



Vivado Debug Tools

Course Agenda
2023

Agenda

- ▶ **Integrated Logic Analyzer (ILA)**
- ▶ **Virtual Input/Output(VIO)**
- ▶ **Integrated Bit Error Ratio Tester (IBERT)**



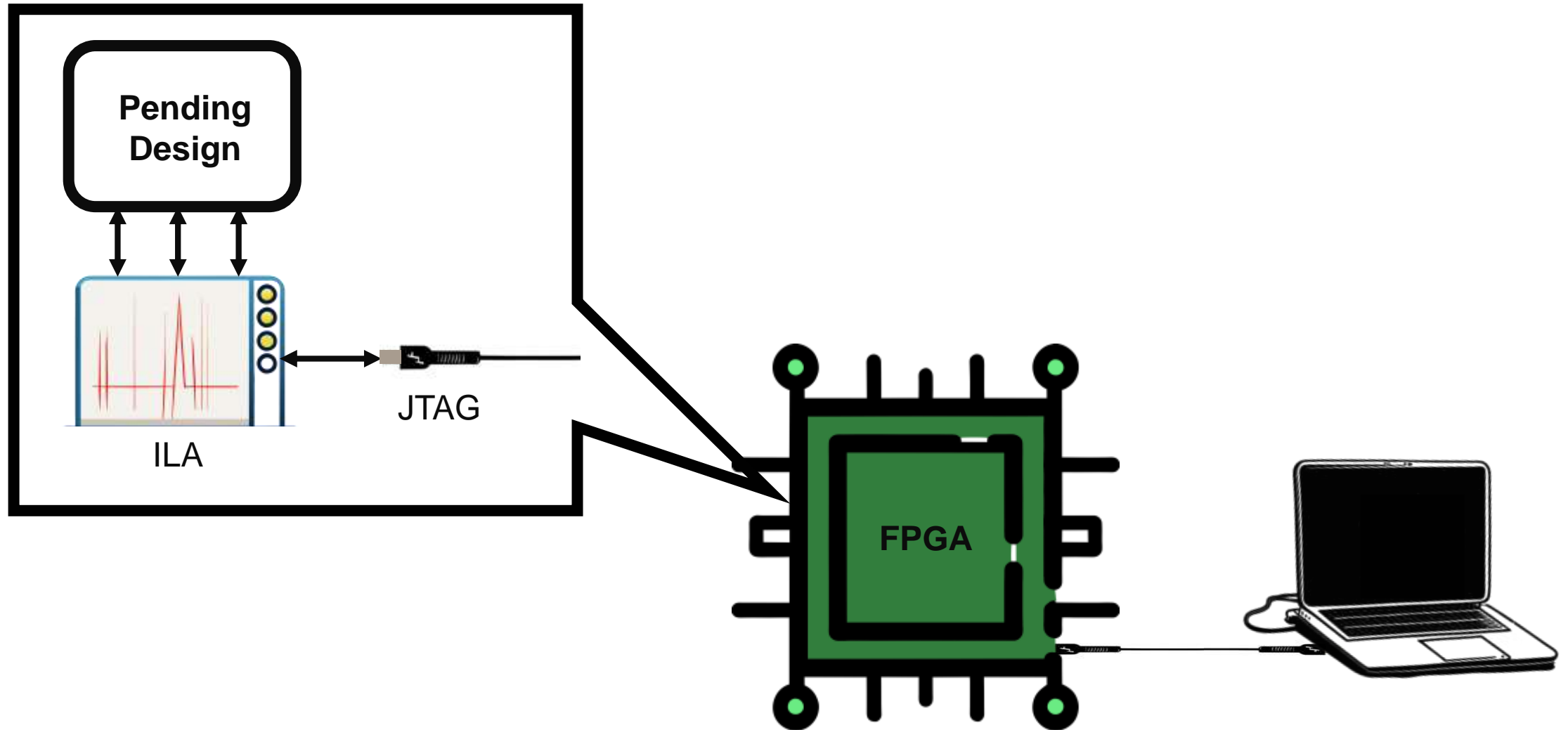
Integrated Logic Analyzer

Course Agenda
2023

Integrated Logic Analyzer

- ▶ **Several ways to add ILA**
- ▶ **ILA debugging**
- ▶ **ILA IP Setting**

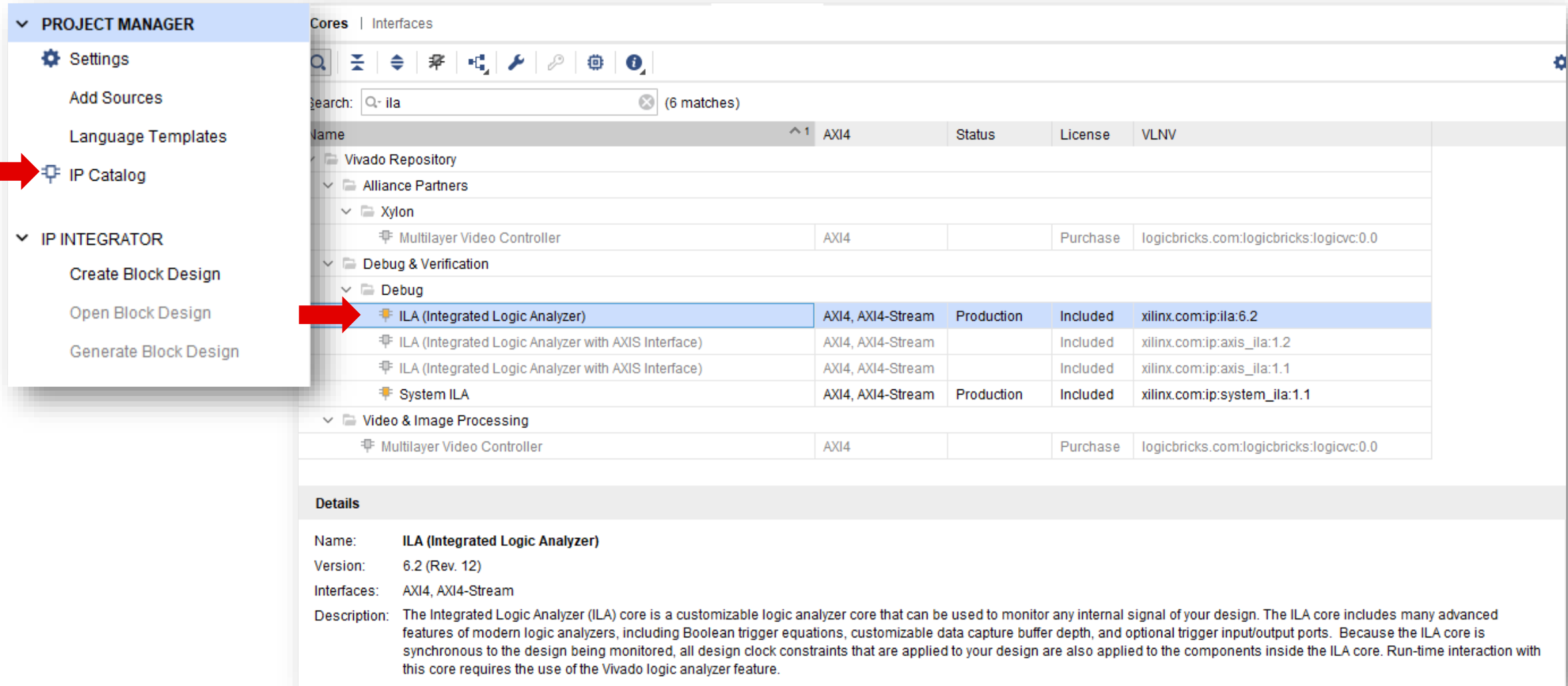
Integrated Logic Analyzer



Several ways to add ILA

- 1 Add ILA core through HDL code**
- 2 Add debug mark in the netlist**
- 3 Add ILA core in the block design**

