Channel Estimation with Interference in OFDM Modulation using GNU Radio

Dr. RAMESHA M
Department of Electronics and Communication Engineering
Gitam Deemed to be University
Bengaluru, India
rmuniyap@gitam.edu

HARIPRIYA RAO M
Department of Electronics and Communication Engieering
Gitam Deemed to be University
Bengaluru, India
hmadhusu@gitam.in

HAMSINI REDDY K S
Department of Electronics and Communication Engineering
Gitam Deemed to be University
Bengaluru, India
hkusredd@gitam.in

KIRAN P S
Department of Electronics and Communication Engineering
Gitam Deemed to be University
Bengaluru, India
kpapishe@gitam.in

Abstract

Orthogonal Frequency Division Multiplexing (OFDM) is a widely used modulation technique in modern communication systems due to its ability to mitigate inter-symbol interference (ISI) and multipath fading. This study focuses on the practical implementation of OFDM using GNU Radio. One software-defined radio (SDR) platform. By leveraging open-source software and readily available hardware, the work aims to provide a flexible and accessible platform for real-world experimentation with OFDM-based digital communication.

The implementation involves designing and configuring OFDM transmitter and receiver blocks within the GNU Radio environment The system is tested under various channel conditions, including Additive White Gaussian Noise (AWGN), Rayleigh (Non-Line-of-Sight, NLOS), and Rician (Line-of-Sight, LOS) models. Key aspects such as synchronization, channel estimation, and error correction are addressed to ensure reliable data transmission.

The work not only serves as an educational resource for understanding OFDM principles and GNU Radio usage but also provides a practical platform for experimenting with SDR technology in real-world communication scenarios. OFDM (Orthogonal Frequency Division Multiplexing) signal transmission has been successfully implemented to accommodate three distinct types of input data: textual, audio, and image inputs and their corresponding waveforms and outputs have been obtained.

Keywords— OFDM, Software-Defined Radio (SDR) Channel estimation, GNU

1. Multiplexing Introduction

Wireless communication is one of the most exciting areas in the communication field today. It is used to transmit information over the air without the use of a guided medium. Due to the sharply increasing in demand for wireless connectivity, it has developed extremely fast during the last two decades. However, wireless communication is not as reliable as guided medium communication, due to fading and other propagation effects. Accordingly, the techniques to improve its capacity and reliability become the essential objectives for current research. Numerous techniques have been proposed to enhance the performance and reduce the interference in wireless communication. A common technique is Orthogonal Frequency Division Multiplexing (OFDM)

2. Orthogonal Frequency Division Multiplexing

OFDM technique divides a hige rate encoded data into sub streams which are parallel to each other. Orthogonal carriers are used to modulate these parallel sub streams, which transmits at different frequencies simultaneously in parallel. Modulation schemes used can be PSK, BPSK, QAM, 16QAM, QPSK. Pilot subcarriers are used to prevent frequency and phase shift errors. Since the subcarriers in OFDM systems are precisely orthogonal to one another, they overlap without interfering and also maximum spectral efficiency is attained without causing adjacent channel interference OFDM is a part of WLAN, DVB and BWA standards and is a strong candidate for some of the 4G wireless technologies. The popularity of OFDM is due to the use of IFFT/FFT which has efficient implementations. Multipath channels result in loss of orthogonality in carriers. Cyclic prefix can be used to restore the orthogonality back by converting a linear convolution channel into a circular convolution channel and also allow the signal to be decoded even if the packet is detected after some delay.

OFDM efficiently deals with multipath fading, channel delay spread, enhancement of channel capacity, modification of modulation density and robustness of narrowband interference. It is sensitive to small carrier frequency offsets, high frequency phase noise and sampling clock offsets. OFDM carrier is generated with all the modulated subcarriers by Inverse Fast Fourier Transform (IFFT) at the transmitter end and the same carrier is demodulated by Fast Fourier Transform (FFT) at the receiver end.

Our approach:

Understanding OFDM: Delve into the principles and advantages of OFDM modulation, exploring its application in modern communication systems.

GNU Radio Setup: Configure the GNU Radio environment, harnessing its vast array of signal processing blocks to design and simulate OFDM modulation and demodulation algorithms.

