M.Tech. Thesis

Analysis and Implementation of DCT & DHT recursive structures in Verilog

ANANYA VERMA

July, 2025

ANALYSIS AND IMPLEMENTATION OF DCT & DHT RECURSIVE STRUCTURES IN VERILOG

A Thesis Submitted
In Partial Fulfillment of the Requirements
for the Degree of

MASTER OF TECHNOLOGY
In
SIGNAL PROCESSING DIGITAL DESIGN
by

ANANYA VERMA
(2K23/SPD/12)

Under the Supervision of
Prof. Priyanka Jain
Professor, ECE Dept.

Department of Electronics and Communication Engineering

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-110042. India

July, 2025

ANALYSIS AND IMPLEMENTATION OF DCT & DHT RESURSIVE SRUCTURES IN VERILOG

A Thesis Submitted
In Partial Fulfillment of the Requirements
for the Degree of

MASTER OF TECHNOLOGY

In
SIGNAL PROCESSING AND DIGITAL DESIGN
by

ANANYA VERMA
(2K23/SPD/12)

Under the Supervision of
Prof. Priyanka Jain,
Professor ECE Dept.

Department of Electronics and Communication Engineering

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-110042. India

July, 2025

ACKNOWLEDGEMENT

The successful completion of this project would not have been possible without the valuable support and encouragement of several individuals, to whom I express my heartfelt gratitude. I would first like to extend my sincere thanks to **Dr. O. P. Verma**, Head of the Department of Electronics and Communication Engineering at Delhi Technological University, for providing a conducive academic environment and for his consistent encouragement throughout the duration of this work. I am equally grateful to all the faculty members of the department, whose insights and feedback played a significant role in shaping this project. I am deeply thankful to my project supervisor, **Prof. Priyanka Jain**, Professor, Department of Electronics and Communication Engineering, for her unwavering support, expert guidance, and valuable suggestions at every stage of the project. Her encouragement to think innovatively and her continuous mentoring have been instrumental in the successful execution of this work. I would also like to acknowledge the contribution of all those who have assisted me, directly or indirectly, during this journey—be it through technical advice, motivational support, or sharing helpful resources. Each gesture of support has been deeply appreciated. Lastly, I extend my deepest gratitude to my parents and well-wishers. Their constant encouragement, patience, and unconditional support have been my greatest source of strength throughout this endeavour.

Ananya Verma
2K23/SPD/12

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-110042.
India

CANDIDATE'S DECLARATION

I, **Ananya Verma**, bearing Roll No. **2K23/SPD/12**, hereby declare that the work presented in this thesis titled **"Analysis and Implementation of DCT & DHT Recursive Structure in Verilog"**, submitted in partial fulfilment of the requirements for the degree of Master of Technology in Signal Processing and Digital Design, is the result of my own efforts and original research. This work was carried out under the guidance of **Prof. Priyanka Jain**, Department of Electronics and Communication Engineering, Delhi Technological University, during the academic period from August 2024 to July 2025. I further certify that this thesis has not been submitted, either in part or in full, towards the award of any other degree or diploma from this or any other institute.

Candidate's Signature

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-110042.
India

CERTIFICATE BY THE SUPERVISOR

This is to certify that the student bearing Roll No. **2K23/SPD/12** has successfully completed the research work presented in this thesis titled **"Analysis and Implementation of DCT & DHT Recursive Structure in Verilog"**, submitted in partial fulfilment of the requirements for the award of the Master of Technology degree in Signal Processing and Digital Design from the Department of Electronics and Communication Engineering, Delhi Technological University, Delhi.

The thesis is a record of original research work carried out independently by the student under my supervision. To the best of my knowledge, the work embodied in this thesis has not been submitted, either wholly or in part, for the award of any other degree or diploma at this or any other university/institution.

Prof. Priyanka Jain
Professor
DTU, New Delhi

ABSTRACT

The paper presents the implementation and analysis of a recursive Discrete Cosine Transform (DCT) and Discrete Hartley Transform (DHT) using Verilog. Knowing that there has been a growing demand for high speed and resource efficient systems there is a need for implementations that are both fast and lightweight, so the proposed work aims to design a hardware optimized version of the DCT and DHT using the recursive algorithm with an IIR filter based structure to improve performance and reduce complexity. DCT computations are mainly used in applications like signal processing and image compression, thus the proposed design aims to enhance the speed and resource efficiency of DCT computations. Earlier implementations on Verilog have been done for $N = 8$, the proposed research work moves forward with the implementation of $N = 2^r$, thus the verilog module are designed in a generalized manner. While, in the implementation and analysis of a recursive Discrete Hartley Transform (DHT) algorithm for input sequences of length N, where N = 2m and m >=2 is done Verilog. The simulation and synthesis results from MATLAB Simulink validate the correctness of the modules and functions applied and the performance of the design.

The advantages of the proposed recursive structure architecture are that it reduces the number of adders and multipliers required for DCT and DHT computations. In traditional methods, the number of operations is dependent on the value of the input sequence length N. Whereas, the proposed recursive structure helps to maintain a fixed number of multipliers and adders regardless of N. This makes the structure more efficient, and this is achieved by reusing intermediate results and processing data in smaller manageable groups. This also helps in reducing the truncation errors, leading to higher accuracy in computations. This methodology is implemented using Verilog to demonstrate how much feasible the recursive architecture is. This makes it suitable for real-time applications on physical hardware structures. This research contributes to the ongoing efforts to optimize the DCT and DHT computations and thus it paves the way for more efficient compression and processing technologies. The proposed architecture suits real-time signal processing systems such as radar communication and imaging. The proposed methodology is applied in Verilog to check its feasibility. The results show that the structure is efficient in computing the DCT and DHT coefficient with minimum input-output ports, hardware complexity, and power consumption which are highly useful in parallel VLSI implementations.

LIST OF PUBLICATIONS

International Conferences:

[1] **Ananya Verma, Priyanka Jain** “Analysis and Implementation of DCT Recursive Structure in Verilog”, *International Conference on Advancement in Communication and Computing Technology, Bangalore, India, 2025 (INOACC)* – **Accepted,** to be published in August 2025.

[2] **Ananya Verma, Priyanka Jain “**Analysis and Implementation of Discrete Hartley Transform using Recursive Algorithm in Verilog”, *4th International Conference on Smart Generation Computing, Communication and Networking (SMARTGEN), Bali, Indonesia* – **Accepted & Registered,** publication forthcoming.

Table of Contents

TITLE

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

WORD		ABBREVIATIONS
1.	Fast Fourier Transform	FFT
2.	Discrete Fourier Transform	DFT
3.	Discrete Cosine Transform	DCT
4.	Discrete Hartley Transform	DHT
5.	Field Programmable Gate Array	FPGA
6.	Application Specific Integrated Circuit	ASIC
7.	Hardware Description Language	HDL
8.	Digital Signal Processing	DSP
9.	Fast Hartley Transform	FHT
10.	Infinite Impulse Response	IIR
11.	Very Large Scale Integrated Circuit	VLSI
12.	Register Transistor Logic	RTL
13.	COordinate Rotation Digital Computer	CORDIC
14.	Look Up Table	LUT
15.	Block Random Access Memory	BRAM
16.	Joint Photographic Experts Group	JPEG
17.	Moving Picture Experts Group	MPEG

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

In the field of signal processing, transforms are important mathematical tools that enable the analysis, manipulation, and interpretation of signals in frequency domain which are not possible in their time domain or spatial domain. Thus, these transforms enable to extract and analyse useful information, reduce complexity, enhance their signal features which helps to perform efficient computations for various applications that ranges from communications and multimedia to biomedical and radar systems. The core feature of these transforms is to convert a signal form from one domain to another, usually from time domain to frequency or another transform domain. By this transform, clearer understanding of the signal's underlying properties is facilitated. For instance, any complex time-domain signal may exhibit patterns or frequencies which are not very clear in its original time-domain but becomes more apparent and easier to manipulate once transformed in frequency domain. As a result, transforms are an essential function for applications such as filtering, compression, modulation, noise reduction, and feature extraction in digital signal processing, speech processing or image processing.

Transforms can be broadly categorized based on two key attributes, the first one is the nature of signal that is, whether the signal is continuous or discrete, and the second criteria is the type of mathematical operation which means whether it uses trigonometric, exponential or kernel-based transformations. Continuous transforms are applied to continuous time signals. These signals vary over time and represented by mathematical functions defined over a continuous range. Examples of continuous transforms include the Laplace Transform and the Fourier Transform, both are widely used in system analysis, control systems and analog processing. These transforms help in understanding how systems respond over a continuous range of frequencies or in the complex s-domain. Discrete transforms are foundational in digital system as the process signals using digital logic or software algorithms. Their implementation are often optimized using fast algorithms like the Fast Fourier Transform (FFT), for DFTs and Fast Hartley Transform (FHT) for DHTs, which significantly reduce the number of computations required.

With the concept of hardware description languages (HDLs) such as Verilog and VHDL, transforms can now be implemented directly on digital hardware such as FPGAs and ASICs. This allows for real-time processing with high speed and efficiency which is the essential requirement in systems which require discrete computations in systems like video encoders, communication transceivers, medical imaging devices, and radar systems. Transforms serve as the foundation for advanced techniques and systems where it is used to isolate frequency components in communication signals, compress images and videos by removing perceptual redundant features (e.g., JPEG uses DCT), design filters in digital audio and biomedical devices, detect and analyse features in radar and sonar systems, perform spectral analysis in scientific measurements, etc. Thus, these transforms bridge the gap between raw signals data and meaningful, actionable insights. Transforms provide the mathematical foundation necessary for modern signal processing systems whether in analog or digital form, continuous or discrete form. The key area of ongoing research is understanding and optimizing these transforms, especially in the context of efficient implementation of hardware. This helps in the technical advancement of the mentioned applications.

1.1.1 Discrete Fourier Transform (DFT)

In Digital Signal Processing (DSP), Discrete Fourier Transform (DFT) is a fundamental and widely used tool which converts the discrete time signal from the time domain to frequency domain. The signal's frequency components are essential in many ways and the transformation of signal from time to frequency domain helps in a detailed analysis of the signal/s frequency components. Unlike the continuous Fourier Transform which deals with the continuous-time signals, the DFT is especially designed for sequences of sampled data. This helps where signals are inherently quantized which makes it suitable for digital applications perfectly. The DFT of a sequence of length N is mathematically defined as:

$$X(k) = \sum_{m=0}^{N-1} x(m). W^{mk} \tag{1.1}$$

for k=0,1,.., N-1, where $W = e^{-j2\pi/N}$ is the primitive N$^{\text{th}}$ root of unity.

There is a very high computational cost of DFT, when the value of N is large, this is where Fast Fourier Transform (FFT) algorithm comes in picture. The FFT drastically reduces the number of multiplications and additions required from $O(N^2)$ operations to $O(N \log N)$ that makes real-time frequency analysis feasible in modern digital systems. Despite the strength of DFT, there are certain limitations which include the spectral leakage that arises when the signal does not contain an integer number of periods within the sampling window, limited frequency resolution which depends on the number of samples N and the sampling rate.

1.1.2 Discrete Cosine Transform (DCT)

The discrete cosine transform DCT is another important transform used primarily in signal and image compression techniques as discussed earlier in DFT it is a complex valued transform which results into complex valued coefficients the DCT produces only real valued coefficients which simplifies hardware and software implementations. The DCT is a finite sequence of real data points as a sum of cosine functions oscillating at various frequencies the main property of DCT is its energy compaction property which allows most of the signal's energy to be concentrated in a few low frequency components so wherever a large portion of the high frequency content can be discarded this property makes it ideal for compression of algorithms with minimal loss of signal quality.

The DCT of a sequence of length N is given by:

$$Y(k) = \frac{\sqrt{2}}{N}\, E_k \sum_{n=0}^{N-1} X[n].\cos\left(\frac{(2n+1)k\pi}{2N}\right) \tag{1.2}$$

for k=0,1,.., N-1, where the normalization factor E_k is defined as:

$$E_k = \begin{cases} \frac{\sqrt{1}}{2}, & k = 0 \\ 1, & k \neq 0 \end{cases}$$

The DCT is extensively used in applications like JPEG image compression mpeg video compression speech coding feature extraction in machine learning when the essential feature of a signal is efficiently captured with minimal coefficients the DCT enables more compact and efficient data representation.

1.1.3 Discrete Sine Transform (DST)

The Discrete Sine Transform (DST) is closely related to the DCT but uses sign functions instead of cosine functions as it's basis, this change results in different boundary conditions which makes the DST particularly useful in solving partial differential equations, image reconstructions and specific signal processing scenarios where odd symmetry is assumed or beneficial. This transform also produces real valued outputs and it also possesses energy compaction properties which are similar to that of DCT thus making it a valuable data compression technique. There are multiple variants of DST just like the DCT (DST I, DST II, DST III, and DST IV) which are each tailored according to particular boundary conditions and application needs.

The DST of a sequence of length N is defined as:

$$X(k) = a_k \sum_{n=0}^{N} x[n].\sin\left(\frac{(2n-1)k\pi}{2N}\right) \tag{1.3}$$

for k=1,2.., N, where:

$$a_k = \frac{1}{\sqrt{2}}.\in_k$$

$$\in_k = \begin{cases} \frac{\sqrt{1}}{2}, & k = N \\ 1, & 1 \leq k \leq N-1 \end{cases}$$

DSTs are less commonly used than DCTs but they play an important role in applications such as signal reconstruction with odd symmetry, certain image and video processing tasks and scientific computing and numerical solutions of PDEs.

1.1.4 Discrete Hartley Transform (DHT)

The discrete Hartley transform DHT is known as a powerful alternative to DFT for analyzing the frequency content of real valued signals. It was introduced by R Braswell in 1983. DHT is entirely real valued which simplifies computation and avoids the need for handling complex numbers which becomes an advantage in both software and hardware implementations. DHT combines the sin and cosine components into a single function "*cas*" which is defined as:

$$cas(\theta) \; = \; cos(\theta) \; + \; sin(\theta) \qquad (1.4)$$

The DHT of sequence of length N is given by:

$$X[k] \; = \sum_{n=0}^{N-1} x[n].cas(2\pi nk/N) \qquad (1.5)$$

for k=0,1,.., N-1.

The DHT retains many of the desirable properties of the DFT including the orthogonality invertibility and energy conservation. DHT is more computationally efficient for real valued signals since it avoids complex arithmetic wherever the requirement of tasks such as radar signal processing, biomedical signal analysis, image compression and reconstruction are there. The DHT is well suited because they are real time processing tasks, moreover many fast algorithms developed for the DFT can be adapted to the DHT which makes it a flexible and efficient transform for various real world applications where phase information is secondary to magnitude analysis.

1.2 PROBLEM STATEMENT

There is a growing demand for efficient and real time signal processing techniques in modern digital systems particularly in applications where image processing, radar communication and multimedia processing is concerned. In such type of applications, the transform like the Discrete Cosine Transform (DCT) and Discrete Hartley Transform (DHT) play a fundamental role because of their ability to combat energy and represent signals in the frequency domain with reduced redundancy. Also, both the transform work for real valued signals thus, making it efficient for the application in hardware systems.

The conventional implementations of DCT and DHT especially for variable input lengths $N = 2^r$ often rely on iterative or matrix based approaches which involve a high number of arithmetic operations and significant hardware resources thus, slowing down the whole computation process, because of this these implementations tend to scale poorly with increasing input size that is, the value of N which leads to increased power consumption latency and moreover hardware complexity while all of these problems are detrimental to real time and embedded applications. There is a critical need for a unified and recursive hardware architecture which can perform DCT and DHT computations efficiently across any variable input lengths and minimizing hardware resource usage. Such type of design must be scalable, modular and capable of reusing the intermediate computations to reduce the number of multipliers, adders and memory elements without sacrificing the computational accuracy or throughput. This research aims to address these limitations by developing memory efficient recursive hardware architectures for both DCT and DHT using Verilog HDL. The objective is to achieve a fixed minimal set of computational resources which is independent of the input sequence length N. Thus, the proposed architecture utilizes a preprocessing stage followed by an Infinite Impulse Response (IIR) filter based recursive structure which makes the implementation suitable for high-speed, low power VLSI applications.

1.3 MOTIVATION OF THE RESEARCH

In modern digital systems there is a growing demand for fast and efficient data processing methods due to the exponential rise in multimedia content radar systems and real time communication technologies. From the mathematical backbone of many such applications Discrete Cosine Transform and Discrete Hartley Transform can compress data and analyze frequency components effectively. Implementing these transforms efficiently in hardware remains a challenge despite of their mathematical importance. The existing implementations often demand higher computational resources and do not scale well with increasing data sizes hardware resources especially in FPGA and ASIC are limited and there is a constant need to explore optimized architectures that balance speed accuracy and resource usage additionally there is a growing interest in real time applications such as image and video compression biomedical signal analysis and radar signal processing which motivates the development of architectures that can deliver fast and accurate results with minimal latency. This research is motivated by the need to bridge the gap between algorithmic efficiency and hardware feasibility this work aims to address current inefficiencies and pave the way for scalable real time transform computation by focusing on the recursive architectures and Verilog based hardware implementation of DCT and DHT.

1.4 OBJECTIVE OF THE WORK

The main objective of this research is to analyze design and implement the recursive hardware architecture for the Discrete Cosine Transform and Discrete Hartley Transform using Verilog Hardware Description Language. When compared to the traditional methods the goal is to develop an efficient scalable and real time capable transform modules which consume fewer hardware resources and hand demonstrate improve performance. Specific objectives include to study and understand the theoretical foundations of DCT and DHT and their applications in signal and image processing. To explore recursive algorithms suitable for hardware implementation particularly using IIR filter based structures , to design a generalized and scalable Verilog HDL architecture for DCT and DHT that supports variable input lengths, to implement a preprocessing stage allowed by recursive computation blocks to reduce the number of arithmetic operations, to verify the functional correctness of the proposed architecture through simulation using Verilog test benches, to evaluate and compare the performance in terms of speed hardware utilization latency and power efficiency.

1.5 THESIS ORGANIZATION

This thesis is organized into five chapters, each of which systematically develops the understanding, methodology, implementation, and findings of the research work on recursive and memory efficient hardware architectures for the Discrete Cosine Transform and Discrete Hartley Transform using Verilog HDL.

Chapter 1: Introduction

This chapter presents the background of signal processing transforms and their importance in various applications. It includes an overview of different transforms such as the DFT, DCT, DST and DHT, followed by the formulation of the problem statement. The motivation for undertaking the research is elaborated and clear objectives are defined to guide the course of the work.

Chapter 2: Literature Review

This chapter discusses the previous works and existing architectures related to DCT and DHT implementations it highlights their limitations reviews classical and modern approaches in terms of computational complexity and hardware efficiencies. It also identifies the key gaps that form the basis for the proposed research.

Chapter 3: Analysis and Implementation of Discrete Cosine Transform in Verilog

This chapter details the theoretical formulation and recursive computation of the DCT the design includes A preprocessing stage and a recursive structure for efficient computation of the DCT coefficients the implementation in Verilog is explained thoroughly which is further followed by the testing methodology, validation metrics, and simulation results. An overview of the developed RTL design is also presented.

Chapter 4: Analysis and Implementation of DHT Recursive Structure in Verilog

This chapter focuses on the recursive architecture of the DHT mathematical background, decomposition of the transform, and the corresponding hardware architecture is covered in this chapter. The Verilog implementation is described along with the preprocessing logic and coefficient generation process. The results include simulation waveforms along with its output streaming behavior and synthesis results are analyzed and discussed.

Chapter 5 Conclusion

The final chapter summarizes the major findings of the work and discusses the contributions made through the recursive and memory optimized Verilog implementations of DCT and DHT it also outlines the scope for future enhancement in terms of scalability performance and real-world application integration.

CHAPTER 2

LITERATURE REVIEW

2.1 EXISTING DCT ARCHITECTURES

The Discrete Cosine Transform is a fundamental tool in digital signal processing, particularly in the fields of image and video compression. This DCT transforms the discrete data points into sum of cos functions helping to analyse the signal in terms of its frequency contents. Its efficiency in compacting energy into a small number of coefficients makes it a preferred choice in compression algorithms like JPEG and MPEG. Among the various types of DCT, the DCT-II is the most widely used. It is characterised by its ability to represent data with minimal information loss, which is critical for applications requiring high fidelity, such as medical imaging and high-definitions video. The mathematical formulation of DCT-II facilities the separation of signal into frequency components, thereby enhancing compression efficiency and reducing computational complexity. Traditional implementations of the DCT-II algorithm involve direct computation methods and fast Fourier transforms-based techniques. While direct computation is straightforward, it is computationally expansive, making it impractical for real-time applications. FFT-based methods offer improved efficiency but still face challenges in terms of hardware complexity and power consumption, particularly for large data sets. Recursive algorithms have been employed in various signal processing tasks to reduce computational overhead. By reusing intermediate results, recursive methods can significantly cut down on the number of operations required. This is particularly useful in the context of real-time processing, where latency and throughput are critical considerations.

The paper [1] involves several key steps to perform the Discrete Cosine Transform, DCT using efficient VLSI architectures, particularly focusing on systolic arrays. These systolic architectures are a parallel computing architecture which are used to efficiently execute regular repetitive computations, particularly those found in digital signal processing and matrix operations. The overall approach focuses on optimizing the computation of DCT by leveraging existing DFT systolic array architectures and employing a prime factor algorithm to reduce computational complexity and enhance the efficiency of VLSI implementations.

