

# DFX Lab

## Course Workbook

# Table of Contents

About this Workbook

[Slide 3](#)

Pre-Lab: Workshop Pre-requisites

[Slide 4](#)

Yocto: TMR MicroBlaze

[Slide 7](#)

# About this Workbook

The contents of this workbook are created by Aduvo Engineering & Training, Ltd.

If you have any questions about the contents, or need assistance, please contact Adam Taylor at [adam@aduvoengineering.com](mailto:adam@aduvoengineering.com).

# Pre-Lab

## Workshop Pre-requisites

# Downloads and Installations

**Step 1** – Download and install the following at least one day prior to the workshop. This may take a significant amount of time and drive space.

Vivado 2021.1	<a href="#">Download</a>

# Lab

## DFX – RTL

# Lab: DFX

**Step 1** – Download the source files from the repo creating a new project for the target board. Add to the files the constraints and the source files in the folder top.

The screenshot shows the Vivado 2021.1 AR76666 Project Manager interface. The main window displays the 'Sources' tree, which includes Design Sources (4) and Constraints (2). The 'Constraint Set Properties' panel for 'constrs\_1' is visible, showing the default directory, file count, format, and target constraint file. The 'Project Summary' panel on the right provides an overview of the project settings, including the project name, location, product family, and board part.

**Project Summary**

Overview | Dashboard

Settings Edit

Project name: dfx  
 Project location: C:/hdl\_projects/dfx  
 Product family: Kintex-UltraScale  
 Project part: Kintex-UltraScale KCU105 Evaluation Platform (xcxu040-#v1156-2-e)  
 Top module name: top  
 Target language: VHDL  
 Simulator language: Mxed

**Board Part**

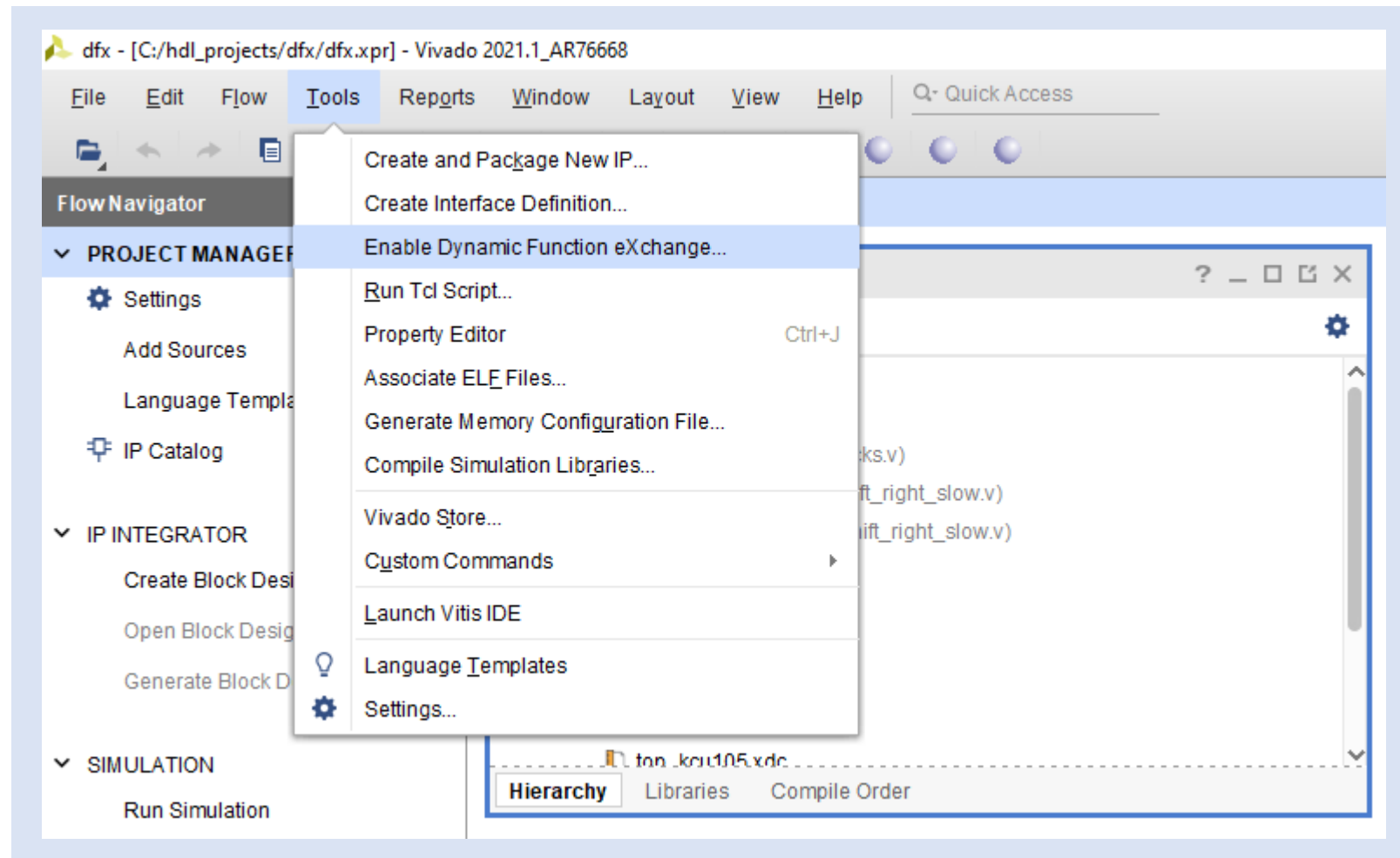
Display name: Kintex-UltraScale KCU105 Evaluation Platform  
 Board part name: xilinx.com:kcu105:part0:1.7  
 Board revision: 1.0  
 Connectors: No connections  
 Repository path: C:/xilinx/vivado/2021.1/data/xhub/boards  
 URL: www.xilinx.com/kcu105  
 Board overview: Kintex-UltraScale KCU105 Evaluation Platform

**Design Runs**

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy
synth_1	constrs_1	Not started															Vivado Synthesis Defaults (Vivado Synthesis 2021)	Vivado Synthesis Default Reports (Vivado Synthesis 2021)
impl_1	constrs_1	Not started															Vivado Implementation Defaults (Vivado Implementation 2021)	Vivado Implementation Default Reports (Vivado Implementation 2021)

# Lab: DFX

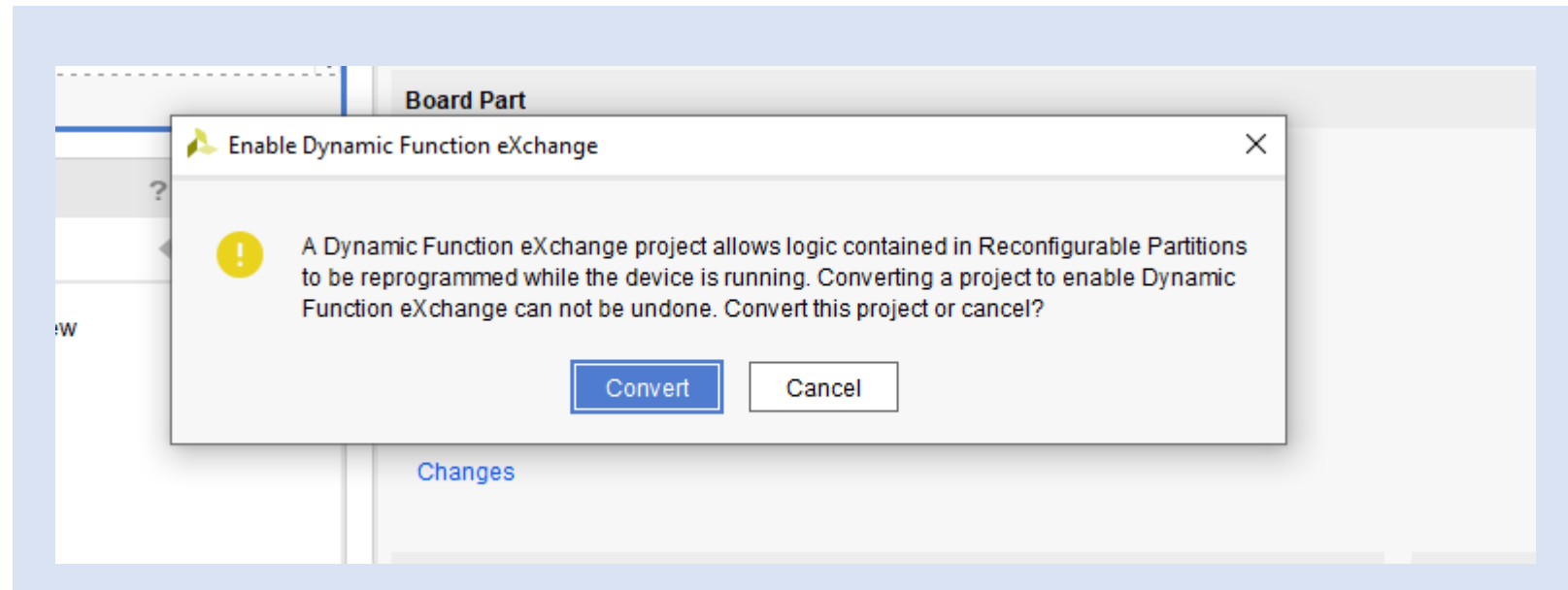
Step 2 – From the Tools menu Enable Dynamic Functional eXchange





