LORA SIGNAL DETECTION USING SDR WAVEGURU KIT

PROJECT REPORT

Submitted by

TARUN N (201EC270)

YASHWANTHKUMAR R (201EC293)

SUDHARSAN R (211EC520)

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING

BANNARI AMMAN INSTITUTE OF TECHNOLOGY
(An Autonomous Institution Affiliated to Anna University, Chennai)
SATHYAMANGALAM-638401

ANNA UNIVERSITY: CHENNAI 600 025

OCTOBER 2023

LORA SIGNAL DETECTION USING SDR WAVEGURU KIT

PROJECT REPORT

Submitted by

TARUN N (201EC270)

YASHWANTHKUMAR R (201EC293)

SUDHARSAN R (211EC520)

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING

BANNARI AMMAN INSTITUTE OF TECHNOLOGY
(An Autonomous Institution Affiliated to Anna University, Chennai)
SATHYAMANGALAM-638401

ANNA UNIVERSITY: CHENNAI 600 025

OCTOBER 2023

BONAFIDE CERTIFICATE

Certified that this project report **"Lora Signal Detection Using SDR Waveguru Kit"** is the bonafide work of **"TARUN N (201EC270), YASHWANTHKUMAR R (201EC293) and SUDHARSAN R (211EC520)"** who carried out the project work under my supervision.

16/05/2024

Dr Poongodi C

HEAD OF THE DEPARTMENT

Department of Electronics and Communication Engineering

Bannari Amman Institute of Technology

Mr Baranidharan V

SUPERVISOR

Department of Electronics and Communication Engineering

Bannari Amman Institute of Technology

Submitted for Project Viva Voce examination held on ....

Internal Examiner I

Internal Examiner II

DECLARATION

We affirm that the project work titled "**Lora Signal Detection using SDR Waveguru kit**" being submitted in partial fulfilment for the award of the degree of **Bachelor of Engineering in Electronics and Communication Engineering (ECE)** is the record of original work done by us under the guidance of **Mr V Baranidharan**, Assistant Professor, Department of Electronics and Communication Engineering. It has not formed a part of any other project work(s) submitted for the award of any degree or diploma, either in this or any other University.

TARUN N	YASHWANTHKUMAR R	SUDHARSAN R
(201EC270)	(201EC293)	(211EC520)

I certify that the declaration made above by the candidates is true.

16/05/2023

Mr V Baranidharan
Assistant Professor,
Dept. of ECE

ACKNOWLEDGEMENT

We would like to enunciate heartfelt thanks to our esteemed Chairman **Dr S V Balasubramaniam**, the respected Director **Dr M P Vijaykumar** and the respected Principal **Dr C Palanisamy** for providing excellent facilities and support during the course of study in this institute.

We are grateful to **Dr C Poongodi**, Head of the Department, Department of Electronics and Communication Engineering for her valuable suggestions to carry out the project work successfully.

We wish to express our sincere thanks to Faculty Guide **Mr V Baranidharan,** Assistant Professor, Department of Electronics and Communication Engineering for his constructive ideas, inspirations, encouragement, excellent guidance and much needed technical support extended to complete our project work.

We would like to thank our friends, faculty and non-teaching staff who have directly and indirectly contributed to the success of this project.

TARUN N (201EC270)

R. yashwanthkumar.

YASHWANTHKUMAR R (201EC293)

SUDHARSAN R (211EC520)

ABSTRACT

Software-Defined Radio (SDR) that can be reconfigured and reprogrammed, is one of the key enabling technologies to use unlicensed devices in licensed bands. The use of software processing to implement the radio system's operations is referred to as software-defined. The main aim of our project is to develop a spectrum sensing method that is best suited for detecting LoRa operating signals as a primary user on the SDR platform. means of USRP boards. The USRP platform is a high-quality realization of SDR. It allows users various functionalities to achieve efficient, real-time realization of very complicated wireless systems that operate in the radio frequency (RF) band. The USRP platform converts an analog signal in the RF band into the digital baseband signal. A programmable USB 2.0 controller communicates between USRP and GNU Radio. The parameters at the receiver were also set to the same centre frequency of LoRa signal, sample rate, gain, and FFT size. Data from the RF spectrum was collected by the USRP Source block. Afterwards, the signal goes through chains. We can observe the received signal through the FFT sink block. The baseband receiver can be used to capture baseband samples from the air. The captured data is put forward in the on-board RAM of the supported USRP hardware to ensure contiguous and high-speed data capture. The capture function returns the data matrix where the two columns correspond to the two channels. Next, we sample the captured data and shift the spectrum to plot the complete frequency band together. Finally, we plot the LoRa frequency spectrum using the spectrum analyser. This will help us to detect the illegal usage of the LoRa signal.

Keywords:

Software Defined Radio, Radio Frequency, Fast Fourier Transform, LoRa Signal, Detection.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

In order to detect and capture the Lora Signals at different operating frequencies (433 MHz, 868 MHz, or 915 MHz) with different modulation schemes, we are planning to use the Spectrum sensing concept by SDR RF waveguru kit. The SDR USRP hardware consists ofa baseband receiver that provides a capture function to capture baseband LoRa signals from the air. This baseband receiver is designed in such a way to receive, sample the captured data and shift the spectrum to plot the complete frequency band together. A Software-Defined Radio (SDR) is a radio communication system where components that have typically been implemented in hardware (e.g., mixers, filters, amplifiers, modulators/demodulators) are instead implemented by means of software on a computer or embedded system.

SDR kits are popular among radio enthusiasts, researchers, and hackers as they allow for a wide range of radio signal processing tasks. Typically, an SDR kit will consist of a hardware component (the radio itself) and softwarethat interfaces with it. You'll need to connect the SDR hardware to your computer. Depending on the specific kit, this connection could be via USB or other interfaces. Installing the software that accompanies the SDR kit. This software is often provided by the manufacturer or developed by the open-source community. Popular SDR software includes GNU Radio, SDR# (SDRSharp), and CubicSDR, among others.

SDR kits are versatile and offer a great platform for experimenting with radio signals. learn about signal processing, radio protocols, and even try to reverse engineer certain signals. To get the most out of SDR kit, it's recommended to explore the online communities, forums, and tutorials related to SDR technology. There's a wealth of information available that can help you understand how to use the kit effectively.

1.1 REQUIREMENTS FOR GNU RADIO SOFTWARE

Driver Installation: Depending on the operating system you're using; you might need to install drivers for the SDR hardware to communicate properly with your computer.

Table 1.1: LORA REGIONAL CHANNEL FREQUENCIES

Name of the Region	LoRa Channel Frequency	Frequency Plan
Europe	863 — 870 MHz	EU863 — 870
USA	902 — 928 MHz	US902 — 928
China	470 — 510 MHz	CN470 — 510
Australia	915 — 928 MHz	AU915 — 928
India	865 — 867 MHz	IN865 — 867

1.2 Software Usage:

Frequency Selection: Most SDR software allows you to select the frequency range to listen to or interact with. This could include FM radio, amateur radio bands, Wi-Fi frequencies, and more.

Modulation/Demodulation: SDR kits can process different types of radio signals, such as AM, FM, SSB, CW, etc. The software will allow you to select the appropriate modulation type for the signal.

Signal Analysis: analyze the spectrum, waterfall plots, and other graphical representations of the received signals.

Signal Decoding: Depending on the signal receiving, use the software to decode digital signals, such as weather data, digital radi, etc.

CHAPTER 2

LITERATURE SURVEY

LoRa signals are widely used in various Internet of Things (IoT) devices, Smart Cities, Defense corridors, Low Power Wide Area Networks (LPWAN) technologies etc. These technologies focus on long-distance communication applications with high energy constraints. These constraints generally affect the cost of reducing the bit error rate. Many researchers have come up with different types of LoRa signal detection techniques to overcome these issues. Some of these techniques are discussed here.

2.1 Maximum Likelihood-based LoRa signal Receiver using GNU radio

Xhonneux et al., have proposed a new type of signal detection techniques based on the maximum likelihood for two different users [1]. LoRa has emerged as one of the most popular LPWAN protocols due to its long-range connectivity, resilience to noise, and simple modulation scheme. However, LoRa uses an ALOHA random access scheme results in uncoordinated transmissions and frequent packet collisions at the gateways, reducing overall network throughput. The successive interference cancellation (SIC) receivers are always supported only for the single-user detector. For LoRaWAN or LPWAN based applications, the concept of Non Orthogonal Multiple Access (NOMA) is exploited. Such techniques do not support multi-users. To overcome these issues, the authors implemented and evaluated a multi-user LoRa receiver capable of decoding two interfering LoRa signals using the same spreading factor. This will significantly increase gateway capacity by enabling non-orthogonal multiple access.

A two-user detector derived mathematically from first principles using the maximum-likelihood criterion. However, maximum-likelihood sequence estimation has prohibitive complexity for long LoRa packets. They have designed symbol-by-symbol decisions to restrict the complexity to each decoding window. The receiver synchronizes to the earliest arriving user, then jointly decodes both by marginalizing phase and evaluating metrics on the contribution of each signal. The proposed receiver architecture has three main stages - preamble detection, synchronization, and demodulation. The preamble detector uses techniques like geometric averaging of repeated up-chirps to robustly estimate parameters of a new incoming user even in interference. The synchronization block implements a state machine tracking one or two active users. It leverages the preamble detector's outputs to synchronize to the strongest user as depicted in the Figure 2.1. Frequency and timing offsets are corrected before the baseband samples are passed to the demodulator.

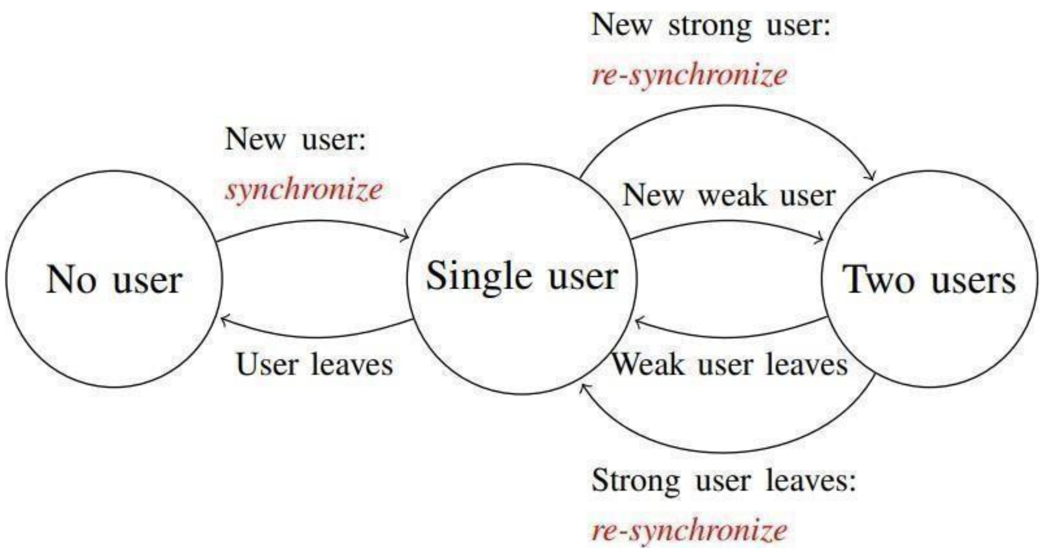

Figure 2.1: Representation of Receiver's Finite State Machine

The demodulator implements the reduced-complexity maximum-likelihood algorithm. It first attempts to remove the contribution of the stronger user, and then uses partial Fourier transforms to match and filter the remaining signal for the second user's symbols across two windows. To avoid phase tracking, the phase of the stronger user is estimated from the Fourier transform peak and marginalized. This relies on accurate synchronization. Soft symbol decisions are fed back to assist in resolving the weaker user's symbols. The methods are implemented on GNU Radio software-defined radios. The synchronization scheme is shown to provide accurate parameter estimation even with interference. In terms of decoding performance, the receiver exploits timing offsets between users to better separate their signals.

