ADIUVO
ENGINEERING AND TRAINING, LTD.

# Professional PYNQ™ Lab

Adam Taylor
Adam@AdiuvoEngineering.com

ADIUVO
ENGINEERING AND TRAINING, LTD.

## Objective

The objectives of this Lab are:

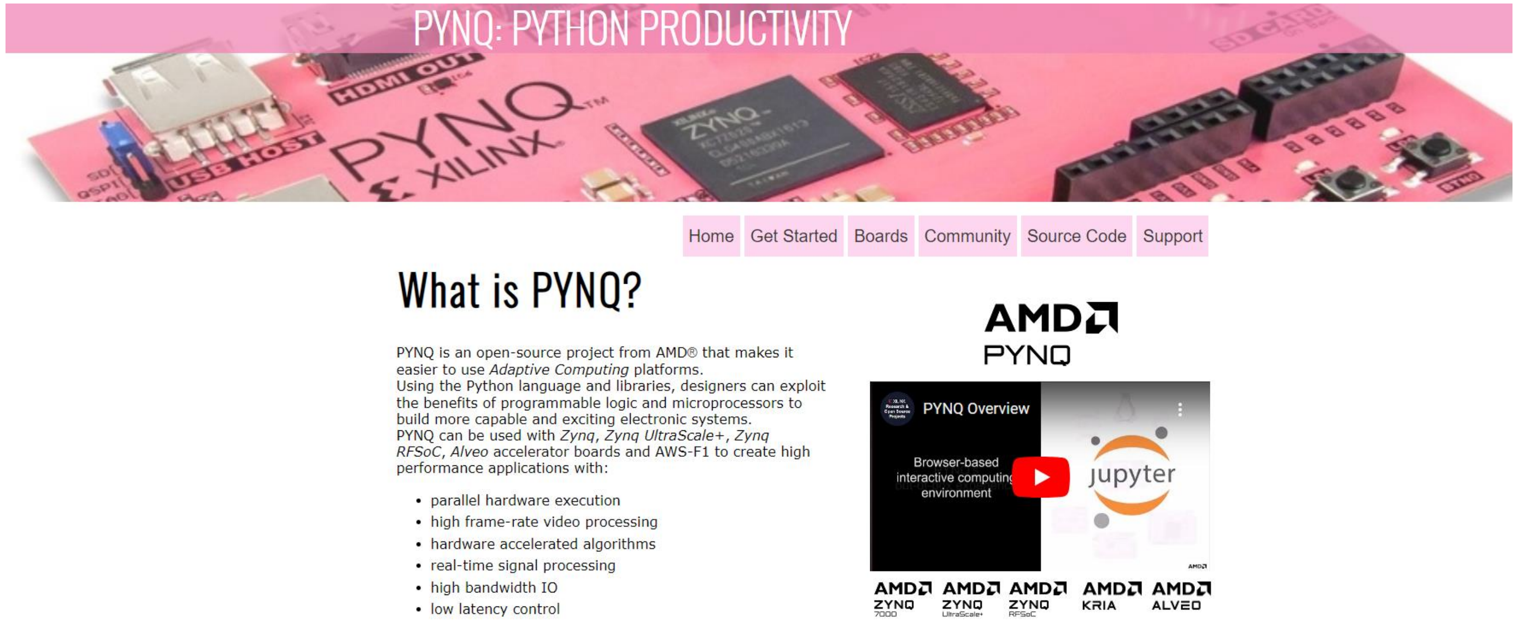
1. Treat the AMD FFT IP core as the unit under test (UUT)
2. Demonstrate how to create an Overlay to test the UUT
3. Demonstrate different methods of control from PS and the impacts possible on performance.
4. Demonstrate how to display performance of the UUT in Jupyter labs

ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Open a browser and go to

[www.pynq.io](http://www.pynq.io)



PYNQ: PYTHON PRODUCTIVITY

## What is PYNQ?

PYNQ is an open-source project from AMD® that makes it easier to use *Adaptive Computing* platforms.
Using the Python language and libraries, designers can exploit the benefits of programmable logic and microprocessors to build more capable and exciting electronic systems.
PYNQ can be used with *Zynq*, *Zynq UltraScale+*, *Zynq RFSoC*, *Alveo* accelerator boards and AWS-F1 to create high performance applications with:

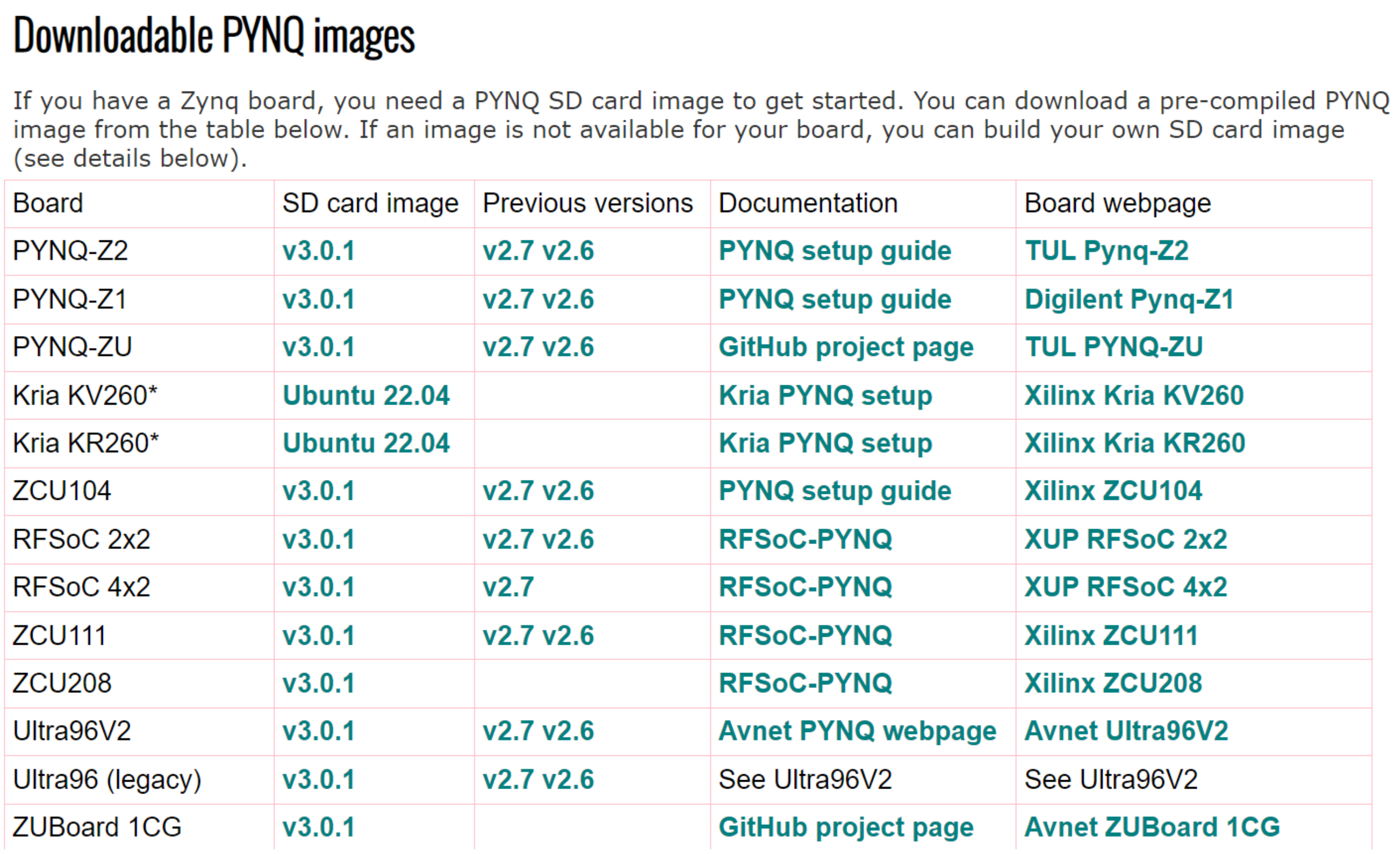
- parallel hardware execution
- high frame-rate video processing
- hardware accelerated algorithms
- real-time signal processing
- high bandwidth IO
- low latency control

# Lab: FFT Verification

Select the boards page and

download the SD card

image for ZU1 CG



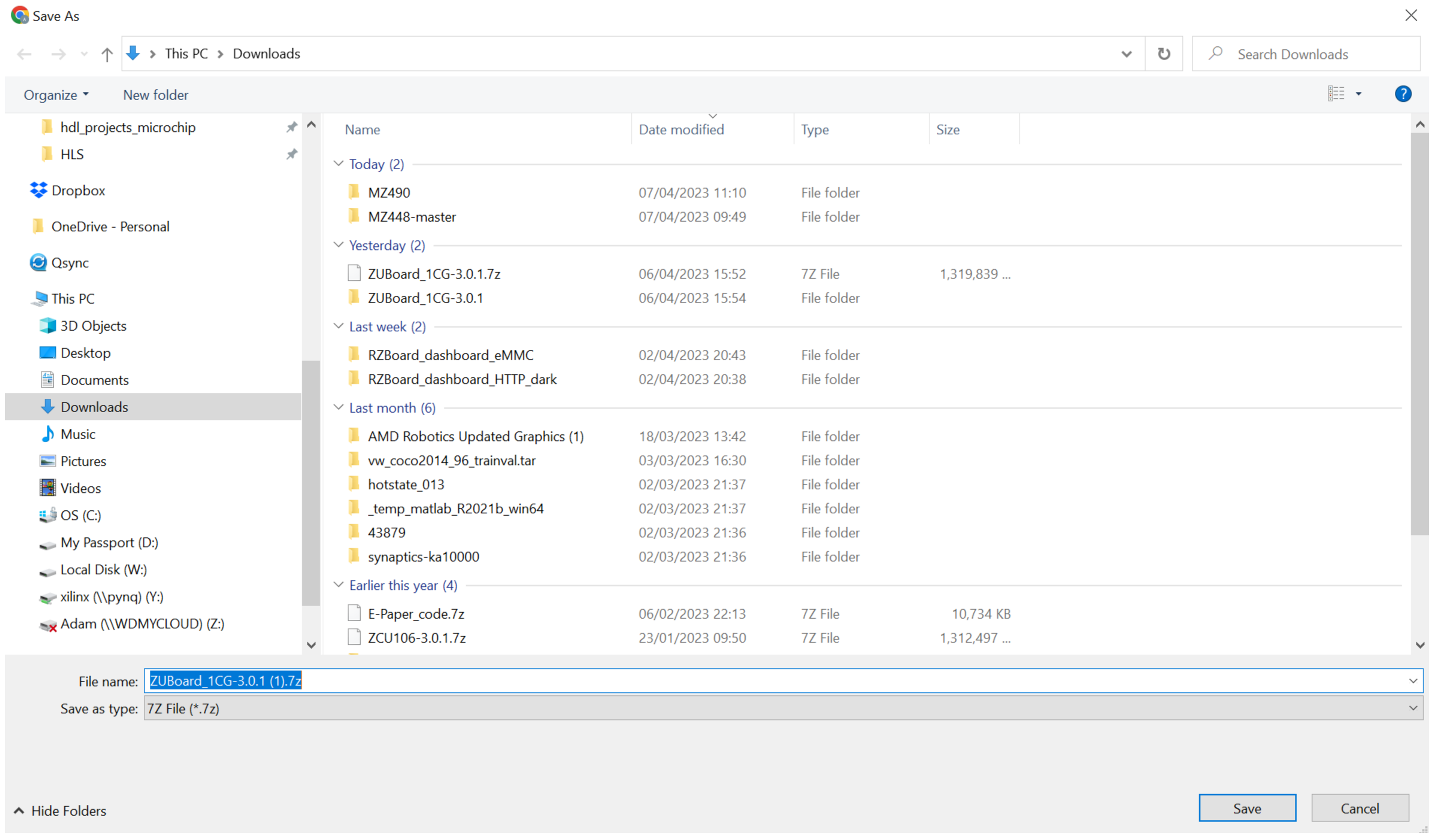
## Downloadable PYNQ images

If you have a Zynq board, you need a PYNQ SD card image to get started. You can download a pre-compiled PYNQ image from the table below. If an image is not available for your board, you can build your own SD card image (see details below).

| Board | SD card image | Previous versions | Documentation | Board webpage |
|---|---|---|---|---|
| PYNQ-Z2 | v3.0.1 | v2.7 v2.6 | PYNQ setup guide | TUL Pynq-Z2 |
| PYNQ-Z1 | v3.0.1 | v2.7 v2.6 | PYNQ setup guide | Digilent Pynq-Z1 |
| PYNQ-ZU | v3.0.1 | v2.7 v2.6 | GitHub project page | TUL PYNQ-ZU |
| Kria KV260* | Ubuntu 22.04 | | Kria PYNQ setup | Xilinx Kria KV260 |
| Kria KR260* | Ubuntu 22.04 | | Kria PYNQ setup | Xilinx Kria KR260 |
| ZCU104 | v3.0.1 | v2.7 v2.6 | PYNQ setup guide | Xilinx ZCU104 |
| RFSoC 2x2 | v3.0.1 | v2.7 v2.6 | RFSoC-PYNQ | XUP RFSoC 2x2 |
| RFSoC 4x2 | v3.0.1 | v2.7 | RFSoC-PYNQ | XUP RFSoC 4x2 |
| ZCU111 | v3.0.1 | v2.7 v2.6 | RFSoC-PYNQ | Xilinx ZCU111 |
| ZCU208 | v3.0.1 | | RFSoC-PYNQ | Xilinx ZCU208 |
| Ultra96V2 | v3.0.1 | v2.7 v2.6 | Avnet PYNQ webpage | Avnet Ultra96V2 |
| Ultra96 (legacy) | v3.0.1 | v2.7 v2.6 | See Ultra96V2 | See Ultra96V2 |
| ZUBoard 1CG | v3.0.1 | | GitHub project page | Avnet ZUBoard 1CG |

# Lab: FFT Verification

Save the SD Card image to a preferred location on your local computer



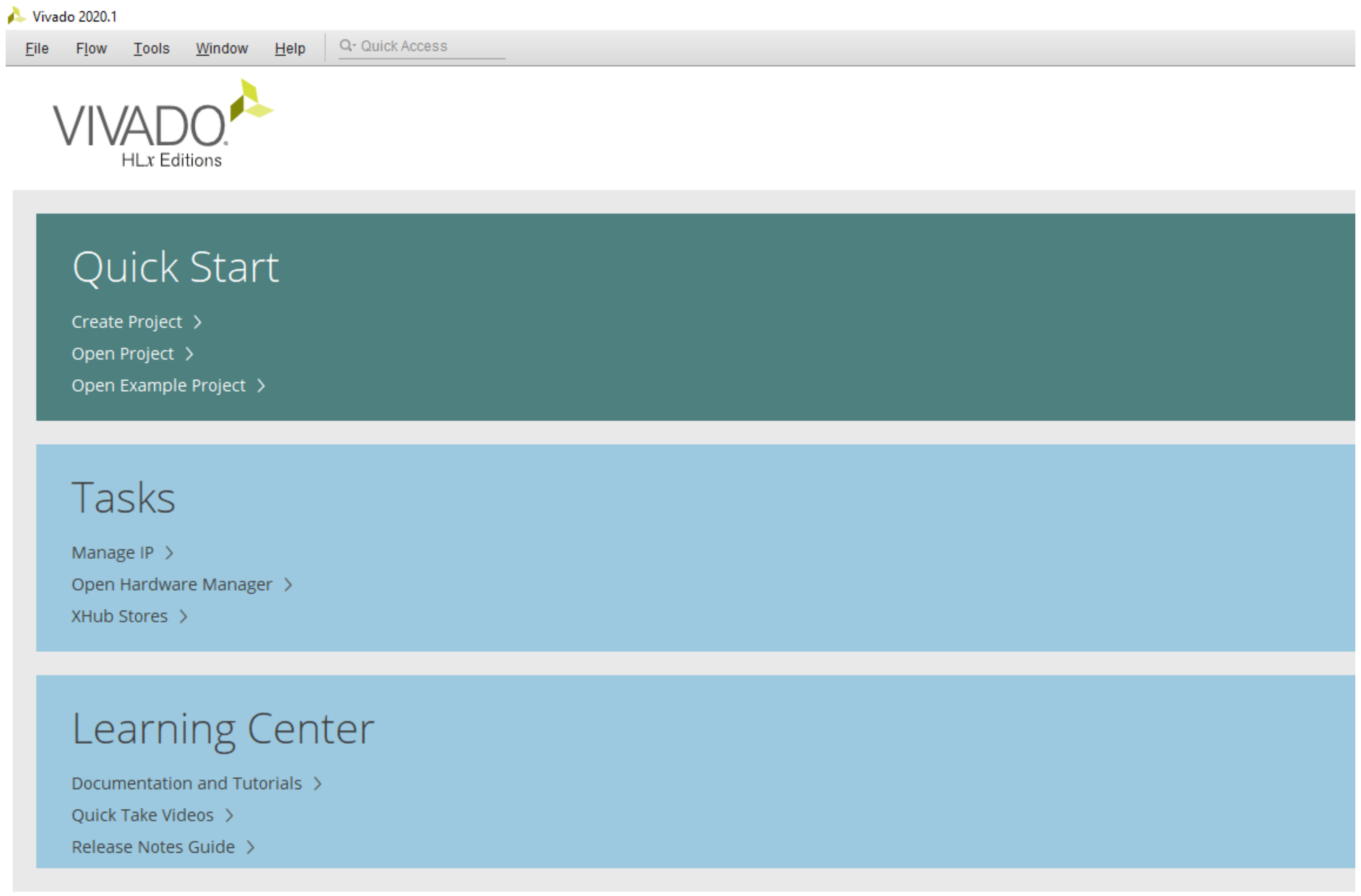
| Name | Date modified | Type | Size |
|---|---|---|---|
| Today (2) | | | |
| MZ490 | 07/04/2023 11:10 | File folder | |
| MZ448-master | 07/04/2023 09:49 | File folder | |
| Yesterday (2) | | | |
| ZUBoard_1CG-3.0.1.7z | 06/04/2023 15:52 | 7Z File | 1,319,839 ... |
| ZUBoard_1CG-3.0.1 | 06/04/2023 15:54 | File folder | |
| Last week (2) | | | |
| RZBoard_dashboard_eMMC | 02/04/2023 20:43 | File folder | |
| RZBoard_dashboard_HTTP_dark | 02/04/2023 20:38 | File folder | |
| Last month (6) | | | |
| AMD Robotics Updated Graphics (1) | 18/03/2023 13:42 | File folder | |
| vw_coco2014_96_trainval.tar | 03/03/2023 16:30 | File folder | |
| hotstate_013 | 02/03/2023 21:37 | File folder | |
| _temp_matlab_R2021b_win64 | 02/03/2023 21:37 | File folder | |
| 43879 | 02/03/2023 21:36 | File folder | |
| synaptics-ka10000 | 02/03/2023 21:36 | File folder | |
| Earlier this year (4) | | | |
| E-Paper_code.7z | 06/02/2023 22:13 | 7Z File | 10,734 KB |
| ZCU106-3.0.1.7z | 23/01/2023 09:50 | 7Z File | 1,312,497 ... |

File name: ZUBoard_1CG-3.0.1 (1).7z
Save as type: 7Z File (*.7z)

# Lab: FFT Verification

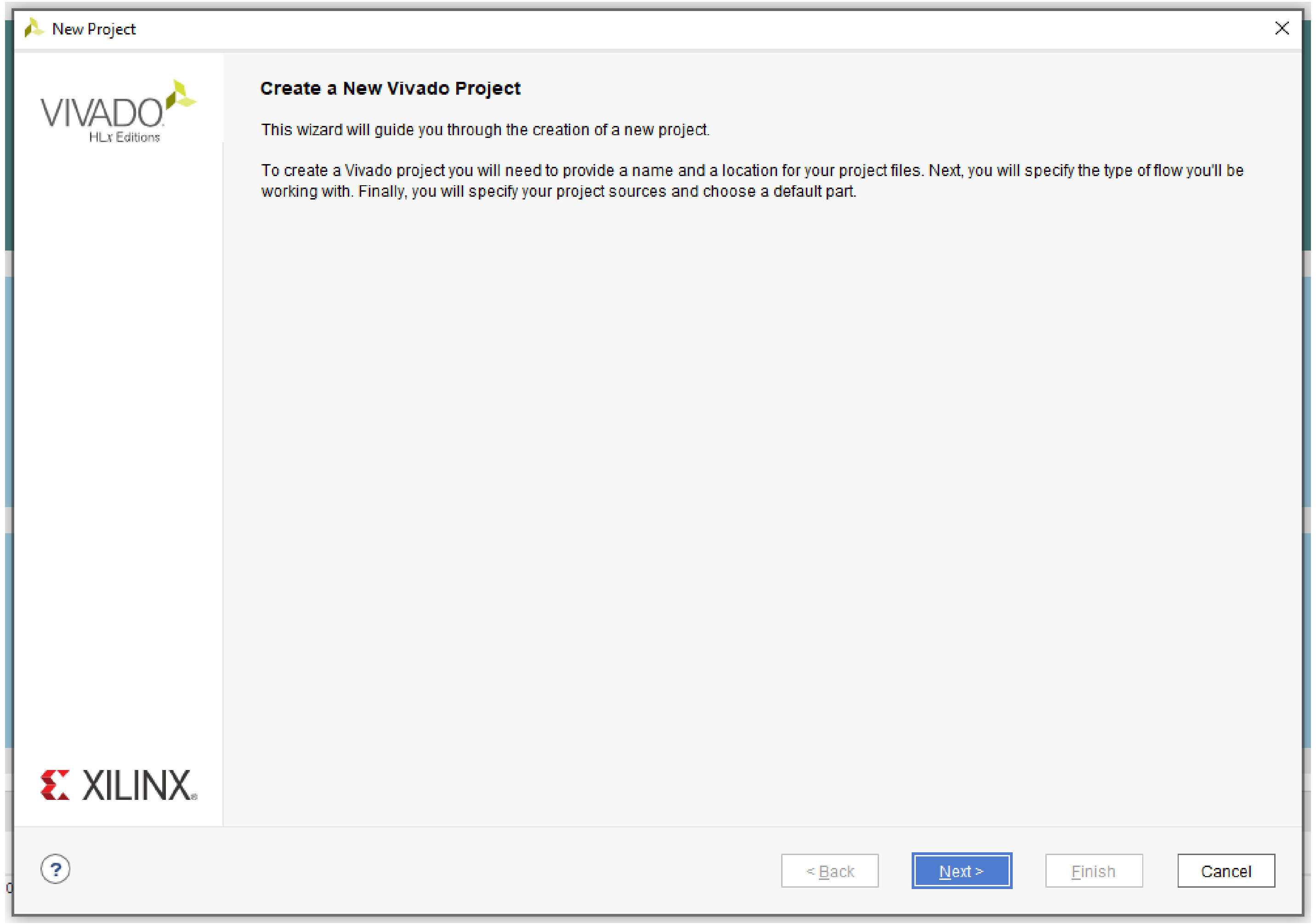
Open Vivado™ and select

Xhub Stores



# Lab: FFT Verification

Create a new project



**Create a New Vivado Project**

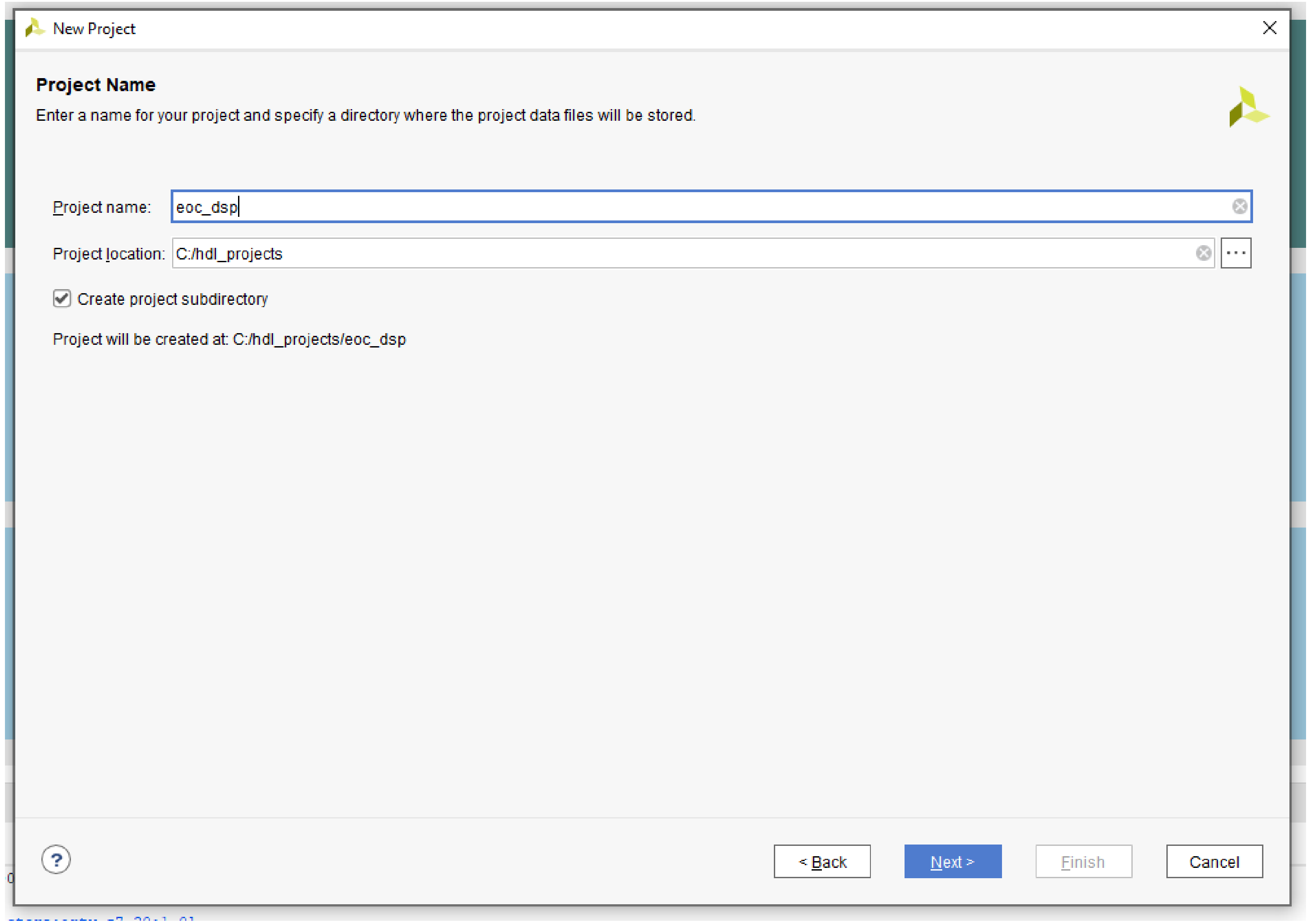
This wizard will guide you through the creation of a new project.

To create a Vivado project you will need to provide a name and a location for your project files. Next, you will specify the type of flow you'll be working with. Finally, you will specify your project sources and choose a default part.

ADIUVO
ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Enter a name and location



New Project

**Project Name**

Enter a name for your project and specify a directory where the project data files will be stored.

