

# Lab3 Intro

## Improving Area and Resources



# Objectives

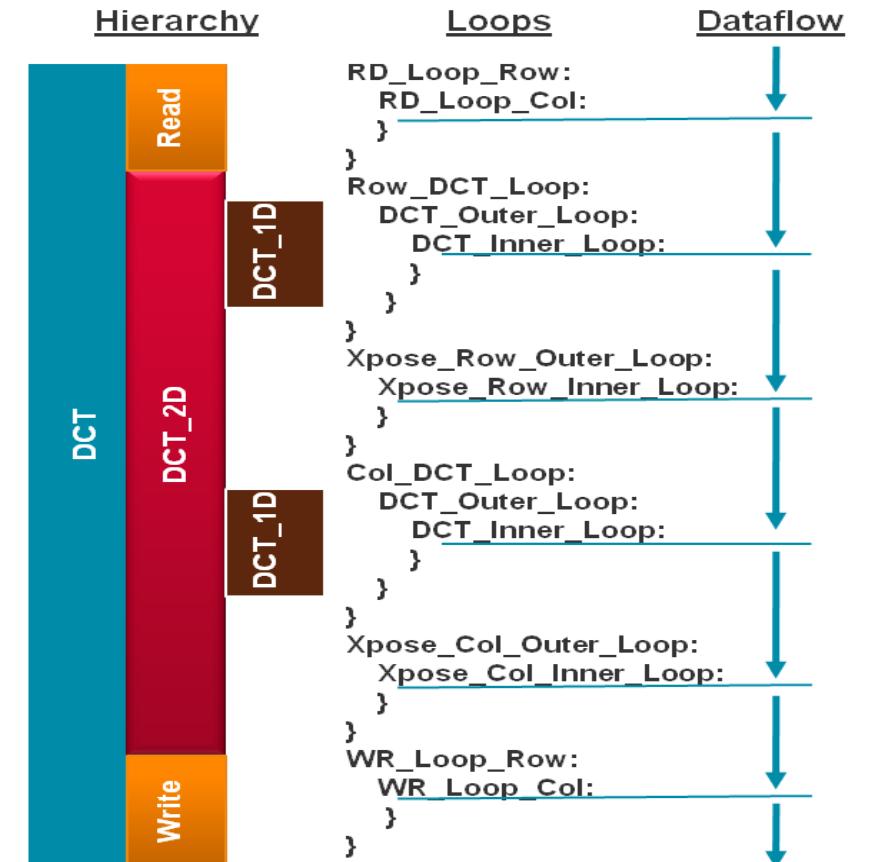
- > **After completing this lab, you will be able to:**
  - >> Manage BRAM and DSP48 resource utilization
  - >> Improve memory bandwidth
  - >> Balance resource utilization and performance
  - >> Distinguish between DATAFLOW directive and Configuration Command functionality

# The Design

## > The design under consideration is a Discrete Cosine Transformation (DCT) function on a 8x8 block of data

- >> The top-level function dct implements 2D DCT algorithm by first processing each row of the input array via a 1D DCT then processing the columns of the resulting array through the same 1D DCT. It calls read\_data, dct\_2d, and write\_data functions.
- >> The read\_data function consists of two loops – RD\_Loop\_Row and RD\_Loop\_Col.
- >> The write\_data function is defined consists of two loops to perform writing the result.

```
78 void dct(short input[N], short output[N])
79 {
80     short buf_2d_in[DCT_SIZE][DCT_SIZE];
81     short buf_2d_out[DCT_SIZE][DCT_SIZE];
82
83     // Read input data. Fill the internal buffer.
84     read_data(input, buf_2d_in);
85
86     dct_2d(buf_2d_in, buf_2d_out);
87
88     // Write out the results.
89     write_data(buf_2d_out, output);
90 }
91 }
```



# Procedure

- > **Compile the design in command mode and perform C-verification**
- > **Open the project in Vivado HLS GUI, synthesize, and review results**
- > **Simulate the design**
- > **Improve performance using pipeline**
- > **Optimize fine-grained parallelism**
- > **Improve memory bandwidth**
- > **Apply DATAFLOW directive to improve performance**
- > **Apply RESHAPE directive and analyze**

# Summary

- > In this lab, you learned various techniques to improve the performance and balance resource utilization. PIPELINE directive when applied to outer loop will automatically cause the inner loop to unroll. When a loop is unrolled, resources utilization increases as operations are done concurrently. Partitioning memory may improve performance but will increase BRAM utilization. When INLINE directive is applied to a function, the lower level hierarchy is automatically dissolved. When DATAFLOW directive is applied, the default memory buffers (of ping-pong type) are automatically inserted between the top-level functions and loops. The RESHAPE directive will allow multiple accesses to BRAM, however, care should be taken if a single element requires modification as it will result in read-modify-write operation for the entire word. The Analysis perspective and console logs can provide insight on what is going on.

**Adaptable.**  
**Intelligent.**

