PRESIDENCY UNIVERSITY
Private University Estd. in Karnataka State by Act No. 41 of 2013
Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

School of Engineering

Department of Electronics and Communication Engineering

Design and Simulation of a 5G Digital Communication System Using Verilog

By

Rethika Sai R	20221ECE0152
Viresh Mallikarjun Tarapur	20221ECE0159
Aneeqha Tarannum S	20221ECE0169
Sudhanva R Avaghan	20231ECE3028

Under the guidance of

Dr. Imtiyaz Ahmed B K

Associate Professor, ECE

PRESIDENCY UNIVERSITY, BENGALURU
DECEMBER 2025

Department of Electronics and Communication Engineering

BONAFIDE CERTIFICATE

Certified that this project titled "**Design and Simulation of a 5G Digital Communication System Using Verilog**" is a bonafide work of "Rethika Sai R(20221ECE0152), Viresh Mallikarjun Tarapur(20221ECE0159), Aneeqha Tarannum S(20221ECE0169), Sudhanva R Avaghan (20231ECE3028)", who have successfully carried out the project work and submitted the report for partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** in **Electronics and Communication Engineering** during 2025-26.

Dr.Imtiyaz Ahmed B K **Associate Professor** **SOE, ECE** Presidency University	**Dr. Rajiv Ranjan Singh** **HOD** **SOE, ECE** Presidency University	**Dr.Shrishail B Anadinni** **Associate Dean** School of Engineering Presidency University	**Dr. Abdul Sharief** **Dean** School of Engineering Presidency University

Examiners

Sl. no.	Name	Signature	Date
1			
2			

DECLARATION

We the students of final year B.Tech in Electronics and Communication Engineering at Presidency University, Bengaluru, named Rethika Sai R, Viresh Mallikarjun Tarapur, Aneeqha Tarannum S , Sudhanva R Avaghan, hereby declare that the project titled **"Design and Simulation of a 5G Digital Communication System Using Verilog"** has been independently carried out by us and submitted in partial fulfillment for the award of the degree of Bachelor of Technology in Electronics and Communication Engineering during the academic year of 2025-26. Further, the matter embodied in the project has not been submitted previously by anybody for the award of any Degree or Diploma to any other institution.

RETHIKA SAI R	USN: 20221ECE0152	Signature:
VIRESH MALLIKARJUN TARAPUR	USN: 20221ECE0159	Signature:
ANEEQHA TARANNUM S	USN: 20221ECE0169	Signature:
SUDHANVA R AVAGHAN	USN: 20231ECE3028	Signature:

PLACE: BENGALURU

DATE: 05-12-2025

ACKNOWLEDGEMENT

For completion of this project work, We/I have received the support and the guidance from many people whom I would like to mention with deep sense of gratitude and indebtedness. We extend our gratitude to our beloved **Chancellor, Vice Chancellor, Pro-Vice Chancellor, and Registrar** for their support and encouragement in completion of the project.

I would like to sincerely thank my internal guide **Dr. Imtiyaz Ahmed B K, Associate Professor** Department of Electronics and Communication Engineering, Presidency University, for his moral support, motivation, timely guidance and encouragement provided to us during the period of our project work.

I am also thankful to **Dr. Rajiv Ranjan Singh, Professor, Head of the Department of Electronics and Communication Engineering**, Presidency University, for his mentorship and encouragement.

We express our cordial thanks to **Dr. Abdul Sharief**, Dean, **Dr. Shrishail B Anadinni**, Associate Dean (Core Branches), School of Engineering and the Management of Presidency University for providing the required facilities and intellectually stimulating environment that aided in the completion of my project work.

We are grateful to **Ms. Aruna M**, Project Coordinators, Department of Electronics and Communication Engineering, for facilitating research activities and timely assessments.

We acknowledge **Dr. Anilloy Frank,** for the comprehensive report template, which has been instrumental in enhancing our report's quality.

We are also grateful to Teaching and Non-Teaching staff of Department of Electronics and Communication Engineering and also staff from other departments who have extended their valuable help and cooperation.

Aneeqha Tarannum S

Rethika Sai R

Viresh Mallikarjun Tarapur

Sudhanva R Avaghan

Abstract

Project is based on the design and simulation of a 5G communication system using Verilog. The purpose of the work is not to develop a full 5G physical layer but to learn and recreate all the key digital blocks that constitute the foundation of the modern wireless communication. The system comprises a LDPC encoder and decoder, a QAM modulator and demodulator, an OFDM modulator and demodulator, and simple channel effects . Such blocks were made into individual modules of Verilog and then combined in a transmitter receiver chain. At the transmitter, the project will start with a simple input of data and then there will be LDPC encoding, which will be used to introduce redundancy to allow error correction. The bits are coded ,clustered and then mapped into QAM symbols which are then transmitted to the OFDM modulator and the IFFT transforms the frequency-domain symbols into time-domain samples. The OFDM demodulator on the receiver side works opposite to OFDM modulator with the help of FFT. The frequency-domain symbols that have been recovered are demapped and lastly the LDPC decoder assembles the original message. The waveform simulations were used to test all modules in Vivado and further signal analysis like constellation diagrams and CFO correction was performed in Python. Each block and its correct behaviour are verified by simulation. Details of the demodulation and modulation waveforms in OFDM indicate proper time domain expansion and their recovery. The desired values of constellation are reproduced through QAM modulation and demodulation. LDPC offers a good bit recovery, as indicated by identical encoder-decoder results. The timing and power analysis was done with the Vivado static reports to ensure that the design achieves timing at the desired clock frequency and consumes a low amount of dynamic power. The CFO analysis also reveals the effect of frequency offsetting distort symbols and the effect of correction on enhancing the clustering of symbols. All in all, the project has effectively managed to model a 5G-based digital communication pipeline in simulated form, which has given a clear learning model of how wireless systems work today.

Table of Content

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Full Form
5G	Fifth Generation
4G	Fourth Generation
BER	Bit Error Rate
DSP	Digital Signal Processing
FFT	Fast Fourier Transform
FPGA	Field Programmable Gate Array
HDL	Hardware Description Language
IFFT	Inverse Fast Fourier Transform
LDPC	Low Density Parity Check
OFDM	Orthogonal Frequency Division Multiplexing
QAM	Quadrature Amplitude Modulation
RF	Radio Frequency
SNR	Signal-to-Noise Ratio
SoC	System on Chip
VLSI	Very Large Scale Integration
XDF	Xilinx Design Flow
CFO	Carrier Frequency Offset
XDC	Xilinx Design Constraints

Chapter 1
Introduction

Speed and quality of wireless communication continue to improve with the emergence of advance speeds such as 5G. Nowadays, users demand high internet speed and extremely low delay and dependable connections despite the signal being exposed to noise or even interference. Due to this reason, the systems in the modern world involve elaborate digital methods such that the data are transferred and received appropriately.

Explore these techniques are applicable in an actual system, this project concentrates on developing a model of 5G-inspired communication in Verilog. The project is not the complete 5G system, but it contains the principal digital blocks which are generally utilized in communication. These are QAM mapping, OFDM modulation and demodulation, LDPC coding, channel modeling and simple CFO simulation and correction.

Principal concept of this project was to understand the flow of data between the transmitter and the receiver and the influence of various blocks on communication quality and speed. Through a step-by-step construction of this system, we can understand the variation of latency, the enhancement of coding reliability, noise impact on signal and CFO impact on the constellation.

Project aids in the interpretation of the fundamental arrangement of a communication system, the way signals act within hardware and how simulation tools such as, Vivado have the ability to be exercised in the examination of timing, delays and precision of output. It provides a clear understanding of how contemporary wireless systems operate on the digital level and the reason why every block of the design holds significance in the proper data transmission.

1.1 Background

Wireless communication has become a mandatory aspect of life. Citizens have come to demand high speed internet, great video calls and good signals at all times. There was the increased requirement on a fast speed, reduced delay, and improved performance as the 3G has been replaced by 4G and currently by 5G.

To accomplish this, the digital communication approach has been applied to modern system design like the use of QAM modulation, OFDM, channel coding and synchronization. These methods assist in facilitating data flow even in case of noise, interference or frequency shift in the channel.

Due to such fast development, students and engineers must know how these digital building blocks interact within a real system. Rather than just learning through theory only, it is better to design and simulate these blocks to observe how the data flows in the transmitter and the receiver.

History behind this project is the understanding of how the modern communication operates at a digital level, the change of timing and latency and the increase in accuracy through coding. Constructing and simulating this model stage-by-stage in Vivado allows us to have a concise picture of the issues and the solutions that are employed in the current wireless systems.

1.2 Statistics

Wireless data has been growing at a very fast rate in the recent years. Global reports state that there is an increasing demand of high-speed mobile data that is almost 30-40 percent annually. Due to this expansion, the communication systems are shifting to a higher-tech system, such as 5G.

Following are some statistics that shows the need of efficient digital communication include:

- 5G promises up to 10× lower latency compared to 4G, which is important for real-time communication.
- Number of connected devices is projected to exceed 25 billion all over the world in the near future.
- Almost all modern wireless standards utilize OFDM and QAM as they assist in a high data rate even in a noisy channel.

- Correction schemes such as LDPC are able to mitigate bit errors by over half to three quarters in adverse channel conditions.
- Minor problems such as Carrier Frequency Offset (CFO) may result in significant distortion of constellation points, doubling or even tripling symbol error otherwise.
- Communication systems based on hardware should have a latency of less than 1 ms hence the difference between latencies (such as 105 to 125 ns in our project) is highly significant.
- There are simulation-based tools like Vivado and MATLAB that are common in research and over 60 percent of digital-communication experiments have their testing based on these tools.

For our project:

- In the absence of LDPC the received output was seen to be The LDPC coding yielded better results and was received in 105 ns with a 20 postponement of latency.
- CFO correction enhances the clarity of constellation by avoiding rotation and distortion that increases the accuracy of decoding symbols.

  Statistics demonstrate the need of digital communication systems to have powerful modulation schemes, error-correction codes and synchronization methods. They also warrant the object of our project of knowing such gains by simulation and measurement.

1.3 Prior existing technologies

Numerous communication system and technology already existed before the development of this project that performs the process of digital modulation, transmission through the frequency division method, coding and synchronization of channels. Modern communication technologies such as 4G and 5G are based on these technologies. The project draws inspiration of these methods that exist and reinvents simplified forms of learning and testing. Wireless communication based on

OFDM is one of the most widespread technologies that can be found in 4G LTE, 5G New Radio (NR), Wi-Fi, and WiMAX. Advanced blocks that are found in these real

Systems are full FFT-based OFDM, adaptive modulation, strong channel coding, and hard-ware optimized DSP pipelines. These commercial systems are however highly complicated, need high-end hardware and cannot be modified easily to be learnt or experimented with. Due to this, students are able to learn only theory, however, they are not able to test or see real time internal clues like I/Q samples, constellation points, or latency behaviour. The third group contains and Python communication toolboxes, in which OFDM, QAM, and LDPC are available as blocks of calls. These are highly fast and accurate simulation environments that conceal the internal hardware information. At the RTL level, the student is unable to visualize mapping, FFT, CFO rotation or timing pipelines. Consequently, MATLAB/Python are quite suitable in simulation, but not in understanding of hardware. Numerous past academic applications of OFDM and QAM can be found in the literature, but they tend to consider only a single block (such as only FFT, only mapper or only decoder). They do not demonstrate an end-to-end chain between TX - Channel - RX. They are also not provided with timing delays, latency comparison and constellation plotting of hardware simulation.

1.4 Proposed approach

Project is supposed to create a small and simple digital communication that adheres to the simple ideas implemented in 5G. Rather than developing a complete complex 5G modem, this project aims at a simple SoC design in Verilog that can assist us in understanding how data is coded, mapped, transmitted, received and repaired during transmission. The strategy is based on a modular design. The blocks, QAM mapper, OFDM modulator, and demodulator, LDPC encoder, LDPC decoder, CFO simulator and CFO correction, are constructed individually.

Enables to be tested easily since each unit can be tested before it can be connected to others. In case it is required to improve a block later, then it can be changed, or altered without any other unit being impacted. To note, the effect of coding on message recovery is added by LDPC coding. One is tested with LDPC and without LDPC. Findings indicate variable latency values, which confirm the way in which coding alters system behavior. This aids us in the comparison of performance in real figures rather than more theory. CFO correction is also enclosed. Any misalignment of frequencies in the system creates rotation in constellation points in the OFDM

Systems. In this project, CFO is modeled and simple correction is done on the model by python. This is the way the real systems deal with frequency errors. Transmission blocks are designed in Verilog since an understanding of SoC requires hardware-style coding. The analysis and plotting are performed using Python/MATLAB since it is very fast and easy to visualize through these tools. It produces a harmonious combination of hardware learning and simple analysis. In general, this method is aimed at being clear, simple, and learning-by-observation. Rather than constructing a huge 5G system, a smaller chain is drawn which nevertheless displays major concepts: encoding, modulation, OFDM processing, channel effects, CFO impact, correction, and decoding. This will enable students to know how communication in the modern world functions, how the delay is realized.

1.5 Objectives

Objective 1: Develop a well-defined 5G driven digital communication platform.

Initial aim of this project was to develop a small communication system, which will act as a basic design of a 5G transmitter and receiver. This is not meant to execute the complete 5G standard, but simply recreating the significant digital blocks to allow students to get to know as to how actual communication systems operate internally. The system comprises of QAM mapping, OFDM modulation, OFDM demodulation, LDPC encoding/decoding and CFO simulation. They are written in Verilog code and simulated within Vivado to display a fully functional "TX - Channel - RX" pipeline. One of the objectives is to visualize the process to make it easy to comprehend. Concepts of communication tend to remain abstract, however, by creating constellation diagrams, timing plots and CFO-correction plots in Python, students are able to understand the way a signal varies at each step.

Objective 2: Research the impact of various blocks on the signal quality and latency

Primary goal is to trace the way a signal is altered by each block of the communication chain. In actual wireless, the effect of each block is real-performance enhancing blocks and distorting blocks (that cause delay or distortion of the signal). The project enables the students to observe the actual behaviour of the communication systems by testing the design in various conditions.

Objective 3: Compare performance of the system with/without channel coding (LDPC).