Project name: eoc_dsp

Project location: C:/hdl_projects

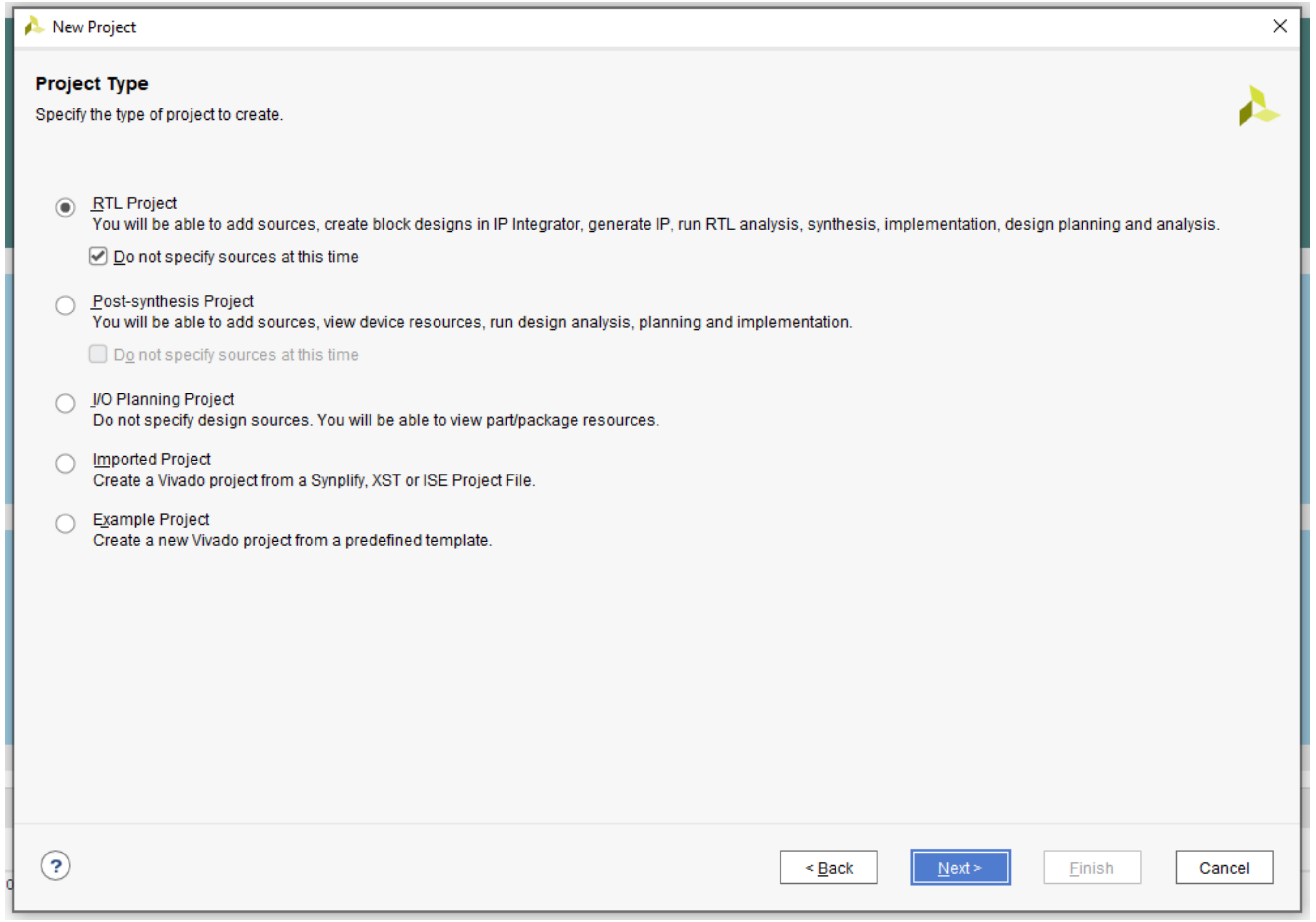
☑ Create project subdirectory

Project will be created at: C:/hdl_projects/eoc_dsp

ADIUVO
ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Select RTL Project


New Project

**Project Type**

Specify the type of project to create.

● RTL Project
You will be able to add sources, create block designs in IP Integrator, generate IP, run RTL analysis, synthesis, implementation, design planning and analysis.
✓ Do not specify sources at this time

○ Post-synthesis Project
You will be able to add sources, view device resources, run design analysis, planning and implementation.
Do not specify sources at this time

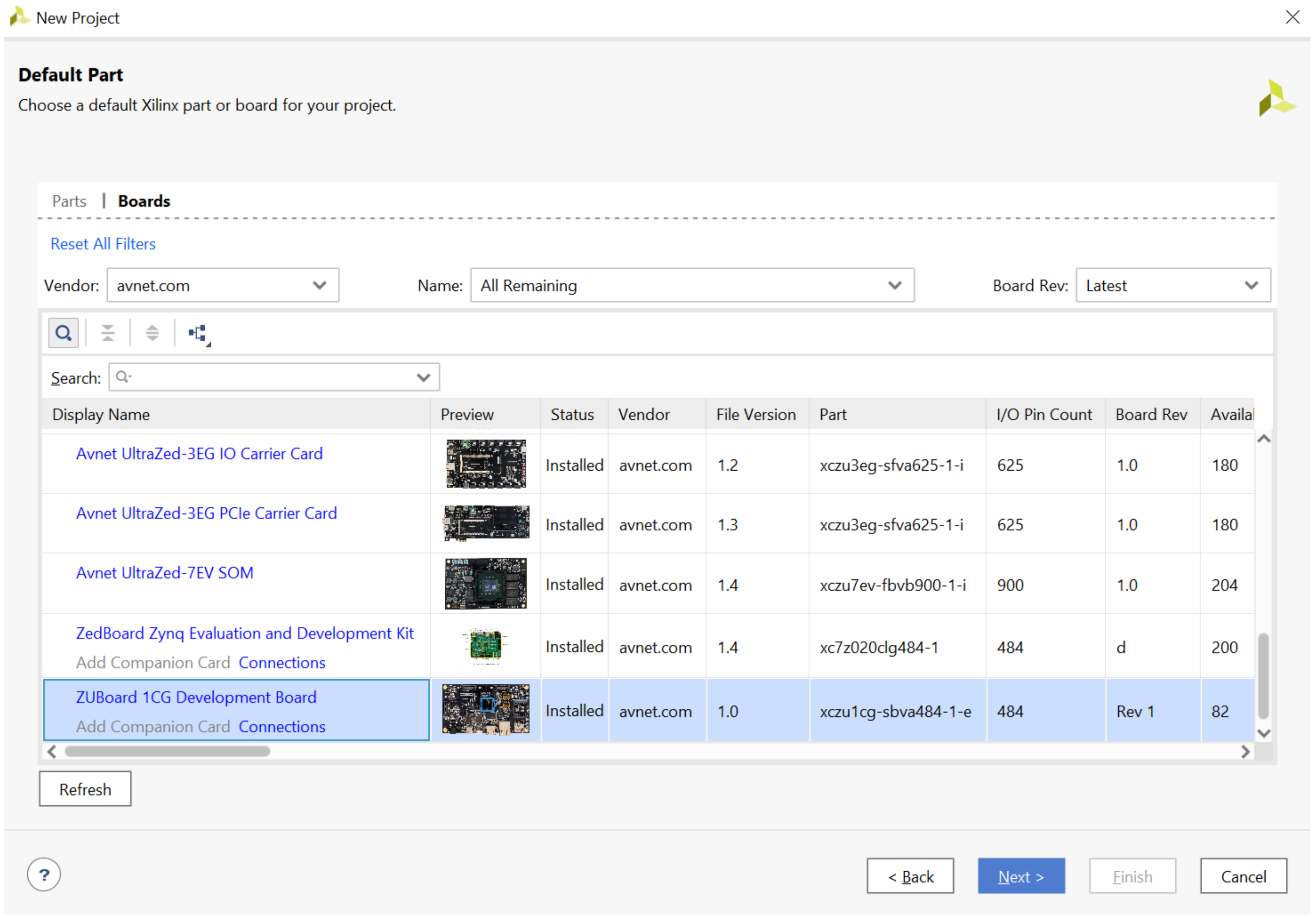
○ I/O Planning Project
Do not specify design sources. You will be able to view part/package resources.

○ Imported Project
Create a Vivado project from a Synplify, XST or ISE Project File.

○ Example Project
Create a new Vivado project from a predefined template.

# Lab: FFT Verification

Select the ZU Board



New Project

**Default Part**

Choose a default Xilinx part or board for your project.

Parts | **Boards**

Reset All Filters

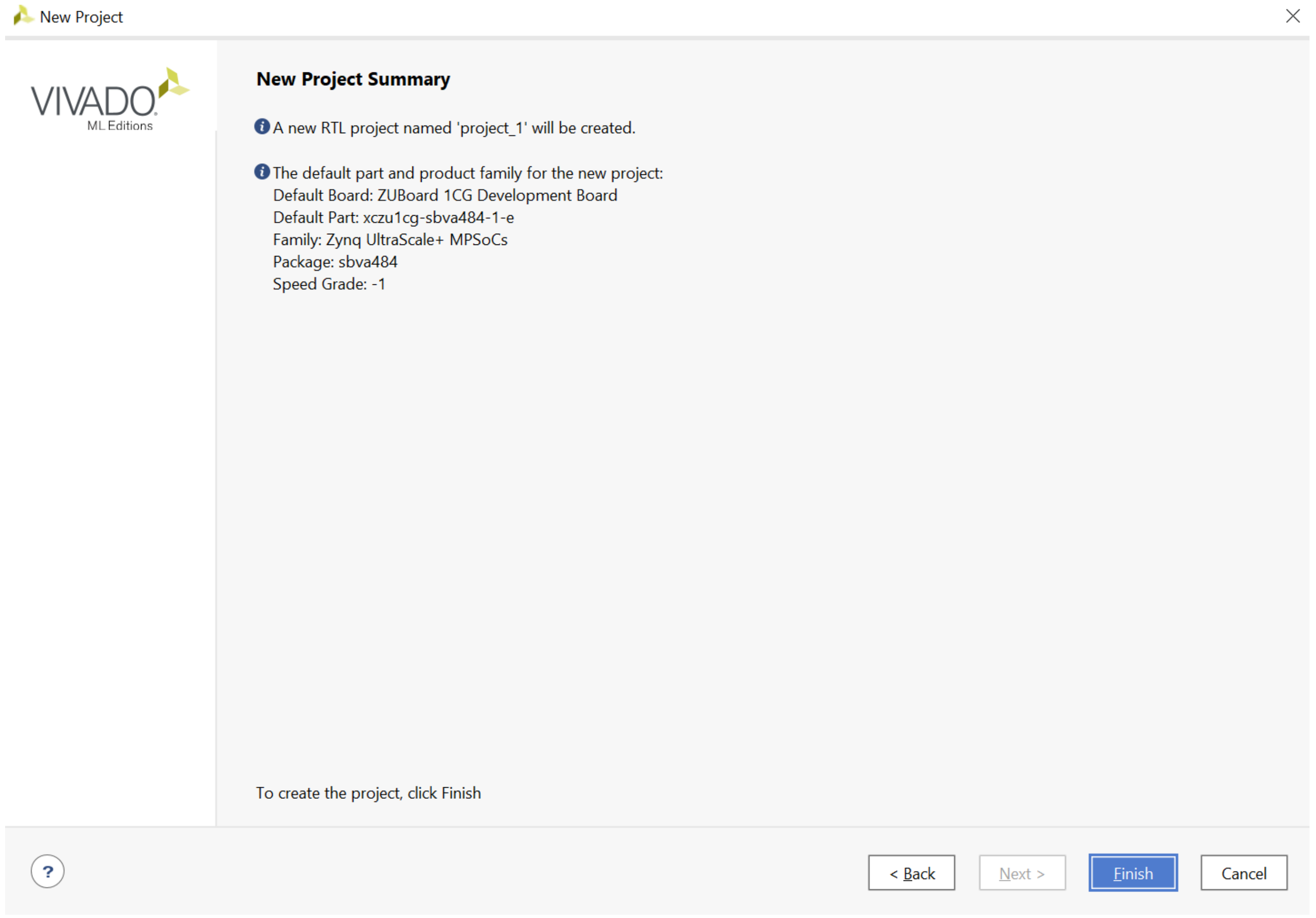
Vendor: avnet.com    Name: All Remaining    Board Rev: Latest

Search:

| Display Name | Preview | Status | Vendor | File Version | Part | I/O Pin Count | Board Rev | Availal |
|---|---|---|---|---|---|---|---|---|
| Avnet UltraZed-3EG IO Carrier Card | | Installed | avnet.com | 1.2 | xczu3eg-sfva625-1-i | 625 | 1.0 | 180 |
| Avnet UltraZed-3EG PCIe Carrier Card | | Installed | avnet.com | 1.3 | xczu3eg-sfva625-1-i | 625 | 1.0 | 180 |
| Avnet UltraZed-7EV SOM | | Installed | avnet.com | 1.4 | xczu7ev-fbvb900-1-i | 900 | 1.0 | 204 |
| ZedBoard Zynq Evaluation and Development Kit<br>Add Companion Card Connections | | Installed | avnet.com | 1.4 | xc7z020clg484-1 | 484 | d | 200 |
| ZUBoard 1CG Development Board<br>Add Companion Card Connections | | Installed | avnet.com | 1.0 | xczu1cg-sbva484-1-e | 484 | Rev 1 | 82 |

# Lab: FFT Verification

Click Finish to create the project



**New Project Summary**

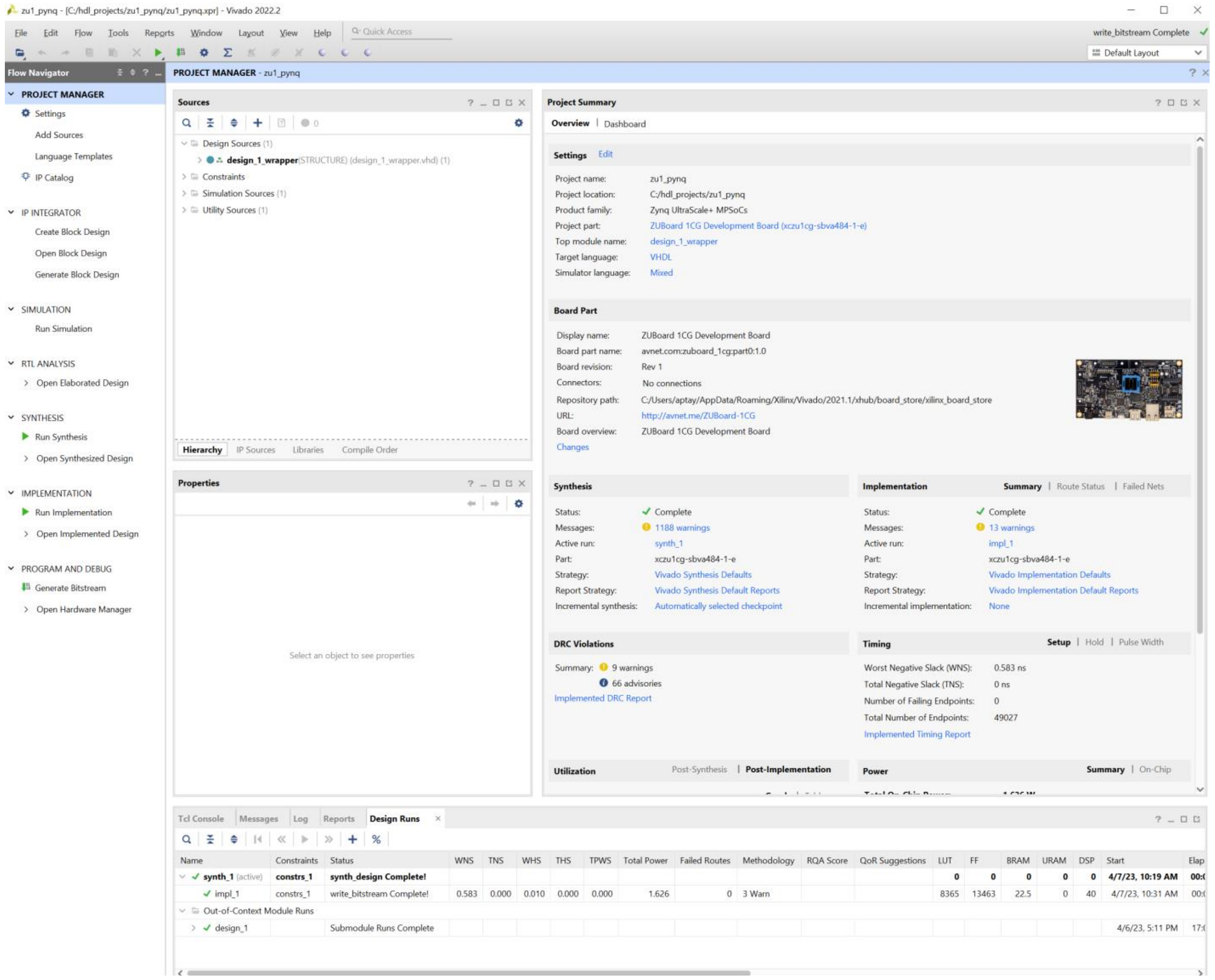
A new RTL project named 'project_1' will be created.

The default part and product family for the new project:
Default Board: ZUBoard 1CG Development Board
Default Part: xczu1cg-sbva484-1-e
Family: Zynq UltraScale+ MPSoCs
Package: sbva484
Speed Grade: -1

To create the project, click Finish

# Lab: FFT Verification

From the Project Manager, select create block diagram



**Settings** Edit

| | |
|---|---|
| Project name: | zu1_pynq |
| Project location: | C:/hdl_projects/zu1_pynq |
| Product family: | Zynq UltraScale+ MPSoCs |
| Project part: | ZUBoard 1CG Development Board (xczu1cg-sbva484-1-e) |
| Top module name: | design_1_wrapper |
| Target language: | VHDL |
| Simulator language: | Mixed |

**Board Part**

| | |
|---|---|
| Display name: | ZUBoard 1CG Development Board |
| Board part name: | avnet.com:zuboard_1cg:part0:1.0 |
| Board revision: | Rev 1 |
| Connectors: | No connections |
| Repository path: | C:/Users/aptay/AppData/Roaming/Xilinx/Vivado/2021.1/xhub/board_store/xilinx_board_store |
| URL: | http://avnet.me/ZUBoard-1CG |
| Board overview: | ZUBoard 1CG Development Board |

Changes

**Synthesis**

| | |
|---|---|
| Status: | ✓ Complete |
| Messages: | 1188 warnings |
| Active run: | synth_1 |
| Part: | xczu1cg-sbva484-1-e |
| Strategy: | Vivado Synthesis Defaults |
| Report Strategy: | Vivado Synthesis Default Reports |
| Incremental synthesis: | Automatically selected checkpoint |

**Implementation** Summary | Route Status | Failed Nets

| | |
|---|---|
| Status: | ✓ Complete |
| Messages: | 13 warnings |
| Active run: | impl_1 |
| Part: | xczu1cg-sbva484-1-e |
| Strategy: | Vivado Implementation Defaults |
| Report Strategy: | Vivado Implementation Default Reports |
| Incremental implementation: | None |

**DRC Violations**

Summary: 9 warnings
66 advisories
Implemented DRC Report

**Timing** Setup | Hold | Pulse Width

| | |
|---|---|
| Worst Negative Slack (WNS): | 0.583 ns |
| Total Negative Slack (TNS): | 0 ns |
| Number of Failing Endpoints: | 0 |
| Total Number of Endpoints: | 49027 |

Implemented Timing Report

**Utilization** Post-Synthesis | Post-Implementation

**Power** Summary | On-Chip

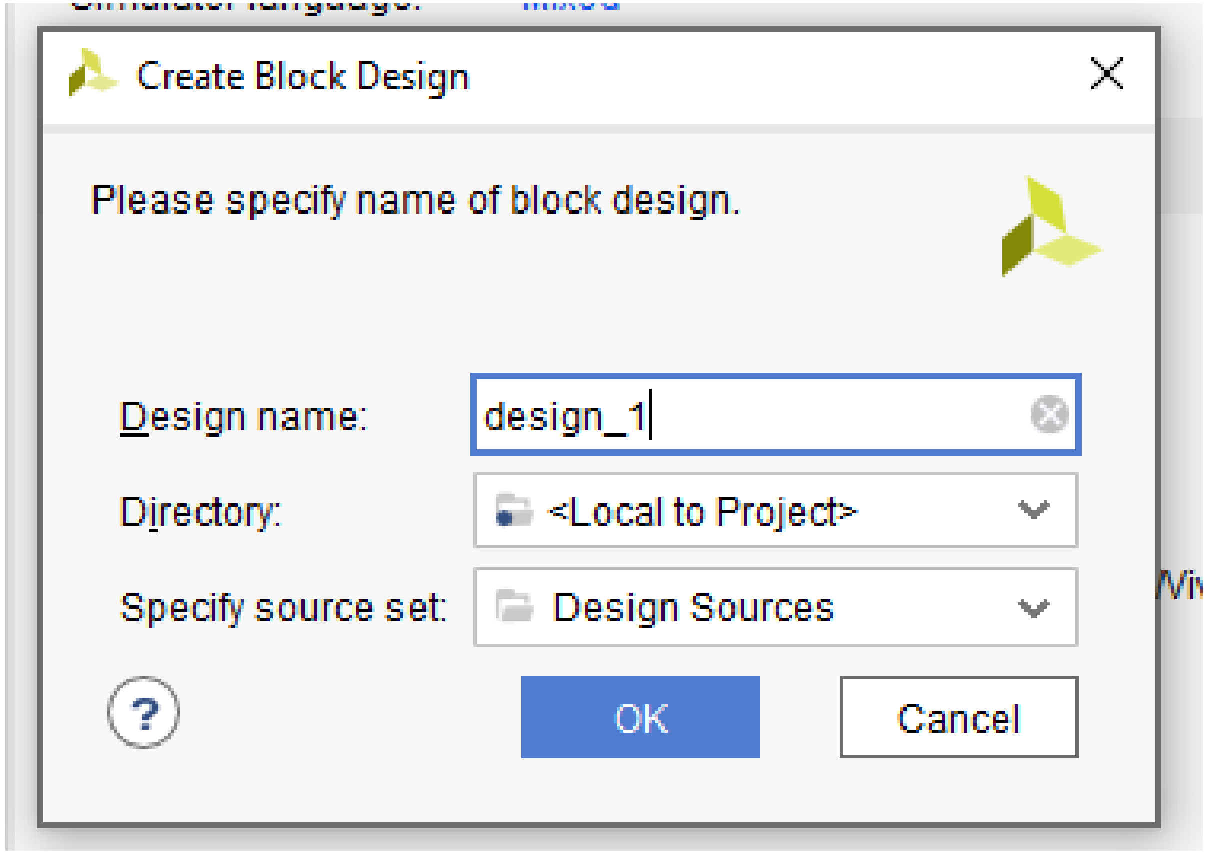
Tcl Console | Messages | Log | Reports | **Design Runs**

| Name | Constraints | Status | WNS | TNS | WHS | THS | TPWS | Total Power | Failed Routes | Methodology | RQA Score | QoR Suggestions | LUT | FF | BRAM | URAM | DSP | Start | Elap |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ✓ synth_1 (active) | constrs_1 | synth_design Complete! | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 4/7/23, 10:19 AM | 00: |
| ✓ impl_1 | constrs_1 | write_bitstream Complete! | 0.583 | 0.000 | 0.010 | 0.000 | 0.000 | 1.626 | 0 | 3 Warn | | | 8365 | 13463 | 22.5 | 0 | 40 | 4/7/23, 10:31 AM | 00: |
| Out-of-Context Module Runs | | | | | | | | | | | | | | | | | | | |
| ✓ design_1 | | Submodule Runs Complete | | | | | | | | | | | | | | | | 4/6/23, 5:11 PM | 17: |

ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Leave, defaults unchanged and click OK



Create Block Design

Please specify name of block design.

Design name: design_1

Directory: <Local to Project>

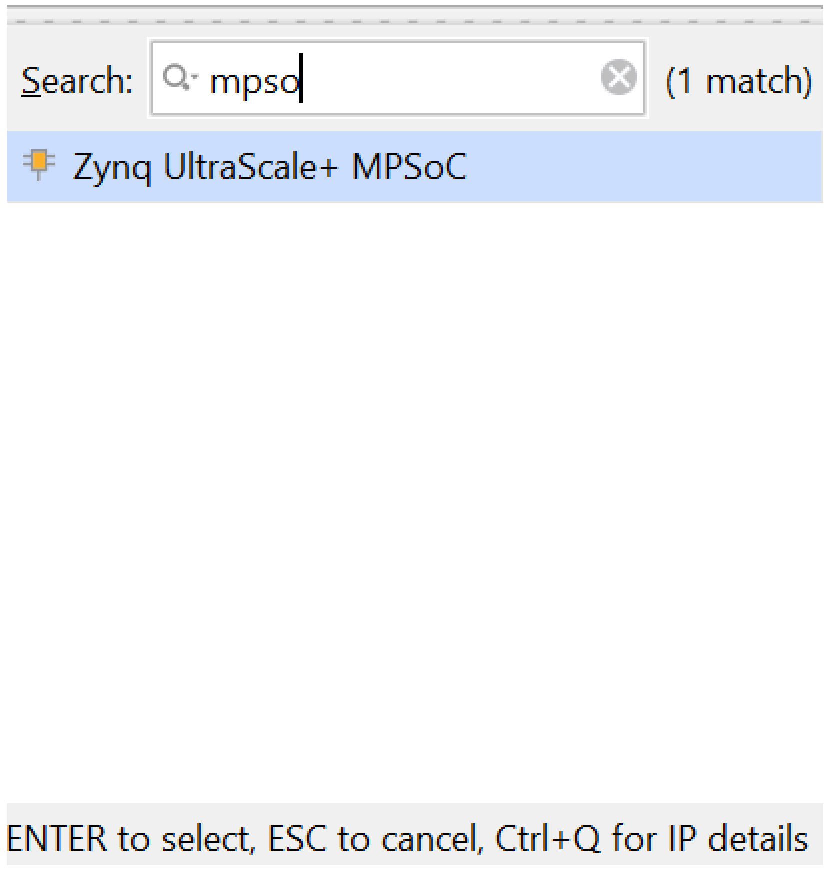
Specify source set: Design Sources

# Lab: FFT Verification

Click on + and in the

search bar type in mpsoc

and press enter

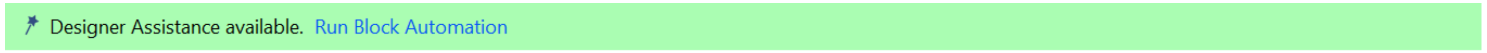

Search: mpso (1 match)

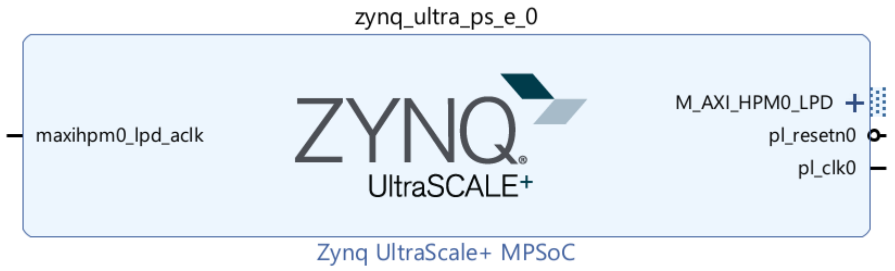
Zynq UltraScale+ MPSoC

ENTER to select, ESC to cancel, Ctrl+Q for IP details

# Lab: FFT Verification

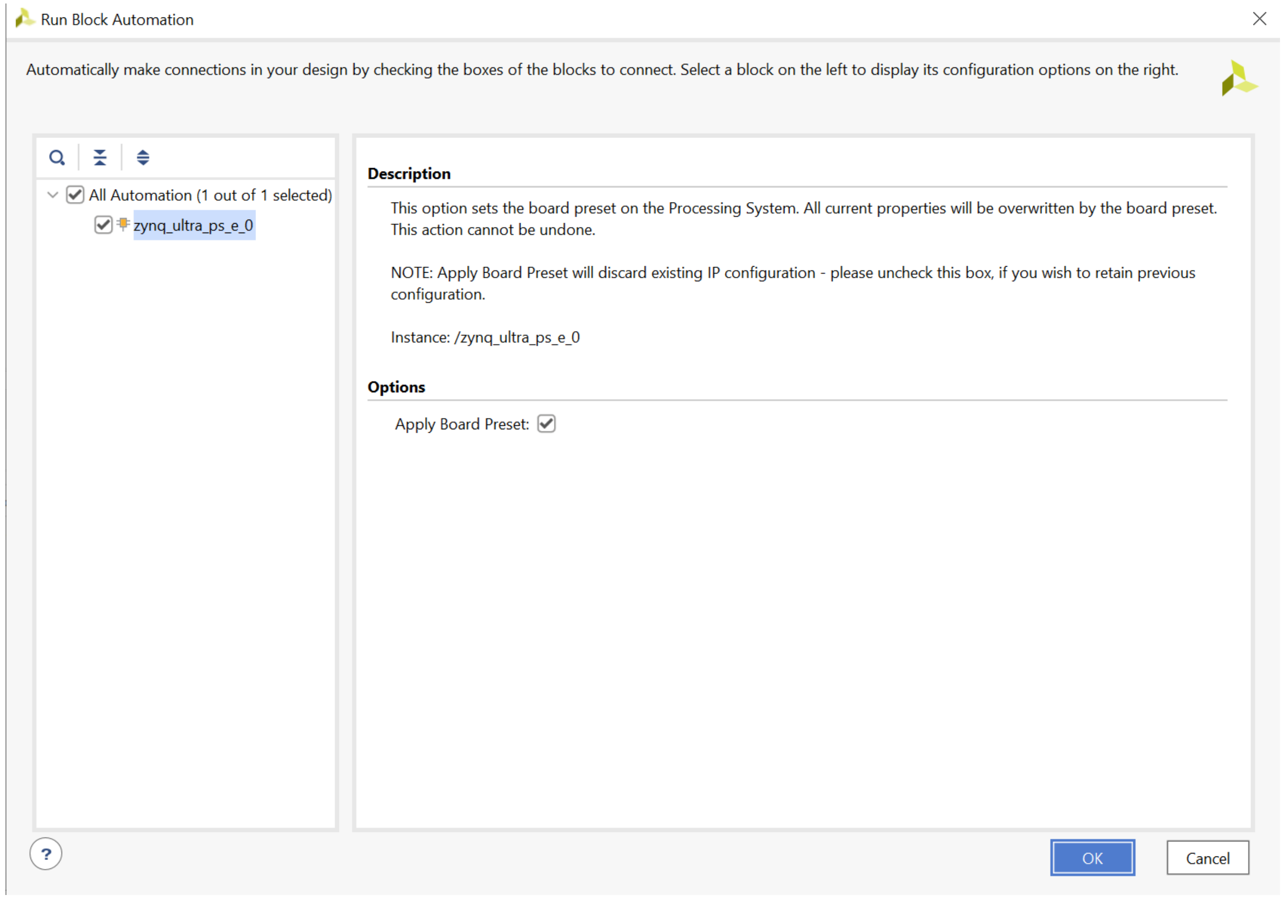
Run the block automation


Designer Assistance available. Run Block Automation



## Lab: FFT Verification

Leave the settings as default and click OK


Run Block Automation

Automatically make connections in your design by checking the boxes of the blocks to connect. Select a block on the left to display its configuration options on the right.