In paper [2], a method of CORDIC based Radix2 DCT algorithm was also introduced in which fast DCT algorithm addressing several limitations of existing methods was explained. This algorithm integrates the advantages of both CORDIC and Radix2 techniques resulting in a solution that will generate higher order DCT, regular data flow, uniform post scaling factor and in-place computation and arithmetic sequence rotation angles. Compared to the traditional DCT algorithms, this method offers low computational complexity, high scalability and modularity, regular and pipelined implementation which are suitable for real-time applications. Thus, the proposed CORDIC based Radix2 fast DCT algorithm represents a significant step forward offering a promising solution for high-speed pipelined VLSI implementations.

In [3], the paper presents a technique based on the fusion of the arithmetic distribution with Loeffler algorithm to overcome the problems such as high silicon occupation area and the number of multipliers to be integrated. It presents the implementation of this algorithm on FPGA Xilinx. As a result, classical integration of one or more multipliers without the loss of data accuracy was eliminated.

Verilog uses a modular approach for mathematical computations and thus it becomes easy for managing the data flow in calculations where feedback is used [4]. In the proposed paper, such techniques are used which helps to perform recursive procedures.

In paper [5], [6], Wang introduced a recursive algorithm realized using the infinite impulse response filter structure used to compute the coefficients of discrete cosine transform. This, as a result was able to use fewer number of computational cycles. The paper presented by Chan was further modified to improve the processing speed using a second order IIR filter structure which was presented in paper [7].This algorithm reduces the problem of requirement of more execution cycles and multipliers. The DCT computation algorithm used in this paper facilitates a structure such that it reduces the hardware complexity.

2.2 EXISTING DHT ARCHITECTURES

There are many previous research and advancements focusing on the hardware efficient realizations for VLSI applications. Few research presents a clear trend towards optimizing the DHT for low-power, high-speed VLSI architectures, through CORDIC/MAC-based computation, systolic arrays and parallel processing, prime -length and general-length support, application-specific adaptations in speech, image and power systems. This review lays the foundation for the proposed work, which focuses on a memory-efficient and recursive DHT architecture using 32-bit scalar registers, aiming to reducing I/O complexity and hardware cost. The transform-based methods are the base of Digital Signal Processing (DSP) applications, which helps to convert the signals from time domain to frequency domain to analyse compression and transmission efficiently. Most of the transforms are complex-valued whereas Discrete Hartley Transform (DHT) [8], [9] has gained importance as an option for real values to Discrete-Fourier Transform (DFT) by which complex arithmetic computations and storage requirements can be avoided. Due to its real valued computation DHT becomes more attractive for area-efficient and low power implementations in VLSI and FPGA systems.

The paper [10], presents a unified VLSI array architecture which is capable of calculating DFT, DCT, DST and DHT using a CORDIC-based processing element. It forms the foundation for versatile transform suitable for applications in data compression and spectrum analysis which results in low-latency and high-throughput performance.

Paper [11] shows its applications in speech processing proposed a hybrid approach which combines DFT and DHT to enhance speech signal by reducing cross-spectral terms. So, the use of Fast Hartley Transform (FHT) improved the memory usage and execution time showcasing the practical edge of DHT in real-time acoustic signal enhancement.

Paper [12] explored DHT-based compression for digital image processing. Subsequent 2-D DHT implementation for an 8x8 matrix demonstrated the transform's utility in image storage systems. Recently, [13] introduces a power system signal compression method using DHT in the smart grid domain. In paper [14], authors leveraged DHT features for acoustic scene classification, underlining the benefit of real arithmetic in achieving low computational complexity. Several works have focused on efficient VLSI implementations of the DHT.

In [15], author introduced a novel cyclic convolution-based architecture, allowing DHT computation for arbitrary-length sequences. [16] proposed systolic array architectures for odd-time generalized DHT and prime-length DHT, respectively ensuring low I/O cost and modular design—key characteristics for scalable VLSI systems. [17] took this further with an ASIC-level implementation of a high-speed DHT processor using parallel addition and MAC-based design outperforming CORDIC, circular convolution, and other traditional methods. In [18], author discussed the replacement of DFT by DHT in VLSI systems where input signals are purely real minimizing the gate-level interconnection overheads with the use of automated HDL-based design flows.

2.3 SUMMARY OF KEY FINDINGS FROM LITERATURE

The proposed method incorporates the algorithm proposed in paper [7] and implement it in Xilinx Vivado using the device specifications product family – Artix 7, project part – 'xc7a200tfbg676-3'. The concept of recursive implementation of DCT-II builds on these principles by dividing the computation into smaller, manageable parts and processing them iteratively. This approach not only reduces the computational load, but also simplifies the hardware design, making it more suitable for implementation on resource-constrained devices such as FPGAs and ASICs.

The Discrete Hartley Transform stands out among other transforms because of it real-valued nature which is suitable in hardware-efficient implementations, particularly in VLSI and FPGA systems. The proposed Verilog-based method aims to take the advantage of the real-valued functionality by developing recursive structure using [19] scalar 32-bits input. This approach gives a more streamlined, modular design that addresses the gaps such as, reducing the memory usage by supporting sequential data streaming, embedding finite state machine control and minimizing I/O port usage. The architecture is implemented in Xilinx Vivado using the device specifications product family – Artix 7, project part –

‘xc7a100tcsg324-3’. The upcoming chapter will explain the module decomposition and Verilog implementation of this DHT system and its resource analysis.

2.4 RESEARCH GAP

There have been extensive research and numerous applications of the discrete cosine transform and discrete Hartley transform in digital signal processing image compression and communication systems despite of this there remain several key challenges and limitations which are still unaddressed in the existing literature. The following research gaps have been identified as the motivation for the thesis:

1. Inefficient hardware utilization in conventional architectures - Most traditional DCT and DHT implementations are based on matrix multiplications your iterative algorithms which require a large number of adders, multipliers and memory elements. This kind of approach consumes significant hardware resources which makes them unsuitable for area and power constraint systems such as embedded or portable devices.

2. What scalability with input size - Many existing hardware architectures are designed for fixed sequence example N is equal to 8 or north is equal to 16 and they do not generalize well for arbitrary lengths N is equal to 2 R in the case of DCT. There are varying data requirements across different applications depending on values of N, so the lack of scalability which means the dependency on the value of N makes the designs rigid and thus they are not adaptable.

3. High latency and power consumption - According to the growing need of low power consumption in modern digital signal processing there arises a need of low computation time and low power consumption. The conventional DCT and DHT designs often involve long combinational paths or complex control logic which increases computation time and power consumption. These factors hinder the feasibility of real time implementation particularly in high-speed applications such as radar or video streaming.

4. Insufficient research on DHT compared to DFT and DCT for hardware implementations - There is a lack of optimized DHT hardware designs particularly those that take advantages of recursive methods. Although DHT offers several advantages over DFT, example real valued computation, simpler arithmetic, it remains relatively underutilized and under researched in hardware implementations.

5. Limited verification on real hardware platforms - many published architectures are only simulated using software tools example MATLAB or ModelSIM without validation on actual hardware platforms. Real-world verification is essential to assess timing performance resource utilization and power metrics.

CHAPTER 3

ANALYSIS AND IMPLEMENTATION OF DISCRETE COSINE TRANSFORM IN VERILOG

3.1 MATHEMATICAL FOUNDATION

The DCT-II algorithm for a sequence of N length is mathematically defined as:

$$X(k) = \frac{\sqrt{2}}{N}\, Ak \sum_{n=1}^{N-1} x(n) \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right] \tag{3.1}$$

Where $k = 0,\ 1,\ (N\text{-}1)$ and,

$$A_k = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0 \\ 1, otherwise \end{cases}$$

Referring to the calculation in [7] it is done in such a way that it involves two stages, firstly the input sequence goes through the preprocessing stage which results in intermediate values $w^p(n)$. These values served as the input to the computational stage which involves the recursive second order DCT structure computation to produce the final output coefficients.

The preprocessing stage ensures that the input data is appropriately prepared for the recursive filter structures optimizing the computation flow. This process can be generalised into the following expression:

$$X(2^p(2i+1)) = X(2i+1), X(2(2i+1))\ldots, X(2^{r-1}(2i+1)) \text{ and } X(0) \tag{3.2}$$

Here, x(n) is the input sequence which is divided as $x^0(n)$, $x^1(n)$, $x^2(n)$, …, $x^p(n)$ in the iterative stages.

3.1.1 Recursive computation of DCT coefficients:

In this algorithm, the first step includes of dividing the DCT coefficients into two groups: odd and even. The odd group has k = odd and p = 0. Even group (k=even) is subdivided into $(r\text{-}1)$ groups. In Eqn 3.2, k is replaced by $2^p(2i+1)$. Here, $p=0,1,2,\ \ldots.\ (r\text{-}1)$ and $i = 0,1,2\ \ldots\ (2^{r-(p+1)}\text{-}1)$, such that

$$X(2^p(2i+1)) = \sum_{n=1}^{N-1} x(n) \cos\left[\, 2^p(2i+1)\alpha i \,\right] \tag{3.3}$$

Where, $$\alpha_i = \frac{2p(2i+1)\pi}{2N} \tag{3.4}$$

By the derivation in paper[7], the final solution for calculating the DCT is given as:

$$G_{ji}^{p}\,(k) = [\,(-1)^i \sin\left(\frac{\alpha_{pi}}{2}\right)\{w^p(j) + w^p(j-1)\} + 2G_{(j-1)i}^{p}(k)\cos(\alpha_{pi}) - G_{(j-2)i}^{p}(k)] \tag{3.5}$$

Such that,

$$G_{ji}^{p}\,(k) = X(k) = X(2^p(2i+1)) \tag{3.6}$$

Where, $j = 0,1,\ ...,\ \frac{N}{2^p+1} - 1$ and $\alpha_{pi} = \frac{2p(2i+1)\pi}{2N}$

The equation for $w^p(n)$ gives the output of preprocessing stage which will act as an input to the DCT structure, and $x^{p+1}(n)$ gives the intermediate inputs to calculate $w^p(n)$. Taking the Z-transform of Eqn. 3.5, it can be implemented using an IIR direct form 2 filter. This equation is given below as*:*

$$\frac{G_{ji}^{p}\,(k)}{w^p(j)} = \frac{(-1)^i \sin\left(\frac{\alpha_{pi}}{2}\right)(1+z^{-1})}{1 - 2\cos(\alpha_{pi})\,z^{-1} + z^{-2}} \tag{3.7}$$

This equation is represented in the following IIR filter structure as shown below:

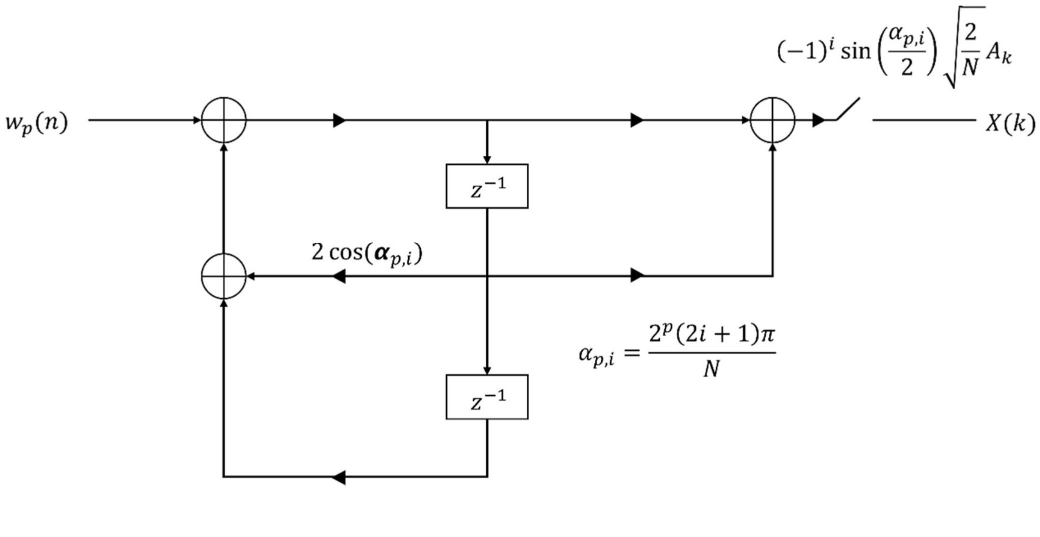

$(-1)^i \sin\left(\frac{\alpha_{p,i}}{2}\right)\sqrt{\frac{2}{N}}A_k$
$w_p(n)$
$X(k)$
z^{-1}
$2\cos(\boldsymbol{\alpha}_{p,i})$
$\alpha_{p,i} = \frac{2^p(2i+1)\pi}{N}$
z^{-1}

Figure 3.1 IIR filter representation of DCT recursive algorithm

Moving forward, when the above IIR filter structure is developed in the Verilog code, it uses the basic evaluation scheme of IIR filter calculation. There form two equations, stated as the following:

$$v(n) = x(n) + 2cos(theta) * v(n-1) - v(n-2) \tag{3.8}$$

$$Y(n)= v(n) + v(n-1) \tag{3.9}$$

As we can see in Eqn. 3.6, the output G_{ji} is evaluated depending on the variables $w^p(n)$, $\cos(\alpha_{pi})$, $\sin\left(\frac{\alpha_{pi}}{2}\right)$, iteration values i and j, so this equation is computed i numbers of time for every value of j. These equations are implemented in Verilog by using modules of multipliers, adders, two's complement, trigonometric lookup table to produce cos and sin values, as Verilog does not support trigonometric calculations.

3.1.2 Preprocessing stage

The values of $w^p(j)$ is evaluated according to the figure shown below in Figure 3.1. To calculate X(0) coefficient of DCT sequence $w^3(0)$ is multiplied with $\sqrt{\frac{2}{N}}Ak$ such that k = 0. The other coeffecients of DCT for $w^0(j)$, $w^1(j)$, $w^2(j)$, $w^3(j)$ are also computed for different values of p. Now, after this stage the outputs are fed as input to the second order IIR filter.

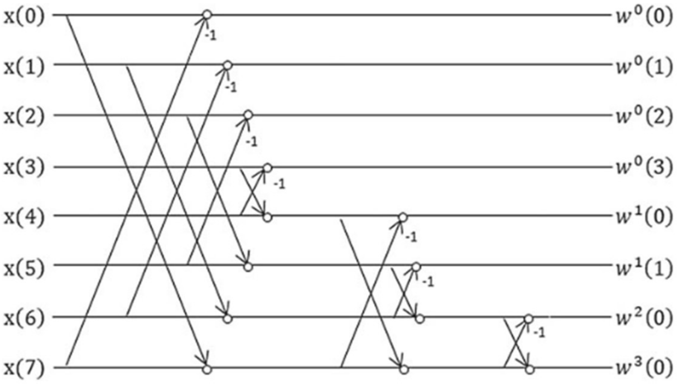


Figure 3.2 Preprocessing stage used for N=8

The above process can be formulated in a mathematical equation as following:

$$w^p(n) = x^p(n) – x^p\left(\frac{N}{2} - n - 1\right) \tag{3.10}$$

$$x^{p+1}(n) = x^p(n) + x^p\left(\frac{N}{2} - n - 1\right) \tag{3.11}$$

The values $x^{p+1}(n)$ are the input values for $w^p(n)$ calculations. When p=0, the input sequence x(n) is used as the input to the preprocessing stage $w^0(n)$. For p=1, $x^{p+1}(n)$ i.e., $x^1(n)$ is calculated which is fed as input to the next preprocessing stage iteration i.e., $w^1(n)$. This iteratively calculates the intermediate values $w^p(n)$.

The above two equations are a part of preprocessing stage, which gives the intermediate resulting values which must be fed as input to the DCT recursive IIR structure. They are implemented in Verilog module such that it executes for every value of p. The output of $x^{p+1}(n)$ becomes the input for the w(n) successive iterations. The preprocessing stage is a part of the Top_Module in the verilog code.

3.1.3 Coefficient calculation:

This section details the recursive computation of Discrete Cosine Transform (DCT) coefficients for both even and odd indices. Once the outputs from the preprocessing stage—corresponding to p=0,1,2,…,(r−1) are obtained, they are passed into the DCT structure for coefficient generation. The flowchart shown in Figure 3.3 illustrates the step-by-step execution of the DCT-II algorithm for a transform length of N=8, outlining the recursive stages involved in coefficient computation. For the given parameters N=8 and recursion level r=3, the algorithm categorizes the coefficients into two main groups: odd and even. The computation begins with the odd-indexed coefficients namely X(1),X(3),X(5),X(7)—followed by the even-indexed coefficients. The even group is further subdivided into two smaller sets: X(2),X(6) and X(4), while the DC coefficient X(0) is computed separately, owing to its distinct structural behaviour in the recursive formulation.

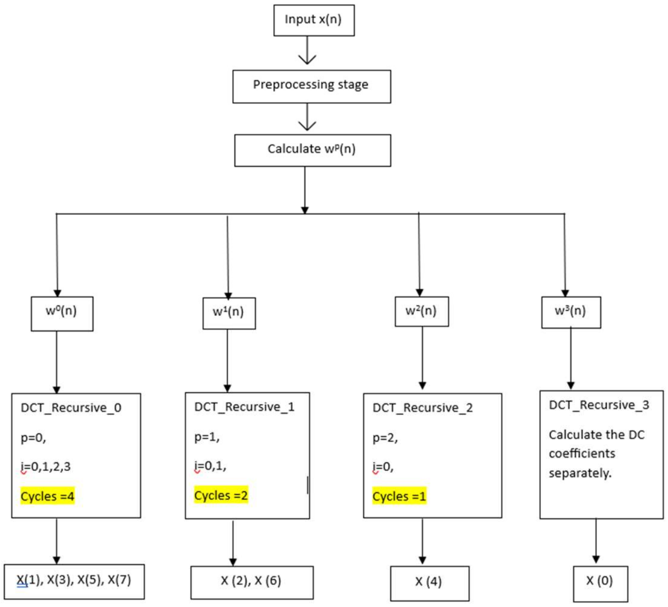


Figure 3.3 Flow chart representing the division of DCT coefficients.

The below process is iterated over number of cycles calculated by the formula $cycles = \frac{N}{2^{p+1}}$ for every value of p. For N=8, at p=0, cycles are equal to 4, and the intermediate outputs $w^0(0)$, $w^0(1)$, $w^0(2)$, $w^0(3)$ are fed to the DCT structure in first iteration successively. After $w^0(0)$ is processed, the output of the DCT structure gives delay 1, delay 2 and DCT_out values. Delay 1 and delay 2 values are fed again to the DCT structure along with the input $w^0(1)$ in the next iteration. This process is repeated for all iterations where p=0 and at the end $w^0(3)$ gives DCT_output value which is then multiplied by $(-1)^i \sin\left(\frac{\alpha_{pi}}{2}\right)\sqrt{\left(\frac{2}{N}\right)Ak}$. For values of $i = 0,1,\ldots, 2^{r-(p+1)}-1$, i.e., i=0,1,2,3 we get odd DCT coefficients X(1), X(3), X(5), X(7).

For next iteration where p=1, the above process is repeated for cycles equal to 2, and the intermediate values $w^1(0)$, $w^1(1)$ are fed to the DCT structure in second iteration along with the delay values. The output of second iteration will give the even DCT coefficients X(2), X(6) and next iteration for p=2, X(4) is calculated by the same procedure.

3.2 Verilog Implementation

An equivalent hardware architecture is developed for the computation of DCT-II algorithm using an IIR filter which is given in the Figure 3.1 by using Eqn 3.8 and 3.9. This is a recursive process which involves a feedback and feed forward loop as shown in the Figure 3.4. This includes the delay element z $^{-1}$ denoted by D flip flop which stores the previous values. The adders and multipliers are used for further calculations.

As we know the output from the preprocessing stage obtained by the equations for $w^p(n)$ serves as the input to the DCT IIR filter structures. Let's say the variable v(n) is the set of inputs calculated over each iteration for the value of p. For eg., N=8, p=0,1,2 there are three cycles ($\frac{N}{2^{p+1}}$), sets of inputs $w^p(n)$. The processing of DCT structure takes v(0) as the first input along with delay values, i.e., v(n-1) and v(n-2) which are initially set to zero. After each iteration, the delay values are updated and given as input along with v(n). Next, Y(n) is computed according to the Eqn 3.9. The output thus obtained after processing of iterations for p=0,

It can be understood easily by the below given parameters:

Table 3.1 DCT Récursive Structure Computation

OPERATOR	**COMPUTATION**	**OUTPUT**
	v(n-1)	Delay 1
	v(n-2)	Delay 2
Multiplier	2cos(theta) * Delay 1	Mult 1
Two's Complement	-Delay 2	Negative Feedback
Adder	Mult 1 + Negative Feedback	Add 1
Adder	w(n) + Add 1	Add 2
Adder	Add 2 + Delay 1	Add 3
Multiplier	A * sin[alpha /2] *Mult 1	Mult 2

Final output : Delay 1 = Add 1

Delay 2 = Delay 1

DCT_out = Mult 2

Due to this, the hardware for IIR filter is drastically reduced as the recursive structure maintains a fixed number of multipliers and adders regardless of N. This efficiency is achieved by reusing intermediate results and processing data in smaller, manageable chunks.