Implementation: Develop GNU Radio flowgraphs to generate OFDM signals, transmit them through the hardware platforms, and receive and demodulate them for analysis

3. Channel Estimation

Channel estimation refers to the process of estimating the characteristics of the wireless channel between the transmitter and receiver, such as signal strength, delay

OFDM communication system consists of channel model through which data symbols are transmitted to the receiver. This channel model produces line of sight communication and also various reflections due to which multipath effect come to picture. To minimize the multipath effect and noise introduced by the channel, we go for channel estimation. Generally, the received signal obtained through a frequency selective multipath fading channel contains the channel impulse response and additive white Gaussian noise which is given by,

In an OFDM system, the received signal can be represented as:

$Y(k)=H(k)X(k) + W(k)$

Where as

- $Y(K)$ is the received signal,
- $H(k)$ is the channel response,
- $X(k)$ is the transmitted signal,
- $W(k)$ is the additive noise.

To estimate the channel response $H(k)$, pilot subcarriers are used in two common arrangements:

1. **Block-type pilots:** Suitable for slow-fading channels where pilots are sent periodically.
2. **Comb-type pilots:** Used in fast-fading channels with pilot signals embedded in every OFDM symbol.

In our implementation, Least Square (LS) and Minimum Mean Square Error (MMSE) methods were used for channel estimation. The LS approach provides a direct solution, whereas MMSE offers improved accuracy by incorporating statistical information.

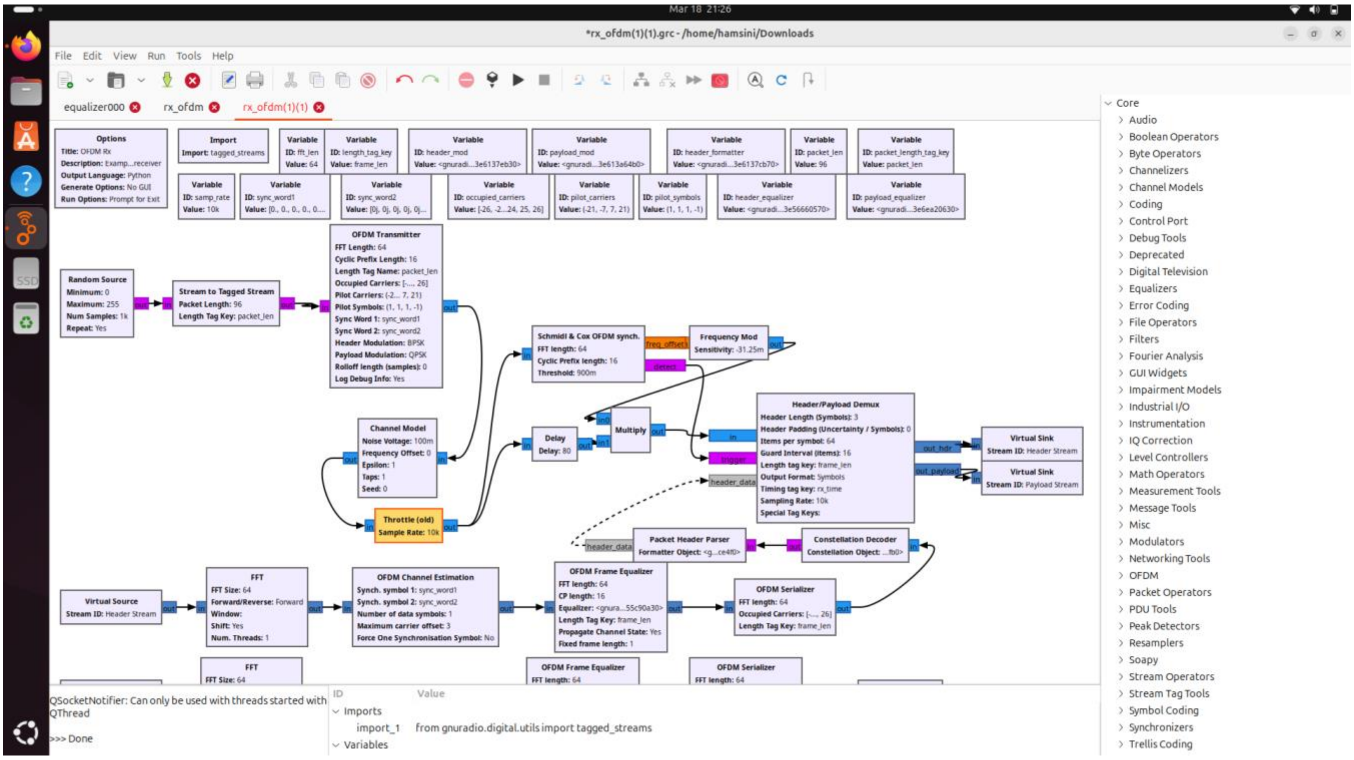
4. Synchronization and Channel Equalization

