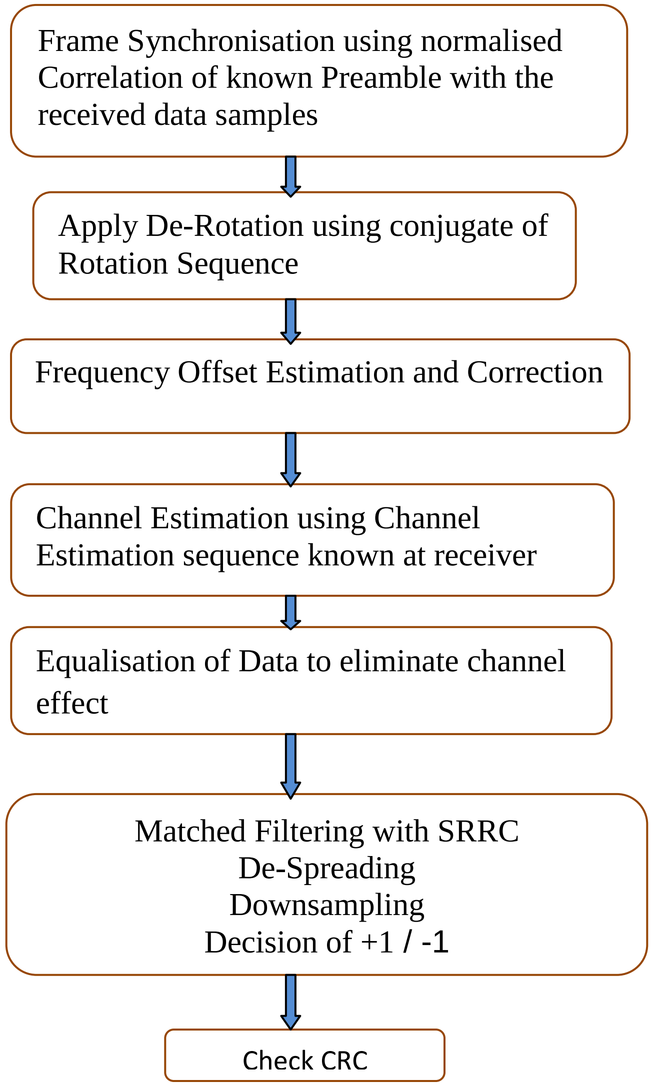


Figure 2.3 Receiver operations sequence

Now we discuss PHY algorithm performed at receiver in detail.

2.2 Frame Synchronization

It is very important to synchronize 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 synchronization algorithm. The steps that are followed in the frame synchronization 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 (of $\vec{X}$ length 1030) and their magnitude stored in $\vec{A}$=|(Σ$\vec{X}$ $\vec{Y}$)| 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 normalization 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 normalized, 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 normalize 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.

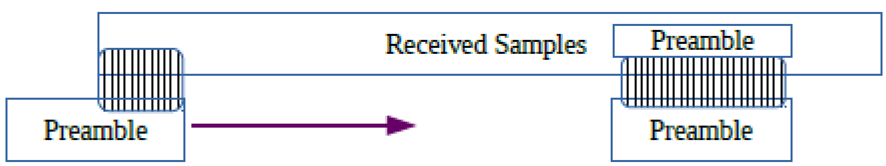


Fig. 2.4 Correlation Procedure

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

(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) Normalization is done as follows:

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

(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.

2.3 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).

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

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 $\quad x(n) = \exp\left(j2\pi f_c \frac{n}{f_s}\right)$

Is received as $\quad \hat{x}(n) = \exp\left(j2\pi\left(f_c + f_{off}\right)\frac{n}{f_s}\right)$

The angle of which should be multiplied by $\frac{-f_s}{2\pi.200}$ 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.

2.4 Channel estimation and equalization

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 $\bar{h}$ - channel coefficient
$\bar{x}$ - transmitted channel estimation sequence
$\bar{w}$- 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,

$\frac{{\sigma_w}^2}{E_x}$ where ${\sigma_w}^2$- noise variance
E_x- signal power

The data sequence is equalised by dividing it with the estimates $\hat{\hat{h}}$ obtained above. The rotating polarisation wave scheme has rotation frequency of 100KHz. It is important to note that that 100 KHz frequency of rotation is used in the advanced stages of the project i.e. Real Time implementation onwards. The rotation Frequency used is 3906.25Hz. The explanation of calculation of rotation frequency is given in the previous reports. For the rotation frequency of 100 Khz 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 equalized by these channel coefficients. In next iteration next 10 data samples are equalized and so on.

2.5 Match filtering, Despreading and Down sampling

Once we get equalized data after frame synchronization, frequency offset correction and channel equalization, we will pass data through SRRC filter to reverse the effect of pulse shaping performed at transmitter to remove the effect of inetersymbol interference caused by the channel. This SRRC filter at receiver is called as match filter which is optimal filter that maximizes signal to noise ratio (snr) in presence of additive noise. This is equivalent to convolving the received signal with a conjugated time-reversed version of the known SRRC filter coefficient.
Data obtained at the output of match filter is downsampled with a factor of 4 since it was upsampled with the same factor at the transmitter. Downsampled data is then multiplied with same 64 bit chip sequence (PN sequence) used at the transmitter and all 64 chips are summed up to get one data bit**.** This process is called as dispreading.

2.6 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

MEDIUM ACCESS CONTROL (MAC) LAYER PROTOCOL AND IMPLEMENTATION

This section focuses on design and implementation of Protocol for MAC layer for RPW in the infrastructure based environment. We have designed protocol for two types of networks. One is centralized network (1:N) where only one device is master and rest of the devices are client. It is a star topology based network. Communication can only be initiated by master. Second network is Peer to peer (N:N) network where N no. of devices are connected to each other without any central coordinator and any device can initiate the communication. There is no device which is master. The devices communicate in a half duplex mode where we permit unicast, multicast and broadcast communication between the master and the clients and can support reliable communication with acknowledgements.

There are three types of frames which are specified in HITACHI document and which are implemented-

- Beacon frame
- Data frame
- ACK frame

Beacon frame: The role of a beacon transmission within the scope of this thesis is the following:

a. Synchronization of other RPW devices
b. Master node identifications
c. Describe structure of the super frame.

Data frame: Role of this frame is contain the sequences of octets that the next higher layer has requested the MAC sub layer to transmit that is it contains actual information to be transmitted.

ACK frame: This frame is sent in response to the data frame. If a device sends data frame and it needs acknowledgement back to confirm whether data frame has been received correctly then this frame is sent. This is sent to increase integrity (reliability) of the transmission.

3.1 General MAC Frame Format

The following figure 3.1 shows the general frame structure of a packet as specified in the document provided by the HITACHI. Numbers written in each box represents no of bits used to represent that particular field .Each field of frame is described below. The value of each field changes depending on the different types of frame and also depends on the requirement.

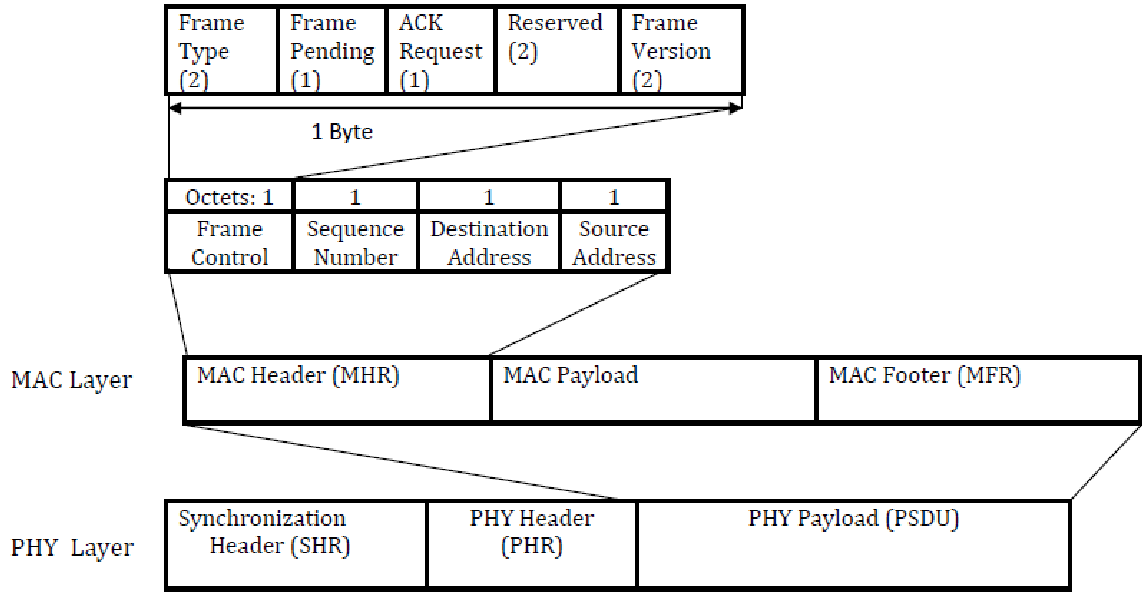


Figure 3.1 General MAC Frame Format

All frame formats in this sub-clause are depicted in the order in which they are transmitted by the PHY, from left to right , where the leftmost bits are transmitted first in time. Unless otherwise specified in this clause, all reserved bits shall be set to zero upon transmission and may be ignored on receipt .

A. <u>MAC Header (MHR)-</u>

I. <u>Frame control</u>

a. Frame type

Frame type value (B1 B0)	Description
0 0	Beacon
0 1	Data
1 0	Acknowledgement
1 1	Mac command

b. Frame pending- It shall be set to 1 if the device sending the frame has more data for the recipient. This field shall be set 0 otherwise.

c. **ACK Request-** It specifies whether an acknowledgement is required from the recipient device on receipt of data. Its value is extracted from file/stream/application. If this field is set to 1, the recipient device shall send an ack or data frame. If this field is set to zero, the recipient device shall not send any response.

d. **Reserved bits-** 2 bits in the MAC Header were specified as reserved bits in the document provided by Hitachi. In our model, we have used the first bit to specify whether response needed from the client is Data or ACK frame. The value of this bit is read from the file/stream/application.

    0 - ACK response
    1 - Data response

    The second bit is unused and currently set to zero by default.

e. **Frame Version:** It specifies the version corresponding to the frame. Currently we are implementing version 1.0 thus this field is set to 00.

II. <u>Sequence number</u>

This field specifies the sequence identifier for the frame. It is an 8 bit field with sequence number ranging from 0 to 255.

III. **<u>Destination Address</u>:** It specifies the 1 byte address of the intended recipient.

- In case of beacon frame we use a value to 0xff to represent broadcast.
- In case of data frame sent by master, this address is of the intended client.
- In case of ACK or Data response from client, this field consists of Master address (0xfe).

IV. **<u>Source Address</u>**: Specifies the address of the originator of the frame. The default Master address is 0xfe.

B. <u>MAC Payload</u>

The Mac payload field contains the information specific to individual frame. The length of this field is variable (0 to 8 bytes in the current specification). Each of the frame types would define individual formats for such payload information. Also, to maintain at PHY layer for the transmission and reception of PHY frames, the total maximum PSDU size (the PHY shall be able to receive) is kept constant.

Mac Payload in Beacon Frame:

• The MAC payload in case of beacon frame specifies information regarding the super frame structure as shown in the figure given below.

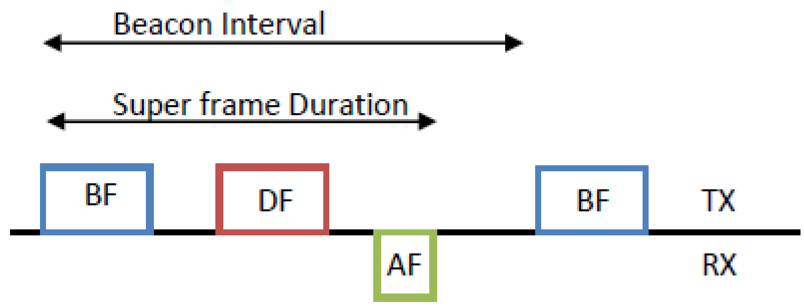


Figure 3.2 Super Frame structure

Where BF= Beacon frame, DF= Data frame, AF=ACK frame

The format of the super frame is defined by the master node.

• The super frame is bound by the network beacons sent by the master.

• MAC payload in case of beacon frame is of 1 byte with bits 0 to 3 specifying Super frame order and bits 4 to 7 specifying Beacon Order.

• *Super frame Order*: Provides information regarding the duration for which the super frame is active.

• *Beacon Order*: Provides information specifying the interval between consecutive beacons.

• As described in the figure 3.1, Super frame duration and beacon interval are derived with the help of Base Duration ,Super-frame order and Beacon order using the formula:

Super frame duration= base duration $* \, 2^{super\ frame\ order}$
Beacon interval= base duration $* \, 2^{beacon\ order}$

MAC Payload in Data Frame:
• In case of data frame we have MAC payload of 8 bytes which consist of binary data read from a file/stream/application.