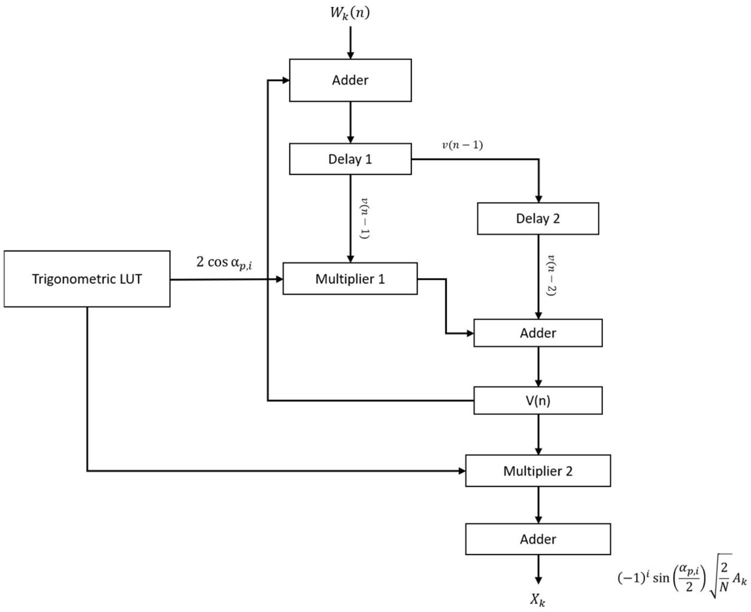


Figure 3.4 Hardware Circuit of DCT realisation for Recursive filter

Fixed Point Arithmetic is used for the fractional implementation in Verilog. This involves the use of IEEE standard for floating points in which implicit normalization is done. This is a technique to represent any fractional number in binary format using half precision, single precision or double precision. In the implemented Verilog code, 32 bits are used to represent the input data. Thus, by using the half precision technique, variables or inputs are defined.

Implicit normalization:

$[(-1)^s * (1.M) * 2^{E-Bias}]$ or $[(-1)^s * (0.M) * 2^{-126}]$

S	E	M

Where, S: Sign bit, **E**: Bias Exponent and **M**: Mantissa

3.3 Testing Methodology

The testing module in a Verilog code defines the validity of the implementation of the recursive Discrete Cosine Transform (DCT structure). This section will consist of the test environment, the process by which input is generated, verification procedures, and the metric which involves how to assess the performance and accuracy.

3.3.1 Test Environment

The DCT structure was implemented and tested using the hardware description language (HDL), specifically Verilog in a simulation environment. The following tools were used for testing.

1. Simulator: Xilinx Vivado for waveform analysis and debugging.
2. Synthesis: Xilinx Vivado for FPGA synthesis
3. Target FPGA: Xilinx Artix
4. Clock Frequency: The design was tested at an operational clock frequency of 100 MHz

3.3.2 Input Data Generation

The major motivation of implementing this structure is to compute the DCT coefficients for any value of $N=2^r$, to test this implementation extensively various sets of input data were used. Any integer sequence for N=8, 16, 32, 64 etc. can be taken. Each integer is a 32 signed bit value given as input.

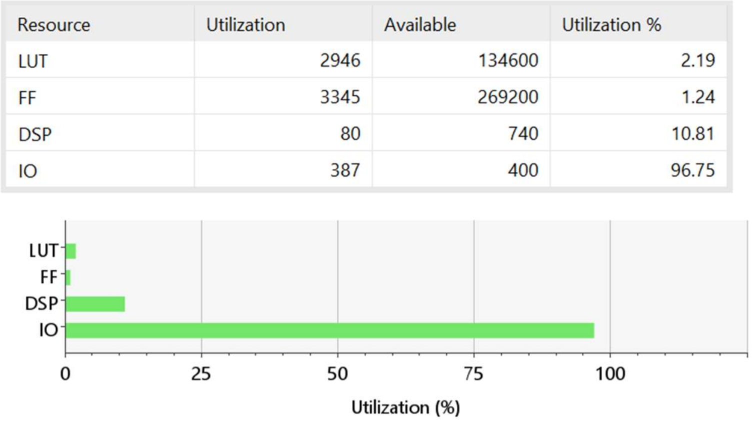
3.3.3 Verification Procedure

1. Simulation Verification: The values obtained in the ‘scope’ and ‘objects’ are analysed, waveform outputs were analysed to ensure intermediate and final results which matches with the expected outputs of the DCT process computed mathematically.
2. Comparison with MATLAB Model: A MATLAB script was also implemented according to the functionality of the DCT recursive structure to compute the results. These results were then matched with the Verilog simulation results to ensure numerical accuracy.

3.3.4 Metrics for validation

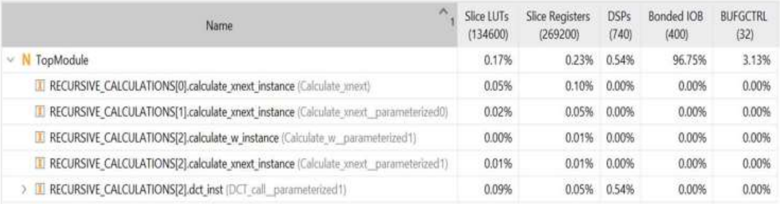
The following metrics are used to evaluate the implementation:

1. Correctness: the output of the code implemented in Verilog was compared against theoretical and MATLAB results for various test cases. Inputs N=8 and N=16 was thoroughly verified.
2. Resource Utilization: FPGA resource utilization (LUTs, flip flops, DSP slices, and BRAM) was also analysed, the report is also given in the below figure.



Resource	Utilization	Available	Utilization %
LUT	2946	134600	2.19
FF	3345	269200	1.24
DSP	80	740	10.81
IO	387	400	96.75

Figure 3.5 Utilization Report 1



Name	Slice LUTs (134600)	Slice Registers (269200)	DSPs (740)	Bonded IOB (400)	BUFGCTRL (32)
TopModule	0.17%	0.23%	0.54%	96.75%	3.13%
RECURSIVE_CALCULATIONS[0].calculate_xnext_instance (Calculate_xnext)	0.05%	0.10%	0.00%	0.00%	0.00%
RECURSIVE_CALCULATIONS[1].calculate_xnext_instance (Calculate_xnext__parameterized0)	0.02%	0.05%	0.00%	0.00%	0.00%
RECURSIVE_CALCULATIONS[2].calculate_w_instance (Calculate_w__parameterized1)	0.00%	0.01%	0.00%	0.00%	0.00%
RECURSIVE_CALCULATIONS[2].calculate_xnext_instance (Calculate_xnext__parameterized1)	0.01%	0.01%	0.00%	0.00%	0.00%
RECURSIVE_CALCULATIONS[2].dct_inst (DCT_call__parameterized1)	0.09%	0.05%	0.54%	0.00%	0.00%

Figure 3.6 Utilization Report 2

3.3.5 Observations

The simulation results showed that the outputs of the preprocessing stage and the DCT coefficient computation stage closely matched the theoretical values, with negligible error due to quantization. Additionally, for larger values of N, the resource utilization increases, leading to an increase in the number of input and output pins. The utilization report presented above shows the statistics only for N = 8.

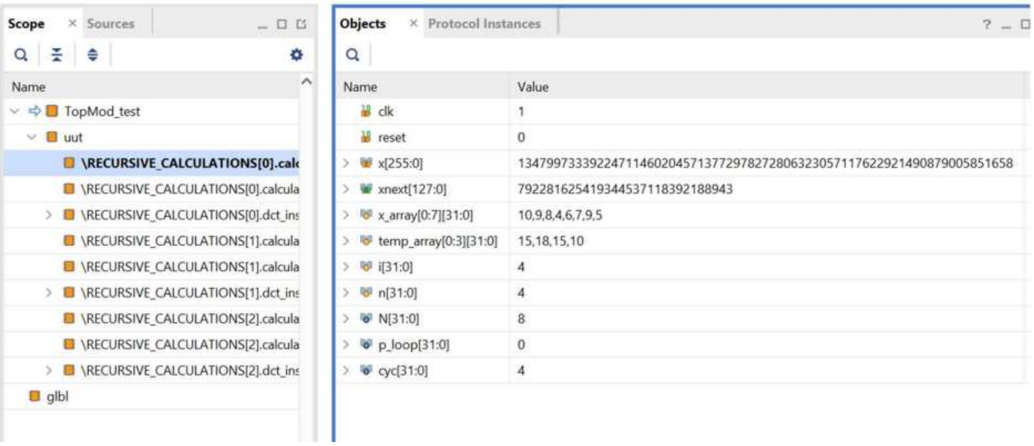
3.4 RESULTS

Input: x[n] = {10,9,8,4,6,7,9,5}

Preprocessing stage Outputs: The values $w^0(n)$ are computed using equation 6 and 7.

Table 3.2 Preprocessing stage results

STAGE (p)	$x^{p+1}(n)$	$w^p(n)$
p = 0	{15, 18, 15, 10}	{5, 0, 1, -2}
p = 1	{25, 33}	{5, 3}
p = 2	{58}	{-8}



```
Scope  × Sources
Name
∨ ⇨ TopMod_test
   ∨ uut
        \RECURSIVE_CALCULATIONS[0].calc
        \RECURSIVE_CALCULATIONS[0].calcula
      > \RECURSIVE_CALCULATIONS[0].dct_ins
        \RECURSIVE_CALCULATIONS[1].calcula
        \RECURSIVE_CALCULATIONS[1].calcula
      > \RECURSIVE_CALCULATIONS[1].dct_ins
        \RECURSIVE_CALCULATIONS[2].calcula
        \RECURSIVE_CALCULATIONS[2].calcula
      > \RECURSIVE_CALCULATIONS[2].dct_ins
   glbl
```

Objects × Protocol Instances

Name	Value
clk	1
reset	0
x[255:0]	134799733392247114602045713772978272806323057117622921490879005851658
xnext[127:0]	792281625419344537118392188943
x_array[0:7][31:0]	10,9,8,4,6,7,9,5
temp_array[0:3][31:0]	15,18,15,10
i[31:0]	4
n[31:0]	4
N[31:0]	8
p_loop[31:0]	0
cyc[31:0]	4

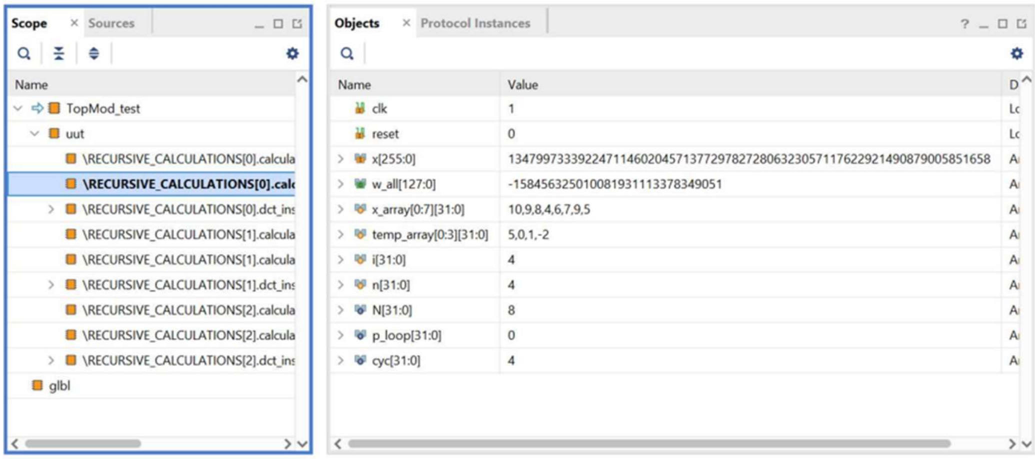
Figure 3.7 Simulation result showing intermediate value of $x^1(n)$

DCT Output:

The Discrete Cosine Transform (DCT) output values in signed decimal format and their corresponding fractional values, calculated using fixed-point Q16.16 for 32 bit precision, are shown below.

Formula used:

$$\boldsymbol{Fractional\,Value} = \frac{\boldsymbol{Signed\ Decimal}}{\boldsymbol{2^{16}}} \tag{3.12}$$



Name
- TopMod_test
  - uut
    - \RECURSIVE_CALCULATIONS[0].calcula
    - **\RECURSIVE_CALCULATIONS[0].calc**
    - \RECURSIVE_CALCULATIONS[0].dct_ins
    - \RECURSIVE_CALCULATIONS[1].calcula
    - \RECURSIVE_CALCULATIONS[1].calcula
    - \RECURSIVE_CALCULATIONS[1].dct_ins
    - \RECURSIVE_CALCULATIONS[2].calcula
    - \RECURSIVE_CALCULATIONS[2].calcula
    - \RECURSIVE_CALCULATIONS[2].dct_ins
- glbl

Name	Value	D
clk	1	Lc
reset	0	Lc
x[255:0]	134799733392247114602045713772978272806323057117622921490879005851658	A
w_all[127:0]	-158456325010081931113378349051	A
x_array[0:7][31:0]	10,9,8,4,6,7,9,5	A
temp_array[0:3][31:0]	5,0,1,-2	A
i[31:0]	4	A
n[31:0]	4	A
N[31:0]	8	A
p_loop[31:0]	0	A
cyc[31:0]	4	A

Figure 3.8 Simulation result showing the intermediate values $w^0(n)$

Table 3.3 DCT Coefficient calculations after simulation

INDEX of DCT []	Signed Decimal Value	Fractional value (used Fixed point Q16.16 for 32 bit precision)
0	1343338	20.50
1	166101	2.53
2	188981	2.88
3	140198	2.14
4	-185360	-2.82
5	42922	0.65
6	-28127	-0.42
7	123476	1.88

3.5 Overview of the RTL design

The RTL (Register Transfer Level) design implementation figure 3.9 represents the hardware implementation of the recursive DCT algorithm. This RTL was generated using the Verilog/VHDL code and was synthesized using Xilinx Vivado, this consists of multiple interconnected various functional blocks of the algorithm. The resulting RTL design has various functional modules, these includes the preprocessing stages, recursive computation blocks and the final output aggregations. The elements such as multipliers and adders known as the data path elements are used to perform the required arithmetic operations whereas the control logic manages the iterative processing across the multiple cycles. The RTL figure 3.9 is hierarchical and thus modularization in the design is enabled by instantiating the modules and submodules in the Top module of the Verilog code in a recursive structure, this is how the data flows sequentially from one stage to another, ensuring efficient processing. Thus, the RTL implementation of the recursive DCT algorithm can successfully captures the functionality of the theoretical model in hardware.

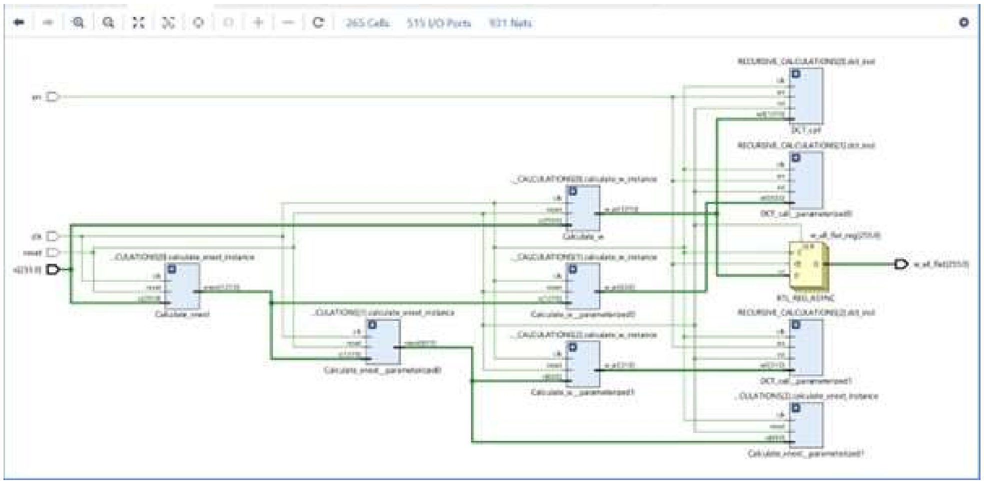

265 Cells
515 I/O Ports
931 Nets
en
clk
reset
RECURSIVE_CALCULATIONS[0].dct_inst
DCT_cal
RECURSIVE_CALCULATIONS[1].dct_inst
DCT_cal__parameterized0
_CALCULATIONS[0].calculate_w_instance
Calculate_w
_CALCULATIONS[1].calculate_w_instance
Calculate_w__parameterized0
_CALCULATIONS[2].calculate_w_instance
Calculate_w__parameterized1
Calculate_xnext
Calculate_xnext__parameterized0
RTL_REG_ASYNC
RECURSIVE_CALCULATIONS[2].dct_inst
DCT_cal__parameterized1
Calculate_xnext__parameterized1

Figure 3.9 Elaborated RTL Design of DCT Structure

3.6 CONCLUSION

The proposed algorithm of DCT coefficient computation shows that the computation complexity can be reduced as the number of multipliers remains the same irrespective of the value of N. By this recursive filter structure, reduction in computational overhead is achieved. This structure efficiently uses the intermediate results and processes the data in smaller manageable groups. The feasibility and effectiveness of this proposed architecture can be presented in the Verilog implementation which makes it highly suitable for real-time signal processing applications. Reduction errors are also reduced by using minimum number of multipliers and adders in this recursive structure which leads to more accurate and reliable computations. This research work has contributed in the ongoing efforts to optimize DCT computations and it also paves the way for more efficient data compression and processing technologies. This approach simplifies the hardware design which makes it an attractive solution for various digital signal processing tasks.

The future scope involves in the research and development using extension to other transforms such as DST and DHT. By formulating similar recursive algorithm for these transforms, the benefits of reduced computational complexity can be realized. Further, the VHDL implementation can be done in the FPGA prototyping which will provide insights into its physical performance and practical usability. It helps to assess some real-world factors such as the power consumption, processing speed and resource utilization. Furthermore, the optimization for larger values of N could be explored. Also integrating the DCT computation with popular compression algorithms like JPEG and MPEG can directly demonstrate its impact on improving compression efficiency and reducing processing time. Adding to this, further research could focus on optimizing the recursive structure for low-power and high-speed applications essential for portable and embedded systems. By addressing these future directions, recursive

DCT computation algorithm can be further defined and adapted to meet the evolving demands of digital signal processing, making it a versatile and valuable tool for wide range of applications.

CHAPTER 4

ANALYSIS AND IMPLEMENTATION OF DHT RECURSIVE STRUCTURE IN VERILOG

4.1 MATHEMATICAL BACKGROUND

DHT is a transform that converts samples of N real-valued numbers x(n) into coefficients of real numbers H(k) with the help of the formula given as

$$H[k] = \sum_{n=0}^{N-1} x(n)\ \mathrm{cas}\left(\frac{2\pi kn}{N}\right) \qquad (4.1)$$

where k = 0 to N-1,

$$\mathrm{cas}(\theta) = \cos(\theta) + \sin(\theta)$$

The Eqn 4.1 can be split into two parts where first part is from n=0 to $\left(\frac{N}{2} - 1\right)$ and $n = \frac{N}{2}$ to N-1.

The properties of symmetry and periodicity are used to decompose the DHT expression. By this, the recursive nature of DHT can be explored. By putting $(n' = n + \frac{N}{2})$ and applying trigonometric properties, the equation can be further manipulated. An important part of the derivation in [19] shows the preprocessing stage outputs.

The right term of the Eqn 4.1 reduces and becomes

$$H[k] = \sum_{n=0}^{N/2-1} x(n)\ \mathrm{cas}\left(\frac{2\pi kn}{N}\right) + \sum_{n=0}^{N/2-1} x(n + N/2)\ \mathrm{cas}\left(\frac{2\pi k(n + N/2)}{N}\right) \qquad (4.2)$$

where $(n' = n + \frac{N}{2})$.

Solving the above equation further and expanding the right term of Eqn 4.2,

$$H[k] = \sum_{n=0}^{N/2-1} x(n)\ \mathrm{cas}\left(\frac{2\pi kn}{N}\right) + \sum_{n=N/2}^{N-1} x(n+N/2)\left[\cos\left(\frac{2\pi kn}{N}+\pi k\right) - \sin\left(\frac{2\pi kn}{N}+\pi k\right)\right] \quad (4.3)$$

where k = 0 to N-1.

$$H[k] = \sum_{n=0}^{N/2-1} x(n)\ \mathrm{cas}\left(\frac{2\pi kn}{N}\right) + \sum_{n=N/2}^{N-1} x(n+N/2)\cdot(-1)^k\left[\mathrm{cas}\left(\frac{2\pi kn}{N}\right)\right] \quad (4.4)$$

where k=0 to N-1

$$W_k(n) = x(n) + (-1)^k \cdot x(n+N/2) \quad (4.5)$$

The resulting W_k is the input given to the DHT structure which will help further to calculate the DHT coefficients. Now, put $(n' = \frac{N}{2} - 1 - n)$ to simplify the expression more and make the structure more symmetric. This is done to simplify the expression within trigonometric functions (such as $(\mathrm{cas}(\theta))$. This also reveals the property of symmetries in the structure of the Discrete Hartley Transform (DHT). This reindexing is done by which the order of summation is reversed.

$$H[k] = \sum_{n=0}^{\frac{N}{2}-1} W_k\left(\frac{N}{2} - 1 - n\right)\cdot \mathrm{cas}\big(\pi k - \theta_k \cdot n(n+1)\big) \quad (4.6)$$

$$H[k] = \sum_{n=0}^{\frac{N}{2}-1} W_k\left(\frac{N}{2} - 1 - n\right)\cdot \mathrm{cms}\big(\theta_k(n+1)\big) \quad (4.7)$$

So the final form of the expression becomes

$$X[k] = (-1)^k \cdot G_i[k] \quad (4.8)$$

where $(i = \frac{N}{2} - 1)$, and(θ_k) is defined as $(\theta_k = \frac{2\pi k}{N})$

After the mathematical manipulations, the resulting equation turns out to be a compact and symmetric form of the DHT expression. This equation becomes the base of the proposed DHT recursive structure and helps to develop efficient recursive computation and hardware-friendly implementations. For $(i = 0,1, \dots \frac{N}{2} - 1)$ multiple stages of G_k is obtained. Using trigonometric properties and performing derivation, the recursion for $(G_i[k])$ from Eqn 4.8 results in:

$$G_i[k] = W_k[i]\cos(\theta_k) + 2G_{i-1}[k]\cos(\theta_k) - W_k[i]\sin(\theta_k) - W_k[i-1] - G_{i-2}[k] \tag{4.9}$$

where $(k \ = \ 0, 1, 2, \dots, N - 1)$.