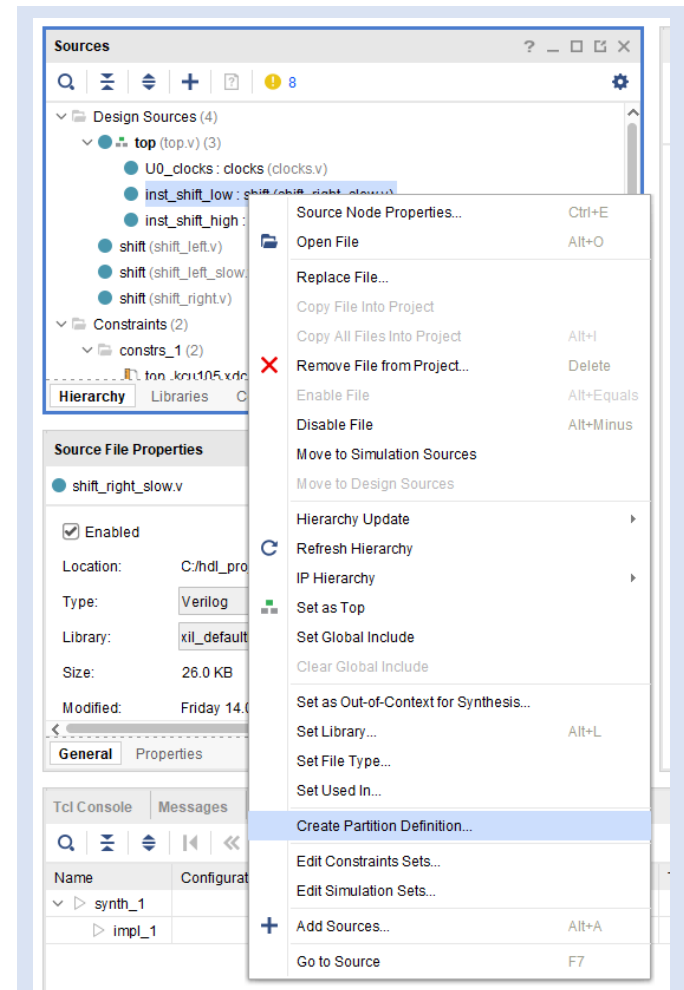
# Lab: DFX

Step 3 – When prompted select convert



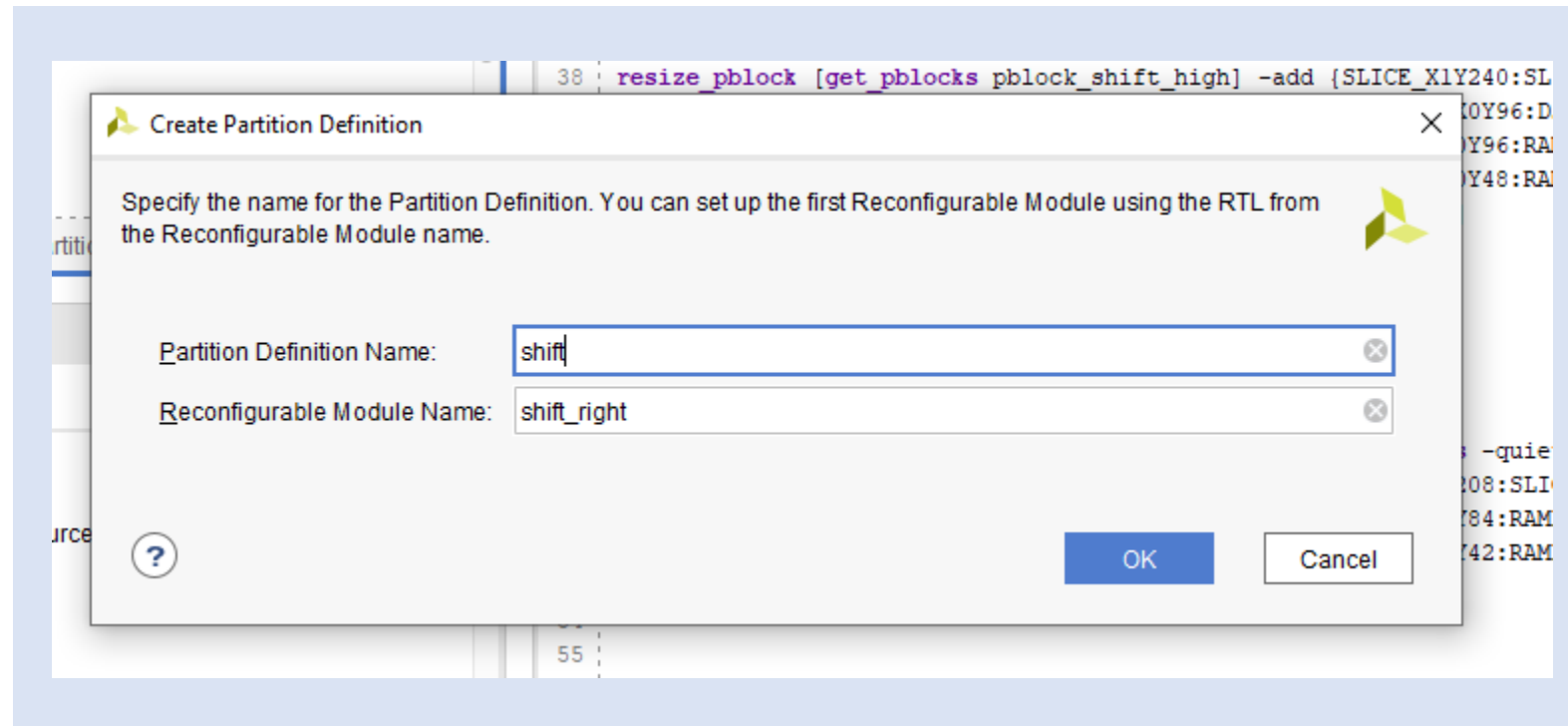
# Lab: DFX

Step 4 – Select the first of the shift instances and select create partition definition



