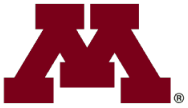


UNIVERSITY OF MINNESOTA

5327 VLSI DESIGN LAB

Project Report II

Submitted by : *Samson Hruday Chinta*

Email : *chint078@umn.edu*

Submitted to :

Prof. Yu Cao ,

Balagopal Anoop

Department of Electrical and Computer Engineering

UMN ,Twincities

Contents

VLSI DESIGN LABORATORY REPORT II

1.1 Introduction and Applications :

The Discrete Cosine Transform (DCT) is a mathematical technique widely used in digital signal and image processing to convert spatial data into the frequency domain. This transformation enables efficient representation and manipulation of data by separating low-frequency components, which usually contain most of the significant information, from high-frequency components, which often represent fine details or noise. The 8x8 DCT focuses on processing data in small 8x8 blocks, making it computationally efficient and suitable for real-world applications.
Each block is transformed into a set of coefficients that represent varying levels of frequency. These coefficients provide a compact representation, where a few low-frequency terms capture most of the block's energy. This compactness is one of the reasons why 8x8 DCT is the foundation of many lossy compression algorithms. The block-based approach makes it particularly effective for images and videos, where adjacent pixels or frames often have significant correlations.

Image Compression: JPEG, where the DCT transforms spatial pixel values into frequency coefficients for better compression.
Video Compression: Standards like MPEG and H.264, where DCT helps reduce redundancy in video frames.
Signal Processing: Applications requiring frequency-domain analysis of 2D signals.

There are numerous architectures for computing DCT and each of them variers with the application and the each of them have different algorithms.

Approach 1: Matrix Multiplication

The 2D DCT is computed by applying a 1D DCT transformation first on the rows and then on the columns of the input matrix. For an $N \times N$ input matrix $\mathbf{X}$, the 2D DCT is defined as:

$$\mathbf{Y} = \mathbf{C} \cdot \mathbf{X} \cdot \mathbf{C}^T$$

where:

- $\mathbf{Y}$ is the resulting $N \times N$ DCT matrix,
- $\mathbf{C}$ is the DCT transform matrix of size $N \times N$, and
- $\mathbf{C}^T$ denotes the transpose of $\mathbf{C}$.

Step-by-Step Computation
1. Define the DCT Matrix $\mathbf{C}$: The elements of $\mathbf{C}$ are defined as:

$$C_{i,j} = \sqrt{\frac{2}{N}} \cos\left(\frac{\pi(2j\ 1)i}{2N}\right), \quad i, j = 0, 1, \ldots, N-1$$

with $C_{0,j} = \sqrt{\frac{1}{N}}$ to account for the scaling factor in the first row.

2.Apply Row-wise Transformation: Multiply the input matrix **X** by **C** to obtain an intermediate matrix:

$$\mathbf{Z} = \mathbf{C} \cdot \mathbf{X}$$

3. Apply Column-wise Transformation: Multiply the result by the transpose of **C** to complete the transformation:

$$\mathbf{Y} = \mathbf{Z} \cdot \mathbf{C}^T$$

4. The resulting matrix **Y** contains the 2D DCT coefficients, which represent the input data in the frequency domain. Low-frequency components are concentrated in the top-left corner of **Y**.This can be extremely at a disadvantage as it is very inefficient for hardware implementations and involves a lot of adds and multiplications.

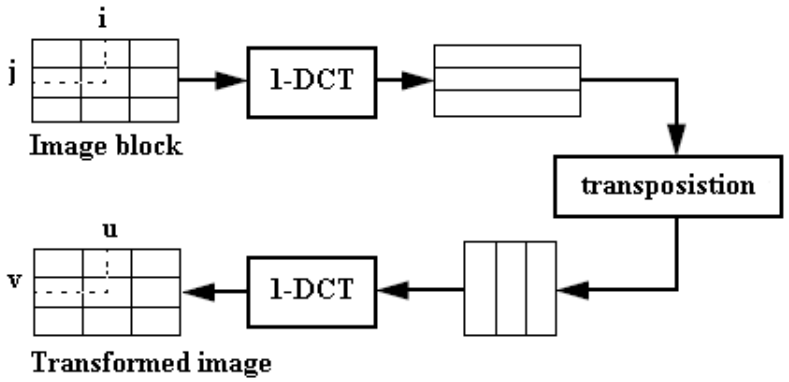


Figure 1: Generic DCT Transformation

Approach 2: Butterfly Approach

The Butterfly structure is a popular method for efficiently computing Discrete Cosine Transform (DCT) coefficients, especially in hardware implementations. Butterfly-based DCT structures divide the computation into stages with simple operations (like additions, subtractions, and shifts), reducing the complexity of multipliers. This approach enables high-speed DCT computations with minimized area and power consumption, making it highly suitable for hardware implementations, such as in VLSI circuits for image and video compression.

The butterfly approach for 1D DCT is efficient for hardware implementations, reducing the need for multipliers by using simpler additions and subtractions, making it suitable for VLSI applications.

1.2 Proposed Architecture(Revised 2D DCT 32 point to 2D 8 point)

This architecture is a Mixture of above mentioned approaches Implements a 8 point 2D DCT through two 1D DCT passes.
The first pass computes the DCT row-wise.The output of the first pass is transposed and fed to the second pass, which computes the DCT column-wise.The overall design ensures that each step is modular and leverages butterfly arithmetic structures to minimize computational complexity. But Computing 1D is based on the Butterfly Approximation . Computed 1D DCT is an approximation of loefflers 8 point DCT computation

but extented to 8 point by instantiating lower level modules .This architecture also has FSM to control the flow of design .which is explained below in detailed .

The design implements a 2D Discrete Cosine Transform (DCT) suitable for image and video processing applications. The architecture is structured to handle an 8x8 input matrix, leveraging the separable property of the DCT to split the 2D computation into two consecutive 1D transforms: row-wise DCT followed by column-wise DCT. The design incorporates FSM-based control, efficient data flow, and pipelined processing to improve throughput.

Main Design Modules:

The design consists of the following key modules.

a.Top-Level Module (dct_8_2d):

Manages the control flow using an FSM. Buffers the input matrix and orchestrates row-wise and column-wise DCT operations.Outputs the final 8x8 DCT-transformed matrix.Tracks computation and loading completion signals.

b.Row and Column 1D DCT (dct_8_module): Performs a single 1D DCT on an 8-element input vector. Decomposes the DCT into even and odd components for efficient computation. Invokes a smaller dct_4_point module to compute results.

c.4-Point DCT Module (dct_4_point): Handles a 1D 4-point DCT for even and odd components. Uses a mix of additions, subtractions, and scaling factors to compute outputs.

d.FSM (Finite State Machine):

Controls the data flow across the design stages:

1.LOAD: Loads an 8x8 input matrix row-by-row.

2.ROW_DCT: Computes row-wise DCT on the loaded matrix.

3. TRANSPOSE: Transposes the intermediate matrix for column DCT.

4.COL_DCT: Computes column-wise DCT to complete the 2D DCT.

1D DCT Modules (dct8, dct4)

The 1D DCT modules break the computation into smaller segments. Each module follows a butterfly structure to compute sums and differences of pairs of inputs.DCT8 Computes the 8-point DCT by splitting the input into 4-point segments . Internally calls DCT4 for these segments. This decomposition allows parallel processing and minimizes multiplications, following a method similar to the Loeffler algorithm.

Transpose Module

Purpose: Transposes the intermediate output from the first DCT pass TO prepare for the second DCT pass.he transpose operation reorders the data, ensuring that rows become columns for the second pass. FSM Control: The FSM triggers trans_start and trans_done to handle the transfer of data to/from the transpose buffer.