2.2 GNU radio LoRa Physical Layer implementation

Joachim Tapparel et al., have proposed a standard –compatible LoRa PHY based software-defined algorithm prototype to receive the LoRa signals effectively. In this proposed work, transmit and receive chains include all required functions like whitening, coding, interleaving, modulation/demodulation, frame synchronization, and offset correction. They validated the SDR implementation experimentally using Universal Software Radio Peripheral (USRP) devices and compared performance against prior research. The nature of the LoRa PHY has restricted innovation and research to improve it. The transmitter and receiver chains of the LoRa PHY layers are depicted in Figure 2.2. Attempts to reverse engineer LoRa have recovered some details but lack robustness for low SNR operation. Existing LoRa SDR implementations fail to properly handle impairments like sampling and carrier frequency offsets. This limits their usable SNR range, preventing full experimental analysis. This proposed algorithm is based on the synchronization and offset correction techniques.

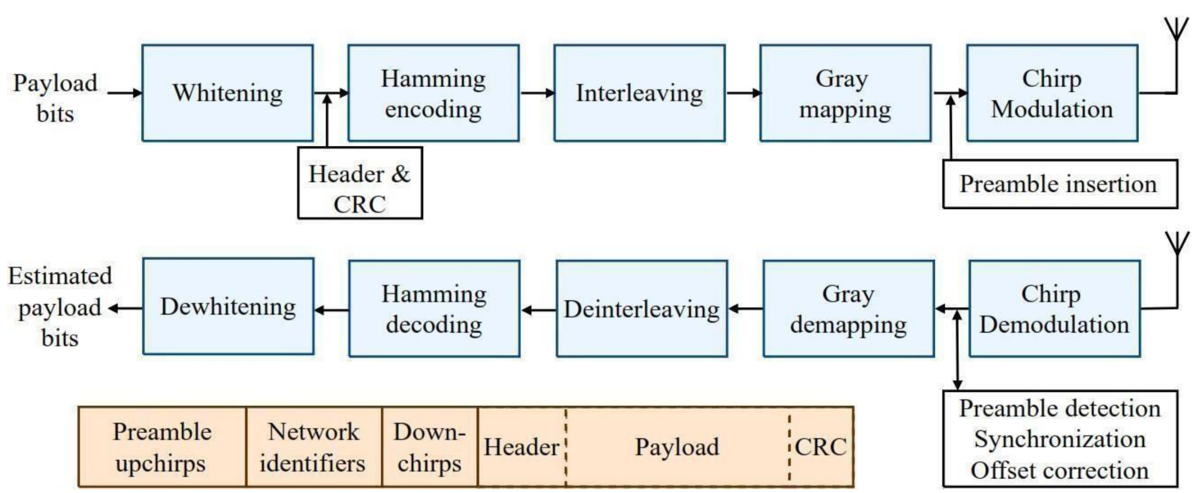

Figure 2.2: Transmitter and Receiver chains along with data packet structure

Joint estimation and compensation of integer and fractional parts of sampling time offset (STO) and carrier frequency offset (CFO). Preamble detection determines coarse offsets. Further fractional STO and CFO estimation leverage the frequency domain, using interpolation between spectral peaks. Compensation is applied in time and frequency domains accordingly. This enables the SDR implementation to reliably operate down to low SNRs. The robust synchronization and impairment mitigation allow exploring performance at low SNRs. This can facilitate analysis and improvements to error rate performance and range. They experimentally validate the SDR implementation using USRP radios, achieving BER within 1 dB in an AWGN channel even with full oscillator drift.

2.3 Multi-node SDR for signal monitoring

Yunhui Yi et al. have proposed new novel architecture based on the neural network for LoRa signal detection and parameters estimation. LoRa networks grow, collisions between packets will increase which degrades efficiency. This includes multiple monitoring nodes, a data fusion center, and a user application. The nodes use software-defined radios for LoRa signal detection and parameter estimation. Each node preprocesses simulated modulated signals to train a neural network classifier. In detection, energy detection identifies a wireless signal, and then the neural network classifies the modulation. Nodes upload results to the fusion center which improves k-means clustering for multi-node data processing. The monitoring node hardware includes an omnidirectional antenna, RF filtering and amplification, and a LimeSDR platform as depicted in Figure 2.3. The software implements energy detection and a 3-layer back propagation neural network.

Neural network construction: Implementation of the BP neural network algorithm, and make use of the neural network to complete the modulation recognition of the signal. The neural network classifier is designed with a three-layer network structure: input layer, a hidden layer and an output layer. The input is the five parameter data extracted from the modulated signal, and the number of hidden layer neurons is taken as 10. The output is the result of the neural network identifying the modulated signal. The activation function used by the input layer to the hidden layer is the S-type logarithmic function. The transfer function used by the hidden layer to the output layer is the S-type tangent function. A noise-free signal plus a fixed signal-to-noise ratio noise can obtain a modulated signal under a certain noise condition. In a certain signal-to-noise ratio case, each node randomly generates 400 sets of data for each type of modulated signal. Then extracts parameters from the 400 sets of data as training set data, and sends them to each node's neural network for training. The nodes get different trained neural networks. Five signal parameters are extracted as inputs: spectral density of normalized instantaneous amplitude, standard deviation of instantaneous phase nonlinear component, standard deviation of instantaneous frequency, and standard deviations of instantaneous amplitude and phase. The hidden layer has 10 neurons. Signal detection algorithm processing: In the software radio detection process, the energy detection algorithm is used to calculate the spectrum data of the signal, and estimating the signal to noise ratio of the signal. It is judged whether there is a wireless signal in the signal. Then, if wireless signals exist, the data is preprocessed and sent to the neural network which corresponds to the signal to noise ratio. And the neural network identifies whether the signal is LoRa and the detection result is uploaded. Different nodes train networks on randomized data.

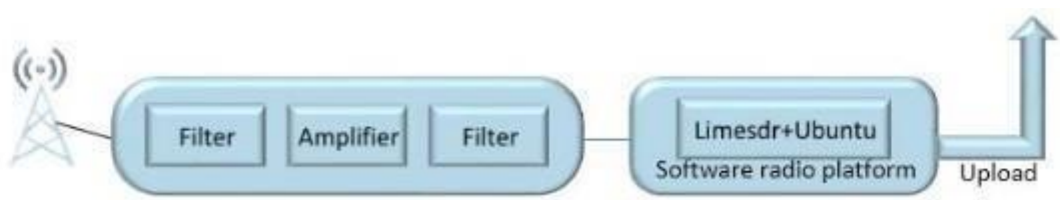

Figure 2.3: Monitoring node using SDR

In detection, energy detection estimates the signal-to-noise ratio and determines if a wireless signal is present. If so, the corresponding neural network classifies the modulation. Nodes upload LoRa results to the fusion center. It clusters the data using Euclidean distance to measure error from centers. Errors are weighted by node hardware accuracy. The k-means algorithm is improved by not presetting cluster numbers, instead setting a termination condition to limit cluster radius and reduce iterations. The cluster centers are continually updated to refine the target LoRa signal detection.

2.4 LoRa decoding using modern LPWAN based SDR

A modern LPWAN based LoRa decoding method is proposed by Matthew Knight et al. using SDR. As LoRa is closed source, the authors had to conduct blind signal analysis to understand the modulation and encoding schemes. They used a Microchip RN2903 LoRa mote to transmit messages to analyze and an Ettus USRP SDR and GNU Radio to capture and process the signals. Since the technical specifications were not publicly available, the author had to analyze the modulation and encoding through black box testing. The modulation of LoRa uses a proprietary chirp spread spectrum (CSS) modulation. This determined that this modulation encodes data as instantaneous frequency changes in chirps, where chirps are signals with linearly changing frequencies. By generating local chirps and mixing them with the received signal, the data chirps can be "de-chirped" into an MFSK-like signal that can be demodulated by taking FFTs.

Forward error correction enables bitsdamaged during transmission to be recovered and corrected. It is similar to using a single parity bit or checksum, but it goes further by providing error correction under certain scenarios as well. LoRa uses Hamming FEC with a variable code- word size ranging from[5:8] bits and fixed data size of 4 bits per code word. This proper FFT timing synchronization is critical to accurately resolve the symbols. This is achieved through detecting the preamble, synchronizing to the start frame delimiter, and then taking FFTs atthe symbol rate.

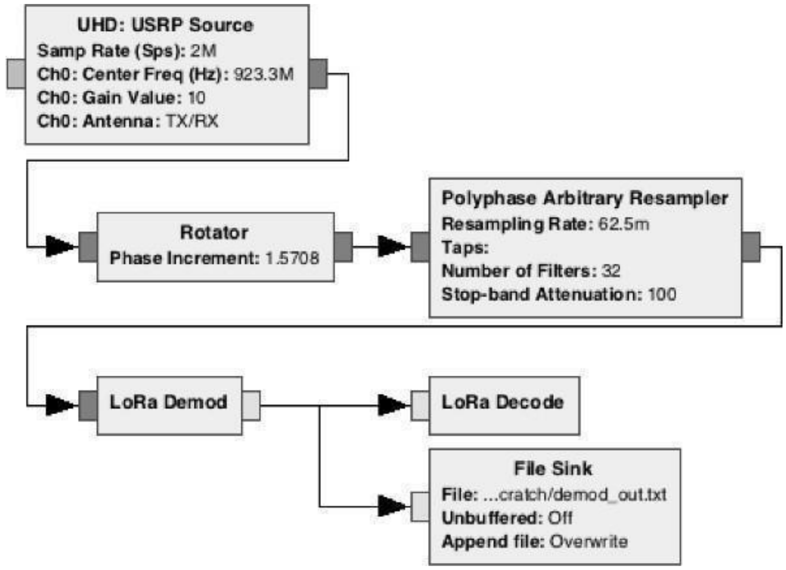

Figure 2.4: Detailed Flow graph of LoRa decoding for LPWAN

The symbols demodulation is carried out by using decoded to extract the raw data. The encoding schemes process is based on the gray indexing to prevent off-by-one errors; whitening to induce randomness, interleaving to handle bursts of interference, and Hamming forward error correction. The existing implementation does not decode and process LoRa's optional PHY header. This is due to the lacking hardware capable of disabling the header – since the header is always present, it was not able to be re- placed with 0s to extract the portion of the whitening sequence that is applied to it. The FFT synchronization requires overlapped FFTs to precisely frame the start delimiter to avoid symbol collisions. In order to determine interleaved structure of the complex process, crafting messages to analyze code word positions within the interleaved data.

CHAPTER 3

OBJECTIVES AND METHODOLOGY

Software-Defined Radio (SDR) that can be reconfigured and reprogrammed, is one of the key enabling technologies to use unlicensed devices in licensed bands. The use of software processing to implement the radio system's operations is referred to as software-defined. The main aim of our project is to develop a spectrum sensing method that is best suited for detecting LoRa operating signals as a primary user on the SDR platform by means of USRP boards. The USRP platform is a high-quality realization of SDR. It allows users various functionalities to achieve efficient, real- time realization of very complicated wireless systems that operate in the radio frequency (RF) band. The USRP platform converts an analog signal in the RF band into the digital baseband signal. A programmable USB 2.0 controller communicates between USRP and GNU Radio.

The parameters at the receiver were also set to the same center frequency of LoRa signal,sample rate, gain, and FFT size. Data from the RF spectrum was collected by the USRP Source block. Afterwards, the signal goes through chains. We can observe the received signal through the FFT sink block. The baseband receiver can be used to capture baseband samples from the air. The captured data is put forward in the onboard RAM of the supported USRP hardware to ensure contiguous and high-speed data capture.

The capture function returns the data matrix where the two columns correspond to the two channels. Next, we sample the captured data and shift the spectrum to plot the complete frequency band together. Finally, we plot the LoRa frequency spectrum usingthe spectrum analyzer. This will help us to detect the illegal usage of the LoRa signal.