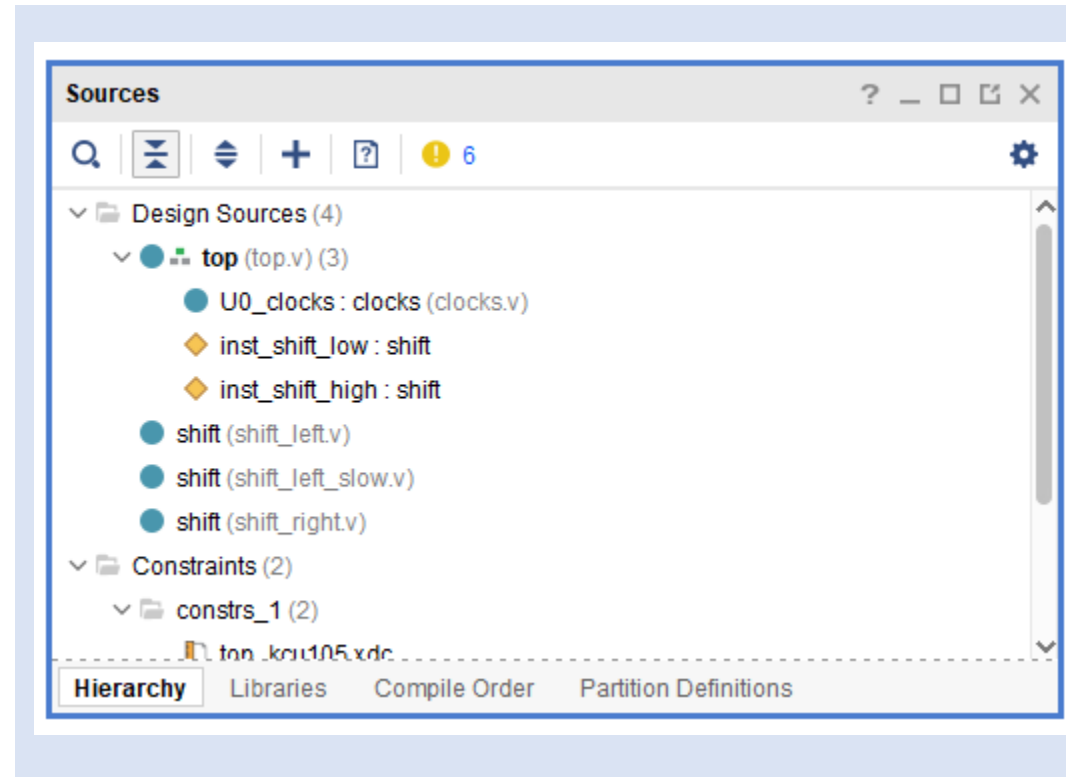
# Lab: DFX

Step 5 – Enter a partition definition name and click ok



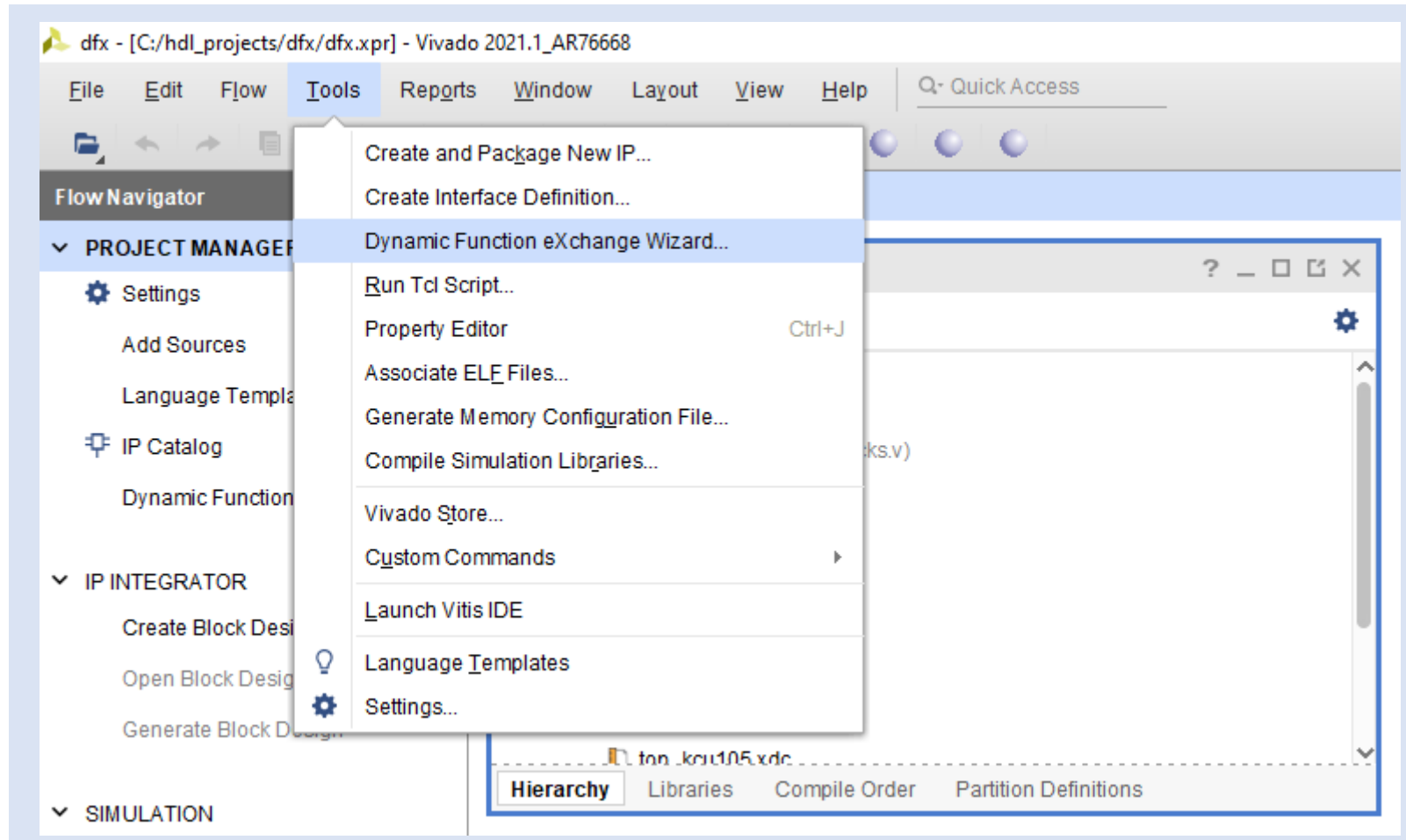
# Lab: DFX

**Step 6** – Note in the sources view both have changed, this is because they are the same module instantiated twice.



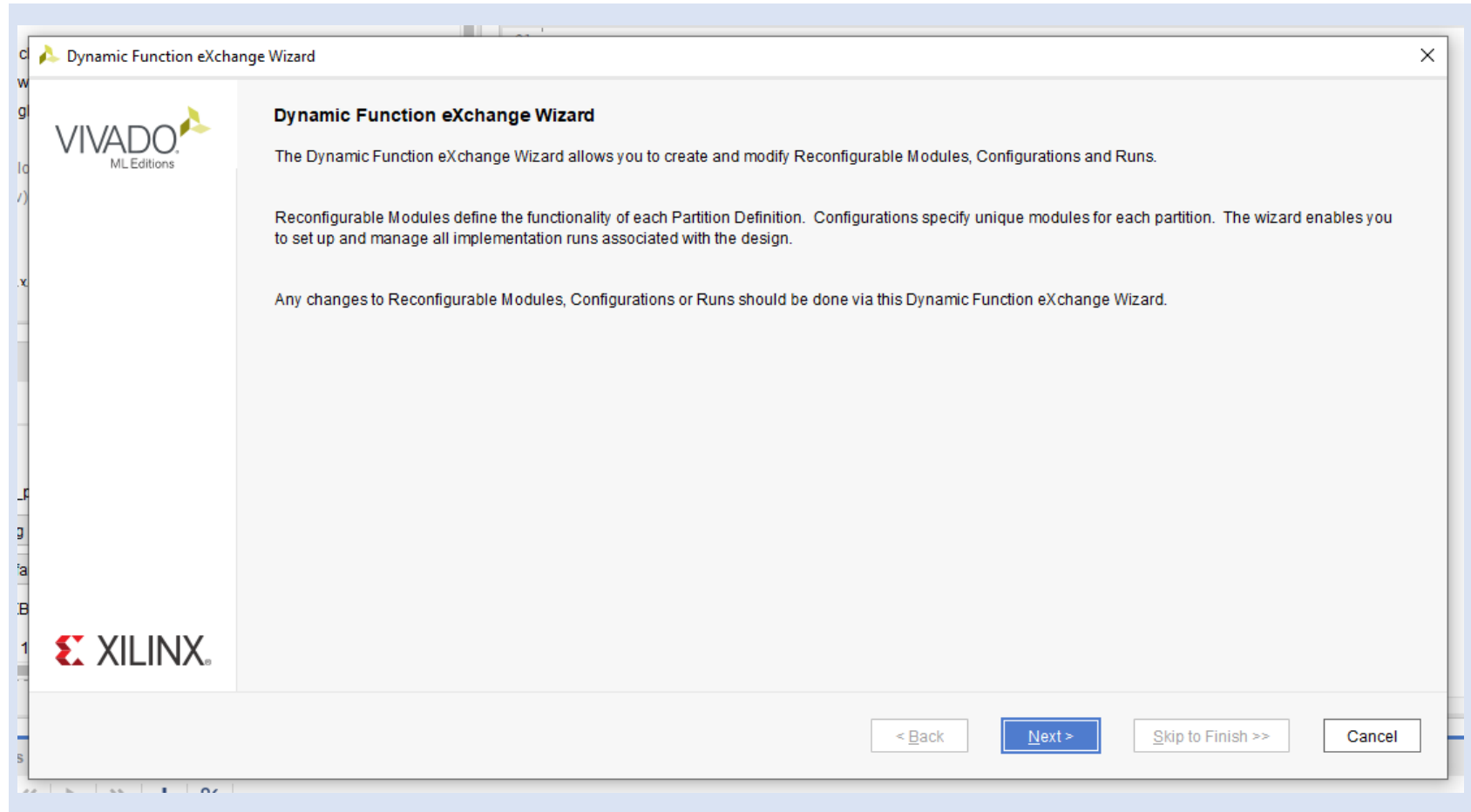
# Lab: DFX

Step 7 – From the tool's menu select Dynamic Functional eXchange Wizard



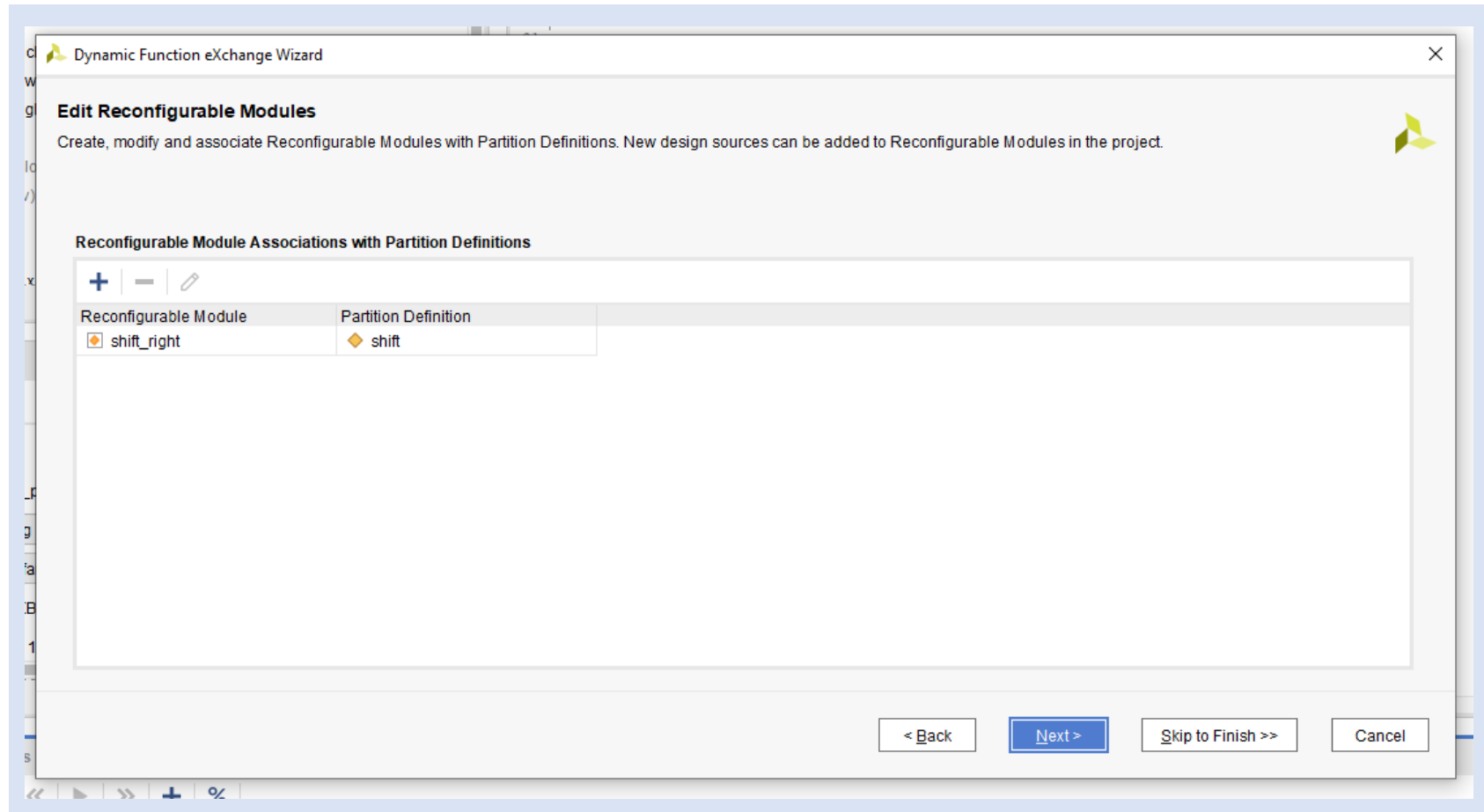
# Lab: DFX

## Step 8 – Click on next



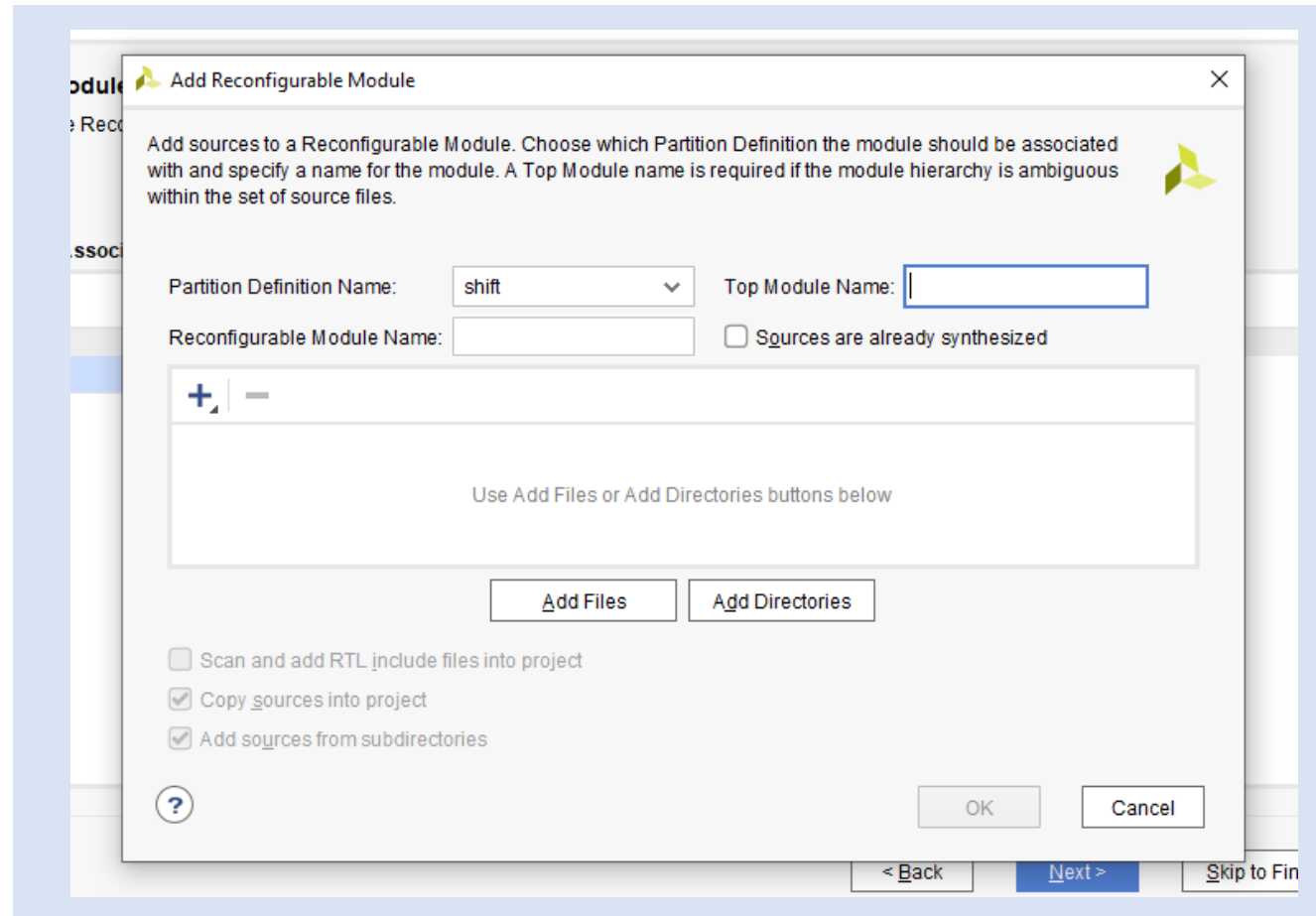
# Lab: DFX

**Step 9** – We can see the primary design – which will do a shift right pattern. This will be placed into the top-level bit file. We can create additional partitions by clicking on the blue +