1.3 Pipelineing

The dct_8_2d module implements an 8x8 2D Discrete Cosine Transform (DCT) using a pipelined architecture controlled by a finite state machine (FSM). The pipeline consists of distinct stages: first, the ROW_DCT stage computes the row-wise 1D DCT in parallel for all 8 rows using 8 instances of the ‘dct_8_module‘,

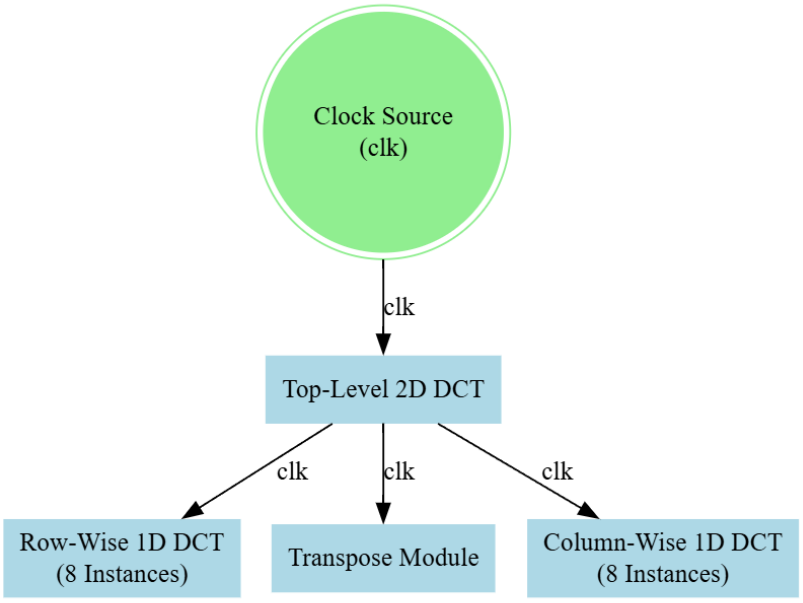


Figure 2: Clock Tree

which leverages symmetry by decomposing the 8-point DCT into even-odd computations with submodules for 4-point DCTs (‘dct_4_point‘). Once row DCTs are computed, the TRANSPOSE stage rearranges the data by swapping rows and columns, preparing it for the next stage. The COL_DCT stage then processes the transposed data, again in parallel for all 8 columns using the same ‘dct_8_module‘, completing the 2D DCT computation. The FSM transitions through states (‘IDLE‘, ‘LOAD‘, ‘ROW_DCT‘, ‘TRANSPOSE‘, ‘COL_DCT‘) to ensure sequential execution without hazards, while parallelism in row and column computations maximizes throughput. The final DCT output matrix is updated at the end of the COL_DCT stage, achieving high efficiency and modularity through reusable submodules and pipelined execution.

1.4 Control and Data Flow Synchronization

The architecture ensures synchronization between different modules using the FSM and counter.

Data Flow:

- Input data flows into the first DCT stage (dct8).
- The intermediate results are scaled and transposed.
- The transposed data goes through the second DCT pass.

FSM Synchronization:

Each stage begins only after the previous one completes. The FSM uses the counter to determine when to move to the next stage. Control signals ensure that only the required modules are active at each stage, saving power.

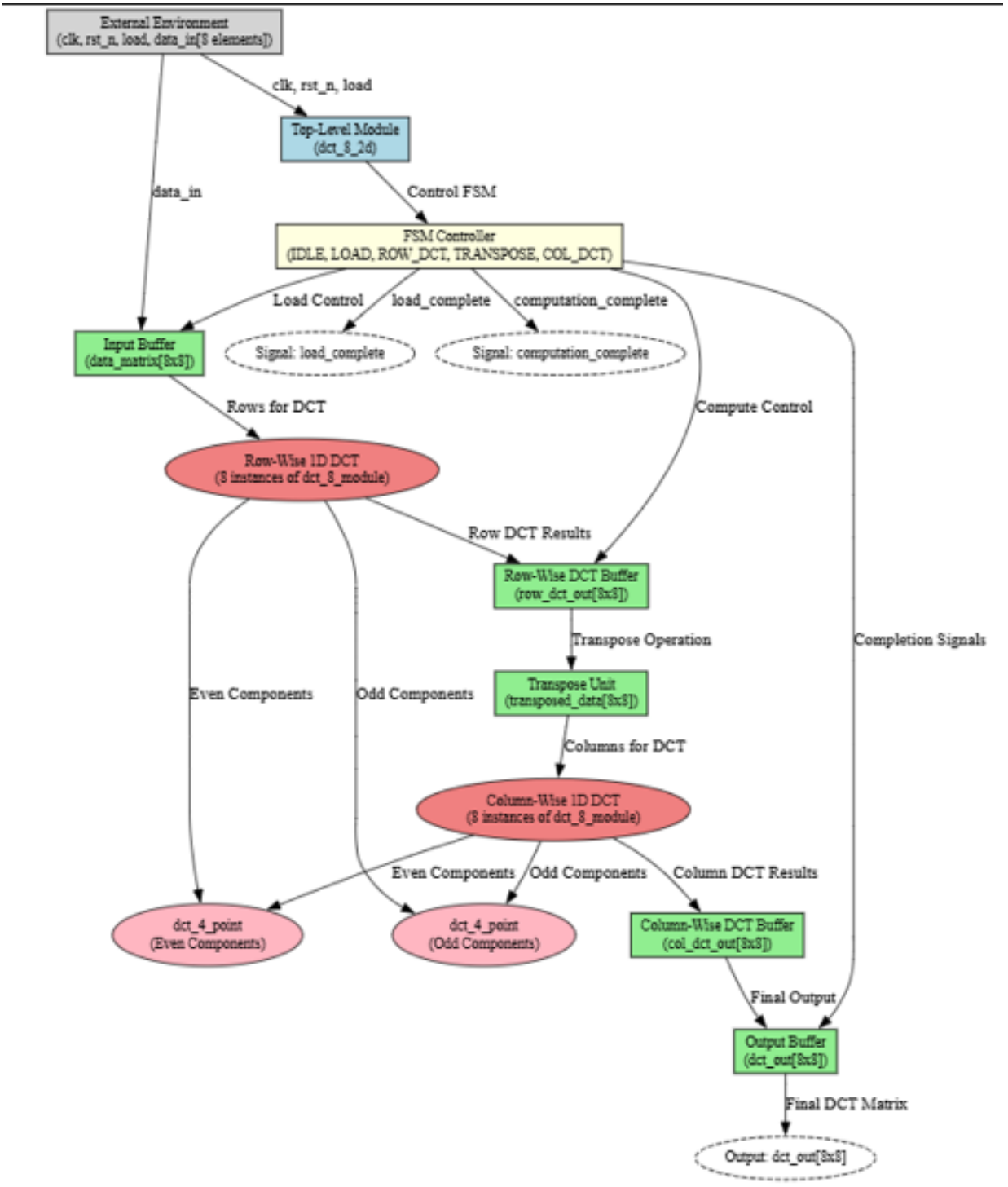


Figure 3: Architecture and Flow Chart

Port Name	Bit-Width	Direction	Description
clk	1	Input	Clock signal for synchronization.
rst_n	1	Input	Active-low reset to initialize the design.
load	1	Input	Signal to start loading input data.
data_in	DATA_WIDTH x 8 x 8	Input	8x8 matrix of input data (e.g., image pixel values).
test_se	1	Input	Scan enable for testing (used for DFT).
test_si	1	Input	Scan input for testing (used for DFT).
test_so	1	Output	Scan output for testing (used for DFT).
dct_out	DATA_WIDTH x 8 x 8	Output	8x8 matrix of DCT output coefficients.
load_complete	1	Output	Signal indicating input data loading is complete.

Table 1: Interface between the 2D DCT Design and the External System

1.5 Verification of logic using Python

The Python script was utilized to verify the implementation of the 2D Discrete Cosine Transform (DCT) logic using an 8x8 chessboard pattern as input. The script computes the DCT transform and then reconstructs the original image by applying the built-in idct function from the scipy.fftpack library.

This verification process confirmed the correctness of the implemented DCT algorithm, as the reconstructed image closely matches the original input after applying inverse DCT. The script also visualizes the input image, the computed DCT output, and the reconstructed image, ensuring the logic's accuracy and demonstrating the ability to faithfully reproduce the input from its frequency-domain representation(with few mis matches).Which can reduced if we can clamp the output to certain levels .