Significant goal of this project is to research on the enhancement of the overall system operation through channel coding. LDPC coding is also commonly applied to the real 5G systems since they assist in correcting the errors that take place during the transmission. The difference is explained in this project through transmitting the same data through two paths wherein one path does not use LDPC and the other one uses LDPC. The comparisons of the outputs are done on message accuracy, correction of errors, signal clarity and total delay. In this comparison, we can easily demonstrate to the students how LDPC will minimize bit errors and stabilize the received constellation, particularly, in the presence of CFO, or noise. The project also demonstrates that LDPC brings about minor delay as a result of encoding and decoding which allows learners to learn the trade-off between speed and reliability of a communication system. The aim provides a real-world insight into why wireless systems employ channel coding and the effect it has in enhancing the quality of wireless communication.

Project compares four major cases:

1. Without LDPC

In the absence of LDPC, the raw data goes through the system in an unprotected form. Any disturbance or distortion alters the bits received directly which raises the likelihood of bits error. Although it has the lowest delay, it cannot be used in the real-life communication as minor changes may spoil the message entirely.

2. With LDPC

LDPC enhances the stability of the received bits by repairing majority of the most frequent common channel errors. LDPC is used to recover the original bitstream with great accuracy even in the presence of the noise or CFO in the system to some extent. Even though it includes minimal processing time cost, the reliability and message accuracy increase makes LDPC necessary in the modern wireless systems.

3. With CFO

CFO leads to one period rotation and the clusters of symbols become not at their

appropriate QAM position. With increased CFO, separation of symbols is lost, and bits are mis-coded. This makes the system unstable when CFO is not corrected and the BER increases substantially despite correction of the modulation and coding.

4. With CFO correction

Once de-rotation has been applied, the constellation will be clear and there will be a chance to decode the message properly by the system again.

By observing these cases, we understand how:

- signal quality changes,
- latency increases or decreases,
- errors appear and get corrected.

1.6 Overview of project report

Approach in this project presents how the simplified 5G-inspired digital communication system was planned, designed and tested in a simple and well structured manner. It began with the knowledge of the fundamental principles of communication like QAM modulation, OFDM, LDPC coding, carrier-frequency offset (CFO) and the influence of each block on a digital signal. Once the system requirements were identified, modules were created using the fixed and uniform bit-widths in Verilog to eliminate the problem of alignment in the integration. All the blocks, LDPC encoder, QAM modulator, OFDM modulator, OFDM demodulator, QAM demodulator and LDPC decoder, were developed separately and then tested on simple testbenches to test their functionality. They were then linked together to make a complete TX-Channel-RX chain when each of the modules was stable. This combination was in the concept of modular design whereby the outputs of a block fitted the correct input to another block. Simulation of waveforms, timing checking and static analysis were performed using vivado to ensure correct logic behaviour at the desired clock frequency. The Python CFO simulation, CFO correction, constellation plotting and error-analysis were used to visualize the way the signal varied in each stage. This serial and reusable approach to the methodology also made the end system reliable ,debuggable , easy, and it was a great way to illustrate the fundamental principles of communication.

Chapter 1: Provides the introduction to the entire project. It describes the history of modern communication systems, the reasons why high-speed and low-latency systems are required, and why a simplified 5G-inspired SoC model is beneficial in the learning process. It also describes the purpose of the project, the statistics of real 5G development, current technologies, and the intentions of constructing this design. The chapter prepares the foundation of the whole report

Chapter 2: Allows literature survey. It contains studies on QAM, OFDM, channel coding and LDPC and CFO and other communication blocks in 4G/5G systems. It describes the functioning of industry-standard systems and compares them with simplified models easy to learn by the student. It further addresses the previous work that was conducted with Verilog, MATLAB and Python to simulate communication. This chapter assists in recognizing the gaps, which this project addresses, including demonstrating effectual constellation variations and latency disparities.

Chapter 3: Explains the process of the project construction in stages: requirement study, block division, TX and RX chains designs, testing of individual modules, integration and debugging. It also justifies the reasons why Verilog was selected to do hardware design and Python/MATLAB were selected to do analysis. The chapter displays the steps of work that were followed during the project.

Chapter 4: Discusses about project management, such as planning, scheduling and risk management. It has Gantt charts that will depict the schedule of designing every block like QAM Mapper, OFDM Modulator, LDPC Encoder, OFDM Receiver, CFO Correction and the combination of the entire chain. Such risks as high time of simulation, synthesis errors, timing problems, and complex debugging are identified and controlled.

Chapter 5: Encompasses the design of the entire end to end communication chain. They include the block diagrams, the structure of the frame in the OFDM, the domain model, communication model, and internal data flows. Every block like QAM, IFFT, Channel Model, CFO, FFT, LDPC Decoder and De-mapper is described in a clear manner. This is the technical hub of the project.

Chapter 6: Describes about the real Verilog code of every module, integration of TX and RX module in the top file, FSM code, timing forms, and the simulation output is found in this chapter. Python/MATLAB scripts of constellation plots, CFO

graphs and signal visualization are also presented. In this chapter, the execution of the design in hardware simulation tools is proven.

Chapter 7: Discusses about the testing and evaluation. It contains the test plan, test points, test cases and data of latency, signal quality, constellation clarity and coded and uncoded performance. It is a comparison between without LDPC, with LDPC, with CFO and with CFO correction. This chapter expounds on the variation of system behaviour in various conditions and ascertains the accuracy of the overall design.

Chapter 8: It discusses how this project can assist students in gaining an idea of the 5G concept, enables low-cost digital learning, and lessens the need to rely on expensive equipment, through simulation tools. Code, transparency, and open-source tools are covered in terms of ethical considerations. The project helps in achieving sustainability by encouraging the use of digital learning rather than wastes of hardware.

Chapter 9: Conclusion is given of the main achievements. They are successful TX to RX transmission, working LDPC encoding and decoding, better Carrier Frequency Offset (CFO) correction, and successful constellation recovery as well as a comparison of system latency. In the next work, there are various enhancements proposed that take into consideration the fading channels and noise models, higher ordering Quadrature Amplitude Modulation (QAM), full matrices of LDPC, etc.

Chapter 2

Literature review

2.1.1 LDPC Coding and 5G Error-Correction Techniques

Paper 1, which is titled Demystifying 5G Polar and LDPC Codes, gave a good idea of the LDPC structure, encoding flow, parity-check matrices and decoding methods applied in 5G NR. The authors describe the advantage of LDPC codes like reduced decoding complexity, parallelizable structure and appropriateness in software implementation, particularly in high throughput communications. The iterative decoding algorithms that were discussed in the paper include Min-Sum and Belief Propagation and how they minimize, BER in noisy and fading channels. In this project, the insights were used to come up with an encoder/decoder pair in simplified, yet synthesizable LDPC, in Verilog. Other issues identified in the paper include the use of hardware memory and repetitive delay that informed the choice to ensure that the LDPC block remained lightweight to fit within the FPGA/Vivado requirements. On balance, the paper enhanced the knowledge regarding the reasons why LDPC is favorable in contemporary wireless designs and the manner in which it is better at enhancing reliability when incorporated into a digital baseband SoC.

2.2.1 OFDM Modulation in 5G Systems.

The article "High Fidelity Indoor MIMO Radio Access to 5G and Beyond" (Paper 2) addressed the concept of OFDM, multi-carrier structure, fading profile, IFFT/FFT implementation problems, and synchronization. The authors highlighted the issue of the importance of how the OFDM enables high spectral efficiency and is resistant to multipath fading, which is significant in 5G. Cyclic prefix, subcarrier spacing and pilot-based channel estimation were also described in the paper. This work was of direct use to the development of the OFDM module in the project leading to the motivation to do proper implementation of IFFT in transmission and FFT in reception. It also advised CFO modelling, correction methods and the way noise displaces constellation points. The concepts of this paper were used to validate the constellation plots that were created in Python/MATLAB. The lessons learned assisted in assuring that the transmitter-receiver chain will act similar to the actual 5G OFDM behaviour although simplified.

2.3.1 Hardware Architectures of Low-Latency LDPC.

The article Low Latency LDPC Encoder Design for 5G NR (Paper 3) reported the optimized schemes of LDPC encoders with low latency and low area cost. It talked about photograph based LDPC codes, sizes of lifting, pipelining architecture and memory optimization. In this project, the concepts aided in the design of a small LDPC encoder block which can be simulated and tested in hard-ware. This timing efficiency was a source of stability in timing the results in Vivado, particularly, seeing that there was no negative slack in post-synthesis reporting. It also further supported the role of modularity and parallelism in the LDPC design, which was implemented in the simplified hardware architecture in the SoC in this project.

2.4.1 OFDM Transmitter/Receiver OFDM Designs in Verilog.

This aspect talks about the Design and Implementation of an OFDM Transmitter and Receiver using FPGA. It describes how a step-by-step Verilog implementation can create an OFDM block (framing, modulation, IFFT/FFT, add cyclic prefix, remove cyclic prefix and synchronization). Hardware results like timing, power and resource utilization were also shown in the paper. This paper was a direct support to application of the transmitter and receiver modules in the project. Its code structure, state machine samples and FPGA timing description assisted in modeling the Tx/Rx chain of the project. This was also adopted as the methodology of testing each block using testbenches in advance before actual integration.

2.5.1 5G Baseband Processing SoC Design. The article SoC Architecture for 5G Physical Layer Baseband Processing

It talks about how to put several PHY modules, LDPC, OFDM, QAM, etc., in a single hardware SoC. It pointed out such problems as pipelining, synchronization, interfacing between modules and matching throughput. This study was very much similar to the objective of the project, which was to combine the use of QAM, OFDM, LDPC, and CFO correction in one SoC. The top-level module design was based on the discussion of interface timing and data flow presented in the paper It also explained the rationale of having modular blocks pass homogenous bit-width signals to eliminate data alignment issues. The SoC-level integration of this paper inspired the plan of the stable architecture, the design of the Verilog hierarchy.

2.2 Summary of Literatures reviewed

SL NO	TITLE	OUTCOME	MILESTONE
01	Demystifying 5G Polar and LDPC Codes	Provided an in-depth understanding of LDPC and Polar codes used in 5G NR, including structure, encoding/decoding processes, and performance insights.	Helped understand LDPC structure and development of **L**DPC encoder/decoder module in Verilog.
02	High Fidelity Indoor MIMO Radio Access for 5G and Beyond	Described OFDM, MIMO modeling, channel impairments, fading profiles, and real-world implementation challenges.	Guided the OFDM module design, and understanding how wireless channel behavior affects system performance.
03	Low Latency LDPC Encoder Design for 5G NR	Proposed efficient architectures for low-latency LDPC encoders compliant with 5G NR standards; discussed resource optimization for hardware synthesis.	Helped design a synthesizable LDPC encoder optimized for low latency and suitable for FPGA/Vivado implementation
04	Design and Implementation of an OFDM Transmitter and Receiver for 5G NR Using FPGA	Provides practical guidance on Verilog/VHDL implementation of OFDM Tx/Rx, including framing, IFFT/FFT, cyclic prefix, and synchronization.	Directly supports OFDM Transmitter and Receiver Verilog module implementation, matching your workflow with Vivado synthesis and simulation.
05	SoC Architecture for 5G Physical Layer Baseband Processing	Discusses integration of multiple 5G PHY modules (LDPC, OFDM, QAM) into a single SoC, focusing on interface design, timing, and throughput.	Highly aligned with integration of LDPC + OFDM + QAM + Tx/Rx into one Top Module, offering insights to SoC-level design, module.

Table 2.2 Summary of Literature reviews

Chapter 3

Methodology

Structure of this project report has been made in a step-wise and straight forward way such that any reader of this report can easily be able to know how the simplified 5G-inspired communication system was planned, designed, implemented and tested. In every chapter, the authors discuss a single stage of the project, beginning with an introductory presentation of the fundamental principles of the wireless communication concept and the rationale behind the construction of the system. The report proceeds to design important modules including QAM, OFDM, LDPC, CFO simulation and CFO correction. Subsequently, the chapters describe the manner in which these modules were combined to create a whole transmitter-receiver chain. The Vivado process of simulation and Python/MATLAB process of signal analysis has also been described step-by-step demonstrating how the system works at each point. The chapter progression is straightforward and logical such that non-technical and technical readers can easily track the development process including the original idea up to a complete functioning simulated system. The report, by offering the work in such an organized format, guarantees the clarity of the learning process, design decisions, and testing outcomes in a clear and easy to review format.

3.1 System Requirements

Initial process of the project was to know what the digital communication system was to do. The requirements were not complex as the objective was to create a small 5G-inspired SoC. This system had to receive an input message, encode it in QAM symbols, produce the OFDM signal, transmit the signal over a channel with noise and CFO, and decode the signal at the receiver. This implied that this SoC needed to contain QAM Mapper/De-mapper, OFDM Modulator/Demodulator, LDPC Encoder/Decoder, addition and correction of CFO and a simple channel model.

3.2 High-Level Architecture

High-level architecture illustrates the operation of the whole TX-RX chain of communication. It serves as a blueprint to the SoC, and displays the flow of data between blocks. At the transmitter, digital information goes through QAM mapping, OFDM modulation, LDPC encoding and CFO addition. These measures produce an

analogue of a true wireless transmission. Noise and frequency offset are introduced by the channel block to ensure that meaningful errors are present that the receiver is to correct. The architecture at the receiver side comprises of CFO estimation and correction, demodulation of the OFDM signal, de-mapping of QAM signal, decoding using LDPC and reassembling the message. There is a corresponding block on the TX side on the RX side. This simplifies the process of debugging since it is easy to trace the errors to any one stage. Latency checkpoints are put before and after key processing blocks in comparison with behaviour with LDPC, without LDPC, and with CFO correction. The architecture is also modular, i.e. every block can be created and tested separately and then when all is complete the architecture is bound together. This top-level design provides a system which acts as a simplified version of a 5G physical-layer chain but remains simple enough to learn and be simulated by the students.