3.1. Methodology

The modulation spreads a narrow band signal over a wider bandwidth leading to an increase in signal-to-noise ratioand resilience against interference. LORA uses forward error correction and can vary parameters like bandwidth, spreading factor and coding rate to trade-off between range, data rate, power consumption etc. Software defined radios (SDRs) have emerged as an extremely flexible tool for wireless experimentation and learning. SDR uses software fordigitizing RF signals and signal processing functions traditionally done in hardware. Thisallows implementing various modulation schemes, encoding methods, networking protocols etc in software. Popular SDR hardware platforms include RTL- SDR, HackRF, LimeSDR and USRP. The Waveguru SDR kit is designed for beginners to easily get started with SDR and wireless experiments. The key steps would include:

- Assembling the SDR hardware with a suitable antenna like the included 433/868MHz monopole antenna. An outdoor antenna can help extend range.
- Installing SDR software like SDR# on your computer and getting familiar withcontrols like tuning, filters, gain etc.
- Scanning the ISM band and identifying any ambient LORA transmissions from IoT devices. The CSS waveform is easily distinguished from other signals.
- Using the LORA demodulator plugin to decode packet data, configure spreadingfactors and error correction settings.
- Transmitting custom LORA messages by generating CSS waveforms in software and modulating an RF carrier.
- Measuring the range, data rate and analyzing the effect of interference to understand LORA radio characteristics.
- Experimenting with different LORA transmission parameter sets to optimize for range vs data rate as needed.

In summary, detecting and studying real-world LORA wireless signals using the software-defined approach enables deep learning about modern IoT modulation techniques and wireless systems. The Waveguru SDR kit provides an accessible toolkit to delve into the physical layer fundamentals underlying long range IoT connectivity and wireless technologies.

The key parameters of LoRa physical layer are bandwidth, spreading factor,coding rate, preamble length, sync word, headers, payload and CRC. LoRa frames consistof the preamble, sync word, header, payload and CRC in order. The chirp signals encode the data symbols which are spread by the spreading factor and whitened with a pseudo- random sequence.

Techniques:

- Use preamble to detect frames and find coarse timing
- Refine using sync word to precisely estimate timing offset and frequency offset.
- FFT-based symbol processing to estimate chirp frequency, magnitude and phase.
- Regenerate chirp signal using estimated parameters and subtract via SIC.
- Carefully handle peak overlaps to avoid erroneous signal cancellation.
- Use modest block sizes to limit complexity and allow faster decoding.

Modulation: LoRa uses a proprietary chirp spread spectrum (CSS) modulation. This determined that this modulation encodes data as instantaneous frequency changes in chirps, where chirps are signals with linearly changing frequencies. By generating local chirps and mixing them with the received signal, the data chirps can be &quoted-chirped" into an MFSK-like signal that can be demodulated by taking FFTs.

Forward error correction: It enables bits damaged during transmission to be recovered and corrected. It is similar to using a single parity bit or checksum, but it goes fur- ther byproviding error correction under certain scenarios as well. LoRa uses Hamming FEC witha variable code- word size ranging from [5:8] bits and fixed data size of 4 bits per codeword.

Synchronization: Proper FFT timing synchronization is critical to accurately resolve thesymbols. This is achieved through detecting the preamble, synchronizing to the start frame delimiter, and then taking FFTs at the symbol rate.

Decoding: The demodulated symbols must be decoded to extract the raw data. The encoding schemes: gray indexing to prevent off-by-one errors, whitening to induce randomness, interleaving to handle bursts of interference, and Hamming forward errorcorrection.

3.2 Flowchart of the proposed work

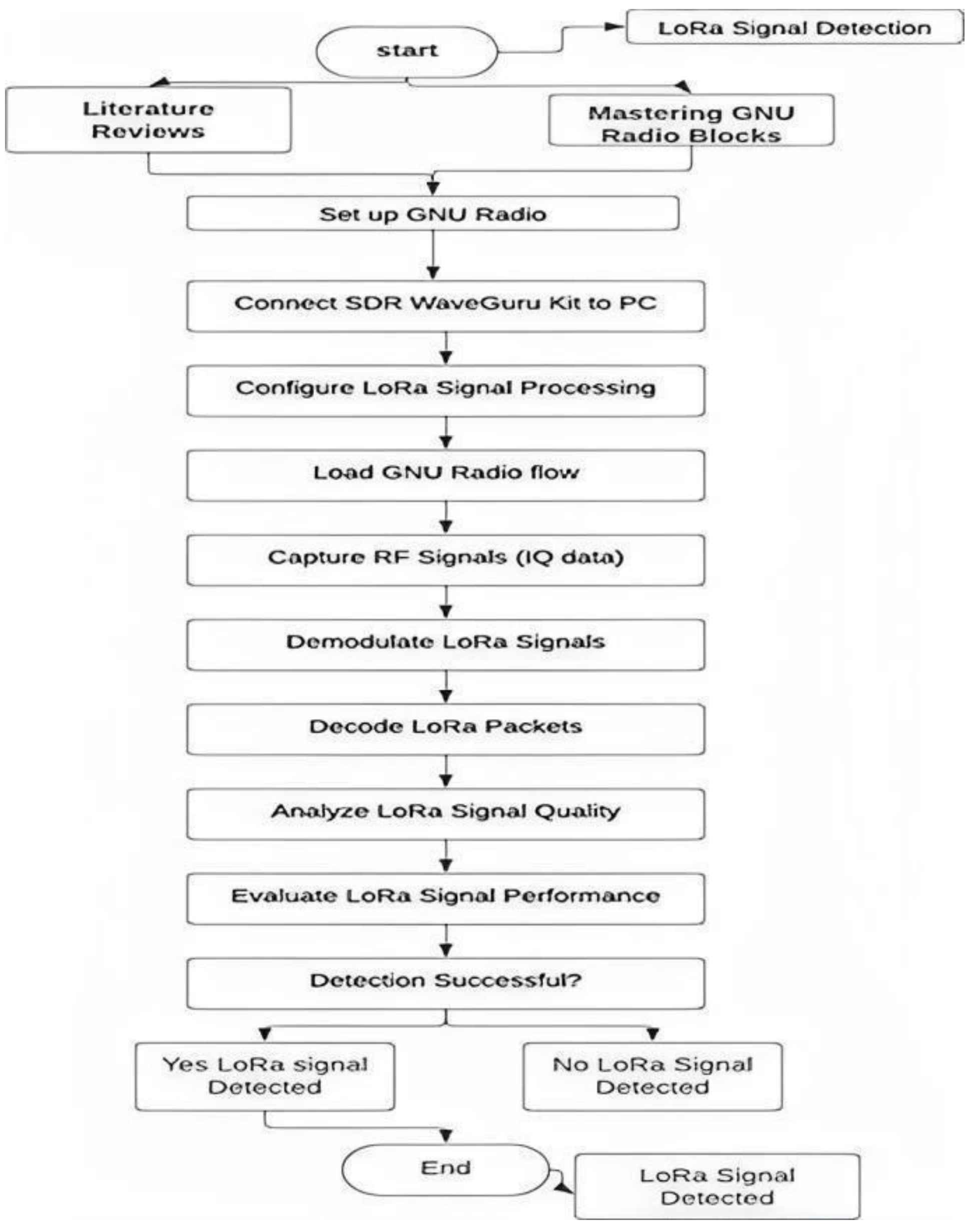

Figure 3.1: Flow Chart

Start: The process begins with initiating the LoRa signal detection using the SDR WaveGuru kit and GNU Radio.

Set up GNU Radio: Prepare the GNU Radio environment on the computer for LoRa signal processing.

Connect SDR WaveGuru Kit to PC: Establish the connection between the SDR WaveGuru kit and the computer.

Configure LoRa Signal Processing: Configure the necessary GNU Radio blocks and modules for LoRa signal demodulation and decoding.

Load GNU Radio flow: Load the GNU Radio flowgraph that contains the LoRa signal processing blocks.

Capture RF Signals (IQ data): Capture the raw In-phase and Quadrature (IQ) data from the SDR WaveGuru kit.

Demodulate LoRa Signals: Process the captured IQ data to demodulate the LoRa signals.

Decode LoRa Packets: Decode the demodulated signals to recover the original LoRa packets.

Analyze LoRa Signal Quality: Analyze the quality and characteristics of the detected LoRa signals.

Evaluate LoRa Signal Performance: Evaluate the performance of the LoRa signaldetection system based on specific metrics and objectives.

Detection Successful? Determine if LoRa signals were successfully detected.

LoRa Signal Detected: If LoRa signals are successfully detected, the process proceeds to this step.

No LoRa Signal Detected: If no LoRa signals are detected, the process concludes without any detected signal.

End: Signal Detected: The process ends after successfully detecting LoRa signals and completing any necessary analysis or evaluation.

CHAPTER 4

PROPOSED WORK

In this project, why have we chosen the loRa module? We use the Lora module for the transmission and receiving part for long-range, wide-area band signals with low power consumption and wireless access. So in this work, we use the LoRa module (433mhz~510mhz) range bandwidth which covers the surrounding radius of 3 km ≈ 8 km. SDR (Software Defined Radio) kit can accept signals ranging from 70 MHz to 6 GHz in real-time.

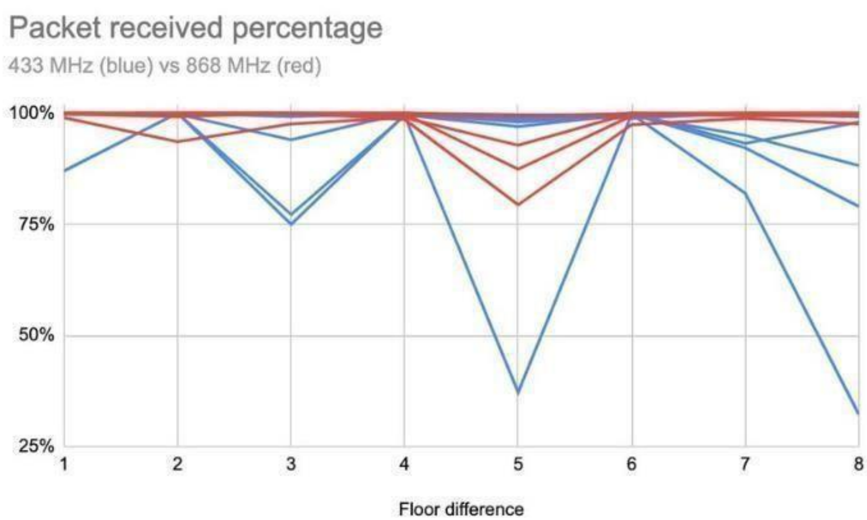

Figure 4.1. Difference in Packet Receiving Graph

In particular, this module is designed only for the specific bandwidth to work efficiently.Instead, it cannot work in different bandwidth ranges. E.g.: Lora32 (433mhz) Transmission & Lora32 (433mhz) Receiver - It works. Lora32 (433mhz) Transmission & Lora32 (868mhz) Receiver - It doesn't work. LoRa Sensing has major advantages in industrial applications like false alarms, and bomb- threatening areas. Mostly this application is more priority used in military base fields and eavesdropping. The eavesdropping in other frames of sentences is nothing but the signals between the user and the client whether the signal has accurate content or data delivery. Signal Calibrations have undergone the process as follows. There are huge differences in the filtration of bands which is beyond its work.

4.1 Communication Part of LoRa:

Work in-band frequency which is Unlicensed or free to access. Chirp SpreadSpectrum modulation to encode information takes wideband linear frequency to make the LoRa Signal a pure ALOHA technique [5]. The Signals are Open access to every user in the world. ALOHA signals mainly work in the data-link layer which follows the procedure of the OSI model. Here we use Arduino IDE for coding and dumping parts for the LoRa Transmitter and LoRa Receiver and a few ports like (COM12-port) depending on the PC configuration it differs but here that is needed to work with it.