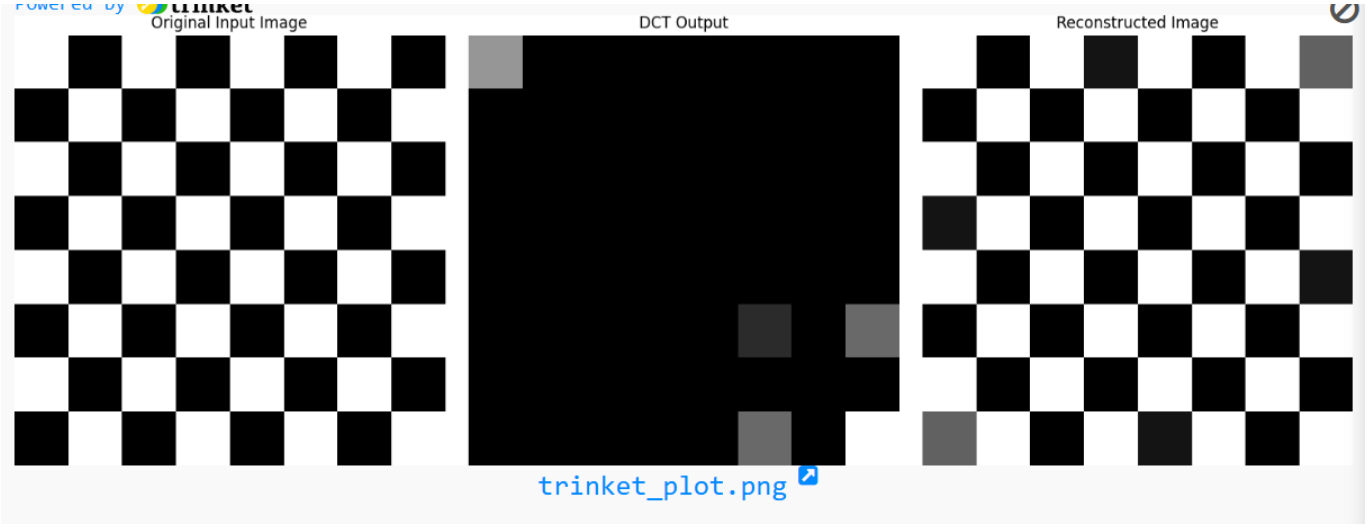


Figure 4: DCT and Reconstruction of image

1.6 RTL Results and comparision with test bench

```
1
2  `timescale 1ns/1ps
3
4
5  // Submodules: 1D DCT
6  module dct_8_module #(
7      parameter DATA_WIDTH = 12
8  )(
9      input  logic clk,
10     input  logic rst_n,
11     input  logic [DATA_WIDTH-1:0] data_in[0:7],
12     output logic signed [DATA_WIDTH-1:0] dct_out[0:7]
13 );
14
15     logic signed [DATA_WIDTH-1:0] even[0:3];
16     logic signed [DATA_WIDTH-1:0] odd[0:3];
17
18     always_ff @(posedge clk or negedge rst_n) begin
19         if (!rst_n) begin
20             for (int i = 0; i < 4; i++) begin
21                 even[i] <= 0;
22                 odd[i] <= 0;
23             end
24         end else begin
25             for (int i = 0; i < 4; i++) begin
26                 even[i] <= $signed(data_in[i]) + $signed(data_in[7 - i]);
27                 odd[i]  <= $signed(data_in[i]) - $signed(data_in[7 - i]);
28             end
29         end
30     end
31
32     dct_4_point #(.DATA_WIDTH(DATA_WIDTH)) dct_even_inst (.clk(clk), ....
       rst_n(rst_n), .data_in(even), .dct_out(dct_out[0:3]));
33     dct_4_point #(.DATA_WIDTH(DATA_WIDTH)) dct_odd_inst  (.clk(clk), ....
       rst_n(rst_n), .data_in(odd),  .dct_out(dct_out[4:7]));
34
35 endmodule
36
37 // Submodule: 1D 4-point DCT
38 module dct_4_point #(
39     parameter DATA_WIDTH = 12
40 )(
41     input  logic clk,
42     input  logic rst_n,
43     input  logic [DATA_WIDTH-1:0] data_in[0:3],
44     output logic signed [DATA_WIDTH-1:0] dct_out[0:3]
45 );
46
```

```
47      logic signed [DATA_WIDTH-1:0] even[0:1];
48      logic signed [DATA_WIDTH-1:0] odd[0:1];
49
50      always_ff @(posedge clk or negedge rst_n) begin
51          if (!rst_n) begin
52              even[0] <= 0; even[1] <= 0;
53              odd[0] <= 0;  odd[1] <= 0;
54          end else begin
55              even[0] <= $signed(data_in[0]) + $signed(data_in[3]);
56              even[1] <= $signed(data_in[1]) + $signed(data_in[2]);
57              odd[0]  <= $signed(data_in[0]) - $signed(data_in[3]);
58              odd[1]  <= $signed(data_in[1]) - $signed(data_in[2]);
59          end
60      end
61
62      always_ff @(posedge clk or negedge rst_n) begin
63          if (!rst_n) begin
64              for (int i = 0; i < 4; i++) dct_out[i] <= 0;
65          end else begin
66              dct_out[0] <= (even[0] + even[1]) / 2;
67              dct_out[2] <= (even[0] - even[1]) / 2;
68              dct_out[1] <= (odd[0] * 118 + odd[1] * 49) / 128;
69              dct_out[3] <= (odd[0] * 49 - odd[1] * 118) / 128;
70          end
71      end
72  endmodule
```

3 types of test benches were developed to validate the functionality of the design. In the first test bench, only a dc constant of the input matrix was provided, and the corresponding results are attached. This initial step helped verify functionality of the design. In the second test bench, a Complete 8X8 Ramp matrix was used as input to thoroughly assess the system's performance and correctness. The results of this full matrix test are also attached for reference .

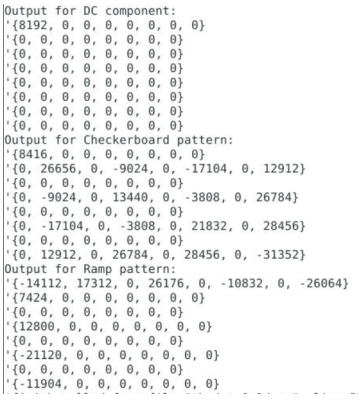
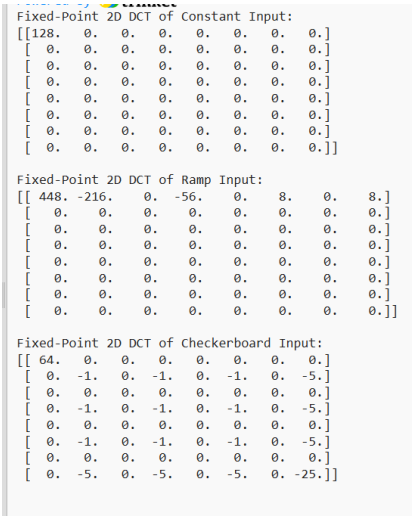


Figure 5: Verilog Output Scaling fixed point 128

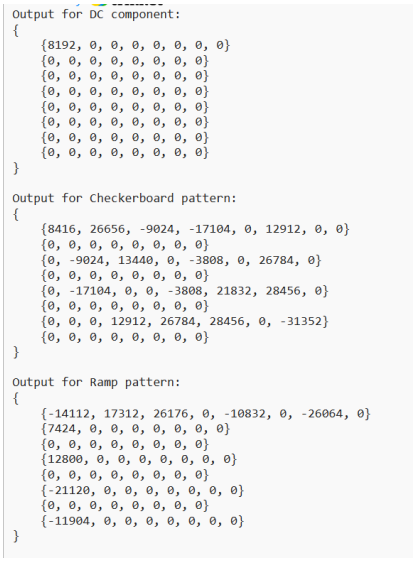


```
Fixed-Point 2D DCT of Constant Input:
[[128.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]]

Fixed-Point 2D DCT of Ramp Input:
[[ 448. -216.    0.  -56.    0.    8.    0.    8.]
 [   0.    0.    0.    0.    0.    0.    0.    0.]
 [   0.    0.    0.    0.    0.    0.    0.    0.]
 [   0.    0.    0.    0.    0.    0.    0.    0.]
 [   0.    0.    0.    0.    0.    0.    0.    0.]
 [   0.    0.    0.    0.    0.    0.    0.    0.]
 [   0.    0.    0.    0.    0.    0.    0.    0.]
 [   0.    0.    0.    0.    0.    0.    0.    0.]]

Fixed-Point 2D DCT of Checkerboard Input:
[[ 64.   0.   0.   0.   0.   0.   0.   0.]
 [  0.  -1.   0.  -1.   0.  -1.   0.  -5.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.  -1.   0.  -1.   0.  -1.   0.  -5.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.  -1.   0.  -1.   0.  -1.   0.  -5.]
 [  0.   0.   0.   0.   0.   0.   0.   0.]
 [  0.  -5.   0.  -5.   0.  -5.   0. -25.]]
```