C. MAC FOOTER (MFR)

MFR field consists of a 2 byte long CRC. It is calculated over MAC header and MAC payload parts of the PHY frame. The following 16 degree generator polynomial is used:

G16 (x) = x16 + x12 + x5 +1

The CRC shall be calculated for transmission using the following algorithm:

- Let $M(x) = b_0x^{k-1} + b_1x^{k-2} + \cdots + b_{k-2}x + b_{k-1}$ be the polynomial representing the sequence of bits for which the checksum is to be computed.
- Multiply M(x) by x^{16}, giving the polynomial $x^{16} * M(x)$.

- Divide $x^{16} * M(x)$ modulo 2 byt the generator polynomial, $G_{16}(x)$, to obtain the remainder polynomial, $R(x) = r_0 x^{15} + r_1 x^{14} + \ldots + r_{14} x^1 + r_{15}$
- The FCS field is given by the coefficients of the remainder polynomial, R(x).Binary polynomials are represented as bit strings, in highest polynomial degree first order.

Now we will show the frame structure of three different types of frames as mentioned earlier and we will see that how they differ in their respective fields.

1. Beacon Frame Format

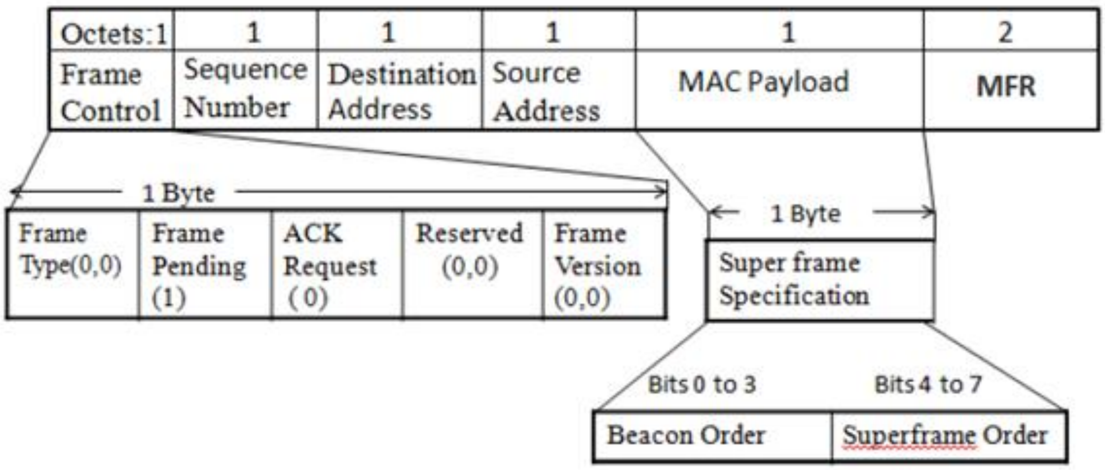


Figure 3.3 Beacon Frame Structure

2. Data frame Format

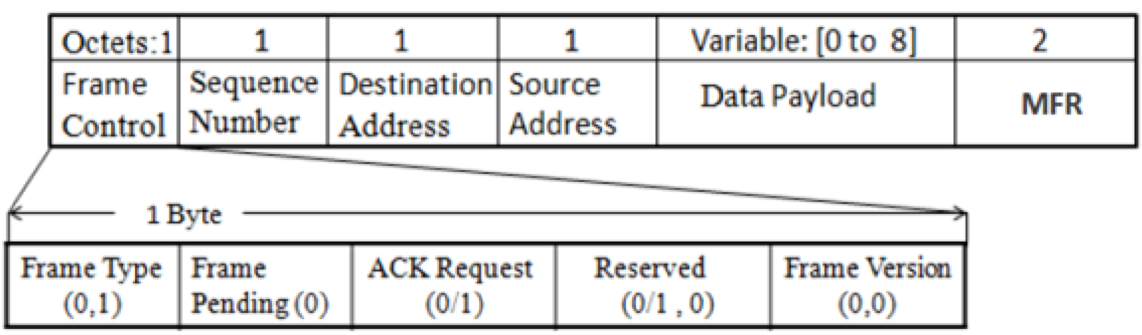


Figure 3.4 Data Frame Structure

3. ACK Frame Format

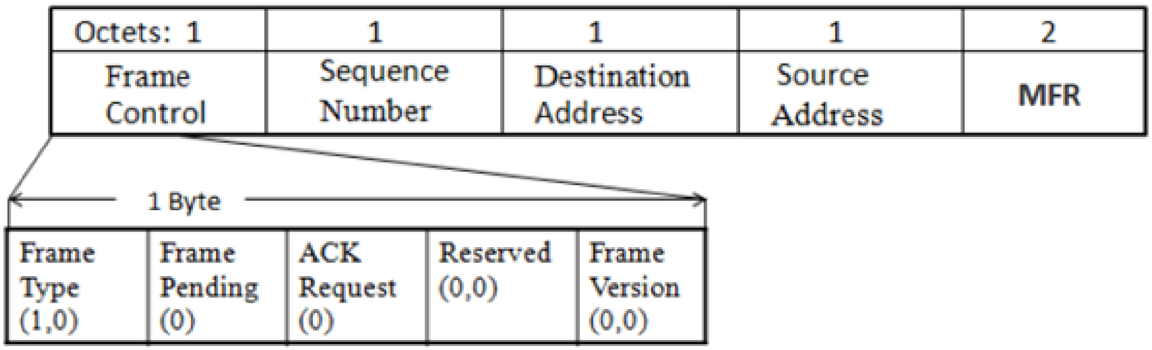
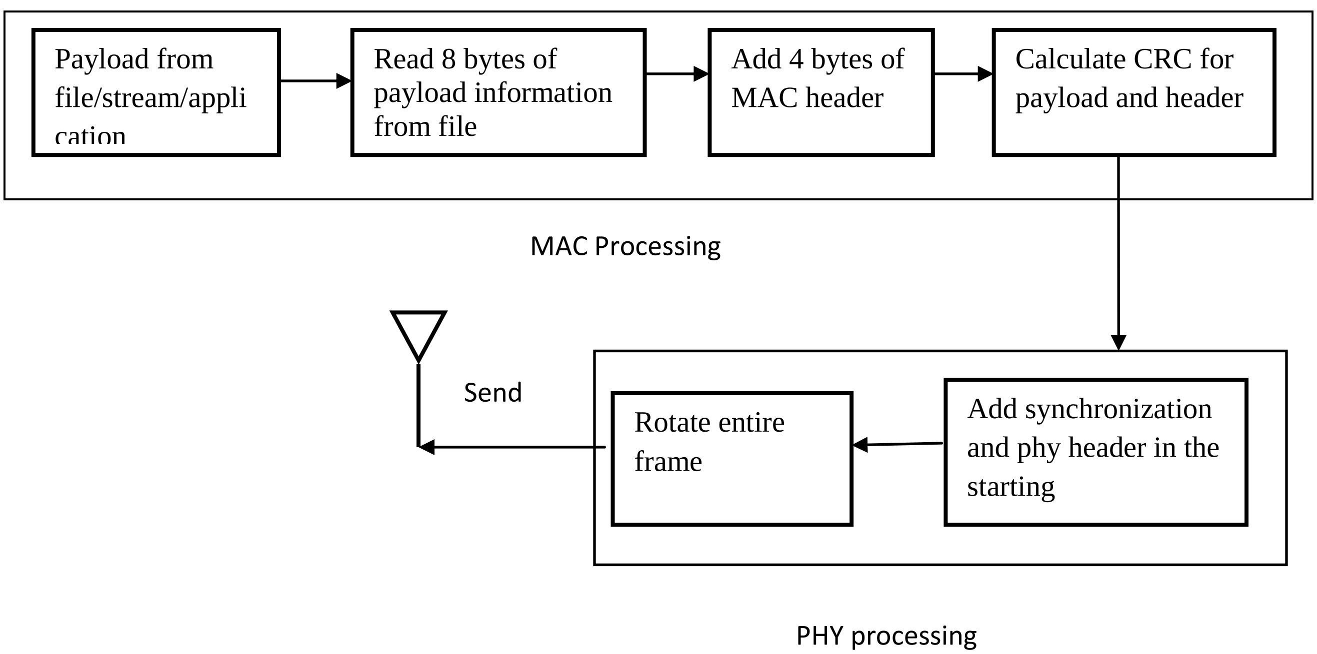
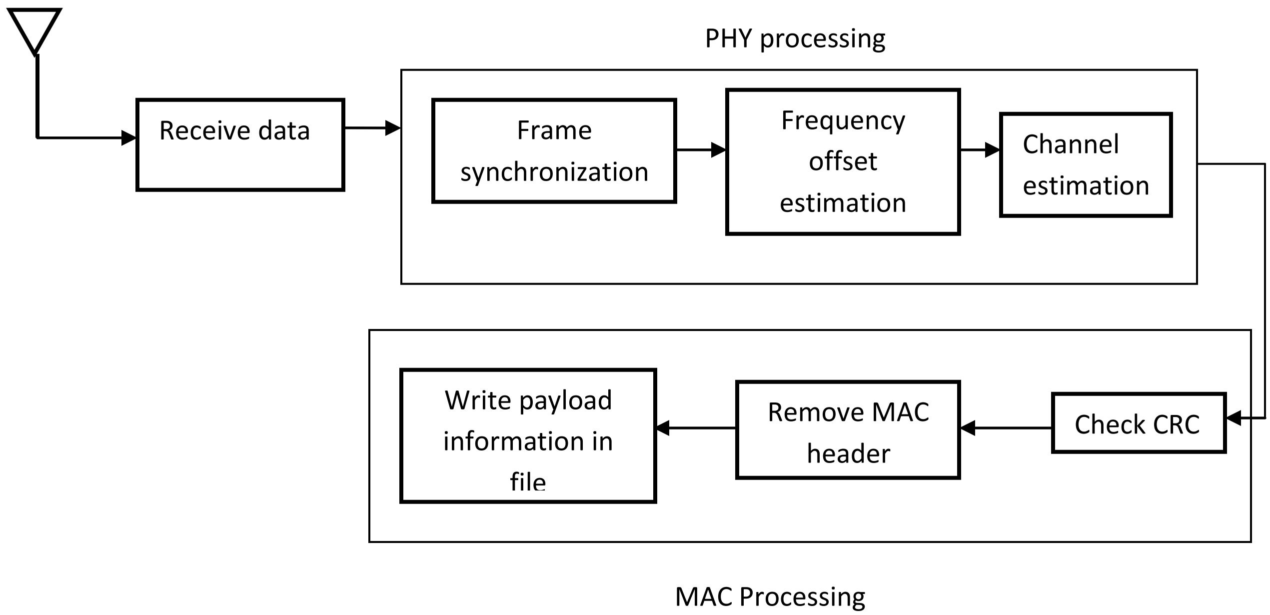


Figure 3.5 ACK Frame Structure

MAC operation sequence at transmitter:



MAC operation sequence at receiver:



3.2 Types of network implemented

Two types of network has been implemented as per defined in the HITACHI's document.

3.2.1 Centralized network (1:N)

We consider star topology based network where every device is connected to the central device. Centralized node is generally referred to as a master which coordinates with other N no. of devices which act as clients. All communication takes place between master and client. Data is stored on the centralized node i.e. at the master node. Master provides services to the client. The Master can decide the schedule and method of each transmission. Master node is also responsible for deciding who can access the channel, and the duration for which that node has control over the channel. This communication can either be Simplex or Half-Duplex or Broadcast in nature.

3.2.1.1 Network architecture

Network architecture of star topology is shown in figure 3.6. In figure every node (computer workstation or any other peripheral) is connected to central node called hub or switch. The switch is the server and the peripherals are the clients. All traffic that traverses the network passes through the central hub. The star topology is considered the easiest topology to design and implement. An advantage of the star topology is the simplicity of adding additional nodes. All peripheral nodes may thus communicate with all others by transmitting to, and receiving from, the central node only. Star topology based centralized network is more scalable and stable.