Add ILA core through HDL code



The screenshot shows the Vivado IP Integrator interface. On the left, the 'PROJECT MANAGER' pane is open, displaying a list of options: 'Settings', 'Add Sources', 'Language Templates', 'IP Catalog' (highlighted with a red arrow), 'IP INTEGRATOR', 'Create Block Design', 'Open Block Design', and 'Generate Block Design'. The main area displays the 'Cores | Interfaces' pane, which is filtered by the search term 'ila'. The search results table is as follows:

Name	AXI4	Status	License	VLNV
Vivado Repository				
Alliance Partners				
Xylon				
Multilayer Video Controller	AXI4		Purchase	logicbricks.com:logicbricks:logicvc:0.0
Debug & Verification				
Debug				
ILA (Integrated Logic Analyzer)	AXI4, AXI4-Stream	Production	Included	xilinx.com:ip:ila:6.2
ILA (Integrated Logic Analyzer with AXIS Interface)	AXI4, AXI4-Stream		Included	xilinx.com:ip:axis_ila:1.2
ILA (Integrated Logic Analyzer with AXIS Interface)	AXI4, AXI4-Stream		Included	xilinx.com:ip:axis_ila:1.1
System ILA	AXI4, AXI4-Stream	Production	Included	xilinx.com:ip:system_ila:1.1
Video & Image Processing				
Multilayer Video Controller	AXI4		Purchase	logicbricks.com:logicbricks:logicvc:0.0

A red arrow points to the 'ILA (Integrated Logic Analyzer)' entry in the table. Below the table, the 'Details' pane for the selected core is shown:

Details

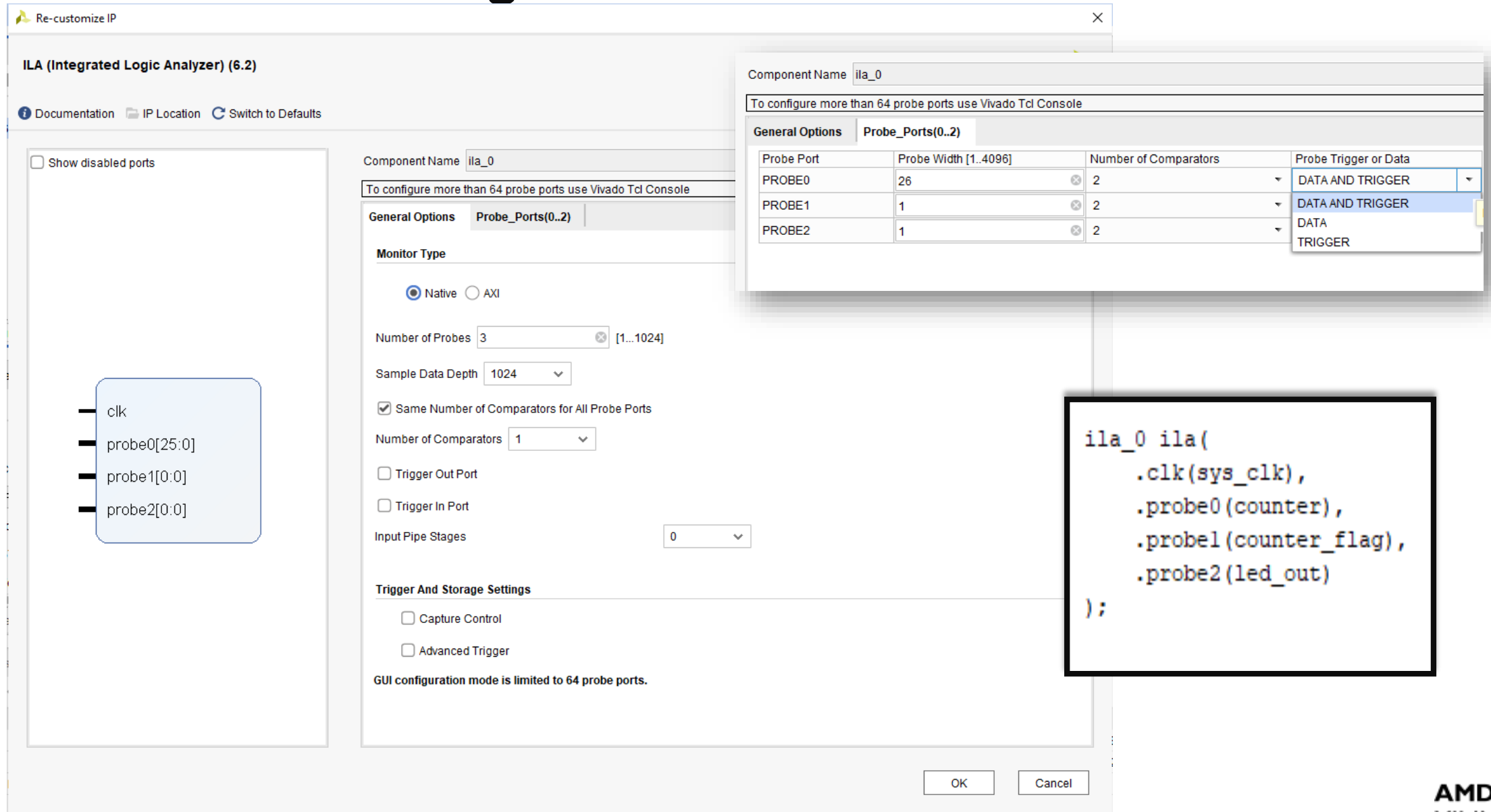
Name: **ILA (Integrated Logic Analyzer)**

Version: 6.2 (Rev. 12)

Interfaces: AXI4, AXI4-Stream

Description: The Integrated Logic Analyzer (ILA) core is a customizable logic analyzer core that can be used to monitor any internal signal of your design. The ILA core includes many advanced features of modern logic analyzers, including Boolean trigger equations, customizable data capture buffer depth, and optional trigger input/output ports. Because the ILA core is synchronous to the design being monitored, all design clock constraints that are applied to your design are also applied to the components inside the ILA core. Run-time interaction with this core requires the use of the Vivado logic analyzer feature.

Add ILA core through HDL code



Re-customize IP

ILA (Integrated Logic Analyzer) (6.2)

Documentation IP Location Switch to Defaults

☐ Show disabled ports

Component Name

To configure more than 64 probe ports use Vivado Tcl Console

General Options **Probe_Ports(0..2)**

Probe Port	Probe Width [1..4096]	Number of Comparators	Probe Trigger or Data
PROBE0	26	2	DATA AND TRIGGER
PROBE1	1	2	DATA AND TRIGGER
PROBE2	1	2	DATA

Monitor Type

☒ Native ☐ AXI

Number of Probes [1...1024]

Sample Data Depth

☒ Same Number of Comparators for All Probe Ports

Number of Comparators

☐ Trigger Out Port

☐ Trigger In Port

Input Pipe Stages

Trigger And Storage Settings

☐ Capture Control

☐ Advanced Trigger

GUI configuration mode is limited to 64 probe ports.