The full derivation is referred from [19]. Applying z-transform to the Eqn 4.9, transfer function:

$$\frac{G_k(z)}{W_k(z)} = \frac{\cos(\theta_k) - \sin(\theta_k) - z^{-1}}{1 - 2\cos(\theta_k)\, z^{-1} + z^{-2}} \tag{4.10}$$

4.2 DHT STRUCTURE

The Eqn 4.1, represents an Infinite Impulse Response (IIR) equation of Direct form -II type which can be used to obtain the structure shown in Fig 4.1. It shows that W_k is the input given to the DHT structure whose calculation is included in the pre-processing stage. This recursive structure has fixed numbers of delays, multipliers and adders which makes the proposed methodology efficient in calculating DHT coefficients by using the same structure again and again. Thus, the number of multipliers and adders remains same for any number of N. The only difference is the number of iterations this structure takes to evaluate the coefficients.

The input sequence is not directly fed into this structure, while the inputs goes through a preprocessing stage which results in W_k, calculated by the help of Eqn 4.5. The output of this preprocessing stage is then fed to the IIR filter structure.

In this structure, there are feedforward paths and feedback paths, where the intermediate results are fed back along with the inputs to achieve the final coefficients, and this makes it recursive in nature. For N inputs, there are $k = N - 1$ cycles and in each cycle, there are $(i = \frac{N}{2} - 1)$ iterations. In any k^{th} cycle, the $\left(\frac{N}{2} - 1\right)^{\text{th}}$ iteration gives the final DHT coefficient for that particular cycle.

4.3 VERILOG IMPLEMENTATION

According to Figure 4.1 an equivalent hardware can be built which incorporates the same functionality by using blocks like multipliers, adders, and delays.

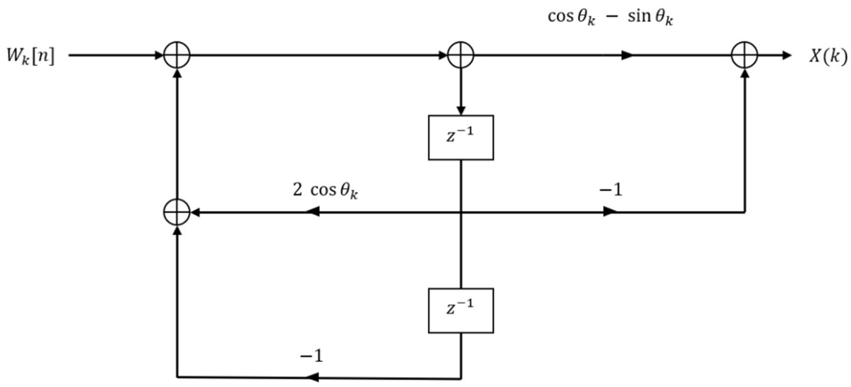


Figure 4.1 IIR filter structure of DHT recursive algorithm [19]

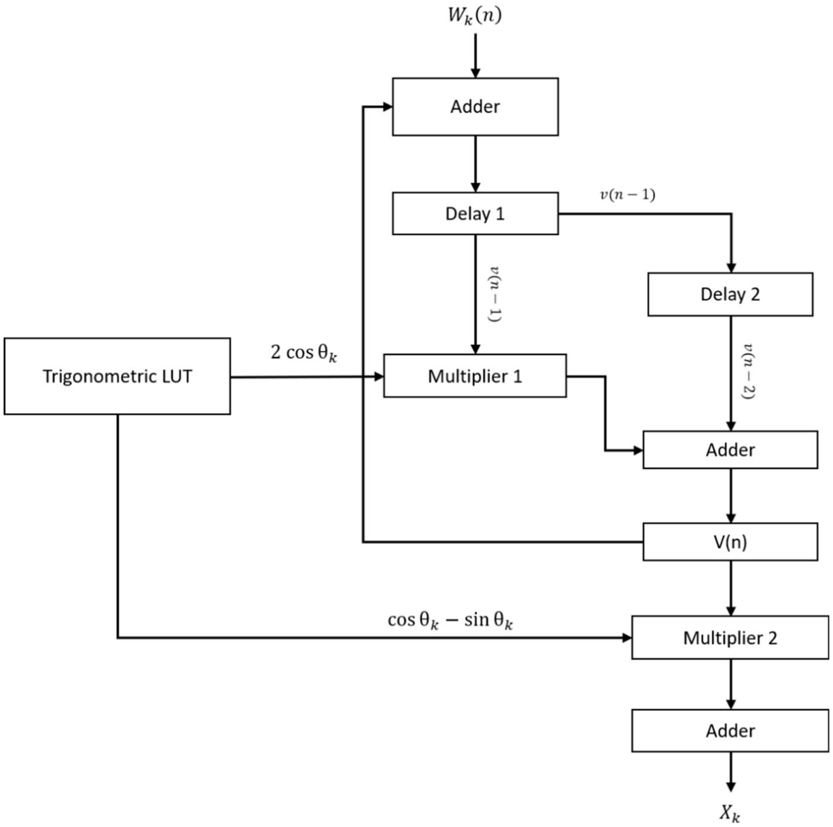


Figure 4.2 Hardware Circuit of DHT realisation for Recursive filter

4.3.1 PREPROCESSING STAGE

The Top Module of the Verilog code is the main controller for the DHT coefficient evaluation. This handles the input collection and streams the output coefficients sequentially. The module begins storing the input N samples in 32 bits each into registers. When all the inputs are collected, the intermediate computation starts. In this calculation phase, the W_k is filled with values by the calculation using Eqn 4.5 which has even and odd symmetry according to k values which range from 0 to N-1.

Let m = 4, so N = 8, n = 0, 1, 2, 3, $(i = \frac{N}{2} - 1)$, and (θ_k) is defined as $(\theta_k = \frac{2\pi k}{N})$. For each value of k from 0 to 7 and n from 0 to 3, $W_k(n)$ is computed. This results into 32 values such as for each k there are 4 values and based on k to be even or odd, the resulting W_k are symmetrical and hence follows a repetitive pattern.

Mathematically,

$$\text{For even } k: \quad w_k(n) = x[n] + x[n+4], \quad \text{for } n = 0 \text{ to } 3$$

$$\text{For odd } k: \quad w_k(n) = x[n] - x[n+4], \quad \text{for } n = 0 \text{ to } 3$$

This symmetry allows to reduce redundant computation and takes only 2 clock cycles to compute W_k for parity of k. This type of pattern can help in simplifying the hardware by enabling reuse of computational blocks based on value of k (even or odd). Now the output of this preprocessing stage becomes the input for DHT IIR recursive filter structure.

4.3.2 COEFFICIENT CALCULATION

This part involves implementing the recursive structure in Verilog and feeding the inputs to it to achieve the required output. By Figure 4.1, IIR filter is implemented in Verilog using Direct form II equations given below:

$$v(n) = w(n) + 2\cos(\theta) * v(n-1) - v(n-2) \qquad (4.11)$$

$$Y(n) = v(n)[\cos(\theta) - \sin(\theta)] + v(n-1) \qquad (4.12)$$

These equations when analyzed into a hardware circuit, includes multipliers, adders, delays and twos complements blocks as shown in the figure. In detail, the outputs attained through the preprocessing stage are fed as inputs into the recursive structure.

Let’s say the variable W_k is the set of inputs calculated over each iteration for the value of k. For eg., N=8, there are *k* cycles and *i* loops. The processing of DHT structure takes $W_k(0)$ as the first input along with delay values, i.e., v(n-1) and v(n-2) which are initially set to zero. After each iteration, the delay values are updated and given as input along with W_k. Next, Y(n) is computed according to the Eqn 4.12.

Verilog code for this recursive DHT structure is written such that the outputs are intermediate coefficient values, v(n-1) and v(n-2). These delay values are given as feedback to next loop and after each cycle is completed these values are again set back to zeros. This recursively achieved by Figure 4.2 which has feedforward and feedback paths. In each cycle, the iteration over *'i'* loops are done where the final output will be at the $(i = \frac{N}{2} - 1)$ loop. This output is the final DHT coefficient which was aimed to be calculated and matches the theoretical tested results. This is explained in the given flow chart in Figure 4.3.

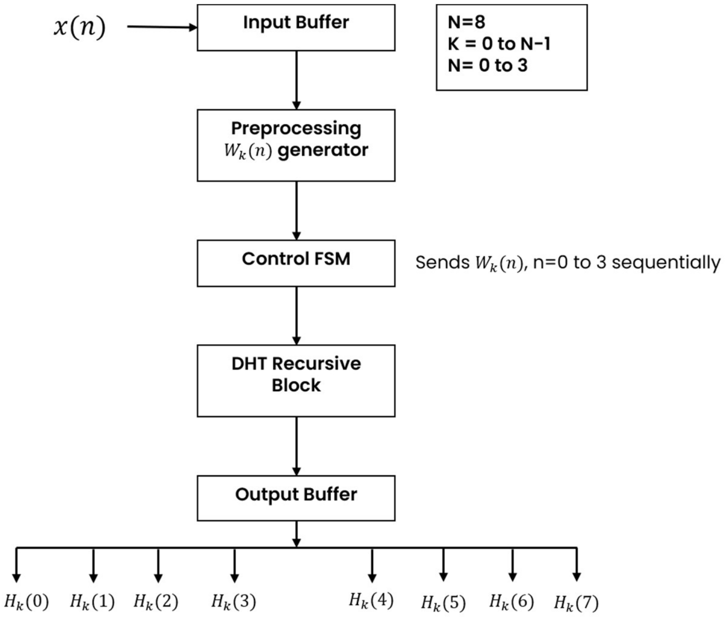


Figure 4.3 Flow Chart for DHT Realisation

Since, in Verilog, fractional values are written in Fixed Point Arithmetic format following IEEE standards so implicit normalization is also done. This method helps to write any fractional number in binary format using various types of precision techniques. In the implemented Verilog code, 32 bits are used to represent the input and output data. Thus, by using the half precision technique, variables or inputs are defined.

4.4 RESULT

This section presents the simulation outputs, streaming behavior, synthesis results, and a comparative analysis of the proposed recursive DHT architecture implemented in Verilog. The results validate the functional correctness, memory efficiency, and suitability for real-time VLSI applications. MATLAB result after evaluating Eqn 4.1 is given as:

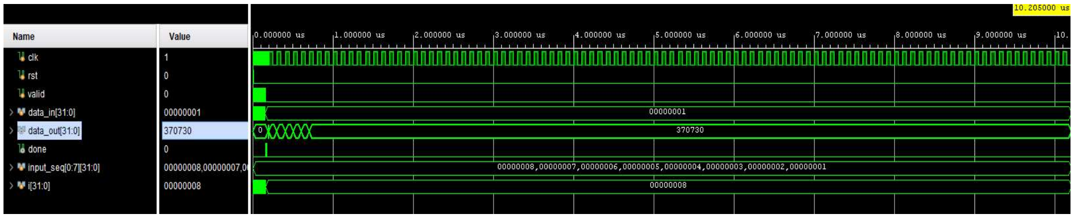
Table 4.1 DHT Coefficient Result through MATLAB

INDEX of X[]	DHT COEFFICIENT
0	36
1	-13.65685424
2	8
3	5.65685424
4	4
5	-2.34314575
6	0
7	-5.65685424

4.4.1 Functional Simulation

The functionality of the proposed design was tested through test benches written in Verilog using Vivado 2023.2 and through the simulation results. The design was tested by giving a known input sequence, and then the output DHT coefficients were compared with the expected results computed using MATLAB. Simulation results of the recursive structure are shown in Figure 4.4.

- **Input Vector:** x(n) = {8, 7, 6, 5, 4, 3, 2, 1}
- **Expected Output:** Computed using standard DHT formula (floating-point) by Eqn 4.1.
- **Observed Output:** Matches expected results up to 3 decimal places



Name	Value
clk	1
rst	0
valid	0
data_in[31:0]	00000001
data_out[31:0]	370730
done	0
input_seq[0:7][31:0]	00000008,00000007,0(
i[31:0]	00000008

Figure 4.4 Simulation Result

4.4.2 Output Streaming Behavior

The proposed architecture outputs one 32-bit DHT coefficient per clock cycle after loading the input stream. Data is processed in a pipeline without requiring memory storage of all coefficients.

- **Input Method:** Streaming 32-bit interface
- **Output Rate**: 1 coefficient per clock cycle
- **Latency:** Approximately 10--12 cycles depending on FSM configuration
- **Throughput**: Sustained at 1 coefficient/cycle after initial delay

4.4.4 Synthesis Results

The Verilog code was synthesized using Xilinx Vivado for Artix-7 (xc7a100tcsg324-1) FPGA. The design shows efficient hardware utilization given in Figure 4.5 and Figure 4.6.

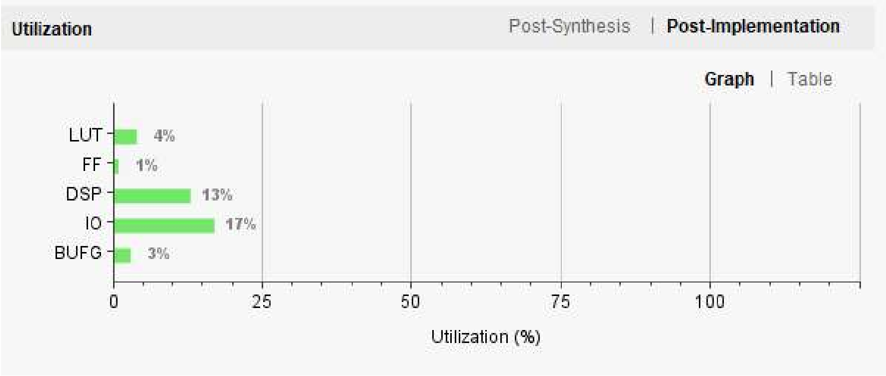
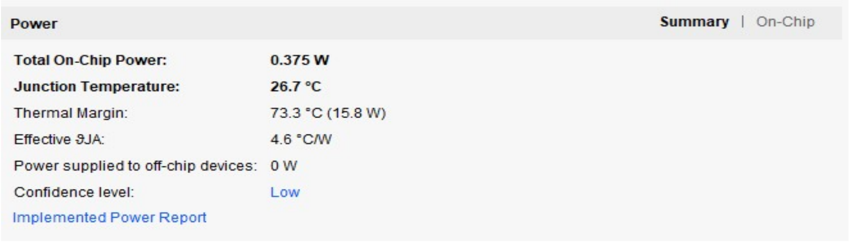


Figure 4.5 Utilization Report after Synthesis for DHT

The resource utilization of the proposed recursive DHT architecture was obtained after synthesizing the Verilog design using **Xilinx Vivado 2023.1** for the **Artix-7 (xc7a100tcsg324-1)** FPGA. The design demonstrates efficient usage of logic resources, with minimal reliance on memory blocks or digital signal processing (DSP) slices.



Total On-Chip Power:	**0.375 W**
Junction Temperature:	**26.7 °C**
Thermal Margin:	73.3 °C (15.8 W)
Effective ϑJA:	4.6 °C/W
Power supplied to off-chip devices:	0 W
Confidence level:	Low

Implemented Power Report

Figure 4.6 Power Utilization report after synthesis for DHT

Due to the use of a serialized streaming input and output mechanism (32-bit word per clock), the design significantly reduces the need for wide parallel ports. The low utilization figures across all resource categories confirm the area efficiency of the proposed design.

Figure 4.7 shows the synthesized netlist of the proposed Discrete Hartley Transform (DHT) architecture as generated by the Xilinx Vivado tool. The netlist represents the gate-level realization of the RTL design, where all high-level Verilog modules are elaborated into interconnected logic primitives such as LUTs, flip-flops, and arithmetic units.

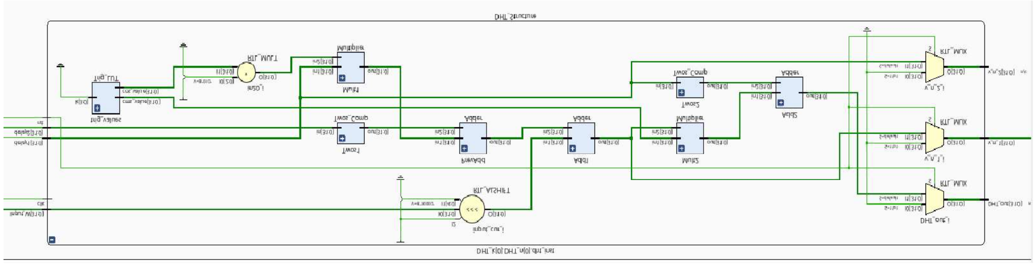


Figure 4.7 DHT Netlist after synthesis

The synthesized netlist validates the modular and hierarchical design approach. Each functional block is instantiated independently and connected using standard interfaces. The netlist figure 4.7 serves as evidence that the recursive DHT architecture is synthesis-friendly and well-suited for real-time hardware implementation. This figure 4.7 only shows cropped part of full netlist structure. For the full structure refer to the Apendix. The input sequence x(n) for which the output is shown in table 4.3 in signed decimal format as received in Verilog along with its fractional values.

Table 4.2 Synthesis Results on Artix-7 FPGA

Resource Type	**Used**	**Available**	**Utilization %**
Slice LUTs	2524	63400	3.98
DSPs	32	240	13.33
Bonded IOB	36	200	17.14
Slice Registers	41	126800	0.03

$$\text{Fractional Value} = \frac{\text{Signed Decimal}}{2^{16}} \qquad (4.13)$$

Table 4.3 DHT coefficients calculations

DHT Coefficients X[]	Signed Decimal Values	Fractional Values
0	2359296	36
1	-895017	-13.65685
2	524288	8
3	-370694	5.65685
4	262144	4
5	-153561	-2.34314
6	0	0
7	370720	-5.65685

4.5 CONCLUSION

In this work, a recursive and memory-efficient Verilog-based implementation of the DHT was successfully developed and evaluated. The design ensures reduced I/O utilization and improved scalability. A finite state machine (FSM) was implemented to manage data loading, computation sequencing, and output streaming. Synthesis results from Vivado confirm that the design meets key constraints for hardware implementation, including optimal utilization of logic resources, minimal use of bonded I/O, and efficient routing. The netlist visualization validates the modular structure.

Overall, the proposed recursive DHT design demonstrates the feasibility of implementing real-time, low-memory signal processing systems in Verilog. Future work may focus on extending this design for higher-order transforms and integrating on FPGA or ASIC platforms.

CHAPTER 5

COCLUSION

5.1 Conclusion

This research presents a unified approach to optimizing DCT and DHT through recursive and memory-efficient hardware implementations using Verilog. The recursive filter structure developed for DCT enables a significant reduction in computational complexity, as the number of multipliers remains independent of the transform length NNN. This approach efficiently utilizes intermediate results, processes data in manageable blocks, and minimizes the use of hardware resources such as multipliers and adders. Consequently, the architecture achieves higher accuracy, improved reliability, and lower implementation overhead—making it highly suitable for real-time signal processing and compression systems.

In parallel, a pipelined, modular, and scalable architecture for DHT was designed and implemented in Verilog. The system incorporates a finite state machine (FSM) to control data loading, computation sequencing, and output streaming, thereby achieving low I/O utilization and high throughput. Synthesis results from Vivado validate that the DHT architecture meets critical design constraints such as optimal logic utilization, efficient routing, and reduced bonded I/O usage. Netlist visualization further confirms the design's modular and scalable structure, emphasizing its applicability in practical hardware systems.

Collectively, these implementations demonstrate the practical viability of recursive structures for DCT and DHT in resource-constrained environments and real-time applications. The developed architectures strike a balance between performance, resource efficiency, and design simplicity, contributing meaningfully to the ongoing efforts in hardware-efficient signal processing.

5.2 Future Scope

The proposed architectures open several avenues for future research and development:

- **Extension to Other Transforms**: The recursive filter methodology can be extended to other transforms such as the DST, thereby generalizing the framework for multiple transform types with reduced computational complexity.

- **VHDL Implementation and FPGA Prototyping**: Porting the design to VHDL and deploying it on FPGA platforms will offer insight into physical design metrics, including power consumption, operating speed, and resource utilization under real-time conditions.
- **Integration with Compression Standards**: Embedding the recursive DCT module into widely used compression algorithms like JPEG and MPEG can directly quantify its impact on compression efficiency, latency, and power performance.
- **Low-Power and Embedded Applications**: Further research could focus on optimizing these designs for ultra-low-power and high-speed performance, targeting portable, battery-operated, and embedded DSP systems.
- **ASIC Implementation**: Exploring ASIC-based implementations could yield custom chip designs for specific real-time applications, improving efficiency, compactness, and power management.