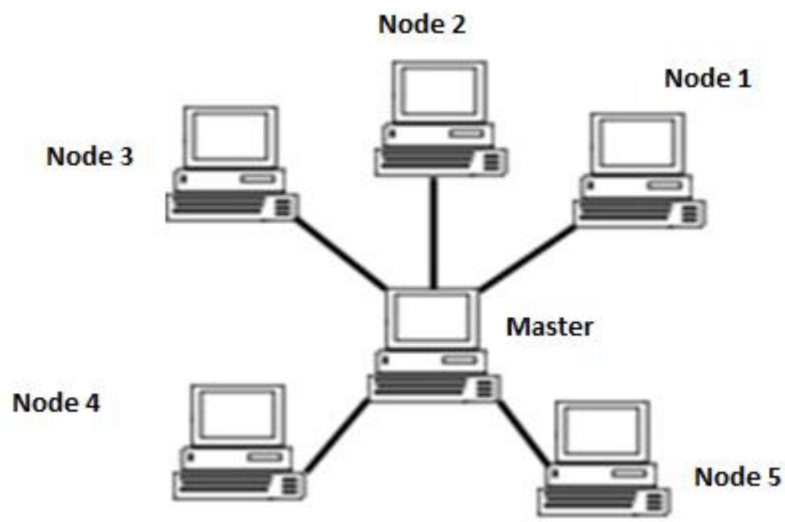


Figure 3.6 Star Topology(1:N) Network architecture

3.2.1.2 Overview of 1: N protocol

The 1 to N protocol uses a star topology where all communication takes place between master and clients as shown in figure 3.6. For each individual device, half duplex communication is employed by the master device. Master will start communication with clients if packets are available in the channel. The key components of MAC protocol are:
a. Master sends Beacon frame (BF) periodically with a fixed inter beacon duration (tib) as shown in fig. 3.7 .
b. Data communication between master and client happens in a polled manner when initiated by the master.
c. Master can send unicast, multicast or broadcast data frames (DF) to clients.
d. Clients on receiving the data packet may respond with an ack (AF) or data packet (RDF).
e. The entire process consisting of the master sending a data packet and receiving response from the client is completed within the inter-beacon duration.

Here we have illustrated the different scenarios possible in the communication between the master & the clients.

1. **No data exchange:**

BF represents beacon frame transmitted by the master at regular intervals. Beacon is sent for synchronization of other RPW clients. Clients receive the beacon packet and no response is generated. After receiving the beacon frame all the stations change their local clocks to this time. This helps with synchronization. Beacon also describes superframe structure. Currently payload of beacon frame is one byte long in which starting 4 bits (0 to 3) specifies beacon order and next 4 bits (4 to 7) specifies superframe order. Following figure shows the defined scenario where t_{ib} is interbeacon time.

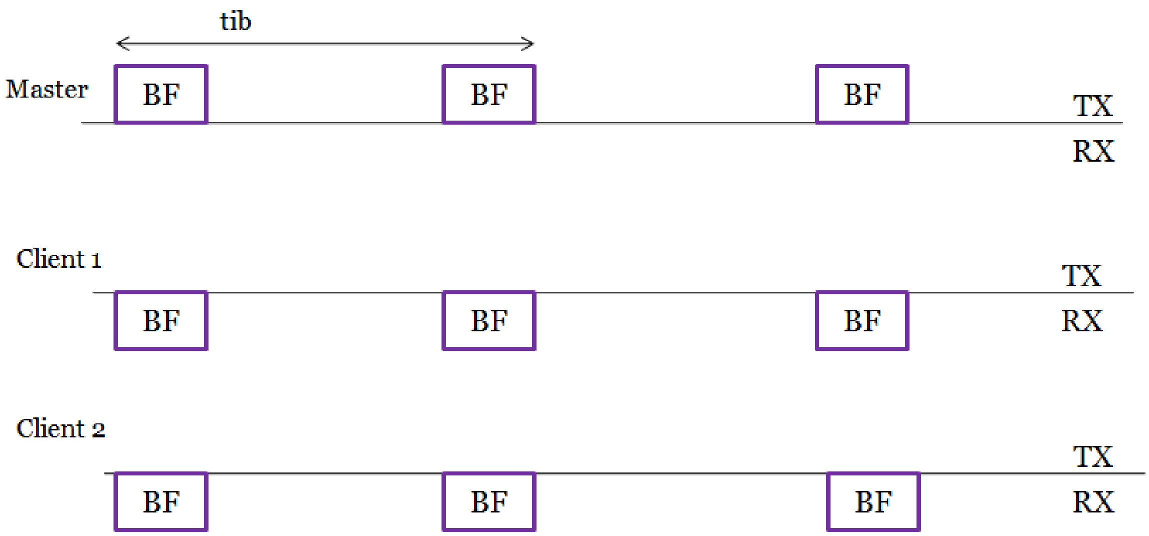


Figure 3.7 No Data Exchange

2. Data exchange with a single client.

This case describes when only one client is present in the network. When master sends a data packet to a client, it will respond to the packet. It may respond with a data packet or an ack packet depending on the ack field present the in the packet. Following figure explains this scenario where when first packet arrives. Client 1 responds back by sending an ack packet and when second packet arrives it responds with data frame.

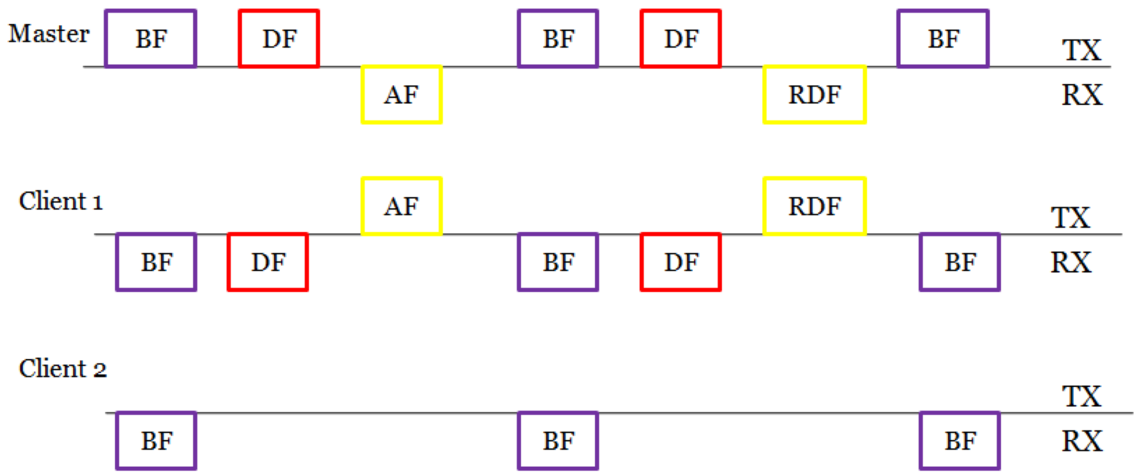


Figure 3.8 Data Exchange with a single client

2. **Data exchange with two clients.**

This case illustrates a scenario where master communicates with clients in a round robin fashion. When master sends a data packet, it will be received by all the clients. Clients will first check destination address present in the packet. If source address of any client matches with destination address of the packet then packet will be intended to that particular client and it will respond to the packet. It may respond with a data packet or an ack packet depending on the ack field present the in the packet. In the following figure 3.9 First packet is intended to client 1 which responded to master by sending an ack. In this duration client 2 will not do anything and will be keep listening to the channel. Second packet is intended to the second client. This time client 2 will respond back by sending an ack or data packet. In the following fig it has responded by sending ack packet. Since it is not intended to client 1, so it will be keep listening to the channel.

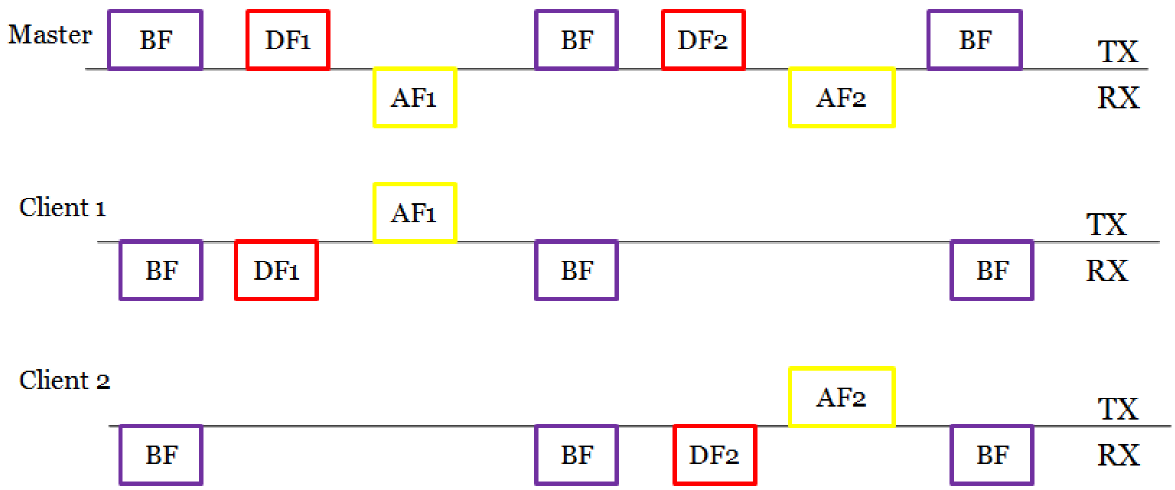


Figure 3.9 Data Exchange with two clients

3.2.1.3 Implementation of the protocol

The following flow charts in figure 3.10 and figure 3.11 describes the outline of the MAC protocol implemented in master and client node. Protocol is implemented with C++ programming language and armadillo & Ipp software libraries.