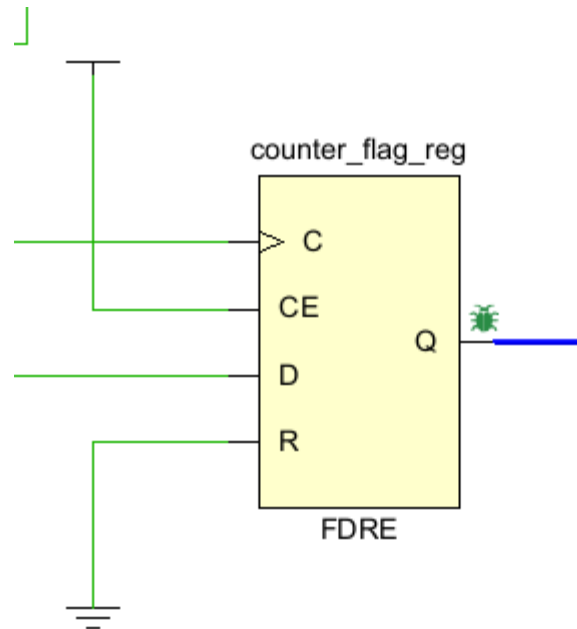
OK Cancel

```
ila_0 ila(  
    .clk(sys_clk),  
    .probe0(counter),  
    .probe1(counter_flag),  
    .probe2(led_out)  
);
```


Add debug mark in the netlist

```
(* mark_debug = "true" *) reg / wire
```

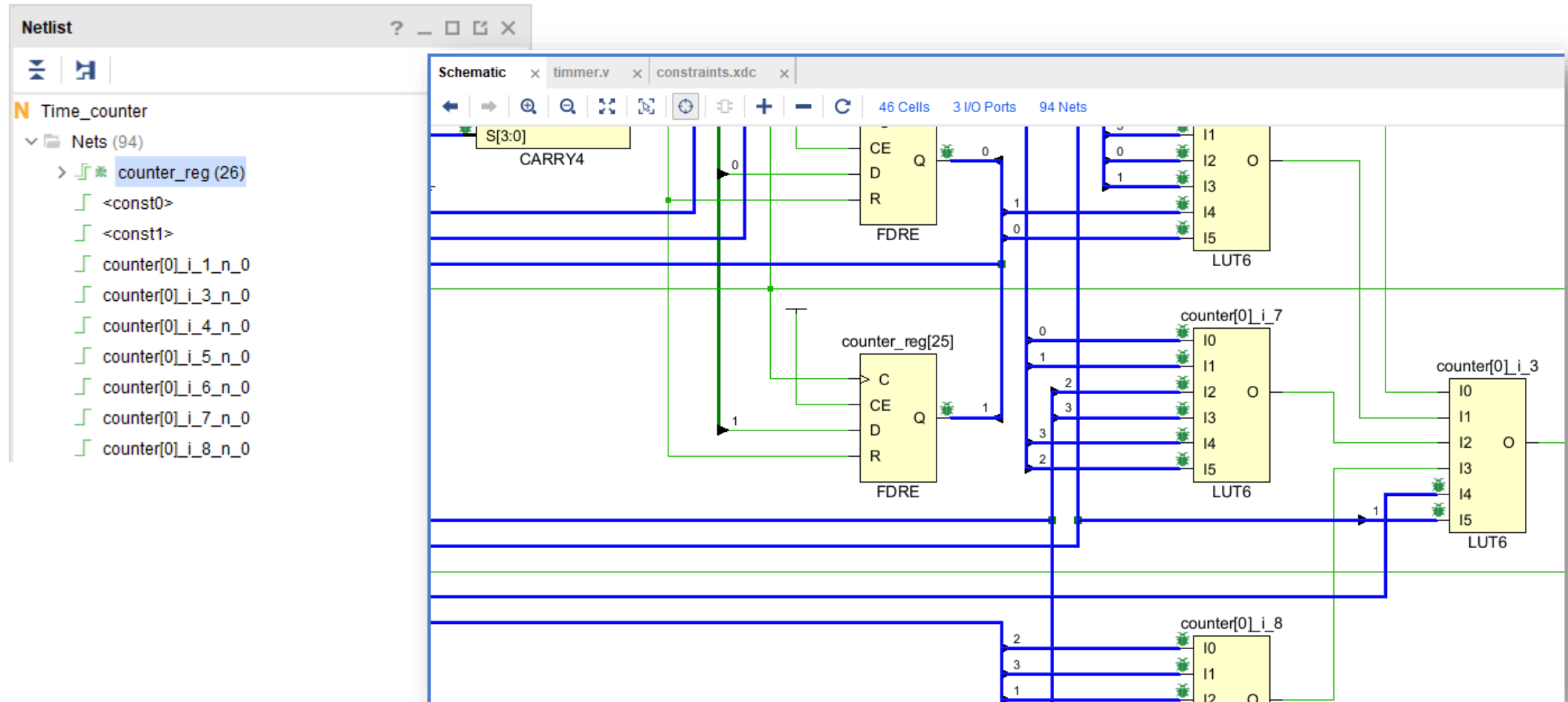
```
(* mark_debug = "true" *) reg counter_flag;
```