- [✓] All Automation (1 out of 1 selected)
  - [✓] zynq_ultra_ps_e_0

**Description**

This option sets the board preset on the Processing System. All current properties will be overwritten by the board preset. This action cannot be undone.

NOTE: Apply Board Preset will discard existing IP configuration - please uncheck this box, if you wish to retain previous configuration.

Instance: /zynq_ultra_ps_e_0

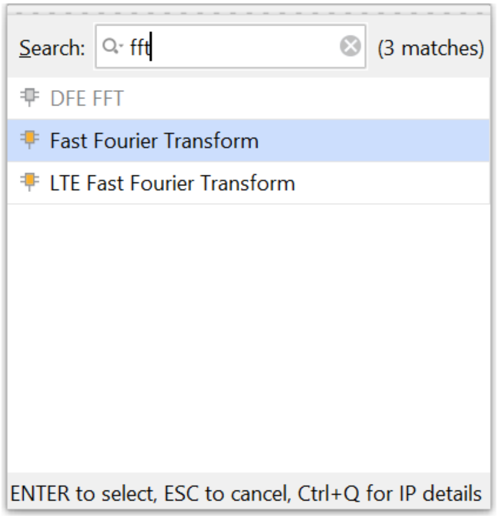
**Options**

Apply Board Preset: ✓

# Lab: FFT Verification

Click on + and add in the

FFT

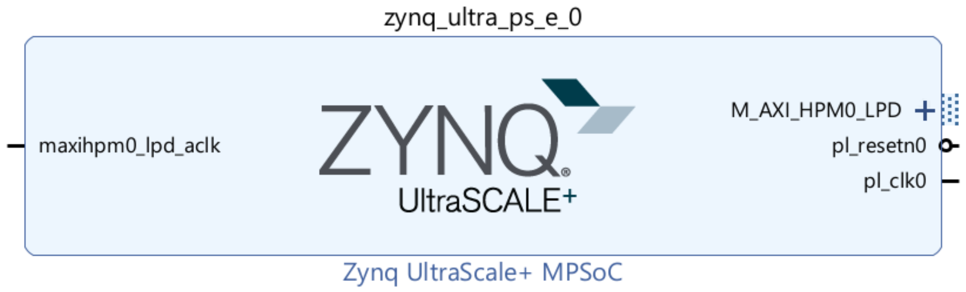

Search: fft (3 matches)

DFE FFT

Fast Fourier Transform

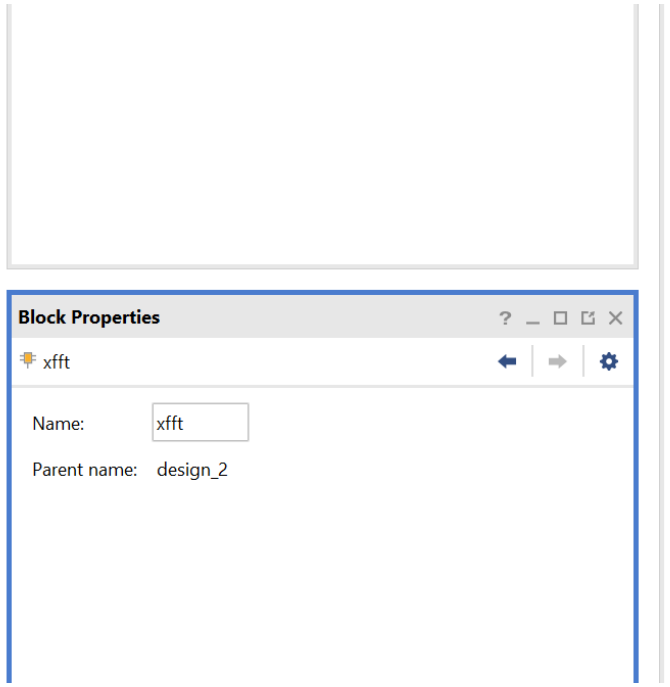
LTE Fast Fourier Transform

ENTER to select, ESC to cancel, Ctrl+Q for IP details

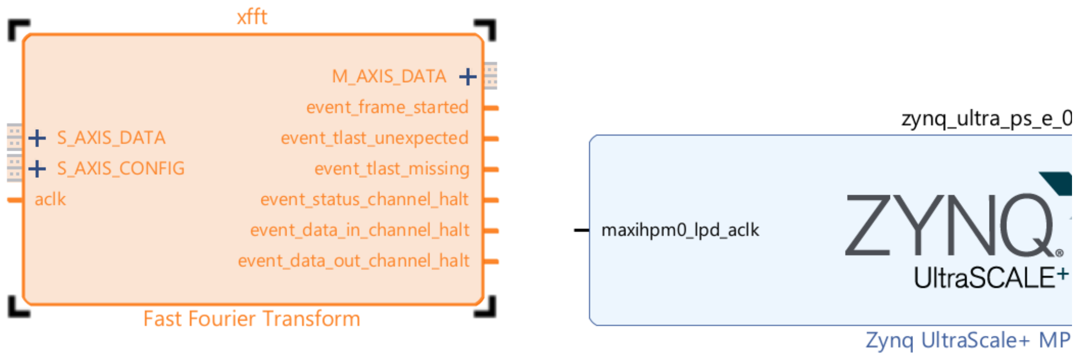


# Lab: FFT Verification

Click on the Fast Fourier Transform and change its name to xfft. Double click on the block to customize it.



Name: xfft
Parent name: design_2

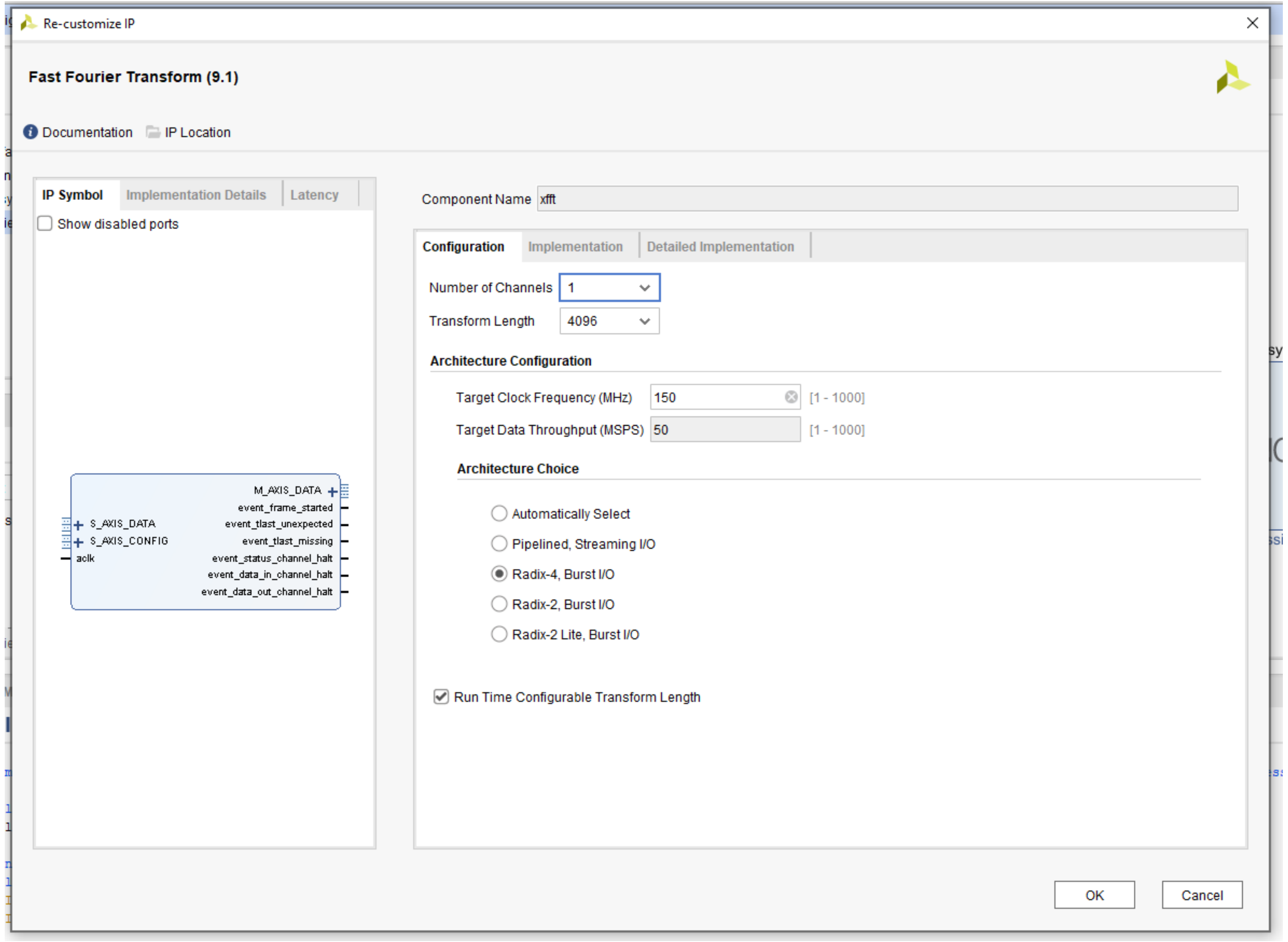


# Lab: FFT Verification

On the configuration tab, select

- Transform length 4096
- Radix-4 Burst I/O
- Target Frequency 150Mhz
- Enable Run Time

  Configurable transform length



Fast Fourier Transform (9.1)

Component Name: xfft

Number of Channels: 1

Transform Length: 4096

Architecture Configuration

Target Clock Frequency (MHz): 150 [1 - 1000]

Target Data Throughput (MSPS): 50 [1 - 1000]

Architecture Choice

- Automatically Select
- Pipelined, Streaming I/O
- [✓ Radix-4, Burst I/O]
- Radix-2, Burst I/O
- Radix-2 Lite, Burst I/O

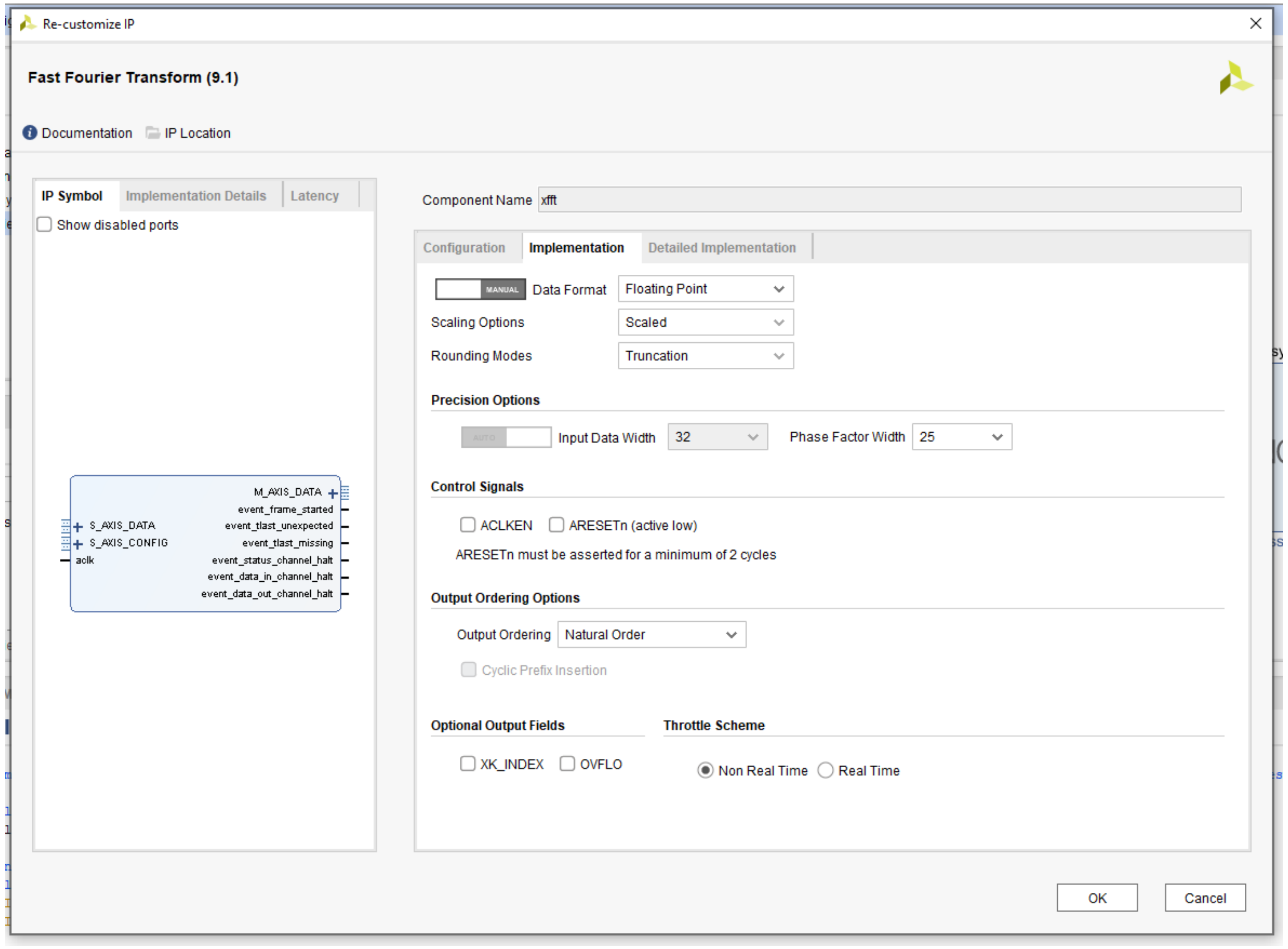
[✓] Run Time Configurable Transform Length

ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

On the implementation tab select

- Floating Point
- Phase Factor Width 25
- Output Ordering Natural
- Non-Real Time Throttle scheme


Re-customize IP

Fast Fourier Transform (9.1)

Documentation IP Location

Component Name xfft

Configuration | Implementation | Detailed Implementation

Data Format: Floating Point
Scaling Options: Scaled
Rounding Modes: Truncation

Precision Options
Input Data Width: 32
Phase Factor Width: 25

Control Signals
ACLKEN ARESETn (active low)
ARESETn must be asserted for a minimum of 2 cycles

Output Ordering Options
Output Ordering: Natural Order
Cyclic Prefix Insertion

Optional Output Fields
XK_INDEX OVFLO

Throttle Scheme
[✓ Non Real Time] Real Time

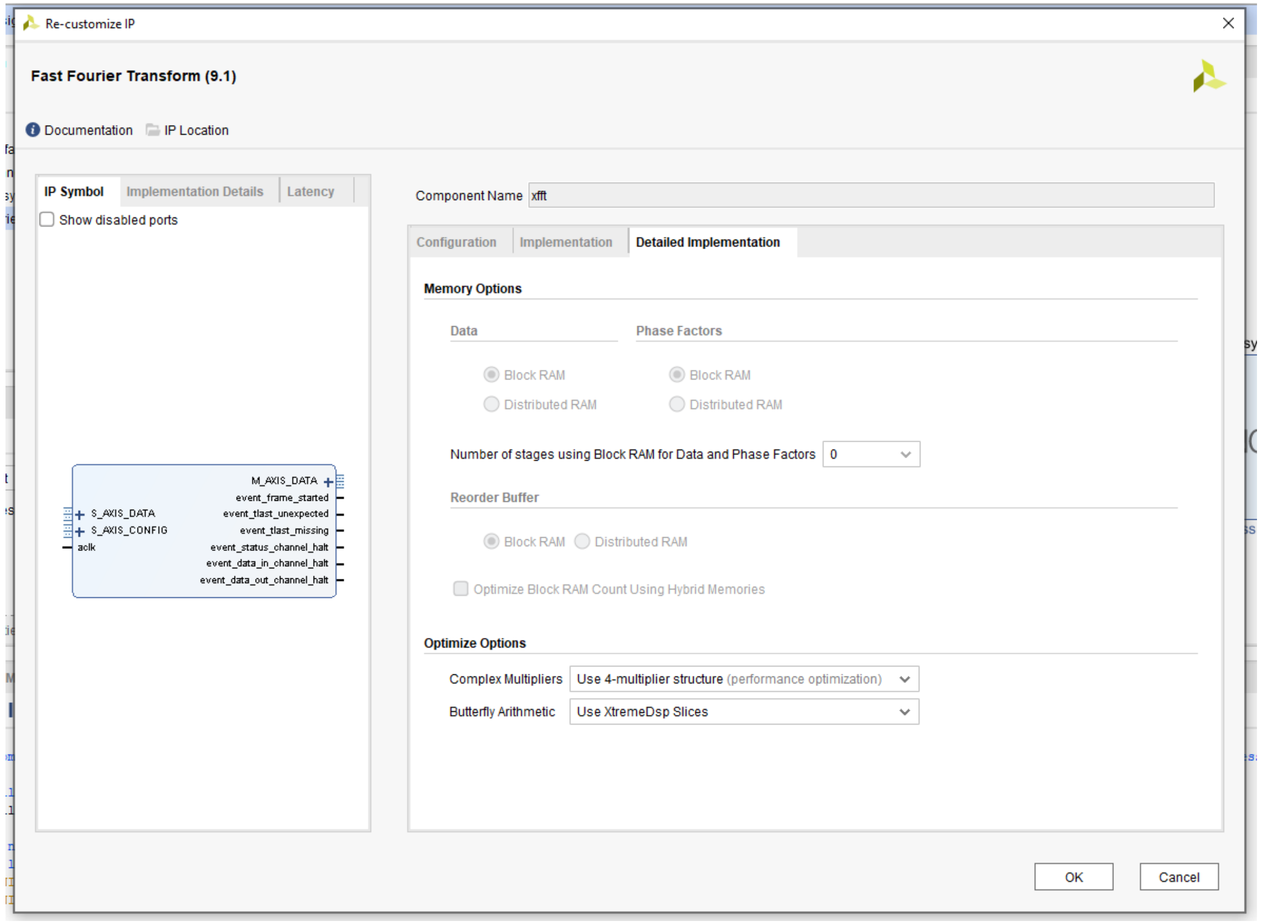
OK Cancel

# Lab: FFT Verification

On the Detailed Implementation tab

select

- Use 4-Multipler Structure
- Use XtremeDSP Slices


Re-customize IP

Fast Fourier Transform (9.1)

Documentation IP Location

IP Symbol | Implementation Details | Latency

Show disabled ports

Component Name xfft

Configuration | Implementation | Detailed Implementation

**Memory Options**

Data: [✓ Block RAM] Distributed RAM

Phase Factors: [✓ Block RAM] Distributed RAM

Number of stages using Block RAM for Data and Phase Factors: 0

Reorder Buffer: [✓ Block RAM] Distributed RAM

Optimize Block RAM Count Using Hybrid Memories

**Optimize Options**

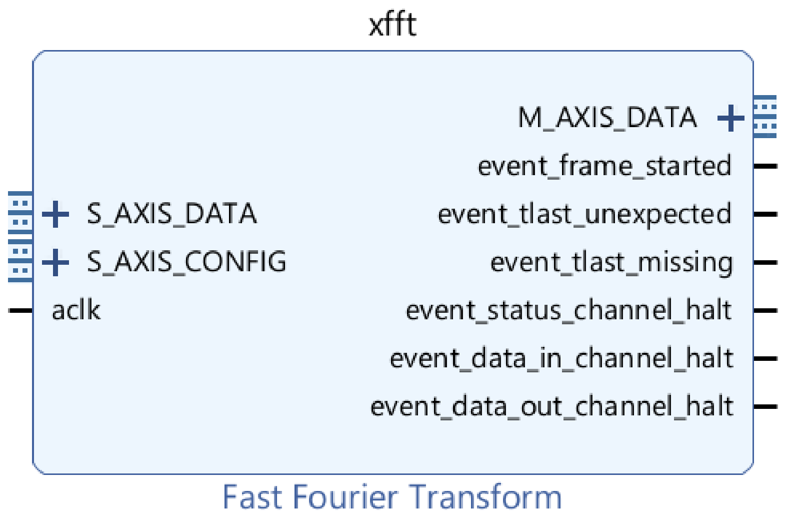
Complex Multipliers: Use 4-multiplier structure (performance optimization)

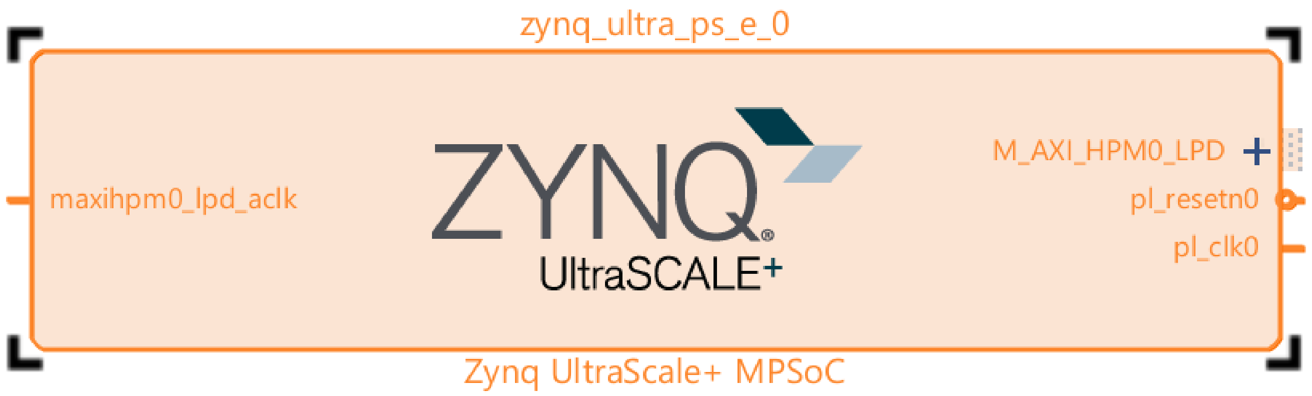
Butterfly Arithmetic: Use XtremeDsp Slices

OK Cancel

# Lab: FFT Verification

Double click on the Processing System to reconfigure it

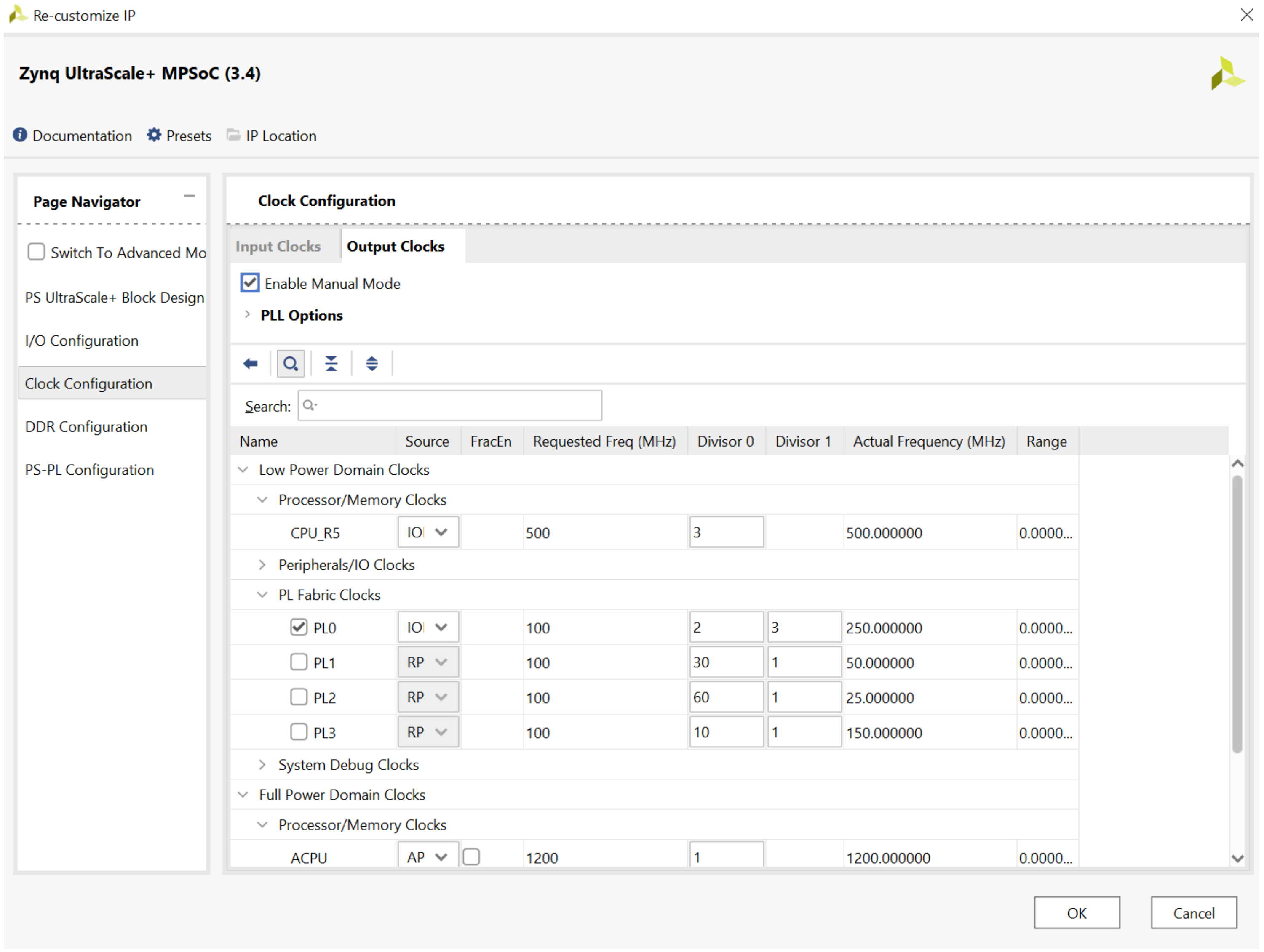




ADIUVO
ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

On the clocking tab change the frequency of clock one to 250MHz



Re-customize IP

**Zynq UltraScale+ MPSoC (3.4)**

Documentation  Presets  IP Location

**Page Navigator**

☐ Switch To Advanced Mo
- PS UltraScale+ Block Design
- I/O Configuration
- Clock Configuration
- DDR Configuration
- PS-PL Configuration

**Clock Configuration**

Input Clocks | **Output Clocks**

☑ Enable Manual Mode

**PLL Options**

Search:

| Name | Source | FracEn | Requested Freq (MHz) | Divisor 0 | Divisor 1 | Actual Frequency (MHz) | Range |
|---|---|---|---|---|---|---|---|
| Low Power Domain Clocks | | | | | | | |
| Processor/Memory Clocks | | | | | | | |
| CPU_R5 | IO | | 500 | 3 | | 500.000000 | 0.0000... |
| Peripherals/IO Clocks | | | | | | | |
| PL Fabric Clocks | | | | | | | |
| ☑ PL0 | IO | | 100 | 2 | 3 | 250.000000 | 0.0000... |
| ☐ PL1 | RP | | 100 | 30 | 1 | 50.000000 | 0.0000... |
| ☐ PL2 | RP | | 100 | 60 | 1 | 25.000000 | 0.0000... |
| ☐ PL3 | RP | | 100 | 10 | 1 | 150.000000 | 0.0000... |
| System Debug Clocks | | | | | | | |
| Full Power Domain Clocks | | | | | | | |
| Processor/Memory Clocks | | | | | | | |
| ACPU | AP | ☐ | 1200 | 1 | | 1200.000000 | 0.0000... |