In conclusion, this work lays a strong foundation for real-time, scalable, and hardware-efficient transform computation. By addressing the outlined future directions, the recursive DCT and DHT designs can be adapted to meet the growing demands of modern digital signal processing applications across various domains.

5.3 Social Impact

The growing demand for real-time signal processing in various applications such as healthcare monitoring, communication systems, and smart infrastructure requires solutions that are not only efficient but also accessible and sustainable. This research contributes to this goal by presenting hardware efficient implementations of DHT and DCT using recursive structures in Verilog.

By minimizing computational complexity, hardware area, and power consumption, the proposed architectures enable the deployment of cost effective and energy efficient DSP solutions. These designs are especially relevant for:

- **Healthcare Devices:** Real time transforms in ECG and PPG processing are critical for wearable and portable diagnostic systems. The low-power and real-time nature of the proposed architecture supports continuous monitoring in remote or underserved area.
- **Sustainable Electronics:** When there is power optimized hardware solution, it reduces the energy demand, and this also aligns with global efforts towards green technology and carbon footprint reduction.
- **Inclusive Technology:** FPGA or ASIC based designs are resource efficient and these designs make advanced signal processing feasible in low cost and low resource environments, broadening acess to digital healthcare, education and communication.
- **Smart Infrastructure:** The real time processing capabilities support applications in smart cities, environmental monitoring, and disaster response

systems, improving safety, responsiveness and quality of life. These can be implemented in monitoring traffic congestion, public safety (CCTV/audio sensors), seismic or vibration detectors which needs to be digitized and the low hardware complexity and low power makes it ideal for battery powered sensor nodes, real time signal processing before transmission, reducing data payload and saving energy.

By enabling scalable, affordable, and environmentally responsible signal processing systems, this research contributes meaningfully to the broader vision of technology for social good.

REFERENCES

[1] N. I. Cho and S. U. Lee, “DCT Algorithms for VLSI Parallel Implementations,” *IEEE Trans Acoust*, vol. 38, no. 1, 1990, doi: 10.1109/29.45624.

[2] H. Huang and L. Xiao, “CORDIC based fast radix-2 DCT algorithm,” *IEEE Signal Process Lett*, vol. 20, no. 5, 2013, doi: 10.1109/LSP.2013.2252616.

[3] M. A. Benayed *et al.*, “New design using a VHDL description for DCT based circuits,” in *Proceedings of the 10th International Conference on Microelectronics, ICM 1998*, 1998. doi: 10.1109/ICM.1998.825575.

[4] S. An and C. Wang, “Recursive algorithm, architectures and FPGA implementation of the two-dimensional discrete cosine transform,” *IET Image Process*, vol. 2, no. 6, 2008, doi: 10.1049/iet-ipr:20080057.

[5] Y. Zhou, C. Wang, and X. Zhou, “DCT-Based Color Image Compression Algorithm Using an Efficient Lossless Encoder,” in *International Conference on Signal Processing Proceedings, ICSP*, 2019. doi: 10.1109/ICSP.2018.8652455.

[6] Z. Wang, G. A. Jullien, and W. C. Miller, “Implementation of the discrete cosine transform and its inverse by digital filtering,” in *Midwest Symposium on Circuits and Systems*, 1994. doi: 10.1109/mwscas.1994.518940.

[7] P. Dahiya and P. Jain, “Realization of Second-Order Structure of Recursive Algorithm for Discrete Cosine Transform,” *Circuits Syst Signal Process*, vol. 38, no. 2, 2019, doi: 10.1007/s00034-018-0885-6.

[8] R. N. Bracewell, “Discrete Hartley transform,” *J Opt Soc Am*, vol. 73, no. 12, 1983, doi: 10.1364/josa.73.001832.

[9] R. N. Bracewell, “The Fast Hartley Transform,” *Proceedings of the IEEE*, vol. 72, no. 8, 1984, doi: 10.1109/PROC.1984.12968.

[10] K. Maharatna, A. S. Dhar, and S. Banerjee, “A VLSI array architecture for realization of DFT, DHT, DCT and DST,” *Signal Processing*, vol. 81, no. 9, 2001, doi: 10.1016/S0165-1684(01)00061-5.

[11] S. Jolad and Shridhar, “Speech enhancement using DFT and DHT with reduction of cross spectral terms,” in *2016 International Conference on Signal and Information Processing, IConSIP 2016*, 2017. doi: 10.1109/ICONSIP.2016.7857453.

[12] H. Jleed and M. Bouchard, “Acoustic environment classification using discrete hartley transform features,” in *Canadian Conference on Electrical and Computer Engineering*, 2017. doi: 10.1109/CCECE.2017.7946646.

[13] D. F. Chiper, “A systolic array algorithm based on band-convolution structure for an efficient VLSI implementation of the odd-time generalized discrete

hartley transform," in *ISSCS 2019 - International Symposium on Signals, Circuits and Systems*, 2019. doi: 10.1109/ISSCS.2019.8801763.

[14] F. A. De Oliveira Nascimento, "Hartley Transform Signal Compression and Fast Power Quality Measurements for Smart Grid Application," *IEEE Transactions on Power Delivery*, vol. 38, no. 6, 2023, doi: 10.1109/TPWRD.2023.3302188.

[15] J. I. Gtiot, C. M. Litrt, and C. W. Jen, "A novel VLSI array design for the discrete Hartley transform using cyclic convolution," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 1994. doi: 10.1109/ICASSP.1994.389609.

[16] D. F. Chiper and V. Munteanu, "New design approach to VLSI parallel implementation of discrete Hartley transform," in *IEEE International Symposium on Industrial Electronics*, 1996. doi: 10.1109/isie.1996.548420.

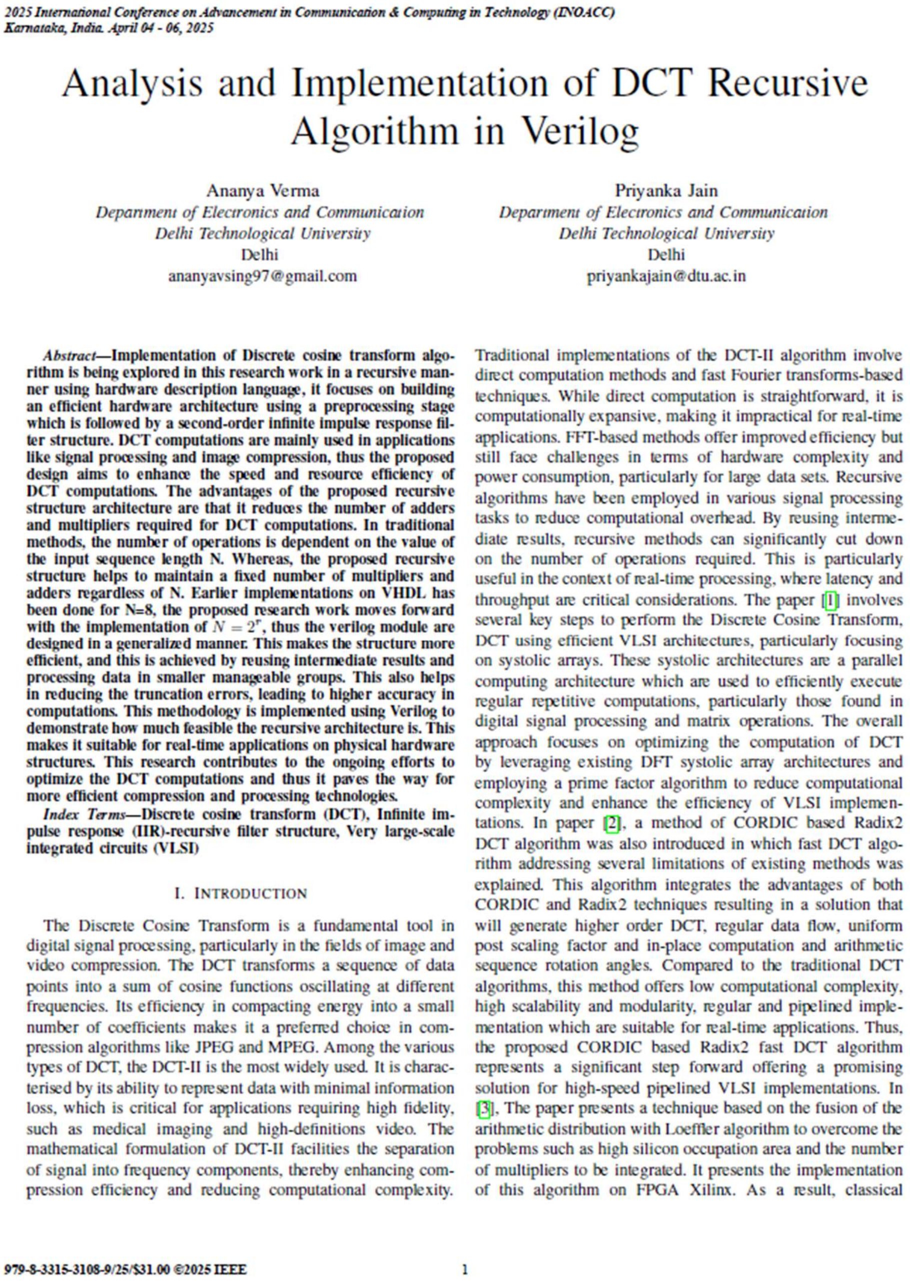
[17] P. Saha, P. Bhattacharyya, D. Kumar, and A. Dandapat, "ASIC implementation of high speed processor for computing fast Hartley transformation," in *Proceedings of the 2013 International Conference on Advanced Electronic Systems, ICAES 2013*, 2013. doi: 10.1109/ICAES.2013.6659424.

[18] D. G. Et. al., "Resourceful Fast Discrete Hartley Transform to Replace Discrete Fourier Transform with Implementation of DHT Algorithm for VLSI Architecture," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 10, 2021, doi: 10.17762/turcomat.v12i10.5329.

[19] Singh, V., Kaushal, D. and Jain, P., 2024. "Discrete hartley transform using recursive algorithm" *Advances in AI for Biomedical Instrumentation, Electronics and Computing* (pp. 583-588). CRC Press, doi: /10.1201/9781032644752.

LIST OF PUBLICATIONS AND THEIR PROOFS

[1] **Ananya Verma, Priyanka Jain** "Analysis and Implementation of DCT Recursive Structure in Verilog" *International Conference on Advancement in Communication and Computing Technology, Bangalore, India, 2025 (INOACC)* – **Accepted,** to be published in August 2025.



2025 International Conference on Advancement in Communication & Computing in Technology (INOACC)
Karnataka, India. April 04 - 06, 2025

Analysis and Implementation of DCT Recursive Algorithm in Verilog

Ananya Verma
Department of Electronics and Communication
Delhi Technological University
Delhi
ananyavsing97@gmail.com

Priyanka Jain
Department of Electronics and Communication
Delhi Technological University
Delhi
priyankajain@dtu.ac.in

***Abstract*—Implementation of Discrete cosine transform algorithm is being explored in this research work in a recursive manner using hardware description language, it focuses on building an efficient hardware architecture using a preprocessing stage which is followed by a second-order infinite impulse response filter structure. DCT computations are mainly used in applications like signal processing and image compression, thus the proposed design aims to enhance the speed and resource efficiency of DCT computations. The advantages of the proposed recursive structure architecture are that it reduces the number of adders and multipliers required for DCT computations. In traditional methods, the number of operations is dependent on the value of the input sequence length N. Whereas, the proposed recursive structure helps to maintain a fixed number of multipliers and adders regardless of N. Earlier implementations on VHDL has been done for N=8, the proposed research work moves forward with the implementation of $N = 2^r$, thus the verilog module are designed in a generalized manner. This makes the structure more efficient, and this is achieved by reusing intermediate results and processing data in smaller manageable groups. This also helps in reducing the truncation errors, leading to higher accuracy in computations. This methodology is implemented using Verilog to demonstrate how much feasible the recursive architecture is. This makes it suitable for real-time applications on physical hardware structures. This research contributes to the ongoing efforts to optimize the DCT computations and thus it paves the way for more efficient compression and processing technologies.**

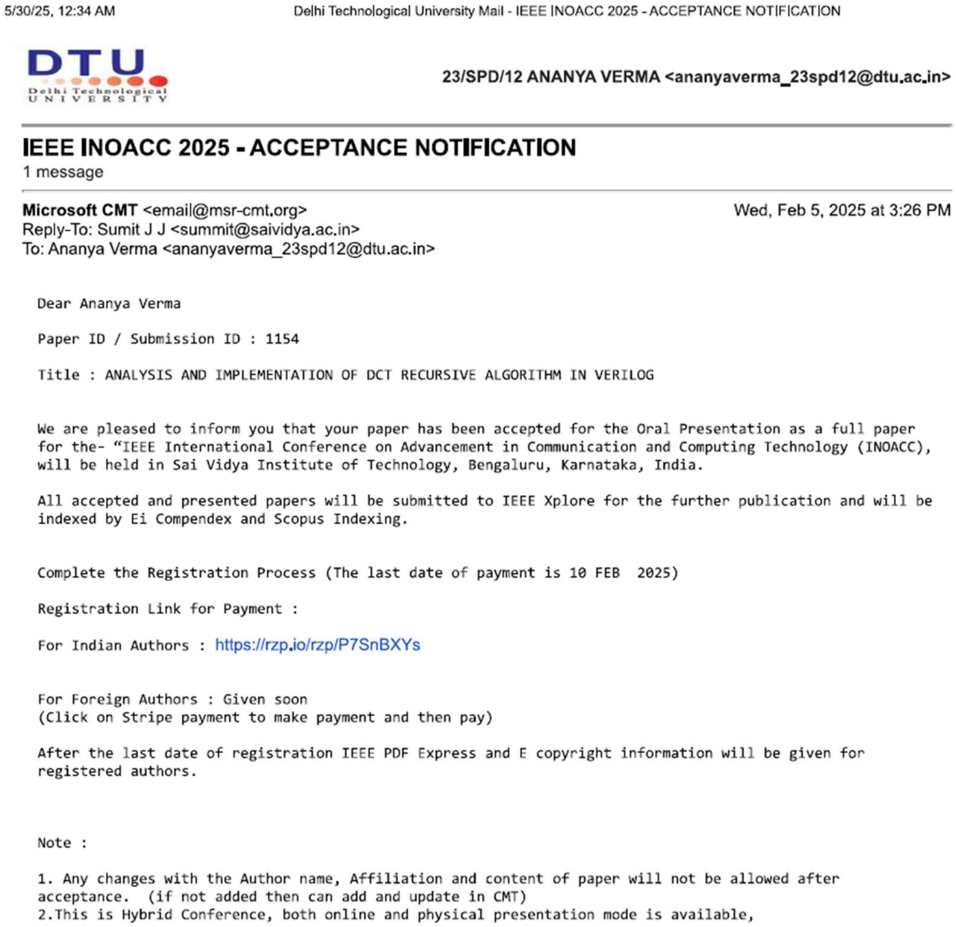
***Index Terms*—Discrete cosine transform (DCT), Infinite impulse response (IIR)-recursive filter structure, Very large-scale integrated circuits (VLSI)**

I. Introduction

The Discrete Cosine Transform is a fundamental tool in digital signal processing, particularly in the fields of image and video compression. The DCT transforms a sequence of data points into a sum of cosine functions oscillating at different frequencies. Its efficiency in compacting energy into a small number of coefficients makes it a preferred choice in compression algorithms like JPEG and MPEG. Among the various types of DCT, the DCT-II is the most widely used. It is characterised by its ability to represent data with minimal information loss, which is critical for applications requiring high fidelity, such as medical imaging and high-definitions video. The mathematical formulation of DCT-II facilities the separation of signal into frequency components, thereby enhancing compression efficiency and reducing computational complexity.

Traditional implementations of the DCT-II algorithm involve direct computation methods and fast Fourier transforms-based techniques. While direct computation is straightforward, it is computationally expansive, making it impractical for real-time applications. FFT-based methods offer improved efficiency but still face challenges in terms of hardware complexity and power consumption, particularly for large data sets. Recursive algorithms have been employed in various signal processing tasks to reduce computational overhead. By reusing intermediate results, recursive methods can significantly cut down on the number of operations required. This is particularly useful in the context of real-time processing, where latency and throughput are critical considerations. The paper [1] involves several key steps to perform the Discrete Cosine Transform, DCT using efficient VLSI architectures, particularly focusing on systolic arrays. These systolic architectures are a parallel computing architecture which are used to efficiently execute regular repetitive computations, particularly those found in digital signal processing and matrix operations. The overall approach focuses on optimizing the computation of DCT by leveraging existing DFT systolic array architectures and employing a prime factor algorithm to reduce computational complexity and enhance the efficiency of VLSI implementations. In paper [2], a method of CORDIC based Radix2 DCT algorithm was also introduced in which fast DCT algorithm addressing several limitations of existing methods was explained. This algorithm integrates the advantages of both CORDIC and Radix2 techniques resulting in a solution that will generate higher order DCT, regular data flow, uniform post scaling factor and in-place computation and arithmetic sequence rotation angles. Compared to the traditional DCT algorithms, this method offers low computational complexity, high scalability and modularity, regular and pipelined implementation which are suitable for real-time applications. Thus, the proposed CORDIC based Radix2 fast DCT algorithm represents a significant step forward offering a promising solution for high-speed pipelined VLSI implementations. In [3], The paper presents a technique based on the fusion of the arithmetic distribution with Loeffler algorithm to overcome the problems such as high silicon occupation area and the number of multipliers to be integrated. It presents the implementation of this algorithm on FPGA Xilinx. As a result, classical

979-8-3315-3108-9/25/$31.00 ©2025 IEEE



5/30/25, 12:34 AM Delhi Technological University Mail - IEEE INOACC 2025 - ACCEPTANCE NOTIFICATION

23/SPD/12 ANANYA VERMA <ananyaverma_23spd12@dtu.ac.in>

IEEE INOACC 2025 - ACCEPTANCE NOTIFICATION

1 message

Microsoft CMT <email@msr-cmt.org> Wed, Feb 5, 2025 at 3:26 PM
Reply-To: Sumit J J <summit@saividya.ac.in>
To: Ananya Verma <ananyaverma_23spd12@dtu.ac.in>

Dear Ananya Verma

Paper ID / Submission ID : 1154

Title : ANALYSIS AND IMPLEMENTATION OF DCT RECURSIVE ALGORITHM IN VERILOG

We are pleased to inform you that your paper has been accepted for the Oral Presentation as a full paper for the- "IEEE International Conference on Advancement in Communication and Computing Technology (INOACC), will be held in Sai Vidya Institute of Technology, Bengaluru, Karnataka, India.

All accepted and presented papers will be submitted to IEEE Xplore for the further publication and will be indexed by Ei Compendex and Scopus Indexing.

Complete the Registration Process (The last date of payment is 10 FEB 2025)

Registration Link for Payment :

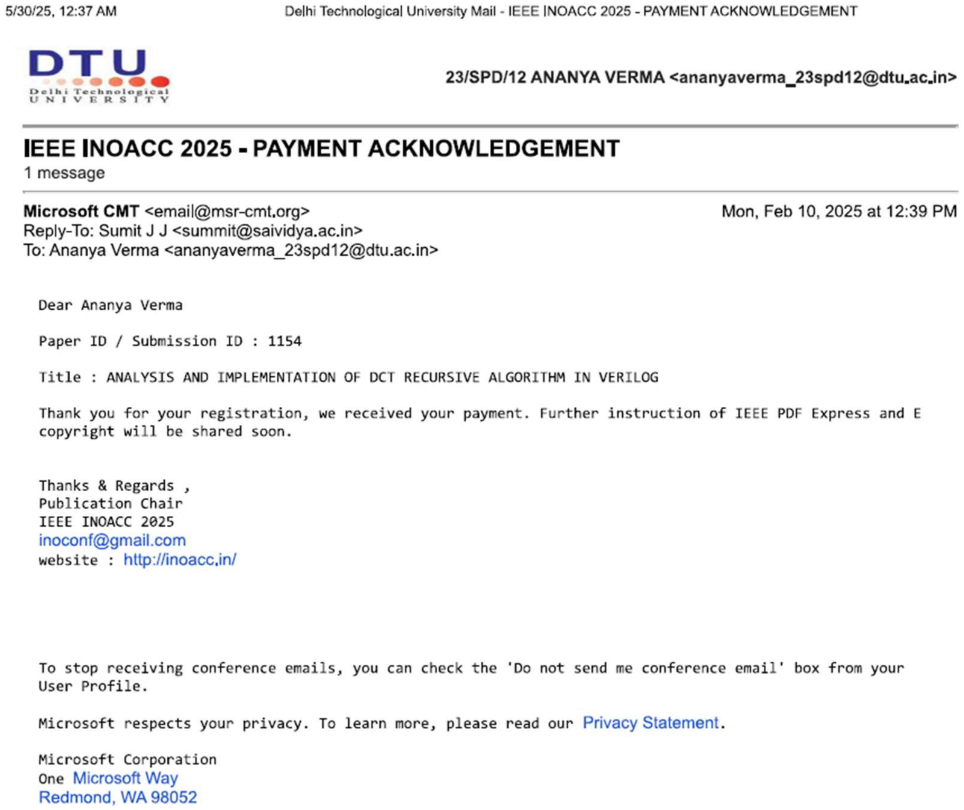
For Indian Authors : https://rzp.io/rzp/P7SnBXYs

For Foreign Authors : Given soon
(Click on Stripe payment to make payment and then pay)

After the last date of registration IEEE PDF Express and E copyright information will be given for registered authors.