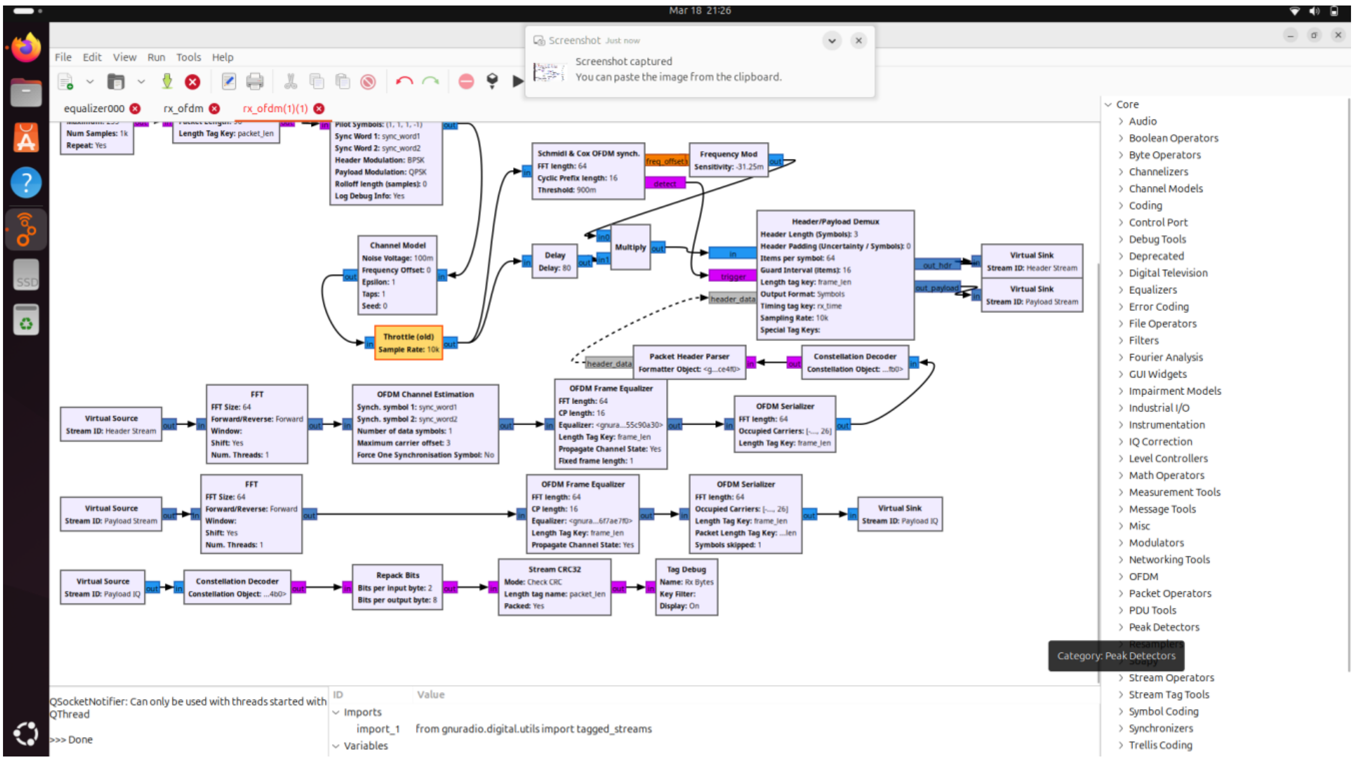
In OFDM communication systems, at the transmitter digital to analog conversion and at the receiver, analog to digital conversion is carried out. DAC and ADC never have exactly the same sampling period. Due to this, intercarrier interference and the slow shift of the symbol timing point occurs and so orthogonality is lost. This results in need of Synchronization and Channel Equalization. Using suitable cyclic prefix and considering the guard time of each symbol is identical, both time and frequency synchronization is exploited. Correlation between guard time and delay time is made and several peaks are produced due to different symbols and peak amplitudes. Due to the above correlation, many side lobes are introduced. When correlation is performed over a very large number of samples, the ratio of side lobes to peak amplitude will go to zero. This is due to independent samples proportional to subcarriers. Since the receiver needs to consider the source of each symbol correctly, synchronization plays a very important function on the receiver side. With the help of the incoming signals calculations are made to find the parameters. In order to set all these parameters, redundancy often brought up to as pilots or preamble depending on its place in the whole transmitted signal