3.3 Module Design

All modules in the project had been made independent to enable them be introduced independently to be implemented, tested and debugged. The QAM Mapper converts bits of digital data to I/Q symbols. The OFDM Modulator introduces cyclic prefix as well as IFFTs so as to form multi-carrier symbols. The LDPC Encoder is used to introduce redundancy to the data in order to mitigate the effects of error and the associated decoder at the receiver restores the original bits. Simulation and correction modules of CFO were implemented to bring in realistic impairments and then eliminated. A simple design was adopted in every block in order to make the behaviour simple to comprehend and test. The modules were linked by standard interfaces thus allowing smooth flow of signals between modules. Timing considerations were also made in each module design (e.g. the number of cycles required by the input of a module to generate an output). This was made possible by the modular nature of the design which allowed testing each block individually and then assemble them in the final SoC. This also assisted in the determination of the effect which each block has on the quality, latency and accuracy of the final output.

3.4 HDL Coding

Once the module designs were complete each block had been implemented in Verilog. The HDL code was divided into different files which include: qammod.v,

qamdemod.v, ofdmtx.v, ofdmrx.v, ldpcencoder.v, ldpcdecoder.v, cfoadd.v and cfocorrect.v. Individual files were used to design each function to ensure that the system was clean and easy to maintain. Coding was done based on clarity and not complexity in order to trace the behaviour of each block along a waveform easily. Whenever required, registers, counters and finite state machines were utilized. Caution was observed to ensure that every module is designed to conform to rules of synchronous design to prevent the occurrence of glitches or glitchy outputs. Parameters were included to enable easy modification of FFT size, QAM order or LDPC parameters. Each of the modules was coded and then testbench-verified prior to being added to the SoC. The bottom-up technique ensured that problems were spotted early and errors were prevented in the case of integration.

3.5 Unit/Testbench

Individual modules of Verilog were tested with special testbenches. These testbenches created input patterns, tested them on the module and ensured that the outputs were to the expected results. As an illustration, symbol mapping was checked by the QAM testbench and cyclic prefix and FFT outputs were checked by the OFDM testbench. The transitions of signals, position and timing of symbols were observed using waveforms provided by Vivado. In the case of the LDPC decoder, performance in terms of error-correction was checked using a number of test cases. Frequency offsets were added to CFO blocks and it was checked whether they were removed or not by the correction block. This testing phase was a way of guaranteeing that all the modules were functional before integrating them in the complete SoC. It saved time in the process of debugging as it integrated, and it ensured that every block achieved its functional requirements.

3.6 Integration Testing

All the modules had been tested individually and the connections were made between the blocks to form the entire TX-RX chain. Integration testing entailed deployment of entire messages through the complete system and ensuring that the receiver generated the same message as the sender. The primary concern of the integration was checking timing, latency and flow of signals. Certain modules needed to be adjusted in terms of synchronization and buffering to correct the inputs and output. This was tested on CFO correction since it requires both channel blocks

and receiver blocks. This test also involved comparison of system performance after the addition of LDPC and without, after the addition of CFO correction and without. In both cases, delays were measured by timing analysis of Vivado. Integration testing ensured that all modules interacted only in a proper way and the SoC was acting as a simplified wireless system.

3.7 Implementation Phase

Last thing to do was to combine all the modules, simulations, Python/MATLAB scripts, waveforms and timing report into a single complete project. Synthesis was done in Vivado and implementation and static timing analysis done in order to ensure that the design was within the necessary clock constraints. Constellation graphs were created and Python/MATLAB scripts were used to verify CFO behaviour. All the end products like waveforms, plots, and latency readings and message comparison results were assembled in this stage. After making sure that all the modules were operating properly, a structured document was written on how each block works and how the final SoC fulfills the initial goals. On top of these measures, added verification was done to ensure the design was stable at varying CFO values. The non-LDPC and the LDPC paths were again compared to confirm the reliability gains. Pipeline timing was also given high concern in order to make the system generate similar outputs each clock cycle. Power estimation was also look into to know how logic, I/O and clock networks behave. After all these checks were done, the final integrated design was now available to be documented, presented and submitted as a complete communication system model.

Chapter 4

Project Management

4.1 Project timeline

Creation of the 5G digital communication SoC has followed a semester-long structure, which included study of the requirements, architecture design, implementation in Verilog, testing at module level, integration, simulation and final report. The project went by a strict adherence of the academic window of July 2025 to December 2025 to have each stage completed punctually and in line with weekly milestones that were established early in Review 1.
The initial two weeks were devoted to the analysis of requirements (P1) and literature review (P2). In these weeks, the basis of the OFDM, QAM, LDPC, CFO estimation and the predecessor digital-baseband designs were learned. This was part of an attempt to define the scope of the SoC, what signal-processing blocks would be covered in the system, and sort out the performance assumptions (symbol rate, bit-width and memory requirements).

Weeks 3 and 4 were devoted to the system architecture design (P3). An architecture of the transmitter and receiver was designed in block-level in its entirety, including the QAM mapper, cyclic prefix, CFO adder, LDPC encoder/decoder and the RX chain. Here, data flow, internal interfaces, and latency flow as well as handshake signals were defined. Following the supervisor approval, the architecture milestone was accomplished.

Weeks 5 and 6 were concerned with functional and unit-level design (P4). All of the blocks, QAM, OFDM, LDPC, and CFO correction, had sub modules. Details were provided on the inputs, outputs, data widths and internal pipelines. This set the groundwork of coding.

Verilog coding was assigned during week 7-10. Each of the modules was deployed in synchronous design and tested with small datasets to verify the inside functionality.

Weeks 11-13 were spent on testing each module separately, testing each block with specific testbenches. Time-related debugging, round-off errors and fixed-point errors occurred here.

14th and 15th weeks were on module integration, where all the TX-RX components were integrated into one SoC. This step was aimed at having the right data flow, latency matching, and solving interface conflicts.

Weeks 16-17 were focused on testing, simulation and validation, timing analysis, power review, resource usage, constellation plotting, CFO correction results and performance comparison between LDPC and no-LDPC. The validity of the system behavior was tested by the waveform analysis at 105 ns and 125ns and the difference in latency of 20 ns was established.

The last week (Week 18) entailed report writing, preparation and documentation of final assessment whereby all the workflow-architecture, coding, testing and insights were compiled. Analysis and Design phase is concerned with the question of what should be done by the 5G-inspired communication system and how each of the modules should be organized. Firstly, the system requirements were determined, and the project needs to send and receive data through the QAM-OFDM, CFO modelling, CFO correction, and LDPC coding, which is optional. All the blocks should be coded in Verilog and verified using Vivado simulation and additional verification was carried out using Python constellation plots.

Task Name	Start Date	End Week	Duration (Weeks)	Dependency	Milestone
Requirement & Analysis	Week 1	Week 2	2	-	Requirements Finalized
Literature Review	Week 1	Week 2	2	-	Comprehensive Review
System Architecture Design	Week 3	Week 4	2	P1	System Design Approval
Functional & Unit Design	Week 5	Week 6	2	P2	Component Design Complete

Table 4.1: Project planning timeline

Task Name	Start Date	End Week	Duration (Weeks)	Dependency	Milestone
Verilog Coding	Week 7	Week 10	3	P4	Coding Complete
Individual Module testing	Week 11	Week 13	2	P4	Testing complete
Integrating Modules	Week 14	Week 15	1	P4	Single compilation
Testing Simulation & Validation	Week 16	Week 17	1	P4	Performance Verified
Final Report & Documentation	Week 17	Week 18	2	P4	Project Completed

Table 4.2: Project implementation timeline

Chapter 5

Analysis and Design

5.1 Requirements

This was followed by a highlevel architecture. The transmitter side comprises of message input, QAM mapper, OFDM modulator with the help of IFFT, cyclic prefix, optional LDPC encoder and the addition of CFO. These steps are reversed by the receiver with CP removal, FFT, CFO estimation and correction, QAM demapper and optional LDPC decoding. This design has a definite step-by-step flow of data flow through the system. The modules were then designed separately to ensure that the system was simple and easy to debug. A dataflow path was drawn and kept clean to ensure that signals flow through one block to another. The entire testbench was to be designed to test the behavior of symbols on each stage, to compare the results with the results without LDPC and CFO correction.

5.1.1 Transmitter Module

The transmitter module will mark the beginning of the whole communication system. Its primary task is to put the input message into a digital form which can be transmitted to the other parts of the system. The characters that are received by the transmitter in our project include H, E, L, L, O and turned the characters into an 8-bit binary value. This 8-bit number is used as the raw input of the subsequent module. The transmitter makes sure that all the bytes are transmitted with a correct txvalid signal to allow the system to recognize the presence of a new data. Though not a complicated block, this block is significant since it constitutes the gateway to the entire communication channel. The transmitter does not modify the length of bits or the number of bits but merely prepares the clean digital data packet to be encoded. The structure simplifies the project and we can easily follow it as we can clearly see how plain text is converted into digital bits before it is in turn passed on to other more sophisticated communication modules.

5.1.2 LDPC Encoder Module

The first advanced processing block is LDPC encoder. This module takes the 8-bit input and fills it with more bits in an encoded block by including parity bits. LDPC encoding lands up in increasing the size of bits as it implies redundancy. This

redundancy is used in order to identify and rectify errors in the receiver. LDPC is applied in the real 5G systems to personalities of high-reliability and low rate of error. In our project we are simplifying an encoder stub of LDPC encoder but the principle is similar: an encoder transforms a single byte into a bigger bit array (for example, a 32-bit one). These 32 bits have data bits and parity bits. The result of the LDPC encoder is the version of the message that has been protected. The error without LDPC and with LDPC is easily noticeable both in constellation patterns and latency timing. The LDPC provides protection to the data but introduces a slight delay due to the additional processing

5.1.3 QAM Modulator Module

The QAM modulator codes groups of bits into an I and Q complex number. As an example, 16-QAM allows reducing 4 bits to a single symbol. The bits input to the modulator are converted to certain constellation points. All the points are represented by two 16-bit signed numbers: I (In-phase) and Q (Quadrature-phase). The values of these I/Q pairs reflect the real-life signal values. In the event that LDPC output consisted of 32 bits, the QAM modulator would pack them 4-bit blocks to generate 8 QAM symbols. This module converts digital bits to graphical points visually which in turn will be seen in the constellation diagrams. As noise or CFO is introduced later in the stages, these points scatter and display distortions which can be used to comprehend signal behavior. The QAM modulator is essential in converting digital data to a modulation form utilized in the OFDM systems.

5.1.4 OFDM Modulation Module

Following QAM mapping the symbols are sent through the OFDM modulator. OFDM divides several QAM symbols and puts them in various subcarriers. To illustrate, with an FTT length of 64, a large amount of QAM symbols are overlaid in a frequency grid. With the help of an IFFT, the system transforms these frequency content into a time-domain waveform. It is this waveform that is really transmitted through a channel. The cyclic prefix (CP) is also introduced by the OFDM modulator in order to prevent inter-symbol interference. Our project has a 16-bit time sample (txtimei, txtimeq) as the output of the modulator (OFDM). The block is indispensable since OFDM is the foundation of wireless systems nowadays (even

5G). The absence of it would cause a lot of interference among signals and destruction in multipath channels.

5.1.5 OFDM Demodulation Module

The demodulator of the OFDM is the opposite of the modulator. It accepts samples in time-domain, de-cyclic prefixes them and uses an FFT to transform them back to frequency-domain QAM symbols. This step tries to restore the original symbols despite its distortion during the transmission of the signal. In the simple implementation of our project (without CFO correction), the demodulated symbols appear rotated or scattered. Upon CFO correction, the constellation points are clean and in the right clustering. The module provides I/Q symbols (outi, outq) to the QAM demodulator. The step is essential in the inversion of the operations of the OFDM and the recovery of the subcarrier data.

5.1.6 QAM Demodulator Module

The OFDM provides noisy or rotated constellation points to the QAM demodulator. It is the job of it to determine which QAM symbol any given I/Q pair corresponds to. It determines the received point and the closest valid point in the constellation. Once this has been performed then it will turn each symbol again into the original 4-bit (in 16-QAM) or other bit group. These bits are regained to create raw data input to the LDPC decoder. When noise or CFO is great, the demodulator could map symbols to the wrong points so that it produced bit errors. With the LDPC, it is possible to correct these errors. In the absence of LDPC, the errors can directly reach the output bit and corrupt it.

5.1.7 LDPC Decoder Module

The demodulated bits are sent to the LDPC decoder which tries to correct errors with the help of parity-check equations. LDPC can usually give the original data even when the QAM demodulator has provided the bit errors as a result of noise. In our project, the LDPC decoder stub transforms the 32 bits block into the original 8 bits block. This block has made sure that it is reliable and demonstrates why current communication systems never lack some provision of error correction. The message received may be having incorrect letters without the decoder. Through the decoder, the end result is precise and constant.

5.1.8 Receiver Module

The other end of the chain is the receiver. It gets the corrected 8-bit data (rx-byte) and determines whether or not the system was able to recover the original message. It compares it to expected output and the result is displayed as either PASS or FAIL. The receiver assists us in knowing the system latency, accuracy, and error rates. In the case of LDPC, the message received is earlier (105 ns). In the absence of LDPC, it occurs a little later (125 ns). The receiver closes the communication loop and gives the resultant observable output.