# Lab: DFX

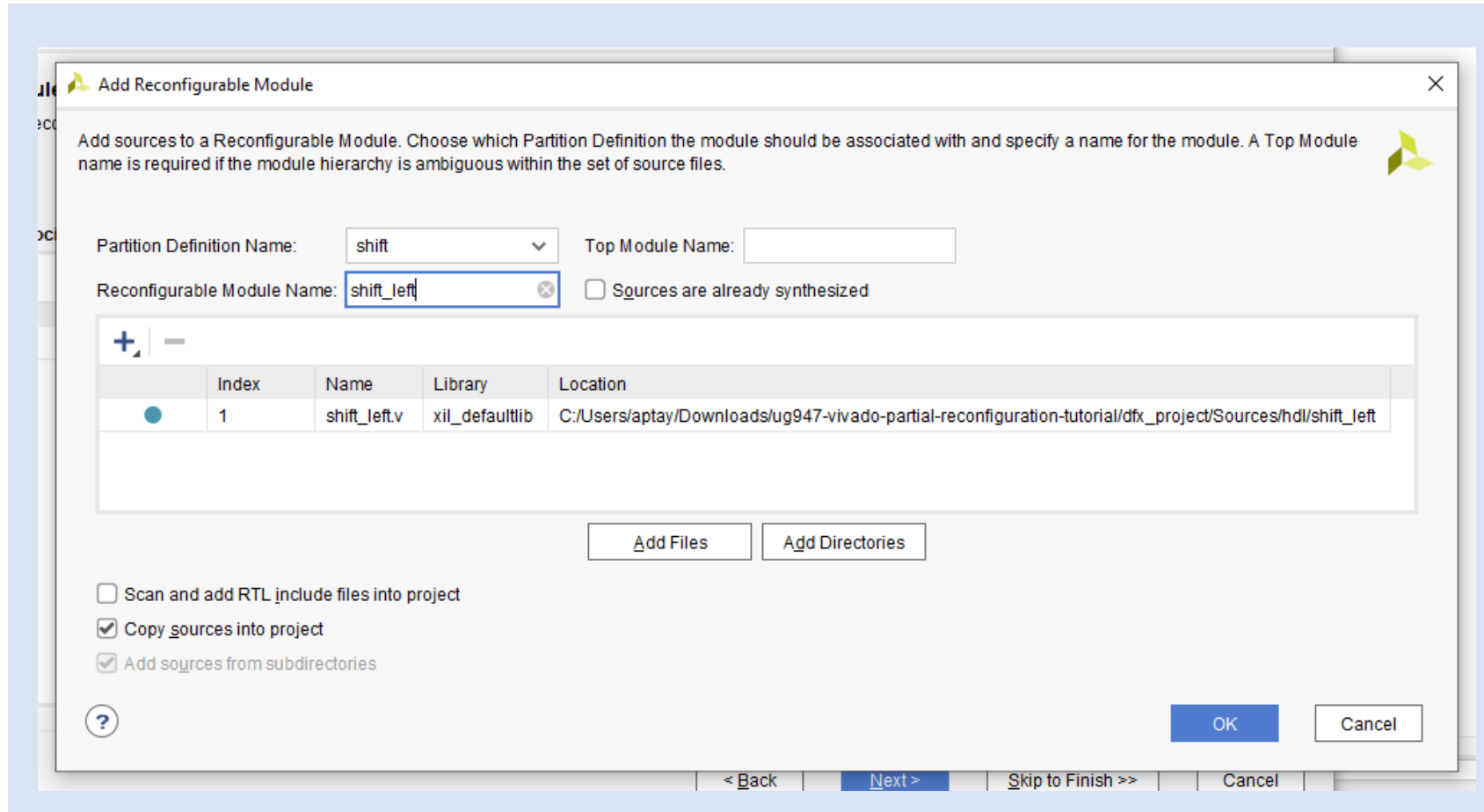
**Step 10** – Click the + to add a new reconfigurable module.





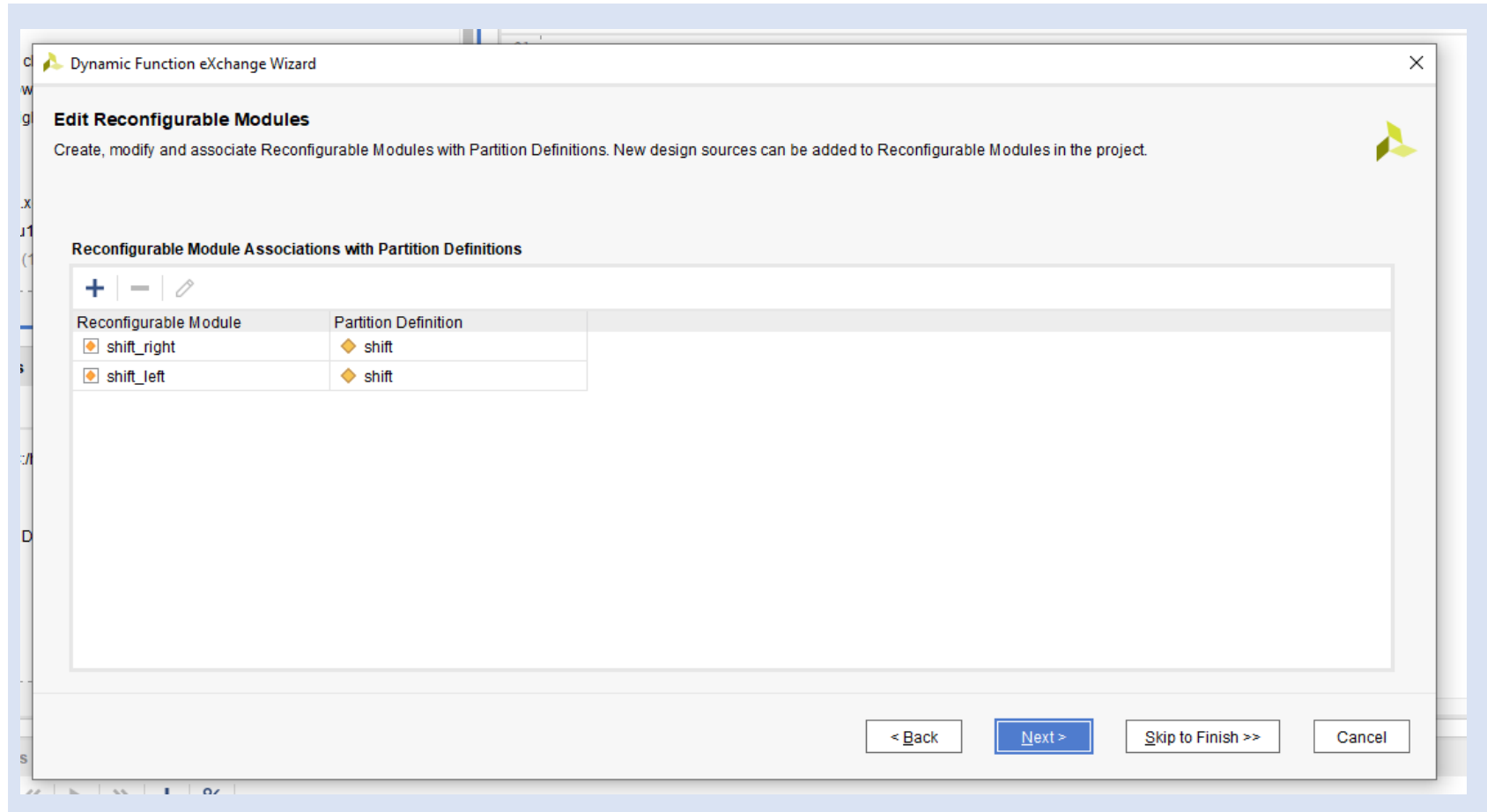
# Lab: DFX

**Step 11** – Add the shift\_left file from the repo set the partition to shift and the reconfigurable name to shift left.