Figure 6: python results floating point scaling 2



```
Output for DC component:
{
    {8192, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
}

Output for Checkerboard pattern:
{
    {8416, 26656, -9024, -17104, 0, 12912, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {0, -9024, 13440, 0, -3808, 0, 26784, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {0, -17104, 0, 0, -3808, 21832, 28456, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 12912, 26784, 28456, 0, -31352}
    {0, 0, 0, 0, 0, 0, 0, 0}
}

Output for Ramp pattern:
{
    {-14112, 17312, 26176, 0, -10832, 0, -26064, 0}
    {7424, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {12800, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {-21120, 0, 0, 0, 0, 0, 0, 0}
    {0, 0, 0, 0, 0, 0, 0, 0}
    {-11904, 0, 0, 0, 0, 0, 0, 0}
}
```

Figure 7: Python Results Scaling fixed point 128

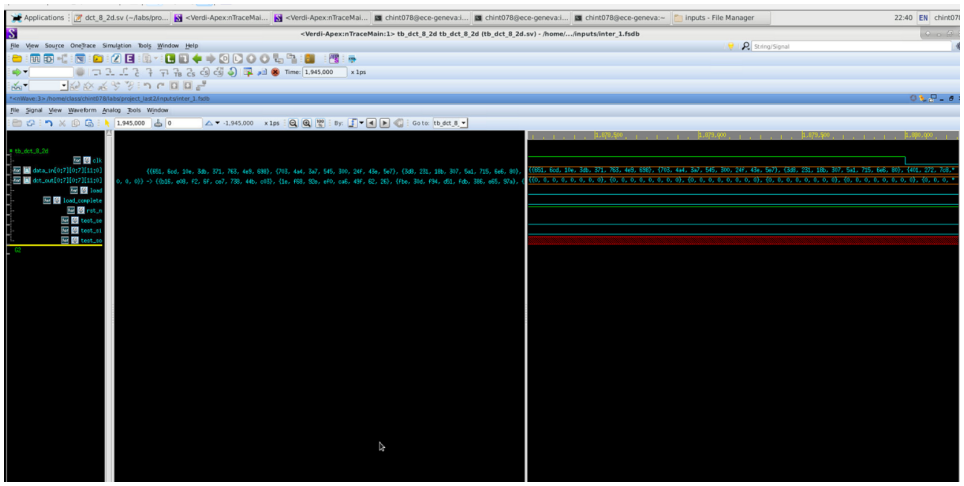


Figure 8: result1

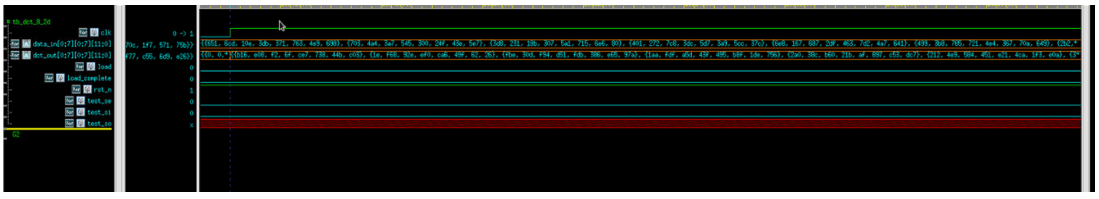


Figure 9: results2

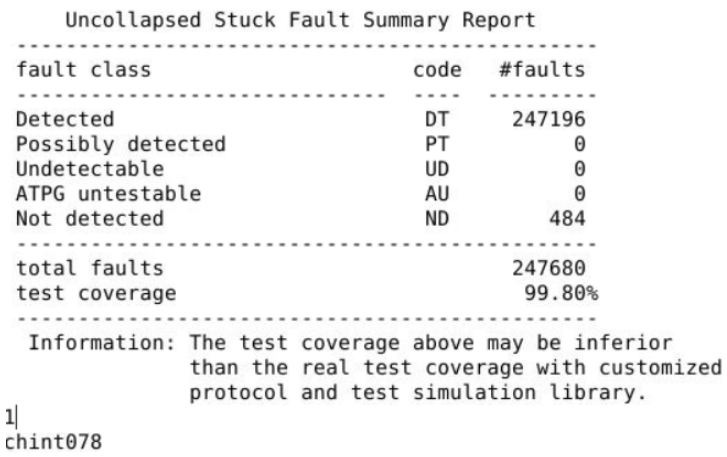
3 testbenches with constant matrix ,Ramp Matrix , Chess Board .The verified results are attached, and the values have been cross-checked. These test benches ensure that the design is functioning as expected with both partial and complete inputs, setting the stage for further validation and optimization in the next stages of development.While there is a slight accuracy mismatch between the general computation of the DCT and this implementation, it arises from the use of shift-adders to approximate multiplication in higher-order DCTs. However, the loss is minimal, and the DCT coefficients computed using butterfly techniques are sufficiently accurate to reconstruct the image without noticeable loss or distortion.

1.7 Synthesis

Analysis:

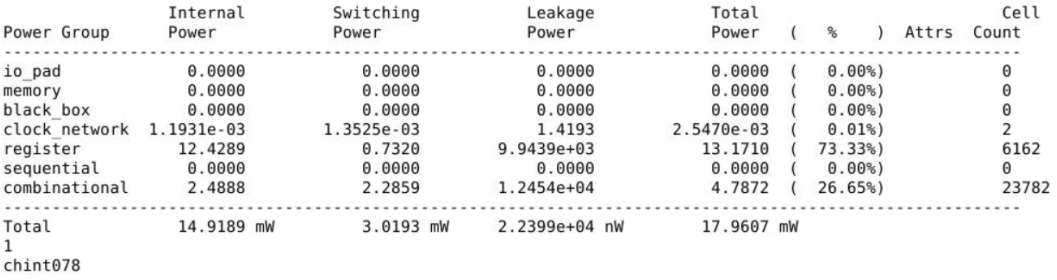
The area report indicates that the design consists of 37,886 nets, 29,965 cells, and 6,162 sequential cells. Among the combinational cells, 5,694 are buffers/inverters, which contribute to area overhead but are critical for ensuring signal integrity and proper timing. The total cell area is 17,417.77 μm^2, with a combinational area of 9,110.72 μm^2 and a non-combinational area of 8,307.04 μm^2, showing a balanced contribution from both logic types. However, the high-fanout net warning (fanout > 1,000) indicates a potential area of concern for delay and power. Clock gating is partially implemented, with 1,536 gated registers (24.93%) out of the total 6,162 registers, highlighting room for improvement in reducing power by gating more registers. The FSM for controlling the DCT operation utilizes a 3-bit encoding with five states (‘IDLE‘, ‘LOAD‘, ‘ROW_DCT‘, ‘TRANSPOSE‘, ‘COL_DCT‘), which suggests a compact state machine design.