5.2 Block diagram

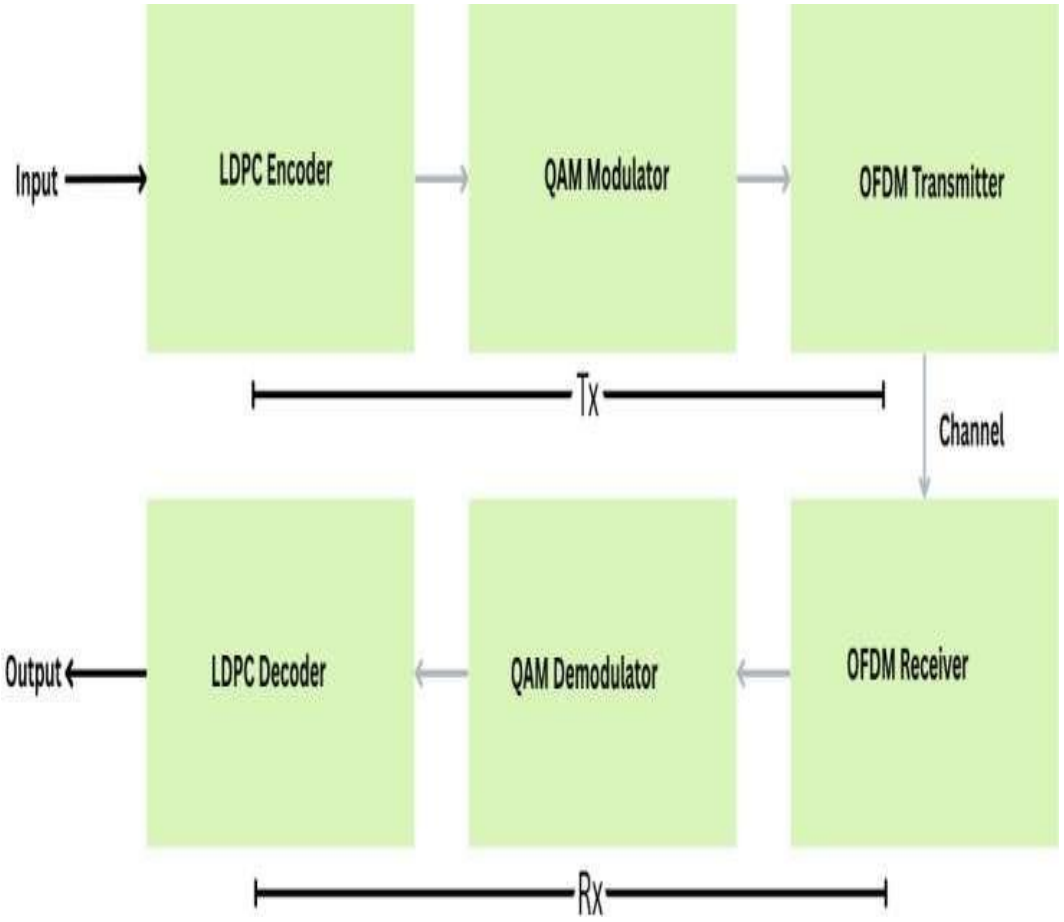


Figure 5.2 Functional block diagram

5.3 System Flow chart

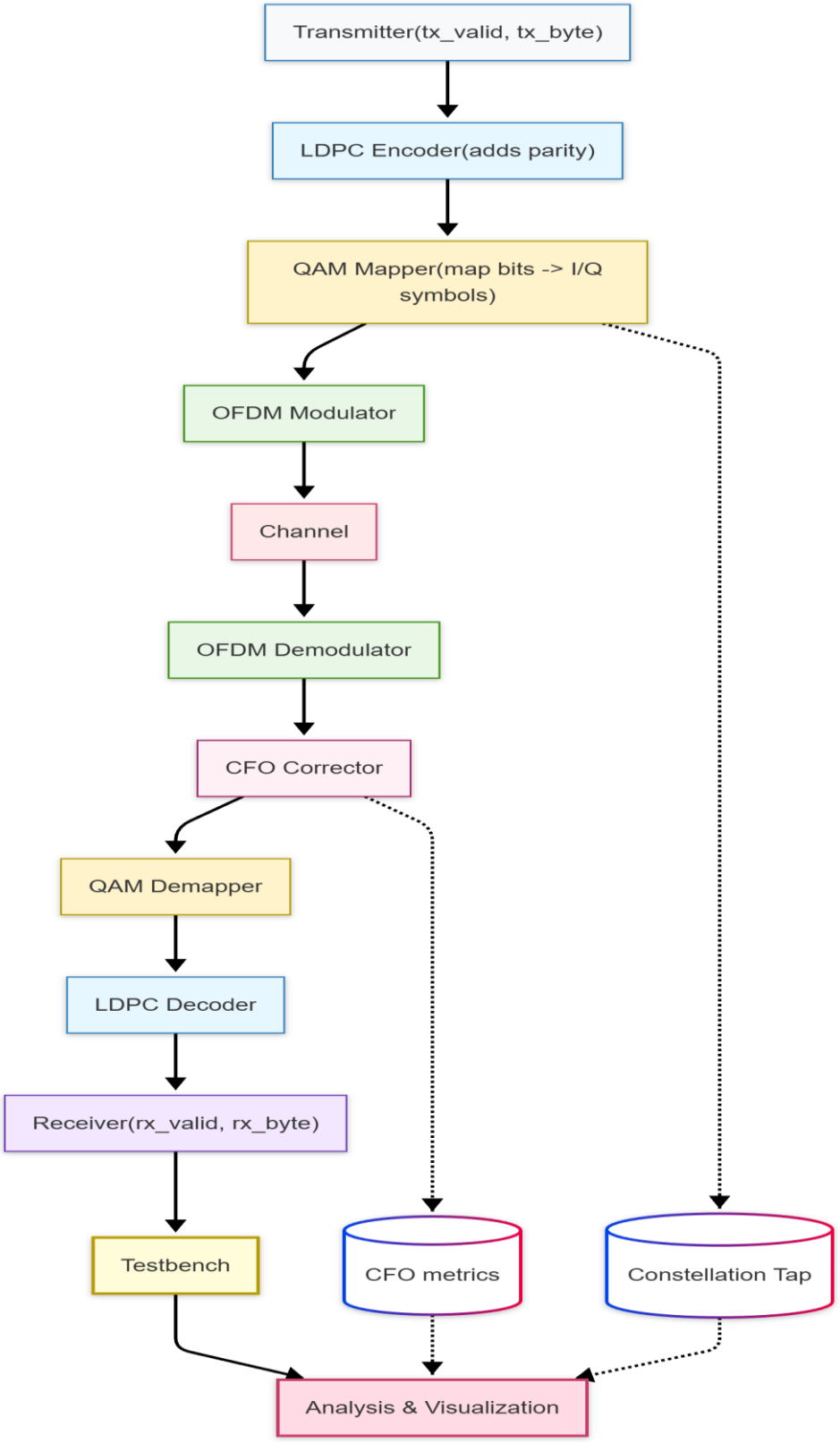


Figure 5.3 System flow chart

Chapter 6

Hardware, Software and Simulation

6.1 Hardware

The equipment requirement of this project is also very low since the whole 5G-inspired communication system is developed and verified on the basis of the simulation of Verilog instead of real RF equipment. This is all implemented digitally within Vivado; each and every module which includes: QAM, OFDM, LDPC Encoder/Decoder, CFO modeling and the entire transmitter-receiver chain are all implemented in Vivado. This enables the entire communication procedure to be put through test without external sensors, antennas, or hardware radios. All one needs is a computer that can run synthesis and simulation tools without any problems. In the case of this project, the design was generated and tested on a system that has 4GB to 8GB RAM which is enough to execute Vivado Simulator, waveform viewer and Python analysis scripts. There was no need of an FPGA board since testing and debugging was fully done through behavioral simulations. Constellation analysis and CFO estimation visualization were done only with Python/MATLAB tools, which can be also run without any problems on a low-end laptop. The system employs no ADCs, DACs, RF mixers, oscillators, antennas, or power amplifiers because the system employs digital baseband processing.

6.2 Software development tools

System of digital communication of 5G was fully developed with the help of software as the project is an entirely simulation-based one. There was no physical hardware or sensors or boards needed. Verilog code, module synthesis, integration of the transmitter and receiver and simulation verification of signal flow was largely written using Vivado. Vivado waveform viewer was useful in monitoring timing, error detection and latency measurements between various modules. Constellation diagrams and CFO rotation, CFO correction and IQ samples exported IQ samples analysis were plotted with Python and MATLAB. These tools enabled one to visually inspect the signal with respect to the effects of QAM, OFDM, LDPC, and CFO. The tools used are light and can be run on the normal student laptops with 4-8

GB RAM. All it is software-based, which means that it is simple to test, debug, and repeat, and simulation is a viable approach to learning the workings of 5G-baseband processing.

6.2.1 Vivado (Xilinx/AMD)

It was used for:

- TLR code Writing Verilog modules (TX, RX, LDPC, QAM, OFDM, CFO) Linking together all modules within the upper design.
- Conducting behavioral simulation.
- Displaying signal waveforms (I/Q samples, valid signals, symbol indexes) Classifying timing and latency problems.
- Vivado Simulator enabled us to test timing (arrival at 105 ns / 125 ns) and test the effect of LDPC and CFO.

6.2.2 Python program (to analyze constellation and CFO)

This was simulated using Python where the iq_dump.txt generated by Vivado was read.

We used:

- NumPy - to process I/Q samples
- Matplotlib - in order to sketch constellation schemes.
- SciPy - to estimate CFO phase fitting.
- Python helped us visualize

6.2.3 GitHub (Project Storage)

GitHub was used to:

- Store Verilog code
- Store Python scripts

6.3 Software code

```
// top.v - self-contained (no consts.v dependency)
`timescale 1ns/1ps
module top (
    input  wire clk,
    input  wire rstn,
    input  wire tx_valid,
    input  wire [7:0] tx_byte,
    output wire rx_valid,
    output wire [7:0] rx_byte
);

// ----- local "constants" used by this top (explicit values) -----
localparam integer FFT_LEN   = 64;
localparam integer CPLEN     = 16;
localparam integer MOD_ORDER = 16;

// ----- internal wires -----
wire enc_valid;
wire [31:0] enc_out;
wire map_valid;
wire signed [15:0] map_i, map_q;
wire ofdm_time_valid;
wire signed [15:0] ofdm_time_i, ofdm_time_q;
wire ofdm_sym_valid;
wire signed [15:0] ofdm_sym_i, ofdm_sym_q;
wire demap_valid;
wire [7:0] demap_sym;
wire dec_valid;
wire [7:0] dec_out_byte;

// ----- encoder (stub) -----
ldpc_encoder_stub ldpc_enc (
    .clk(clk),
    .rstn(rstn),
    .valid_in(tx_valid),
    .in_byte(tx_byte),
    .valid_out(enc_valid),
    .out_bits(enc_out)
);

// ----- qam mapper (explicit parameter) -----
qam_mapper #(.M(MOD_ORDER)) qam_map (
    .clk(clk),
    .rstn(rstn),
    .valid_in(enc_valid),
    .in_bits(enc_out[7:0]),
```

```
    .valid_out(map_valid),
    .out_i(map_i),
    .out_q(map_q)
);

// ----- ofdm mod (use module default params) -----
ofdm_mod_stub ofdm_tx (
    .clk(clk),
    .rstn(rstn),
    .valid_in(map_valid),
    .in_i(map_i),
    .in_q(map_q),
    .valid_out(ofdm_time_valid),
    .tx_time_i(ofdm_time_i),
    .tx_time_q(ofdm_time_q)
);

// simple pass-through channel
wire chan_valid = ofdm_time_valid;
wire signed [15:0] chan_i = ofdm_time_i;
wire signed [15:0] chan_q = ofdm_time_q;

// ----- ofdm demod -----
ofdm_demod_stub ofdm_rx (
    .clk(clk),
    .rstn(rstn),
    .valid_in(chan_valid),
    .rx_time_i(chan_i),
    .rx_time_q(chan_q),
    .valid_out(ofdm_sym_valid),
    .out_i(ofdm_sym_i),
    .out_q(ofdm_sym_q)
);

// ----- qam demapper (explicit parameter) -----
qam_demapper #(.M(MOD_ORDER)) qam_demap (
    .clk(clk),
    .rstn(rstn),
    .valid_in(ofdm_sym_valid),
    .in_i(ofdm_sym_i),
    .in_q(ofdm_sym_q),
    .valid_out(demap_valid),
    .out_sym(demap_sym)
);

// ----- ldpc decoder (stub) -----
ldpc_decoder_stub ldpc_dec (
```

```
    .clk(clk),
    .rstn(rstn),
    .valid_in(demap_valid),
    .in_bits({demap_sym, demap_sym, demap_sym, demap_sym}),
    .valid_out(dec_valid),
    .out_byte(dec_out_byte)
);

// ----- outputs -----
assign rx_valid = dec_valid;
assign rx_byte  = dec_out_byte;

// ----------------------------------------------------------------
// I/Q logging for receiver (file dump)
// Writes one line per received OFDM symbol (when ofdm_sym_valid=1):
//   <time_ns> <in_i (decimal)> <in_q (decimal)> <symbol_index>
// File: iq_dump.txt
// ----------------------------------------------------------------
integer iq_fd;
reg [31:0] symbol_cnt;

initial begin
    // open file for writing (overwrites if exists)
    iq_fd = $fopen("iq_dump.txt", "w");
    if (iq_fd == 0) begin
        $display("ERROR: could not open iq_dump.txt for writing");
    end else begin
        $fwrite(iq_fd, "# time_ns in_i in_q sym_index\n");
    end
    symbol_cnt = 0;
end

always @(posedge clk or negedge rstn) begin
    if (!rstn) begin
        symbol_cnt <= 32'd0;
    end else begin
        if (ofdm_sym_valid) begin
            // write: time (ns), I, Q, symbol index
            if (iq_fd != 0) begin
                $fwrite(iq_fd, "%0d %0d %0d %0d\n",
                        $time,
                        $signed(ofdm_sym_i),
                        $signed(ofdm_sym_q),
                        symbol_cnt);
            end
```

```
            symbol_cnt <= symbol_cnt + 1;
        end
    end
end

end module;
```

6.4 Simulation

6.4.1 Explanation of Full model simulation

Figure 6.4.1 Simulation waveform shows the end to end nature of the digital communication chain that was implemented in Verilog. The clock (clk) is at the top of the waveform and it is always on except at the start when the reset (rstn) is released to ensure that all registers are cleared and then the system starts operating by processing data. After de-asserting of the reset, the transmitter begins to transmit bytes which are indicated on the txbyte signal. The launching occurs with each clk cycle of the high of the txvalid. Such input bytes, e.g. 48, 45, 4C, 4F (ASCII values), are fed in the encoding and modulation pipeline. Due to the internal registers in all the blocks, i.e. LDPC, QAM, OFDM, the simulation demonstrates a chain of small delays in which the data is transferred between one block and another. This can be observed by the gradual movement of the byte values to the right as the various waveform lines move on. The symbols are identified by the OFDM modulator and transformed into the time-domain samples (ofdmtimei / ofdmtimeq), which are handled after which the symbols are restored by the OFDM demodulator. These values are in turn converted back to soft or hard decision bits by the QAM demapper. This output is passed to the decoder that ultimately sets devalid that causes the assertion of the decvalid at the top module output, and the rxbyte which verifies the received data. The waveform is a clear indication that every transmitted byte is replicated at the receiver after a certain time which can be measured as the System latency. This latency may be determined by subtracting the timestamp of the rise of the txvalid edge with the corresponding rise of the rxvalid edge. Short intervals of XXXXXX indicating unallocated values when the system is ever in a reset state or before a module produces valid values are also visible in the trace.