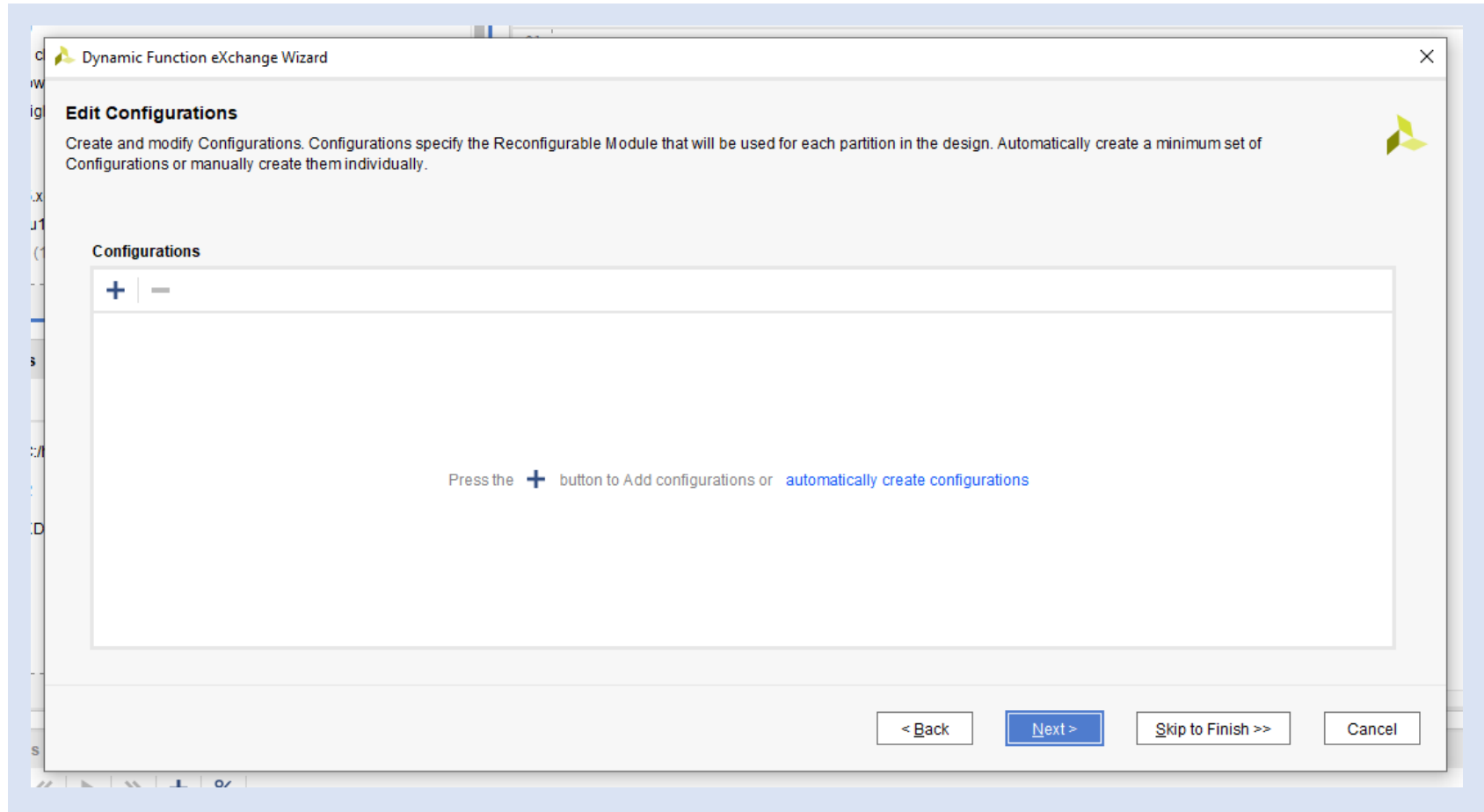
# Lab: DFX

**Step 12** – This will create the reconfigurable partition for the shift left pattern as well as shift right



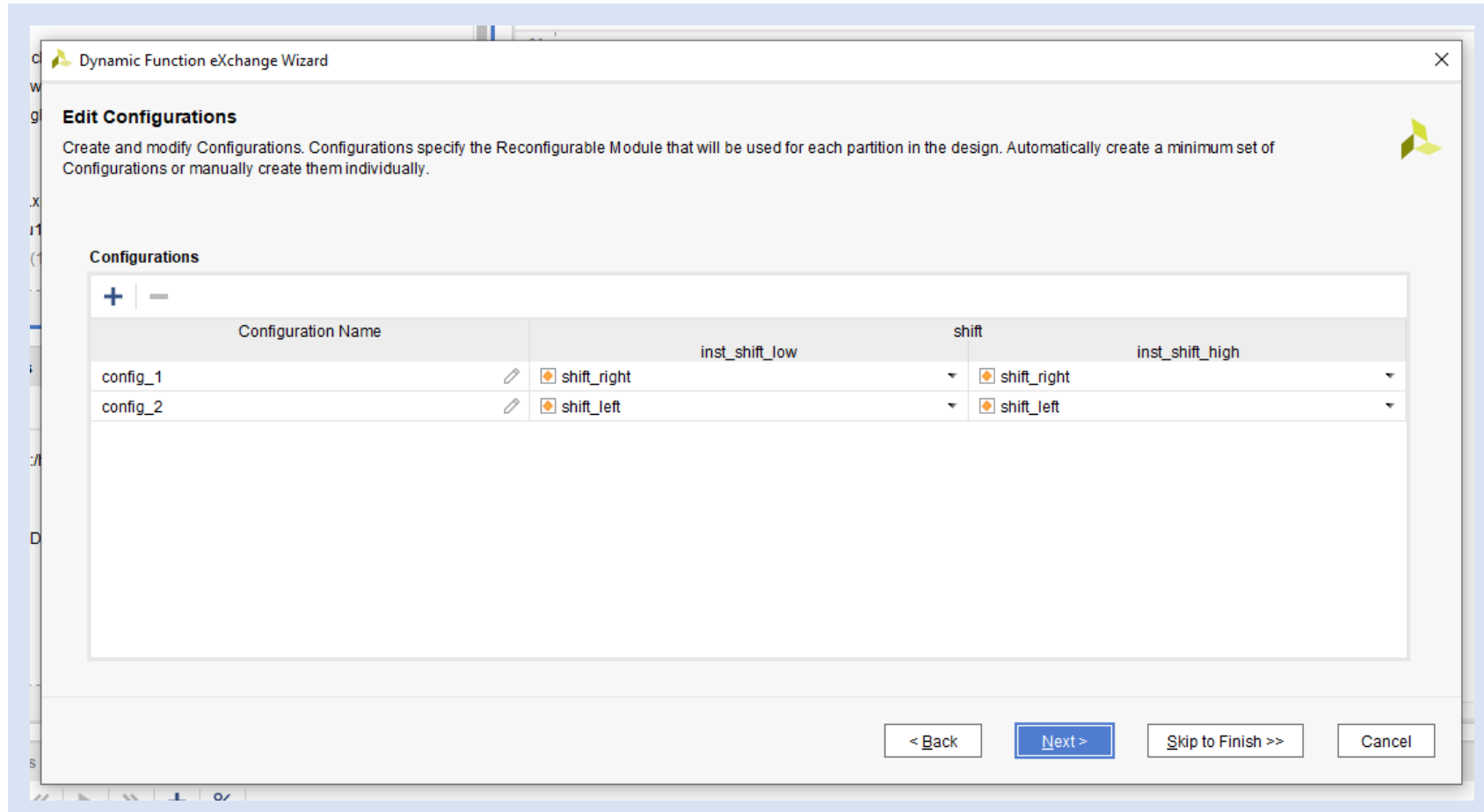
# Lab: DFX

## Step 13 – Click automatically create configurations



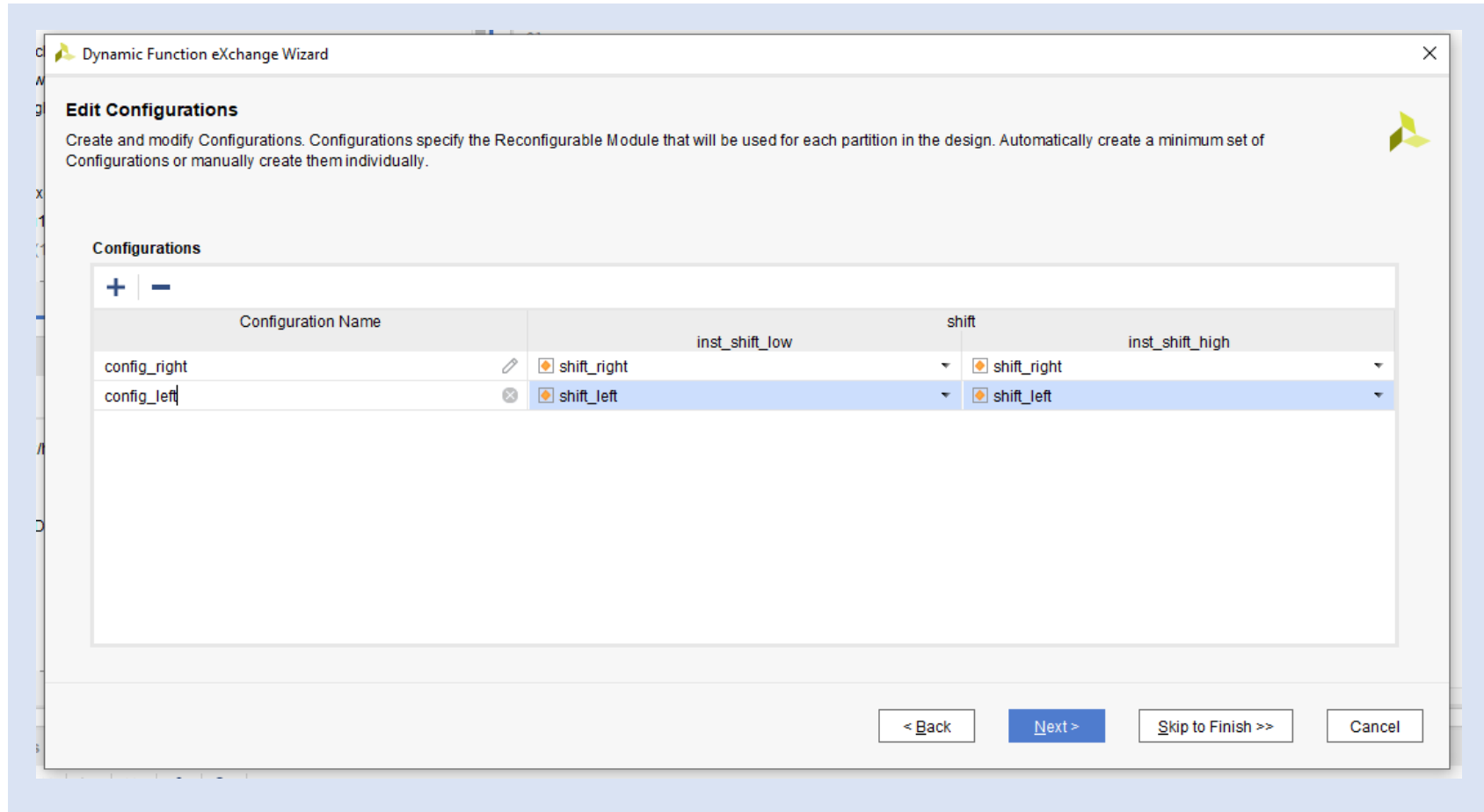
# Lab: DFX

**Step 14** – Once the configurations are created, use the edit feature to name the config\_left and config\_right



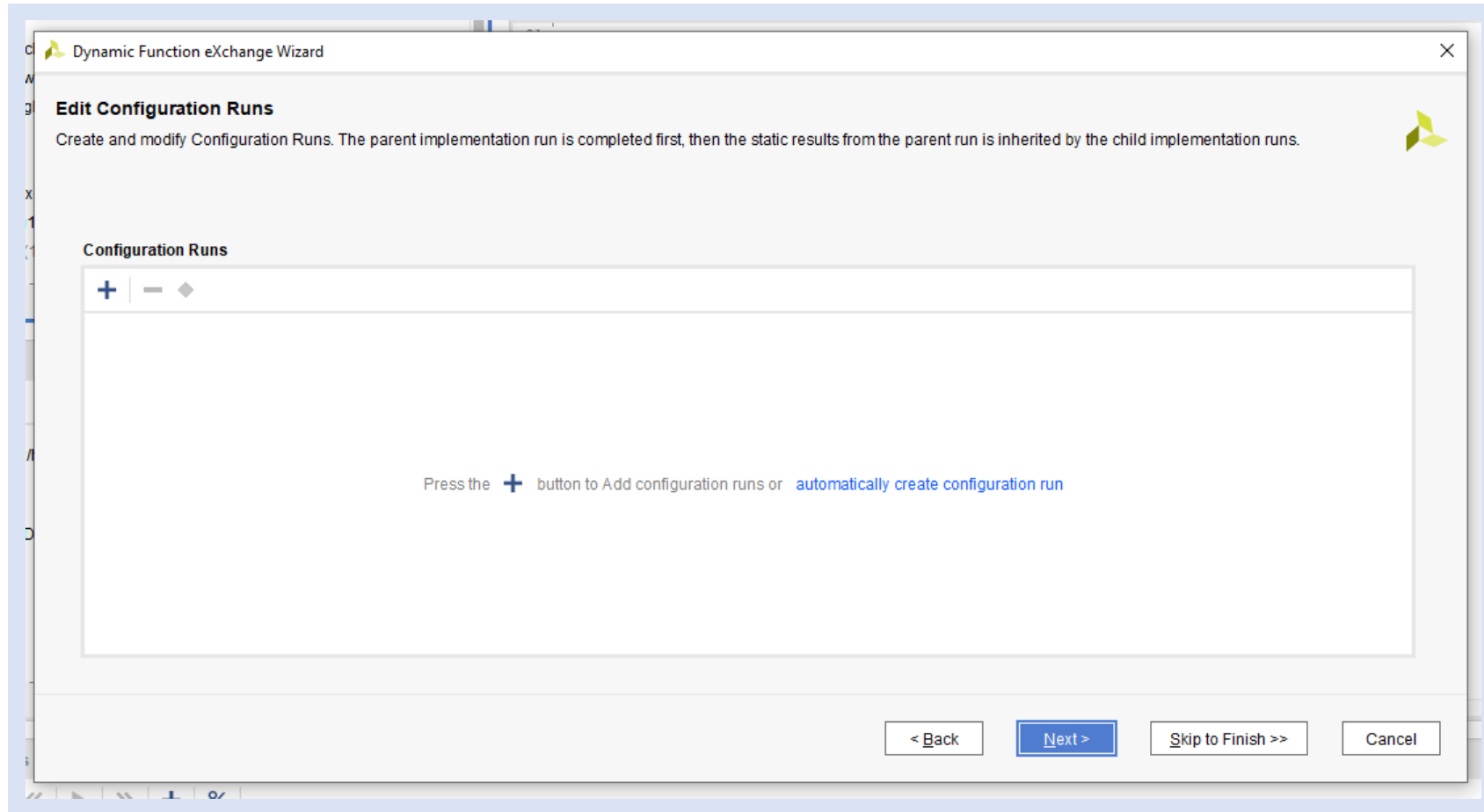
# Lab: DFX

Step 15 – Click next once the modules are renamed



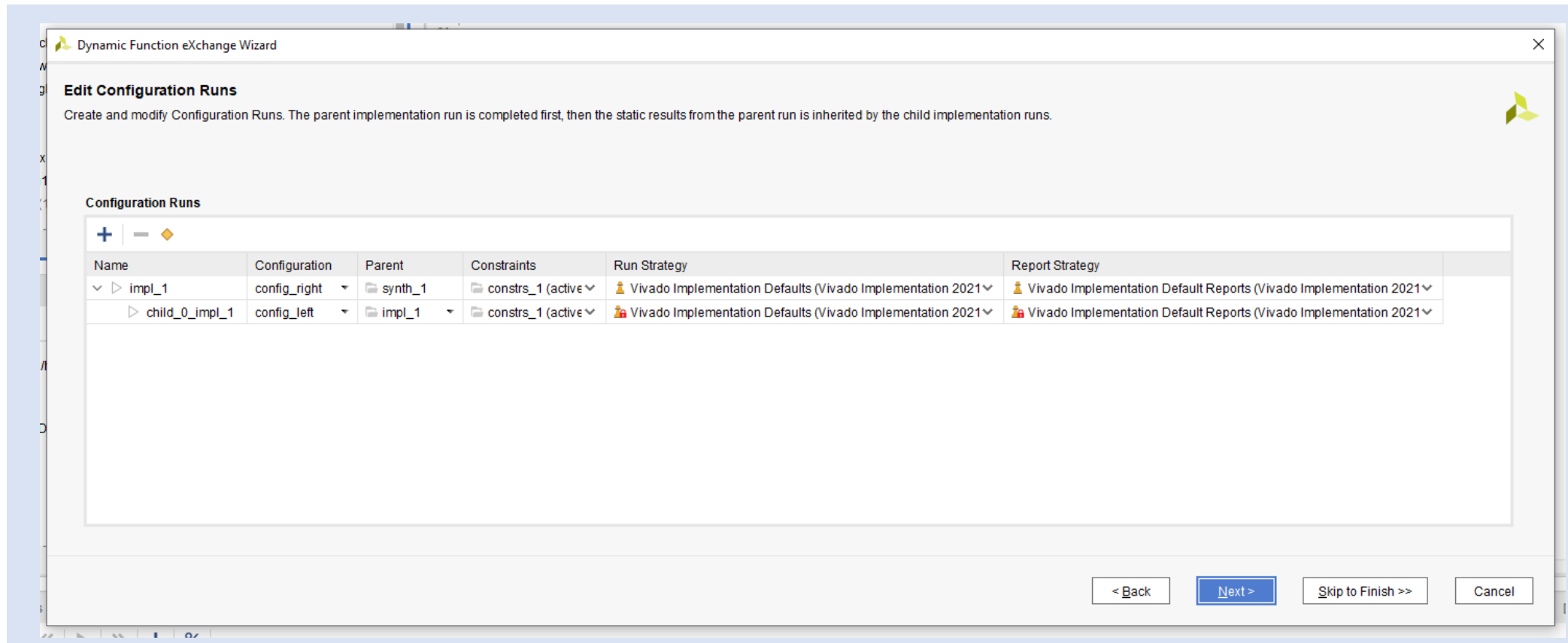