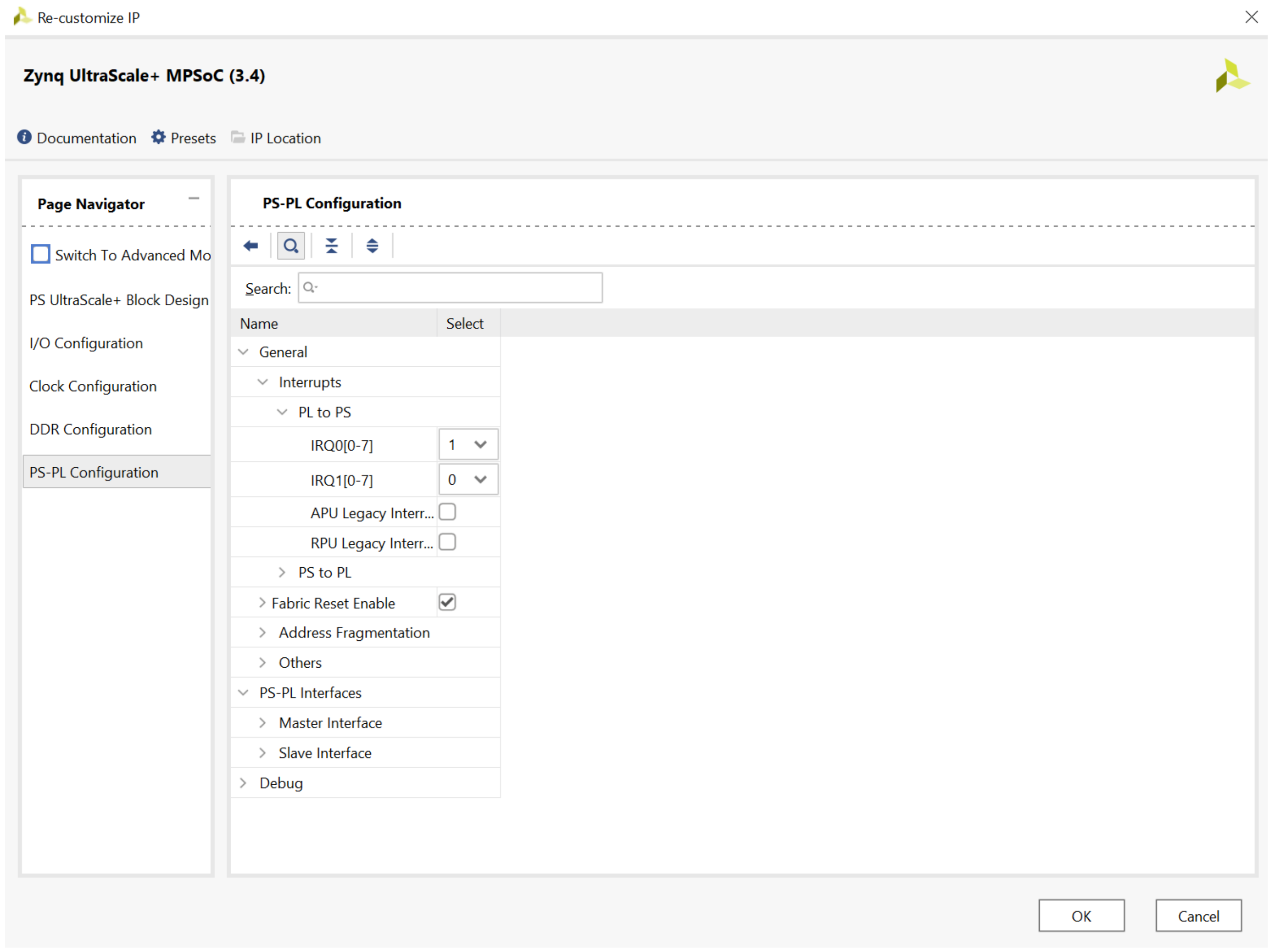
OK  Cancel

ADIUVO
ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

On the Interrupts Tab enable the

IRQ[0:7]


Re-customize IP

Zynq UltraScale+ MPSoC (3.4)

Documentation Presets IP Location

Page Navigator
- Switch To Advanced Mo
- PS UltraScale+ Block Design
- I/O Configuration
- Clock Configuration
- DDR Configuration
- PS-PL Configuration

PS-PL Configuration

Search:

| Name | Select |
|---|---|
| General | |
| Interrupts | |
| PL to PS | |
| IRQ0[0-7] | 1 |
| IRQ1[0-7] | 0 |
| APU Legacy Interr... | ☐ |
| RPU Legacy Interr... | ☐ |
| PS to PL | |
| Fabric Reset Enable | ✓ |
| Address Fragmentation | |
| Others | |
| PS-PL Interfaces | |
| Master Interface | |
| Slave Interface | |
| Debug | |

OK Cancel

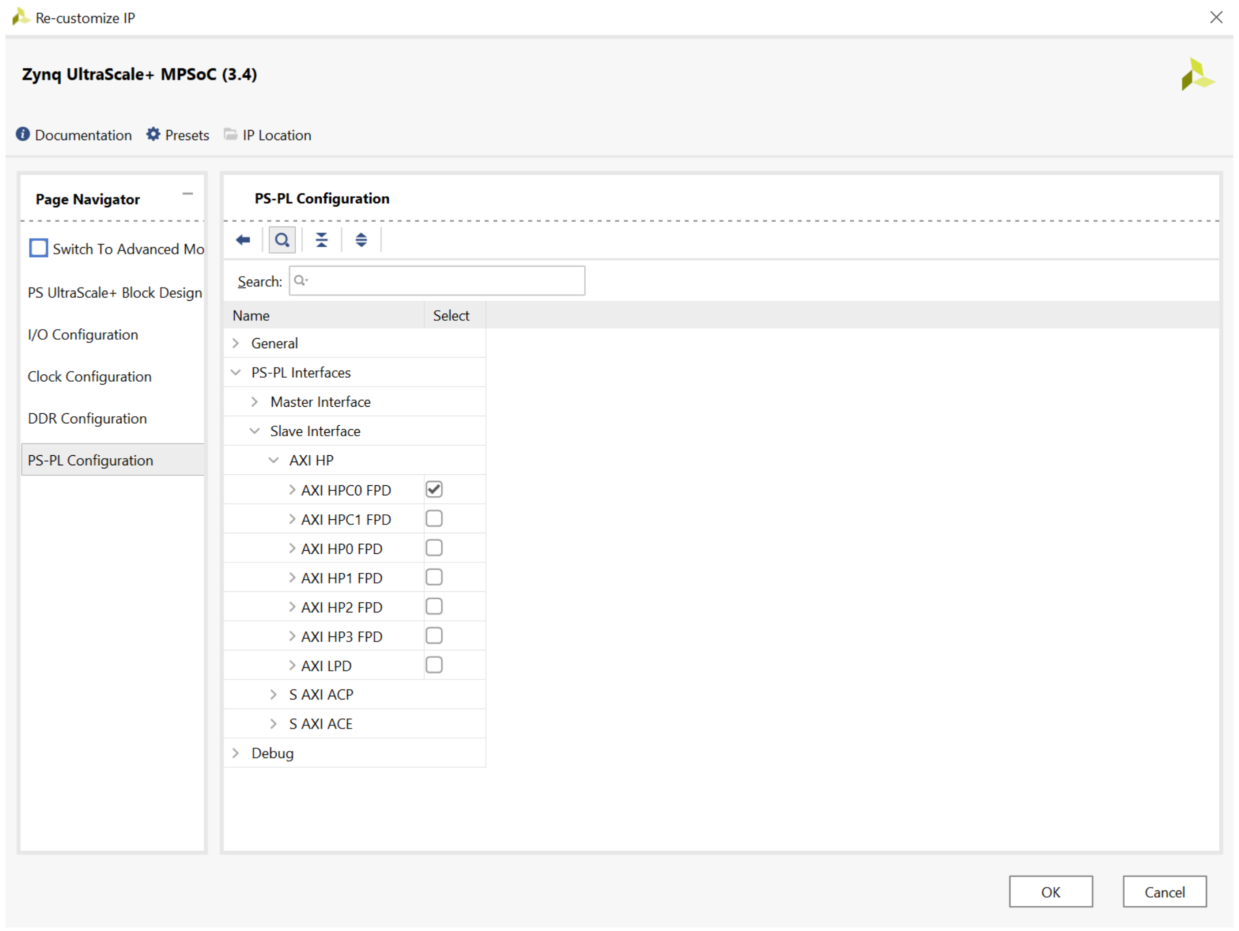
ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

On the PS/PL interface select the HP

Slave AXI Interface

Enable AXI HPC0 FPD Interface



Re-customize IP

Zynq UltraScale+ MPSoC (3.4)

Documentation Presets IP Location

Page Navigator

- [ ] Switch To Advanced Mo
- PS UltraScale+ Block Design
- I/O Configuration
- Clock Configuration
- DDR Configuration
- PS-PL Configuration

PS-PL Configuration

Search:

| Name | Select |
|---|---|
| General | |
| PS-PL Interfaces | |
| Master Interface | |
| Slave Interface | |
| AXI HP | |
| AXI HPC0 FPD | ☑ |
| AXI HPC1 FPD | ☐ |
| AXI HP0 FPD | ☐ |
| AXI HP1 FPD | ☐ |
| AXI HP2 FPD | ☐ |
| AXI HP3 FPD | ☐ |
| AXI LPD | ☐ |
| S AXI ACP | |
| S AXI ACE | |
| Debug | |

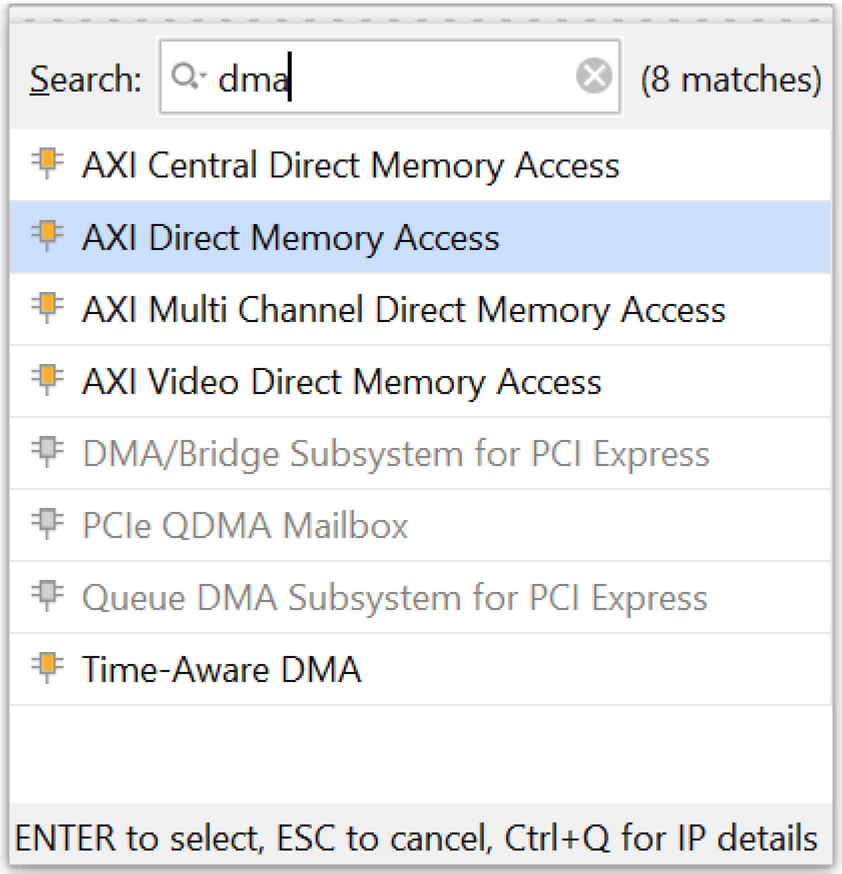
OK Cancel

ADIUVO
ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Click + and select AXI

Direct Memory Access



Search: dma (8 matches)

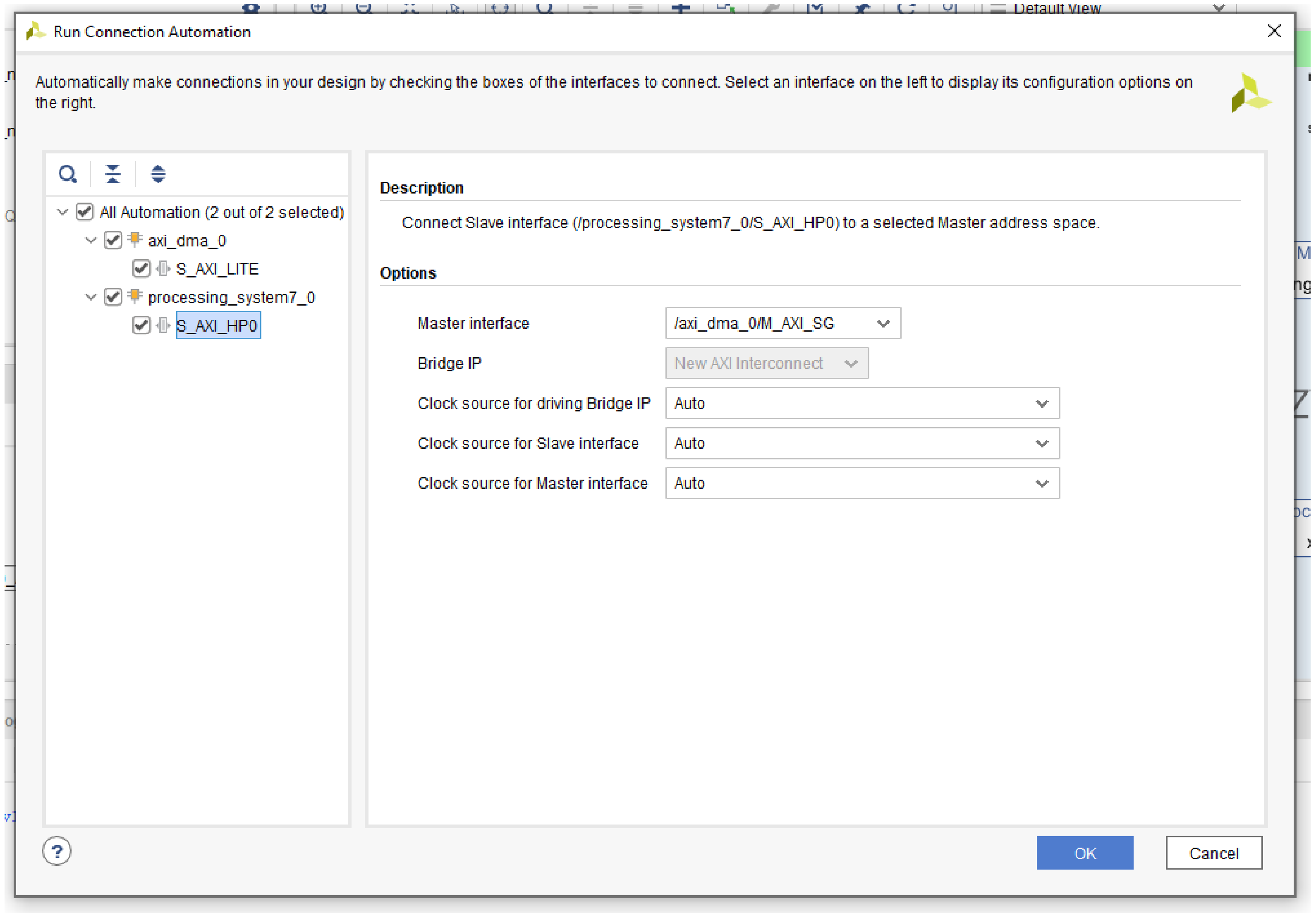
AXI Central Direct Memory Access
AXI Direct Memory Access
AXI Multi Channel Direct Memory Access
AXI Video Direct Memory Access
DMA/Bridge Subsystem for PCI Express
PCIe QDMA Mailbox
Queue DMA Subsystem for PCI Express
Time-Aware DMA

ENTER to select, ESC to cancel, Ctrl+Q for IP details

# Lab: FFT Verification

Run the connection automation.

Leave the defaults as standard and click OK.


Run Connection Automation

Automatically make connections in your design by checking the boxes of the interfaces to connect. Select an interface on the left to display its configuration options on the right.

- ☑ All Automation (2 out of 2 selected)
  - ☑ axi_dma_0
    - ☑ S_AXI_LITE
  - ☑ processing_system7_0
    - ☑ S_AXI_HP0

**Description**

Connect Slave interface (/processing_system7_0/S_AXI_HP0) to a selected Master address space.

**Options**

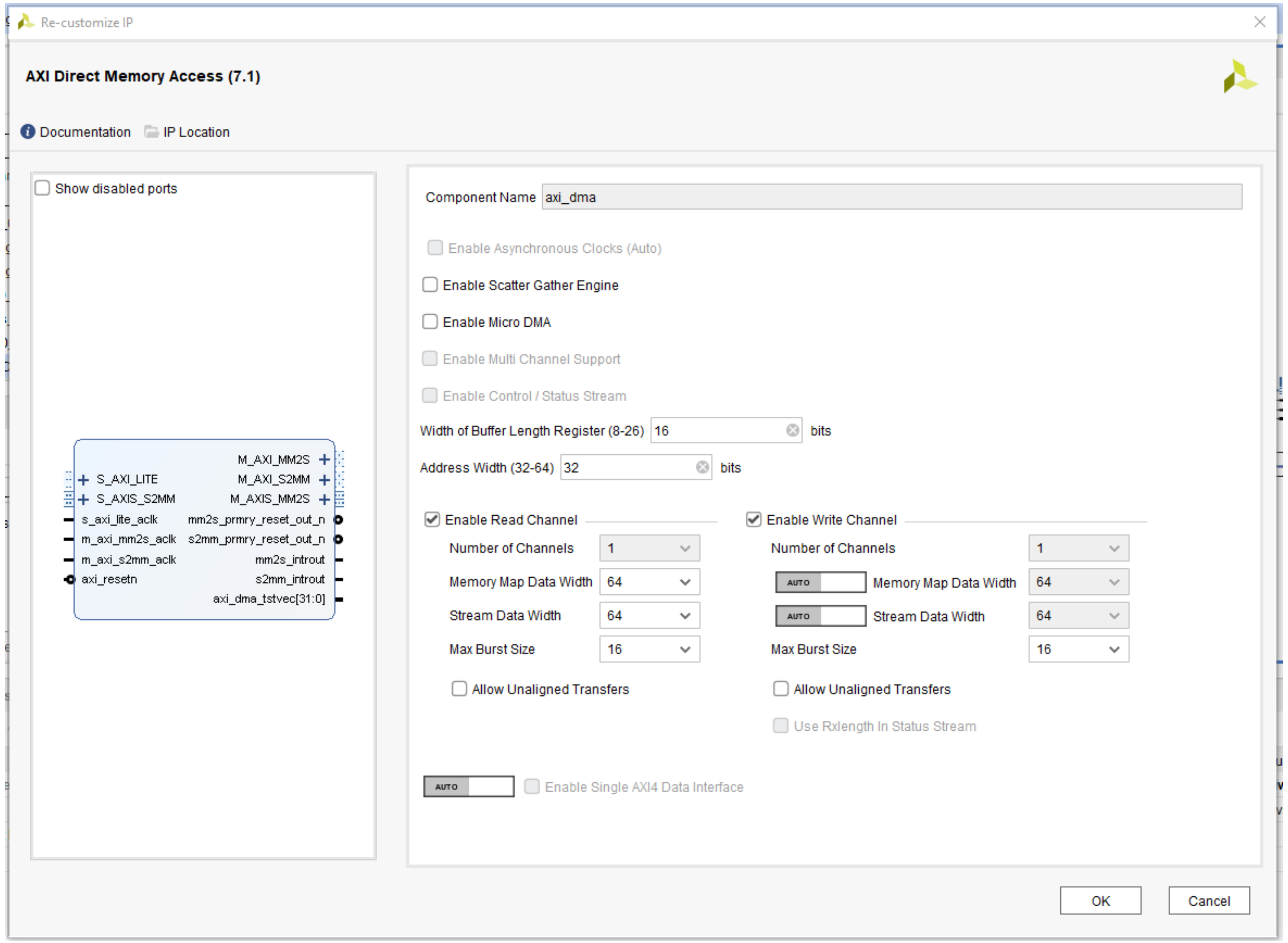
| | |
|---|---|
| Master interface | /axi_dma_0/M_AXI_SG |
| Bridge IP | New AXI Interconnect |
| Clock source for driving Bridge IP | Auto |
| Clock source for Slave interface | Auto |
| Clock source for Master interface | Auto |

# Lab: FFT Verification

Select the DMA, double click on it

and configure it

- Width of Buffer length 16
- Stream data width 64
- Max burst size 16



AXI Direct Memory Access (7.1)

☐ Show disabled ports

Component Name: axi_dma

☐ Enable Asynchronous Clocks (Auto)
☐ Enable Scatter Gather Engine
☐ Enable Micro DMA
☐ Enable Multi Channel Support
☐ Enable Control / Status Stream

Width of Buffer Length Register (8-26): 16 bits

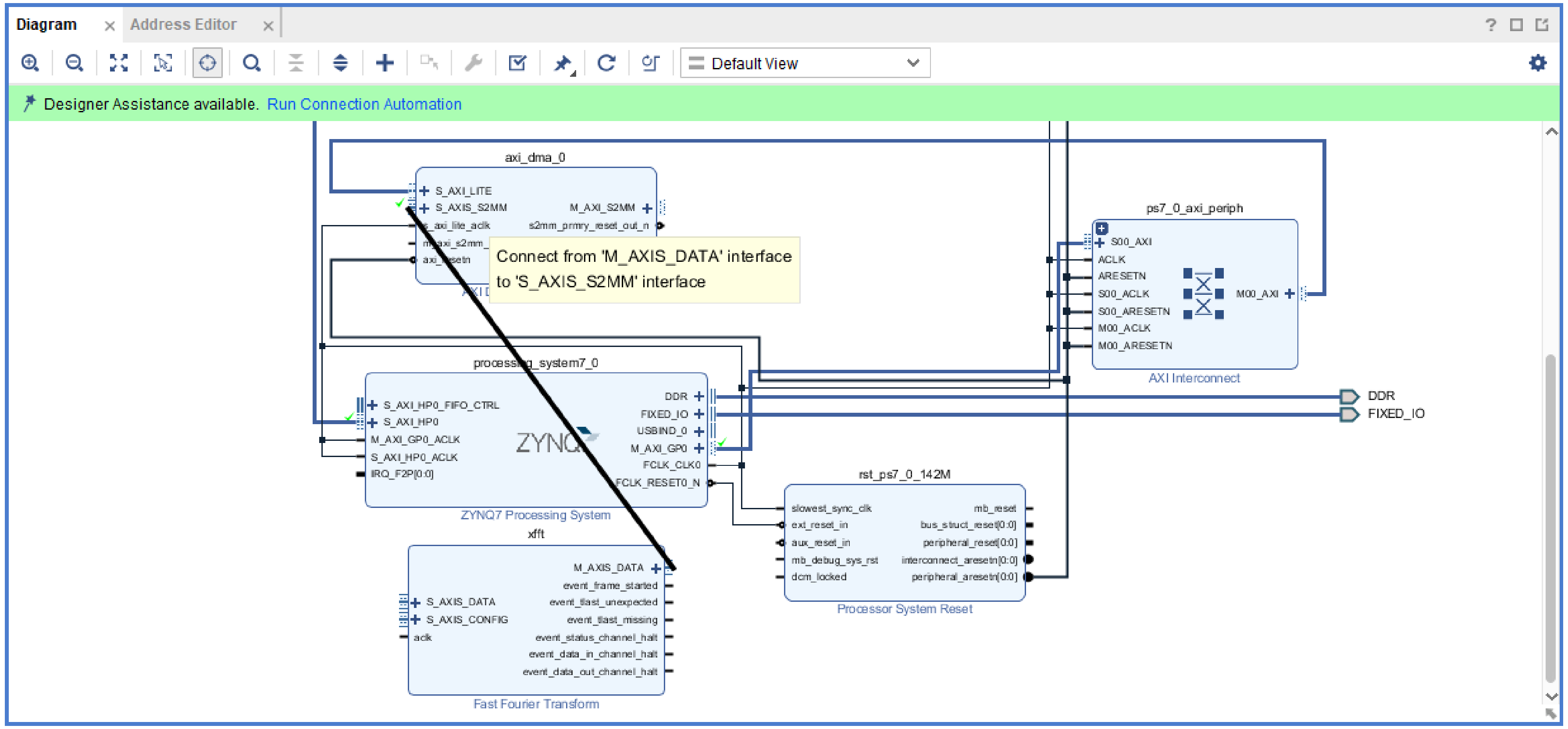
Address Width (32-64): 32 bits

| ✓ Enable Read Channel | | ✓ Enable Write Channel | |
|---|---|---|---|
| Number of Channels | 1 | Number of Channels | 1 |
| Memory Map Data Width | 64 | AUTO Memory Map Data Width | 64 |
| Stream Data Width | 64 | AUTO Stream Data Width | 64 |
| Max Burst Size | 16 | Max Burst Size | 16 |
| ☐ Allow Unaligned Transfers | | ☐ Allow Unaligned Transfers | |
| | | ☐ Use Rxlength In Status Stream | |

AUTO ☐ Enable Single AXI4 Data Interface

# Lab: FFT Verification

Connect the xFFT M AXIS data to the DMA, S AXIS S2MM port


Designer Assistance available. Run Connection Automation

Connect from 'M_AXIS_DATA' interface to 'S_AXIS_S2MM' interface

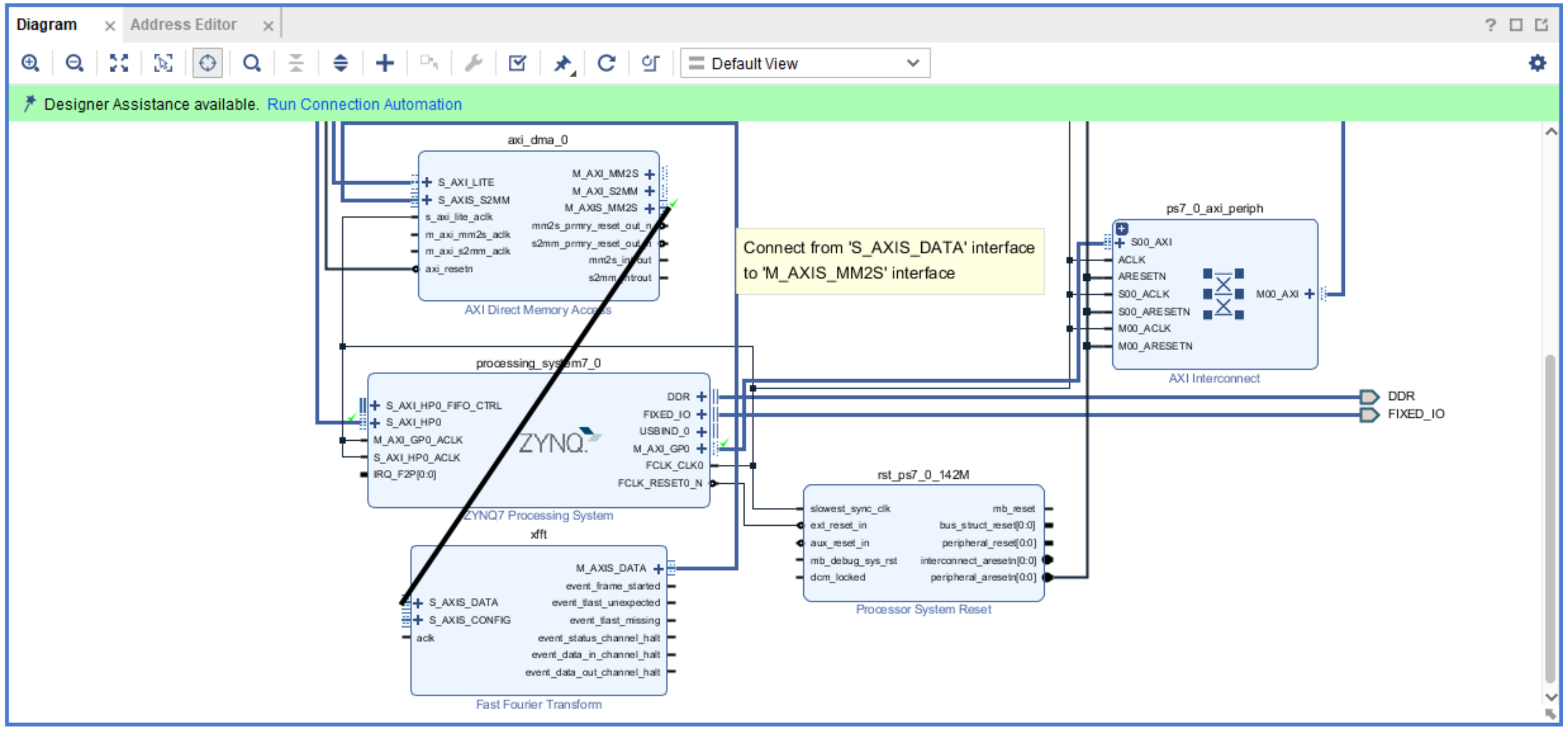
[AMD Official Use Only - General]

# Lab: FFT Verification

Connect the DMA M

AXIS MM2S to the xFFT

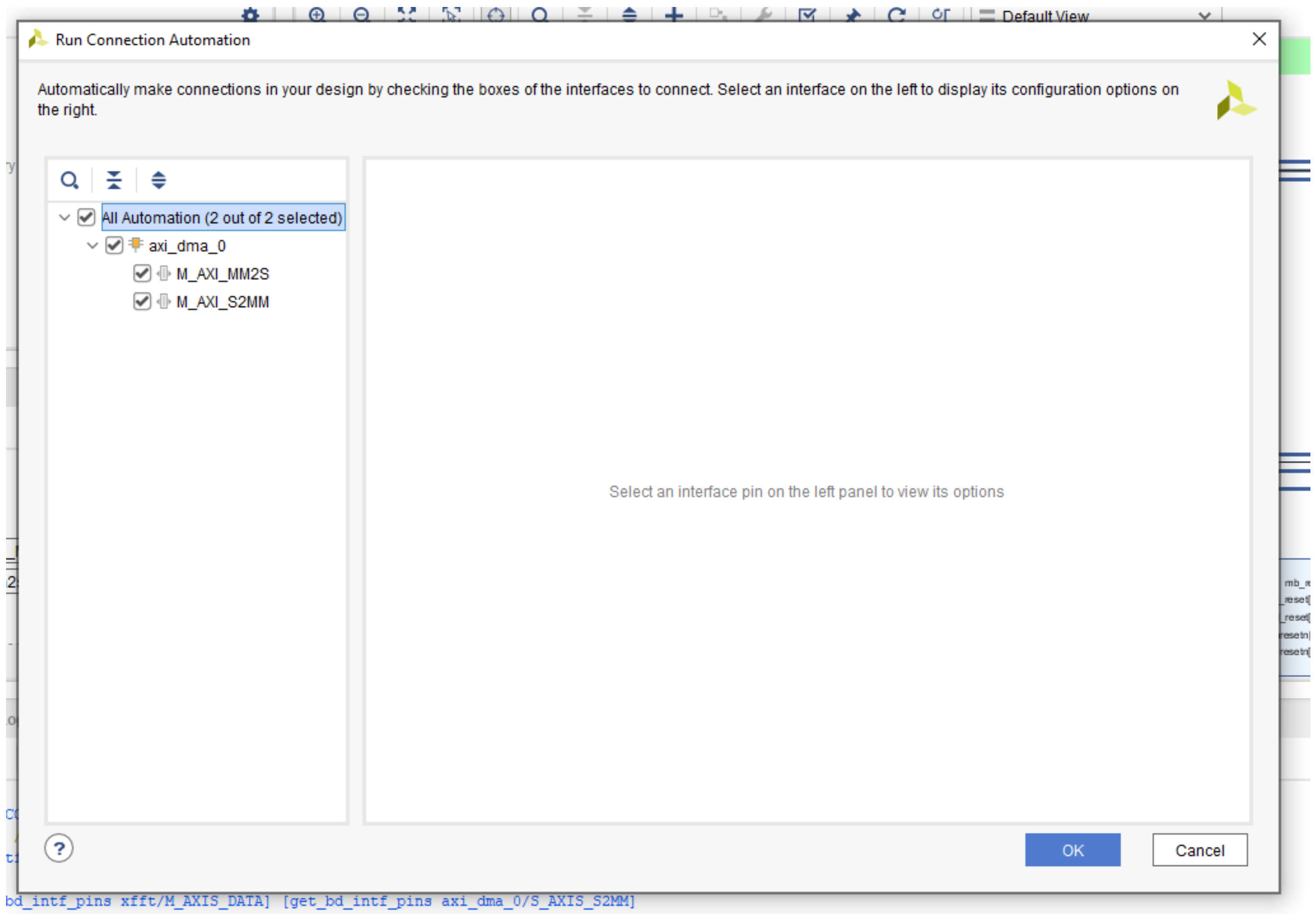
S AXIS Data



ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Run the connection automation


Run Connection Automation

Automatically make connections in your design by checking the boxes of the interfaces to connect. Select an interface on the left to display its configuration options on the right.