Implementation of 1: N at Master

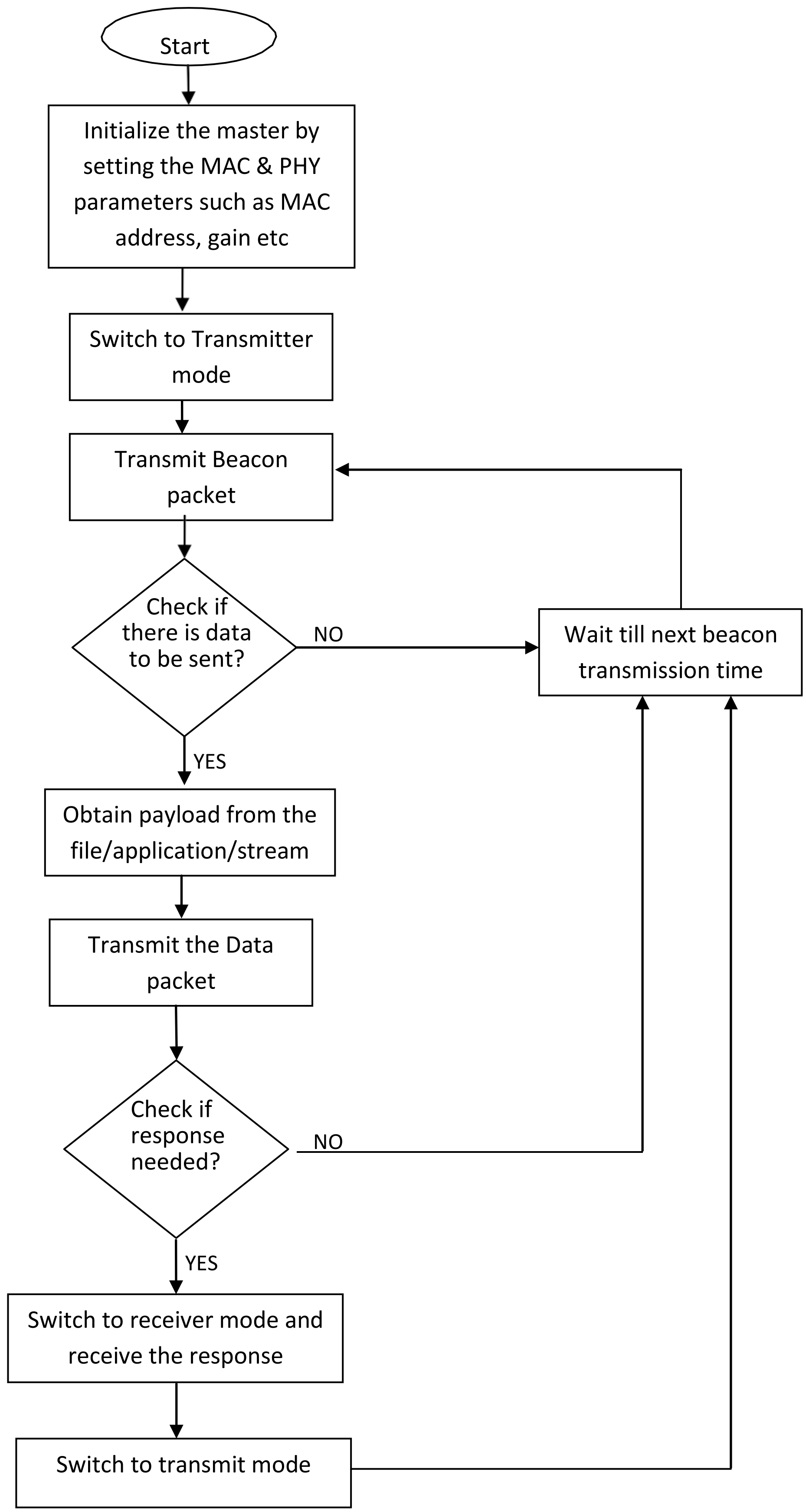


Figure 3.10

Implementation of 1:N at client

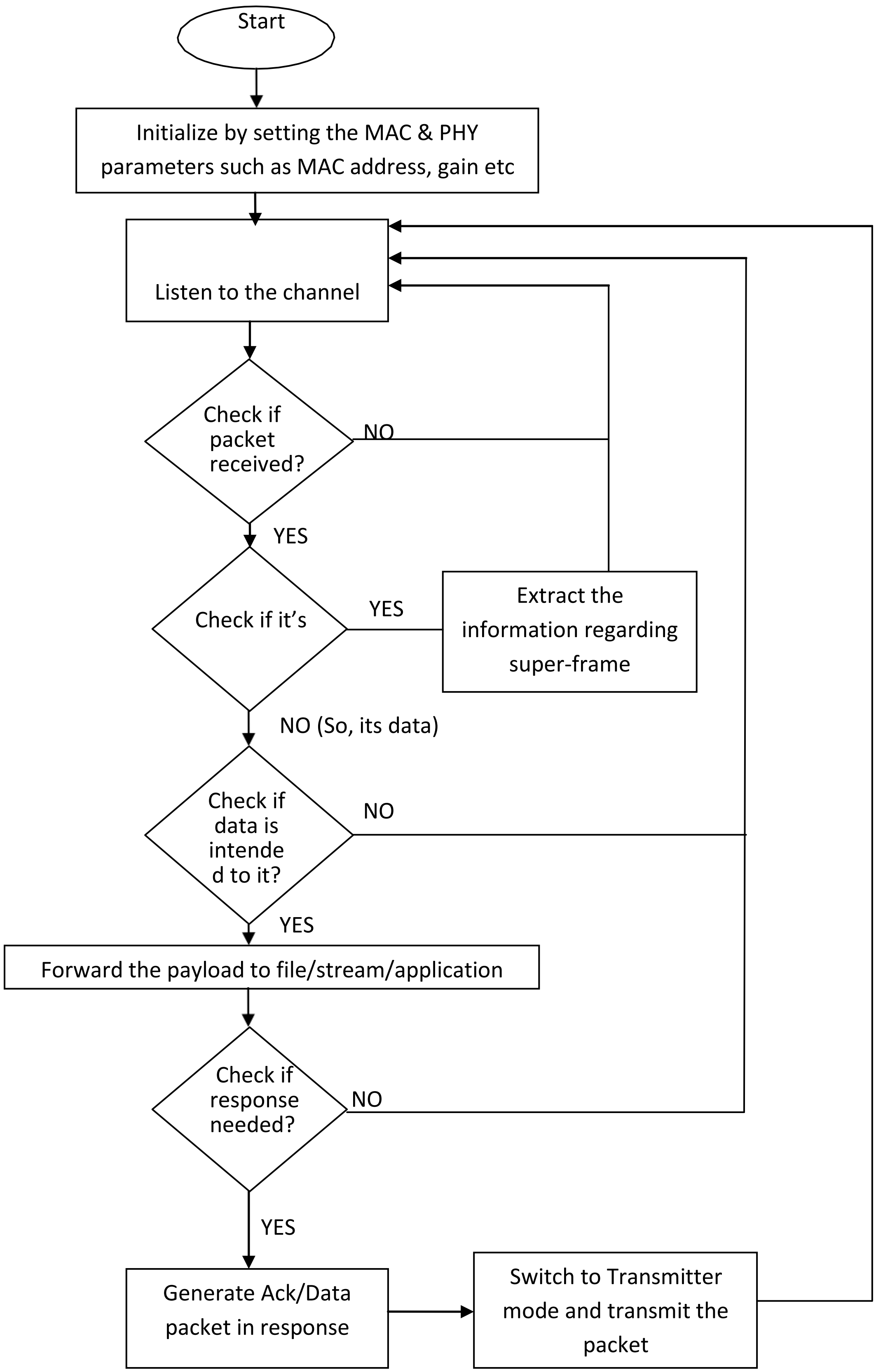
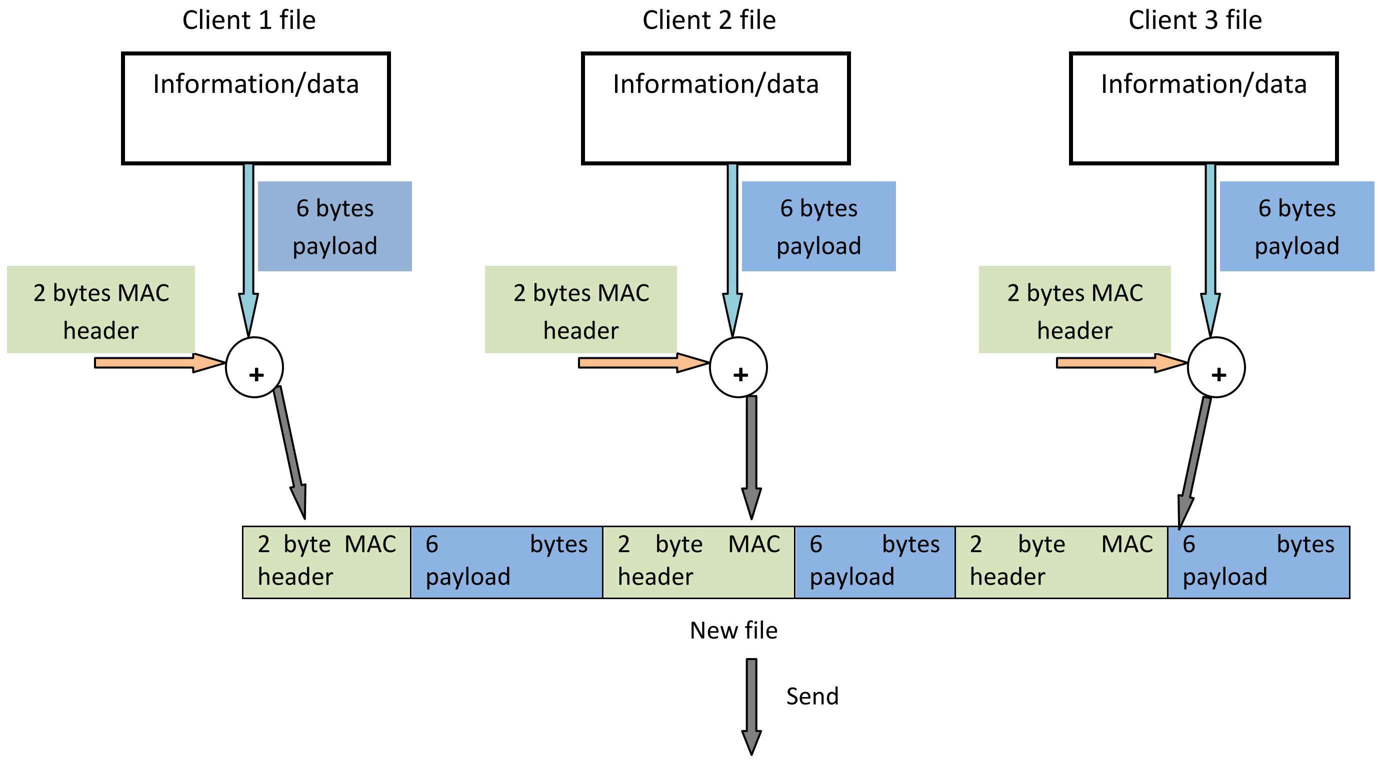


Figure 3.11



Process of sending and receiving information at master node, between master and a single client is explained in flow chart as shown in fig 3.10. In order to send data to N no. of clients, master first writes all information which it wants to send to each client in a single new separate file. It will read one packet from file of each individual client. Mac payload of each packet is of 6 bytes long. Master will read 6 bytes of information from the file which it needs to send to client 1 and it will append 2 bytes header in the starting of data payload. First two bits of header contain information about acknowledgement that is whether acknowledgement is needed and the type of acknowledgement (ack or data) needed in response. Next 6 bits are used to store sequence no. of the packet. Second byte of header contains destination address of the particular client. Next 6 bytes contains information to intended client, so length of each packet becomes 8 bytes. Similarly master will read one packet from each client's file and write it to the new file. When master has to send data to N clients it will read a packet from new file, send it to first client and wait until acknowledgement comes. Now it will send first packet of second client which is also stored in new file. Similarly it will send data to all clients present in the network. This is how master node has been implemented in 1 to N network.

Since each client receives and responds only to single node which is master, figure 3.11 explains process of receiving and transmitting data in detail.

Timing Diagram:

Timing diagram describes packet durations of all type of frames and switching time between transmit mode to receive mode for one cycle of transmission at master as well as at the client node which is shown in the figure 3.12.

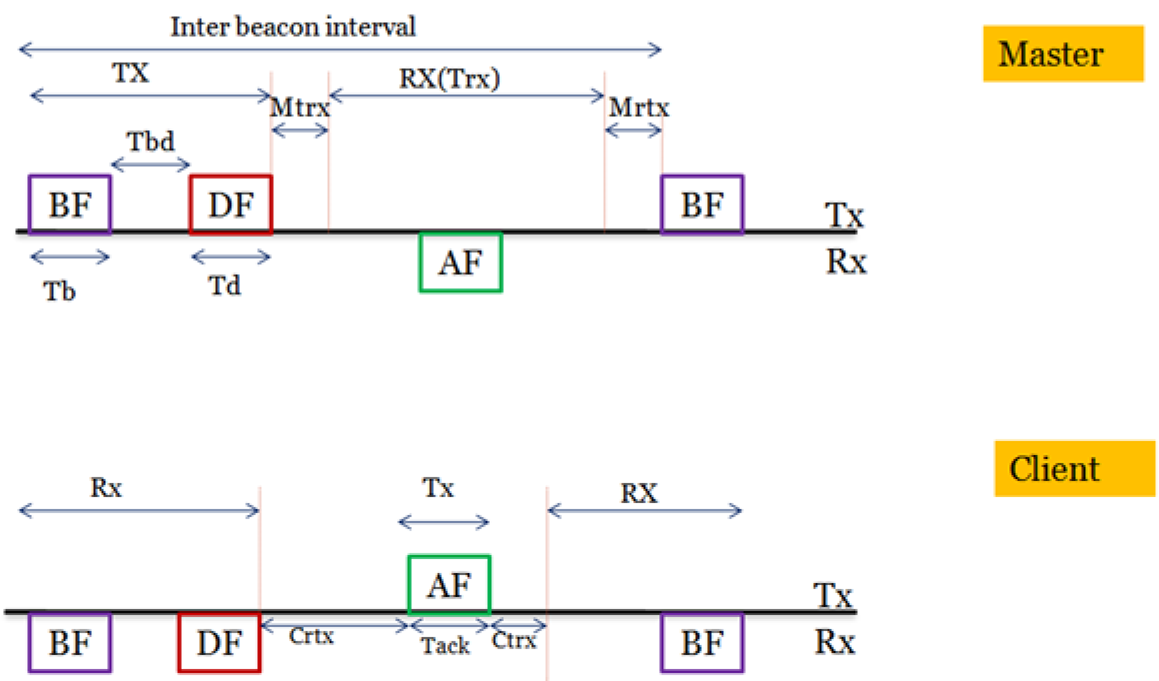


Figure 3.12 Implemented Timing Diagram

Tb**:** Beacon frame duration
Td: Data frame duration
Tack: ACK frame duration
Tbd : Time duration between Beacon and Data
Mtrx: Switching time of the master from transmitter to receiver
Ctrx : Switching time of the client from receiver to transmitter
Trx: Time interval during which Master is in receive mode
Mrtx: Switching time of the master from receiver to transmitter
Beacon Interval : Tb+Tbd+Td+Mtrx+Trx+Mrtx

Timing of packet is calculated based on the size of the frame. For example one Beacon frame is of 7 bytes including mac header, mac payload and mac footer which is equivalent to 56 bits. After spreading with the spreading factor of 64,it is upsampled with a factor of 4 and then passed through a rrc filter for pulse shaping which will give the length of PHY payload. Synchronisation header of PHY frame contains preamble and channel pilots with suitable no. of zeros as a guard bits. PHY header of PHY frame contains value of length of frame in bytes. For beacon frame, length is 7 bytes which can be represented by 8 bit value, which will give 2053 bits after spreading,upsampling and after pulse shaping. Total length of frame is then calculated by assembling synchronization header,PHY header and PHY payload. We can calculate total length of sequence (LS) using following 2 equations:

LD= (Nbits*spread-1)*osf+(osf*SPAN+1)
LS= Nguard_1+LP+Nguard_2+LC+Nguard_3+fr_len+LD

Where LD is length of mac payload
Nbits = 56
Spread = 64
Osf(oversampling factor) = 4
SPAN(of rrc filter) = 2
LP is length of preamble = 1030
LC is length of channel pilot = 1004
fr_len =2053
Nguard _1,Nguard_2 & Nguard_3 are guard bits.
Nguard_1 = 40
Nguard_2 = 40
Nguard_3 = 36

Sampling frequency at the receiver is 1 MHz which gives a sample of duration 1 µs. Multiplying total no. of the bits with sampling duration will give the beacon frame duration. Similarly duration of other frames can also be found.