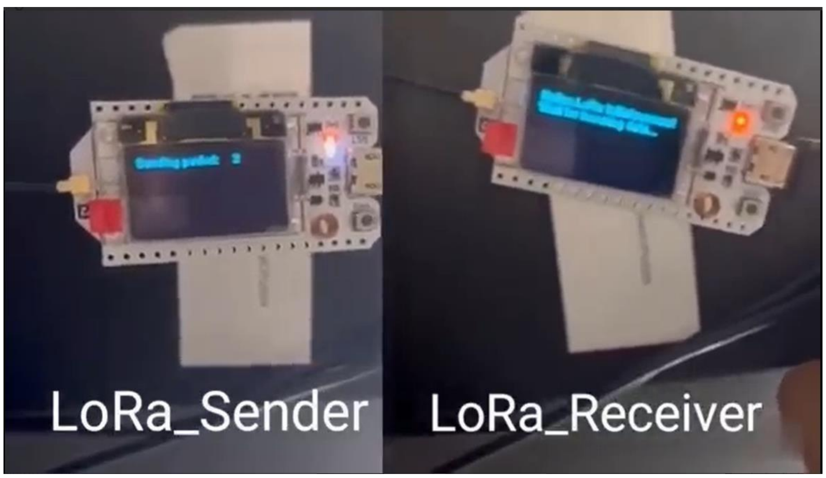

Figure 4.1.1 Communication Tx & Rx

4.1.1 The Modulation Part of LoRa:

The LoRa Communication Part works in unique CSS technology which has a Spontaneous Response of Cyclic shift in frequency. This part undergoes encoding data operations in the carrier signal every part of modulation goes into wireless technology or wireless procedure. Up-chirp LoRa signal works for the bases of Modulation to enrich the signal Classification without any disturbance or noise and it is represented bythe spreading factor which depends on the higher the spreading factor with longer the timeon air follows and it is expressed and follows to complete the cyclic parts of signals.

$$s(t) = \sqrt{\frac{E_b}{N_o}} \sqrt{\mathrm{SF}} \exp\left(j2\pi \left[\left(\gamma(m) + \frac{\beta}{2} t \right)_{\mathrm{mod}\ B} - \frac{B}{2} \right] t \right) \qquad (1)$$

The word ‘Chirp’ defines the signal at which the signal's Frequency rate changes to the fixed rate, at which the signals may be in the form of constant or exponential [21]. The formula 1 in theimage is the equation for the baseband signal of a single-carrier frequency division multiple access (SC-FDMA) system. SC-FDMA is a digital modulation scheme that is used in many wireless communication systems, such as LTE and 5G.

Where s(t) is the baseband signal, SF is the spreading factor, j is the imaginary unit, π is themathematical constant pi, f is the carrier frequency, B is the subcarrier bandwidth, t isthe time, l(m) is the data symbol at index m, N is the number of subcarriers, B2 is the symbol rate, j is the imaginary unit, π is the mathematical constant pi, f is the carrier frequency, and t is the time. The formula can be explained in the following steps:

The data symbols are multiplied by the spreading factor. This is done to spread the energy of the signal over a wider frequency band, which makes it more resistant to interference [20]. The data symbols are modulated onto a complex exponential carrier signal. This is done by multiplying the data symbols by the complex exponential carrier signal. The modulated signal is passed through a filter that has the shape of a raised cosine function. This is done to reduce the spectral sidelobes of the signal. The filteredsignal is transmitted over the air.

At the receiver, the signal is demodulated and the data symbols are recovered. Multiplying the data symbols by the spreading factor. The spreading factor is a numberthat determines how much the energy of the signal is spread over a wider frequency band. A higher spreading factor results in a more spread-out signal, which is more resistant to interference.

Modulating the data symbols onto a complex exponential carrier signal. Modulating the data symbols onto a complex exponential carrier signal is a way of shifting the signal to a higher frequency. This is done by multiplying the data symbols by the complex exponential carrier signal [15] . Passing the modulated signal through a filter that has the shape of a raised cosine function. Passing the modulated signal through a filter that has the shape of a raised cosine function is done to reduce the spectral sidelobes of the signal. Spectral sidelobes are unwanted signals that are generated when a signal is filtered. Transmitting the filtered signal over the air. Once the signal has been filtered, it is transmitted over the air.

4.1.2 Gray Indexing Process for LoRa Signals:

Here, LoRa Signals are gray-indexed for the carryover of the signals for longrange by using the process of the symbol so the loss of signals will be reduced so output can be precise. Gray-indexing refers to the two successive values one bit differs in binary numeral systems. This process is similar to a randomizer operation like adding binary bits to make the flow of signal stronger so no value-added noise will be generated [14]. Here, the LoRa Receiver needs to make inverse gray code operations signals in the form of Symbols; mostly wireless communication systems will use these operations often.

4.1.3 Whitening Process for LoRa Signals:

The Reduce the Process of Randomness symbol and with the XOR operations for the recovery of the Inbuilt clock generator recovery operations. This process is done in the Receiving part of Signals where the Symbols match to the Transmitted Part of Signals [11]. To make sure of the blind Analysis of the signal decoded process. So these techniques have beenintroduced to maintain the consistency of Decoding or Receiving Signals.

4.1.4 The Demodulation Part of LoRa:

This Part acts like a signal receiver and decoding of data in which the down chirp (invert chirp) technique follows to reject the unwanted spikes and noise in the RF signals to have the clarity of data. Every signal needs to follow the shift keying procedure here we use Frequency

Shift Keying (FSK). LoRa encodes and decodesmultiple bits of valuable information into each accurate symbol to have the information correct and appropriate output. This formula gives an accurate and valid definition of chirprate.

$$chirp\ rate = \frac{dfrequency}{dt} = \frac{bandwidth}{2^{spreadingfactor}} \tag{2}$$

The formula 2 in the image is the equation for the chirp rate of a chirp signal. A chirp signal is a signal whose frequency changes over time. Chirp signals are used in a variety of applications, such as radar, sonar, and communication systems. The formula for thechirp rate is as follows:

$$\text{chirp rate} = \text{frequency} * \text{bandwidth} * \text{spreading factor} \tag{3}$$

chirp rate is the rate at which the frequency of the signal changes, frequency is the change in frequency, bandwidth is the change in bandwidth, spreading factor is a factorthat determines how much the energy of the signal is spread out over a wider frequencyband. The formula can be explained in the following steps:

The change in frequency is multiplied by the change in bandwidth. This gives the total amount of frequency change that occurs over the entire signal. The total amount of frequency change is multiplied by the spreading factor. This gives the chirp rate of the signal. Thespreading factor is a number that determines how much the energy of the signal is spread out over a wider frequency band. A higher spreading factor results in a more spread-out signal, which is more resistant to interference. Suppose we have a chirp signal with a change in frequency of 1 GHz and a change in bandwidth of 100 MHz. The spreading factor is 2. The chirp rate of the signal would be calculated as follows:

$$\text{chirp rate} = 1\ \text{GHz} * 100\ \text{MHz} * 2 = 200\ \text{GHz/s} \tag{4}$$

This means that the frequency of the signal changes by 200 GHz per second. Signal Bandwidth to the Spreading factor is the chirp rate definition. Stepwise demodulation techniques undergo chirping format signals to de-chirp the formation of signals. A few steps to make the signal a unique constant frequency include channelization of complex baseband signals, and multiplying the results to have the locally generated chirp signal. "Rotate" operation to make the chirp signal produce the IQ streams and result in the chirp rate 0.

MFSK (Multiple Frequency shift keying) is followed in the de-chirp signal, and conversion from the time domain to the frequency domain includes the FFT (Fast Fourier Transform). To get Accurate Synchronization is solved with the appropriate resolving technique and results in symbols.

4.2 Pack Structure of LoRa Module:

In LoRa the working procedure it follows to have the fine structure of LoRa module packets,

4.2.1 Preamble Actuator:

This part of LoRa Transmitter and Receiver packets has a different conditional variable number (N_{pr}) of Up-chirp & Down-chirp(de-chirp). To work in these conditions, it is necessary to have the default value is 8 (N_{pr}) so this helps in the Preambledetection for a very large number of large ranges of effective Signal-to-noise ratioeliminations.

4.2.2 Payload and CRC:

The message carrier signal has the packet that has the required numbers of payload information. An additional requirement is 16-bit CRC [22]. To make an error-free signal there is a cyclic redundancy check (CRC) involved to finalize the required results.

4.2.3 Cognitive Radio Block:

This Blocks which is required to have high-end hybrid technology that involves the Software Defined Radio (SDR). Cognitive Radio technology is nothing but the changes in the spectrum sensing operations depending on the circumference of the range it changes. Few cases we have noted that the bands in the mobile phones change to the surrounding dependence of 4G to 3G and vice versa. This happens because of the spreadof spectrum communications. The term “Quick adapt” will situate this topic and also the

RSSI operations that take place in this topic. The advantages are the detections of False Alarm which have nothing but the bomb detection frequency ranges and it is highly used in military bases and it is flexible to its surrounding areas.

4.2.4 Energy Detection Block:

Here it is the Energy detection part that makes the spectrum sensing part to detect and work efficiently and the signal presence is a very important part of that frequency sensing spectrum. These signals can be achieved by the comparison algorithm to make it precise and particular threshold value to its dependence level of output energy generations [17]. It doesn’t need any prior information to detect and the conversion pattern is taken over by FFT techniques and vice versa.

GNU Radio Blocks make the part easier to get the real-time measurements with zero corrections and noise levels both in the hardware and software part of SDR and GNU Radio block.

4.3 USRP functionality:

UNIVERSAL SOFTWARE RADIO PERIPHERAL (USRP) This part is the main and actual operations that take place and background signal calibrations and modulation, demodulation, and randomizer to make the signal stronger while the signalcondition is too weak, Analog to digital conversion (ADC) process, Automatic Gain Control (AGC) to initiate the signal by adding a mixer and local oscillator to kick start thesignal to the initial stage of the working scenario. Moreover, the USRP board is designed only for Rf signal which means Radio frequency signals and Lora are only accessible in unlicensed bands like 433Mhz, 868Mhz, and so on.

4.3.1 GNU Radio Blocks and Modules:

The message Sink, the Audio sink, and so on likely here it is receiving part of the message or information region, and Every source module is like transmitting a signalto the channel. Every sink module is like receiving the end of that signal. And few small Application parts are taken by the python code (.py) or Cplusplus (.cpp) code format or code snippet which is taken part as a background process. Likewise, more blocks are developed through this process of background coding with this programming language and blocks can be customized for our needs. Everything will come under the custom block partfor the visualized purpose and for the user's comfort.

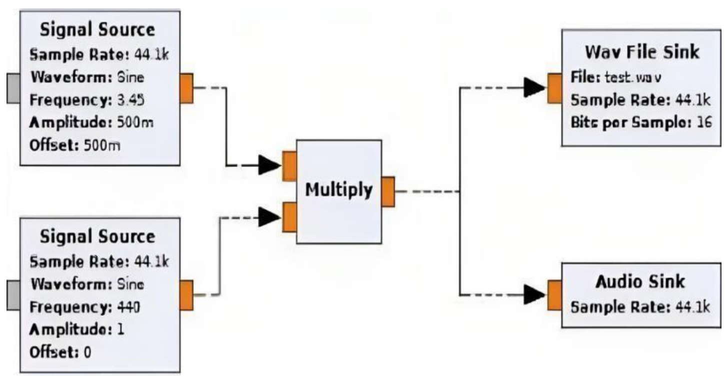

Figure 4.3. Source & Sink GNU block

And here we use a 2x2 MIMO antenna for better (2-Tx) Transmission and (2-Rx) Receiving part and after every future part the LoRa Module and GNU Radio the block willgenerate a GUI (Graphical User Interface) to visualize the signal in two domains X & Y Domain. Here, SDR is designed in the structural manner of a 2x2 MIMO antenna.