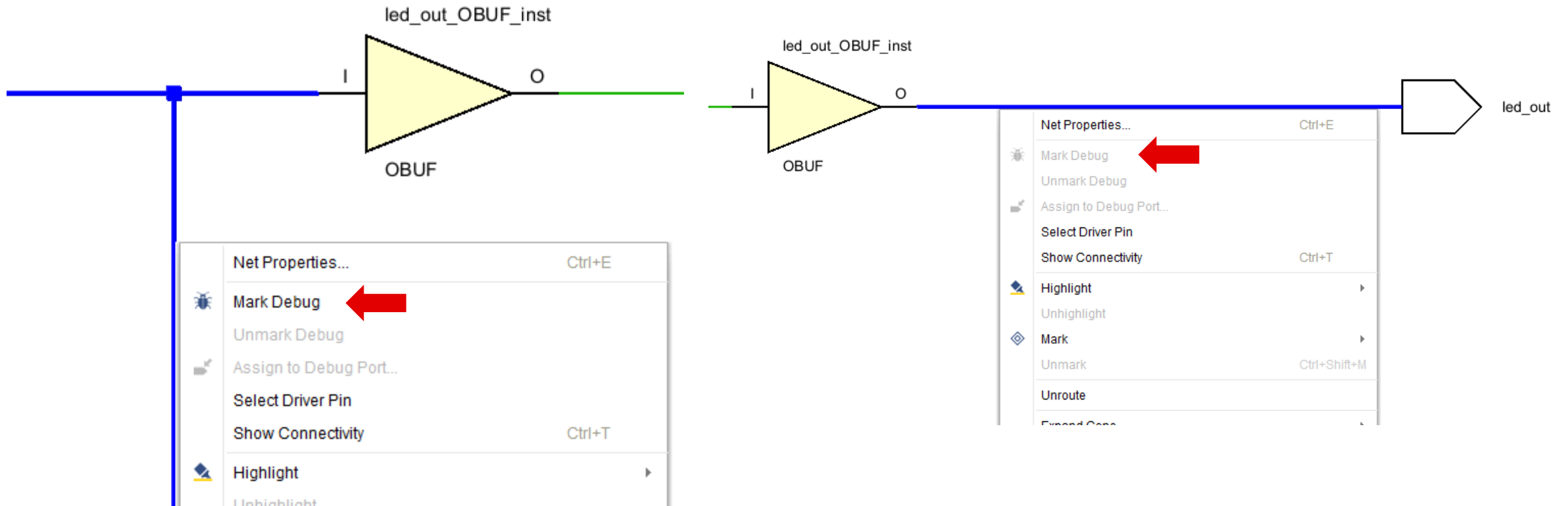
Add debug mark in the netlist

The image shows two windows from the Xilinx Vivado IDE. The left window, titled 'Netlist', displays a tree view of the design hierarchy. Under 'Time_counter', the 'Nets (94)' folder is expanded, and 'counter_reg (26)' is selected. A context menu is open over this net, showing options like 'Bus Net Properties...', 'Mark Debug', 'Unmark Debug', 'Assign to Debug Port...', 'Select Driver Pin', 'Schematic', 'Show Connectivity', 'Show Hierarchy', 'Highlight', 'Unhighlight', 'Mark', and 'Unmark'. The 'Mark Debug' option is highlighted. Below the tree view, the 'Bus Net Properties' dialog is open, showing the 'Name' as 'counter_reg' and the 'Number of nets' as '26'. The 'General' tab is selected, showing 'Scalar Nets'. The right window, titled 'Schematic', shows a circuit diagram with various components and connections. A context menu is open over the schematic, with 'Mark Debug' highlighted. The 'Mark Debug' option is also visible in the 'Bus Net Properties' dialog.

Add debug mark in the netlist



Add debug mark in the netlist



Add debug mark in the netlist

SYNTHESIS

Run Synthesis

Open Synthesized Design

Constraints Wizard

Edit Timing Constraints

Set Up Debug

Report Timing Summary

Report Clock Networks

Report Clock Interaction

Report Methodology

Report DRC

Report Noise

Report Utilization

Report Power

Schematic

Set Up Debug

Nets to Debug

The nets below will be debugged with ILA cores. To add nets click "Find Nets to Add". You can also select nets in the Netlist or other windows, then drag them to the list or click "Add Selected Nets".

Q

⌵

⌶

⌷

⌸

+

-

⚙

Name	Clock Domain	Driver Cell	Probe Type
> counter_reg (26)	sys_clk_IBUF_BUFG	FDRE	Data and Trigger
counter_flag	sys_clk_IBUF_BUFG	FDRE	Data and Trigger
led_out_OBUF	sys_clk_IBUF_BUFG	FDRE	Data and Trigger

Find Nets to Add...

Nets to debug: 28

?

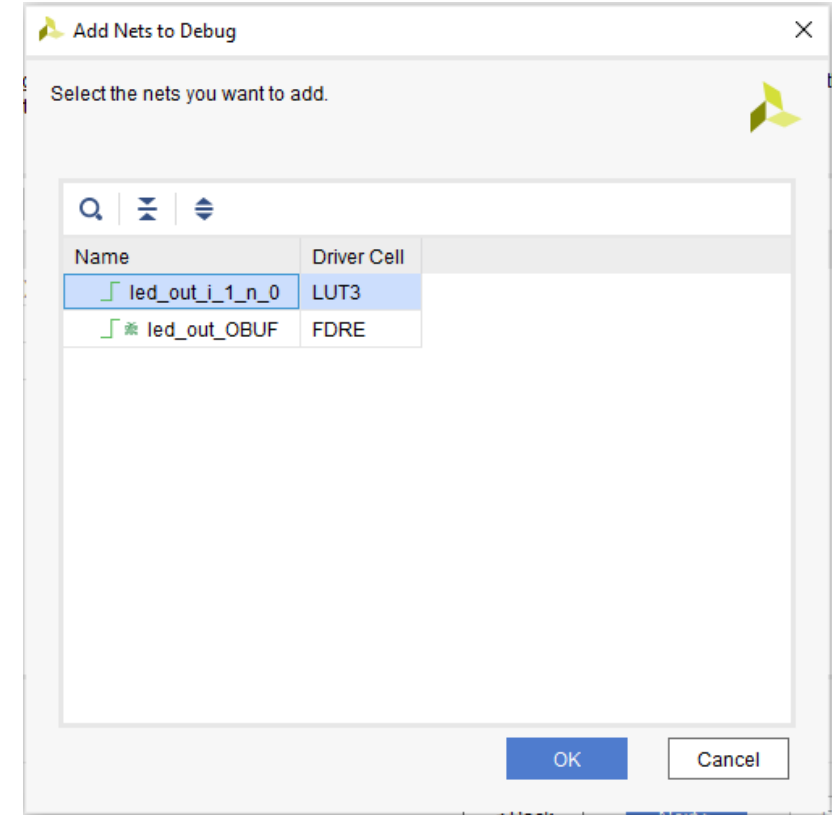
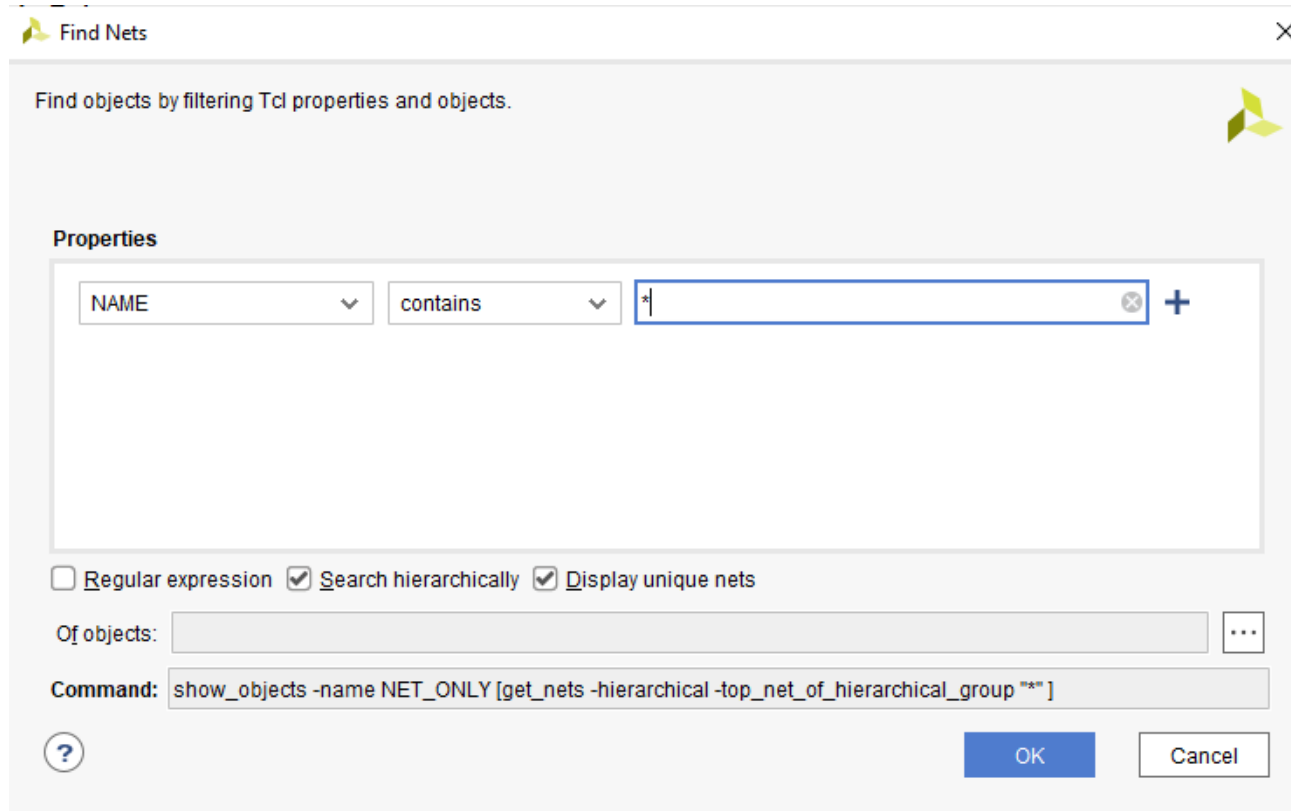
< Back