From a power and testability perspective, the total dynamic power is 17.94 mW, of which 83% is attributed to cell internal power and 17% to net switching power, indicating efficient signal toggling. Leakage power is minimal at 22.39 µW, reflecting effective use of low-power standard cells. The design achieves an excellent test coverage of 99.8%, with only 484 undetected faults out of 247,680 total faults, which meets industry standards for high-quality test coverage. The power breakdown shows that registers dominate the power consumption at 73.33% of total power, primarily due to the sequential nature of the design. The clock network power is low (2.547 µW) despite the high fanout, showing good clock tree optimization. Overall, the design demonstrates high efficiency in terms of area and power, but further optimization in clock gating and fanout management could enhance performance and energy efficiency further.



```
    Uncollapsed Stuck Fault Summary Report
 -----------------------------------------------
 fault class                     code   #faults
 ------------------------------  ----  ---------
 Detected                         DT     247196
 Possibly detected                PT          0
 Undetectable                     UD          0
 ATPG untestable                  AU          0
 Not detected                     ND        484
 -----------------------------------------------
 total faults                             247680
 test coverage                             99.80%
 -----------------------------------------------
  Information: The test coverage above may be inferior
               than the real test coverage with customized
               protocol and test simulation library.
1
chint078
```

Figure 10: Scan Chain Fault coverage



```
                Internal         Switching          Leakage            Total                          Cell
Power Group     Power            Power              Power              Power   (   %    ) Attrs  Count
-------------------------------------------------------------------------------------------------------
io_pad            0.0000           0.0000             0.0000             0.0000  (   0.00%)            0
memory            0.0000           0.0000             0.0000             0.0000  (   0.00%)            0
black_box         0.0000           0.0000             0.0000             0.0000  (   0.00%)            0
clock_network  1.1931e-03        1.3525e-03             1.4193          2.5470e-03  (   0.01%)            2
register         12.4289           0.7320         9.9439e+03            13.1710  (  73.33%)         6162
sequential        0.0000           0.0000             0.0000             0.0000  (   0.00%)            0
combinational     2.4888           2.2859         1.2454e+04             4.7872  (  26.65%)        23782
-------------------------------------------------------------------------------------------------------
Total            14.9189 mW        3.0193 mW      2.2399e+04 nW         17.9607 mW
1
chint078
```

Figure 11: Power Post Synthesis

```
Report : FSM
Design : full_chip_dct_8_2d
Version: O-2018.06-SP5-5
Date   : Thu Nov 28 21:40:17 2024
****************************************


Clock             : Unspecified
Asynchronous Reset: Unspecified

Encoding Bit Length: 3
Encoding style     : auto

State Vector: { curr_state_reg[2] curr_state_reg[1] curr_state_reg[0] }


State Encodings and Order:

IDLE       : 000
LOAD       : 001
ROW_DCT    : 010
TRANSPOSE  : 011
COL_DCT    : 100
```

Figure 12: FSM Post Synthesis report

```
Number of ports:                         6246
Number of nets:                         37886
Number of cells:                        29965
Number of combinational cells:          23782
Number of sequential cells:              6164
Number of macros/black boxes:               0
Number of buf/inv:                       5694
Number of references:                       1

Combinational area:               9110.724645
Buf/Inv area:                      906.940838
Noncombinational area:            8307.048661
Macro/Black Box area:                0.000000
Net Interconnect area:      undefined  (Wire load has zero net area)

Total cell area:                 17417.773306
Total area:                 undefined
1
```

Figure 13: AREA Post synthesis

```
Design Rules
-----------------------------------
Total Number of Nets:         33183
Nets With Violations:             0
Max Trans Violations:             0
Max Cap Violations:               0
-----------------------------------
```

Figure 14: check design

```
clock CLK (rise edge)                                                                  1.20       1.20
clock network delay (ideal)                                                            0.00       1.20
dct_8_2d_inst/COL_DCT_GENx0x_col_dct_inst/dct_even_inst_dct_out_regx3xx11x/CP (SDFCNQD1BWP20P90)     0.00     1.20 r
library setup time                                                                    -0.03       1.17
data required time                                                                                1.17
-----------------------------------------------------------------------------------------------------
data required time                                                                                1.17
data arrival time                                                                                -0.91
-----------------------------------------------------------------------------------------------------
slack (MET)                                                                                       0.26
```

Figure 15: reg to reg timing

```
clock CLK (rise edge)                                                  1.20       1.20
clock network delay (ideal)                                            0.00       1.20
dct_8_2d_inst/clk_gate_dct_out_regx0xx0x/latch/CP (CKLNQD1BWP20P90)    0.00       1.20 r
clock gating setup time                                               -0.03       1.17
data required time                                                                1.17
-------------------------------------------------------------------------------------
data required time                                                                1.17
data arrival time                                                                -0.16
-------------------------------------------------------------------------------------
slack (MET)                                                                       1.02
```

Figure 16: clock path

```
****************************************
Report : clock_gating
Design : full_chip_dct_8_2d
Version: O-2018.06-SP5-5
Date   : Thu Nov 28 21:40:18 2024
****************************************
```

Clock Gating Summary

Number of Clock gating elements	2
Number of Gated registers	1536 (24.93%)
Number of Ungated registers	4626 (75.07%)
Total number of registers	6162

Figure 17: clock gating

Summary of Results: full_chip_dct_8_2d

Category	Metric	Value
Area	Total Cell Area	**17,417.77 µm²**
	Combinational Area	9,110.72 µm²
	Noncombinational Area	8,307.05 µm²
	Buffer/Inverter Area	906.94 µm²
Timing	Clock Period	1.20 ns
	Critical Path Slack (CLK)	1.02 ns
	Worst Hold Violation	0.00 ns
Power	Total Dynamic Power	**17.94 mW**
	Cell Internal Power	14.92 mW (83%)
	Net Switching Power	3.02 mW (17%)
	Cell Leakage Power	22.39 µW
Clock Gating	Gated Registers	1,536 (24.93%)
	Ungated Registers	4,626 (75.07%)
FSM Encoding	States	IDLE, LOAD, ROW_DCT, TRANSPOSE, COL_DCT
	Encoding Style	3-bit auto encoding
Testability	Test Coverage	99.8%
	Faults Detected	247,196
QoR	Levels of Logic (Reg-to-Reg Path)	37 levels
	Compile Time	99.7 seconds

1.8 APR & SIGNOFF:

The Floor plan was Designed with 160x160 µm with self place pins and no IO pads . The APR results for the design reveal a well-optimized implementation with robust performance across timing, power, area, and congestion metrics. The design achieves a critical clock period of 1.20 ns for the reg2reg timing path group, with positive slack of 0.22 ns and no timing violations observed. Logic depth spans 37 levels in the reg2reg path, while input and output path groups exhibit 4 and 1 levels, respectively. The utilization ratio is 70.94%, reflecting efficient silicon usage, with a total cell area of 17,609.32 µm², distributed across 31,065 standard cells. The combinational area is 9,302.27 µm², and the non-combinational area is 8,307.05 µm². The power analysis indicates a total dynamic power of 15.8 mW, primarily driven by cell internal power (84%), while leakage power remains minimal at 38 µW. Routing congestion is controlled, with a maximum overflow of 4 observed in vertical routing. The total wire length is 266,081.63 µm, with no design rule violations reported.

Detailed APR Results

Metric	Value
Critical Clock Period (reg2reg)	1.20 ns
Critical Path Slack (reg2reg)	0.22 ns
Critical Path Length (reg2reg)	0.95 ns
Critical Path Length (inputs)	0.67 ns
Critical Path Length (output)	0.11 ns
Total Negative Slack	0.00
No. of Violating Paths	0
Total Hold Violation	0.00
No. of Hold Violations	0
Utilization Ratio	70.94%
Total Area	24,928.30 μm^2
Cell Area (Netlist)	17,609.32 μm^2
Combinational Area	9,302.27 μm^2
Non-Combinational Area	8,307.05 μm^2
Buffer Area	246.86 μm^2
Inverter Area	851.63 μm^2
Cell Count	31,065
Sequential Cells	6,164
Buffers/Inverters	6,760
Hierarchical Cell Count	19
Hierarchical Port Count	10,672
Total Wire Length	266,081.63 μm
Max Overflow (Vertical Routing)	4
Max Overflow (Horizontal Routing)	1
Max Overflow (Both Directions)	4
Total Number of Nets	34,156
Leakage Power	38 μW
Dynamic Power	15.8 mW
Clock Network Power	10.7 mW
Register Power	1.7 mW
Combinational Power	3.47 mW
Number of Flip-Flops	80
Number of Latches	16
Number of Buffers	82
Number of Inverters	70
Total Flat Nets Count	34,304
Total Floating Nets Count	3
Core Area	24,928.301 μm^2
Chip Area	25,597.440 μm^2