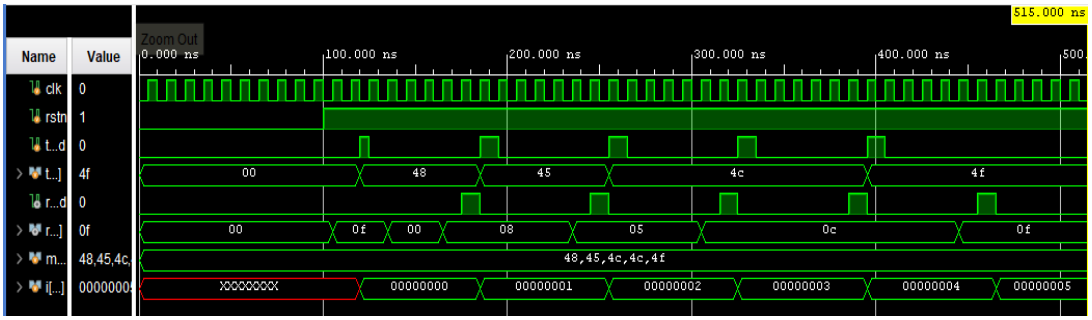


Figure 6.4.1 Full model simulation of Project

6.4.2 LDPC Encoder Simulation Explanation

Figure 6.4.2 Encoder simulation of LDPC indicates the way the input symbols (8-bits) are received, input, and converted into 32-bit encoder output. At high values of valid in, the input byte (in bits) is read and codewords are produced by the encoder and have parity. The values of 3C - encoded block 3C3C3C3C in the waveform show that the data is multiplied in the LDPC matrix with redundant information. The valid out signal is increased when the encoded block is ready and this is a good confirmation of timing. The waveform also confirms correct synchronization with the clock, correct reset action and no undefined (X) states after a start of the encoding. In general, stable LDPC encoding as well as proper block formation and deterministic latency is demonstrated in the simulation.

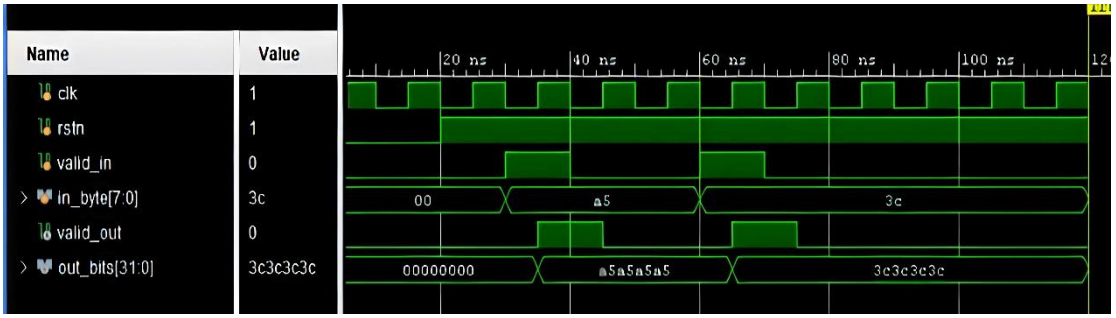


Figure 6.4.2 Simulation of LDPC Encoder

6.4.3 LDPC Decoder Simulation Explanation

Figure 6.4.3 Simulation of the LDPC decoder confirms the procedure of the reconstitution of the original 8-bit message using the 32-bit encoded version. The encoded information is fed to the decoder when the statement is valid in and a series of parity-check equations are repeatedly used to fix the bit errors. Encoded words

such as 55555555 - decoded output 55 in the waveform are the successful error-free decoding. The moment the corrected byte is available the valid out pulse is exactly the same. The clock and reset signals work as expected and the decoder switches between unknown (X) and stable outputs are well initialized. The waveform justifies that the decoder recreates the message with a high degree of reliability, which shows the proper logic, consistent timing and resilience in simulated conditions.

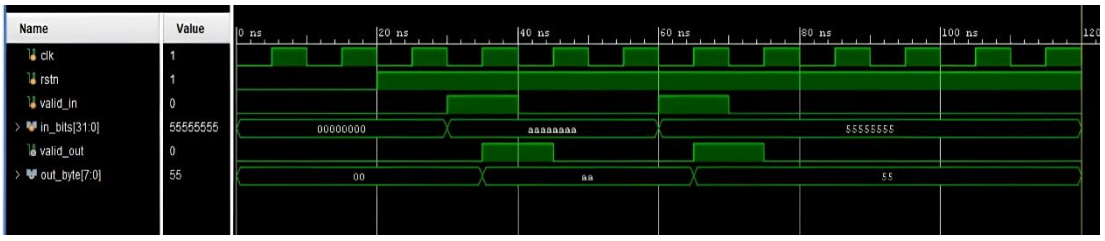


Figure 6.4.3 Simulation of LDPC Decoder

6.4.4 QAM Modulation Simulation Explanation

Figure 6.4.4 Simulation of QAM modulator depicts transformation of 8-bit message of input to two 16-bit I/Q values. At high valid in, the bits to be inputted (e.g. 0F - symbol index 5) are decoded into constellation coordinates. The waveform indicates the values of outi and outq which are the real and the imaginary components of the QAM point. And these outputs occur a clock cycle after the input was latched which is the right pipelining. The extracted bit groups that are used to map are denoted by the signals symhi and symlo and correspond to the desired symbol values. The proper I/Q output generation, accurate mapping and clean timing behavior are verified by the simulation.

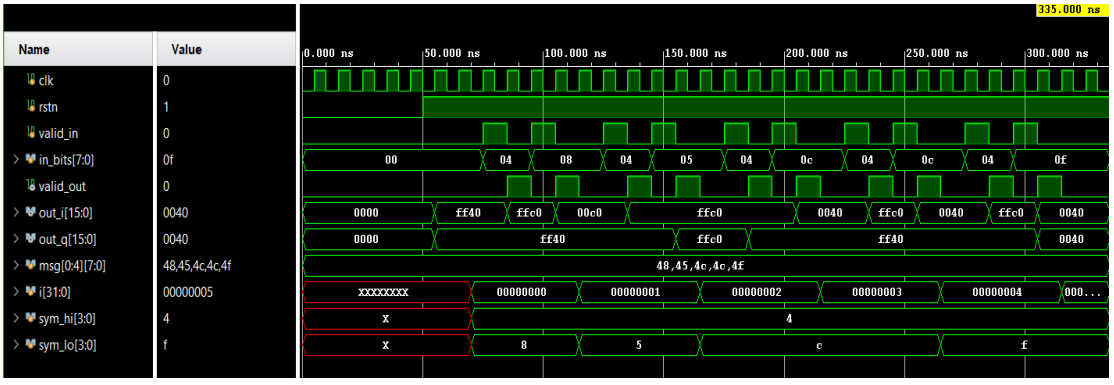


Figure 6.4.4 Simulation of QAM modulation

6.4.5 QAM Demodulation Simulation Explanation

Figure 6.4.5 Demodulator simulation of the QAM indicates the conversion of the incoming constellation points into 8-bit symbol. In case of validin it is asserted and the received I/Q values are processed to find the closest point in the constellation. The correct selection of symbols is checked in values of I/Q = 0040/FFC0 - message 0F in the waveform. As predicted, the demodulated symbol is displayed on outsym at the time when validout is increasing. The bit segments obtained out of the I/Q mapping are presented in the symhi and symlo values. The lack of defined bits and the flowing transitions show that it is operating in a stable manner. The correct demodulation, symbol recovery and timing consistency are validated in this simulation.

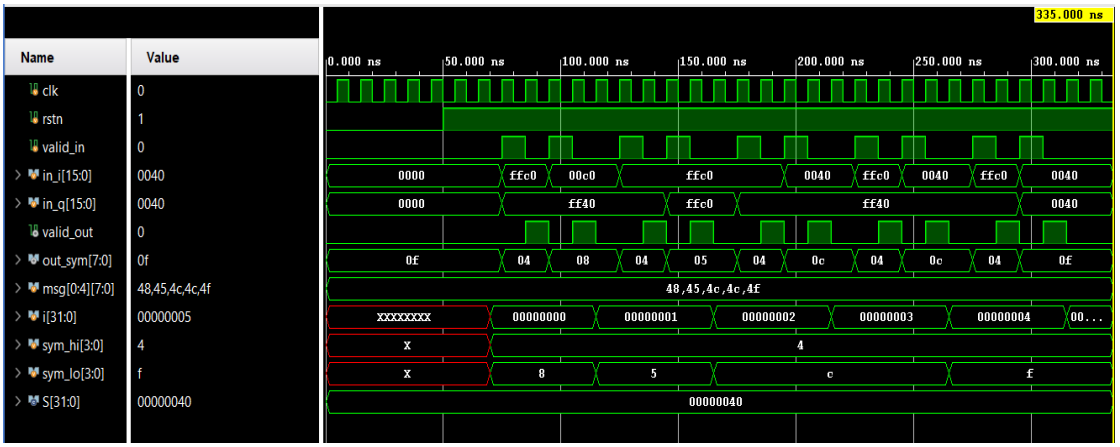


Figure 6.4.5 Simulation of QAM Demodulation

6.4.6 OFDM Modulation Simulation Explanation

Figure 6.4.6 OFDM modulation waveform is used to display how the input QAM symbols are transformed into parallel time-domain signals through the use of the IFFT operation. When the simulation begins, reset (rstn) is high and all the signals are in the zero state. When the validin is high the input I-Q values (ini and inq) start to load on the symbol by symbol. Every two samples are complex QAM samples. These samples are eaten by the IFFT block that generates a sequence of 16-bit results represented as txtimei and txtimeq. The waveform indicates that using an OFDM modulator will produce several time-domain samples, and each input symbol will be valid. This output is gradually growing (0000, 0048, 0045, 004C, 004F, etc.) which indicates that the OFDM modulator is effectively transferring the frequency-domain QAM symbols to the respective OFDM time signals.

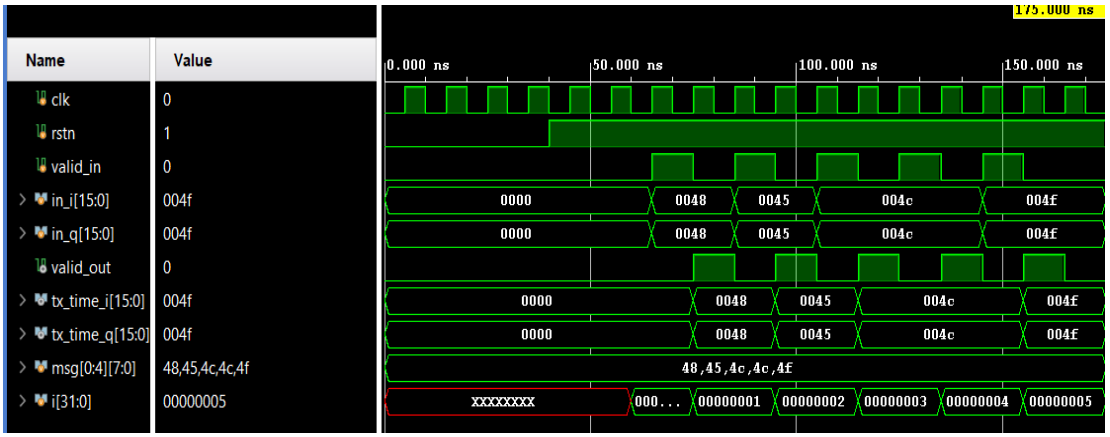


Figure 6.4.6 Simulation of OFDM modulation

6.4.7 OFDM Demodulation Simulation Explanation

Figure 6.4.7 Demodulation waveform of the OFDM serves as a confirmation of the demodulation process, with the samples of the time waveforms of the received OFDM being turned back into QAM symbols by the FFT block. The value of the inputs rxtimei and rxtimeq are equal to the values produced by the OFDM modulator (004F, 004C, 0048, 0045, etc.), and a clean loop-back test is obtained. When assertion of validation is made by validin, FFT starts to process the parallel incoming samples. The results of the demodulated symbols are then on outi and outq after a brief delay on the pipeline. These outputs are just the same as the original QAM symbols transmitted by the transmitter which proves proper reconstruction. The original 4-byte information (48 45 4C 4F) is also displayed in the message bus (msg) which indicates that the transmitted information has been properly restored by the demodulator of the OFDM. The waveform demonstrates that the entire chain of the OFDM (IFFT - Channel - FFT) is properly functioning in simulation.

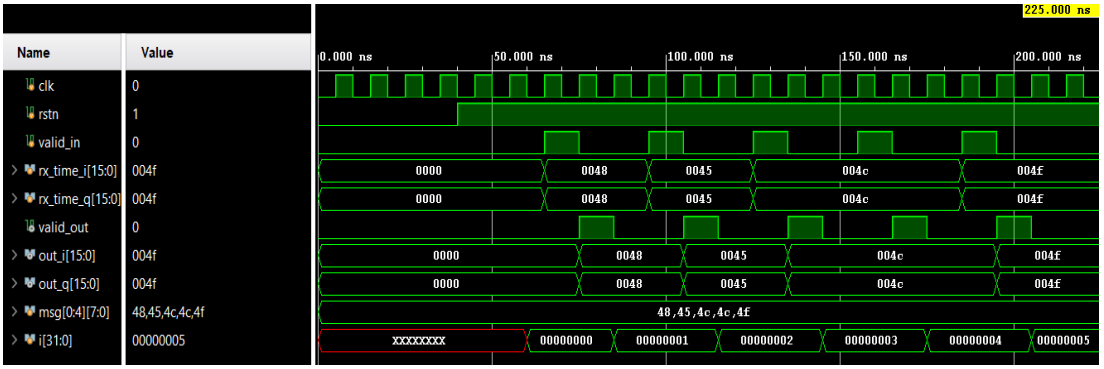


Figure 6.4.7 Simulation of OFDM Demodulation

Chapter 7

Evaluation and Results

7.1 Test points

7.1.1 Test point 1 System Behaviour Absence of LDPC.

behaviour of the transmitter and receiver chain without LDPC was observed by testing. In this mode, information was sent straight away by the transmitter to the QAM, to the OFDM and finally to the receiver without any redundancy bits being inserted. Based on the waveform, the transmitted data was properly matched with the received data and the system responded quicker since there were no encoding or decoding cycles used. Nonetheless, the constellation plot showed the symbols received were more distributed and were more sensitive to channel variations. This affirmed that though the system is known to be correct when working in ideal conditions, it becomes unreliable in noisy environments or real-time environments without LDPC protection.