Following figure 3.13 explains implemented timing diagram for different spreading factors with corresponding time durations that have been employed in the code.

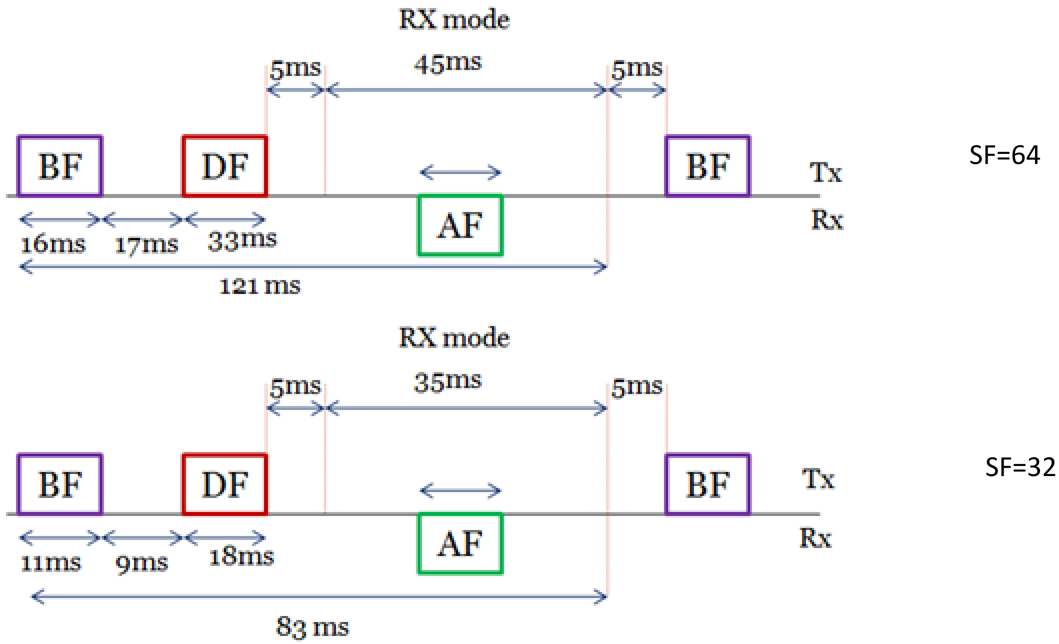


Figure 3.13 Timing Diagram for different SF

Phy data rate for spreading factor of 64 comes out to be 6.2 Kbps and 3.4 Kbps for spreading factor of 32. In chapter 4, we will see that performance of network for both the spreading factor is almost similar. So we can choose any of the spreading factor depending upon the application. For e.g. if we need higher data rate for application then we will choose spreading factor of 32 and vice versa.

<u>3.2.2 Peer to Peer network (N:N)</u>

In this network, there is no central coordinator that is there is no master device unlike 1 to N network, which will control communication between other devices connected to the network. There is N no. of devices which can be connected to the N-1 no. of devices. Any device/node can start communication and can send data to any other device/node. Instead of passing data through a central hub, we can pass data back and forth directly from one "peer" to another "peer". In effect, every connected device is at once a server and a client. There's no special node which will tell to whom to send and from whom to receive. Each node decides on its own when is the best time for it to transmit such that it is the only one which transmits to avoid collision.

3.2.2.1 Network architecture

Typical architecture of N to N network is shown in figure 3.14. We consider a collocated infrastructure-less wireless network comprising of N wireless devices/nodes. There are four nodes in the network where each node is connected to three other nodes. Nodes can exchange information directly without third party. This type of network does not have any servers.

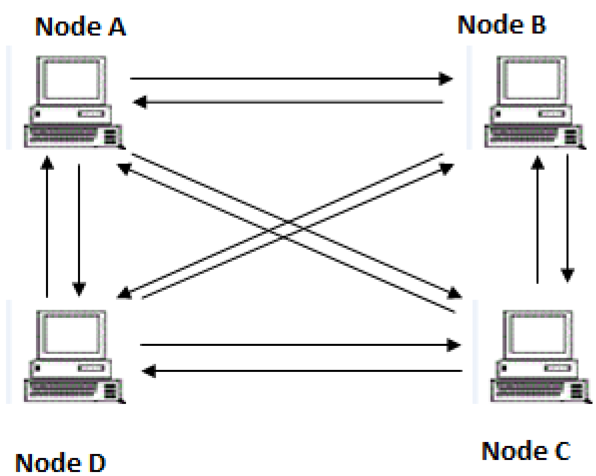


Figure 3.14 N to N Network Architecture

3.2.2.2 Overview of Protocol

Computers and routers are interconnected in a wired or wireless manner by a shared medium. Thus, all the computers and the router need to compete to be able to transmit a data packet between two computers or between a computer and the router. (On the other hand, two routers connected by a dedicated medium, say, a fiber link, do not compete to transmit data, because the medium is dedicated to the pair of routers.) A shared medium is also called a broadcast channel, multi-access channel, or random access channel.

Problem with a shared channel: Assuming that computers are independent entities, how to determine who gets to use the medium (channel) next when they compete to access the channel. (If there is a dedicated channel between two computers, the computers can transmit data at their will.) The goal of designing a MAC protocol is to resolve the above issue.

We consider a collocated infrastructure-less wireless network comprising of N wireless devices that communicate among themselves without a coordinator. The lack of a coordinator leads to simultaneous transmissions over the broadcast channel resulting in possible collisions.
We propose the following random access protocol to minimize collisions and to provide reliable communication in the infrastructure-less setup.

- All the nodes listen to the channel by default.

- If a node that has a packet to transmit will postpone the transmission by choosing a random backoff time. The random backoff time is measured in multiples of discrete time units called as slots, where a slot is of duration T_s seconds. Backoff time is the maximum waiting time a node should wait before sending a data packet in the channel. We will assume that the random backoff time (in slots) is distributed uniformly between 1 and T_{max} (where T_{max} is a design parameter).

- At the end of backoff time, node will sense the channel. Two scenarios can happen.
    i. **Channel is busy**: If channel is busy then it would select another backoff time and again wait for the channel to become free.

    ii. **Channel is idle:** If channel is idle then transmitter will send the data packet present in the queue immediately and wait for the response from receiver until backoff time expires. Response can be either data or ACK packet depending upon the application. There can be two cases:

    a. If it receives response within backoff duration then it will select another backoff time following same distribution and shall proceed to transmit the next packet in the queue.

    b. If it does not receive response back before a timeout, then it will retransmit the data packet. Maximum no. of retransmission that can happen is K_{max} which is a design parameter. If maximum no. of retransmissions reaches to k_{max} and still it did not get the response then it will drop the packet.

- Client While receiving data packet at the receiver end, can experience following two cases:
    i. It receives data packet correctly. In this case, it will send back an ACK or data packet depending upon application. After sending response if backoff time has expired then it will select a random backoff again. If backoff has not expired then it will continue to listen to the channel.
    ii. It does not receive data packet. This may happen because of bad channel condition or collision. In such case, it will be keep listening to the channel.

3.2.2.3 Design parameters of the MAC protocol:

1. **Slot duration (T_s):** The backoff duration is measured in discrete time units called as slots. We assume that a slot is of duration T_s seconds. The choice of T_s must depend on the diameter of the wireless network. T_s is also the minimum duration for which a channel must be sensed idle before a node initiates a packet transmission. Here, we note that acknowledgement and other data responses must be transmitted (by a receiver) with a time gap smaller than T_s. In practical scenarios, we expect T_s to be much smaller than the packet transmission times.

2. **Maximum backoff duration (T_{max}):** The random backoff (in slots) is distributed uniformly between 1 and T_{max} for all the attempts. The choice of T_{max} should depend on the number of nodes in the network and packet transmission durations. In future, we may

consider an exponential backoff that permits an adaptive transmission rate based on the channel quality and congestion in the network.

3. **Maximum permitted retransmissions (K_{max})**: A packet shall be retransmitted (in the event of a collision or channel error) K_{max} times before it is dropped. The choice of Kmax must depend on the application and channel quality.

Now we illustrate the different scenarios possible in the communication between different nodes. We describe the scenarios with a simple network comprising of three nodes.

1. **<u>Contention free transmission</u>**

   If there is no contention in the channel for e.g. node 1 is transmitting to node 3 and there is no other node in the channel which transmits at the same time. There would not be any collision in this case. This scenario is shown in the following figure and can be explained in following steps:

   i. Node 1 transmits a data packet after a random backoff time. There is no contention in the channel and the packet transmission is successful.

   ii. Node 3 receives the packet (transmitted by Node 1) and responds with an acknowledgement packet. The acknowledgement packet is received without error as well.

   iii. Node 2 does not have any data to send and will be keep listening to the channel.

   iv. In the figure 3.15, Td represents the packet transmission duration, TACK represents the transmission duration of the acknowledgement packet and Ts represents the slot duration. We note that the DATA-ACK intertransmission time is less than the slot duration Ts seconds.

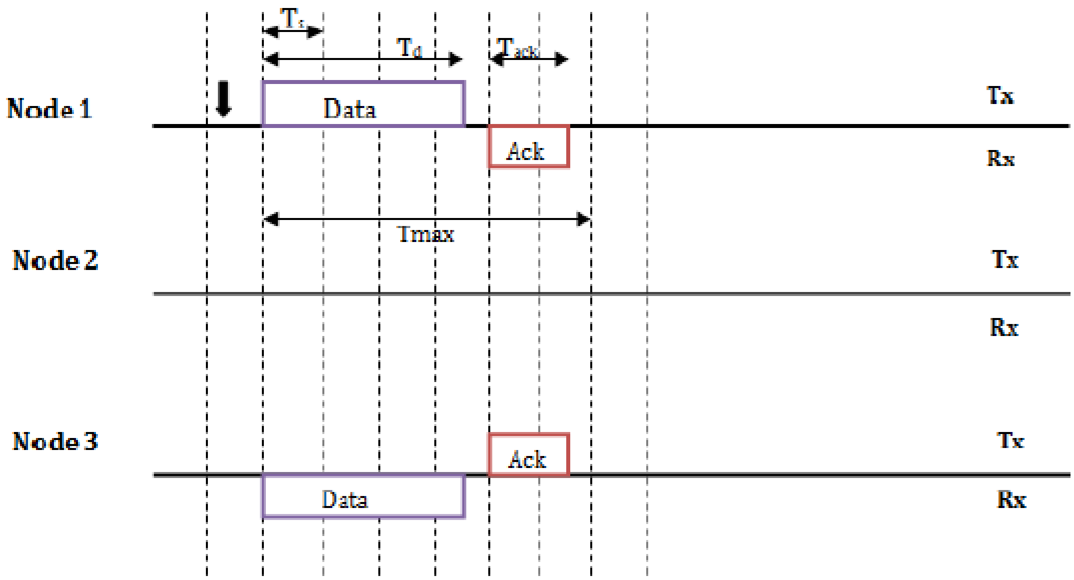


Figure 3.15 Contention Free Transmission

2. <u>**Transmissions involved with a Collision**</u>

    i. Random backoff time for Nodes 1 and 2 expires at the same time. Both node sense the channel and find its idle, so both nodes transmit a packet in the same slot.

    ii. The nodes fail to hear a response before the ACK timeout. Since collision has happened. Hence, the nodes select another random backoff time to wait before the retransmission.

    iii. Node 1 completes waiting time (random backoff time) first and transmits a data packet. The packet communication with Node 3 is successful.

    iv. Node 2 random waiting time is more than node 1.When its backoff time expires it will find channel busy (since node 1 transmission is going on) and will continue to listen to the channel.

    v. This scenario is shown in the figure 3.16

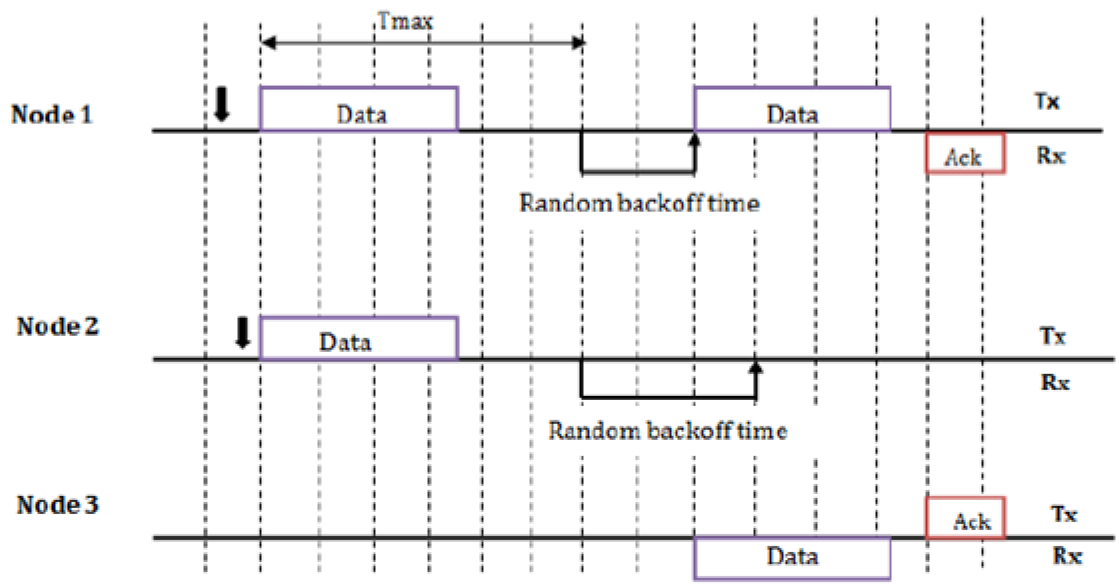


Figure 3.16 Transmission involved with collision

---

3. <u>Transmission without collision</u>

i. Figure 3.17 explains this scenario.