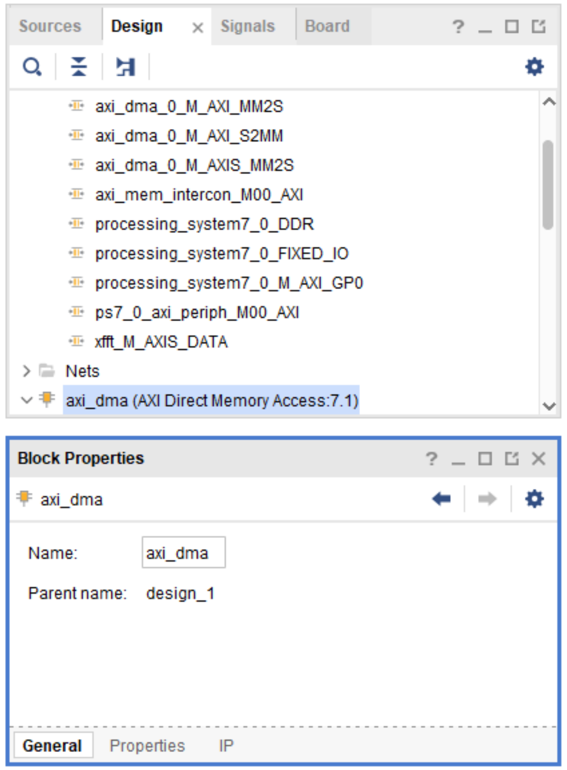
- [✓] All Automation (2 out of 2 selected)
  - [✓] axi_dma_0
    - [✓] M_AXI_MM2S
    - [✓] M_AXI_S2MM

Select an interface pin on the left panel to view its options

```
bd_intf_pins xfft/M_AXIS_DATA] [get_bd_intf_pins axi_dma_0/S_AXIS_S2MM]
```

# Lab: FFT Verification

The diagram should look like below



- axi_dma_0_M_AXI_MM2S
- axi_dma_0_M_AXI_S2MM
- axi_dma_0_M_AXIS_MM2S
- axi_mem_intercon_M00_AXI
- processing_system7_0_DDR
- processing_system7_0_FIXED_IO
- processing_system7_0_M_AXI_GP0
- ps7_0_axi_periph_M00_AXI
- xfft_M_AXIS_DATA

Nets

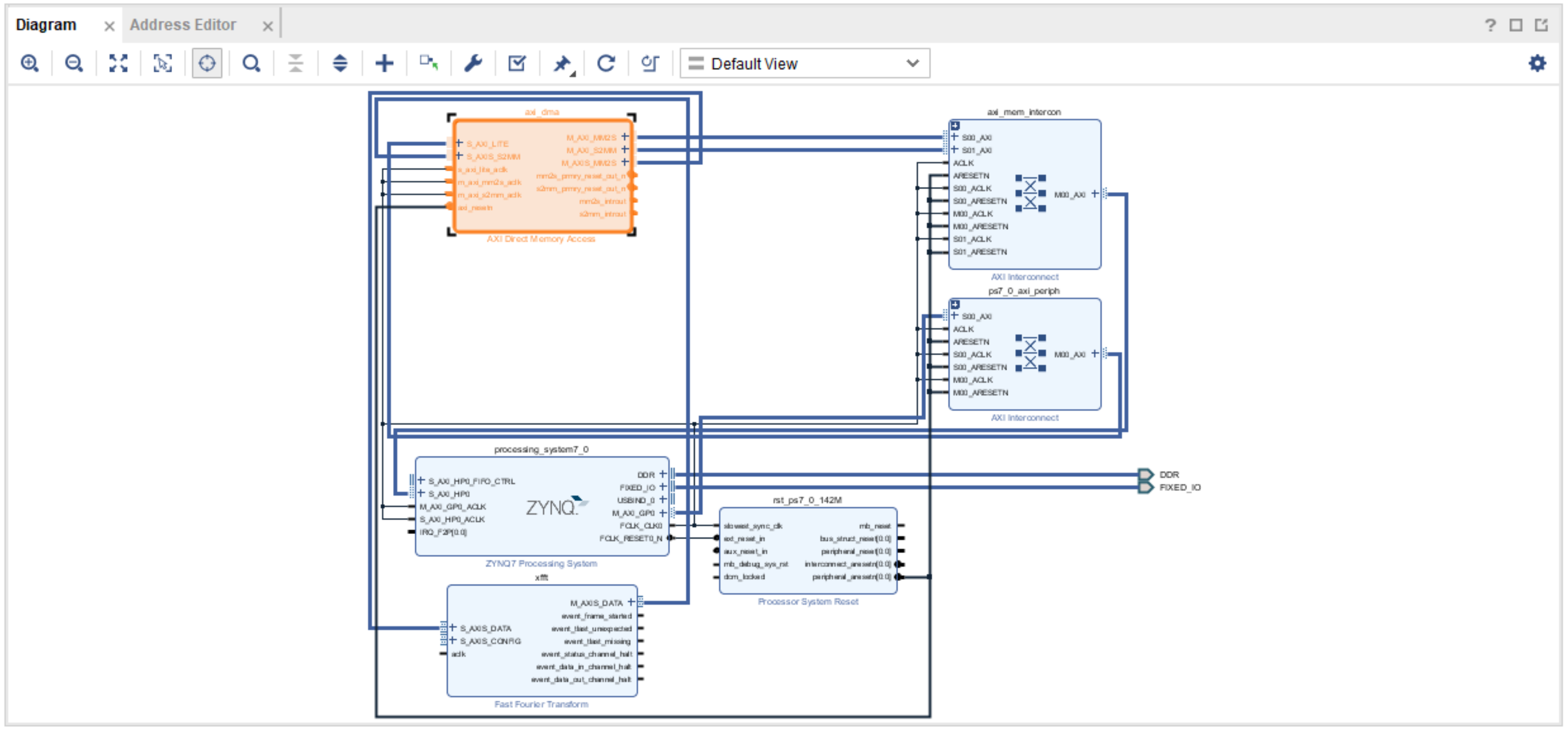
axi_dma (AXI Direct Memory Access:7.1)

**Block Properties**

axi_dma

Name: axi_dma

Parent name: design_1

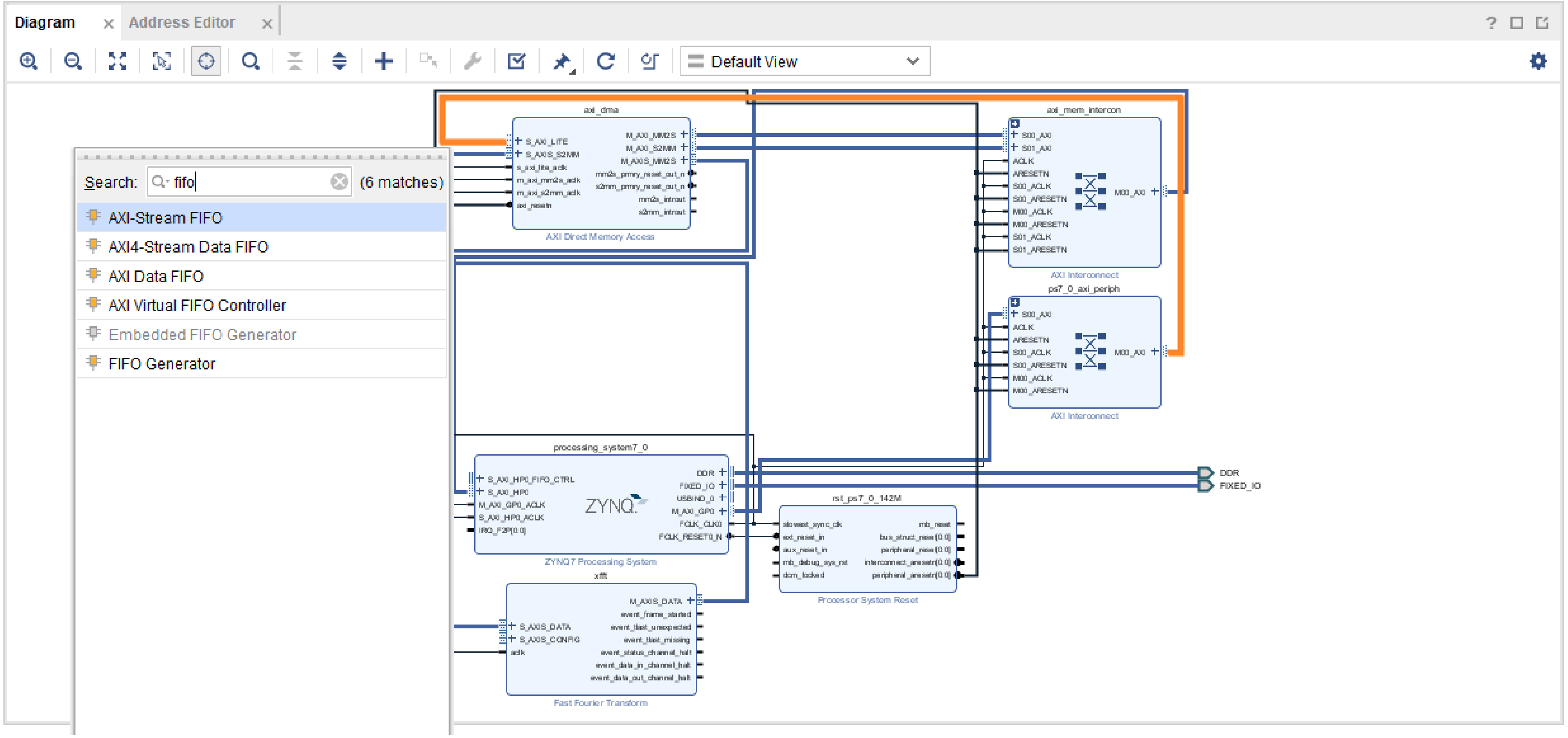


# Lab: FFT Verification

Click on + and add

in an AXI-Stream

FIFO

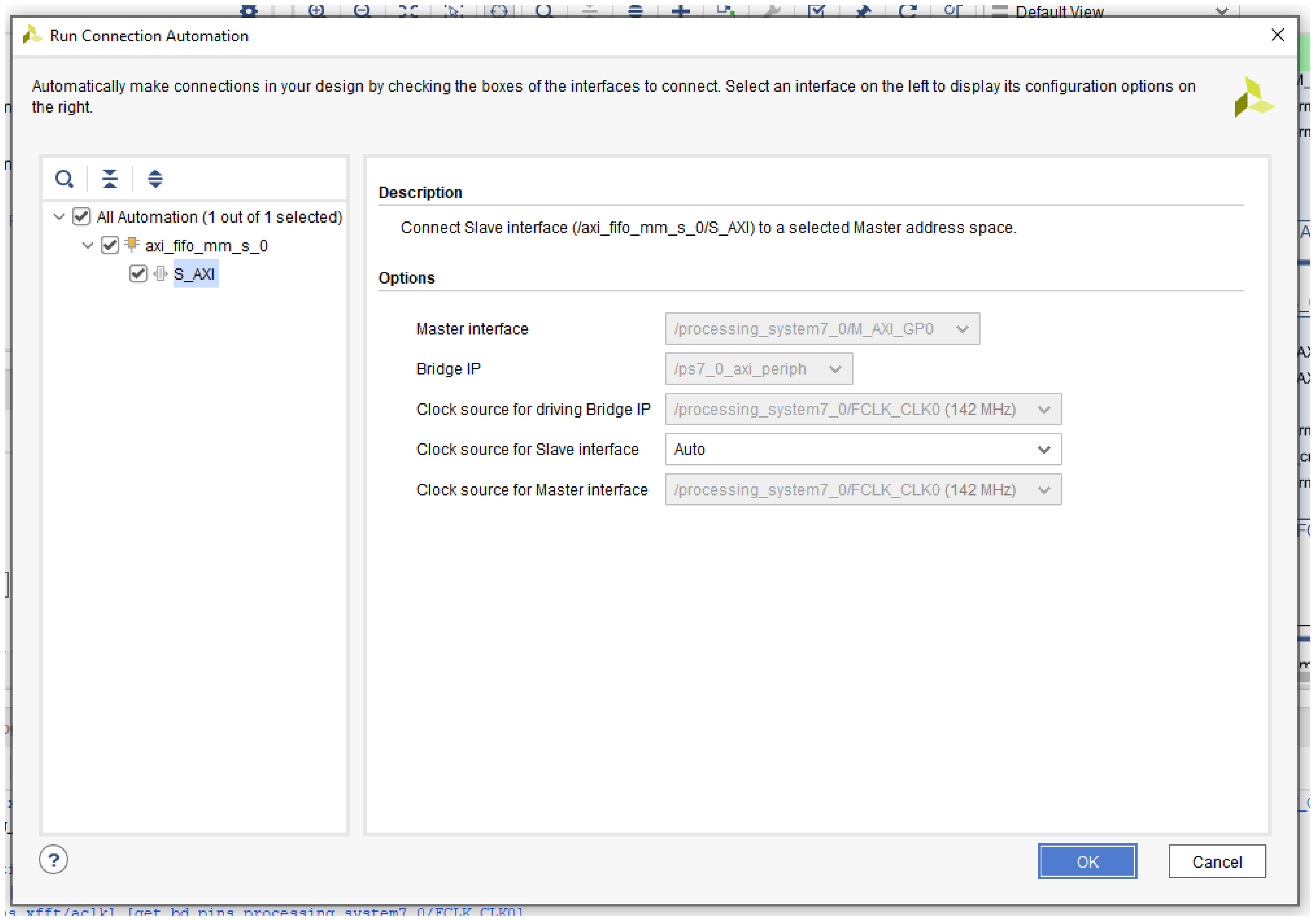


ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Run the connection

automation and click on OK



Run Connection Automation

Automatically make connections in your design by checking the boxes of the interfaces to connect. Select an interface on the left to display its configuration options on the right.

- ☑ All Automation (1 out of 1 selected)
  - ☑ axi_fifo_mm_s_0
    - ☑ S_AXI

**Description**

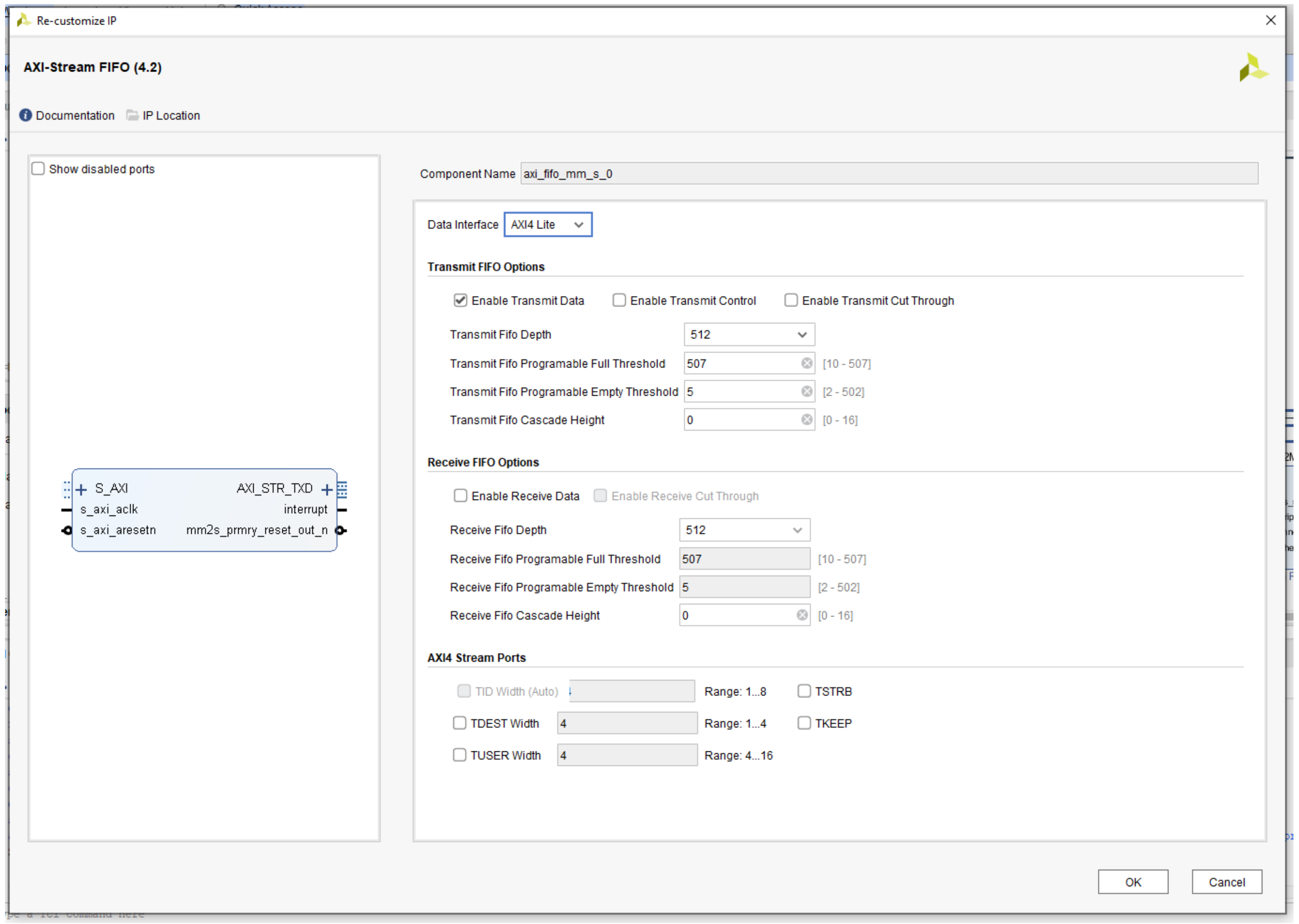
Connect Slave interface (/axi_fifo_mm_s_0/S_AXI) to a selected Master address space.

**Options**

| | |
|---|---|
| Master interface | /processing_system7_0/M_AXI_GP0 |
| Bridge IP | /ps7_0_axi_periph |
| Clock source for driving Bridge IP | /processing_system7_0/FCLK_CLK0 (142 MHz) |
| Clock source for Slave interface | Auto |
| Clock source for Master interface | /processing_system7_0/FCLK_CLK0 (142 MHz) |

# Lab: FFT Verification

Double click on the AXI Stream FIFO and configure it to have an AXI Lite Interface and only enable the Transmit Data Interface leave all else unchanged.



Re-customize IP

**AXI-Stream FIFO (4.2)**

Documentation   IP Location

☐ Show disabled ports

S_AXI  AXI_STR_TXD
s_axi_aclk  interrupt
s_axi_aresetn  mm2s_prmry_reset_out_n

Component Name: axi_fifo_mm_s_0

Data Interface: AXI4 Lite

**Transmit FIFO Options**

☑ Enable Transmit Data  ☐ Enable Transmit Control  ☐ Enable Transmit Cut Through

| | | |
|---|---|---|
| Transmit Fifo Depth | 512 | |
| Transmit Fifo Programable Full Threshold | 507 | [10 - 507] |
| Transmit Fifo Programable Empty Threshold | 5 | [2 - 502] |
| Transmit Fifo Cascade Height | 0 | [0 - 16] |

**Receive FIFO Options**

☐ Enable Receive Data  ☐ Enable Receive Cut Through

| | | |
|---|---|---|
| Receive Fifo Depth | 512 | |
| Receive Fifo Programable Full Threshold | 507 | [10 - 507] |
| Receive Fifo Programable Empty Threshold | 5 | [2 - 502] |
| Receive Fifo Cascade Height | 0 | [0 - 16] |

**AXI4 Stream Ports**

| | | | |
|---|---|---|---|
| ☐ TID Width (Auto) | | Range: 1...8 | ☐ TSTRB |
| ☐ TDEST Width | 4 | Range: 1...4 | ☐ TKEEP |
| ☐ TUSER Width | 4 | Range: 4...16 | |