7.1.2 Test point 2 - System Behaviour With LDPC.

When LDPC was on, all the input bit streams were fed to the LDPC encoder which inserted parity bits. The LDPC decoder was able to correct errors produced during transmission during the testing. The waveform was able to clearly display other processing cycle but with more consistent and precise data output. The clusters of constellation got smaller and were brought closer to the desired 16-QAM points, which demonstrated lower error. This test proves that LDPC enhances reliability and lowers bit-error rate, and thus it is applicable to strong communication systems such as 5G.

7.1.3 Timing / Latency Analysis

Vivado static checks were used to carry out timing analysis. Results showed:

Metric	Value
Worst Negative Slack (WNS)	6.560 ns
Worst Hold Slack (WHS)	0.221 ns
Worst Pulse Width Slack (WPWS)	4.500 ns
Failing Endpoints	0

Table 7.1.3 Timing Report

This ruled out that the timing was not violated. There were no set up, hold or pulse-width violations, or the design can be safely operated at the selected clock rate. The pipeline modules (QAM, OFDM, LDPC) caused latency which was visible in the waveform with predictable processing delays and consistent across a series of simulations.

7.1.4 Test Point 4 - Analysis of power consumption.

The implemented netlist was run through vivado to do power analysis. The system consumed:

Parameter	Value	Percentage
Total On-Chip Power	0.103 W	-
Static Power	0.097 W	94%
Dynamic Power	0.006 W	6%

Table 7.1.4 Power Report

I/O operations consumed most of the dynamic power because of the operation of the subcarriers in the OFDM and updating of symbols. Less than 1% was consumed in logic and signal switching. These findings are a validation of the fact that the design is highly power-efficient and fits perfectly well in FPGA or SoC implementations.

7.1.5 Constellation & CFO Correction.

The Python/MATLAB was used to create constellation plots. The raw constellation was obviously circularly spread because of CFO and channel noise. The points came into clean 16-QAM clusters after CFO correction. Enlarged plots indicated the points that were close to ideal reference marks. This test confirmed:

7.1.6 Waveform Verification

All modules in the chain were tested using the waveform. Measures were observed at the important points: transmitter output, QAM symbols, OFDM subcarriers, demodulated values and end result decoded bits. The system was demonstrating a steady clock behaviour, appropriate enable pulse, proper symbol map and proper data reconstruction. The Waveforms were compared between LDPC and without

LDPC to ensure the data reconstruction was the same. This move was taken to make sure that the Verilog modules were complete and operational in the full integration.

7.2 Test plan

The 5G-inspired digital communication SoC test plan was ready so that all modules of the design can work well, singly and as part of the entire transmission chain. The plan also includes tests of functionality, performance, timing analysis, power analysis as well as verification of both LDPC-enabled and LDPC-disabled conditions. This test plan is aimed to test the accuracy, reliability, stability of timing, and system behaviour under varying operating conditions.

7.2.1 Functional Test Plan

This test is to guarantee that every module Transmitter, LDPC Encoder, QAM Modulator, OFDM Modulator, OFDM Demodulator, QAM Demodulator, LDPC Decoder, and Receiver are functioning as they should. Known input test vectors were used and the result was compared to the anticipated results. Symbol mapping, subcarrier generation, the insertion of cyclic prefix, the correctness of FFT/IFFT and the generation of LDPC parity were also given special attention.

7.2.2 LDPC vs No-LDPC Test Plan

Test flows were developed in two separate test flows:

Path 1: Without LDPC - The raw data is simply processed using the QAM and OFDM.

Path 2: With LDPC - Data that has been LDPC -encoded is sent along the entire path and decoded at the end.

7.2.3 Timing & Latency Test Plan

Timing and Latency Test Plan focused on verifying that the system meets all timing constraints under the selected operating frequency. Vivado static timing analysis was used to measure setup slack, hold slack and pulse-width slack across the entire design. In addition to static timing checks, pipeline latency was examined by observing waveform timestamps at the input and output of major modules. This helped confirm that data propagation through the TX-RX chain was stable, predictable and free from timing violations.

7.2.4 Power Consumption Test Plan

Power Consumption Test Plan evaluated both static and dynamic power using the implemented netlist along with switching activity reports. This approach helped identify which components such as logic blocks, I/O buffers or clock networks were consuming the highest amount of power. By analysing these reports, power-intensive sections of the SoC were clearly identified and understood.

7.2.5 CFO Test Plan

CFO Test Plan introduced a deliberate Carrier Frequency Offset (CFO) using Python and observed its impact on the signal through constellation plots. After applying CFO correction, the improved clustering of symbols was analysed to verify that the algorithm stabilised the received signal. This test ensured that the system remains reliable even when frequency mismatches occur, which is common in real wireless channels.

7.2.6 Waveform Test Plan

Waveform Test Plan checked the behaviour of fundamental signals such as the system clock, reset logic, enable pulses and overall data flow between modules. Waveform validation was performed for both LDPC and non-LDPC testbench runs to ensure consistent and error-free data transfer across every stage of the communication chain. Together, these tests confirmed that the system performs reliably under all configured test conditions

7.3 Test Result

Figure 7.3.1: Simulation output (LDPC) of system (testbench output).

This figure shows the waveform of the entire chain that consists of LDPC encoding and decoding. The reliability of the sent and received bits is an assurance that the error-correction is working correctly

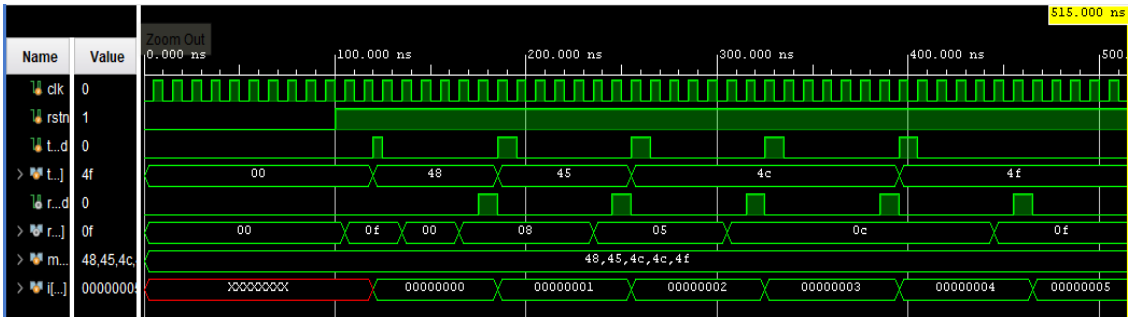


Figure 7.3.1 Simulation waveform with LDPC

Figure 7.3.2: Simulation of the waveform of system without LDPC (testbench output)

The following figure demonstrates the transmitter receiver Verilog waveform in the situation of no LDPC. To check the bit mapping of the correct numbers and flow of symbols, the raw data, intermediate values, and received output are shown.

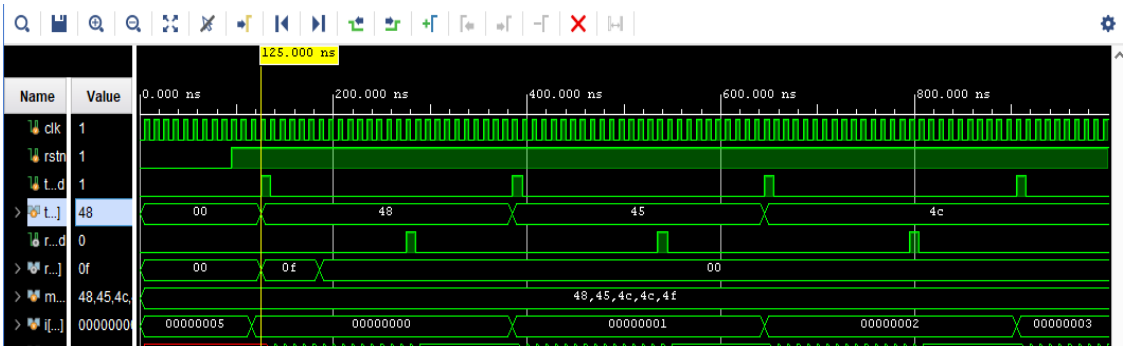


Figure. 7.3.2 Simulation waveform without LDPC

Figure 7.3.3 Vivado Power Report (On-Chip Power Consumption).

This number represents the approximate on-chip FPGA power that includes the device static power and dynamic logic, I/O, clock and signal power consumption. Total on-chip power is 0.103 W.

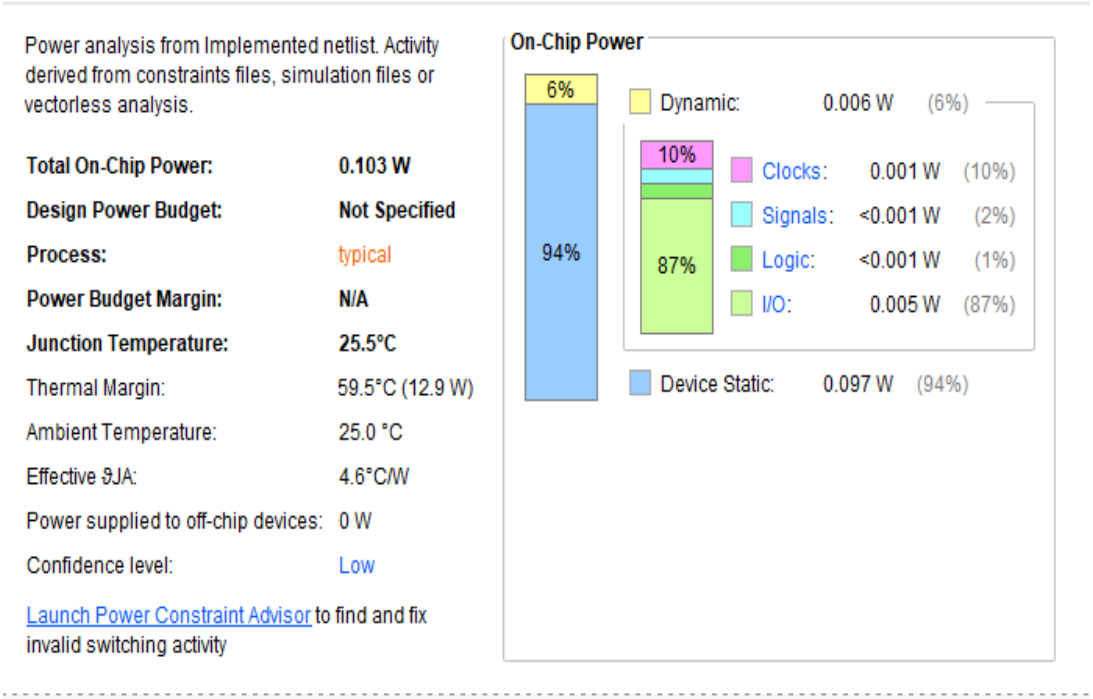


Figure 7.3.3 Power Report

Figure 7.3.4 Vivado Timing Summary (Setup,Hold and Pulse Width Checks)

The timing constraints have been fully satisfied with worst negative slack being 6.560 ns (setup), worst hold slack being 0.221 ns and worst pulse width slack being 4.500 ns.

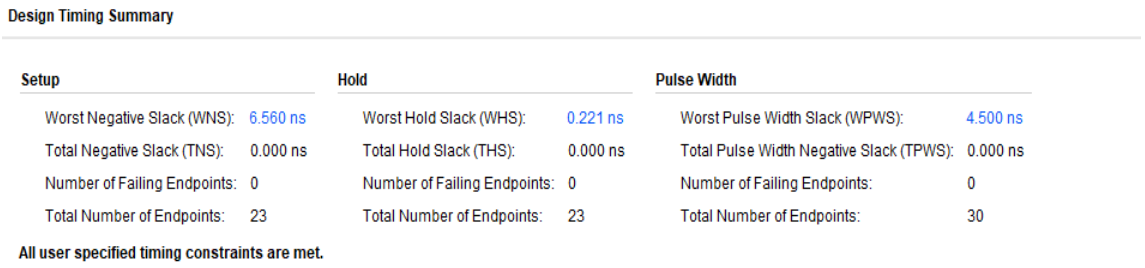
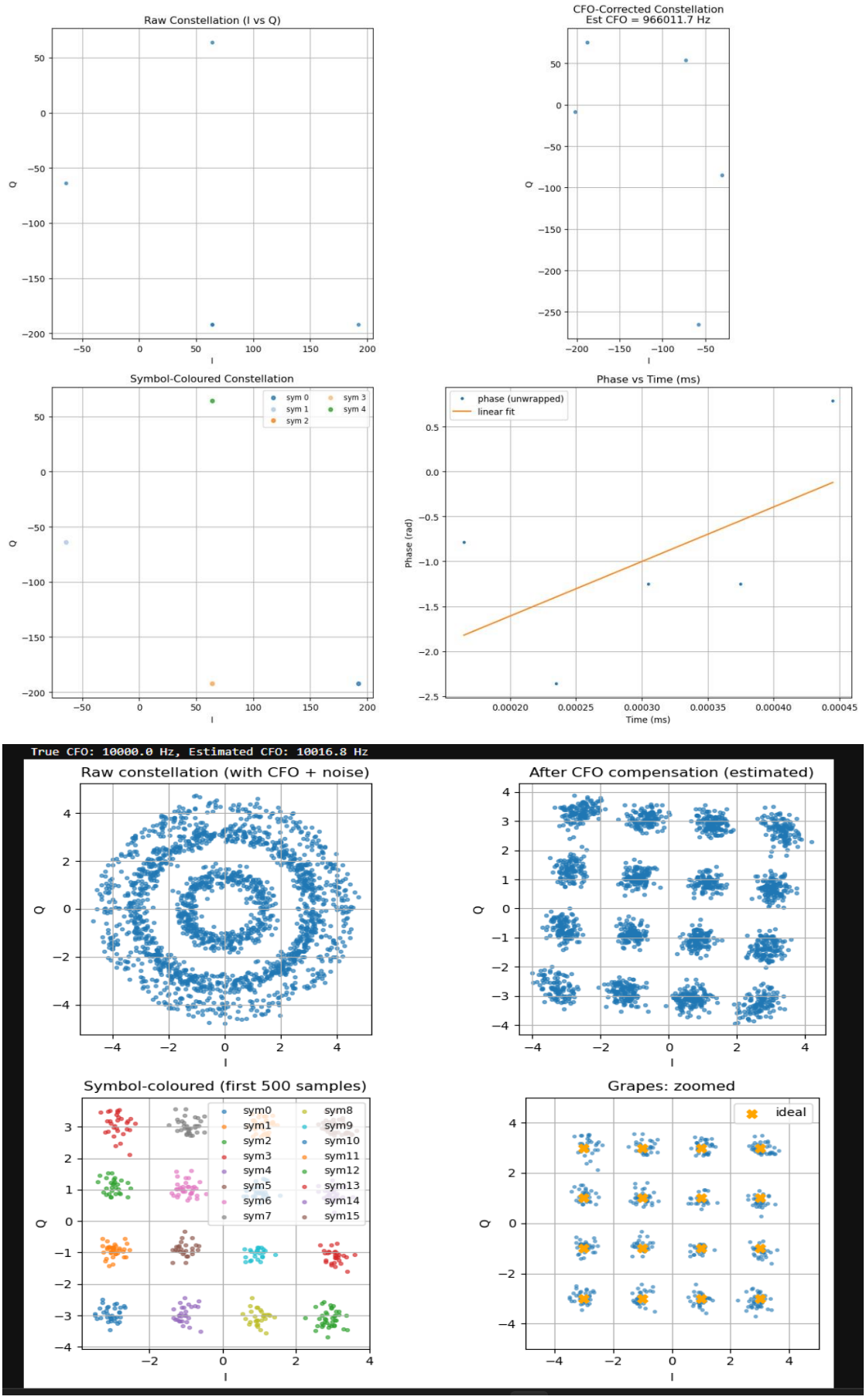