Reliable estimation of the channel in time and frequency can be ensured using a cyclic pilot pattern. For initial training, preambles of symbol are used and then the inserted pilots with in the data symbol can be made used for tracking the remaining offsets. Fig. 1. shows the channel equalization model in which equalization is done as an adaptive system with relatively simple structure. Comparing with the features of frequently used algorithms like Minimum Mean-Square Error (MMSE) and Least Mean Squares (LMS), adaptive algorithm has three unique features: robustness in fixed point implementation, high adaptation rate and low computational complexity. The adaptive equalizer works in two modes namely, training mode and decision direction fashion. The difference between the original signal and the estimate available through the pilot signal is the estimation error. The difference between the estimate and the detected symbols are determined in decision direction fashion

5. GNU Radio Implementation

GNU Radio is an open-source software-defined radio (SDR) platform that enables the design, simulation, and testing of communication systems using modular signal processing blocks. It provides a flexible environment for implementing Orthogonal Frequency Division Multiplexing (OFDM) by leveraging both Python scripting and a graphical interface, GNU Radio Companion (GRC). This section delves into the theoretical underpinnings of our OFDM system implementation, detailing the signal flow, key algorithms, and practical considerations for transmitter design, receiver processing, channel simulation, and interference mitigation.





5.1 Theoretical Basis of OFDM in GNU Radio

OFDM divides a high-rate data stream into multiple low-rate substreams, each modulated onto orthogonal subcarriers. The orthogonality condition ensures that subcarriers, spaced by delta f=1/Ts .(where Ts is symbol duration) , do not interfere despite spectral overlap. This is mathematically realized using the Inverse Fast Fourier Transform (IFFT) at the transmitter and the Fast Fourier Transform (FFT) at the receiver. GNU Radio's modular architecture implements this process through a series of interconnected blocks, allowing precise control over subcarrier allocation, modulation, and timing.

The transmitted OFDM signal in the time domain can be expressed as:

$$x(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X(k) e^{j2\pi kn/N}$$

where (N) is the FFT size, (X(k)) is the modulated symbol on the (k)-th subcarrier, and (n) is the time index. The cyclic prefix (CP), a copy of the last portion of each symbol appended to its beginning, extends the symbol duration to Ts+Tcp, mitigating inter-symbol interference (ISI) by absorbing multipath delays up to Tcp

5.2 Transmitter Design

The transmitter's role is to convert input data into a time-domain OFDM signal suitable for transmission. The theoretical

Flow involves several stage:

1. Data Mapping and Modulation:

Input data (e.g., bits from text, audio, or images) is mapped to complex symbols using a chosen modulation scheme (e.g.,QPSK, where each symbol represents 2 bits). The constellation points are defined as X(k)=a+jb , where (a) and (b) depend on the modulation type. In GNU Radio, this is implemented using a digital modulation block, ensuring Gray coding to minimize bit errors during demodulation.