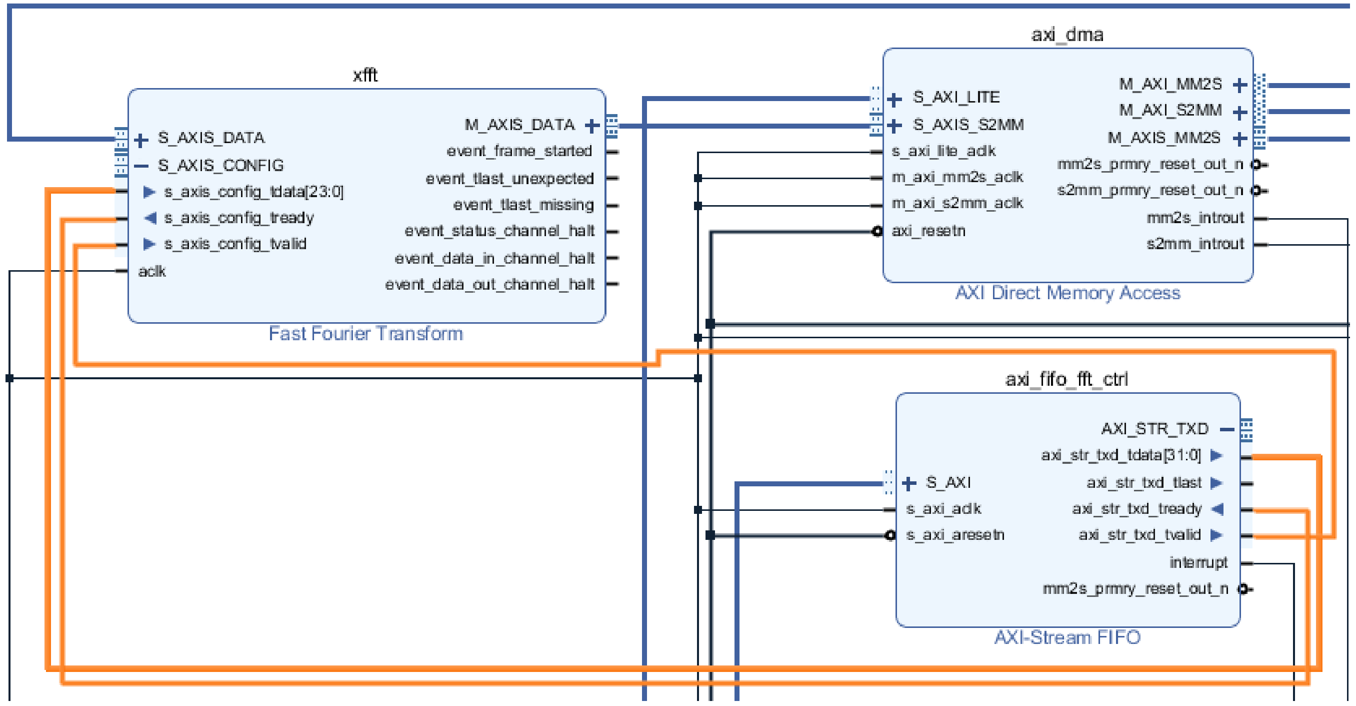
OK  Cancel

# Lab: FFT Verification

Connect the AXI STR TXD tdata,

tlast and tvalid signals to the

xFFT S AXIS config



# Lab: FFT Verification

Click on + and add in a AXI

Interrupt Controller

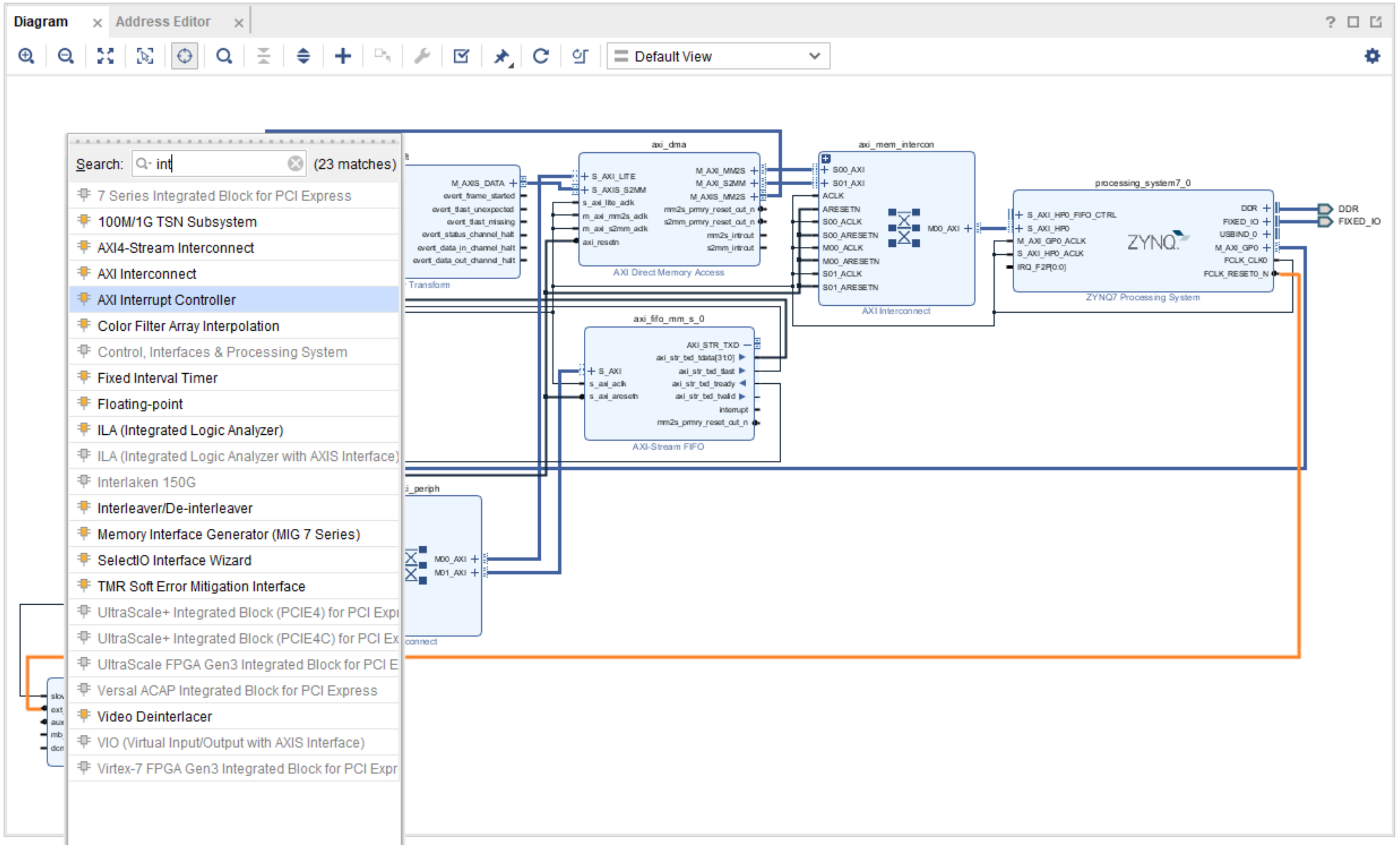
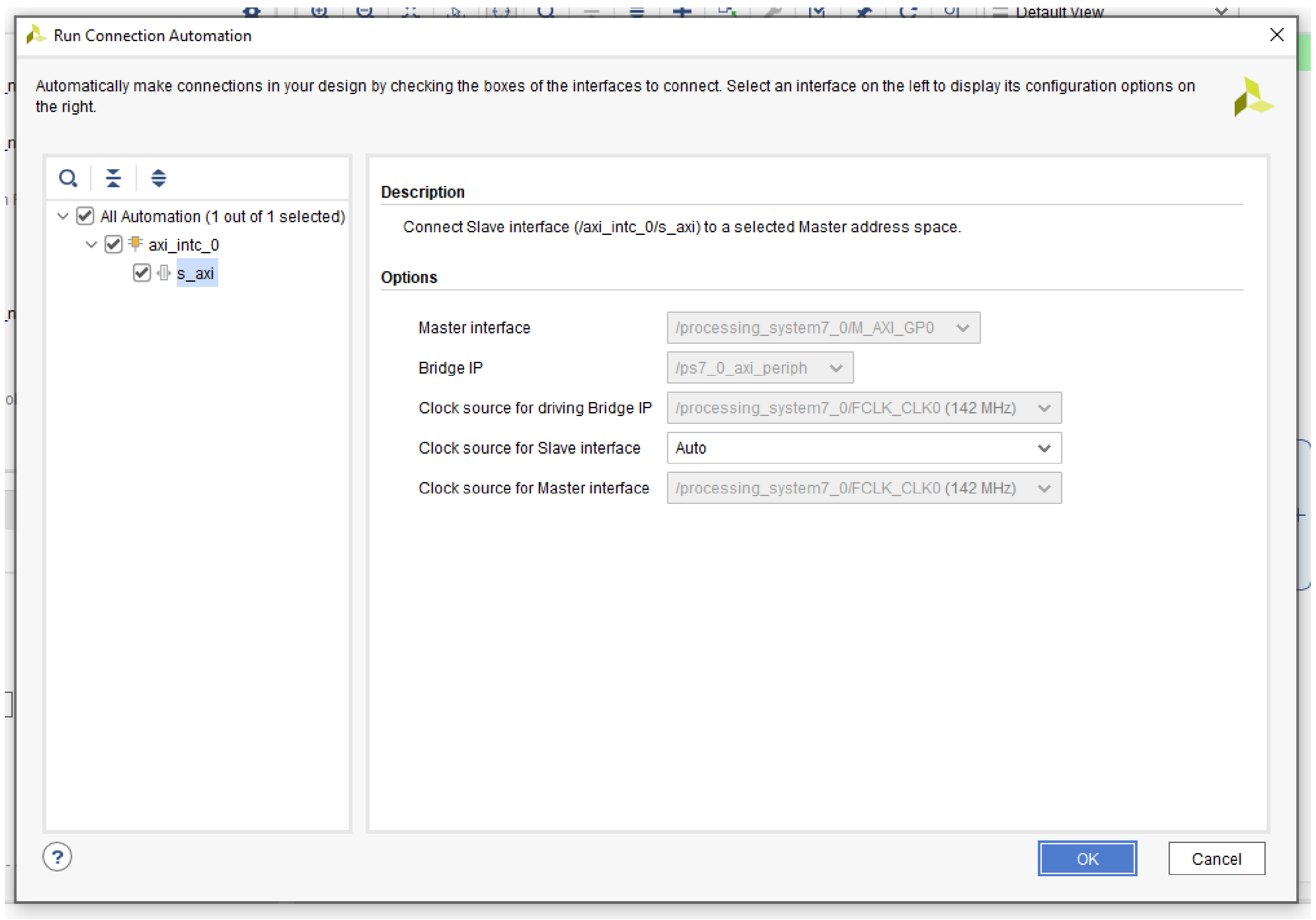


Diagram
Address Editor
Default View
Search: int
(23 matches)
7 Series Integrated Block for PCI Express
100M/1G TSN Subsystem
AXI4-Stream Interconnect
AXI Interconnect
AXI Interrupt Controller
Color Filter Array Interpolation
Control, Interfaces & Processing System
Fixed Interval Timer
Floating-point
ILA (Integrated Logic Analyzer)
ILA (Integrated Logic Analyzer with AXIS Interface)
Interlaken 150G
Interleaver/De-interleaver
Memory Interface Generator (MIG 7 Series)
SelectIO Interface Wizard
TMR Soft Error Mitigation Interface
UltraScale+ Integrated Block (PCIE4) for PCI Expr
UltraScale+ Integrated Block (PCIE4C) for PCI Ex
UltraScale FPGA Gen3 Integrated Block for PCI E
Versal ACAP Integrated Block for PCI Express
Video Deinterlacer
VIO (Virtual Input/Output with AXIS Interface)
Virtex-7 FPGA Gen3 Integrated Block for PCI Expr
axi_dma
AXI Direct Memory Access
axi_mem_intercon
AXI Interconnect
processing_system7_0
ZYNQ7 Processing System
axi_fifo_mm_s_0
AXI-Stream FIFO
DDR
FIXED_IO

# Lab: FFT Verification

Run the connection automation



Run Connection Automation

Automatically make connections in your design by checking the boxes of the interfaces to connect. Select an interface on the left to display its configuration options on the right.

- ☑ All Automation (1 out of 1 selected)
  - ☑ axi_intc_0
    - ☑ s_axi

**Description**

Connect Slave interface (/axi_intc_0/s_axi) to a selected Master address space.

**Options**

| | |
|---|---|
| Master interface | /processing_system7_0/M_AXI_GP0 |
| Bridge IP | /ps7_0_axi_periph |
| Clock source for driving Bridge IP | /processing_system7_0/FCLK_CLK0 (142 MHz) |
| Clock source for Slave interface | Auto |
| Clock source for Master interface | /processing_system7_0/FCLK_CLK0 (142 MHz) |

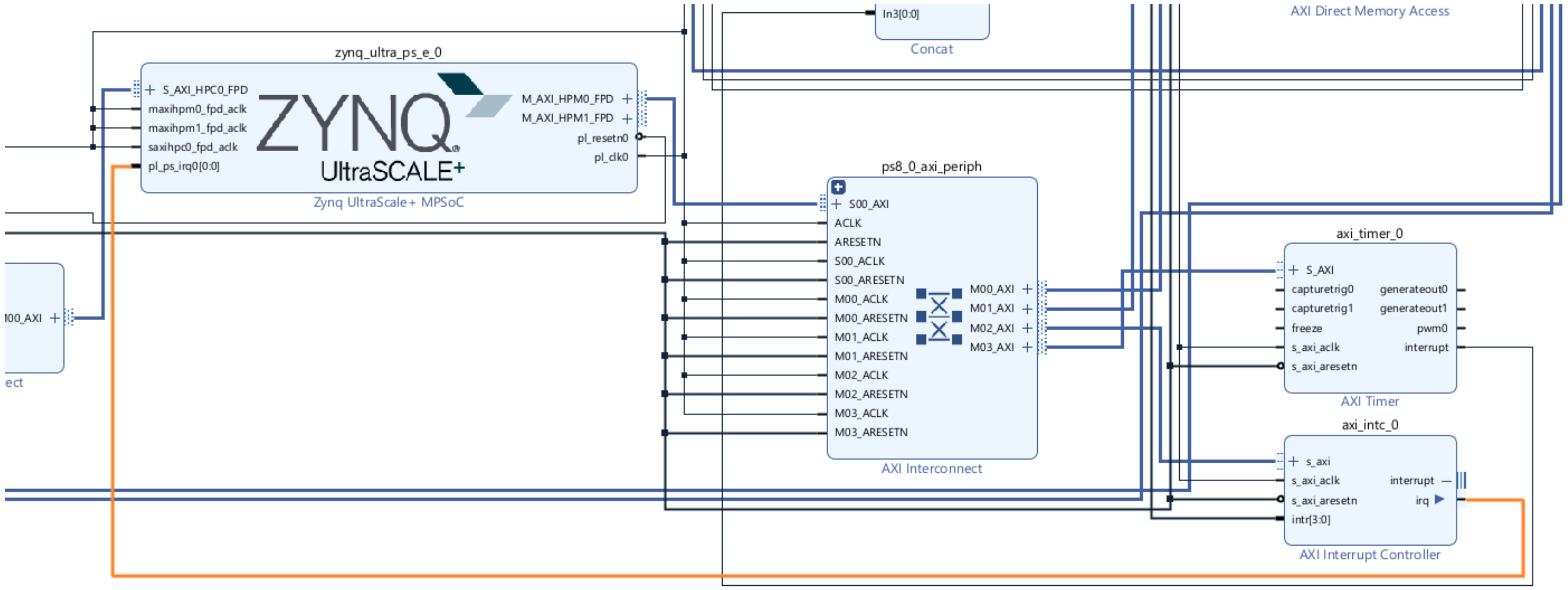
# Lab: FFT Verification

Connect the IRQ output

from the AXI Interrupt

Controller to the PL PS

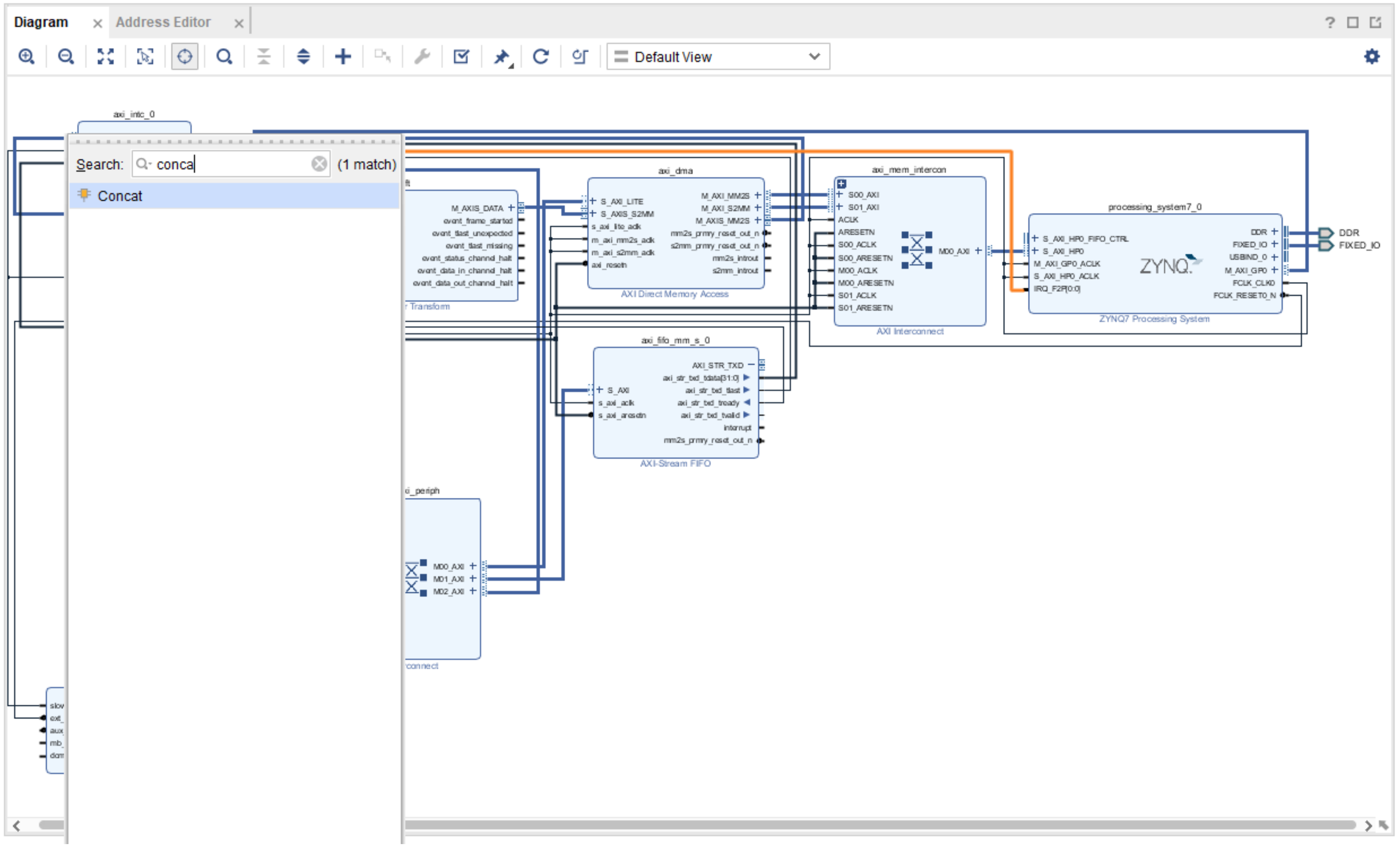
IRQ port on the Zynq



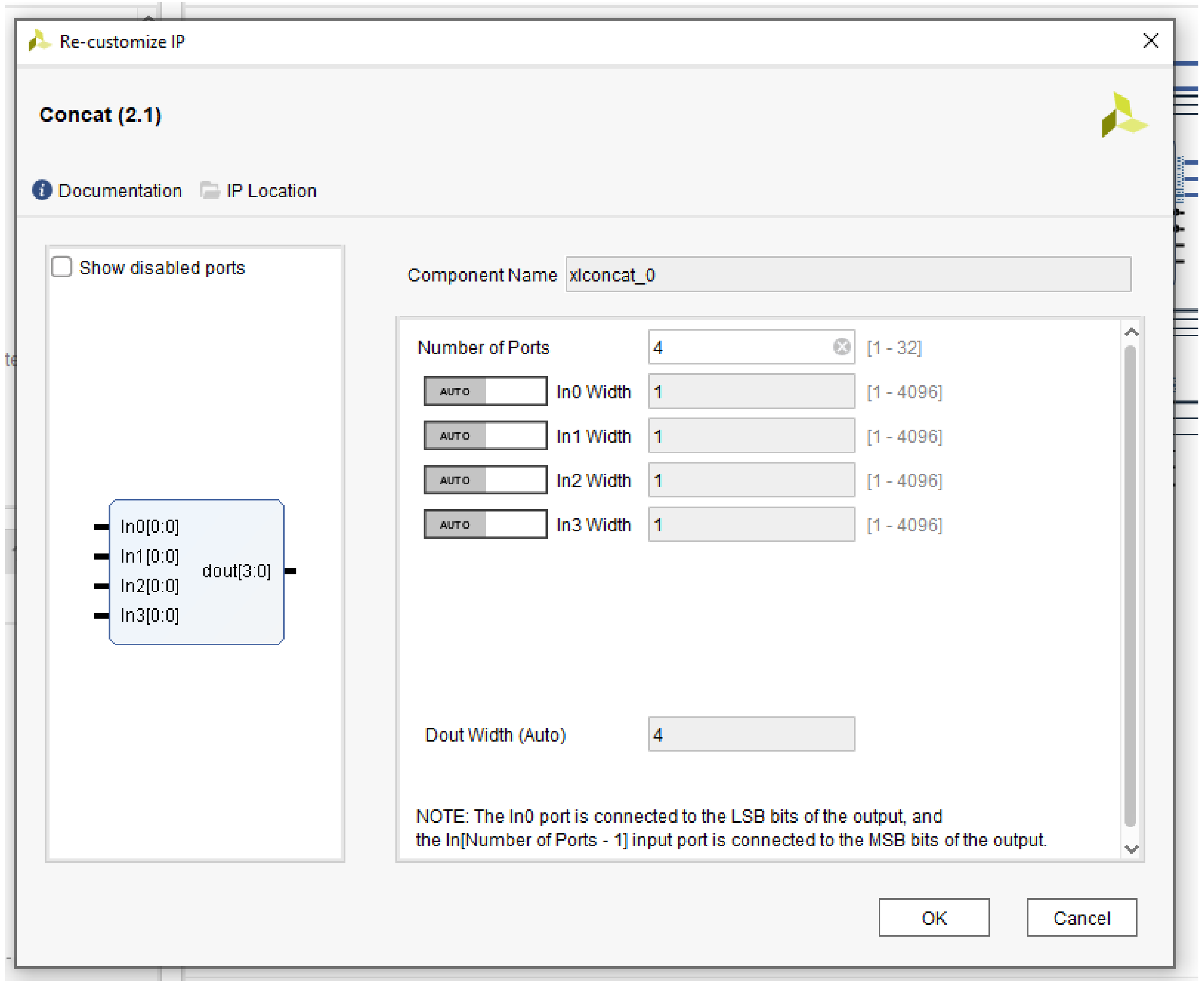
# Lab: FFT Verification

Click on + and add in a

concat block



# Lab: FFT Verification



Re-customize IP

Concat (2.1)

Documentation  IP Location

☐ Show disabled ports

Component Name xlconcat_0

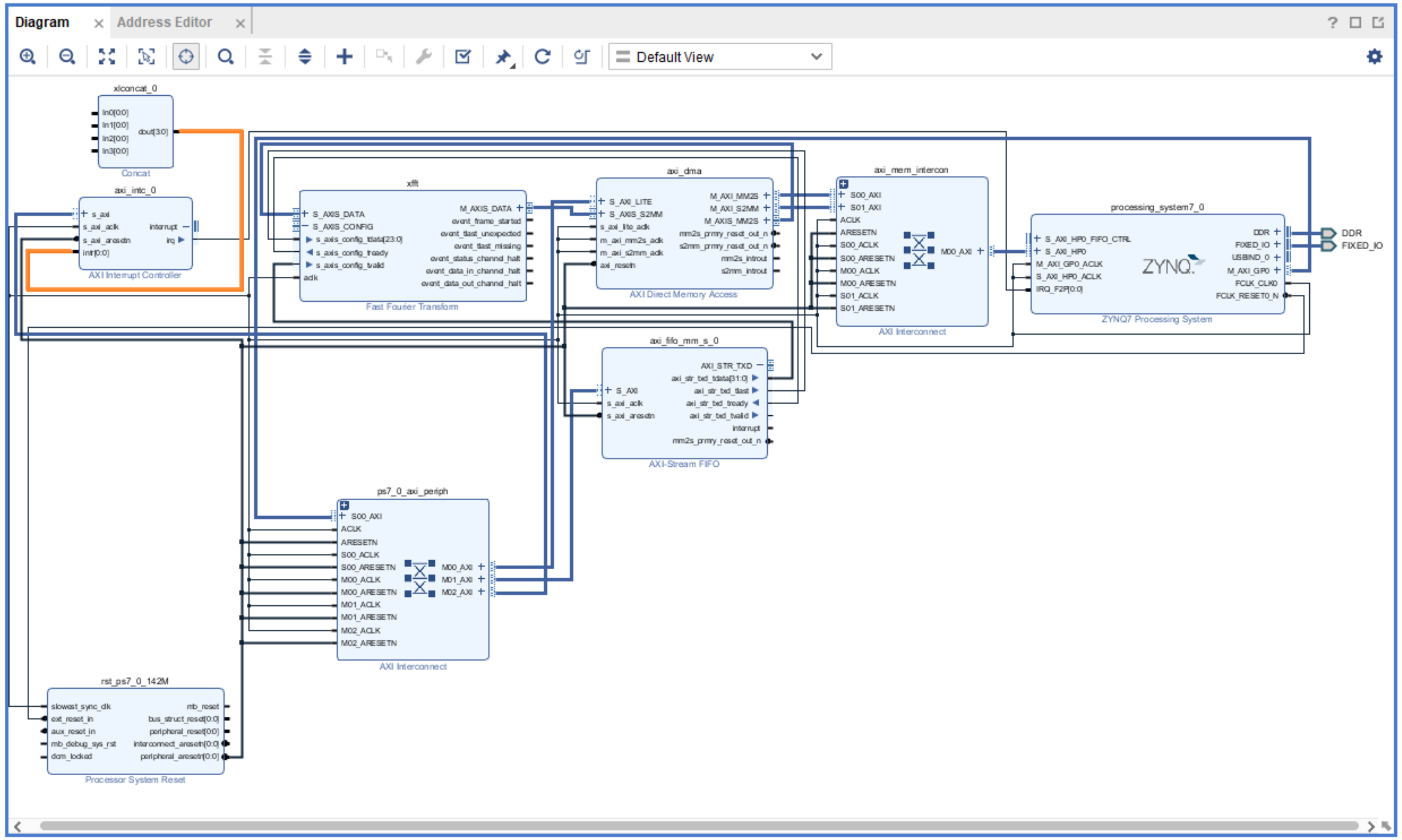
| | | |
|---|---|---|
| Number of Ports | 4 | [1 - 32] |
| AUTO In0 Width | 1 | [1 - 4096] |
| AUTO In1 Width | 1 | [1 - 4096] |
| AUTO In2 Width | 1 | [1 - 4096] |
| AUTO In3 Width | 1 | [1 - 4096] |
| Dout Width (Auto) | 4 | |

NOTE: The In0 port is connected to the LSB bits of the output, and the In[Number of Ports - 1] input port is connected to the MSB bits of the output.

OK  Cancel

# Lab: FFT Verification

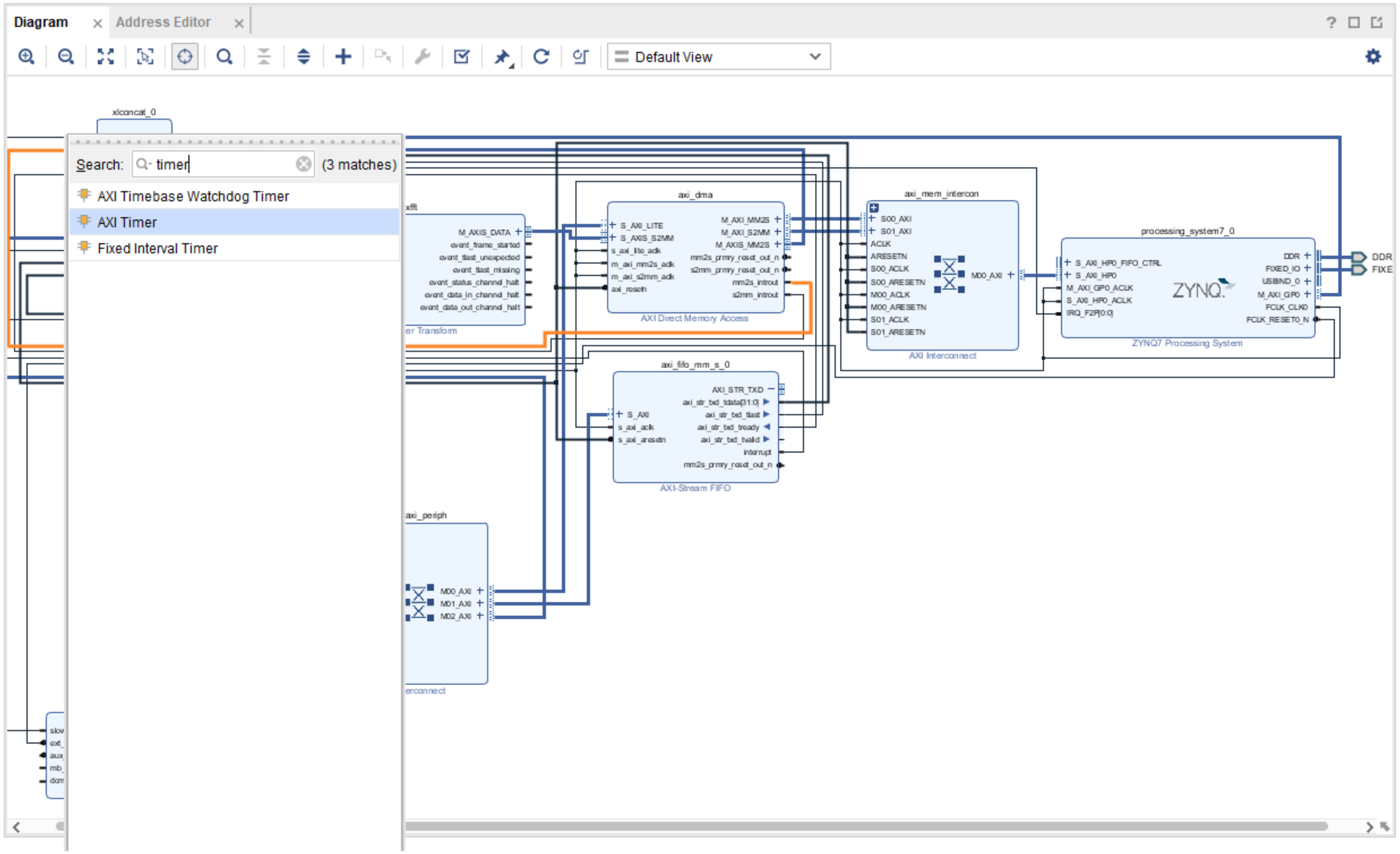
Connect the output of the concat block to the AXI Interrupt controller INT input



# Lab: FFT Verification

Click on the + symbol

and add in a AXI Timer

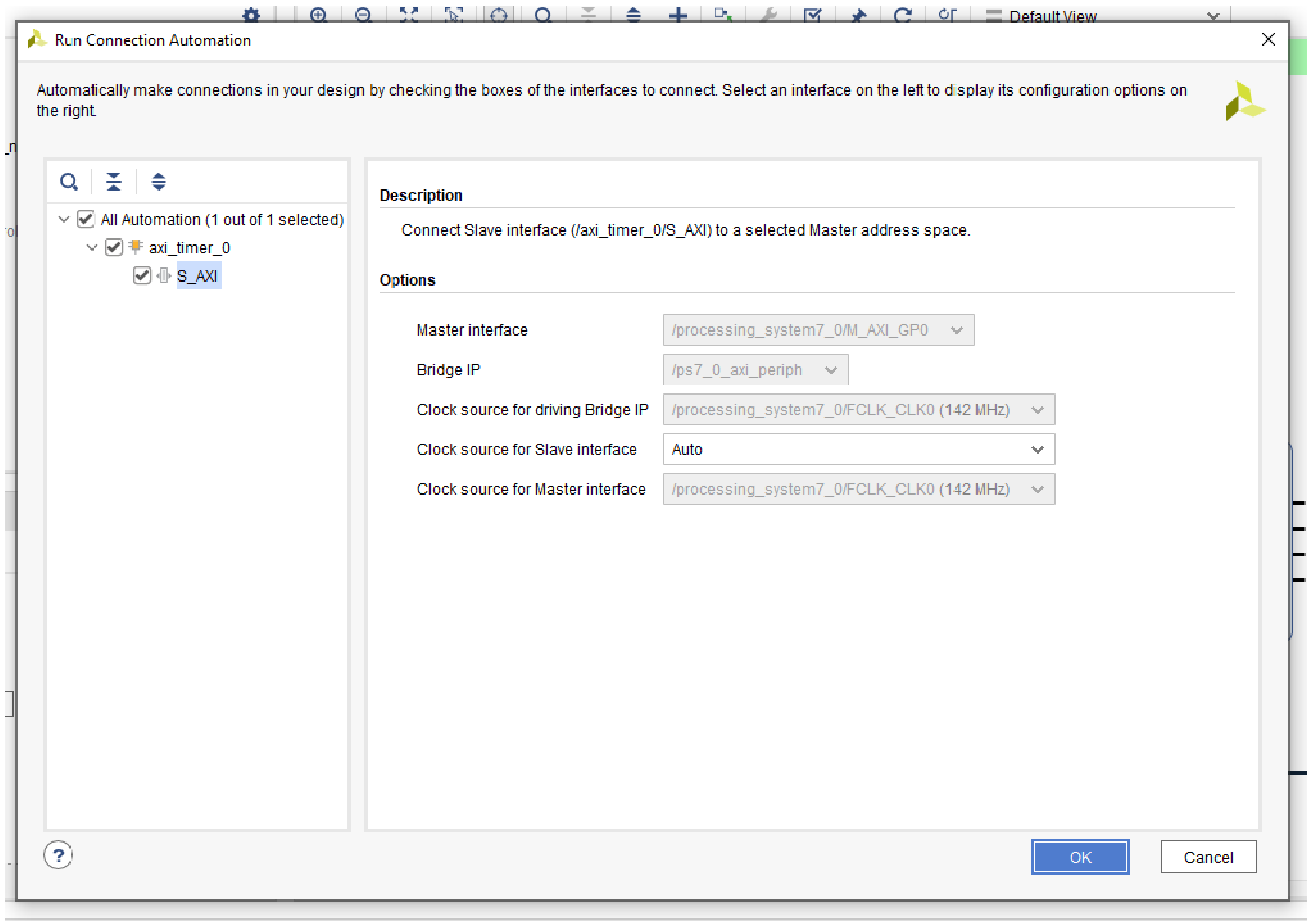

Search: timer (3 matches)

- AXI Timebase Watchdog Timer
- AXI Timer
- Fixed Interval Timer

ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Run the connection automation



Run Connection Automation

Automatically make connections in your design by checking the boxes of the interfaces to connect. Select an interface on the left to display its configuration options on the right.