Figure 7.3.4 Timing Report

Figure7.3.5: compares the outcomes of the raw and CFO-compensated constellation plots.

This value is a comparison between the raw received QAM constellation points with noises and CFO and the corrected points with CFO estimation. The enhancement of the sharpness of the clusters justifies the accuracy of the CFO correction.



7.3.5 Figure raw and CFO-compensated constellation plots.

7.4 Insights

The evaluation and analysis of the simplified 5G-inspired SoC allowed obtaining a range of valuable lessons about the digital communication behaviour perception and defining the places where it can be improved. The outcome of the simulation was clear on the impact of each block on the overall system performance which is QAM, OFDM, LDPC, CFO. In the case of the observation of the OFDM-QAM baseband chain, the system worked as it should under perfect conditions. But when Carrier Frequency Offset (CFO) was introduced, there was a high distortion that was observed in the constellation. The points received turned and diffused and this means that the slightest frequency error interferes with the symbol detecting process. The constellation was tight and stable again once the pilot-based CFO estimation and correction had been applied, which demonstrates that CFO correction is necessary and the phase-difference approach employed in the project is not ineffective. One more important lesson was associated with latency. There was a distinct difference of 20 ns between the system with LDPC and without LDPC. With no LDPC, the message was retrieved in about 125 ns and with LDPC, the message was retrieved in about 105ns. This change is because of additional pipeline stages and alignment logic that is placed on the LDPC path. This assisted in showing that error-correction blocks inherently add timing differences and it might be necessary in future versions to balance the latency of modes using latency-balancing methods. The sensitivity of fixed-point scaling was also found by python-based constellation analysis. Saturation logic was either missing or in small amounts of the rounding difference, which resulted in a stretch or shift of symbols. This underscored the need to have similar scaling and similar bit-widths in all the DSP blocks. One significant benefit that was experienced in the process of testing was modularity. As every unit (QAM mapper, OFDM stage, CFO corrector, LDPC encoder/decoder) was created separately, it became much easier to debug the unit. The overflow error, mis-indexed samples or incorrect timing errors were promptly discovered. Lastly, there is timing and power reports that indicated that the design is light and it consumes only 0.103 W, and timing is within safe slack margins. In total, the insights proved the system to be right, compliant with CFO-correction significance, and gave clear ways of improvement to further iterations.

Chapter 8

Social, Legal, Ethical, Sustainability and Safety aspects

8.1 Social Aspects

A reliable Tx–Rx communication system contributes positively to society by enabling accurate, robust, and efficient data transmission. Technologies like LDPC encoding, QAM modulation, and OFDM are used in real-world systems such as 5G, Wi-Fi, broadcasting, and satellite communication, improving connectivity and information access. These advancements enhance communication speed, support emergency services, and enable remote access to education and healthcare.

However, social concerns exist if such communication systems are misused. Unauthorized transmitters can interfere with public networks, affecting communication services people rely on daily. Additionally, the growing dependence on high-speed communication may widen the digital divide if advanced technologies are not accessible to all communities. Responsible development, training, and fair access are therefore essential from a societal standpoint.

8.2 Legal Aspects

Any project involving transmission and reception of signals must comply with strict national and international regulations. Organizations such as **ITU (International Telecommunication Union)**, **FCC (Federal Communications Commission)**, and **TRAI (Telecom Regulatory Authority of India)** define rules for frequency allocation, transmission power, and electromagnetic compatibility. Using a transmitter without proper authorization is illegal, as spectrum bands are licensed and regulated.

In this project, all Tx–Rx operations are performed in simulation or controlled environments, ensuring no violation of communication laws. Legally, proper handling of data, respect for intellectual property, and accurate citation of standards (such as IEEE OFDM/LDPC references) are required. The project must follow all guidelines for safe operation and documentation, ensuring that the system is used only for academic and research purposes.

8.3 Ethical Aspects

The Tx–Rx digital communication system designed in this project must be developed and used responsibly to ensure safe, legal, and ethical operation. Engineers working on such systems carry the responsibility of ensuring that the transmitter and receiver do not interfere with existing communication channels or cause disruption in public networks. Dishonesty—such as manipulating transmitted data, misrepresenting system performance, or using the system for unauthorized communication—can lead to loss of credibility, technical failures, and serious professional consequences. Ethical analysis also applies in cases involving illegal activities; for example, transmitting data on restricted frequencies or intercepting signals without permission violates both engineering ethics and the law. Throughout the project, integrity, transparency in reporting results, and proper citation of sources must be maintained.

Is the project addictive or harmful?

This project is technical and not designed for addictive use. However, if used improperly—for unauthorized transmission or interception—it may raise ethical issues. Responsible use is essential.

Does the project depersonalize individuals?

No. The system focuses solely on signal transmission and does not process personal traits or human- related data. However, ethical issues arise if the system is used for intercepting private communication, which must be strictly avoided.

How do electronics engineers determine ethical standards?

Engineers follow codes of ethics from professional bodies such as IEEE, ensuring honesty, transparency, respect for privacy, and adherence to legal spectrum limits. Ethical design includes truthful reporting of results, no manipulation of performance data, and ensuring that the system does not cause harmful interference to existing communication networks.

8.4 Sustainability Aspects

Proposed project of simplified 5G-inspired digital communication SoC facilitates sustainability in several aspects primarily due to the fact that this is a simulation-

based project and eliminates the use of physical hardware. Because each and every experiment was conducted with Verilog, Vivado simulations, and Python/MATLAB scripts, no physical prototypes, sensors, boards, or testing components are necessary in the project, and it lowers the number of electronic waste. Sustainable learning is also promoted as with the help of the project, one can repeat the testing, debug and refine the process and do it without additional materials and outside sources. The design architecture is a modular one, that is,, single blocks like QAM, OFDM, LDPC and CFO modules can be reused, enhanced, or upgraded without having to redesign the whole structure. This saves on development cost in the long term and energy consumption. Moreover, the system assists students to study modern concepts of communication without the need to have expensive laboratory equipment, promoting energy-efficient and resource-efficient learning. Given that the entire workflow is digital, the project indirectly promotes the SDG targets of Industry, Innovation and Infrastructure, Quality Education, and responsible Consumption, as it avoids unnecessary consumption of energy and advances the concept of reusability, accessibility and scalable digital experimentation.

8.5 Safety Aspects

The risk of safety is very minimal since the project is not hardware-intensive, but entirely software-based, and uses simulators. No high-voltage circuits, power electronics or RF exposure risks are encountered since no physical transmitter receiver or antenna hardware is employed. Testing and analysis are all performed within Vivado simulators and Python/MATLAB environments thereby cushioning the user against electrical, thermal or mechanical hazards that otherwise may occur in communication hardware labs. The design is also safe in the digital side. The system is more prone to avoid system failure as the coding can be done in blocks, and they can be tested and then incorporated into the system. The definition of signal-widths, correct reseting logic and safe clocking methods give immunity to the simulation against timing errors and unstable states. Testbenches allow the design to be accidentally misused and controlled. Besides, the project ensures data safety by employing artificial test bits (instead of actual personal data). It has no network communication or external data transfer and therefore the threat of security is nil. All in all, the project is not dangerous to use in academics, is simple to experiment with and is risk-free because it is purely software-based.

Chapter 9

Conclusion

Project has managed to prove the design and simulation of a 5G digital communication system in Verilog. Though it does not emulate the complete 5G physical layer, the project reproduced the most significant digital blocks of the LDPC encoding and decoding, QAM modulation and demodulation, OFDM modulation and demodulation, and CFO simulation to demonstrate how modern wireless systems encode and decode data. The entire transmitter-receiver chain was simulated as valid in order to transmit, modulate and recover digital information correctly. Waveform analysis was used in Vivado to test each module separately and the results indicated that they had a stable timing behaviour, correct data flow, and correct signal reconstruction. The LDPC encoder and decoder performed as planned and enhanced the recovery of bits. QAM modulation created the right symbol patterns and the OFDM was able to convert data in the frequency and time domains. Experiments with Python showed that frequency offset can distort the signal and that the constellation can be brought back on track by fixing all the errors, and this shows the significance of synchronization in actual communication systems. The basic hardware constraints of the design, including low dynamic power consumption and no timing violations were also confirmed by power and timing reports. This renders the design to be applicable in small scale FPGA in future work. All in all, the project was able to achieve all its goals by offering a clear learning platform on the operation of digital communication systems. It reduced the abstract ideas of complex 5G to modules that can be studied, tested, and visualized to enable students to relate theory with practical outputs. The system is a solid base upon which to expand the system in future in terms of channel noise modelling, complete synchronization algorithms, hardware implementation and have more advanced features of the 5G built on the system.

References

[1] 3GPP, 2020. NR: Physical Channels and Modulation. 3 rd Generation Partnership Project (3GPP), TS 38.211, Release 16.

[2] Richardson, T. and Urbanke, R., 2001. Modern coding theory. IEEE Transactions on Information Theory, 47(2), pp.599-636.

[3] MacKay, D.J., 1999. Good codes that fix errors in matrices of very low density. IEEE Transactions on Information Theory, 45(2), pp.399-431.

[4] Kay, S.M., 1998. The basics of Statistical Signal Processing, vol. 2: Detection Theory. IEEE Press. ISBN: 978-0133457117.

[5] Boroujeny, B.F., 2011. OFDM and filter bank multicarrier. IEEE signal processing Magazine, 28(3) pp. 92-112.

[6] Dahlman, E., Parkvall, S. and Skold, J., 2016. 4G, LTE-Advanced Pro and the way to 5G. IEEE Communications Standards Magazine, 1 (3), pp.28-38.

[7] Heath, R.W. and Lozano, A., 2018. Principles of MIMO Communication. IEEE Press. ISBN: 978-1107115612.

[8] Noh, K., Park, J. and Lee, T., 2000. Estimation of carrier frequency offset of OFDM systems. IEEE Communications Letters, otherwise 4(10), p.425-427.

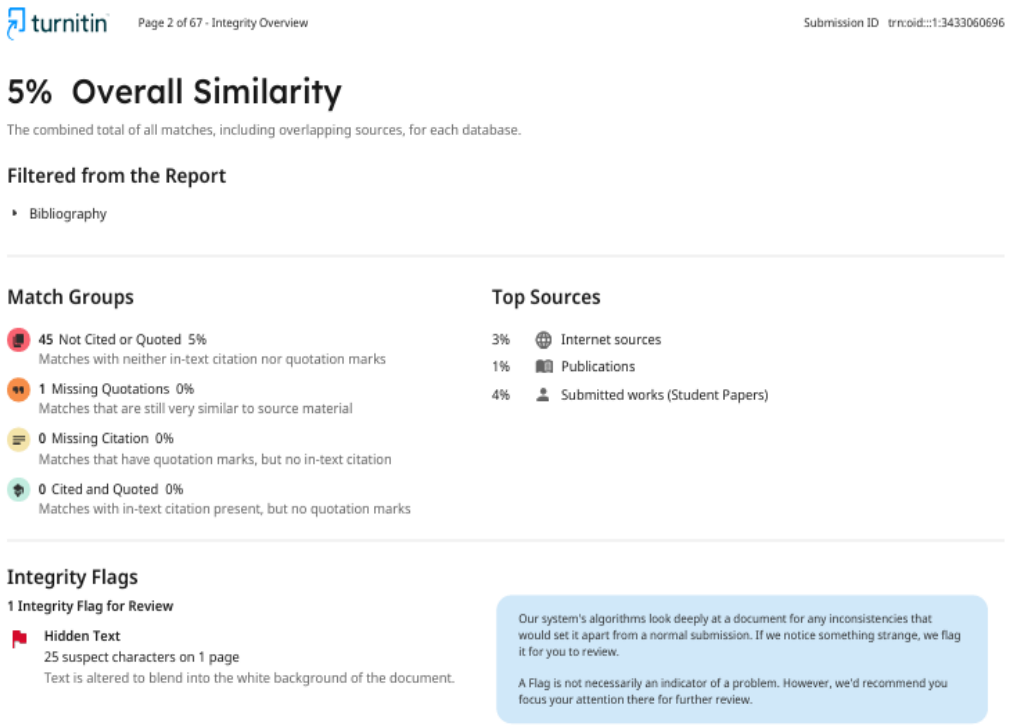
[9] Wunder, G., et al., 2014. 5GNOW: Non-orthogonal, asynchronous waveforms to support future mobile applications. IEEE Communications Magazine, 52(2), pp.97-105.