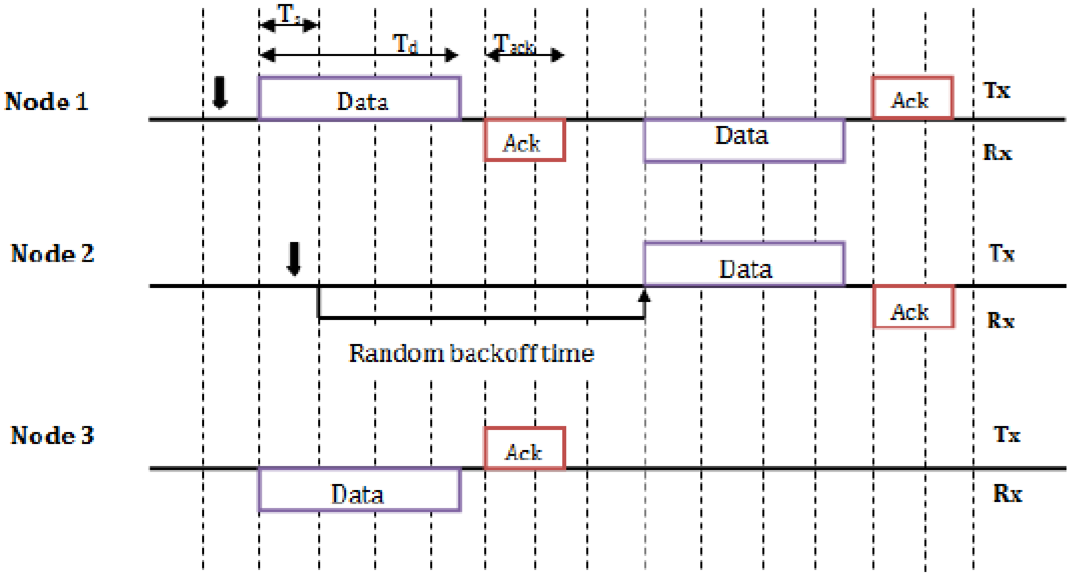
ii. Node 1 transmits a data packet (to node 3) after a random backoff and after sensing the channel to be idle for more than Ts seconds.

iii. Node 2 senses the channel to be busy at the end of its random backoff time. Hence, Node 2 selects another random backoff time to wait before the transmission.

iv. Node 3 receives the packet (transmitted by Node 1) and responds with an acknowledgement packet. The acknowledgement packet is received without error as well.

v. Node 2 senses the channel to be idle at the end of its random backoff. Now, Node 2 initiates a transmission to its intended receiver Node 1. Node 1 receives acknowledgement packet and transmission is successful.

vi. So transmission is avoided because of choosing random backoff time.



Figue 3.17 Tranmision without Collision

3.2.2.4 Implementation

The following flow chart describes the outline of the MAC protocol for N to N network implemented in each node.

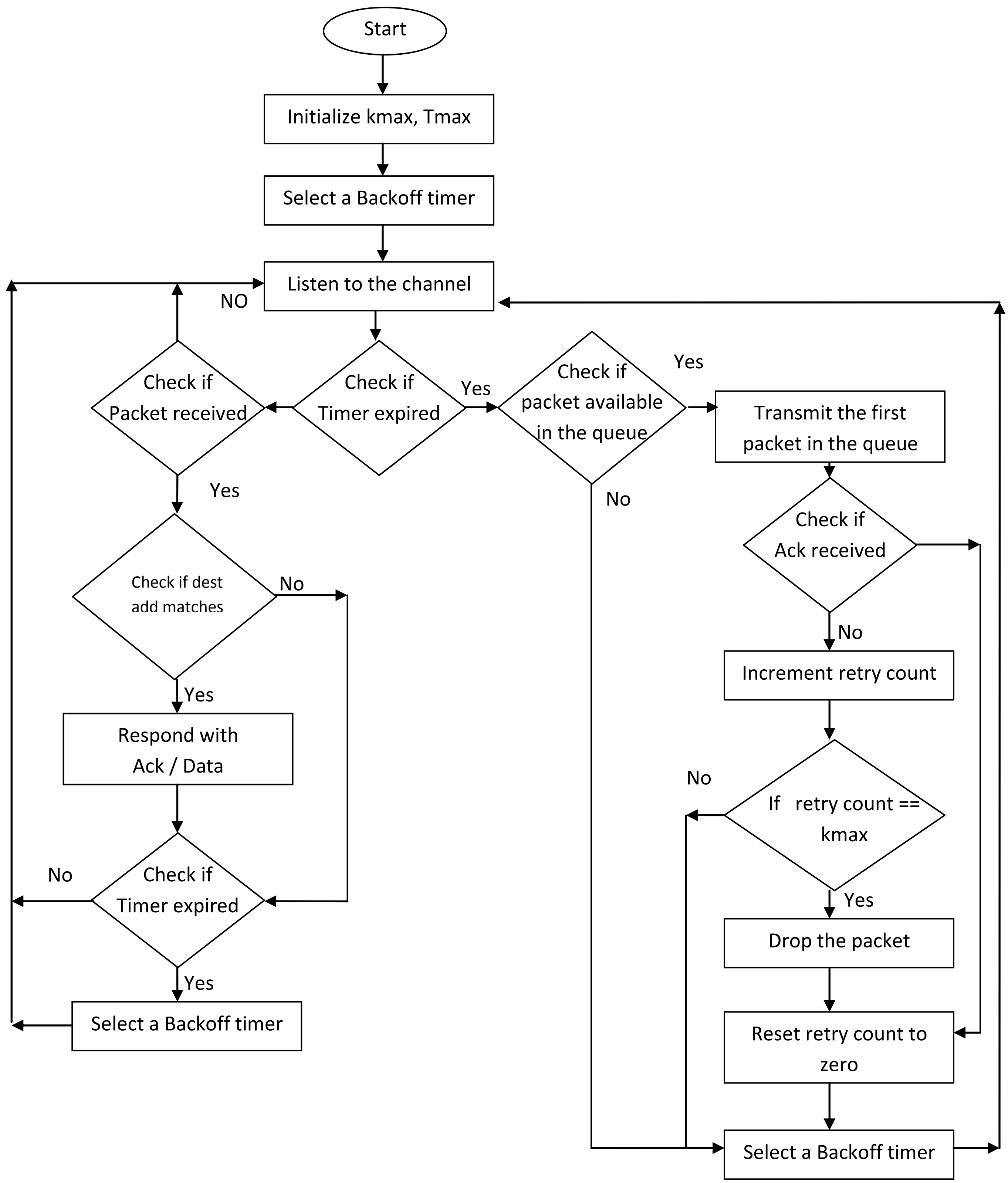


Figure 3.18

CHAPTER 4

PERFORMANCE EVALUATION OF RPW MAC

4.1 Performance evaluation of star network

4.1.1Experimental Setup for 1: N

The Machines Lab in the Department of Electrical Engineering, IIT Madras is chosen as the venue to conduct the experiments. In order to mimic an industrial scenario, we require heavy metallic machineries with no line of sight path between the transmitter and receiver. The Machines lab under consideration, shown in fig.4.1, consists of heavy metallic machineries. This indoor environment under consideration is very static as all the equipments are fixed and there is no human or environmental interaction. Hence, the chances of prolonged deep fades are more prominent here. Thus, we have an environment perfectly similar to a factory set up. The layout of machines lab showing the transmitter and receiver locations is shown in fig. 4.2. The location represented by TX shows the position of the transmitter. Receivers are deployed at positions 1 through 6.

(a)

(b)

Figure 4.1 Pictures of machines lab

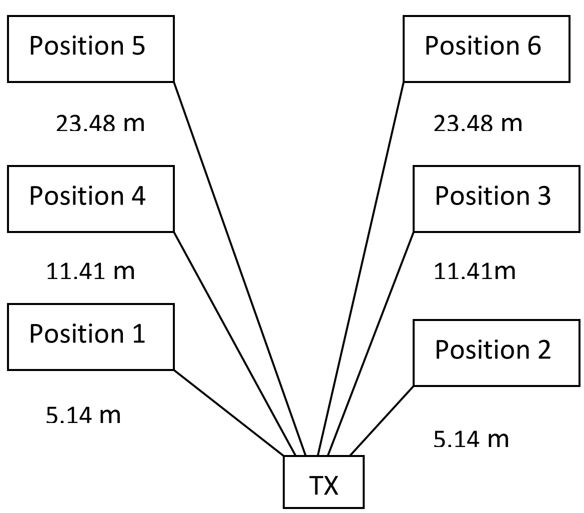


Figure 4.2 Layout of experimental set up

4.1.2 Performance analysis for 1:N

Since we have used polled communication between master and clients where only master can initiate communication to the intended client, there is no chance of collision in the network. Because master can communicate with only one node at a time. However, frames may go wrong because of errors in hardware used for example antennae, phase shifters etc, UHD error and channel errors. The goal of this experiment is to check MAC performance of star topology based network (1:N) and to evaluate frame error rate for uplink and downlink transmission for RPW devices. The experiment is performed with N equals to 2 that is only one master node and two client nodes are present in the network. Experiment is performed with different gains and for different distances.
We define a successful communication between the master and the client as follows:

1. Master sending a Data frame to the intended client.

2. The respective client upon receiving it, decodes the necessary bits and in response transmitting a “Data frame” or an “Acknowledgement Frame”.

3. Master receiving the Data frame or ACK frame

Around 7688 packets transmitted in each test by master using spreading factors of 32 and 64. MAC payload was kept as 8 bytes. We recorded the number of times successful communication took place between the master and the receiver. FER can be calculated as follows:

$$\text{Frame error rate(FER)} = \frac{\text{Total data packets sent by master} - \text{no. of successful trips}}{\text{Total no. of data packets sent by master}}$$

Average FER with varying Tx and Rx gain for Downlink and Uplink at fixed distance for spreading factor of 32 and 64 is shown in the figure 4.3 and 4.4 respectively. We can see from graph that as Tx & Rx gain increases FER reduces as expected. It is also clear from the graph that for higher spreading factor FER is lesser as compare to lower spreading factor since as spreading factor increases channel estimation will be more reliable and chances of more frames in errors will reduce. However, performance is almost similar for both spreading factors. Moreover, lower spreading factor provides more data rate compare to higher spreading factor, so we can use either of SF depending upon application for which protocol is designed.