- ☑ All Automation (1 out of 1 selected)
  - ☑ axi_timer_0
    - ☑ S_AXI

**Description**

Connect Slave interface (/axi_timer_0/S_AXI) to a selected Master address space.

**Options**

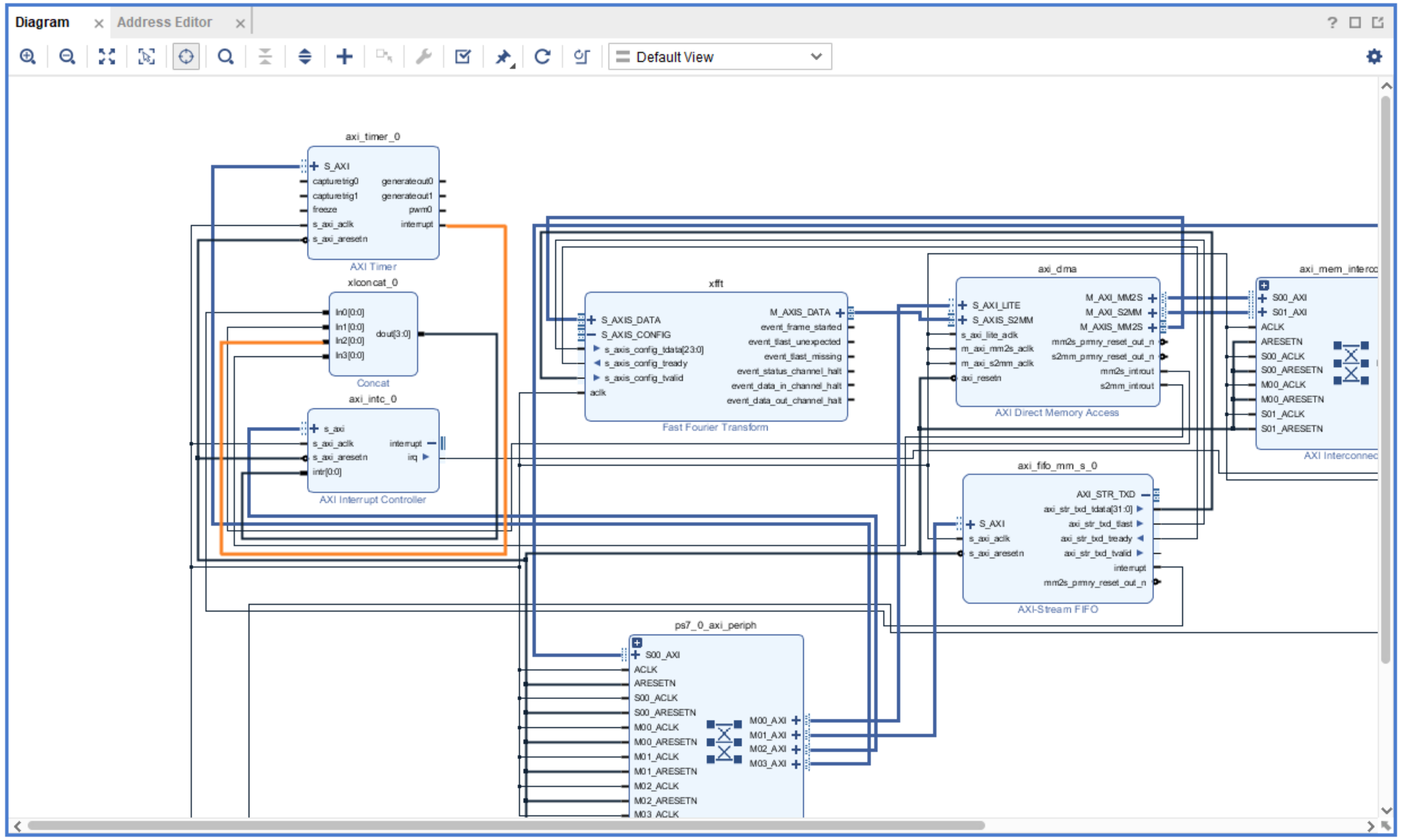
| | |
|---|---|
| Master interface | /processing_system7_0/M_AXI_GP0 |
| Bridge IP | /ps7_0_axi_periph |
| Clock source for driving Bridge IP | /processing_system7_0/FCLK_CLK0 (142 MHz) |
| Clock source for Slave interface | Auto |
| Clock source for Master interface | /processing_system7_0/FCLK_CLK0 (142 MHz) |

ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Connect the interrupts to the concat block

- AXI Timer
- AXI DMA MM2S_INTOUT
- AXI DMA S2MM_INTOUT
- AXI Stream FIFO



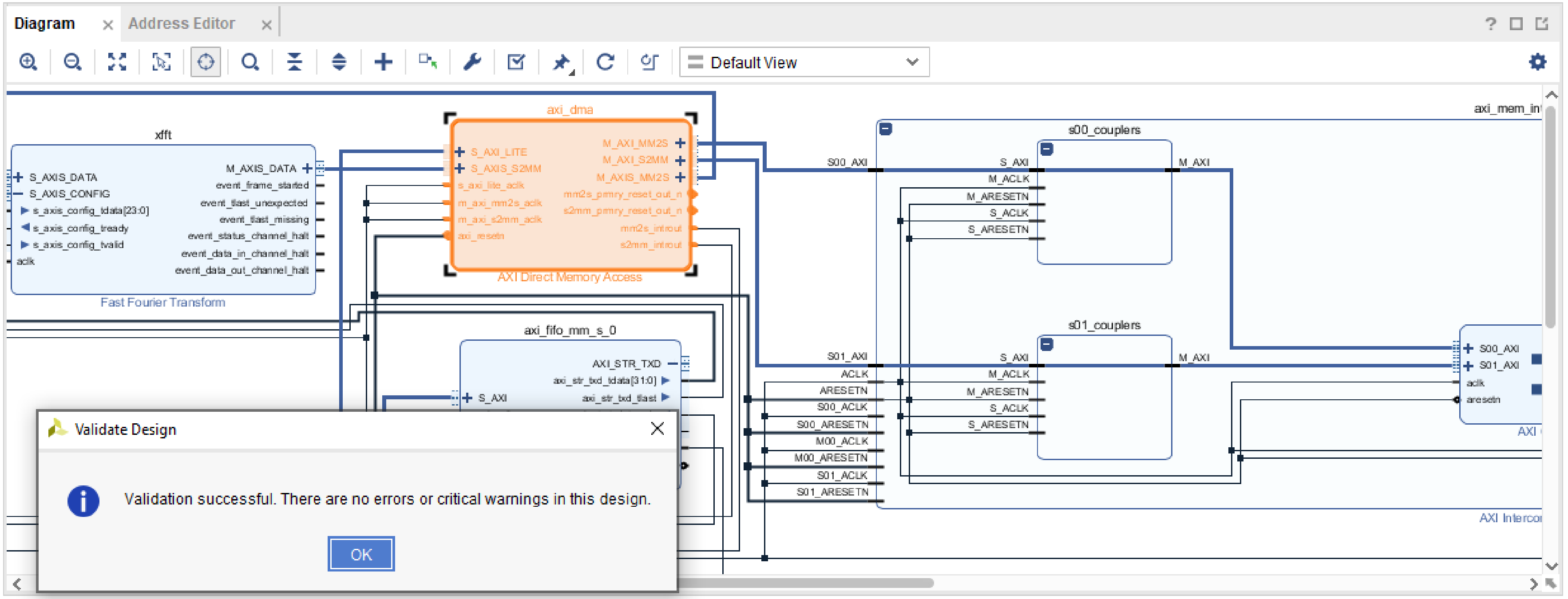
# Lab: FFT Verification

Validate the design

there should be no

error or critical

warnings

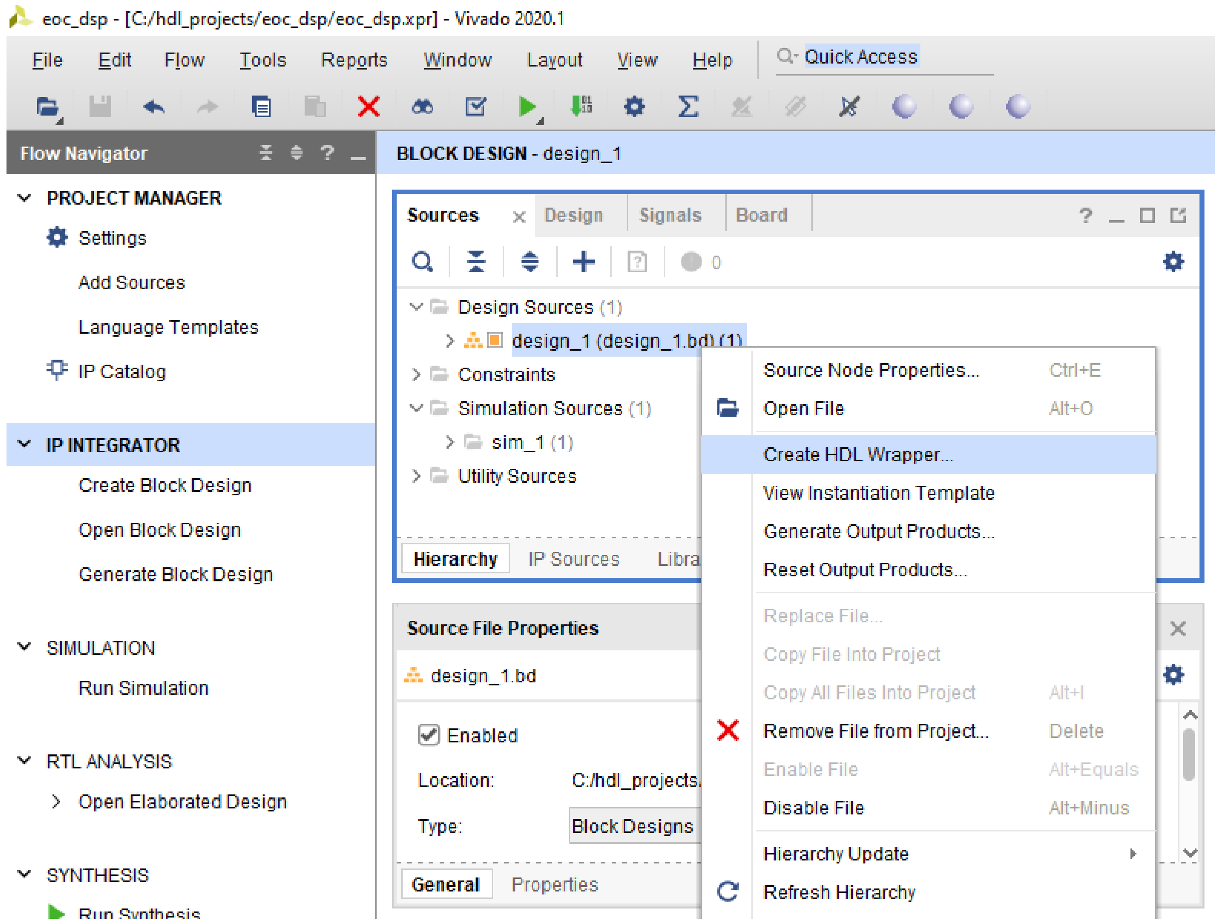


Validation successful. There are no errors or critical warnings in this design.

# Lab: FFT Verification

Right click on the design and

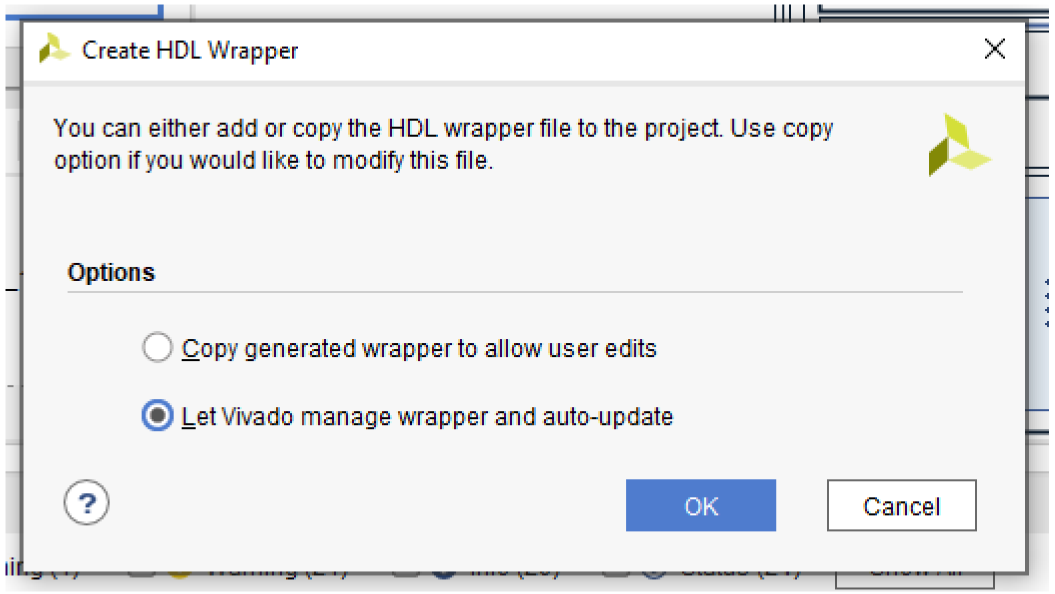
select Create HDL Wrapper



ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Let Vivado™ manage the wrapper


Create HDL Wrapper

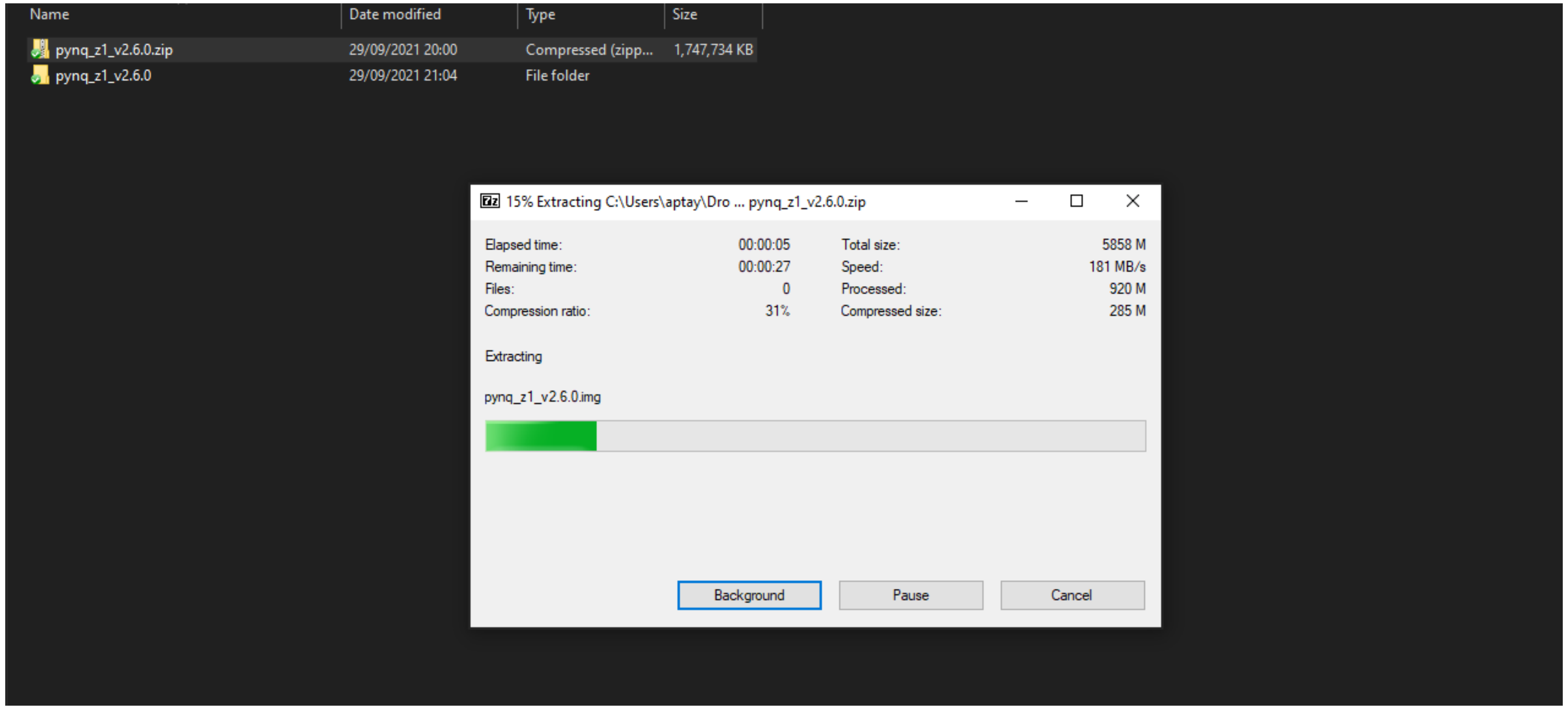
You can either add or copy the HDL wrapper file to the project. Use copy option if you would like to modify this file.

**Options**

- ○ Copy generated wrapper to allow user edits
- ● Let Vivado manage wrapper and auto-update

# Lab: FFT Verification

Extract the downloaded PYNQ™ image



| Name | Date modified | Type | Size |
|---|---|---|---|
| pynq_z1_v2.6.0.zip | 29/09/2021 20:00 | Compressed (zipp... | 1,747,734 KB |
| pynq_z1_v2.6.0 | 29/09/2021 21:04 | File folder | |

15% Extracting C:\Users\aptay\Dro ... pynq_z1_v2.6.0.zip

| | | | |
|---|---|---|---|
| Elapsed time: | 00:00:05 | Total size: | 5858 M |
| Remaining time: | 00:00:27 | Speed: | 181 MB/s |
| Files: | 0 | Processed: | 920 M |
| Compression ratio: | 31% | Compressed size: | 285 M |

Extracting

pynq_z1_v2.6.0.img

ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Write the image to a SD card. Once completed, insert the SD card in the Avnet ZU Board. Connect an Ethernet cable and power via a USB cable

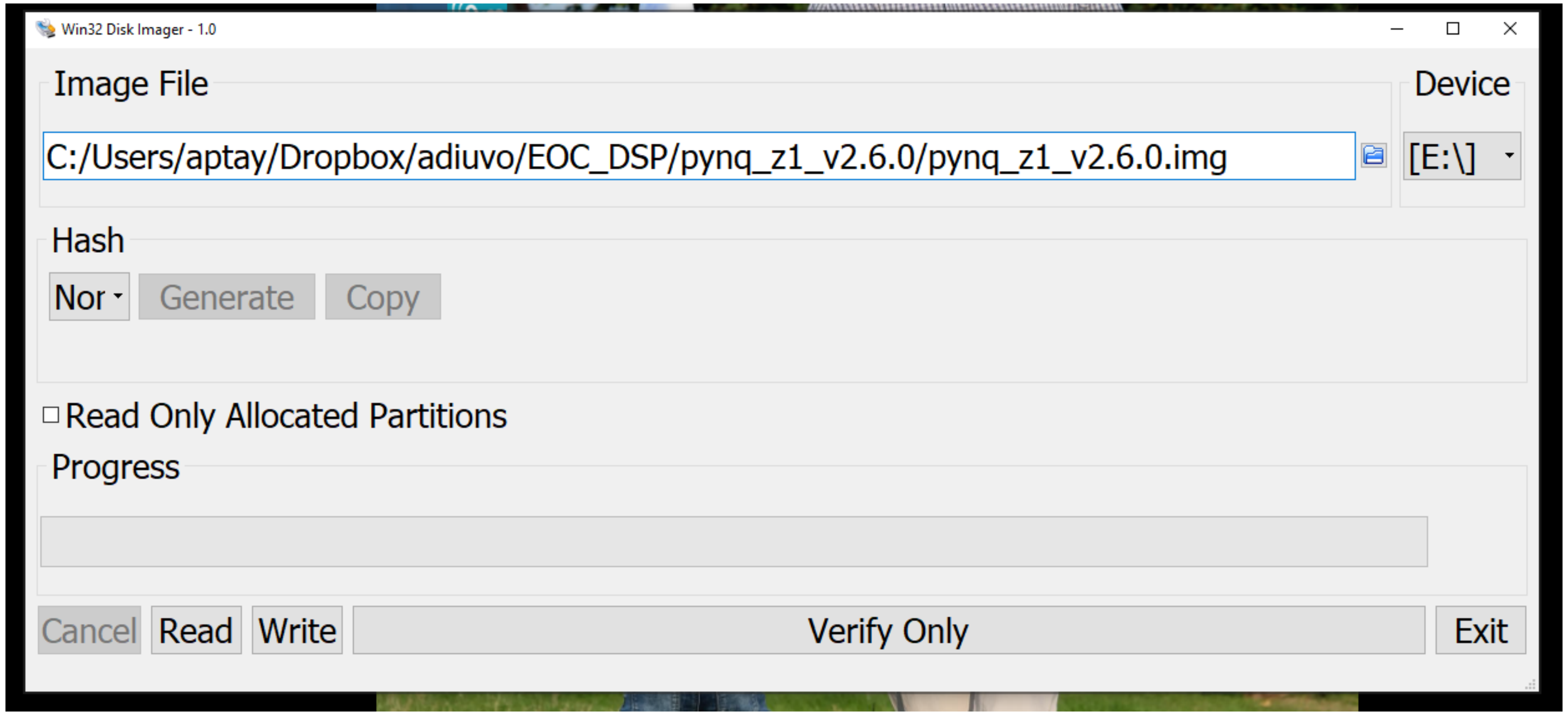

Win32 Disk Imager - 1.0

Image File: C:/Users/aptay/Dropbox/adiuvo/EOC_DSP/pynq_z1_v2.6.0/pynq_z1_v2.6.0.img

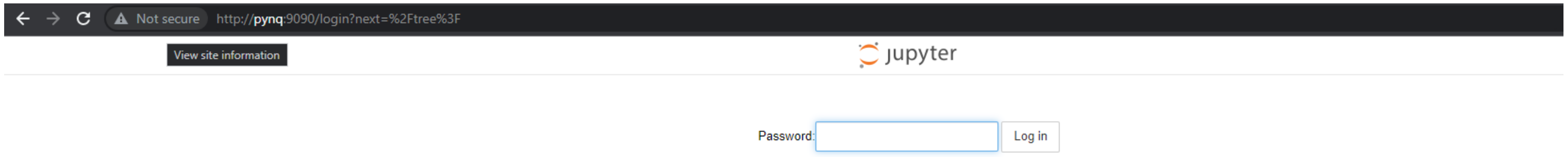
Device: [E:\]

Hash: Nor

# Lab: FFT Verification

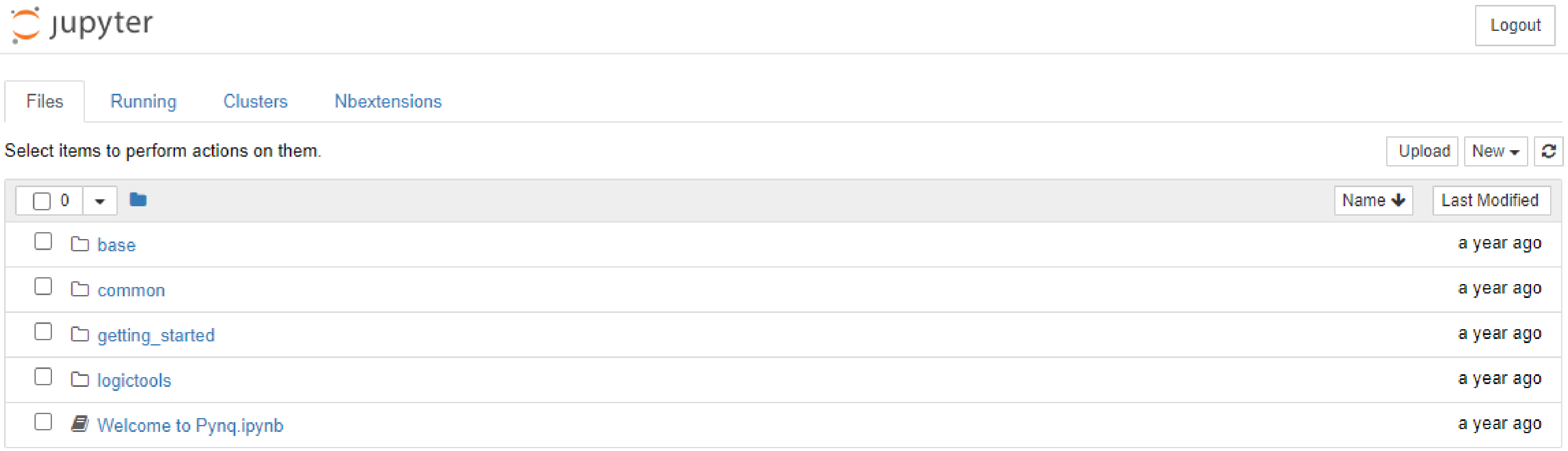
Once the board boots, wait for the LEDs to flash. In a browser enter the address pynq:9090

when prompted enter the password xilinx



## Lab: FFT Verification

Once logged in, you should see the folder structure below



Select items to perform actions on them.

| | |
|---|---|
| base | a year ago |
| common | a year ago |
| getting_started | a year ago |
| logictools | a year ago |
| Welcome to Pynq.ipynb | a year ago |

# Lab: FFT Verification

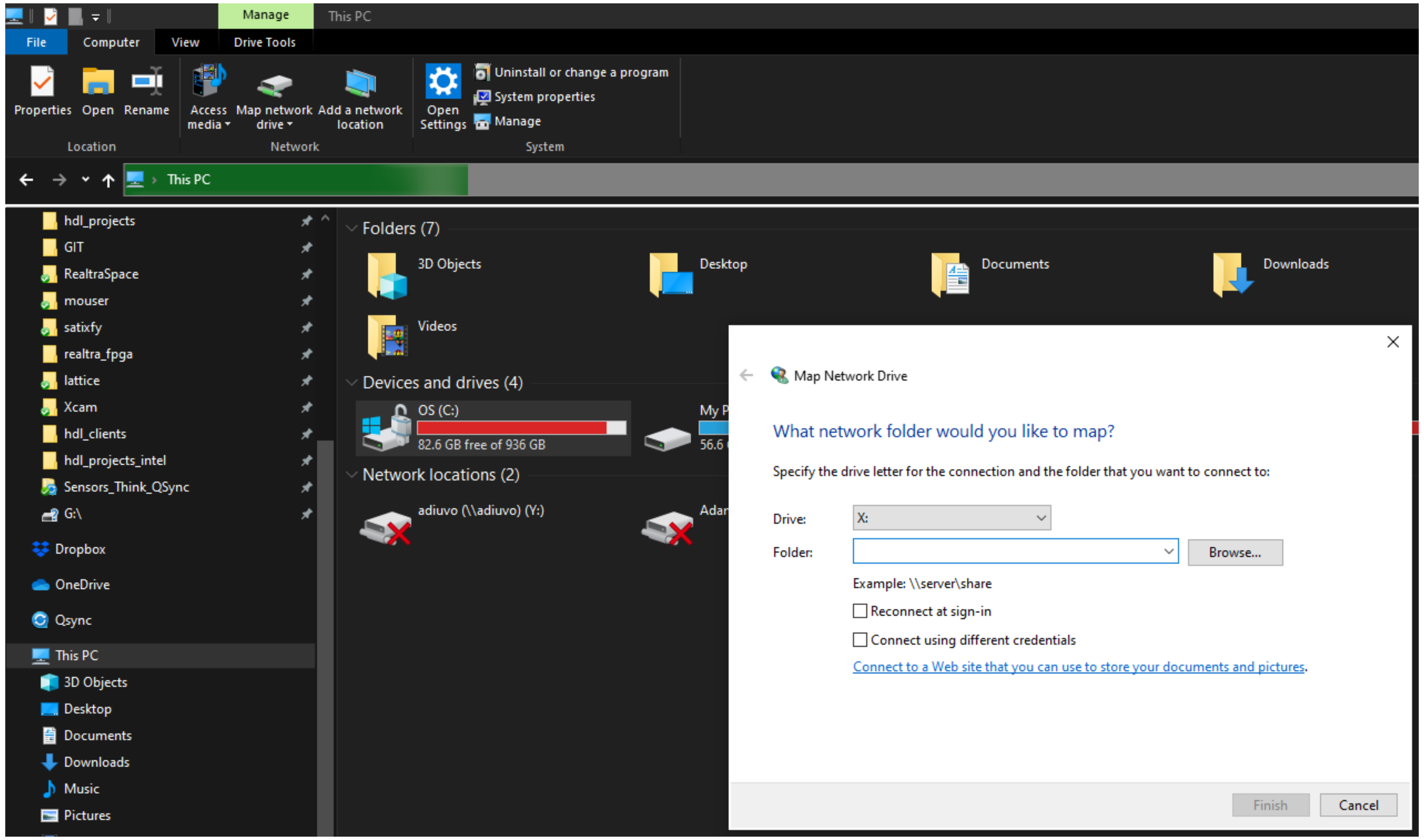
In a file explorer map a

network drive to

\\pynq \xilinx

Select connect using different

credentials



Map Network Drive

What network folder would you like to map?