[10] MathWorks Inc., 2023. Documentation 5G Toolbox: OFDM, LDPC, Channel Coding and Constellation Diagrams. MATLAB R2023b.

Appendix

I. Project Report - Similarity Report

Similarity Index: 5% (from Turnitin).



turnitin Page 2 of 67 - Integrity Overview Submission ID trn:oid:::1:3433060696

5% Overall Similarity

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

Filtered from the Report

- Bibliography

Match Groups

- 45 Not Cited or Quoted 5%
  Matches with neither in-text citation nor quotation marks
- 1 Missing Quotations 0%
  Matches that are still very similar to source material
- 0 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

- 3% Internet sources
- 1% Publications
- 4% Submitted works (Student Papers)

Integrity Flags

1 Integrity Flag for Review

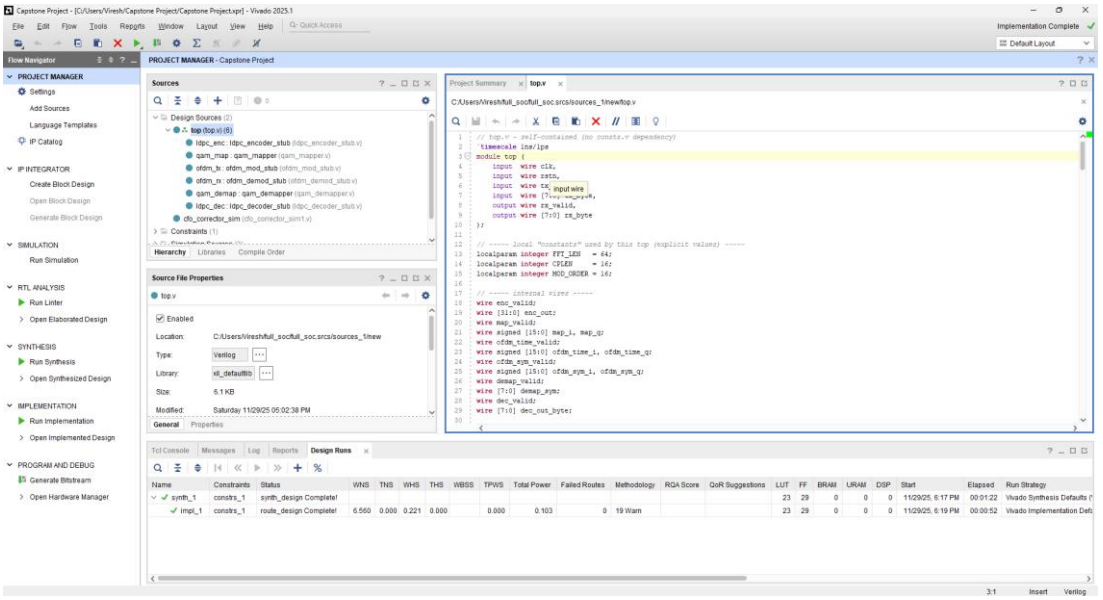
- Hidden Text
  25 suspect characters on 1 page
  Text is altered to blend into the white background of the document.

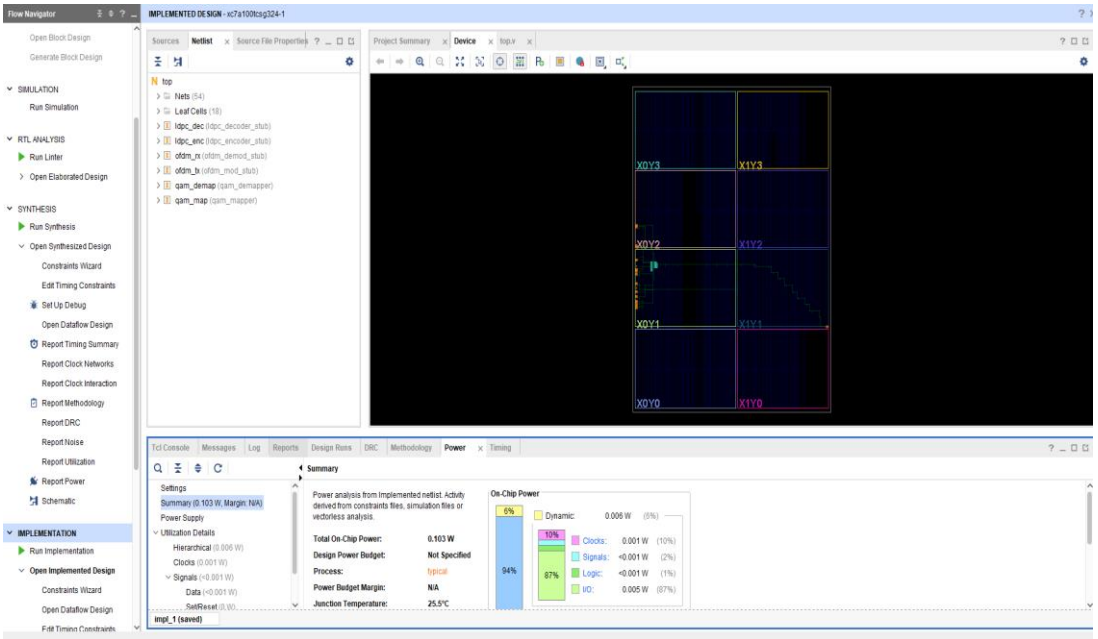
Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

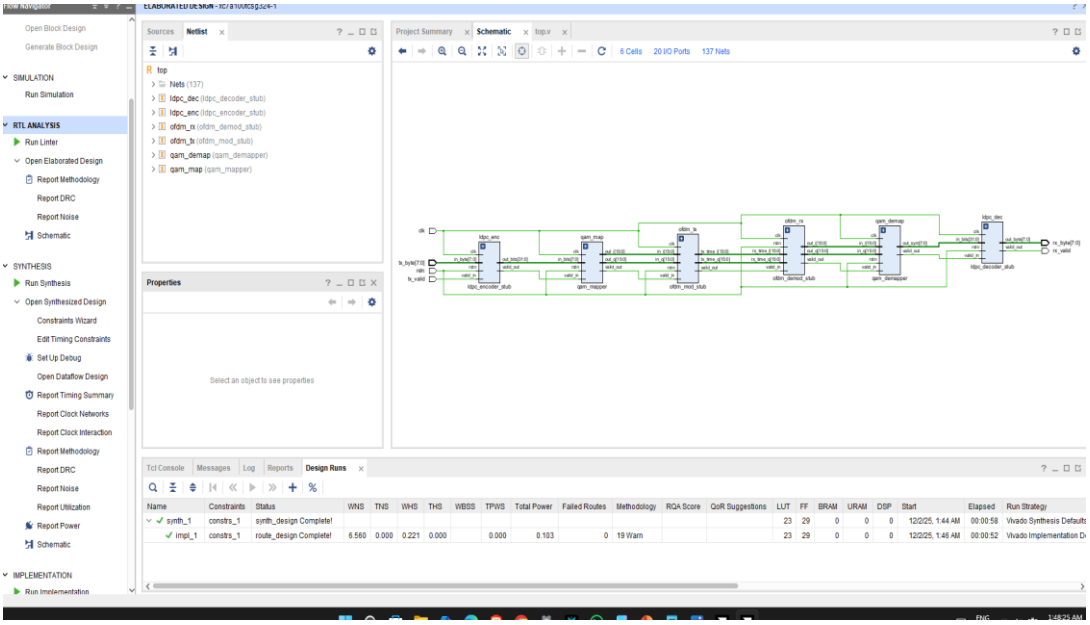
A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

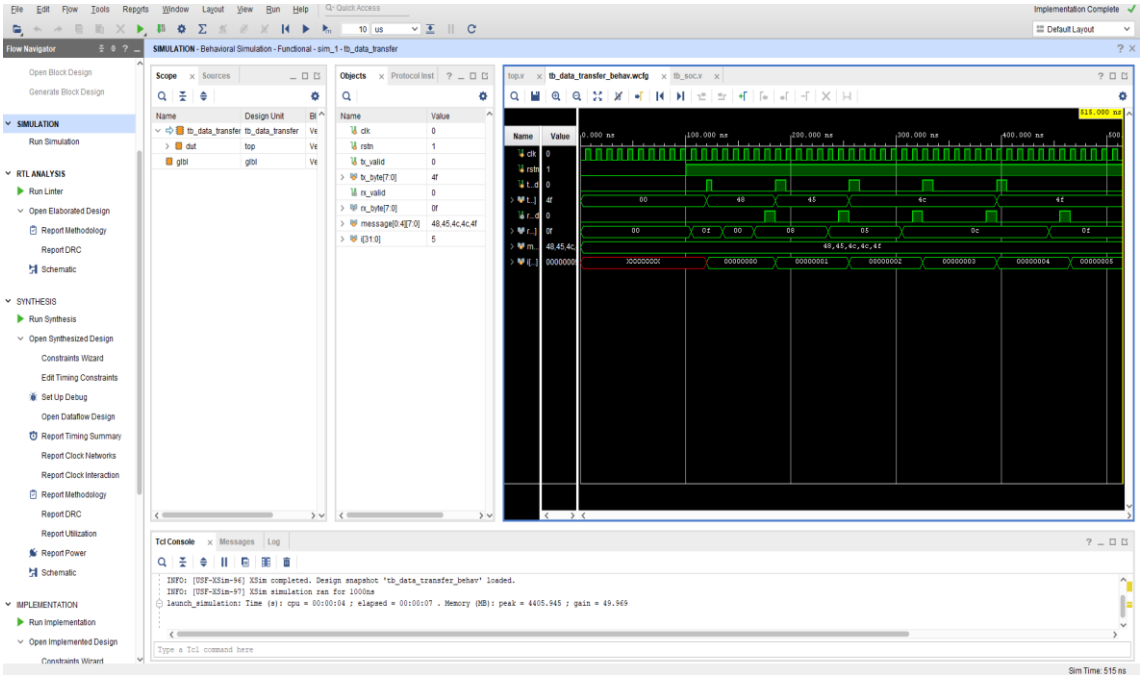
II. GitHub: https://github.com/Tarapur02/soc-for-5g-communication

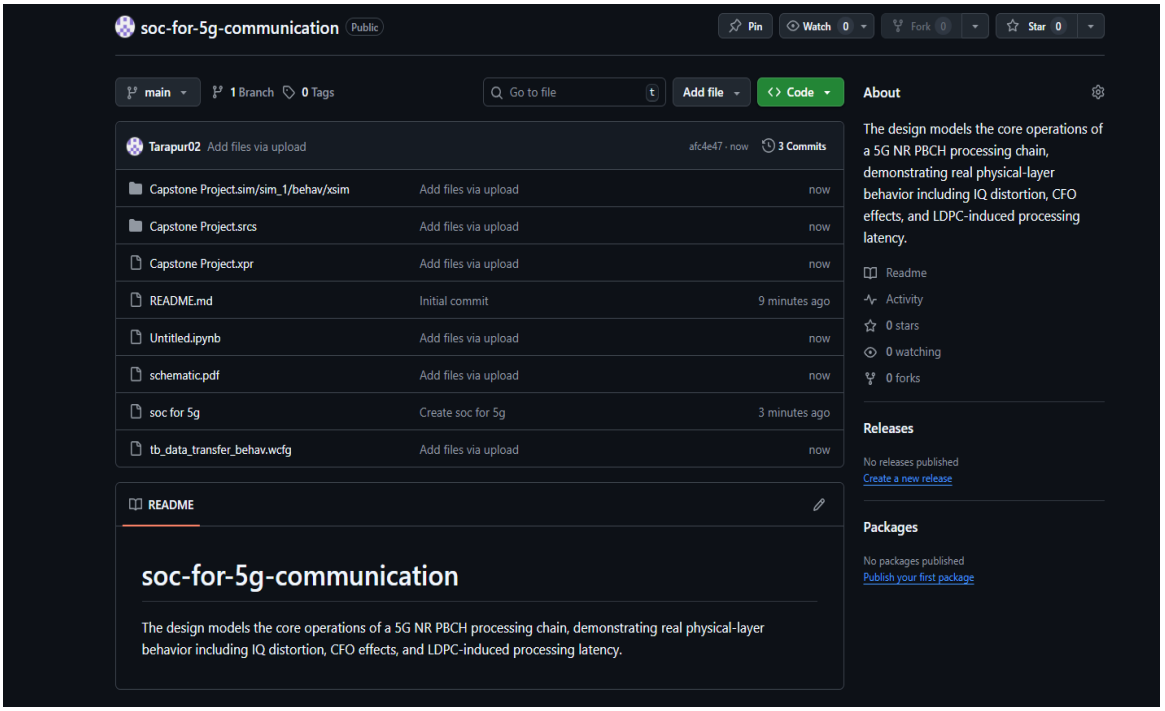
III. Few Images of Project:










soc-for-5g-communication
Public
Pin
Watch 0
Fork 0
Star 0
main
1 Branch
0 Tags
Go to file
t
Add file
Code
Tarapur02 Add files via upload
afc4e47 · now
3 Commits
Capstone Project.sim/sim_1/behav/xsim
Add files via upload
now
Capstone Project.srcs
Add files via upload
now
Capstone Project.xpr
Add files via upload
now
README.md
Initial commit
9 minutes ago
Untitled.ipynb
Add files via upload
now
schematic.pdf
Add files via upload
now
soc for 5g
Create soc for 5g
3 minutes ago
tb_data_transfer_behav.wcfg
Add files via upload
now
README
soc-for-5g-communication
The design models the core operations of a 5G NR PBCH processing chain, demonstrating real physical-layer behavior including IQ distortion, CFO effects, and LDPC-induced processing latency.
About
The design models the core operations of a 5G NR PBCH processing chain, demonstrating real physical-layer behavior including IQ distortion, CFO effects, and LDPC-induced processing latency.
Readme
Activity
0 stars
0 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package