SDR kit which accepts the signal ranges till 6 Ghz so it is far reaching to communicate with satellite kind of signals. SDR waveguru kit works in every gadget that currently updates mobile phones and other electronic devices. A few lines for the usage ofthe SDR kit are Phones and getting updates in recent times while 2G, and 3G We just needto switch to keypad phones to Android phones for a better user experience and latency of signals so here, the SDR kit can track every kind of signal so this part is something unique to handle it and use to track, trace, and more operations like sensing signals.Generations-wise technique fields are updated day by day so from Starting button mobile phones to recent high-end touch mobile phones can accept the signals in the SDR waveguru kit (70 Mhz ~ 6 Ghz).

4.3.2 LoRa Transmitter GNU Block:

In GNU Radio Companion, the following block and following operation are running in the background and these are the mandatory blocks needed to get filtered signals for the LoRa Transmitter chain. These are the chains to be worked out and neededfor the accuracy of loRa Transmitter signals, below these are the blocks that need to be practically worked out.

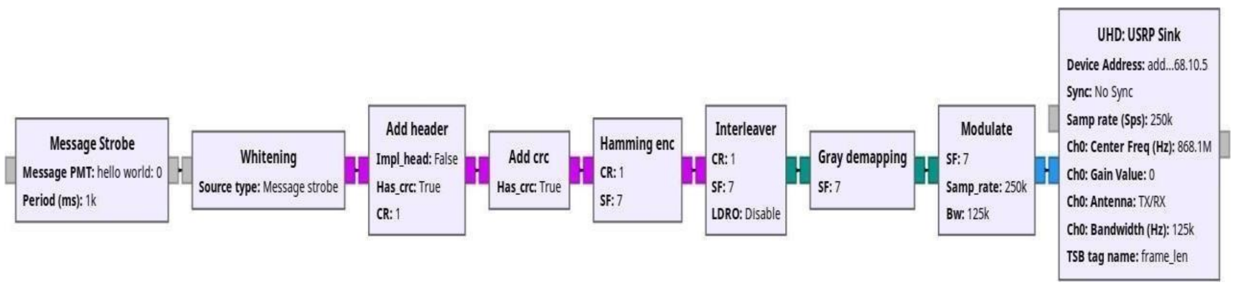

Figure 4.4. LoRa Transmitter Chain Block

- USRP Sink: This block receives the LoRa signal from the USRP device.
- Modulate: This block demodulates the LoRa signal using FFT demodulation.
- Whitening: This block whitens the demodulated signal.
- Message PMT: This block calculates the packet metadata for the demodulatedsignal.
- Add header: This block adds the header to the demodulated signal.
- Source type: This block specifies the source type of the demodulated signal.
- Impl head: This block specifies whether to use implicit header decoding.
- Has crc: This block specifies whether the demodulated signal has a CRC.
- CR: This block specifies the coding rate of the demodulated signal.
- Add crc: This block adds the CRC to the demodulated signal if it does not alreadyhave one.
- Hamming enc: This block encodes the demodulated signal using Hamming coding.
- SF: This block specifies the spreading factor of the demodulated signal.
- Interleaver: This block interleaves the encoded signal.
- LDRO: This block performs low-density parity-check (LDPC) decoding on the

demodulated signal.

- Gray demapping: This block demaps the demodulated signal from Gray code tobinary code.
- Sampling rate: This block specifies the sampling rate of the demodulated signal.
- Gain Value: This block specifies the gain of the demodulated signal.
- Bandwidth: This block specifies the bandwidth of the demodulated signal.
- Antenna: This block specifies the antenna to use to receive the LoRa signal.
- Print header: This block specifies whether to print the header of the demodulatedsignal to the console.
- Print rx msg: This block specifies whether to print the received message to theconsole.
- CRC verif: This block verifies the CRC of the demodulated signal.
- Output CRC check: This block specifies whether to check the output CRC of the demodulated signal.
- OS factor: This block specifies the decimation factor of the demodulated signal.

The following is a brief overview of the sequence of events that occur in the LoRa SDR receiver:

- The USRP Sink block receives the LoRa signal from the USRP device.
- The Modulate block demodulates the LoRa signal using FFT demodulation.
- The Whitening block whitens the demodulated signal.
- The Message PMT block calculates the packet metadata for the demodulated signal.
- The Add header block adds the header to the demodulated signal.
- The Hamming enc block encodes the demodulated signal using Hamming coding.
- The Inter leaver block interleaves the encoded signal.
- The LDRO block performs low-density parity-check (LDPC) decoding on the demodulated signal.
- The Gray demapping block demaps the demodulated signal from Gray code tobinary code.
- The CRC verif block verifies the CRC of the demodulated signal.
- The Print header block prints the header of the demodulated signal to the console if

enabled.

- The Print rx msg block prints the received message to the console if enabled.
- The Output CRC check block checks the output CRC of the demodulated signal if enabled.
- The os factor block decimates the demodulated signal by the specified factor.
- The LoRa SDR receiver can be used to receive and decode LoRa signals from a variety of devices, such as LoRa gateways, sensors, and actuators. It is a versatile tool for developing and testing LoRa applications.

4.3.3 LoRa Receiver GNU Block:

In GNU Radio Companion, the following block and following operation are running in the background and these are the mandatory blocks needed to get filtered signals for the LoRa Receiver chain. These are the chains to be worked out and needed for the accuracy of loRa Receiver signals, below these are the blocks that need to be practically worked out.

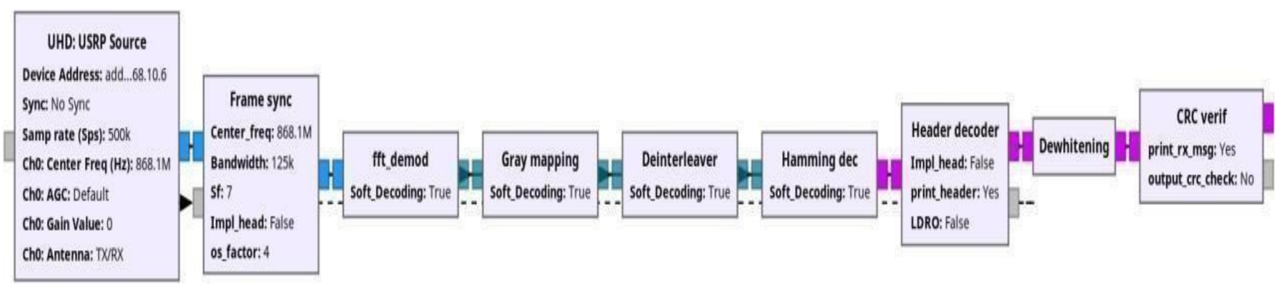

Figure 4.5. LoRa Receiver Chain Block

- The receiver synchronizes to the signal frame.
- The receiver demodulates the signal using FFT demodulation.
- The receiver decodes the signal using soft decoding.
- The receiver deinterleaver the decoded signal.
- The receiver decodes the signal using Hamming decoding.
- The receiver decodes the header of the signal.
- The receiver whitens the signal.

- The receiver prints the header of the signal.
- The receiver checks the CRC of the signal.
- The receiver prints the received message.
- The receiver checks the output CRC of the signal.
- The receiver decimates the signal by a factor of 4.

The "Impl head: False" blocks indicate that the receiver is not using implicit header decoding. The "os factor: 4" block indicates that the receiver is decimating the signal by a factorof 4. This reduces the data rate of the signal, but it also reduces the noise in the signal. The "print header: Yes" and "print_rx_msg: Yes" blocks indicate that the receiver willprint the header and receive a message to the console. The "CRC verification" block checks the CRC of the signal. If the CRC is not valid, the receiver will discard the signal. The "output crc check: No" block indicates that the receiver will not check the outputCRC of the signal.

CHAPTER 5

RESULTS & DISCUSSION

5.1 Design of Demodulation and Decoding in Result:

LoRa Transmitter and LoRa Receiver Blocks are explained with the important content and here are the below sample signals which undergoes this particular signals samples for more benefit in future signals predictionsand validate (or) cross check the output signals to get the fine and reliable output.

5.2 GNU Radio filter design:

Magnitude Response:

The Magnitude Response plot of a digital filter. The magnitude response of a filter shows how much the filter amplifies signals at different frequencies. In this case, the filterhas a passband from 20 kHz to 140 kHz, with a peak gain of 0 dB at 100 kHz. The filter also has a stopband above 160 kHz, where it amplifies signals by more than 100 dB [18]. The graph shows that the filter has a very sharp cutoff frequency between the passband and thestopband. This means that the filter can effectively separate signals in the passband from signals in the stopband.

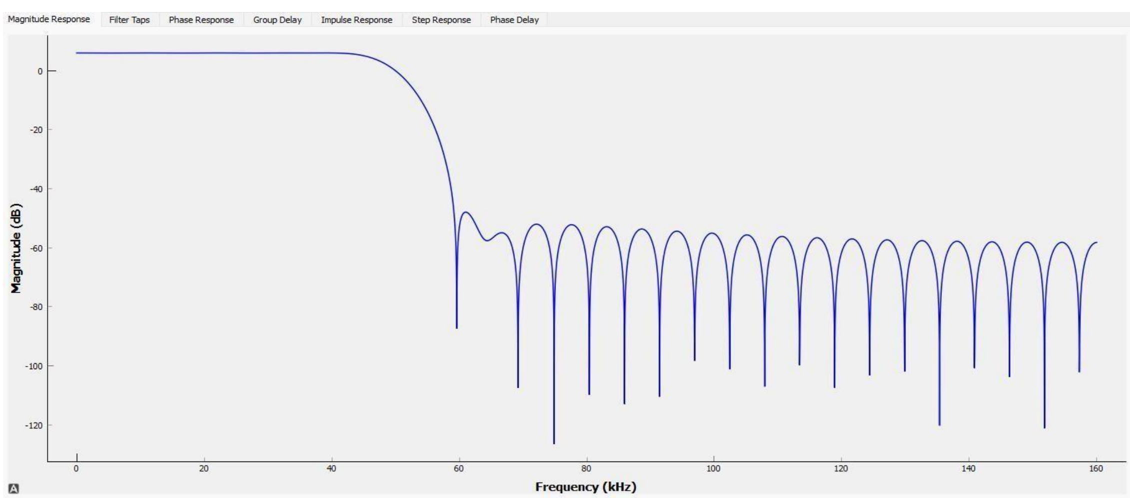

Figure 5.1 Magnitude Response

This type of filter is often used in signal processing applications such as audio filtering and image processing. Magnitude (dB): This is the y-axis of the graph. It shows the amplification of the filter at different frequencies. The values are in decibels (dB), where 0 dB is no amplification, and each increase of 3 dB corresponds to a doubling of the signal power. Frequency (kHz): This is the x-axis of the graph. It shows the frequency of the signal in kilohertz (kHz). Band Diagram: The band diagram is a shaded area that shows the passband and stopband of the filter. The passband is the frequency range where the filter amplifies signals, and the stopband is the frequency range where the filter amplifies signals. Pole- Zero Plot: The pole-zero plot is a graphical representation of the filter's transfer function. It shows the poles and zeros of the filter, which are complex numbers that determine the filter's frequency response.

Overall, the graph shows that the filter is very good at separating signals in thepassband from signals in the stopband. It has a sharp cutoff frequency and a high amplitude in the stopband. This type of filter is often used in signal processing applications such as audio filtering and image processing.

Filter Taps:

The filter taps in a digital filter are the coefficients that are multiplied by the input signal samples to produce the output signal. The number of filter taps determines the orderof the filter, and the values of the filter taps determine the filter's frequency response. Thefilter has 64 taps. This means that the filter's output signal is computed by multiplying theinput signal samples by 64 different coefficients and summing the results. The values of these coefficients are not shown in the image, but they can be determined using a variety of filter design methods. The filter taps are important because they control the filter's frequency response. The number of filter taps and the values of the filter taps determine the filter's passband, stopband, and transition band. The passband is the frequency range where the filter amplifies signals, the stopband is the frequency range where the filter amplifies signals, and the transition band is the frequency range between the passband and the stopband.