# Lab: DFX

Step 16 – Click on automatically create configuration run



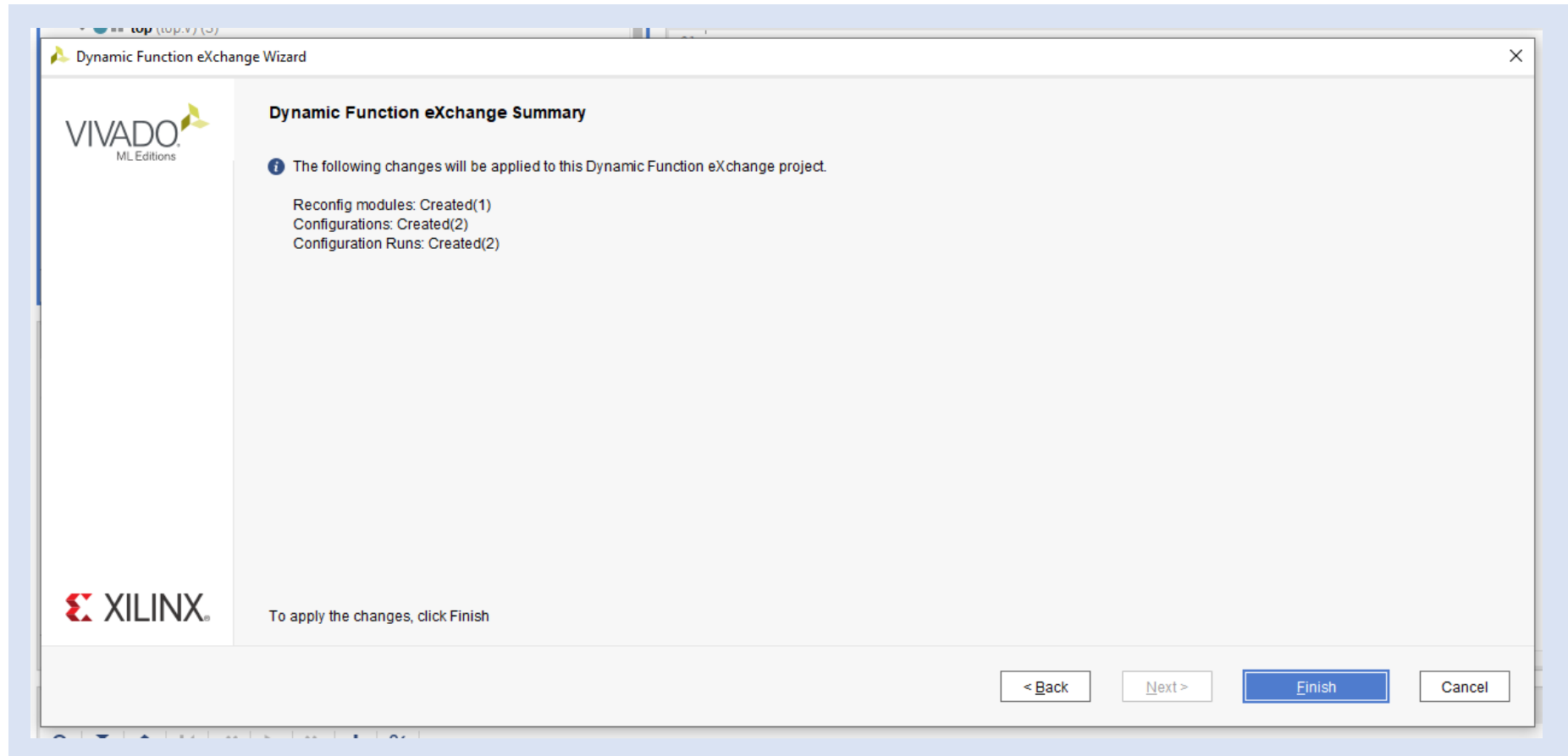
# Lab: DFX

**Step 17** – Once the runs have been created you will see the parent and child configurations. The child configurations are the partial reconfiguration zones. Under the configuration you can change the initial design region contents, if you desire.



# Lab: DFX

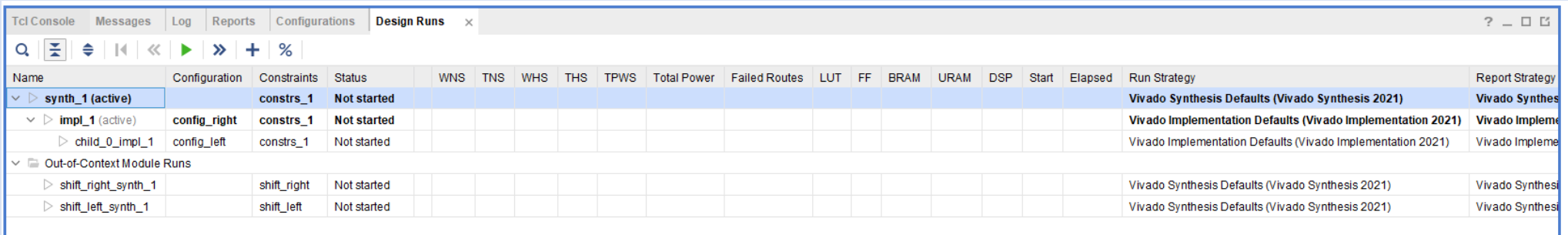
**Step 18** – Click finish, this should have created the runs necessary to create the top level design and the child partial reconfiguration zones.