Note :

1. Any changes with the Author name, Affiliation and content of paper will not be allowed after acceptance. (if not added then can add and update in CMT)
2.This is Hybrid Conference, both online and physical presentation mode is available,



5/30/25, 12:37 AM Delhi Technological University Mail - IEEE INOACC 2025 - PAYMENT ACKNOWLEDGEMENT

DTU
Delhi Technological
UNIVERSITY

23/SPD/12 ANANYA VERMA <ananyaverma_23spd12@dtu.ac.in>

IEEE INOACC 2025 - PAYMENT ACKNOWLEDGEMENT

1 message

Microsoft CMT <email@msr-cmt.org> Mon, Feb 10, 2025 at 12:39 PM
Reply-To: Sumit J J <summit@saividya.ac.in>
To: Ananya Verma <ananyaverma_23spd12@dtu.ac.in>

Dear Ananya Verma

Paper ID / Submission ID : 1154

Title : ANALYSIS AND IMPLEMENTATION OF DCT RECURSIVE ALGORITHM IN VERILOG

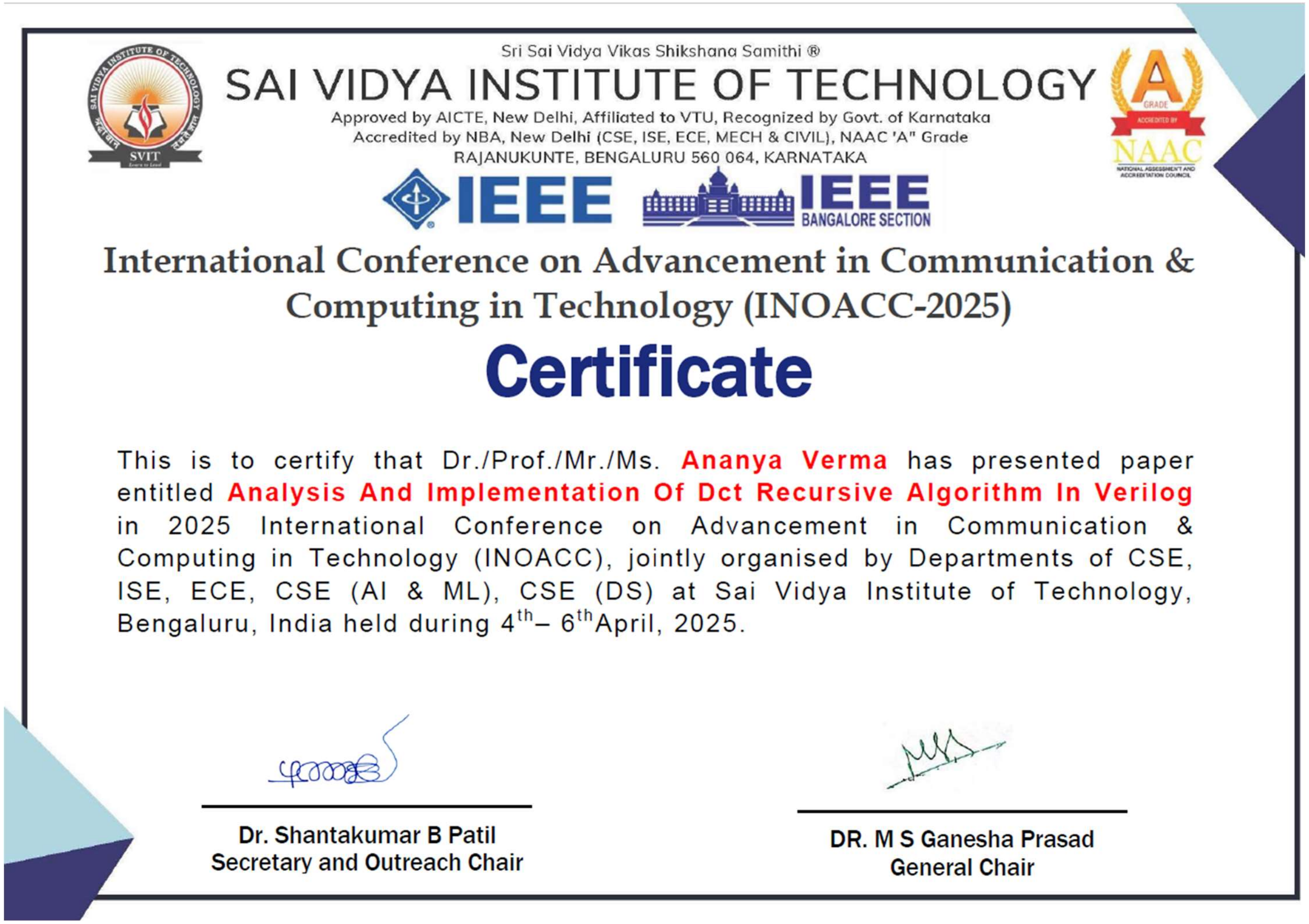
Thank you for your registration, we received your payment. Further instruction of IEEE PDF Express and E copyright will be shared soon.

Thanks & Regards ,
Publication Chair
IEEE INOACC 2025
inoconf@gmail.com
website : http://inoacc.in/

To stop receiving conference emails, you can check the 'Do not send me conference email' box from your User Profile.

Microsoft respects your privacy. To learn more, please read our Privacy Statement.

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052



Sri Sai Vidya Vikas Shikshana Samithi ®

SAI VIDYA INSTITUTE OF TECHNOLOGY

Approved by AICTE, New Delhi, Affiliated to VTU, Recognized by Govt. of Karnataka
Accredited by NBA, New Delhi (CSE, ISE, ECE, MECH & CIVIL), NAAC 'A" Grade
RAJANUKUNTE, BENGALURU 560 064, KARNATAKA

IEEE IEEE BANGALORE SECTION

International Conference on Advancement in Communication & Computing in Technology (INOACC-2025)

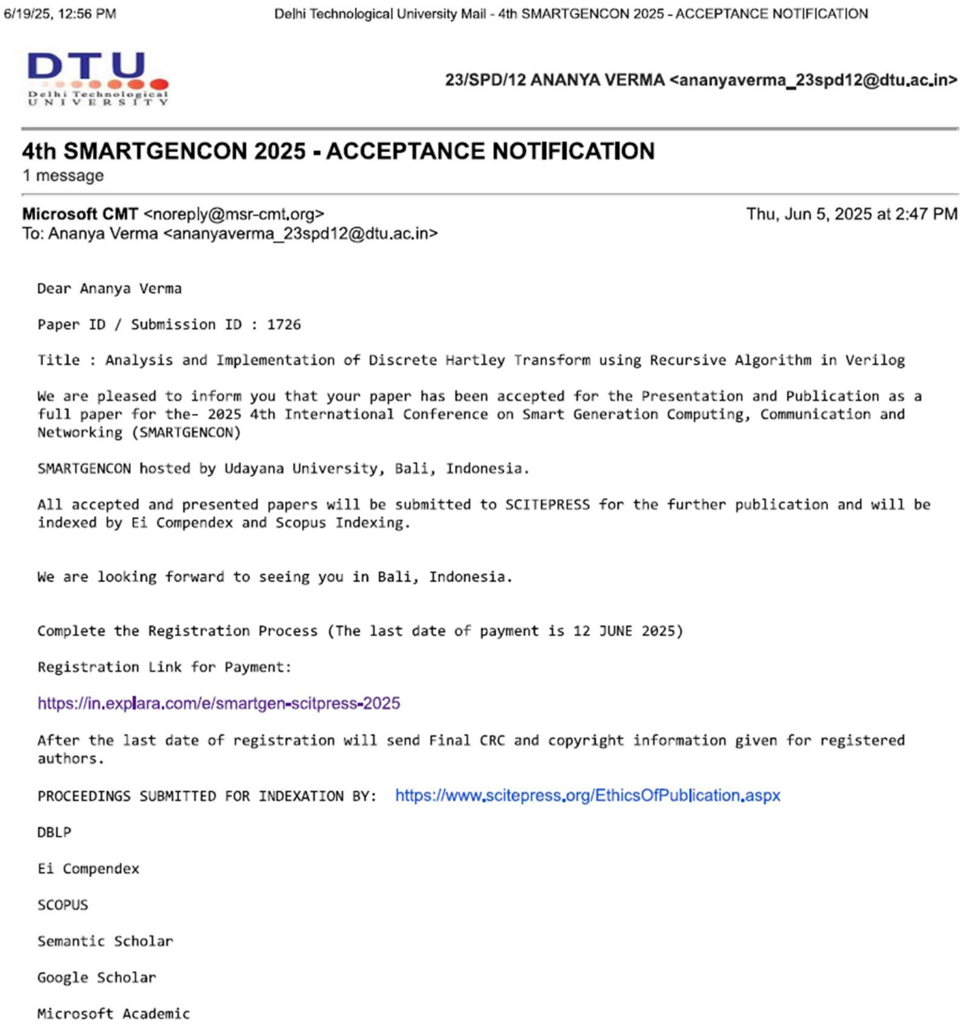
Certificate

This is to certify that Dr./Prof./Mr./Ms. **Ananya Verma** has presented paper entitled **Analysis And Implementation Of Dct Recursive Algorithm In Verilog** in 2025 International Conference on Advancement in Communication & Computing in Technology (INOACC), jointly organised by Departments of CSE, ISE, ECE, CSE (AI & ML), CSE (DS) at Sai Vidya Institute of Technology, Bengaluru, India held during 4th– 6thApril, 2025.

Dr. Shantakumar B Patil
Secretary and Outreach Chair

DR. M S Ganesha Prasad
General Chair

[2] **Ananya Verma, Priyanka Jain “**Analysis and Implementation of Discrete Hartley Transform using Recursive Algorithm in Verilog”, *Submitted Unpublished in the 4th International Conference on Smart Generation Computing, Communication and Networking (SMARTGEN), Bali, Indonesia* **– Accepted & Registered,** publication forthcoming.



6/19/25, 12:56 PM Delhi Technological University Mail - 4th SMARTGENCON 2025 - ACCEPTANCE NOTIFICATION

23/SPD/12 ANANYA VERMA <ananyaverma_23spd12@dtu.ac.in>

4th SMARTGENCON 2025 - ACCEPTANCE NOTIFICATION

1 message

Microsoft CMT <noreply@msr-cmt.org> Thu, Jun 5, 2025 at 2:47 PM
To: Ananya Verma <ananyaverma_23spd12@dtu.ac.in>

Dear Ananya Verma

Paper ID / Submission ID : 1726

Title : Analysis and Implementation of Discrete Hartley Transform using Recursive Algorithm in Verilog

We are pleased to inform you that your paper has been accepted for the Presentation and Publication as a full paper for the- 2025 4th International Conference on Smart Generation Computing, Communication and Networking (SMARTGENCON)

SMARTGENCON hosted by Udayana University, Bali, Indonesia.

All accepted and presented papers will be submitted to SCITEPRESS for the further publication and will be indexed by Ei Compendex and Scopus Indexing.

We are looking forward to seeing you in Bali, Indonesia.

Complete the Registration Process (The last date of payment is 12 JUNE 2025)

Registration Link for Payment:

https://in.explara.com/e/smartgen-scitpress-2025

After the last date of registration will send Final CRC and copyright information given for registered authors.

PROCEEDINGS SUBMITTED FOR INDEXATION BY: https://www.scitepress.org/EthicsOfPublication.aspx

DBLP

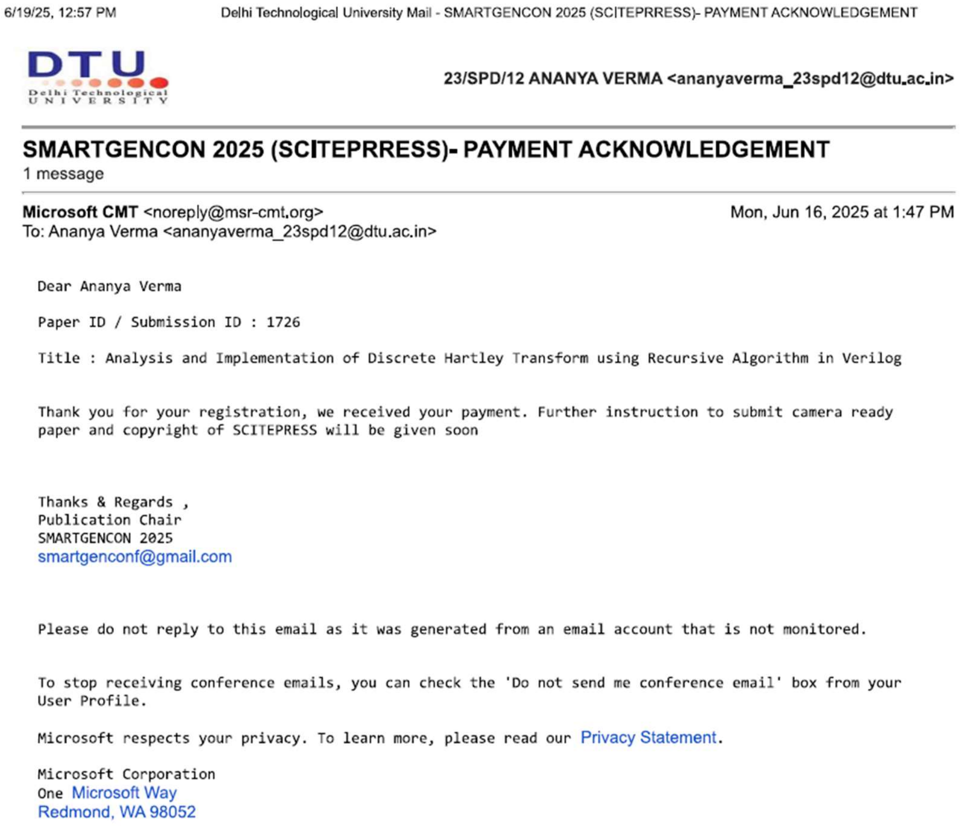
Ei Compendex

SCOPUS

Semantic Scholar

Google Scholar

Microsoft Academic



6/19/25, 12:57 PM Delhi Technological University Mail - SMARTGENCON 2025 (SCITEPRRESS)- PAYMENT ACKNOWLEDGEMENT

DTU Delhi Technological University

23/SPD/12 ANANYA VERMA <ananyaverma_23spd12@dtu.ac.in>

SMARTGENCON 2025 (SCITEPRRESS)- PAYMENT ACKNOWLEDGEMENT

1 message

Microsoft CMT <noreply@msr-cmt.org> Mon, Jun 16, 2025 at 1:47 PM
To: Ananya Verma <ananyaverma_23spd12@dtu.ac.in>

Dear Ananya Verma

Paper ID / Submission ID : 1726

Title : Analysis and Implementation of Discrete Hartley Transform using Recursive Algorithm in Verilog

Thank you for your registration, we received your payment. Further instruction to submit camera ready paper and copyright of SCITEPRESS will be given soon

Thanks & Regards ,
Publication Chair
SMARTGENCON 2025
smartgenconf@gmail.com

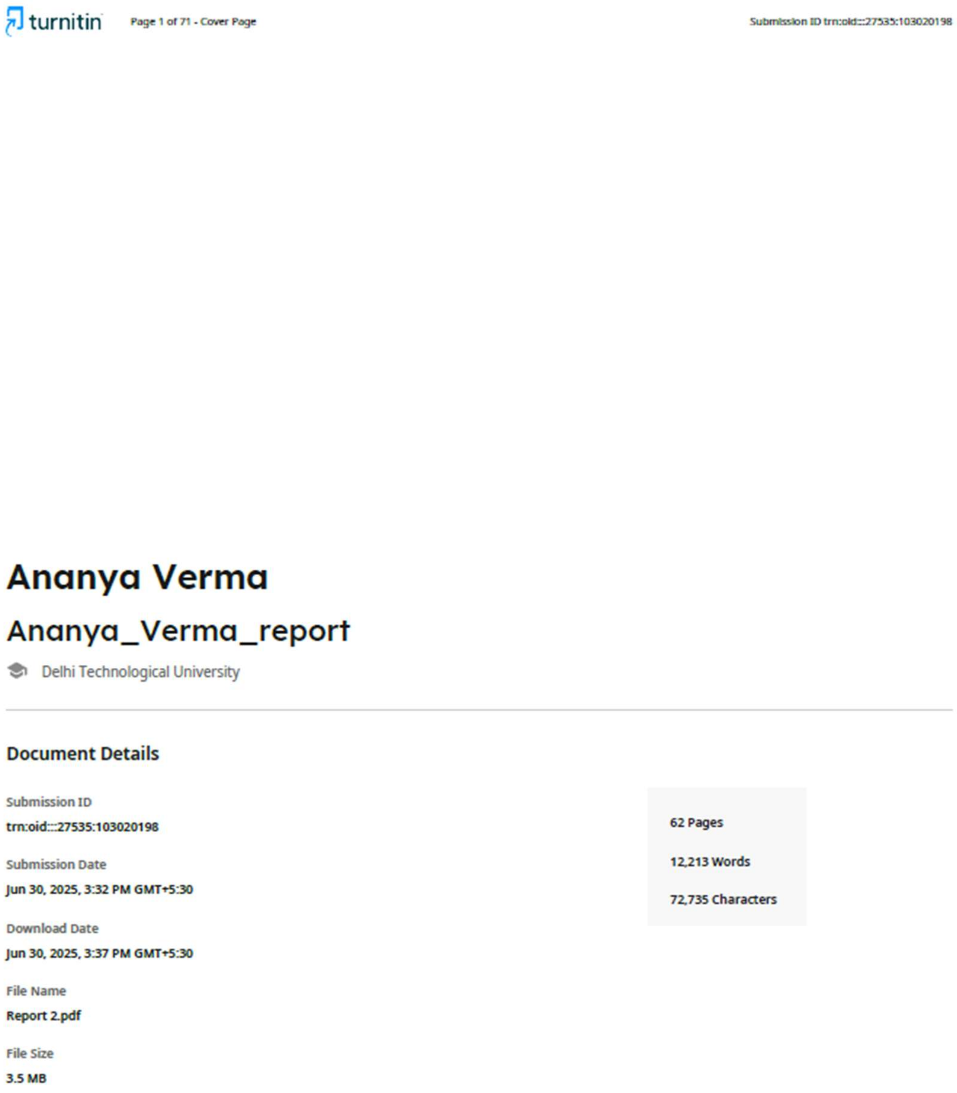
Please do not reply to this email as it was generated from an email account that is not monitored.

To stop receiving conference emails, you can check the 'Do not send me conference email' box from your User Profile.

Microsoft respects your privacy. To learn more, please read our Privacy Statement.