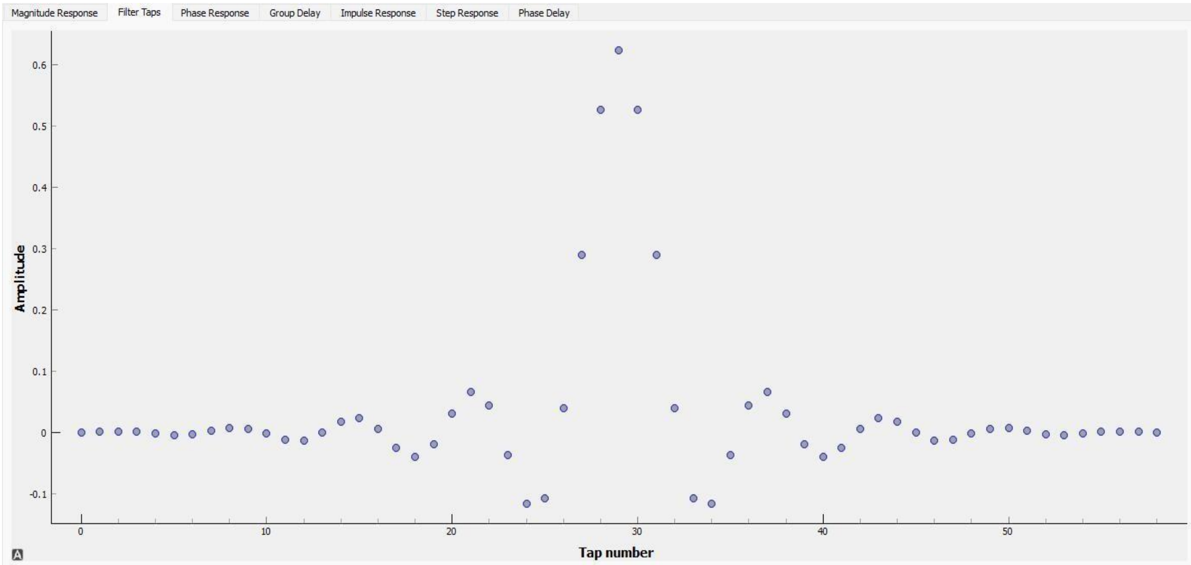

Figure 5.2 Filter Taps

For example, a low-pass filter has a passband at low frequencies and a stopband at high frequencies. The filter taps are chosen to ensure that the filter amplifies signals in the passband and amplifies signals in the stopband. The number of filter taps also affects the filter's sharpness. A filter with more taps will have a sharper cutoff frequency between the passband and the stopband. This means that the filter will be able to more effectively separate signals in the passband from signals in the stopband. Overall, the filter taps are an important part of digital filters. They control the filter'sfrequency response, sharpness, and computational complexity.

Phase Response:

The phase response of a digital filter is a plot of the phase shift of the filter's frequency response. The phase shift of the frequency response is a measure of how much the filter delays signals at different frequencies [6]. It is typically measured in degrees.The image shows the phase response of a digital filter with a passband from 10 kHz to 30kHz. The filter has a linear phase response in the passband, which means that there is no phase shift of the signal. This is important for some applications, such as audio filtering, where phase distortion can be undesirable. The phase response is also important for other applications, such as image processing, where group delay distortion can cause blurring. Group delay is the delay of different frequency components in a signal.

A linear phase response ensures that all frequency components in the passband are delayed by the same amount. This helps to prevent blurring in image processing applications. Here are some ofthe important features of a phase response plot. Linear phase response: A linear phase response means that there is no phase shift of the signal in the passband. This is importantfor some applications, such as audio filtering and image processing.

Group delay: The group delay is the delay of different frequency components in a signal. A linear phase response ensures that all frequency components in the passband are delayed by the same amount. This helps to prevent blurring in image processing applications. Phase ripple: Thephase ripple is the variation in the phase response within the passband. A low phase rippleis important for some applications, such as communication systems.

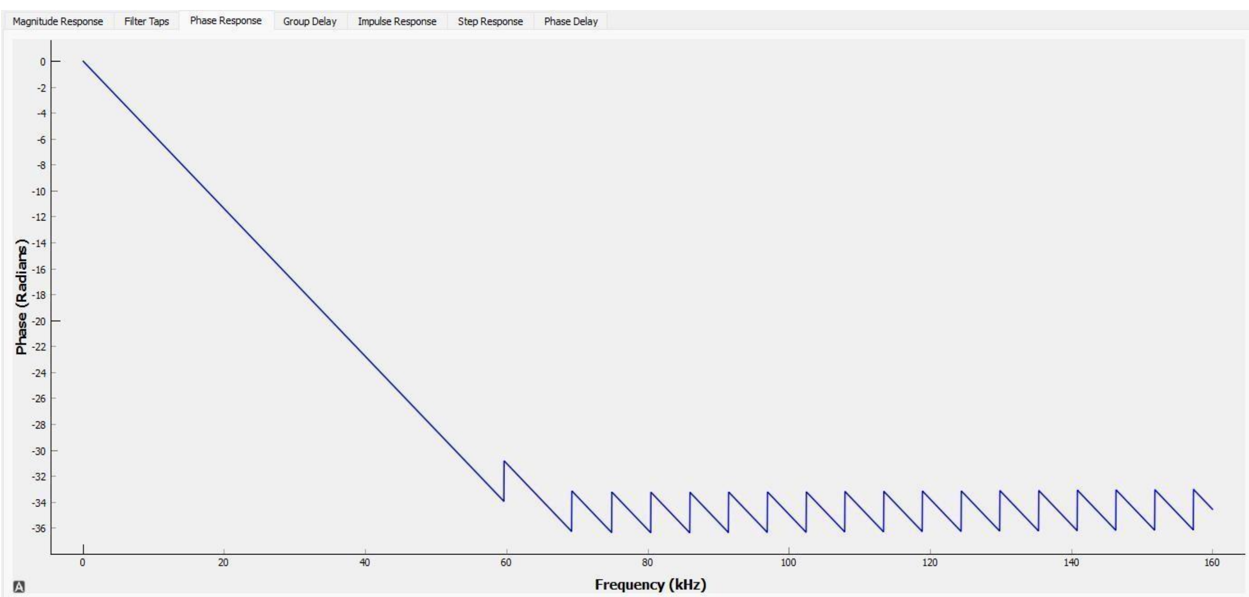

Figure 5.3 Phase Response

Phase response plots are an essential tool for understanding and designing digital filters. Here are some examples of how the phase response of a filter can be used in different applications. Audio filtering: A linear phase response is important for audio filtering because it prevents phase distortion. Phase distortion can cause undesirable effects, such as making audio sound muffled or harsh. Image processing: A linear phase response is alsoimportant for image processing because it prevents group delay distortion [9]. Group delay distortion can cause blurring in images. Communication systems: A low phase ripple is important for communication systems because it reduces intersymbol interference. Intersymbol interference is a type of distortion that can occur when multiple symbols are transmitted in close proximity.

Group Delay:

The group delay of a digital filter with a passband from 10 kHz to 30 kHz. The group delay is relatively constant in the passband, at about 1 sample. This means that the filter delays all frequency components in the passband by the same amount of time. This is important for some applications, such as image processing, where group delay distortioncan cause blurring. Group delay distortion occurs when different frequency components ina signal are delayed by different amounts. This can cause the edges of objects in an imageto become blurred. A linear phase response, which is what this filter has, ensures that all frequency components in the passband are delayed by the same amount.

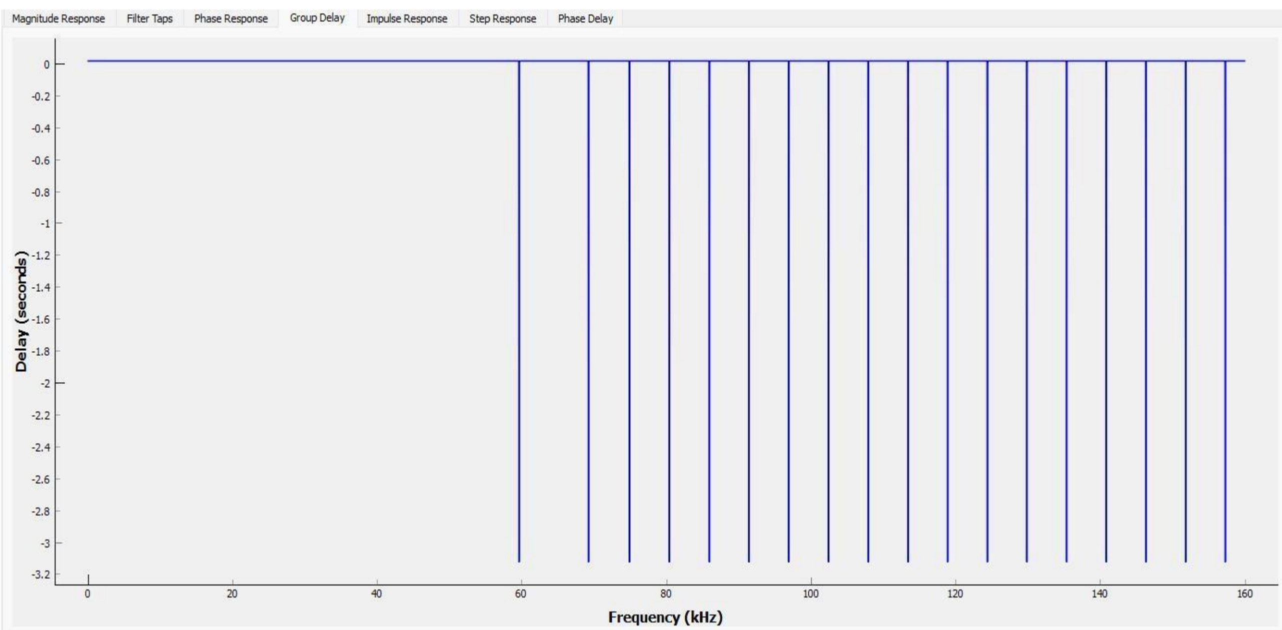

Figure 5.4 Group Delay

This helps to prevent blurring in image processing applications. Here is a more detailed explanation of the group delay plot: X-axis: The x-axis of the plot shows the frequency in normalized units. The normalized frequency is defined as the frequency divided by the sampling frequency. Y- axis: The y-axis of the plot shows the group delay in samples. The group delay is the difference in time between the outputs of the filter at two different frequencies.

Group delay: The group delay is shown as a blue line on the plot. The group delay is relatively constant in the passband, at about 1 sample. This means that the filter delays all frequencycomponents in the passband by the same amount of time. The passband is shaded in blue on the plot. The passband is the frequency range where the filter amplifies signals. The group delay is relatively constant in the passband, which is important for image processingapplications.

Overall, the group delay plot shows that the filter has a linear phase response in the passband. This is important for image processing applications, where group delay distortion can cause blurring.