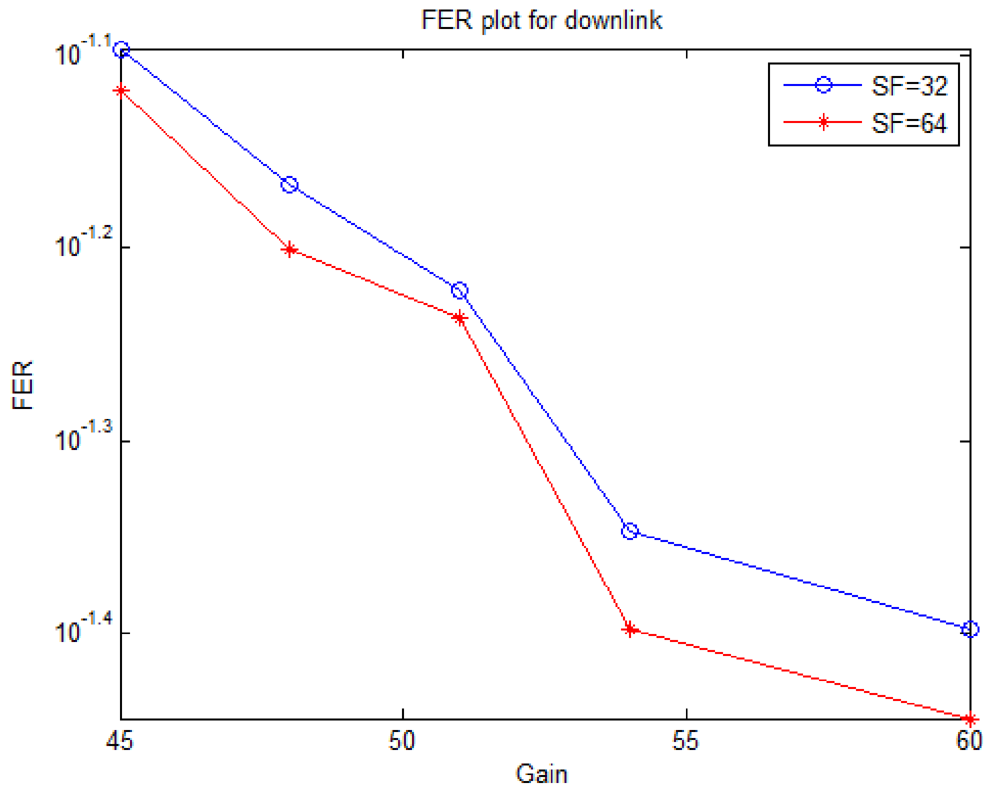


Figure 4.3 FER plot 1:N Downlink

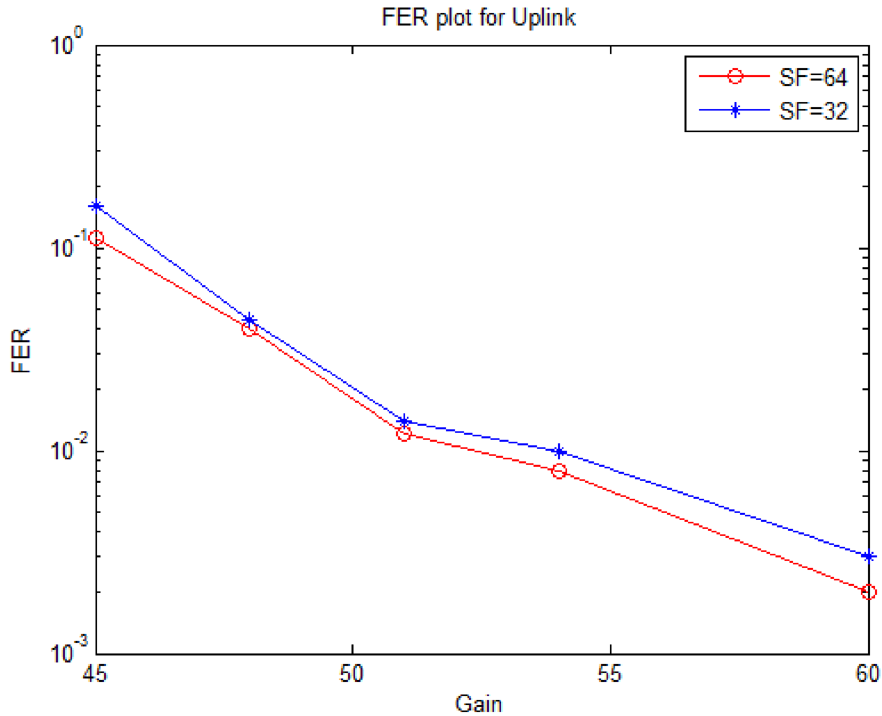


Figure 4.4 FER plot for 1:N Uplink

Also, average FER for different distances is calculated. FER plot with varying Tx and Rx gain at three positions shown in fig.

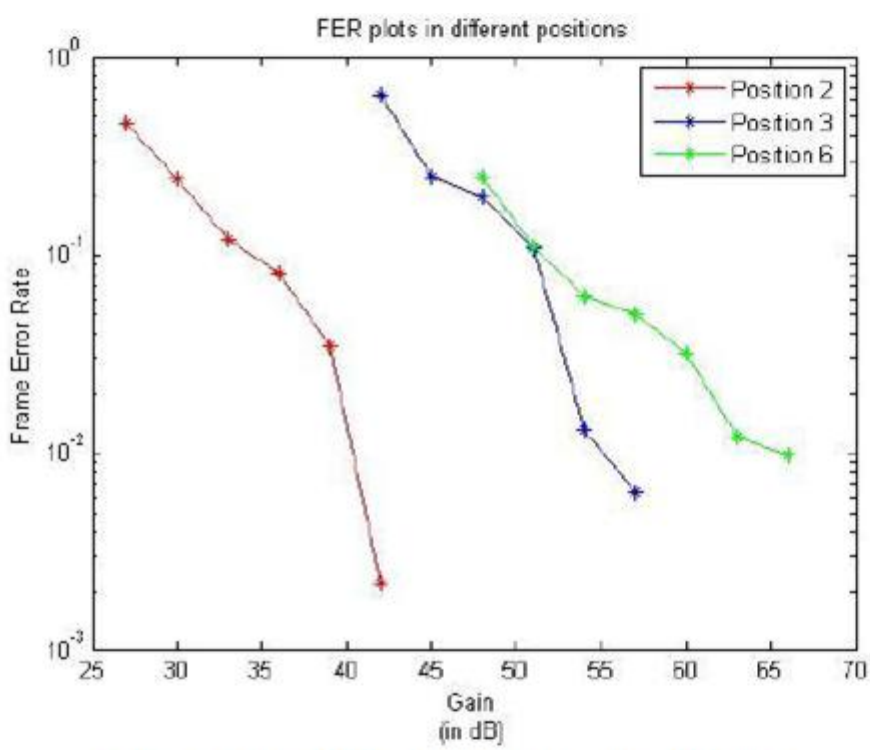


Figure 4.5 FER plot for 1: N at different positions

Also, To increase reliability of 1: N protocol packets which has been either damaged or lost are retransmitted. Retransmission rate which is defined by below equation has been calculated for spreading factor of 64 with different gains.

$$\text{retransmission rate} = \frac{\text{no. of packets retransmitted}}{\text{Total no. of packets sent}}$$

As we can see from the figure that retransmission rate decreases as gain increases that is at higher gains only few frames are getting lost or damaged because of channel errors.

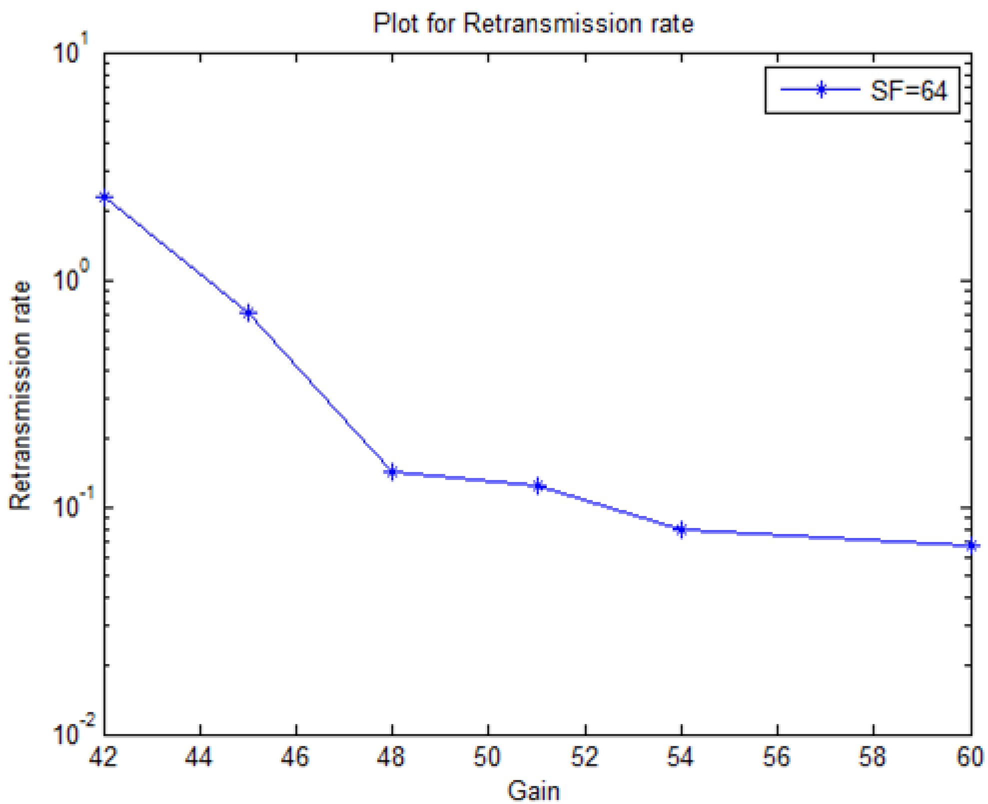


Figure 4.6 Retransmission rate for 1:N

4.2 Performance evaluation of N to N network

4.2.1 Experimental set up for N : N

In this case, again experiments are performed in the machines lab with the three nodes which are connected in the arrangement as shown in the figure 4.3 with their relative distances. The aim of experiment was to check the performance of MAC layer congestion control protocol implemented for RPW devices connected in peer to peer (N:N) fashion.

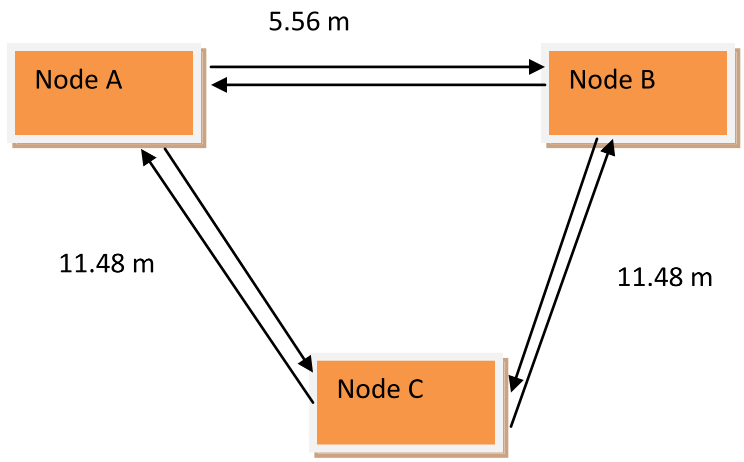


Figure 4.7 experimental layouts for peer to peer

4.2.2 Performance analysis for N : N

Reliable protocol provides reliability properties with respect to the delivery of data to the intended recipient(s), as opposed to an unreliable protocol, which does not provide notifications to the sender as to the delivery of transmitted data. In order to provide reliability for N to N MAC protocol, when one node transmits a data packet, receiver will transmit an ACK/data frame in response. Also, if transmitting node does not receive acknowledgement from receiver either because packets have been lost or damaged then it will retransmit the data packet for some fixed number of the times. We have allowed retransmission of lost packets, thereby making sure that all data transmitted is (eventually) received. We make use of combination of acknowledgements and retransmission of missing or damaged packet to increase reliability of protocol.

Performance measure parameters for MAC protocol are throughput, retransmission rate with varying waiting or back off time, slot duration and traffic symmetry.

Mac Throughput:

Mac throughput can be defined as:

$$\text{MAC Throughput} = \frac{\text{MAC payload(in bits)}}{\text{Inter frame duration}} * \frac{\text{No. of packets received successfully}}{\text{Total no. of packets sent}}$$

Mac payload is of 8 bytes. Inter frame duration is kept as 122 ms for SF of 64 and 83 ms for SF of 32. Total no. of data packets that were sent in each transmission was 1200 in which 1000 data packets were successfully received. Throughput in bytes per second with different spreading factor is shown in following table:

Spreading factor	Mac Throughput(Kbps)
32	0.642
64	0.440

Inferences:
For lower spreading factor (SF) throughput is more and vice versa. It can be further increased by reducing SF. But the cost it has to pay for increasing SF is bit error rate will increase.