Specify the drive letter for the connection and the folder that you want to connect to:

Drive: X:

Folder: Browse...

Example: \\server\share

☐ Reconnect at sign-in

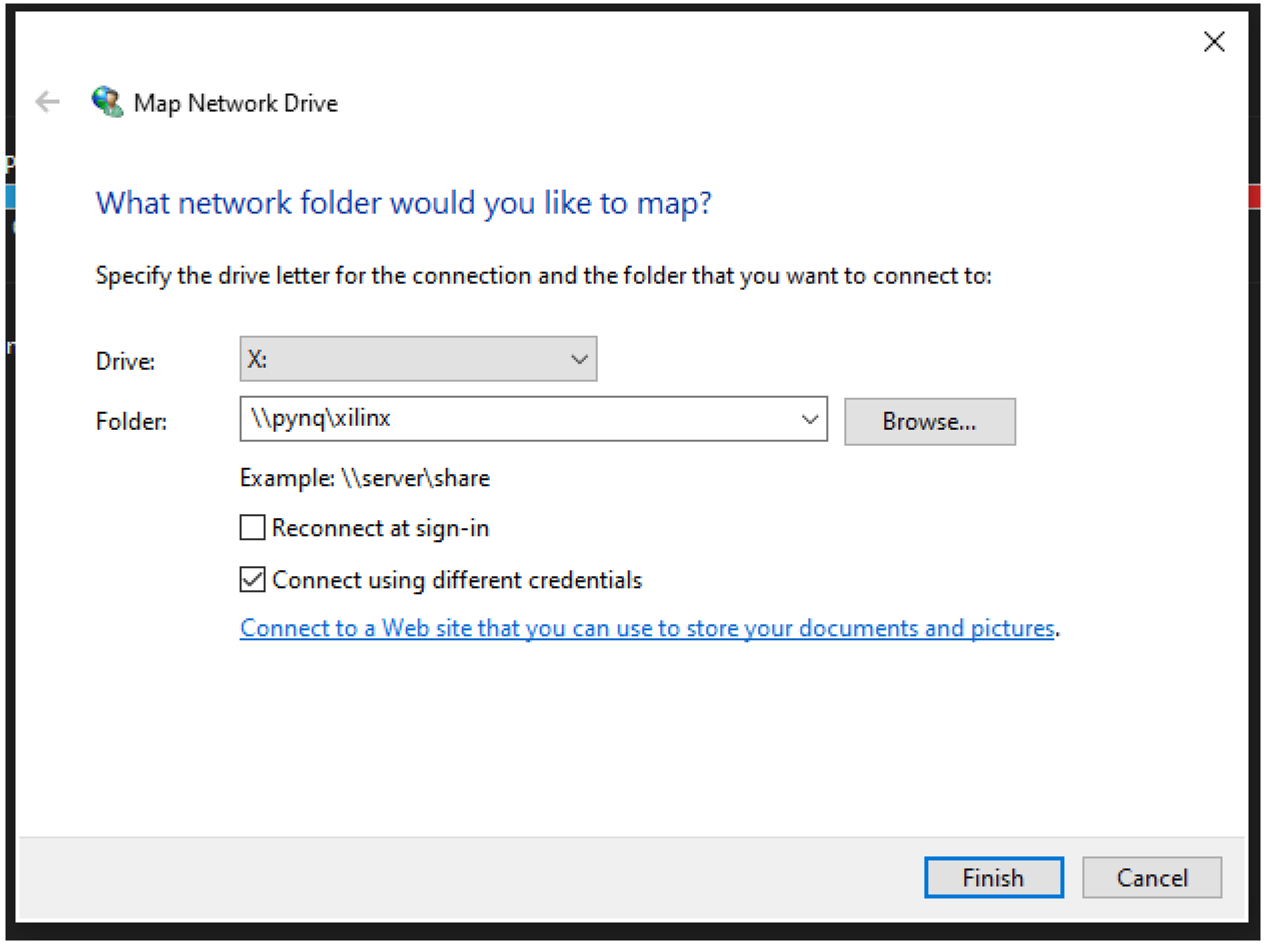
☐ Connect using different credentials

Connect to a Web site that you can use to store your documents and pictures.

Finish Cancel

# Lab: FFT Verification

Completed Map drive, click OK



Map Network Drive

What network folder would you like to map?

Specify the drive letter for the connection and the folder that you want to connect to:

Drive: X:

Folder: \\pynq\xilinx

Example: \\server\share

☐ Reconnect at sign-in

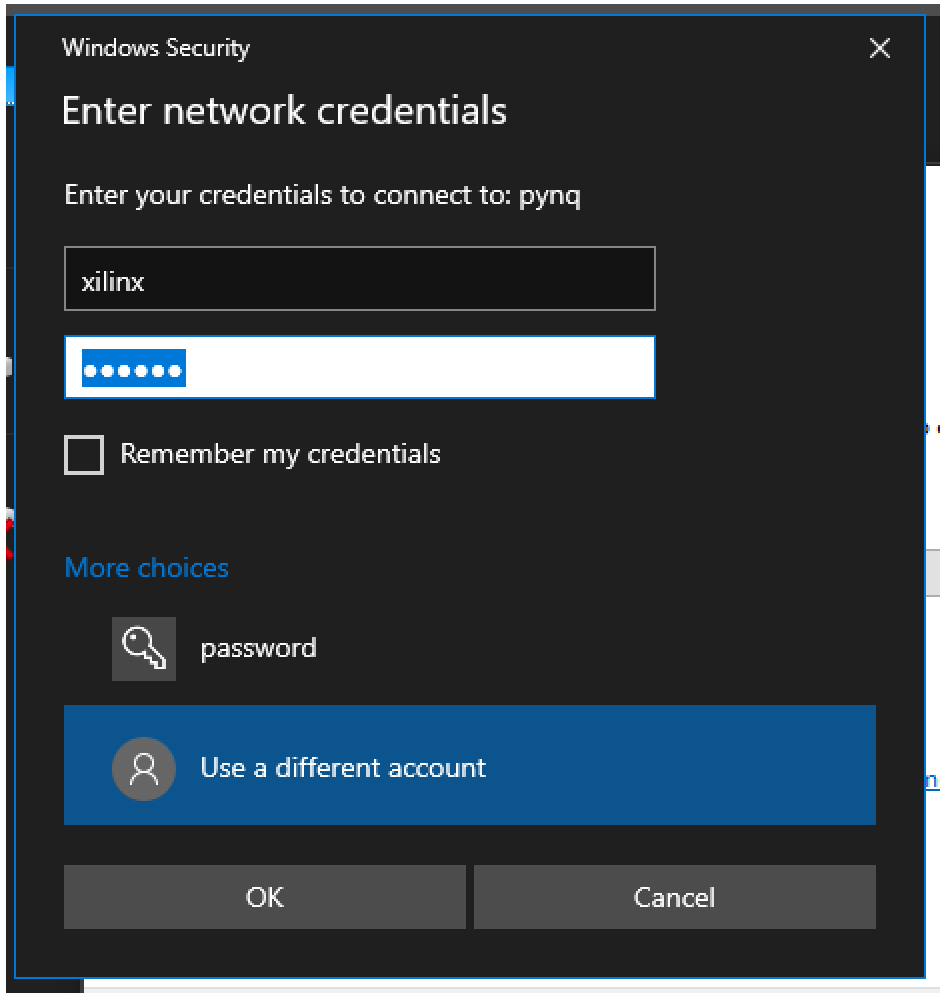
☑ Connect using different credentials

Connect to a Web site that you can use to store your documents and pictures.

# Lab: FFT Verification

Enter the username and password as

Xilinx click OK


Windows Security

Enter network credentials

Enter your credentials to connect to: pynq

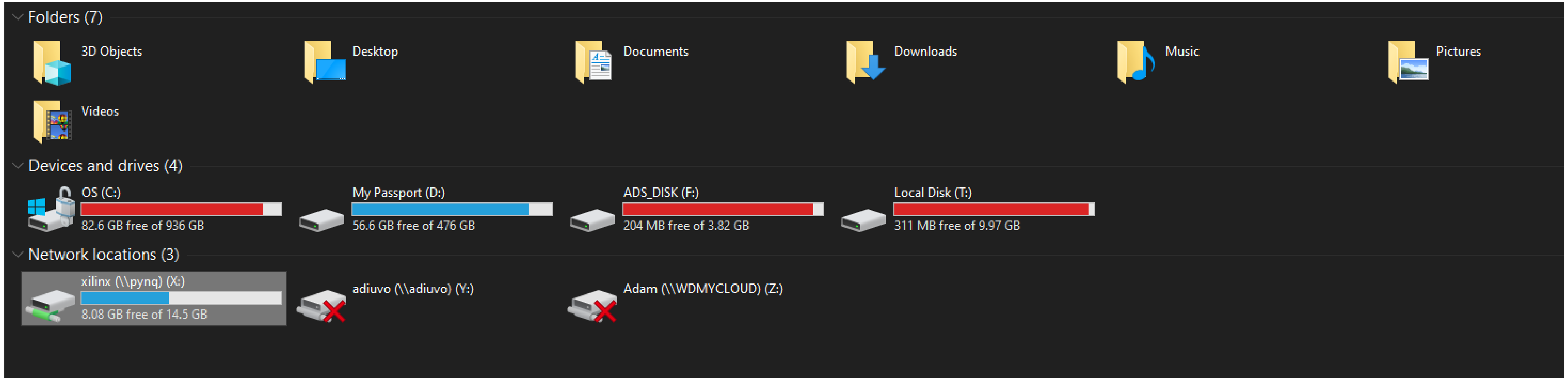
xilinx

••••••

Remember my credentials

# Lab: FFT Verification

The PYNQ™ drive should not appear as a samba server



Folders (7)

3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos

Devices and drives (4)

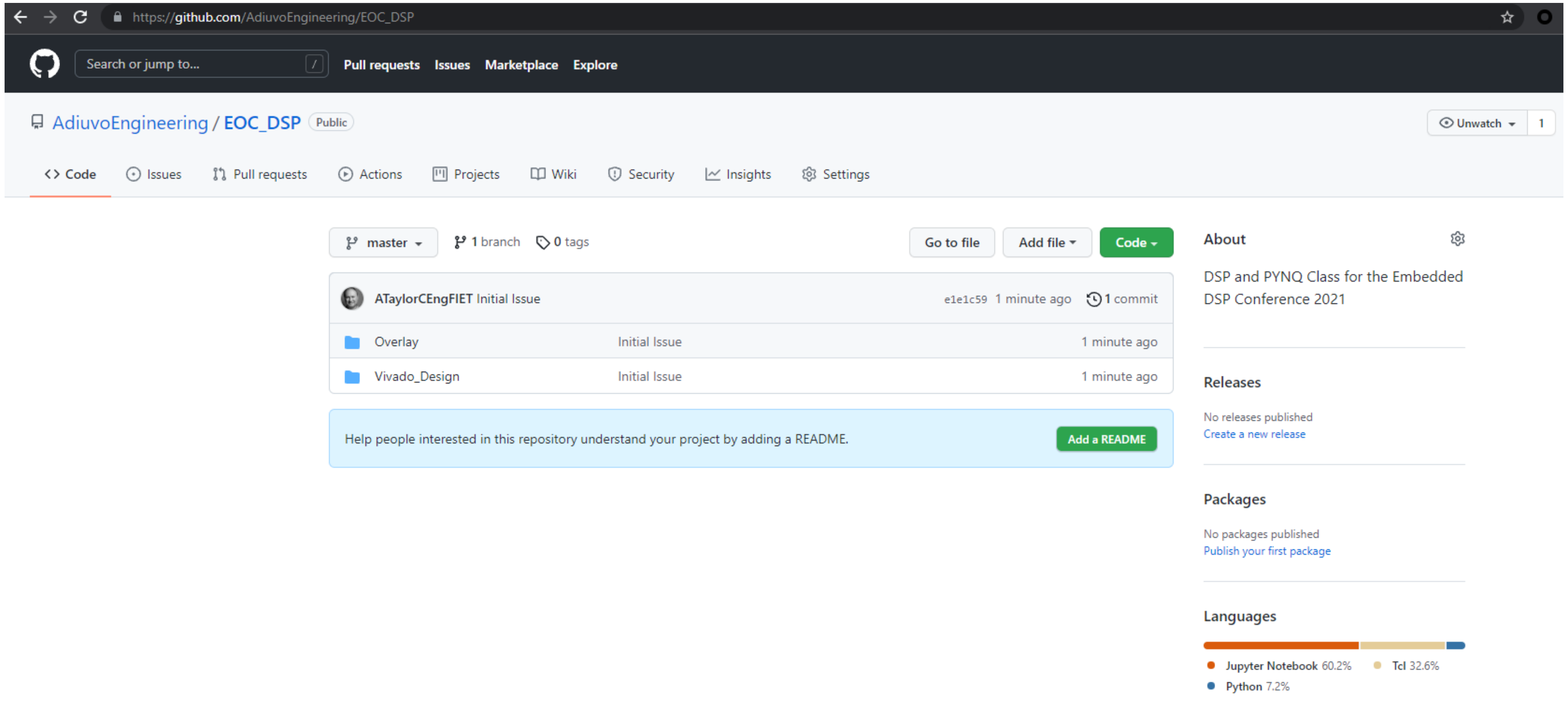
OS (C:) 82.6 GB free of 936 GB
My Passport (D:) 56.6 GB free of 476 GB
ADS_DISK (F:) 204 MB free of 3.82 GB
Local Disk (T:) 311 MB free of 9.97 GB

Network locations (3)

xilinx (\\pynq) (X:) 8.08 GB free of 14.5 GB
adiuvo (\\adiuvo) (Y:)
Adam (\\WDMYCLOUD) (Z:)

# Lab: FFT Verification

Clone the repository - https://github.com/ATaylorCEngFIET/MZ490



https://github.com/AdiuvoEngineering/EOC_DSP

AdiuvoEngineering / EOC_DSP Public

master · 1 branch · 0 tags

ATaylorCEngFIET Initial Issue · e1e1c59 1 minute ago · 1 commit

| | | |
|---|---|---|
| Overlay | Initial Issue | 1 minute ago |
| Vivado_Design | Initial Issue | 1 minute ago |

Help people interested in this repository understand your project by adding a README.

About

DSP and PYNQ Class for the Embedded DSP Conference 2021

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

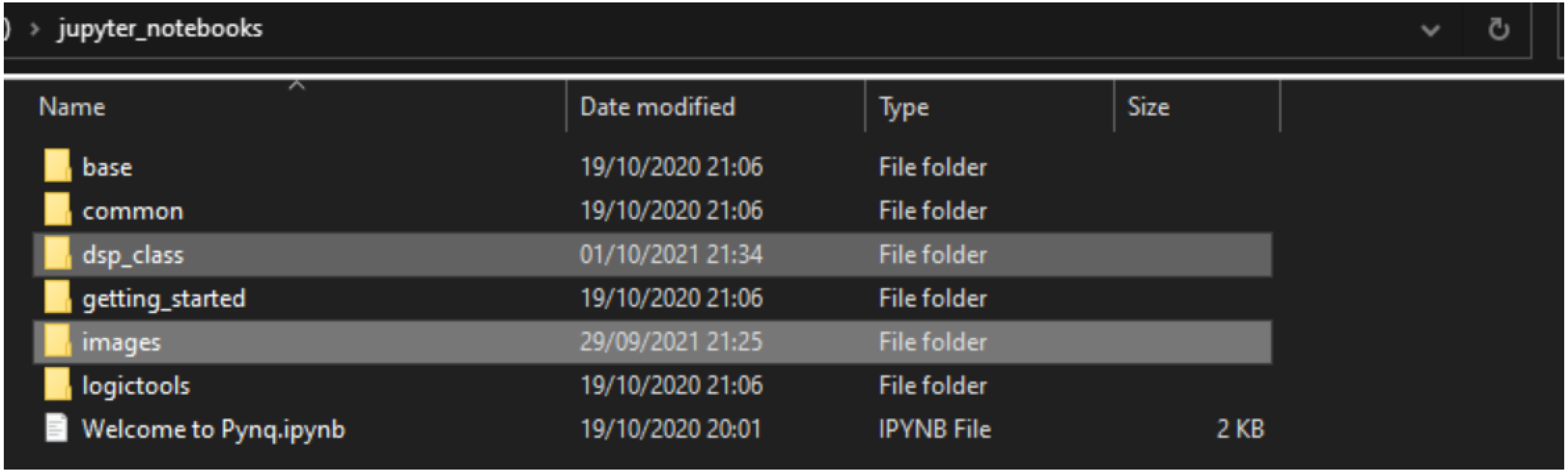
Languages

Jupyter Notebook 60.2% · Tcl 32.6% · Python 7.2%

ADIUVO ENGINEERING AND TRAINING, LTD.

## Lab: FFT Verification

From the Cloned Repo copy the directory Images and dsp_class to the PYNQ™ boards Jupyter notebooks directory



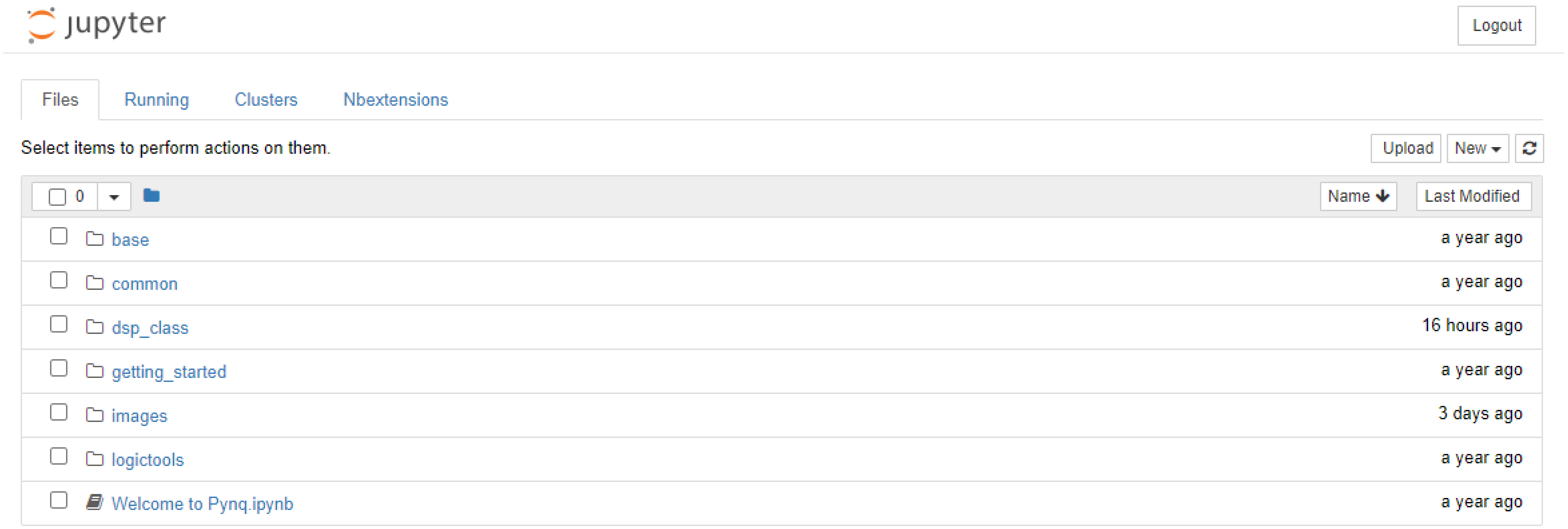
jupyter_notebooks

| Name | Date modified | Type | Size |
|---|---|---|---|
| base | 19/10/2020 21:06 | File folder | |
| common | 19/10/2020 21:06 | File folder | |
| dsp_class | 01/10/2021 21:34 | File folder | |
| getting_started | 19/10/2020 21:06 | File folder | |
| images | 29/09/2021 21:25 | File folder | |
| logictools | 19/10/2020 21:06 | File folder | |
| Welcome to Pynq.ipynb | 19/10/2020 20:01 | IPYNB File | 2 KB |

ADIUVO ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

You should see a new directory in the PYNQ™ environment. Select DSP_CLASS



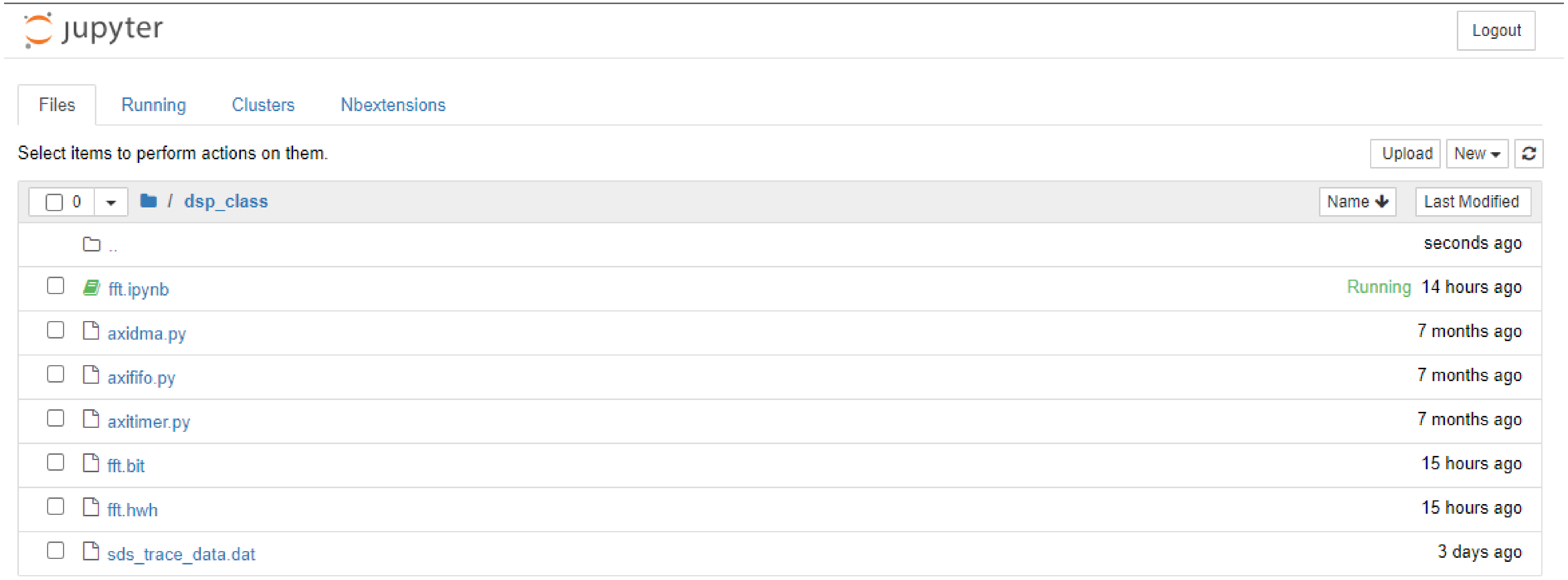
Select items to perform actions on them.

| | |
|---|---|
| base | a year ago |
| common | a year ago |
| dsp_class | 16 hours ago |
| getting_started | a year ago |
| images | 3 days ago |
| logictools | a year ago |
| Welcome to Pynq.ipynb | a year ago |

ADIUVO
ENGINEERING AND TRAINING, LTD.

# Lab: FFT Verification

Select fft.ipynb it will open and start running



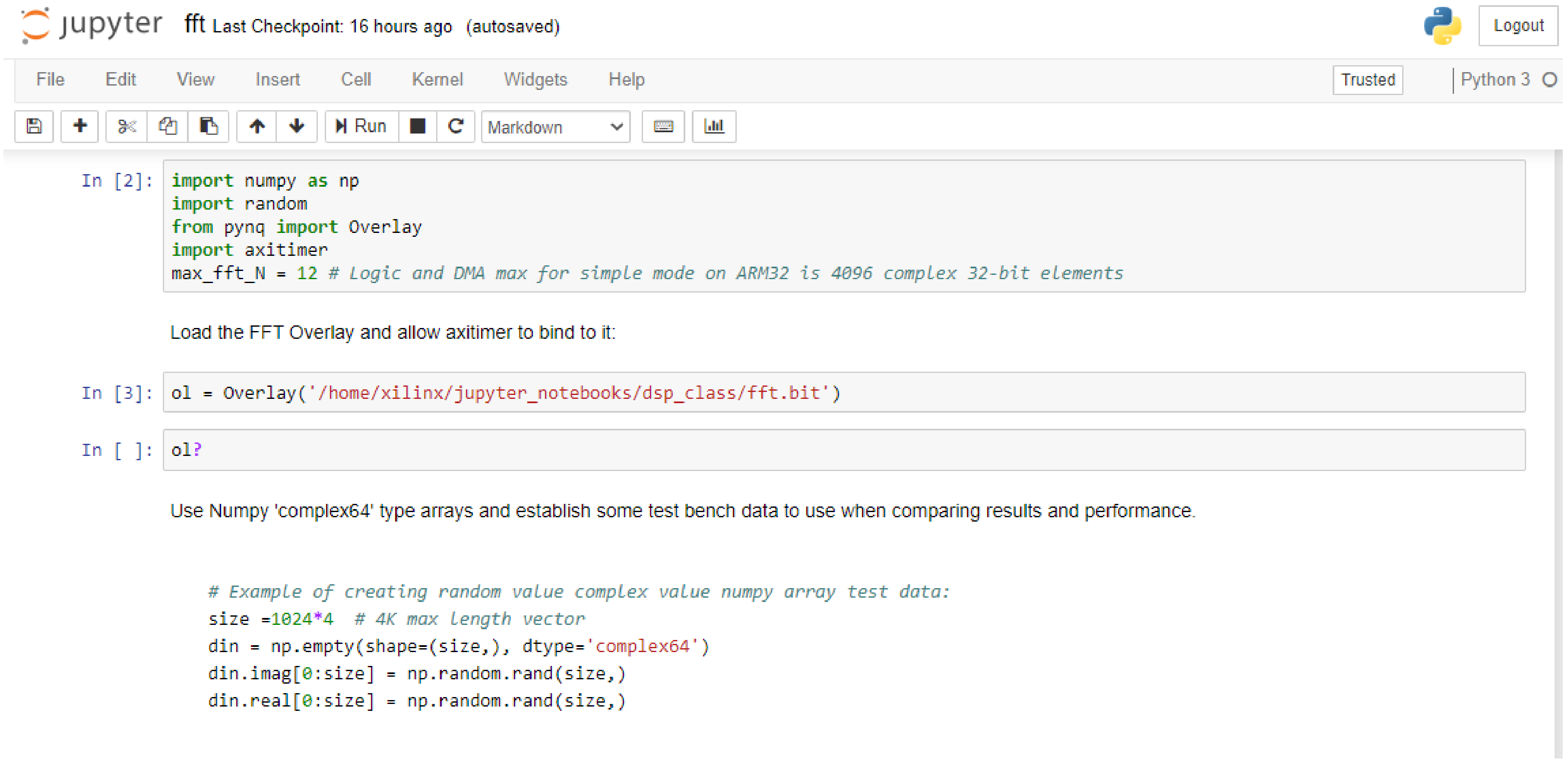
Select items to perform actions on them.

/ dsp_class

| | |
|---|---|
| .. | seconds ago |
| fft.ipynb | Running 14 hours ago |
| axidma.py | 7 months ago |
| axififo.py | 7 months ago |
| axitimer.py | 7 months ago |
| fft.bit | 15 hours ago |
| fft.hwh | 15 hours ago |
| sds_trace_data.dat | 3 days ago |

# Lab: FFT Verification

Run each cell in turn in the notebook and notice the difference in performance between SW and HW Implementations



```
In [2]: import numpy as np
        import random
        from pynq import Overlay
        import axitimer
        max_fft_N = 12 # Logic and DMA max for simple mode on ARM32 is 4096 complex 32-bit elements
```

Load the FFT Overlay and allow axitimer to bind to it:

```
In [3]: ol = Overlay('/home/xilinx/jupyter_notebooks/dsp_class/fft.bit')
```

```
In [ ]: ol?
```

Use Numpy 'complex64' type arrays and establish some test bench data to use when comparing results and performance.

```
# Example of creating random value complex value numpy array test data:
size =1024*4  # 4K max length vector
din = np.empty(shape=(size,), dtype='complex64')
din.imag[0:size] = np.random.rand(size,)
din.real[0:size] = np.random.rand(size,)
```

ADIUVO
ENGINEERING AND TRAINING, LTD.

ADIUVO

ENGINEERING AND TRAINING, LTD.

www.adiuvoengineering.com

adam@adiuvoengineering.com