Impulse Response:

The impulse response of a digital filter is the output of the filter when the input is a single impulse signal. The impulse response is a good way to characterize the filter's behavior because it shows how the filter responds to all possible input signals. The impulse response of the digital filter in the image you sent is shown as a blue line on the plot. The impulse response is finite in duration, which means that the filter output settles to zero after a finite amount of time. This is in contrast to infinite impulse response (IIR) filters, whose impulse response may continue indefinitely. The impulse response of a digital filter is related to the filter's frequency response by the following equation:

$$H(f) = DTFT\{h(n)\} \tag{5}$$

where:

H(f) is the filter's frequency response

h(n) is the filter's impulse response

DTFT is the discrete-time Fourier transform

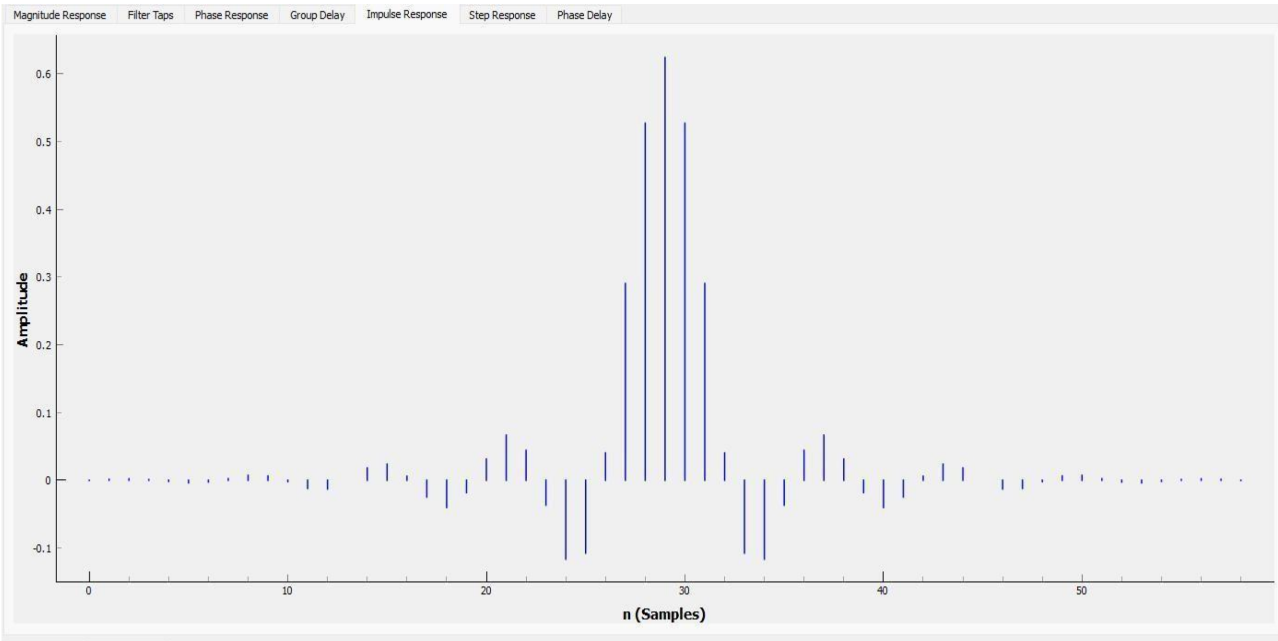

Figure 5.5 Impulse Response

This equation tells us that the frequency response of a digital filter is simply the Fourier transform of the filter's impulse response. The impulse response of a digital filter can be used to design and analyze filters. For example, a filter designer can use the impulse response to ensure that a filter has the desired frequency response.

TABLE 5.1: Tested LoRa Settings

Bandwidth	Hamming	Spreading (Symbols/ chirp)	Claimed Range	Measured bps (± 1)
125	4/5	7	medium	3,428
31.35	4/8	9	long	175
125	4/8	12	longer	104

Here are someof the important features of an impulse response plot: Finite duration: A finite impulse response (FIR) filter has an impulse response that is finite in duration. This means that thefilter output settles to zero after a finite amount of time. Length: The length of the impulseresponse is equal to the number of taps in the filter. The number of taps determines the order of the filter and the sharpness of the filter's cutoff frequency. Symmetry: The impulse response of a linear phase filter is symmetrical about its center point. This means that the impulse response is the same when flipped over its center point. Impulse response plots arean essential tool for understanding and designing digital filters. Here is an example of howthe impulse response of a filter can be used in a digital signal processing application

Matched filtering: A matched filter is a type of filter that is used to detect a knownsignal in a noisy environment. The impulse response of a matched filter is equal to the time-reversed version of the known signal. When a matched filter is applied to a signal, theoutput of the filter is maximized when the input signal contains the known signal.

Step Response:

The step response of a digital filter is the output of the filter when the input is aunit step signal. The step response is a good way to characterize the filter's behavior because it shows how the filter responds to a sudden change in the input signal. The step response of the digital filter in the image you sent is shown as the blue line on the plot.

The step response is finite in duration, which means that the filter output settles to a steady state value after a finite amount of time. This is in contrast to infinite impulse response (IIR) filters, whose step response may continue indefinitely.

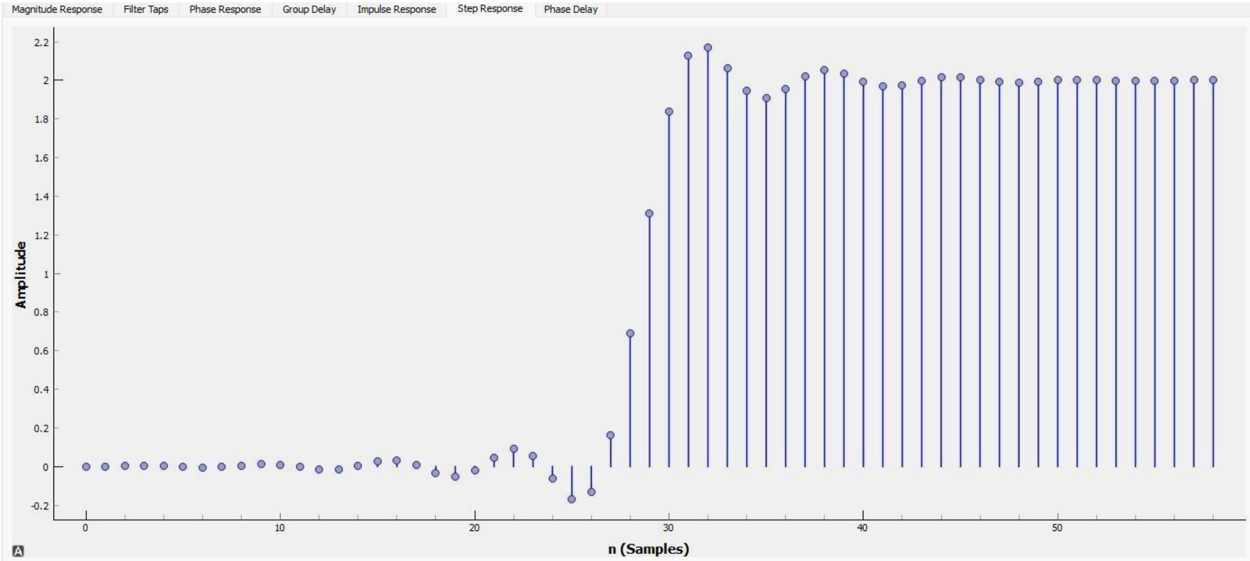

Figure 5.6 Step Response

The step response of the digital filter in the image you sent has the following features: Rise time: The rise time is the time it takes for the filter output to rise from 10% to 90% of its steady state value. The rise time of the step response is approximately 10 samples. Settling time: The settling time is the time it takes for the filter output to settle within 2% of its steady state value. The settling time of the step response is approximately 20 samples. Overshoot: The overshoot is the amount by which the filter output exceeds its steady state value before settling down. Theovershoot of the step response is approximately 5%. The step response of a digital filter can be used to design and analyze filters. For example, a filter designer can use the step response to ensure that a filter has the desired rise time, settling time, and overshoot characteristics. Here is an example of how the step response of a filter can be used in a digital signal processing application: Control systems: Control systems often use filters tocontrol the response of a system to a desired input signal. The step response of a filter canbe used to design a controller that will produce the desired output response.

Phase Delay:

The phase delay of a digital filter is the amount of time that each frequency component in the input signal is delayed by the filter. The phase delay is a function of frequency, and it can be calculated using the following equation: The phase delay of a filter is typically measured in seconds. A higher phase delay means that the filter is delaying the output signal more in the time domain. The phase delay of a filter can be compensated for using the Filter Delay block in GNU Radio. The Filter Delay block delays the input signal by the amount of phase delay introduced by the filter. This ensures that the output of the filter has a linear phase response, which means that all frequency components in the passband are delayed by the same amount. The image you sent shows the phase delay of a digital filter with a passband from 20 kHz to 140 kHz and a stopband above 160 kHz. The filter has a peak phase delay of about 0.2 seconds at 100 kHz. This means that the filter is delaying the output signal by up to 0.2 seconds at 100 kHz. The phase delay of this filter is relatively high, but it is still acceptable for many applications.

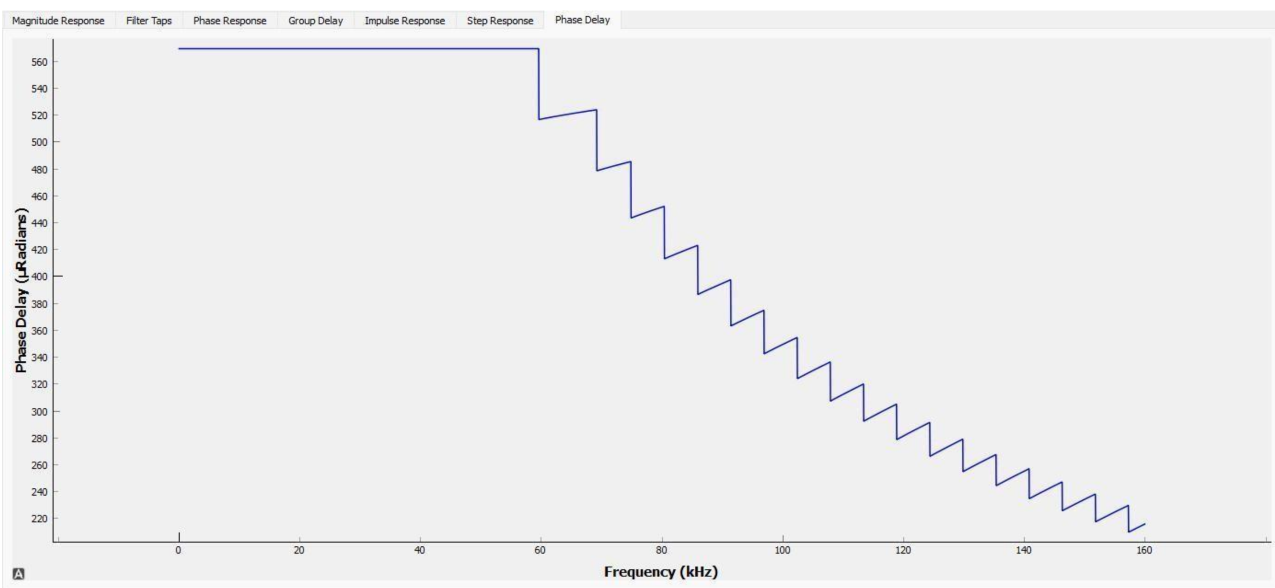

Figure 5.7 Phase Delay

For example, a phase delayof 0.2 seconds is not noticeable in audio applications because the human ear cannot detect delays that are less than about 10 milliseconds. However, a phase delay of 0.2 seconds would be noticeable in video applications, because the human eye can detect delays that are less than about 1 millisecond. Here are some of the important features of a phase delayplot:

Linear phase response: A linear phase response means that there is no phase shift of the signal in the passband. This is important for some applications, such as audio filtering and image processing. The group delay is the average of the phase delay across all frequencies in the passband. A constant group delay means that all frequency components in the passband are delayed by the same amount of time. This is important for image processing applications, where group delay distortion can cause blurring and phase ripple is the variation in the phase response in the passband. A low phase ripple is important for some applications, such as communication systems. Phase delay plots are an essential tool for understanding and designing digital filters. Phase delay plots are an essential tool for understanding and designing digital filters.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

In this paper, a LoRa signal monitoring system scheme and monitoring method in software radio platform is proposed. The LoRa signal is preprocessed and parameter information is extracted. Firstly, we have shown that the combination of the SDR WaveGuru Kit and GNU Radio is a powerful and flexible platform for signal processing and analysis. This platform not only provides the capability to receive and demodulate LoRa signals but also allows for real-time visualization and further signal manipulation, making it a valuable tool for researchers and engineers working with LoRa technology. Secondly, we have presented a detailed analysis of LoRa signal characteristics, including signal power, bandwidth, and modulation schemes. This analysis provides essential insights into the technical aspects of LoRa communication, which can aid in optimizing LoRa network deployments and improving overall performance. Furthermore, we have explored the impact of various parameters on LoRa signal detection, such as signal strength, interference, and signal-to-noise ratio (SNR). Understanding these factors is crucial for ensuring reliable and robust LoRa communication systems in real-world scenarios. The results obtained in this project can be beneficial for LoRa technology developers, network planners, and researchers. They provide insights into signal detection and analysis, enabling the optimization of LoRa networks for various applications, including IoT and remote monitoring.