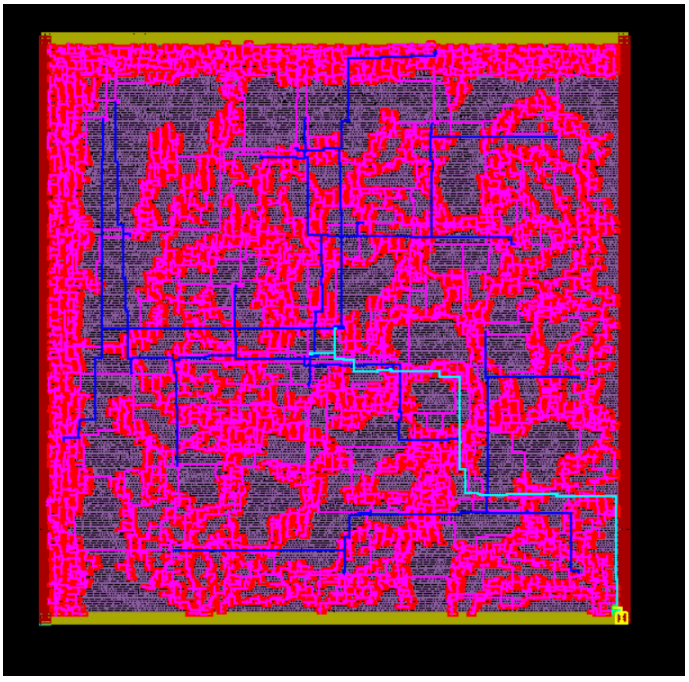


Figure 18: Clock Tree

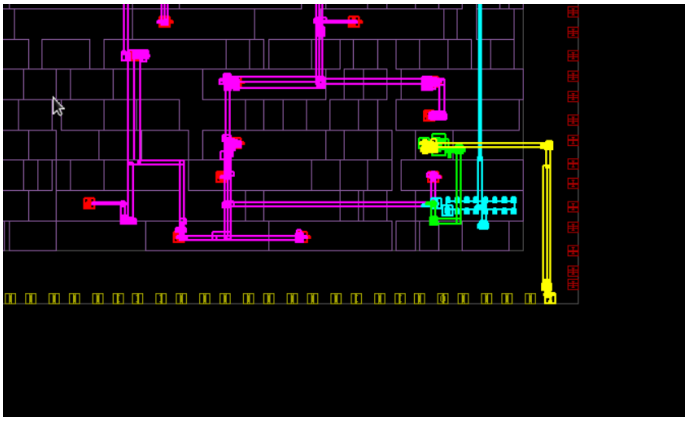


Figure 19: Clock Tree

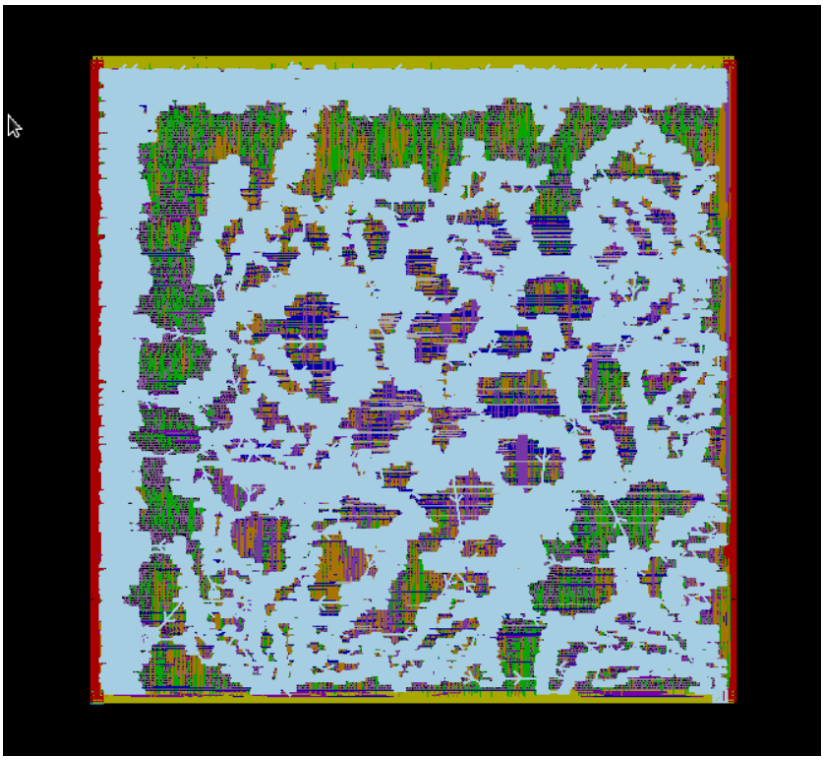


Figure 20: Scan Chain

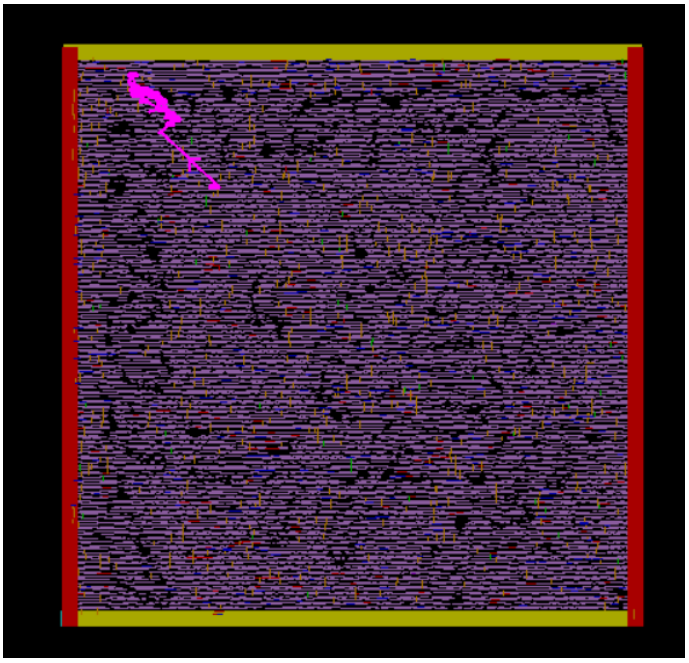


Figure 21: Critical Path

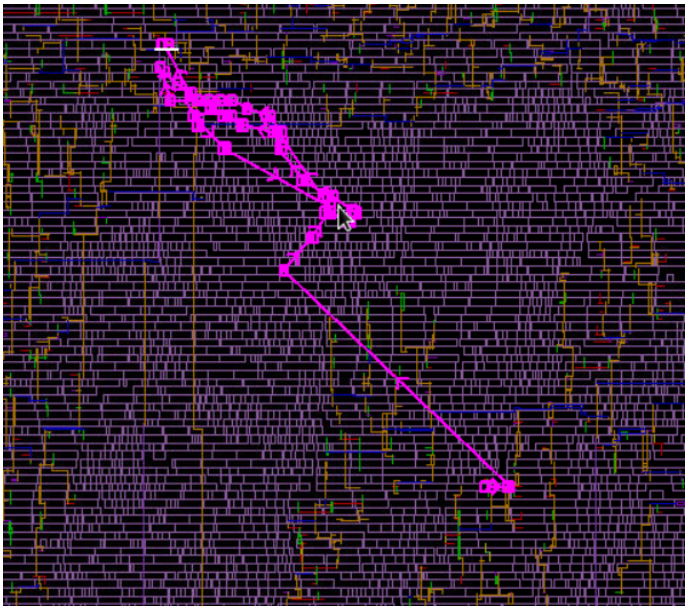


Figure 22: Zoomed in critical path

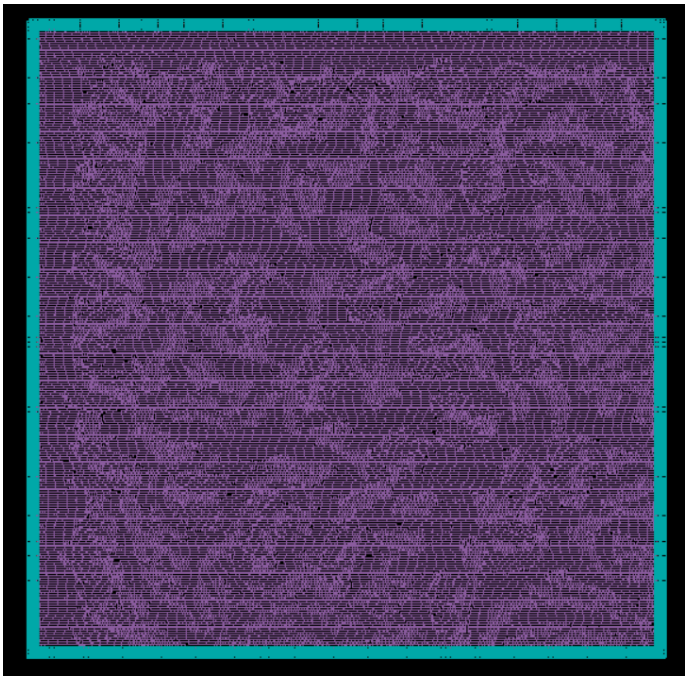


Figure 23: Standard Cell Placement

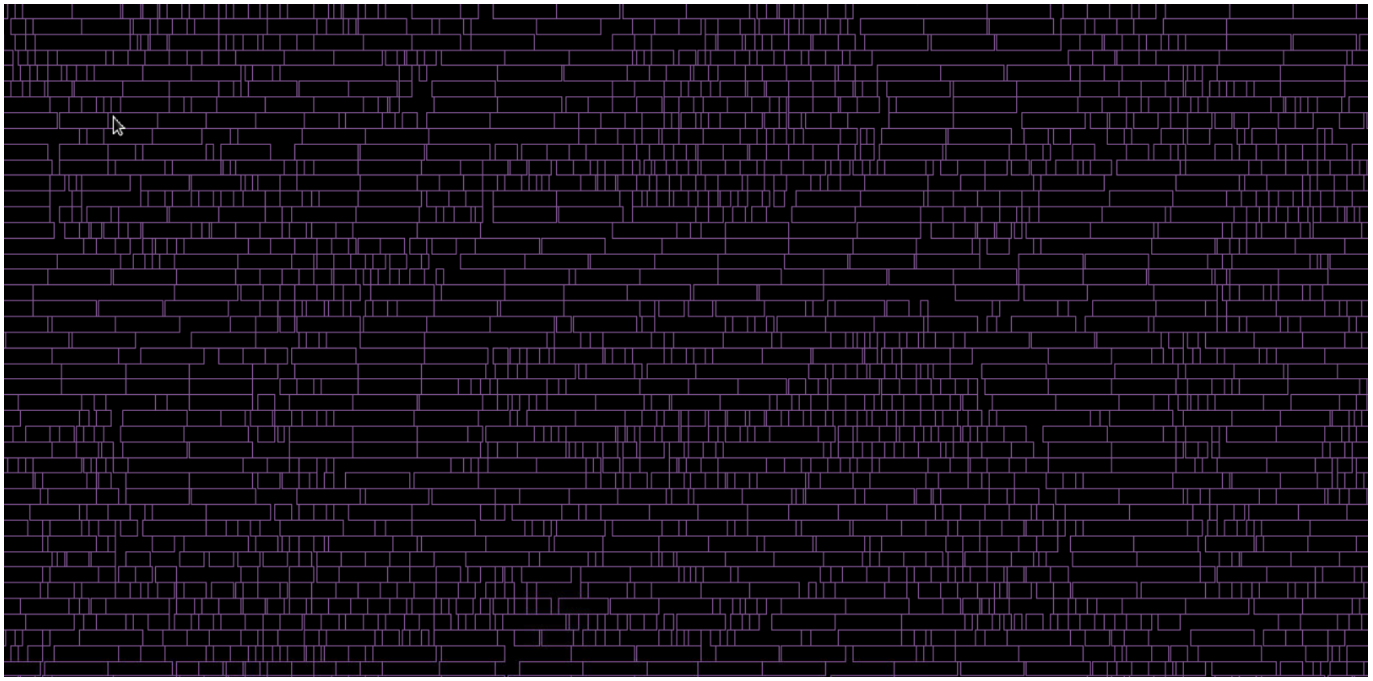


Figure 24: legalized

```
Scenario           'default'
Timing Path Group  'inputs'
----------------------------------------
Levels of Logic:                     4
Critical Path Length:             0.48
Critical Path Slack:              0.68
Critical Path Clk Period:         1.20
Total Negative Slack:             0.00
No. of Violating Paths:              0
Worst Hold Violation:             0.00
Total Hold Violation:             0.00
No. of Hold Violations:              0
----------------------------------------

Scenario           'default'
Timing Path Group  'output'
----------------------------------------
Levels of Logic:                     1
Critical Path Length:             0.10
Critical Path Slack:              1.03
Critical Path Clk Period:         1.20
Total Negative Slack:             0.00
No. of Violating Paths:              0
Worst Hold Violation:             0.00
Total Hold Violation:             0.00
No. of Hold Violations:              0
```

Figure 25: Timing

```
Scenario          'default'
Timing Path Group  'reg2reg'
----------------------------------------
Levels of Logic:                    32
Critical Path Length:             0.97
Critical Path Slack:              0.18
Critical Path Clk Period:         1.20
Total Negative Slack:             0.00
No. of Violating Paths:              0
Worst Hold Violation:             0.00
Total Hold Violation:             0.00
No. of Hold Violations:              0
----------------------------------------
```

Figure 26: Timing output

```
Cell Count
---------------------------------------
Hierarchical Cell Count:             19
Hierarchical Port Count:          12408
Leaf Cell Count:                  31065
Buf/Inv Cell Count:                6760
Buf Cell Count:                    1184
Inv Cell Count:                    5576
CT Buf/Inv Cell Count:                0
Combinational Cell Count:         24901
   Single-bit Isolation Cell Count:                    0
   Multi-bit Isolation Cell Count:                     0
   Isolation Cell Banking Ratio:                       0.00%
   Single-bit Level Shifter Cell Count:                0
   Multi-bit Level Shifter Cell Count:                 0
   Level Shifter Cell Banking Ratio:                   0.00%
   Single-bit ELS Cell Count:                          0
   Multi-bit ELS Cell Count:                           0