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

PLAGIARISM REPORT



turnitin Page 1 of 71 - Cover Page Submission ID trn:oid:::27535:103020198

Ananya Verma

Ananya_Verma_report

Delhi Technological University

Document Details

Submission ID
trn:oid:::27535:103020198

Submission Date
Jun 30, 2025, 3:32 PM GMT+5:30

Download Date
Jun 30, 2025, 3:37 PM GMT+5:30

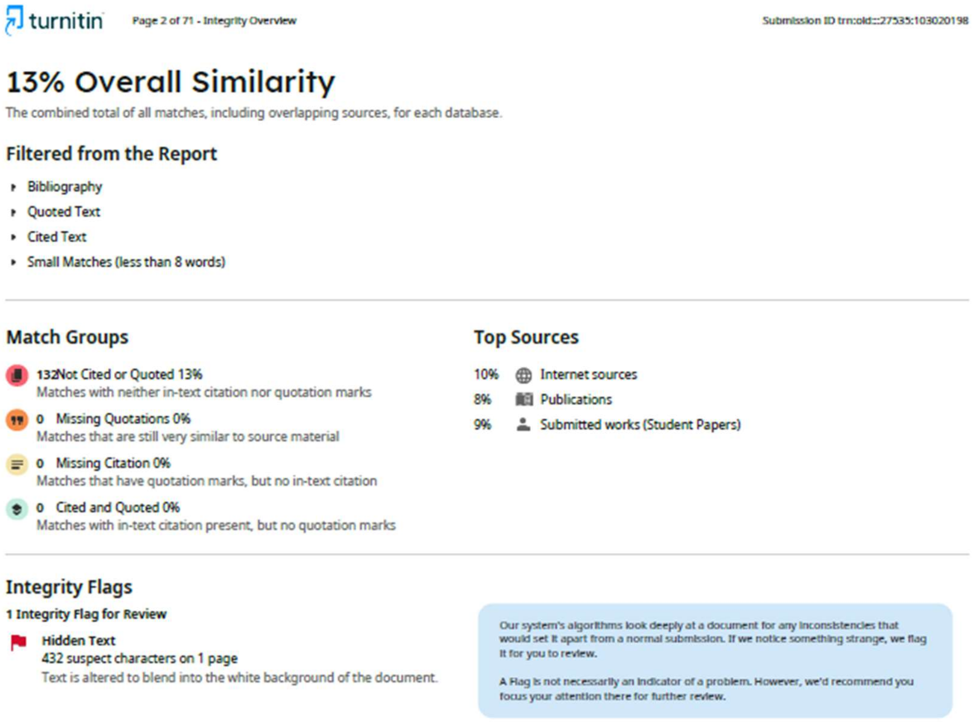
File Name
Report 2.pdf

File Size
3.5 MB

62 Pages

12,213 Words

72,735 Characters



turnitin Page 2 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

13% Overall Similarity

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

Filered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 8 words)

Match Groups

132Not Cited or Quoted 13%
Matches with neither in-text citation nor quotation marks

0 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

10%	Internet sources
8%	Publications
9%	Submitted works (Student Papers)

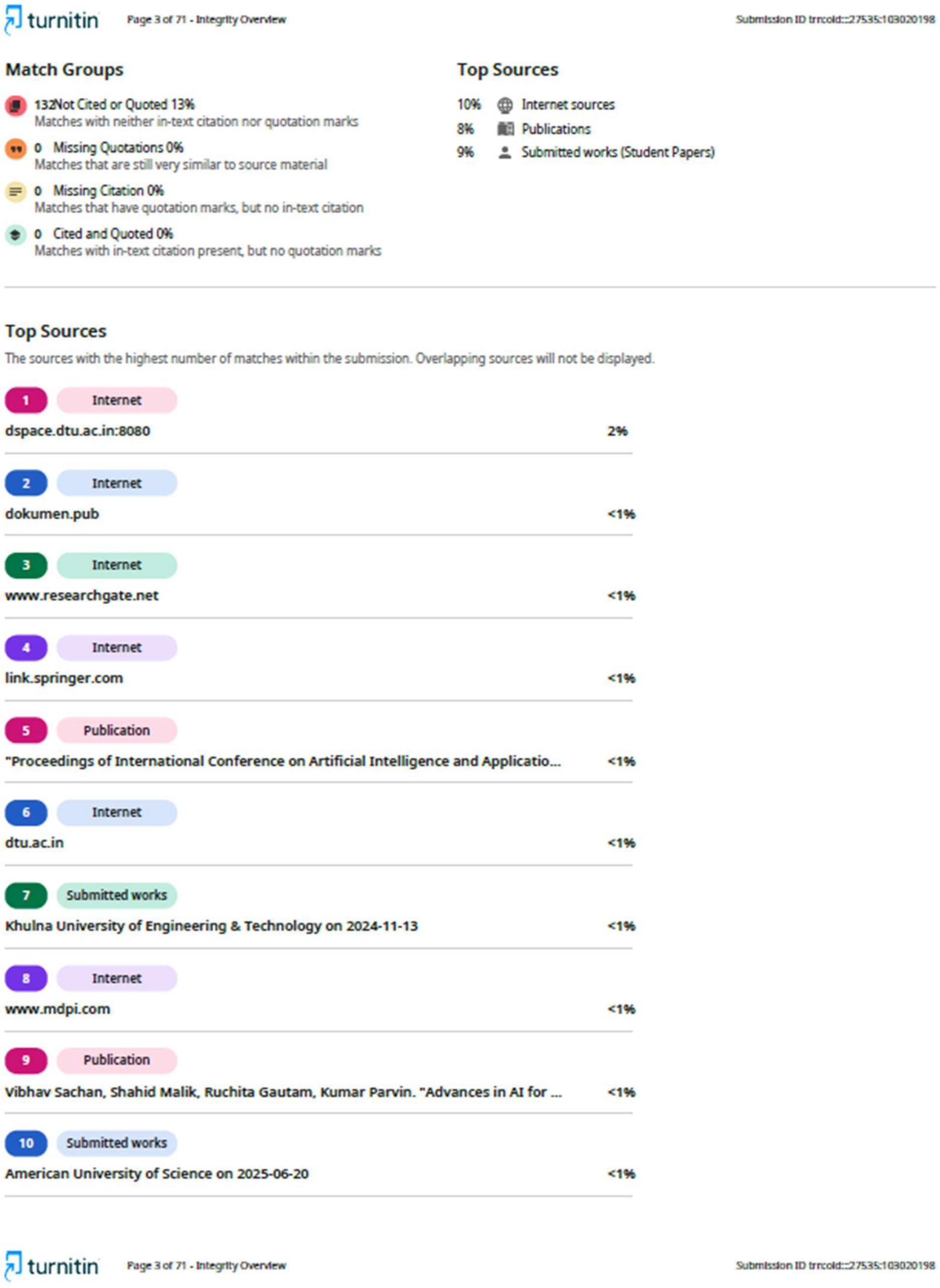
Integrity Flags

1 Integrity Flag for Review

Hidden Text
432 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.



turnitin Page 3 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

Match Groups

132Not Cited or Quoted 13%
Matches with neither in-text citation nor quotation marks

0 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

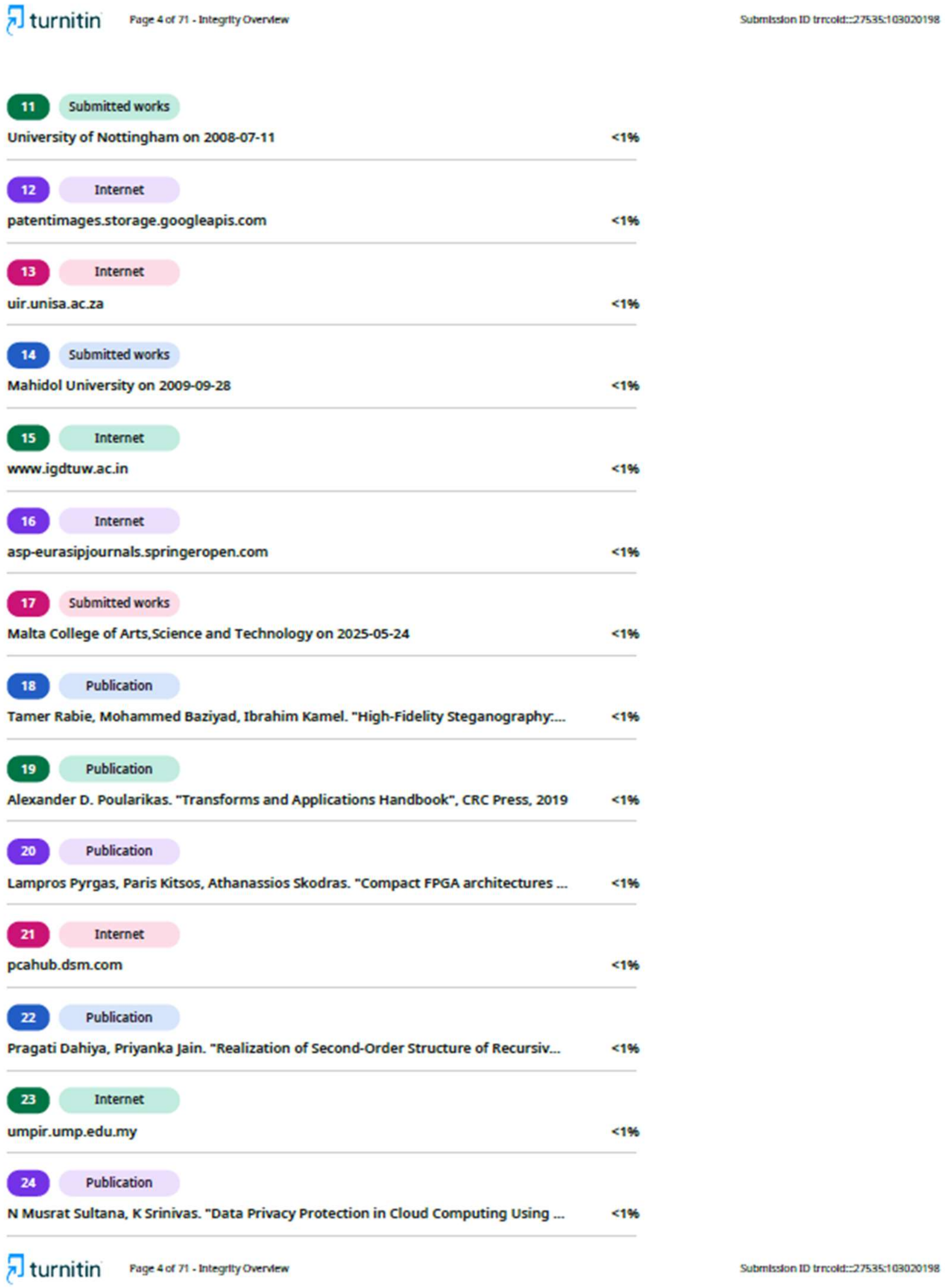
10%	Internet sources
8%	Publications
9%	Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	dspace.dtu.ac.in:8080	2%
2	Internet	dokumen.pub	<1%
3	Internet	www.researchgate.net	<1%
4	Internet	link.springer.com	<1%
5	Publication	"Proceedings of International Conference on Artificial Intelligence and Applicatio...	<1%
6	Internet	dtu.ac.in	<1%
7	Submitted works	Khulna University of Engineering & Technology on 2024-11-13	<1%
8	Internet	www.mdpi.com	<1%
9	Publication	Vibhav Sachan, Shahid Malik, Ruchita Gautam, Kumar Parvin. "Advances in AI for ...	<1%
10	Submitted works	American University of Science on 2025-06-20	<1%

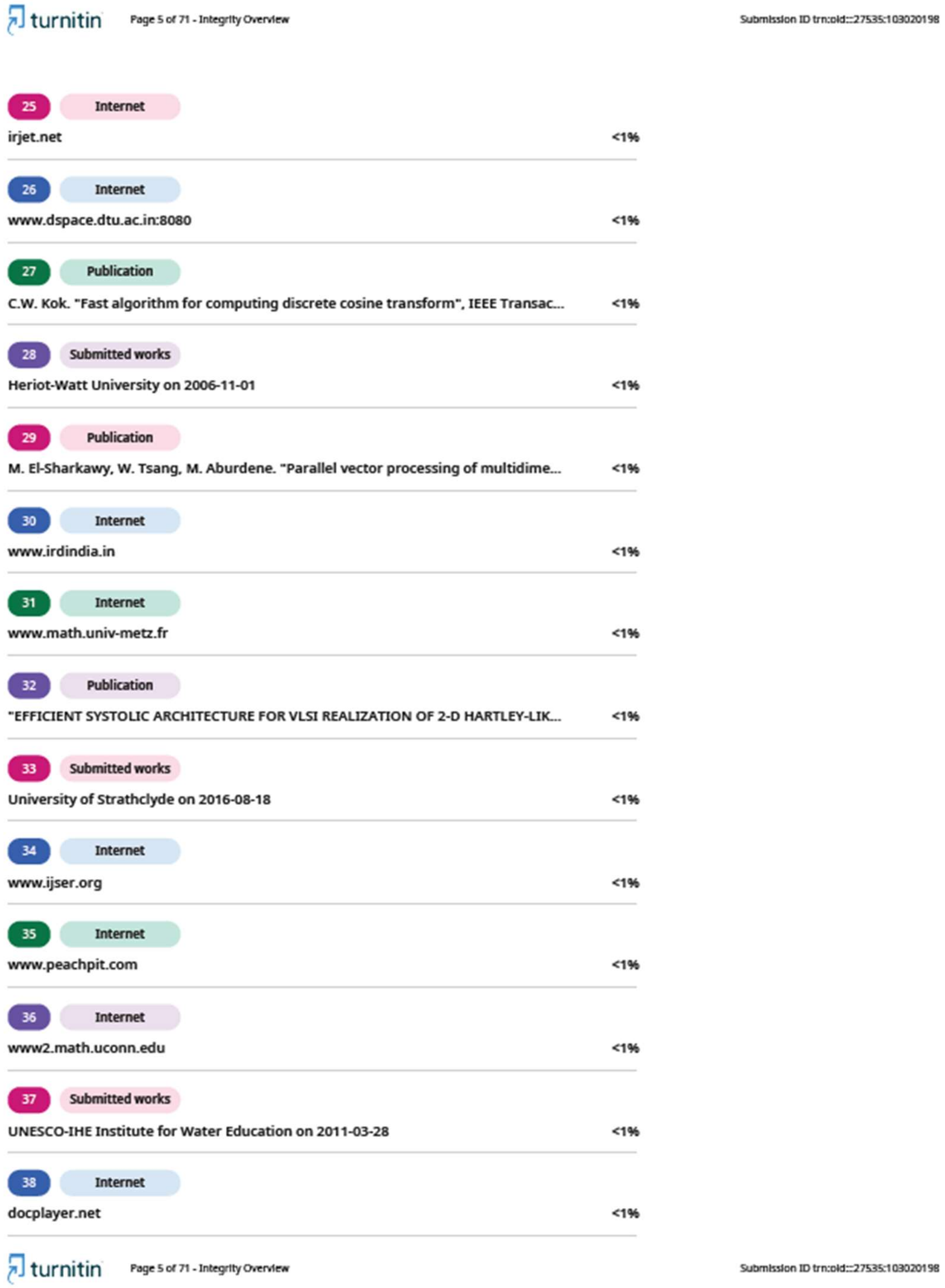
turnitin Page 3 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198



turnitin Page 4 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

11	Submitted works	University of Nottingham on 2008-07-11	<1%
12	Internet	patentimages.storage.googleapis.com	<1%
13	Internet	uir.unisa.ac.za	<1%
14	Submitted works	Mahidol University on 2009-09-28	<1%
15	Internet	www.igdtuw.ac.in	<1%
16	Internet	asp-eurasipjournals.springeropen.com	<1%
17	Submitted works	Malta College of Arts,Science and Technology on 2025-05-24	<1%
18	Publication	Tamer Rabie, Mohammed Baziyad, Ibrahim Kamel. "High-Fidelity Steganography:...	<1%
19	Publication	Alexander D. Poularikas. "Transforms and Applications Handbook", CRC Press, 2019	<1%
20	Publication	Lampros Pyrgas, Paris Kitsos, Athanassios Skodras. "Compact FPGA architectures ...	<1%
21	Internet	pcahub.dsm.com	<1%
22	Publication	Pragati Dahiya, Priyanka Jain. "Realization of Second-Order Structure of Recursiv...	<1%
23	Internet	umpir.ump.edu.my	<1%
24	Publication	N Musrat Sultana, K Srinivas. "Data Privacy Protection in Cloud Computing Using ...	<1%

turnitin Page 4 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

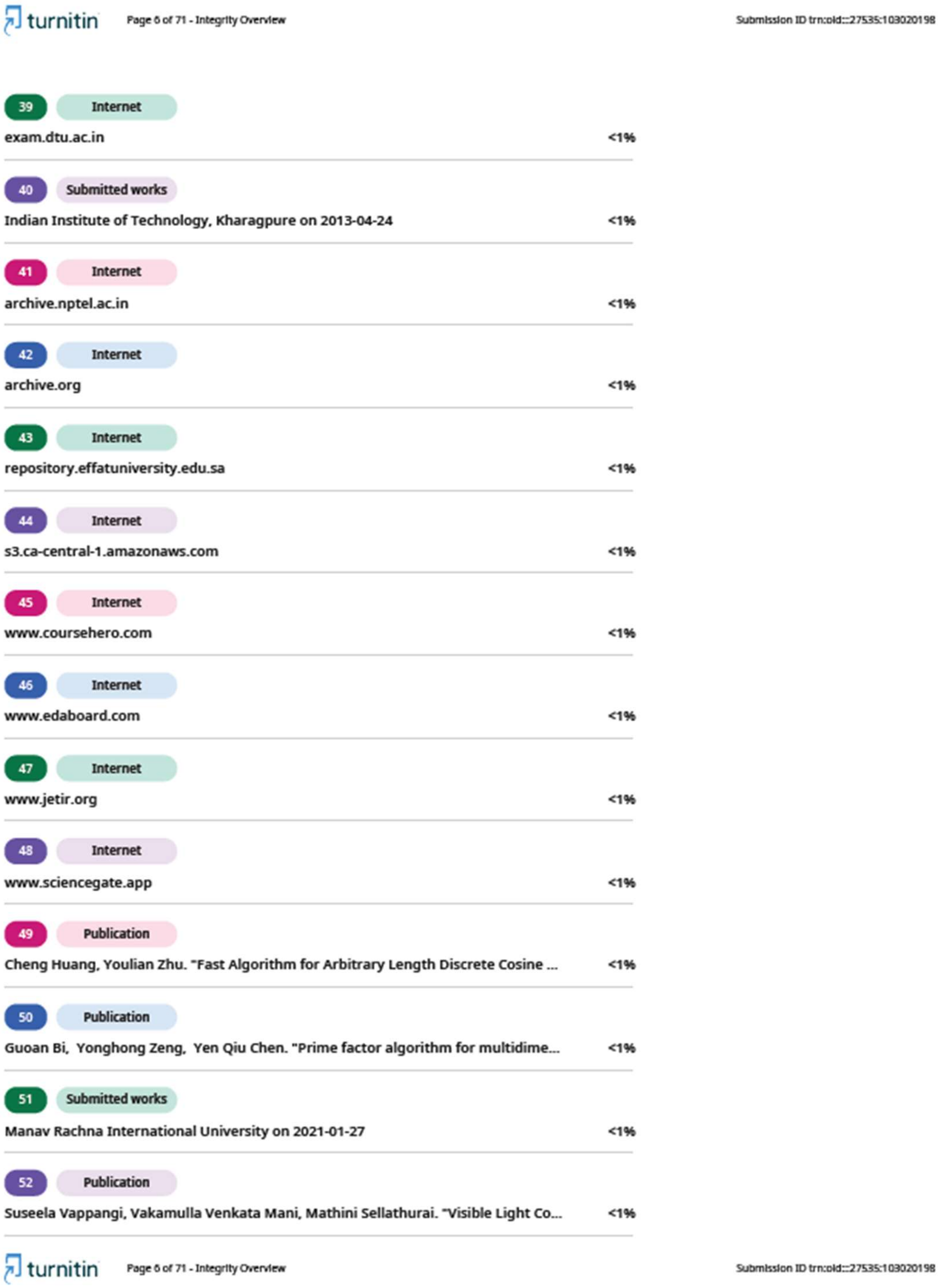


turnitin Page 5 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

25	Internet	irjet.net	<1%
26	Internet	www.dspace.dtu.ac.in:8080	<1%
27	Publication	C.W. Kok. "Fast algorithm for computing discrete cosine transform", IEEE Transac...	<1%
28	Submitted works	Heriot-Watt University on 2006-11-01	<1%
29	Publication	M. El-Sharkawy, W. Tsang, M. Aburdene. "Parallel vector processing of multidime...	<1%
30	Internet	www.irdindia.in	<1%
31	Internet	www.math.univ-metz.fr	<1%
32	Publication	"EFFICIENT SYSTOLIC ARCHITECTURE FOR VLSI REALIZATION OF 2-D HARTLEY-LIK...	<1%
33	Submitted works	University of Strathclyde on 2016-08-18	<1%
34	Internet	www.ijser.org	<1%
35	Internet	www.peachpit.com	<1%
36	Internet	www2.math.uconn.edu	<1%
37	Submitted works	UNESCO-IHE Institute for Water Education on 2011-03-28	<1%
38	Internet	docplayer.net	<1%

turnitin Page 5 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

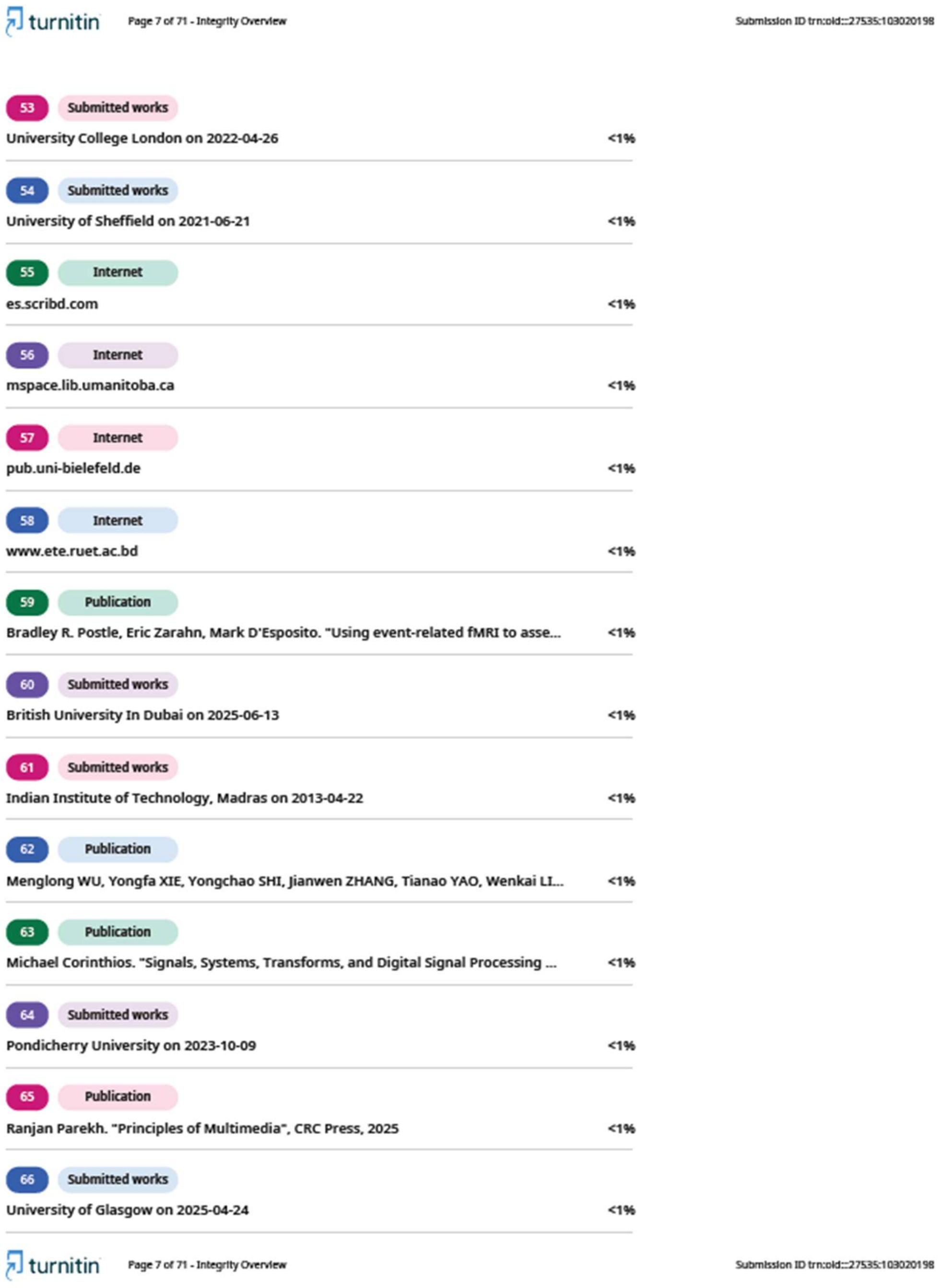
]



turnitin Page 6 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

39	Internet	exam.dtu.ac.in	<1%
40	Submitted works	Indian Institute of Technology, Kharagpure on 2013-04-24	<1%
41	Internet	archive.nptel.ac.in	<1%
42	Internet	archive.org	<1%
43	Internet	repository.effatuniversity.edu.sa	<1%
44	Internet	s3.ca-central-1.amazonaws.com	<1%
45	Internet	www.coursehero.com	<1%
46	Internet	www.edaboard.com	<1%
47	Internet	www.jetir.org	<1%
48	Internet	www.sciencegate.app	<1%
49	Publication	Cheng Huang, Youlian Zhu. "Fast Algorithm for Arbitrary Length Discrete Cosine ...	<1%
50	Publication	Guoan Bi, Yonghong Zeng, Yen Qiu Chen. "Prime factor algorithm for multidime...	<1%
51	Submitted works	Manav Rachna International University on 2021-01-27	<1%
52	Publication	Suseela Vappangi, Vakamulla Venkata Mani, Mathini Sellathurai. "Visible Light Co...	<1%