2.Subcarrier Allocation:

The modulated symbols are assigned to Nocc occupied subcarriers (a subset of the total (N) subcarriers), with the remainder reserved for guard bands or pilots. Pilot subcarriers, inserted at fixed intervals (e.g., every 8th subcarrier), carry known symbols to aid channel estimation at the receiver. This allocation adheres to the frequency-domain representation (X(k)), where (k) indexes the subcarriers.

3.IFFT

The IFFT transforms the frequency-domain symbols into a time-domain signal. The orthogonality of subcarriers is preserved because the IFFT ensures that each subcarrier's frequency is an integer multiple of delta f . The resulting signal (x(n)) is a superposition of modulated sinusoids, efficiently computed using the Cooley-Tukey algorithm in GNU Radio's FFT block.

4.Cyclic Prefix:

The CP is appended to combat ISI and maintain orthogonality in multipath channels. Theoretically, the CP converts the linear convolution of the channel impulse response (h(n)) with (x(n)) into a circular convolution, simplifying equalization at the receiver

5.Serialization:

The parallel time-domain samples are serialized into a continuous stream for transmission, aligning with SDR hardware or simulation requirements. This step ensures compatibility with downstream processing or physical layer interfaces.

5.3 Receiver Design

The receiver reverses the transmitter's operations while addressing channel-induced impairments. The received signal

Modeled as :

$$y(n)=x(n)*h(n)+w(n)$$

where * denotes convolution, (h(n)) is the channel impulse response, and (w(n)) is additive white Gaussian noise (AWGN). The receiver's theoretical tasks include:

1.Synchronization:

Timing and frequency synchronization are critical due to sampling mismatches between the transmitter's digital-to-analog converter (DAC) and the receiver's analog-to-digital converter (ADC). The CP enables timing synchronization by correlating the received signal with its delayed version, exploiting the CP's periodicity. Frequency offsets, which disrupt orthogonality and cause intercarrier interference (ICI), are estimated and corrected using pilot-based techniques or preamble correlation. GNU Radio's synchronization blocks use algorithms like Schmidl-Cox to detect symbol boundaries and carrier offsets.

2.CP Removal and FFT:

After synchronization, the CP is discarded, and the remaining (N) samples undergo an FFT to recover the frequency-domain subcarriers:

$$Y(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} y(n) e^{-j2\pi kn/N}$$

In an ideal channel, Y(k)=X(k), but in practice, Y(k)=H(k)·X(k)+W(k) where (H(k)) is the channel frequency response and (W(k)) is the noise in the frequency domain.\

3.Channel Estimation:

The channel response (H(k)) is estimated using pilot subcarriers. For Least Squares (LS) estimation, the channel is approximated as:

$$\hat{H}_{\text{LS}}(k) = \frac{Y(k)}{X_{\text{pilot}}(k)}$$

4.Equlization:

The estimated H^(k) used to equalize the received symbols:

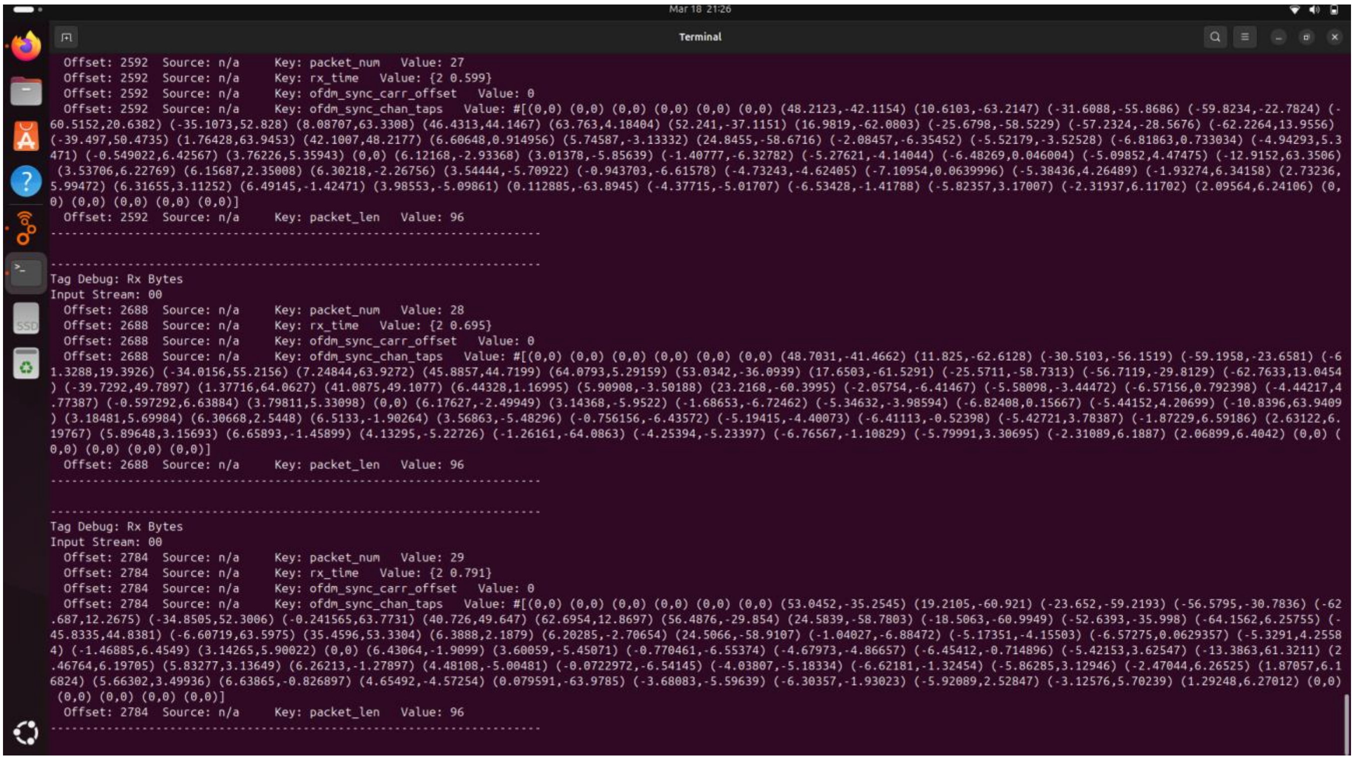
$$\hat{X}(k) = \frac{Y(k)}{\hat{H}(k)}$$

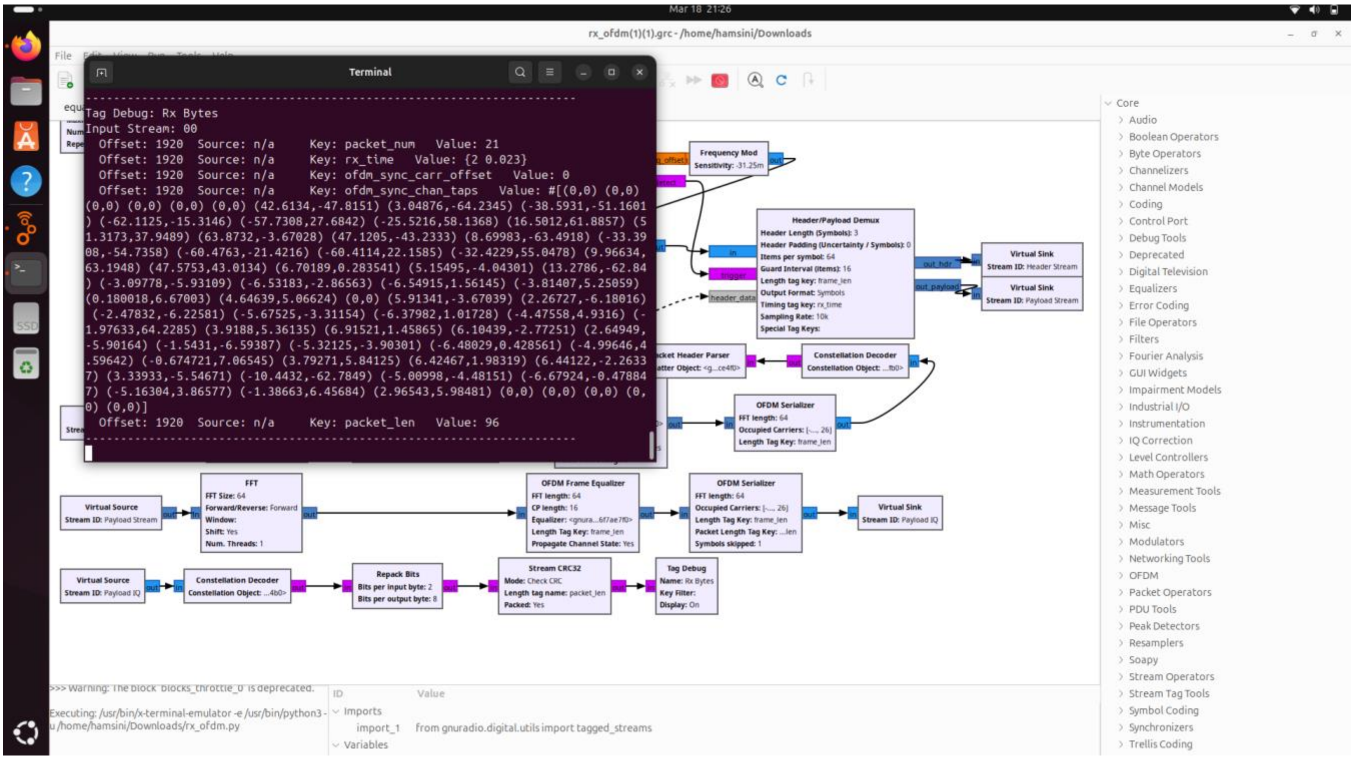
or a more robust MMSE-based approach incorporating noise statistics. Adaptive equalization, implemented in GNU Radio, iteratively refines H^(k) using training pilots and decision-directed feedback, enhancing performance in time-varying channels.