Next >

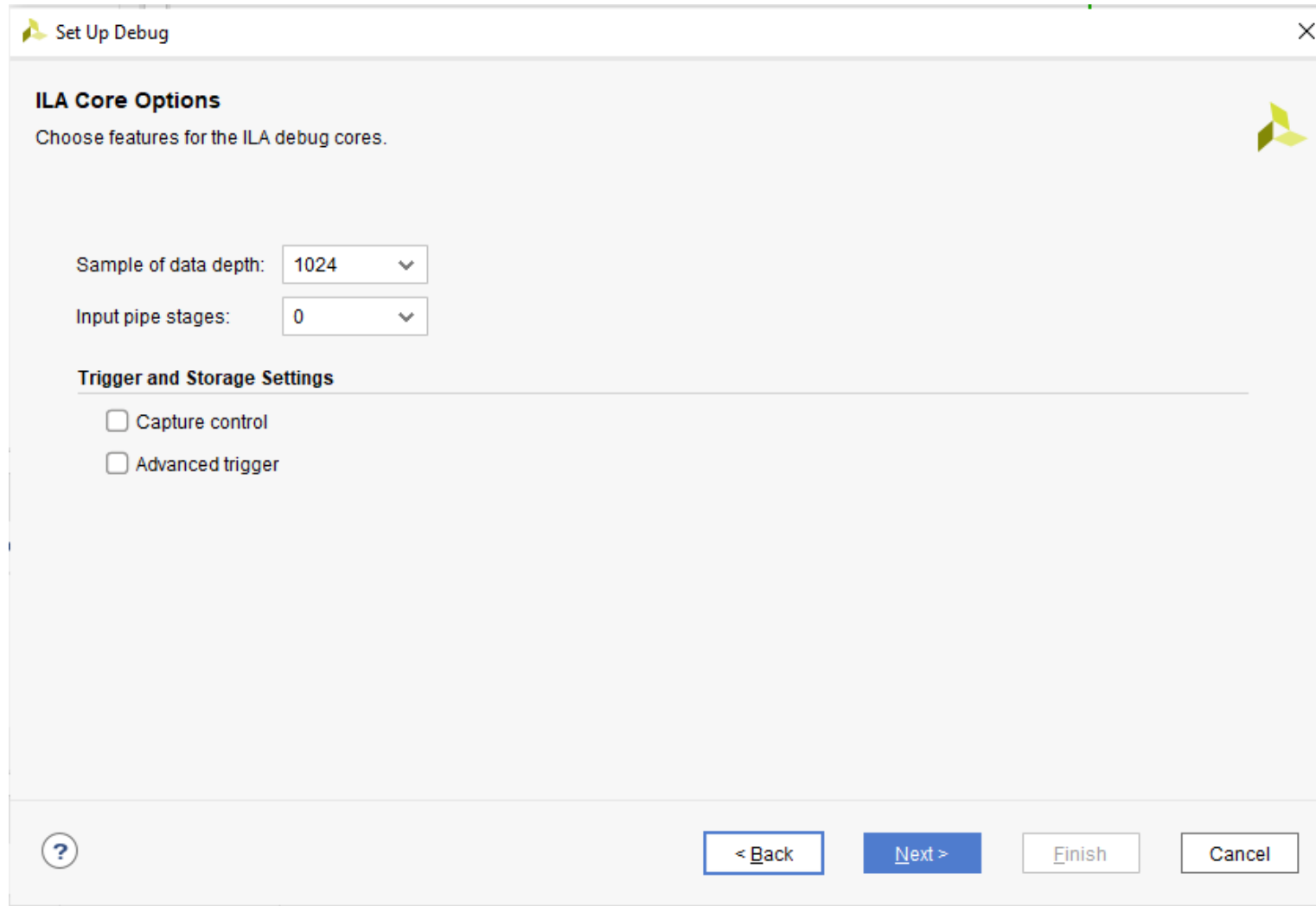
Finish

Cancel

Add debug mark in the netlist



Add debug mark in the netlist



The image shows a 'Set Up Debug' dialog box with a title bar containing a yellow triangle icon and a close button. The main content area is titled 'ILA Core Options' and includes the instruction 'Choose features for the ILA debug cores.' Below this, there are two dropdown menus: 'Sample of data depth:' set to '1024' and 'Input pipe stages:' set to '0'. A section titled 'Trigger and Storage Settings' is separated by a horizontal line and contains two unchecked checkboxes: 'Capture control' and 'Advanced trigger'. At the bottom, there is a help icon (question mark in a circle) on the left and four buttons: '< Back' (disabled), 'Next >' (active), 'Finish' (disabled), and 'Cancel' (disabled).

Set Up Debug

ILA Core Options
Choose features for the ILA debug cores.