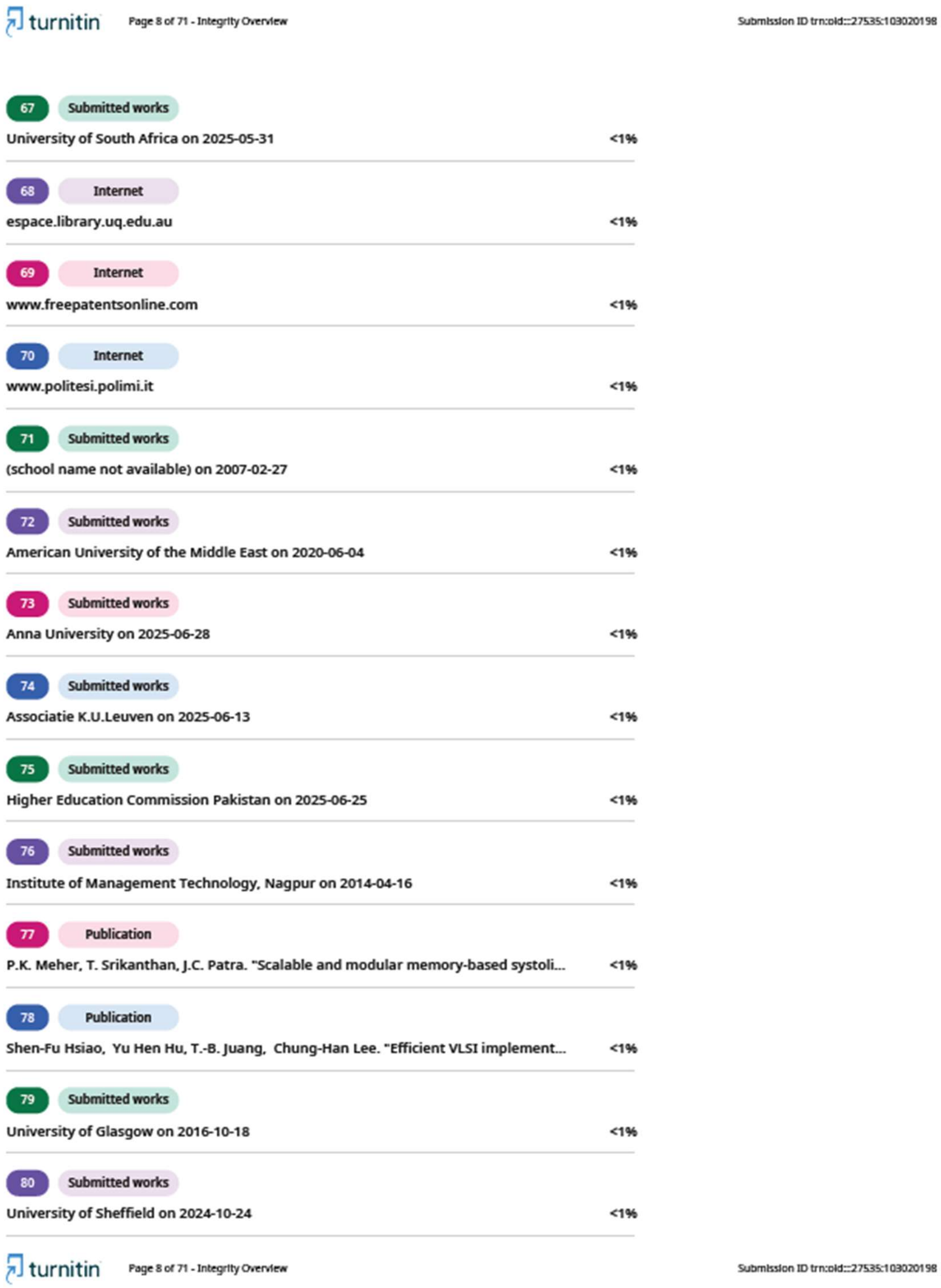
turnitin Page 6 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198



turnitin Page 7 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

53	Submitted works	University College London on 2022-04-26	<1%
54	Submitted works	University of Sheffield on 2021-06-21	<1%
55	Internet	es.scribd.com	<1%
56	Internet	mspace.lib.umanitoba.ca	<1%
57	Internet	pub.uni-bielefeld.de	<1%
58	Internet	www.ete.ruet.ac.bd	<1%
59	Publication	Bradley R. Postle, Eric Zarahn, Mark D'Esposito. "Using event-related fMRI to asse...	<1%
60	Submitted works	British University In Dubai on 2025-06-13	<1%
61	Submitted works	Indian Institute of Technology, Madras on 2013-04-22	<1%
62	Publication	Menglong WU, Yongfa XIE, Yongchao SHI, Jianwen ZHANG, Tianao YAO, Wenkai LI...	<1%
63	Publication	Michael Corinthios. "Signals, Systems, Transforms, and Digital Signal Processing ...	<1%
64	Submitted works	Pondicherry University on 2023-10-09	<1%
65	Publication	Ranjan Parekh. "Principles of Multimedia", CRC Press, 2025	<1%
66	Submitted works	University of Glasgow on 2025-04-24	<1%

turnitin Page 7 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198



turnitin Page 8 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

67 Submitted works

University of South Africa on 2025-05-31 <1%

68 Internet

espace.library.uq.edu.au <1%

69 Internet

www.freepatentsonline.com <1%

70 Internet

www.politesi.polimi.it <1%

71 Submitted works

(school name not available) on 2007-02-27 <1%

72 Submitted works

American University of the Middle East on 2020-06-04 <1%

73 Submitted works

Anna University on 2025-06-28 <1%

74 Submitted works

Associatie K.U.Leuven on 2025-06-13 <1%

75 Submitted works

Higher Education Commission Pakistan on 2025-06-25 <1%

76 Submitted works

Institute of Management Technology, Nagpur on 2014-04-16 <1%

77 Publication

P.K. Meher, T. Srikanthan, J.C. Patra. "Scalable and modular memory-based systoli... <1%

78 Publication

Shen-Fu Hsiao, Yu Hen Hu, T.-B. Juang, Chung-Han Lee. "Efficient VLSI implement... <1%

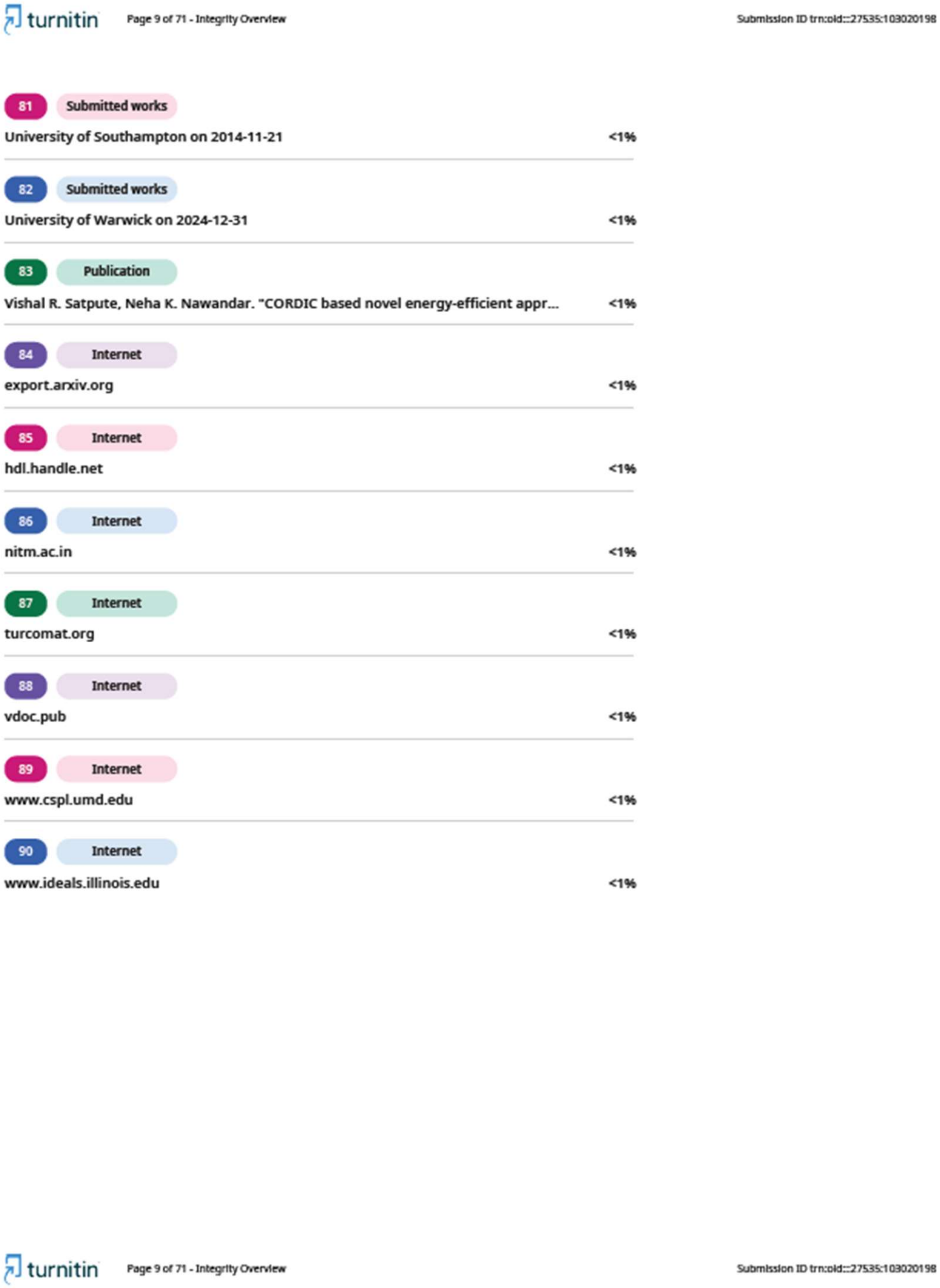
79 Submitted works

University of Glasgow on 2016-10-18 <1%

80 Submitted works

University of Sheffield on 2024-10-24 <1%

turnitin Page 8 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198



turnitin Page 9 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

81	Submitted works	University of Southampton on 2014-11-21	<1%
82	Submitted works	University of Warwick on 2024-12-31	<1%
83	Publication	Vishal R. Satpute, Neha K. Nawandar. "CORDIC based novel energy-efficient appr...	<1%
84	Internet	export.arxiv.org	<1%
85	Internet	hdl.handle.net	<1%
86	Internet	nitm.ac.in	<1%
87	Internet	turcomat.org	<1%
88	Internet	vdoc.pub	<1%
89	Internet	www.cspl.umd.edu	<1%
90	Internet	www.ideals.illinois.edu	<1%

turnitin Page 9 of 71 - Integrity Overview Submission ID trn:oid:::27535:103020198

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-110042.
India

PLAGIARISM VERIFICATION

Title of the Thesis

'Analysis and Implementation of DCT & DHT Recursive Structure in Verilog"

Total Pages **62**

Name of the Scholar **Ananya Verma**

Supervisor **Priyanka Jain**

Department of Electronics and Communication Engineering

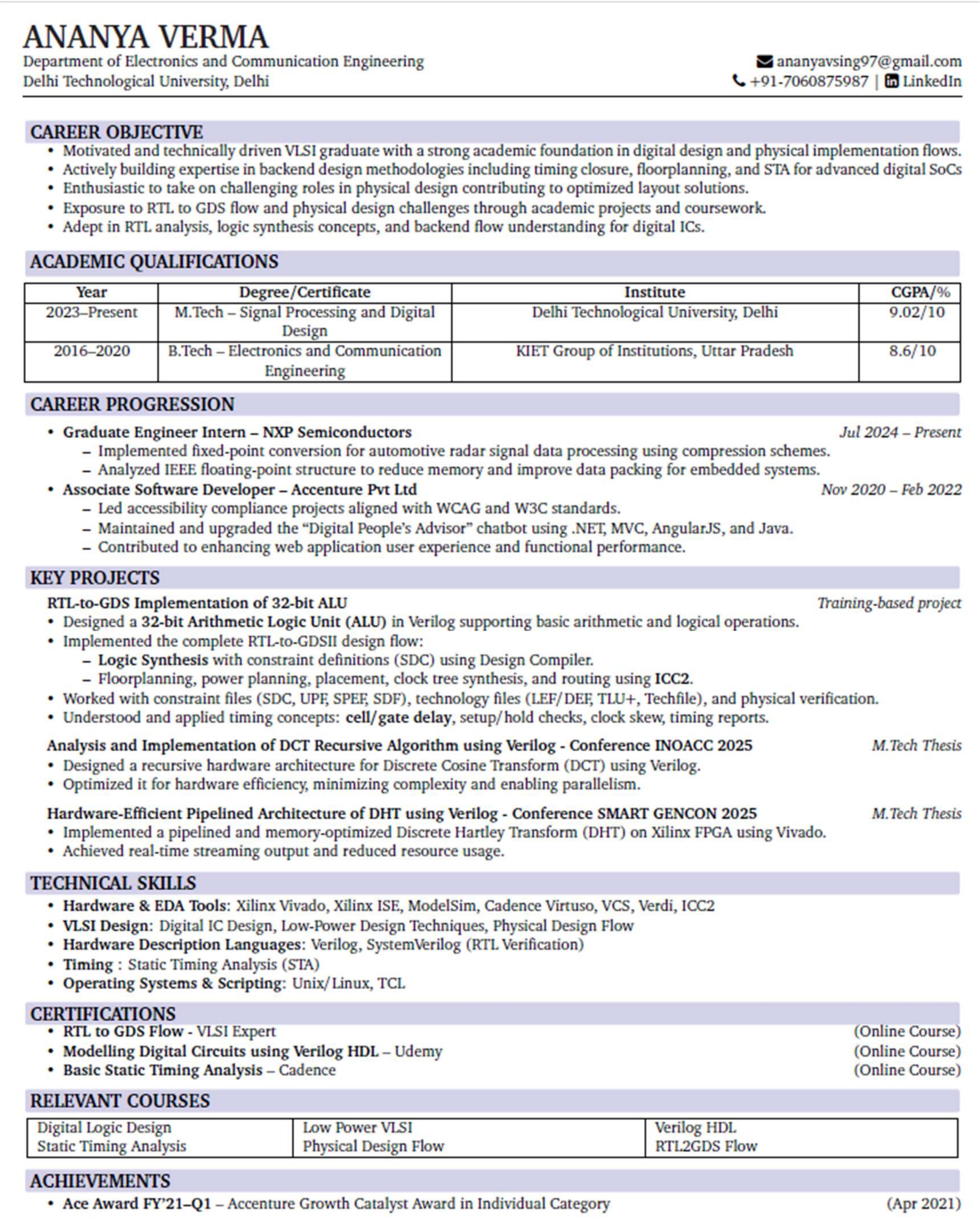
This is to report that the above thesis was scanned for similarity detection. Process and outcome is given below:

Software used: **Turnitin** Similarity Index: **13%** , Total Word Count: **12,213**

Date: **June 30, 2025**

Candidate's Signature **Signature of Supervisor(s)**

CURRICULUM VITAE



ANANYA VERMA

Department of Electronics and Communication Engineering
Delhi Technological University, Delhi

✉ ananyavsing97@gmail.com
☎ +91-7060875987 | LinkedIn

CAREER OBJECTIVE

- Motivated and technically driven VLSI graduate with a strong academic foundation in digital design and physical implementation flows.
- Actively building expertise in backend design methodologies including timing closure, floorplanning, and STA for advanced digital SoCs
- Enthusiastic to take on challenging roles in physical design contributing to optimized layout solutions.
- Exposure to RTL to GDS flow and physical design challenges through academic projects and coursework.
- Adept in RTL analysis, logic synthesis concepts, and backend flow understanding for digital ICs.

ACADEMIC QUALIFICATIONS

Year	Degree/Certificate	Institute	CGPA/%
2023–Present	M.Tech – Signal Processing and Digital Design	Delhi Technological University, Delhi	9.02/10
2016–2020	B.Tech – Electronics and Communication Engineering	KIET Group of Institutions, Uttar Pradesh	8.6/10

CAREER PROGRESSION

- **Graduate Engineer Intern – NXP Semiconductors** *Jul 2024 – Present*
  - Implemented fixed-point conversion for automotive radar signal data processing using compression schemes.
  - Analyzed IEEE floating-point structure to reduce memory and improve data packing for embedded systems.
- **Associate Software Developer – Accenture Pvt Ltd** *Nov 2020 – Feb 2022*
  - Led accessibility compliance projects aligned with WCAG and W3C standards.
  - Maintained and upgraded the "Digital People's Advisor" chatbot using .NET, MVC, AngularJS, and Java.
  - Contributed to enhancing web application user experience and functional performance.

KEY PROJECTS

RTL-to-GDS Implementation of 32-bit ALU *Training-based project*

- Designed a **32-bit Arithmetic Logic Unit (ALU)** in Verilog supporting basic arithmetic and logical operations.
- Implemented the complete RTL-to-GDSII design flow:
  - **Logic Synthesis** with constraint definitions (SDC) using Design Compiler.
  - Floorplanning, power planning, placement, clock tree synthesis, and routing using **ICC2**.
- Worked with constraint files (SDC, UPF, SPEF, SDF), technology files (LEF/DEF, TLU+, Techfile), and physical verification.
- Understood and applied timing concepts: **cell/gate delay**, setup/hold checks, clock skew, timing reports.

Analysis and Implementation of DCT Recursive Algorithm using Verilog - Conference INOACC 2025 *M.Tech Thesis*

- Designed a recursive hardware architecture for Discrete Cosine Transform (DCT) using Verilog.
- Optimized it for hardware efficiency, minimizing complexity and enabling parallelism.

Hardware-Efficient Pipelined Architecture of DHT using Verilog - Conference SMART GENCON 2025 *M.Tech Thesis*

- Implemented a pipelined and memory-optimized Discrete Hartley Transform (DHT) on Xilinx FPGA using Vivado.
- Achieved real-time streaming output and reduced resource usage.

TECHNICAL SKILLS

- **Hardware & EDA Tools**: Xilinx Vivado, Xilinx ISE, ModelSim, Cadence Virtuso, VCS, Verdi, ICC2
- **VLSI Design**: Digital IC Design, Low-Power Design Techniques, Physical Design Flow
- **Hardware Description Languages**: Verilog, SystemVerilog (RTL Verification)
- **Timing** : Static Timing Analysis (STA)
- **Operating Systems & Scripting**: Unix/Linux, TCL

CERTIFICATIONS

- **RTL to GDS Flow** - VLSI Expert (Online Course)
- **Modelling Digital Circuits using Verilog HDL** – Udemy (Online Course)
- **Basic Static Timing Analysis** – Cadence (Online Course)

RELEVANT COURSES

Digital Logic Design	Low Power VLSI	Verilog HDL
Static Timing Analysis	Physical Design Flow	RTL2GDS Flow

ACHIEVEMENTS

- **Ace Award FY'21–Q1** – Accenture Growth Catalyst Award in Individual Category (Apr 2021)