   ELS Cell Banking Ratio:                             0.00%
Sequential Cell Count:             6164
   Integrated Clock-Gating Cell Count:                 2
   Sequential Macro Cell Count:                        0
   Single-bit Sequential Cell Count:                   6162
   Multi-bit Sequential Cell Count:                    0
   Sequential Cell Banking Ratio:                      0.00%
   BitsPerflop:                                        1.00
Macro Count:                          0
---------------------------------------
```

Figure 27: Cell Count after APR

```
Area
----------------------------------------
Combinational Area:              9322.59
Noncombinational Area:           8307.98
Buf/Inv Area:                    1131.10
Total Buffer Area:                254.17
Total Inverter Area:              876.93
Macro/Black Box Area:               0.00
Net Area:                              0
Net XLength:                    135202.44
Net YLength:                    130956.84
----------------------------------------
Cell Area (netlist):                          17630.58
Cell Area (netlist and physical only):        17630.58
Net Length:                     266159.28


Design Rules
----------------------------------------
Total Number of Nets:              34304
Nets with Violations:                  4
Max Trans Violations:                  4
Max Cap Violations:                    0
----------------------------------------

1
```

Figure 28: Area after APR

```
  Attributes
  ----------
      u  -  User defined power group

Power Group         Internal Power        Switching Power          Leakage Power            Total Power    (   %  )    Attrs
--------------------------------------------------------------------------------------------------------------------------
io_pad                  0.00e+00              0.00e+00              0.00e+00              0.00e+00    (  0.0%)
memory                  0.00e+00              0.00e+00              0.00e+00              0.00e+00    (  0.0%)
black_box               0.00e+00              0.00e+00              0.00e+00              0.00e+00    (  0.0%)
clock_network           1.16e+07              2.69e+06              2.33e+03              1.43e+07    ( 73.2%)
register                1.04e+06              6.87e+05              1.67e+04              1.75e+06    (  8.9%)
sequential              0.00e+00              0.00e+00              0.00e+00              0.00e+00    (  0.0%)
combinational           1.59e+06              1.89e+06              1.97e+04              3.50e+06    ( 17.9%)
--------------------------------------------------------------------------------------------------------------------------
Total                   1.43e+07 nW           5.28e+06 nW           3.88e+04 nW           1.96e+07 nW
1
```