Sample of data depth: 1024

Input pipe stages: 0

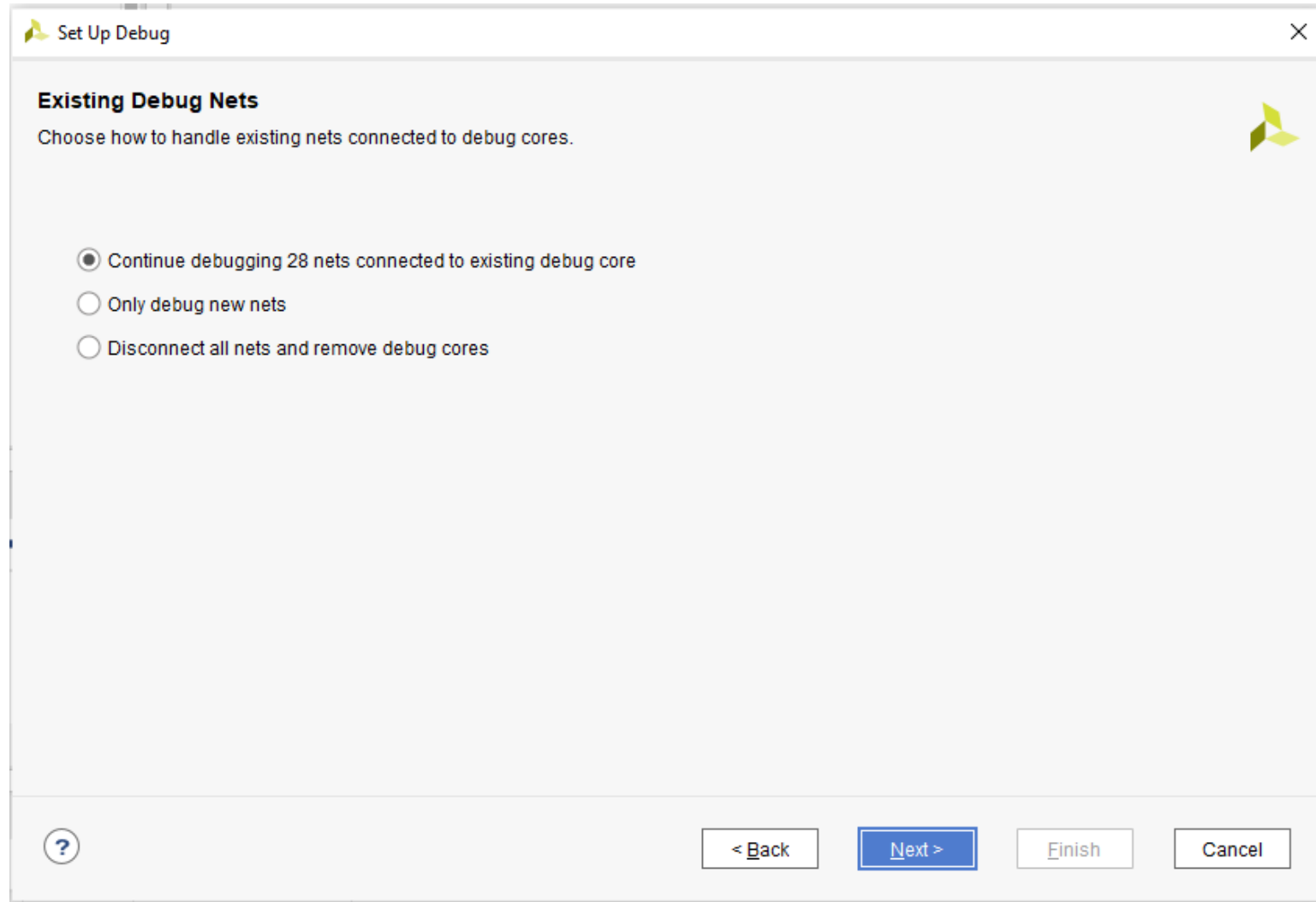
Trigger and Storage Settings

☐ Capture control

☐ Advanced trigger

? < Back Next > Finish Cancel

Add debug mark in the netlist



Add ILA core in the block design

The image shows two windows from the Vivado IDE. The left window is the 'Add IP' search results panel. It has a search bar with 'ILA' entered, showing 4 matches. The results list includes 'ILA (Integrated Logic Analyzer)', 'ILA (Integrated Logic Analyzer with AXIS Interface)', 'Multilayer Video Controller', and 'System ILA'. A red arrow points to 'System ILA'. At the bottom, it says 'ENTER to select, ESC to cancel, Ctrl+Q for IP details'.

The right window is the 'Re-customize IP' dialog for 'System ILA (1.1)'. It has tabs for 'IP Symbol', 'Resources', and 'General Options'. The 'General Options' tab is active, showing a 'Monitor Type' dropdown menu with options 'NATIVE', 'NATIVE', 'INTERFACE', and 'MIX'. The 'Number of Comparators' is set to 1, and 'Input Pipe Stages' is set to 0. There are checkboxes for 'Same Number of Comparators for All Probe Ports', 'Trigger Out Port', 'Trigger In Port', 'Capture Control', and 'Advanced Trigger'. The 'Probe width propagation' is set to 'AUTO'. At the bottom, there are 'OK' and 'Cancel' buttons.

Add ILA core in the block design

To configure more than 64 probe ports use Vivado Tcl Console

General Options

Interface Options

Monitor Type

Monitor Type INTERFACE

Number of Interface Slots 1

Sample Data Depth 1024

☒ Same Number of Comparators for All Probe Ports

Component Name system_ila_0

To configure more than 64 probe ports use Vivado Tcl Console

General Options

Interface Options

Configuration for Slot SLOT0

Interface Type xilinx.com:interface:aximm rtl:1.0

AUTO

AXI-MM ID WidthAUTO

AUTO

AXI-MM Data WidthAUTO

AUTO

AXI-MM Address WidthAUTO

☐ Enable AXI-MM/Stream Protocol Checker

☒ Enable Transaction Tracking Counters

Number Of Outstanding Read Transactions 2

Number Of Outstanding Write Transactions 2

Data and/or Trigger configuration for AXI-MM Interface channel

Read Address

Read Data

Write Address

Write Data

Write Response

☒ Configure AXI Read Address Channel signals as Data

☒ Configure AXI Read Address Channel signals as Trigger

Read Address Channel arid, arvalid and arready signals are configured as Data & Trigger