# Lab: DFX

**Step 19** – Observe the confirmation of the design runs, two RTL modules for OOC synthesis and one implementation set to the configuration selected and a child configured to the other configuration for the partial bit stream.

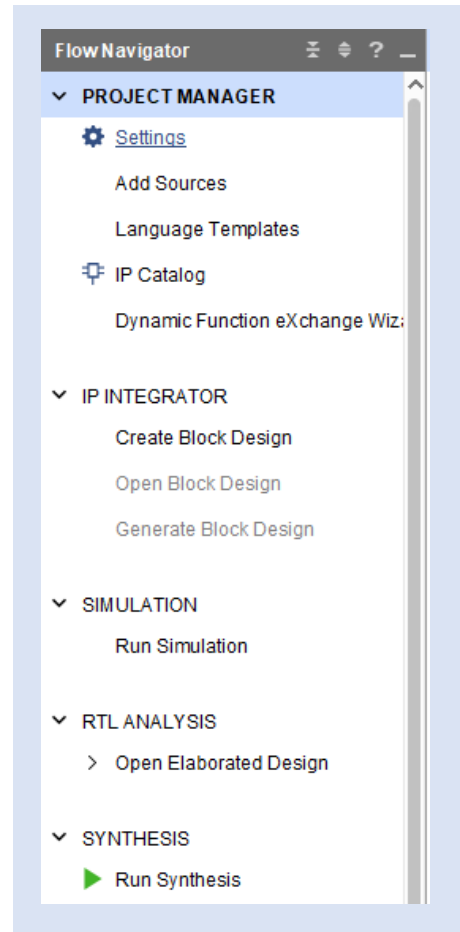


The screenshot shows the 'Design Runs' window in Vivado. The table below represents the data visible in the window.

Name	Configuration	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy
synth_1 (active)		constrs_1	Not started															Vivado Synthesis Defaults (Vivado Synthesis 2021)	Vivado Synthes
impl_1 (active)	config_right	constrs_1	Not started															Vivado Implementation Defaults (Vivado Implementation 2021)	Vivado Impleme
child_0_impl_1	config_left	constrs_1	Not started															Vivado Implementation Defaults (Vivado Implementation 2021)	Vivado Impleme
Out-of-Context Module Runs																			
shift_right_synth_1		shift_right	Not started															Vivado Synthesis Defaults (Vivado Synthesis 2021)	Vivado Synthes
shift_left_synth_1		shift_left	Not started															Vivado Synthesis Defaults (Vivado Synthesis 2021)	Vivado Synthes