5.Demodulation and Data Recovery :

Equalized symbols X^(k) are demodulated to bits using the inverse of the transmitter's modulation scheme (e.g., QPSK). The parallel bit streams are then serialized to reconstruct the original data, with error metrics like bit error rate (BER) calculated to assess performance.

7.Result





Breaking Down the Output

1. Packet Number & Offset

Each received packet is identified by a packet number and an offset in the data stream.

Example: Offset: 2592 Source: n/a Key: packet_num Value: 27

- Here, packet number 27 is received at offset 2592.
- The offset refers to the position of this packet in the received data stream.

2. Reception Time (rx_time)

Key: rx_time Value: (2 0.599)

- This indicates the packet was received at time 2.599 seconds.
- The first number (2) represents integer seconds, and the second (0.599) represents the fractional part.

3. OFDM Synchronization Data

a) Carrier Frequency Offset (ofdm_sync_carr_offset)

Key: ofdm_sync_carr_offset Value: 0

- This shows the frequency offset correction applied during synchronization.
- A value of 0 suggests there was no major frequency shift, or it was already corrected.

b) Channel Estimation Taps (ofdm_sync_chan_taps)

Key: ofdm_sync_chan_taps Value: #((0,0) (0,0) (0,0) ... (48.2123, -42.1154) ...)

- This contains channel tap values, which represent how the transmitted signal has been affected by the wireless medium.
- Each pair (real, imaginary) is a complex number showing the estimated channel gain for different subcarriers.
- Large values indicate strong multipath effects or signal distortions.

4. Packet Length

Offset: 2592 Source: n/a Key: packet_len Value: 96

- Each packet has a length of 96 bytes.
- This means the receiver has successfully received 96 bytes of data in this packet.

6.Conclusion

In this paper, we designed and implemented an OFDM-based communication system featuring channel modeling, channel estimation, and Eualization using GNU Radio, an open-source platform. Through simulation results, we observed and compared the Bit Error Rate (BER) of the OFDM system with and without channel estimation. The results clearly demonstrate that the BER is significantly lower when channel estimation is applied, highlighting the importance of this technique.

- This output logs the reception and synchronization details of an OFDM signal.
- The rx_time records the arrival time, while ofdm_sync_,carr_offset and ofdm_sync, chan_taps provide insights into frequency correction and channel effects.
- The system is correctly processing OFDM packets and applying channel estimation for better data recovery.

This analysis is useful for understanding how OFDM-based wireless systems handle real-world distortions and synchronize data accurately