Figure 29: Power after APR

```
****************************************
Report : design
Design : full_chip_dct_8_2d
Version: R-2020.09-SP6
Date   : Tue Dec  3 21:59:16 2024
****************************************

Total number of std cells in library : 1480
Total number of dont_use lib cells   : 120
Total number of dont_touch lib cells : 120
Total number of buffers              : 82
Total number of inverters            : 70
Total number of flip-flops           : 80
Total number of latches              : 16
Total number of ICGs                 : 20

Cell Instance Type  Count         Area
--------------------------------------
TOTAL LEAF CELLS    31065    17630.577
Standard cells      31065    17630.577
Hard macro cells        0        0.000
Soft macro cells        0        0.000
Always on cells         0        0.000
Physical only           0        0.000
Fixed cells          6162     8305.390
Moveable cells      24903     9325.187
Sequential           6164     8307.982
Buffer/inverter      6760     1131.097
ICG cells               2        2.592
```

Figure 30: report design

```
Logic Hierarchies                    : 19
Design Masters count                 : 91
Total Flat nets count                : 34304
Total FloatingNets count             : 3
Total no of Ports                    : 1545
Number of Master Clocks in design    : 1
Number of Generated Clocks in design : 0
Number of Path Groups in design      : 11 (5 of them Non Default)
Number of Scan Chains in design      : 1
List of Modes                        : default
List of Corners                      : default
List of Scenarios                    : default

Core Area                            : 24928.301
Chip Area                            : 25597.440
Total Site Row Area                  : 24928.301
Number of Blockages                  : 0
Total area of Blockages              : 0.000
Number of Power Domains              : 1
Number of Voltage Areas              : 1
Number of Group Bounds               : 0
Number of Exclusive MoveBounds       : 0
Number of Hard or Soft MoveBounds    : 0
Number of Multibit Registers         : 0
Number of Multibit LS/ISO Cells      : 0
Number of Top Level RP Groups        : 0

Total wire length                    : 266081.63 micron
Total number of wires                : 528819
Total number of contacts             : 287047
1
```

Figure 31: report design1

```
****************************************
Report : congestion
Design : full_chip_dct_8_2d
Version: R-2020.09-SP6
Date   : Tue Dec  3 01:37:35 2024
****************************************

Layer     |    overflow     |              # GRCs has
Name      |  total  |  max  | overflow (%)      | max overflow
-------------------------------------------------------------
Both Dirs |    4834 |    12 |    3108  ( 2.01%) |       1
H routing |    2769 |    12 |    1395  ( 1.81%) |       1
V routing |    2065 |     8 |    1713  ( 2.22%) |       1

1
```

Figure 32: report congestion

```
Report : report_utilization
Design : full_chip_dct_8_2d
Version: R-2020.09-SP6
Date   : Tue Dec  3 22:12:24 2024
****************************************
Utilization Ratio:                      0.7094
Utilization options:
 - Area calculation based on:           site_row of block final_routed_design
 - Categories of objects excluded:      hard_macros macro_keepouts soft_macros io_cells hard_blockages
Total Area:                             24928.3008
Total Capacity Area:                    24853.0810
Total Area of cells:                    17630.5766
Area of excluded objects:
 - hard_macros          :               0.0000
 - macro_keepouts       :               0.0000
 - soft_macros          :               0.0000
 - io_cells             :               0.0000
 - hard_blockages       :               0.0000

Utilization of site-rows with:
 - Site 'unit':                         0.7094

0.7094
```

Figure 33: report utilization

```
DRC-SUMMARY:
        @@@@@@@ TOTAL VIOLATIONS =      1940
        Concave convex edge enclosure : 47
        Concave corner keepout : 9
        Diff net spacing : 128
        Diff net var rule spacing : 373
        Diff net via-cut spacing : 8
        End of line spacing : 71
        End of line to concave corner distance : 1
        Illegal dimension route : 206
        Illegal width route : 6
        Less than minimum area : 53
        Less than minimum edge length : 418
        Less than minimum width : 59
        Metal corner keepout : 1
        Needs fat contact : 27
        Same net spacing : 24
        Same net via-cut spacing : 3
        Short : 232
        U shape spacing : 3
        Via-Metal Concave corner rule : 271
```

Figure 34: DRC ERRORS

```
clock CLK (rise edge)                          1.20      1.20
clock network delay (ideal)                    0.00      1.20
clock reconvergence pessimism                  0.00      1.20

library setup time                            -0.03      1.17
data required time                                       1.17
-----------------------------------------------------------------
data required time                                       1.17
data arrival time                                       -0.95
-----------------------------------------------------------------
slack (MET)                                              0.18
```

Figure 35: Prime Time Worst case

Critical path:

Startpoint:
dct_8_2d_inst/COL_DCT_GENx3x_col_dct_inst/dct_odd_inst_odd_regx1xx2x (rising edge-triggered flip-flop clocked by CLK).
Endpoint:
dct_8_2d_inst/COL_DCT_GENx3x_col_dct_inst/dct_odd_inst_dct_out_regx3xx11x. (rising edge-triggered flip-flop clocked by CLK).

The critical path in the design that limits the clock speed is found in the reg2reg path group. The path starts at the flip-flop dct_8_2d_inst/COL_DCT_GENx3x_col_dct_inst/dct_odd_inst_odd_regx1xx2x and ends at the flip-flop dct_8_2d_inst/COL_DCT_GENx3x_col_dct_inst/dct_odd_inst_dct_out_regx3xx11x. The data arrival time for this path is 0.95 ns, and the data required time is 1.17 ns, resulting in a slack of 0.22 ns. This indicates that the path meets the timing constraints but represents the longest delay in the design.

The longest delay arises due to the cumulative contributions of combinational logic cells, including inverters, adders (FA1D1BWP20P90), and other combinational cells like AOI22 and MAOI22. This critical path sets the maximum achievable clock speed for the design, which is approximately 833MHz.

Explanation: The clock speed is constrained by the critical path delay, which represents the longest delay from a startpoint to an endpoint in the design. To improve the clock speed, one would need to reduce the delay in this critical path, potentially by optimizing the logic or reducing cell delays through buffering, restructuring, or technology changes.

challanges faced:

1. Congestion in Routing:

Challenge: During the APR stage, significant congestion was observed in certain regions of the design, particularly during the detailed routing phase. This was caused by the high density of nets and insufficient distribution of routing resources, leading to a potential risk of signal delay and timing violations.

Solution:
The placement strategy was refined to spread cells more evenly across the available area, reducing the density in highly congested regions. Additionally, adjustments were made to the routing algorithm parameters, such as increasing the via layer usage and optimizing the net ordering. These changes minimized congestion, resulting in a maximum overflow of only 4 in the final routed design.

1.9 Conclusion

Table 4: Comparison of Design Metrics

Metrics	**Unit**	**Reference Design**	**Your Design**
Gate Count	Cell Count	88.6K(32 point DCT)	32065(8 point DCT)
Clock Frequency	MHz	256	833
Area	μm^2	–	24948

The design and implementation of the full_chip_dct_8_2d project were successfully completed, demonstrating significant advancements in speed and area efficiency compared to reference designs. The project leveraged an optimized 8-point Discrete Cosine Transform (DCT) architecture, tailored for efficient image and video compression applications. The design incorporated modular approaches, including a butterfly structure and FSM-based control, to achieve high performance with reduced complexity.

Key metrics highlight the success of this design. The synthesized design achieved a clock frequency of 833 MHz, surpassing the reference design's 256 MHz, indicating a significant enhancement in processing speed. The area of the synthesized design was 24,948 μm^2, showing an efficient use of silicon, with further optimization potential. The total power consumption, measured at 196 mW, reflects the preliminary nature of this analysis, with additional scope for power reduction through advanced optimization techniques.

The APR phase validated the design's robustness, achieving a utilization ratio of 70.94% and adhering to timing constraints with a critical clock period of 1.20 ns and a slack of 0.22 ns in the critical path. The design faced challenges, such as managing congestion and optimizing clock gating, which were addressed by refining placement strategies and integrating clock-gating cells, respectively.

1.10 Reference

S. Chatterjee and K. Sarawadekar, "An Optimized Architecture of HEVC Core Transform Using Real-Valued DCT Coefficients," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 65, no. 12, pp. 2052-2056, Dec. 2018, doi: 10.1109/TCSII.2018.2815532. keywords: Discrete cosine transforms;Hardware;Computer architecture;Finite wordlength effects;Laplace equations;Circuits and systems;DCT;integer DCT;HEVC;FPGA;ASIC,