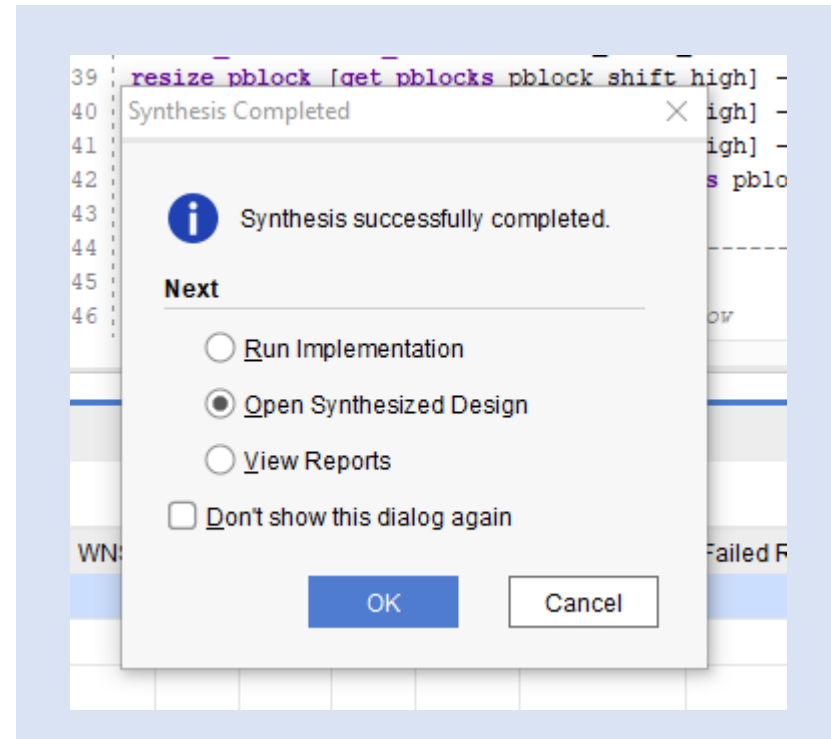
# Lab: DFX

## Step 20 – Run the synthesis



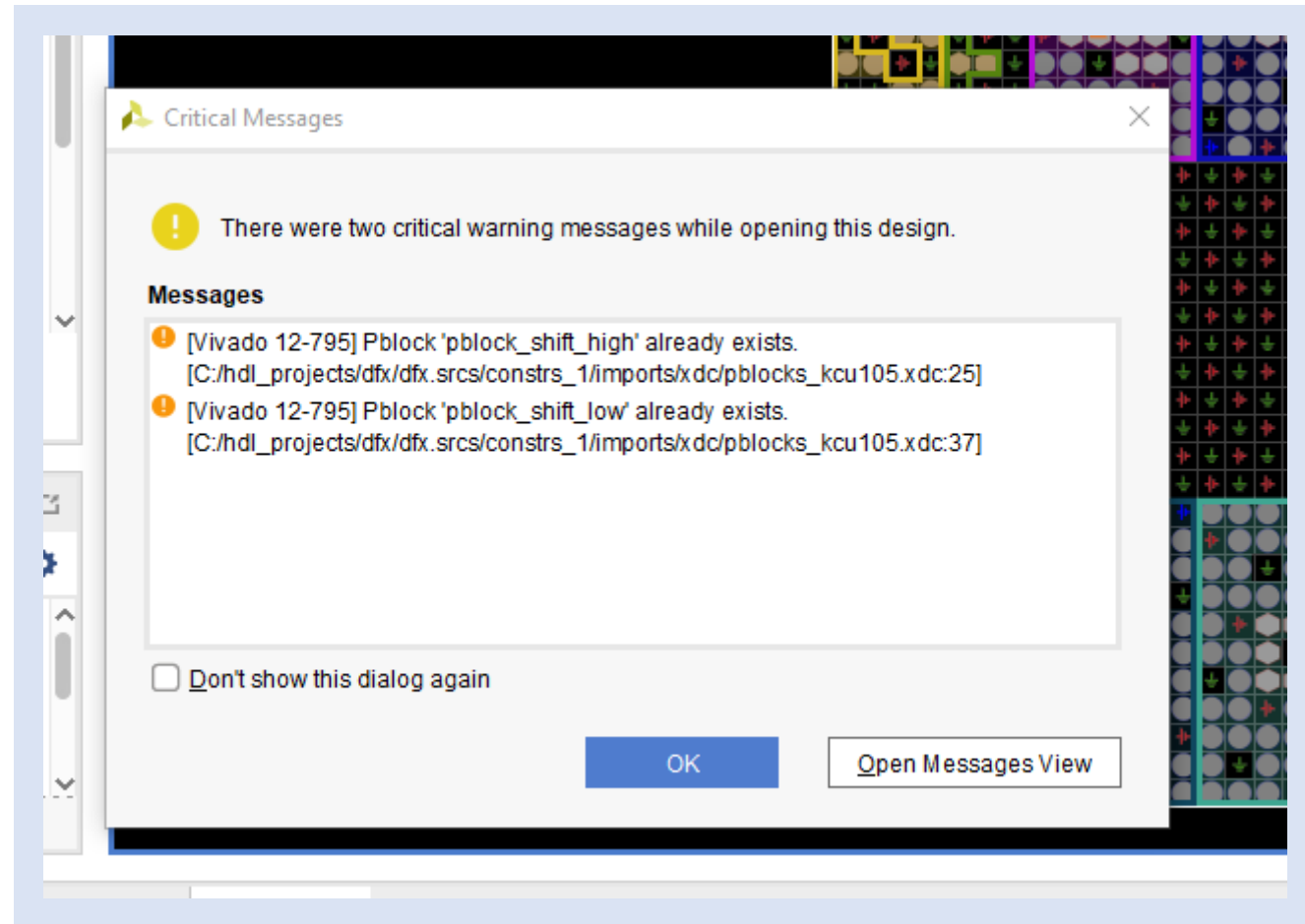
# Lab: DFX

Step 21 – Open the Synthesis view when completed



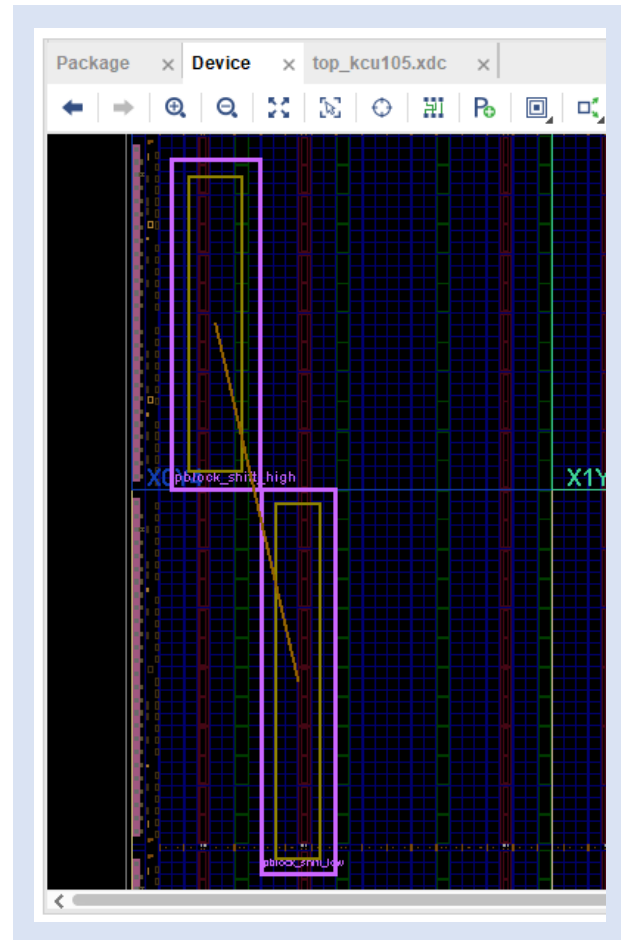
# Lab: DFX

Step 22 – If the following warning appears click OK



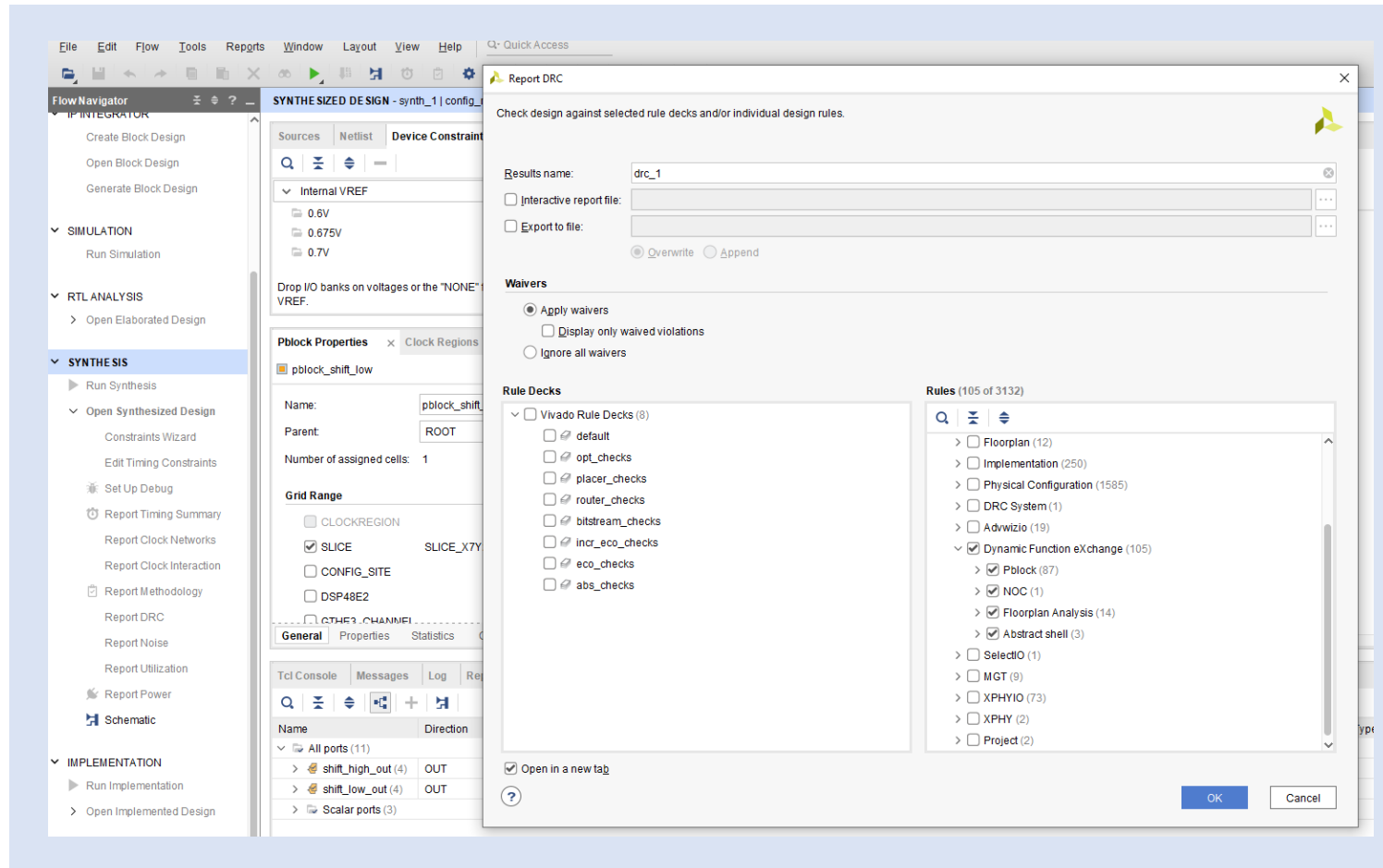
# Lab: DFX

**Step 23** – In the synthesis view you should see the Pblocks with the functions associated with them – this is like an Isolation flow approach.



# Lab: DFX

**Step 24** – From the reports option run the DRC to ensure the Pblocks are OK for the partial reconfiguration



# Lab: DFX

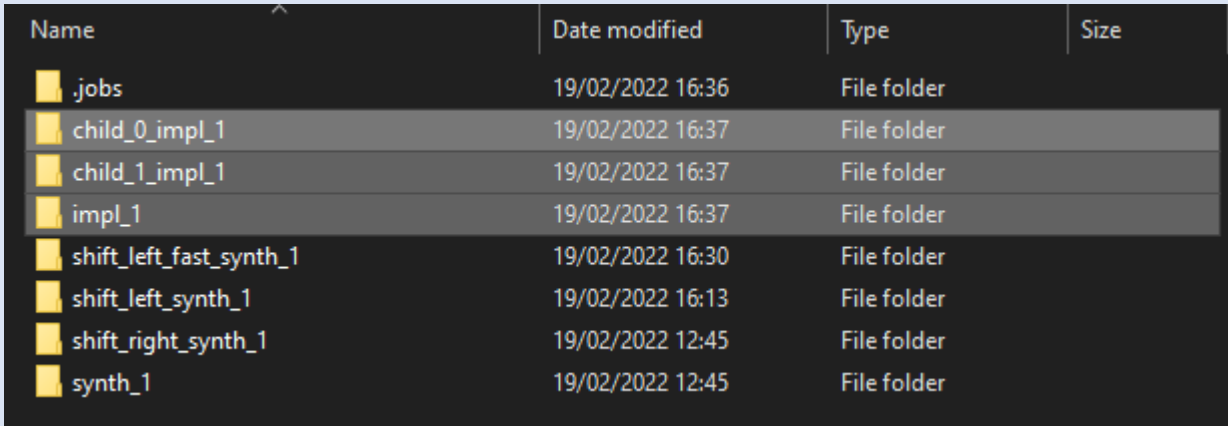
## Step 25 – Implement the design and generate the bitstreams

The screenshot displays the Design Runs table in the Vivado IDE. The table shows the following data:

Name	Configuration	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start
✓ synth_1 (active)		constrs_1	synth_design Complete!								1	35	0.0	0	0	2/19/22, 12:45 PM
✓ impl_1 (active)	config_right	constrs_1	route_design Complete!	8.303	0.000	0.034	0.000	0.000	0.619	0	3	35	2.0	0	0	2/19/22, 1:05 PM
✓ child_0_impl_1	config_left	constrs_1	route_design Complete!	8.303	0.000	0.034	0.000	0.000	0.619	0	3	35	2.0	0	0	2/19/22, 1:07 PM
Out-of-Context Module Runs																
✓ shift_right_synth_1		shift_right	synth_design Complete!								1	0	1.0	0	0	2/19/22, 12:44 PM
✓ shift_left_synth_1		shift_left	synth_design Complete!								1	0	1.0	0	0	2/19/22, 12:44 PM

# Lab: DFX

**Step 26** – The implementation directory will show one implementation for the total design and another for each child created. Under these you will find the complete and the partial bitstreams. Both can be programmed via Hardware Manager into the target board.

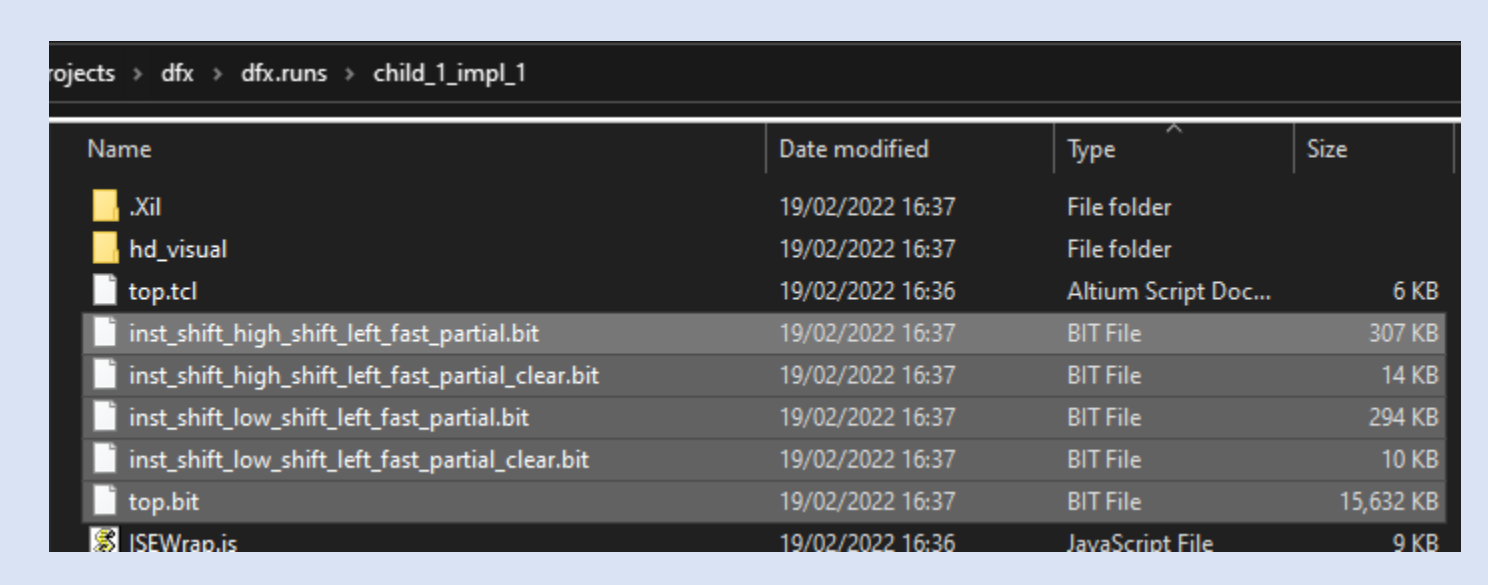


Name	Date modified	Type	Size
.jobs	19/02/2022 16:36	File folder	
child_0_impl_1	19/02/2022 16:37	File folder	
child_1_impl_1	19/02/2022 16:37	File folder	
impl_1	19/02/2022 16:37	File folder	
shift_left_fast_synth_1	19/02/2022 16:30	File folder	
shift_left_synth_1	19/02/2022 16:13	File folder	
shift_right_synth_1	19/02/2022 12:45	File folder	
synth_1	19/02/2022 12:45	File folder	



# Lab: DFX

**Step 27** – Note if targeting a UltraScale device not UltraScale+ the clean.bit must be loaded in first before the partial bitstream.



Name	Date modified	Type	Size
.Xil	19/02/2022 16:37	File folder	
hd_visual	19/02/2022 16:37	File folder	
top.tcl	19/02/2022 16:36	Altium Script Doc...	6 KB
inst_shift_high_shift_left_fast_partial.bit	19/02/2022 16:37	BIT File	307 KB
inst_shift_high_shift_left_fast_partial_clear.bit	19/02/2022 16:37	BIT File	14 KB
inst_shift_low_shift_left_fast_partial.bit	19/02/2022 16:37	BIT File	294 KB
inst_shift_low_shift_left_fast_partial_clear.bit	19/02/2022 16:37	BIT File	10 KB
top.bit	19/02/2022 16:37	BIT File	15,632 KB
ISEWrap.js	19/02/2022 16:36	JavaScript File	9 KB