6.2 FUTURE SCOPE

As future work, we will improve the design of monitoring GNU blocks to improve the accuracy of parameter estimation and reduce recognition errors of LoRa signal and more extensive field testing, the development of advanced signal processing techniques, and the integration of LoRa technology into emerging wireless networks.

REFERENCE

[1] Y. Yi, H. Zhao and Y. Wang, & quot;LoRa Signal Monitoring System of Multi-Node Software Defined Radio," 2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), Seoul, Korea (South), 2020, pp. 1-5, doi: 10.1109/WCNCW48565.2020.9124898.

[2] J. Tapparel, O. Afiliados, P. Mayoraz, A. Balatsoukas-Stimming and A. Burg, "An Open-Source LoRa Physical Layer Prototype on GNU Radio," 2020 IEEE 21stb International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Atlanta, GA,USA, 2020, pp. 1-5, doi:10.1109/SPAWC48557.2020.9154273.

[3] K. Hill, K. K. Gagneja and N. Singh, "LoRa PHY Range Tests and Software Decoding - Physical Layer Security," 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 2019, pp. 805-810, doi: 10.1109/SPIN.2019.8711682.

[4] M. A. Ben Temim, G. Ferré, B. Laporte-Fauret, D. Dallet, B. Minger and L. Fuché, & quot;An Enhanced Receiver to Decode Superposed LoRa-Like Signals," in IEEE Internet of Things Journal,vol.7, no.8, pp. 7419-7431, Aug. 2020, doi: 10.1109/JIOT.2020.2986164.

[5] Dakic, B. Al Homssi, A. Al-Hourani and M. Lech, "LoRa Signal Demodulation Using Deep Learning, a Time-Domain Approach," 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), Helsinki, Finland, 2021, pp. 1-6, doi: 10.1109/VTC2021- Spring51267.2021.9448711.

[6] Zhe Huang, Weidong Wang and Yinghai Zhang, "Design and implementation of cognitive radio hardware platform based on USRP," IET International Conference on Communication Technology and Application (ICCTA 2011), Beijing, 2011, pp. 160-

164, doi: 10.1049/cp.2011.0651.

[7] U. Raza, P. Kulkarni, and M. Sooriyabandara, “Low power wide area networks: An overview,” IEEE Communications Surveys & Tutorials, vol. 19, no. 2, pp. 855–873, 2017.

[8] J. Haxhibeqiri et al., “A survey of LoRaWAN for IoT: From technology to application,” Sensors, vol. 18, no. 11, p. 3995, 2018.

[9] A. Augustin et al., “A study of LoRa: Long range & low power networks for the Internet of Things,” Sensors, vol. 16, no. 9, p. 1466, 2016.

[10] D. Croce, M. Gucciardo, S. Mangione, G. Santaromita, and I. Tinnirello, “LoRa technology demystified: From link behavior to cell-level performance,” IEEE Transactions on Wireless Communications, vol. 19, no. 2, pp. 822–834, 2019.

[11] O. Afisiadis, M. Cotting, A. Burg, and A. Balatsoukas-Stimming, “On the error rate of the LoRa modulation with interference,” IEEE Transactions on Wireless Communications, vol. 19, no. 2, pp. 1292–1304, 2019.

[12] R. Fernandes, R. Oliveira, M. Lu ıs, and S. Sargento, “On the real capacity of LoRa networks:the impact of non-destructive communications,” IEEE Communications Letters, vol. 23, no. 12, pp. 2437–2441, 2019.

[13] M. A. B. Temim et al., “An enhanced receiver to decode superposed LoRa-like signals,” IEEEInternet of Things Journal, 2020.

[14] S. Tong, J. Wang, and Y. Liu, “Combating packet collisions using non-stationary signal scaling in LPWANs,” in Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services, 2020, pp. 234–246.

[15] B. Hu, Z. Yin, S. Wang, Z. Xu, and THe, “SCLoRa: Leveraging multidimensionality

in decoding collided LoRa transmissions,” in IEEE 28th International Conference on Network Protocols. IEEE, 2020, pp. 1–11.

[16] X. Xia, Y. Zheng, T. Gu et al., “FTrack: Parallel decoding for LoRa transmissions,” IEEE/ACM Transactions on Networking, 2020.

[17] M. Chiani and A. Elzanaty, “On the LoRa modulation for IoT: Waveform properties and spectral analysis,” IEEE Internet of Things Journal, vol. 6, no. 5, pp. 8463–8470, 2019.

[18] R. Ghanaatian, O. Afiliados, M. Cotting, and A. Burg, “LoRa digital receiver analysis and implementation,” in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech andSignal Processing (ICASSP). IEEE, 2019, pp. 1498–1502.

[19] C. Bernier, F. Dehmas, and N. Deparis, “Low complexity LoRa frame synchronization For ultra-low power software-defined radios,” IEEE Transactions on Communications, vol. 68, no. 5, pp. 3140–3152, 2020.

[20] M. Xhonneux, D. Bol, and J. Louveaux, “A low-complexity synchronization scheme for LoRaend nodes,” arXiv preprint 1912.11344, 2019.

[21] J. Tapparel, O. Afiliados, P. Mayoraz, A. Balatsoukas-Stimming, and A. Burg, “An open- source LoRa physical layer prototype on GNU Radio,” IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), 2020.

[22] O. Afisiadis et al., “On the advantage of coherent LoRa detection in the presence of interference,” arXiv preprint arXiv:2010.00507, 2020.

[23] Z. Xu et al., “FlipLoRa: Resolving collisions with up-down quasi orthogonality,” in 17th Annual IEEE International Conference on Sensing, Communication, and Networking. IEEE, 2020,pp. 1–9.

APPENDICES

INDIVIDUAL CONTRIBUTION

Project Title: LoRa Signal Detection using SDR Waveguru Kit.

Student Name: Tarun N

Register No: 201EC270

Correcting the overall malware errors & maintenance. Analysing of Antenna connections & its positioning. Connection Path between PC & SDR WAVEGURU KIT. Researching and understanding the LoRa signal modulation and demodulation process. Design and implement a signal processing algorithm to detect LoRa signals in the SDR WaveGuru data stream. Develop a graphical user interface (GUI) to display the detected LoRa signals and provide user controls.

(TARUN N)

INDIVIDUAL CONTRIBUTION

Project Title: LoRa Signal Detection using SDR Waveguru Kit.

Student Name: Yashwanthkumar R Register No: 201EC293

Set up the SDR WaveGuru hardware. Collect and pre-process the raw SDR data stream. Integrate the signal processing block into the SDR WaveGuru software. Test and refine the system to ensure that it can reliably detect LoRa signals. Optimize SDR settings to maximize signal reception and minimize interference. GNU custom block creator.

R.yashwanthkumar.

(YASHWANTHKUMAR R)

INDIVIDUAL CONTRIBUTION

Project Title: LoRa Signal Detection using SDR Waveguru Kit.
Student Name: Sudharsan R Register No: 211EC520

OS handler and Maintaining the system vulnerability. LORA work plan. Design a graphical user interface (GUI) to control and monitor the LoRa signal detection system. Develop visualization tools to display received LoRa signals and their relevant parameters (e.g., signal strength, frequency, modulation parameters). Implement data logging and storage for signal analysis and historical data tracking. Ensure that the user interface provides clear feedback and options for the end user.

(SUDHARSAN R)

Similarity Report

PAPER NAME

final Project report SDR.pdf

AUTHOR

tarun

WORD COUNT

11414 Words

CHARACTER COUNT

60447 Characters

PAGE COUNT

54 Pages

FILE SIZE

1.2MB

SUBMISSION DATE

Oct 14, 2023 4:28 PM GMT+5:30

REPORT DATE

Oct 14, 2023 4:29 PM GMT+5:30

● 23% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 15% Internet database
- Crossref database
- 15% Submitted Works database
- 10% Publications database
- Crossref Posted Content database

● Excluded from Similarity Report

- Bibliographic material
- Cited material
- Quoted material
- Small Matches (Less then 8 words)

Summary

16/10/2023

V. BARANIDHARAN, B.E.,M.Tech.,
Assistant Professor
Dept. of Electronics & Communication Engg.,
Bannari Amman Institute of Technology,
Sathyamangalam - 638 401, Erode Dist.

SIMATS ENGINEERING

Approved By AICTE | IET-UK Accreditation

INTERNATIONAL CONFERENCE ON CURRENT TRENDS IN MODERNIZED COMPUTING AND COMMUNICATION TECHNOLOGIES (MCCT) 2023

CERTIFICATE

This is to certify that Mr / Ms **TARUN N**

has presented a paper titled as

"DESIGN AND IMPLEMENTATION OF LORA SIGNAL DETECTION USING SDR AND GNU RADIO COMPANION"

from BANNARI AMMAN INSTITUTE OF TECHNOLOGY

in the International Conference on Current Trends in Modernized Computing and Communication Technologies 2023 (MCCT- 2023) held on 11th October 2023 at Department of Augmented Reality & Virtual Reality, Institute of Computer Science and Engineering, SIMATS Engineering, SIMATS, Chennai - 602105.

Convenor
Dr. M. Gunasekaran

Principal
Dr. B. Ramesh

SIMATS ENGINEERING

ENGINEER TO EXCEL

Approved By AICTE | IET-UK Accreditation

INTERNATIONAL CONFERENCE ON CURRENT TRENDS IN MODERNIZED COMPUTING AND COMMUNICATION TECHNOLOGIES (MCCT) 2023

CERTIFICATE

This is to certify that Mr / Ms **YASHWANTHKUMAR R**

has presented a paper titled as

"DESIGN AND IMPLEMENTATION OF LORA SIGNAL DETECTION USING SDR AND GNU RADIO COMPANION"

from BANNARI AMMAN INSTITUTE OF TECHNOLOGY

in the International Conference on Current Trends in Modernized Computing and Communication Technologies 2023 (MCCT- 2023) held on 11th October 2023 at Department of Augmented Reality & Virtual Reality, Institute of Computer Science and Engineering, SIMATS Engineering, SIMATS, Chennai - 602105.

Convenor
Dr. M. Gunasekaran

Principal
Dr. B. Ramesh

SIMATS ENGINEERING

Approved By AICTE | IET-UK Accreditation

INTERNATIONAL CONFERENCE ON CURRENT TRENDS IN MODERNIZED COMPUTING AND COMMUNICATION TECHNOLOGIES (MCCT) 2023

CERTIFICATE

This is to certify that Mr / Ms **SUDHARSAN R**

has presented a paper titled as

"DESIGN AND IMPLEMENTATION OF LORA SIGNAL DETECTION USING SDR AND GNU RADIO COMPANION"

from BANNER AMMAN INSTITUTE OF TECHNOLOGY

in the International Conference on Current Trends in Modernized Computing and Communication Technologies 2023 (MCCT- 2023) held on 11th October 2023 at Department of Augmented Reality & Virtual Reality, Institute of Computer Science and Engineering, SIMATS Engineering, SIMATS, Chennai - 602105.

Convenor
Dr. M. Gunasekaran

Principal
Dr. B. Ramesh