Add ILA core in the block design

IP Symbol **Resources**

☐ Show disabled ports

+ SLOT_0_AXI
+ SLOT_1_AXI
clk
probe0[0:0]
probe1[0:0]
probe2[0:0]
resetn

Component Name

To configure more than 64 probe ports use Vivado Tcl Console

General Options **Probe_Ports(0..7)** **Interface Options**

Monitor Type

Monitor Type

Number of Probes Native Probe width propagation

Number of Interface Slots

Sample Data Depth

☒ Same Number of Comparators for All Probe Ports

Number of Comparators

☐ Trigger Out Port

☐ Trigger In Port

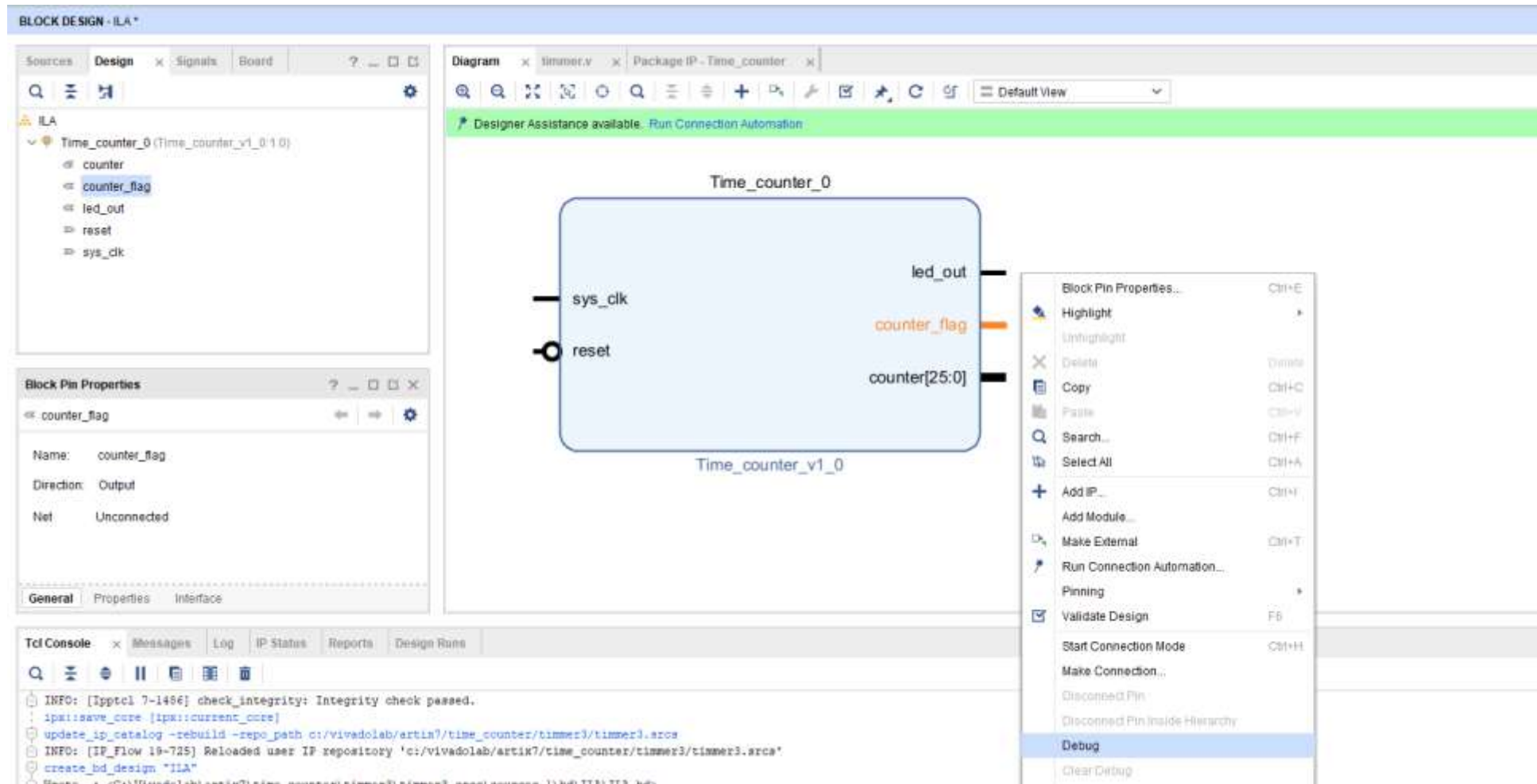
Input Pipe Stages

Trigger And Storage Settings

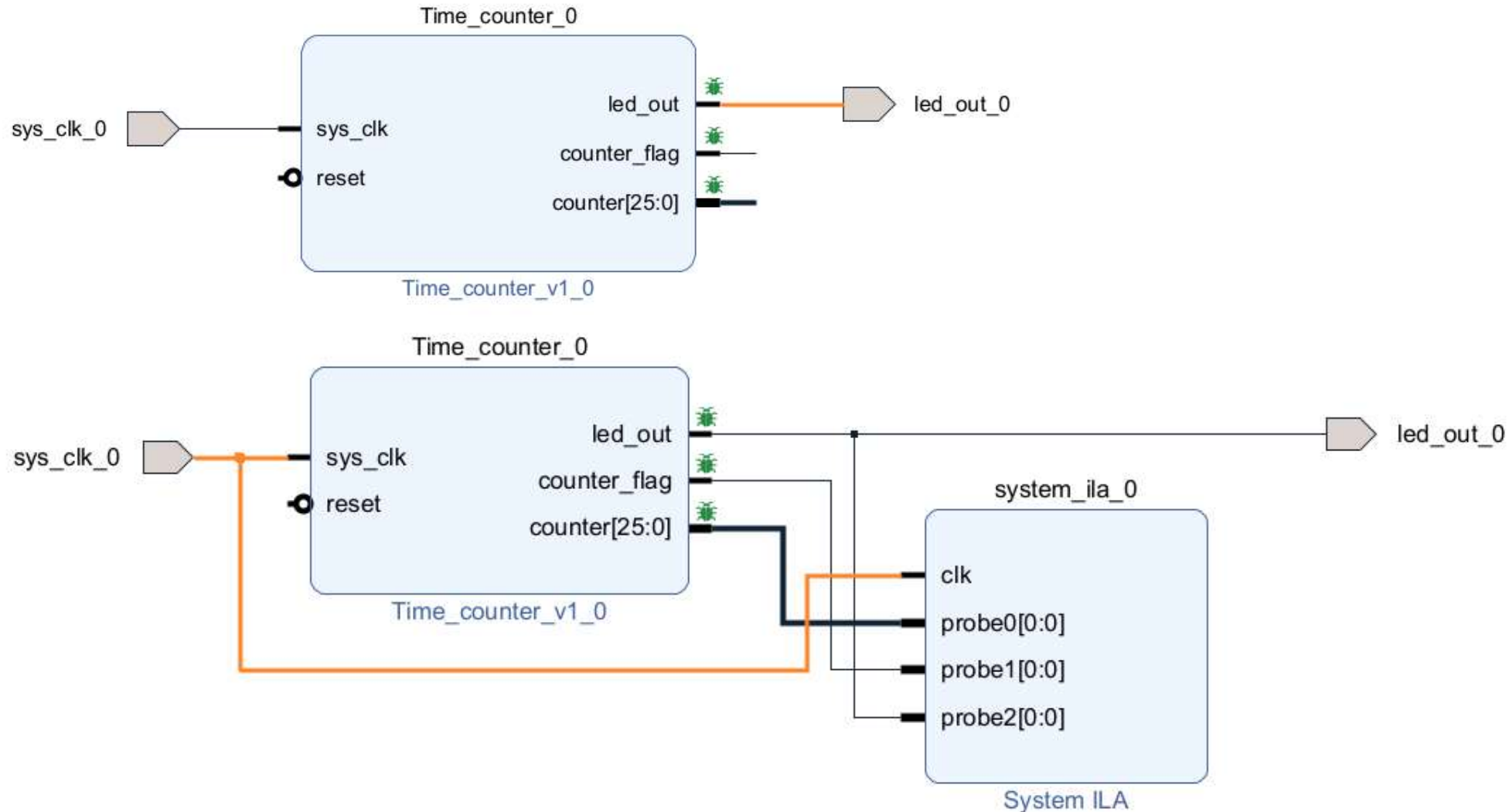
☐ Capture Control

☐ Advanced Trigger

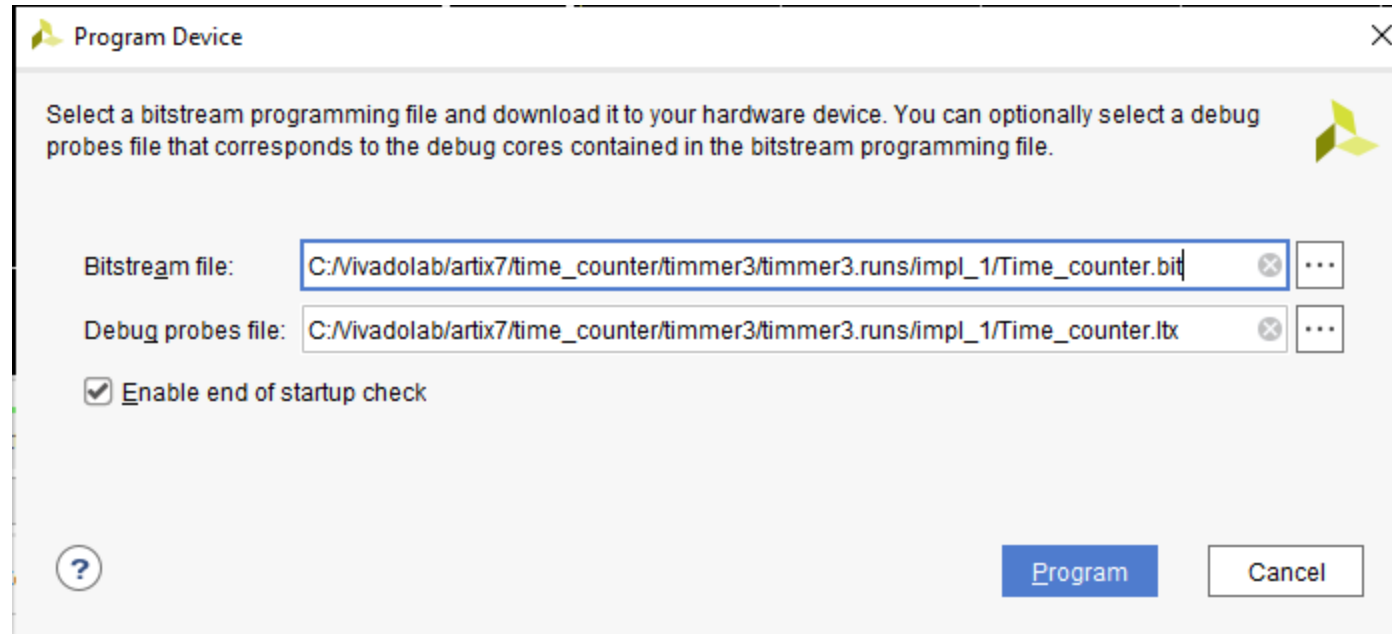
Add ILA core in the block design



Add ILA core in the block design



ILA debugging



ILA debugging

Hardware

localhost (1)
xilinx_tcf/Digilent/210352B47292/ Open
xc7s50_0 (3) Programmed
XADC (System Monitor)
hw_ila_1 (ila) Idle