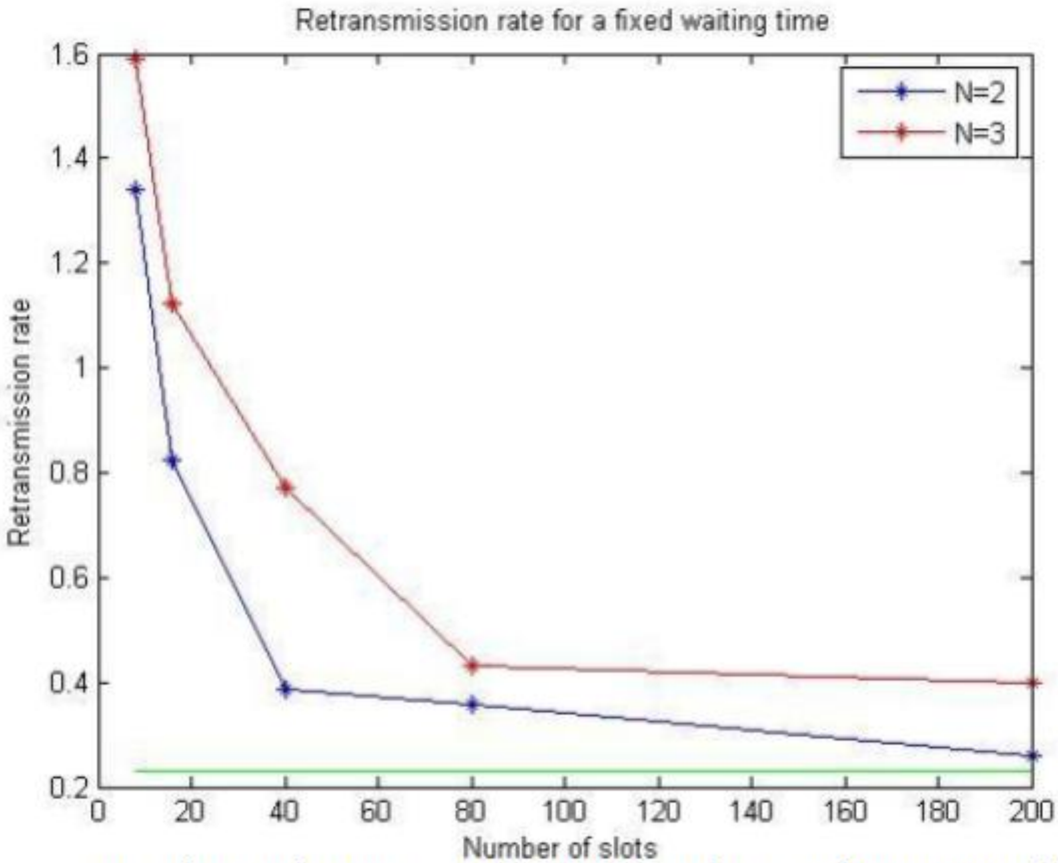
Retransmission rate:

As we have seen in chapter 3 that total waiting is divided into discrete slots. Theoretically, for a fixed waiting (back off) time, as number of slots increases that is as each slot duration decreases probability of choosing same slot for all the clients is decreases since it increases randomness. Similarly, if we keep slot duration fixed and increase the waiting time, number of slots will increase which will again increase randomness and probability of collision will reduce. Experimental results are shown in figure 4.8 & 4.9 with varying slot duration while keeping the waiting time constant and vice versa. Performance with asymmetric traffic is also checked.

In first case, no. of slots is varied from 20 to 200 by changing slot duration from 2 to 50 ms and waiting time is kept constant at 400 ms. Retransmission rate is defined as:

$$\text{retransmission rate} = \frac{\text{no.of packets retransmitted}}{\text{Total no.of packets sent}}$$

Retransmission rate with respect to the no. of slots is plotted when there are two and three nodes present in the network respectively.



Green line in plot represents the case when there is only one way traffic

Figure 4.8 Retransmission rate for N: N for fixed waiting time

Inferences:

1. As the slot duration increases, number of slots increases. Thus we see a decrease in the retransmission rate due to less probability of collisions.

2. As number of nodes increases, the traffic increases causing more collisions which leads to increase in retransmission rate.

In case 2, waiting time is varied from 400 to 800 ms. so it increases no. of slots which varies from 10 to 80. This scenario is shown in the figure 4.9

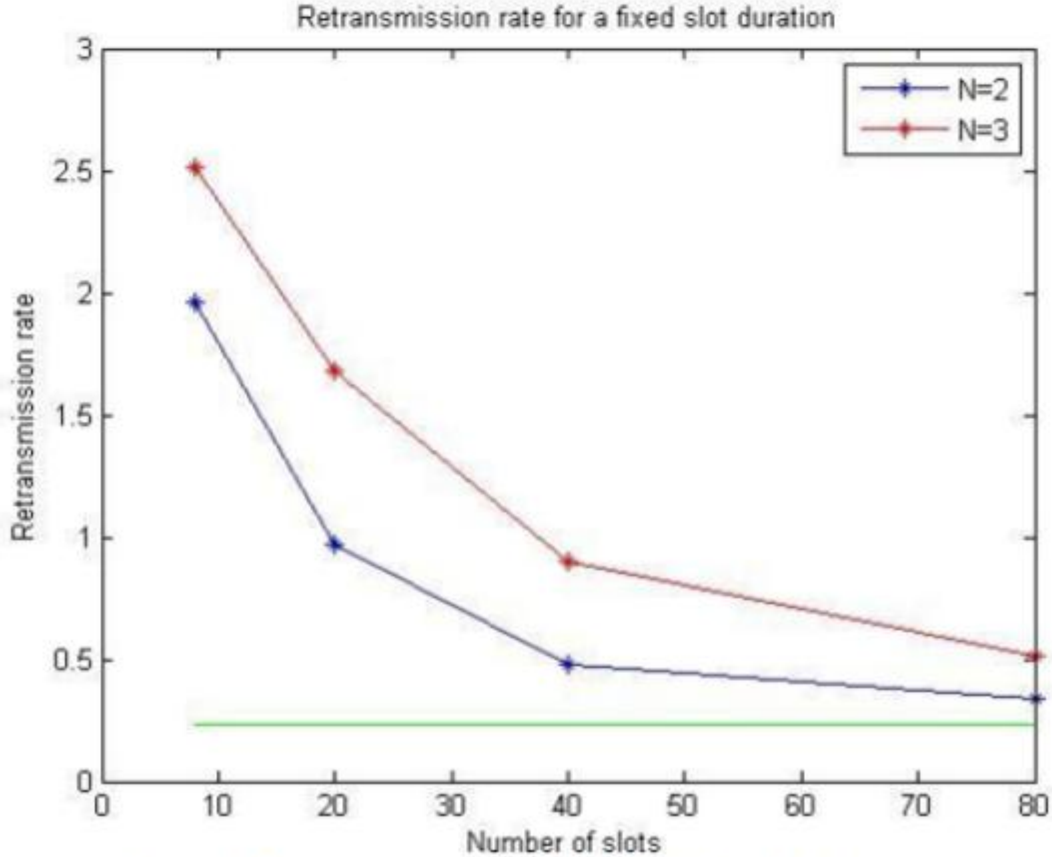


Figure 4.9 Retransmission rate for N: N for fixed slot duration

Inferences:

1. As the waiting time increases, number of slots increases, which leads to a decrease in the retransmission rate due to less probability of collisions.
2. As number of nodes is increased, the traffic in the channel increases causing more collisions which leads to an increase in the retransmission rate.

In third scenario, traffic is kept asymmetric i.e. all devices in the network do not have same data rates. One node can transmit and receive data at the rate higher or lower than the other node. So we have defined a new term called as traffic ratio which is defined as the ratio of loads in different nodes. We have fixed load in one node called as reference node and varied load in other nodes.

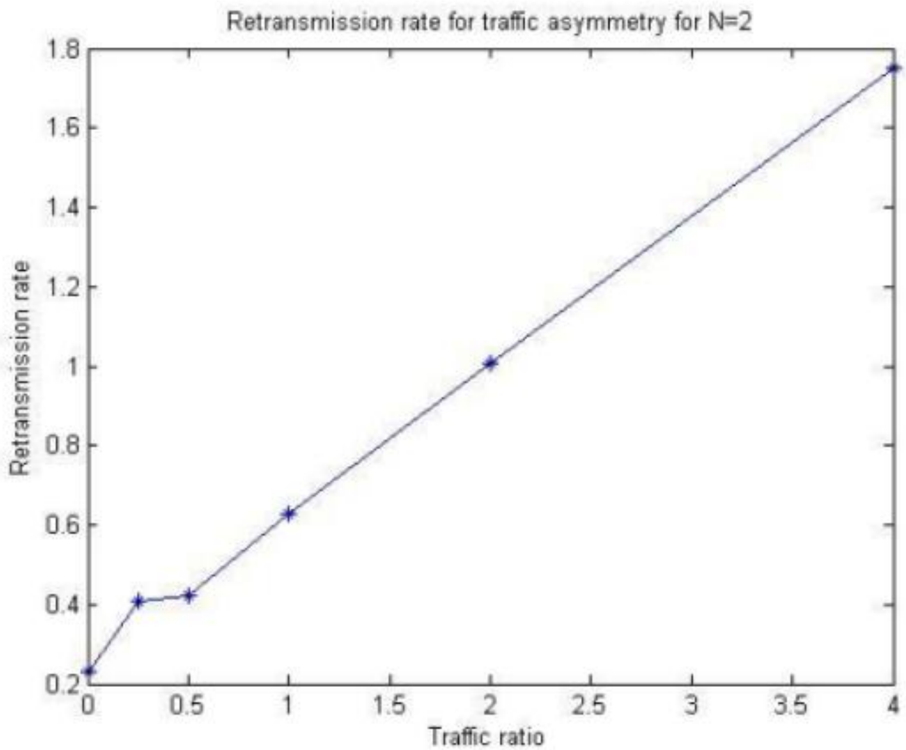


Figure 4.10 Retransmiision rate for traffic asymmetry for N=2

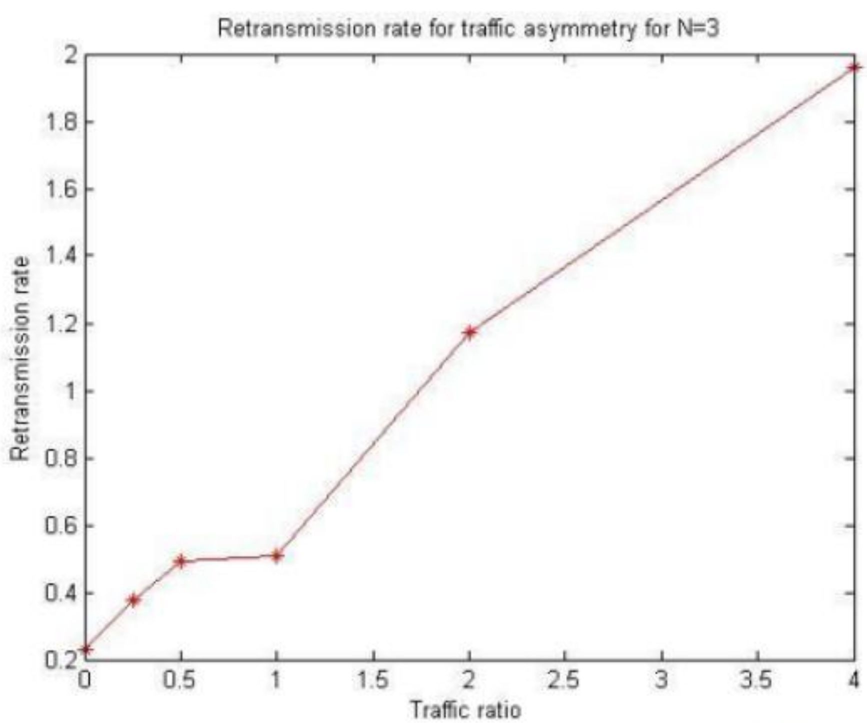


Figure 4.11 Retransmission rate for N: N for traffic asymmetry for N=3

Inferences:

1. As the traffic in the channel increases, the collision probability increases causing more retransmissions.

   Above results show that channel sensing algorithm to avoid collisions in N to N network is working reliably as expected.

CHAPTER 5

CONCLUSION AND FUTURE WORK

In this thesis, we have discussed about physical layer sequence of operations performed at transmitter and receiver. We have discussed about PHY layer algorithms such as frame synchronization, frequency offset estimation and correction and channel estimation and equalization.

After completion of physical layer, our next task was to propose mechanism for channel access that make it possible for several terminals or network nodes to communicate within a multiple access network that incorporates a shared medium. We have designed and implemented medium access protocol for two types of network architecture, one is for star topology based network (1: N) also called as centralized network and the other one is peer to peer (N: N) network. In centralized (1 to N) network; only master can initiate communication in polled mode so there is no chance of collision. However, In peer to peer network any node can send to any other node present in the shared network. So there are chances that two or more nodes can send packets at the same time and collision may happen. To avoid collisions we have proposed channel sensing algorithm where we have discussed about different scenario possible in shared medium.

We have also discussed the real time implementation of the proposed protocols. We have analyzed performances of two proposed MAC protocols. Frame error rate (FER) for uplink and downlink for star network shows that as transmitter gain increases FER reduces. It is intuitive since increase in gain will increase transmitter power which in turn will increase signal to noise ratio (snr) at the receiver which reduces the probability of bits being in error. FER for different spreading factors is also evaluated which shows that MAC protocol performs almost similar for spreading factor of 32 and 64. So depending upon application lower spreading factor can be chosen since it provides higher data rate. Performance measure for N to N network was throughput and retransmission rate. It has been shown that as spreading factor decreases throughput increases. Retransmission rate for different scenarios has been obtained which confirms reliability of protocol. As number of slots increases i.e. slot duration reduces, probability of selecting same slot, in other words probability of two nodes sending data packet at the same time decreases thus no. of packets getting collided drops off. Collision rate reduces, consequently retransmission rate reduces. So we can say that our channel sensing mechanism is working as expected.

Next stage of project can be conversion of c/c++ code which has been implemented, to hardware description language (HDL)(which is a specialized computer language used to describe the structure and behavior of electronic circuits) in order to implement PHY layer algorithm into hardware.