s25fl128sxxxxx0-spi-x1_x2_x

Properties

timmer.v x hw_ila_1 x

Waveform - hw_ila_1

No content

Hardware

localhost (1)
xilinx_tcf/Digilent/210352B47292/ Open
xc7s50_0 (3) Programmed
XADC (System Monitor)
hw_ila_1 (design_1/system) Idle
s25fl128sxxxxx0-spi-x1_x2_x

ILA Core Properties

hw_ila_1

Name: hw_ila_1

design_1_wrapper.v x hw_ila_1 x

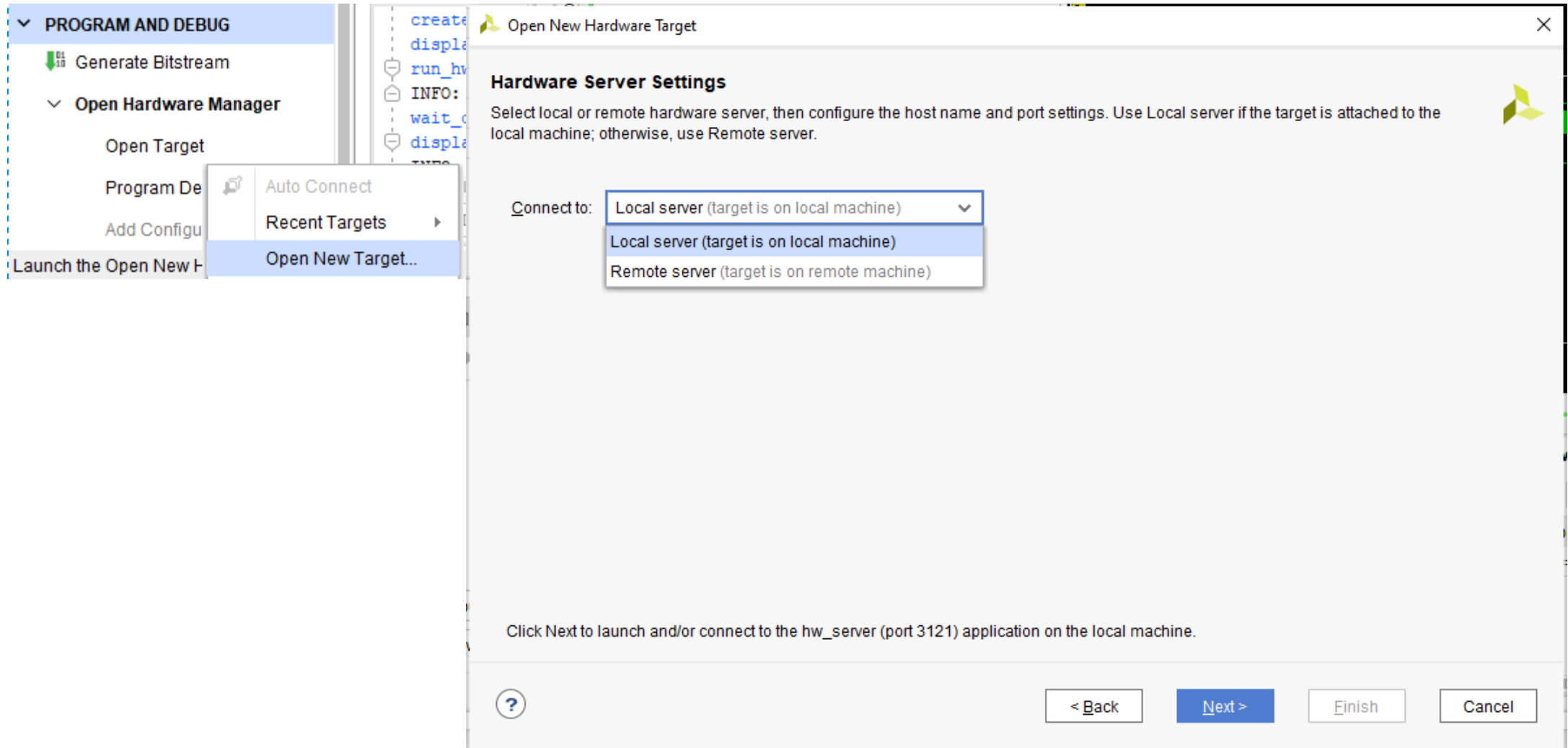
Waveform - hw_ila_1

ILA Status: Idle


Name	Value

Updated at: 2023-Jul-26 09:46:16

ILA debugging




ILA debugging

 Open New Hardware Target ✕

Select Hardware Target
Select a hardware target from the list of available targets, then set the appropriate JTAG clock (TCK) frequency. If you do not see the expected devices, decrease the frequency or select a different target.


Hardware Targets

Type	Name	JTAG Clock Frequency
 xilinx_tcf	Digilent/210352B47292A	3000000 Hz


3000000 Hz
3750000 Hz
5000000 Hz
6000000 Hz
7500000 Hz
10000000 Hz
15000000 Hz
30000000 Hz

Virtual Cable (XVC)

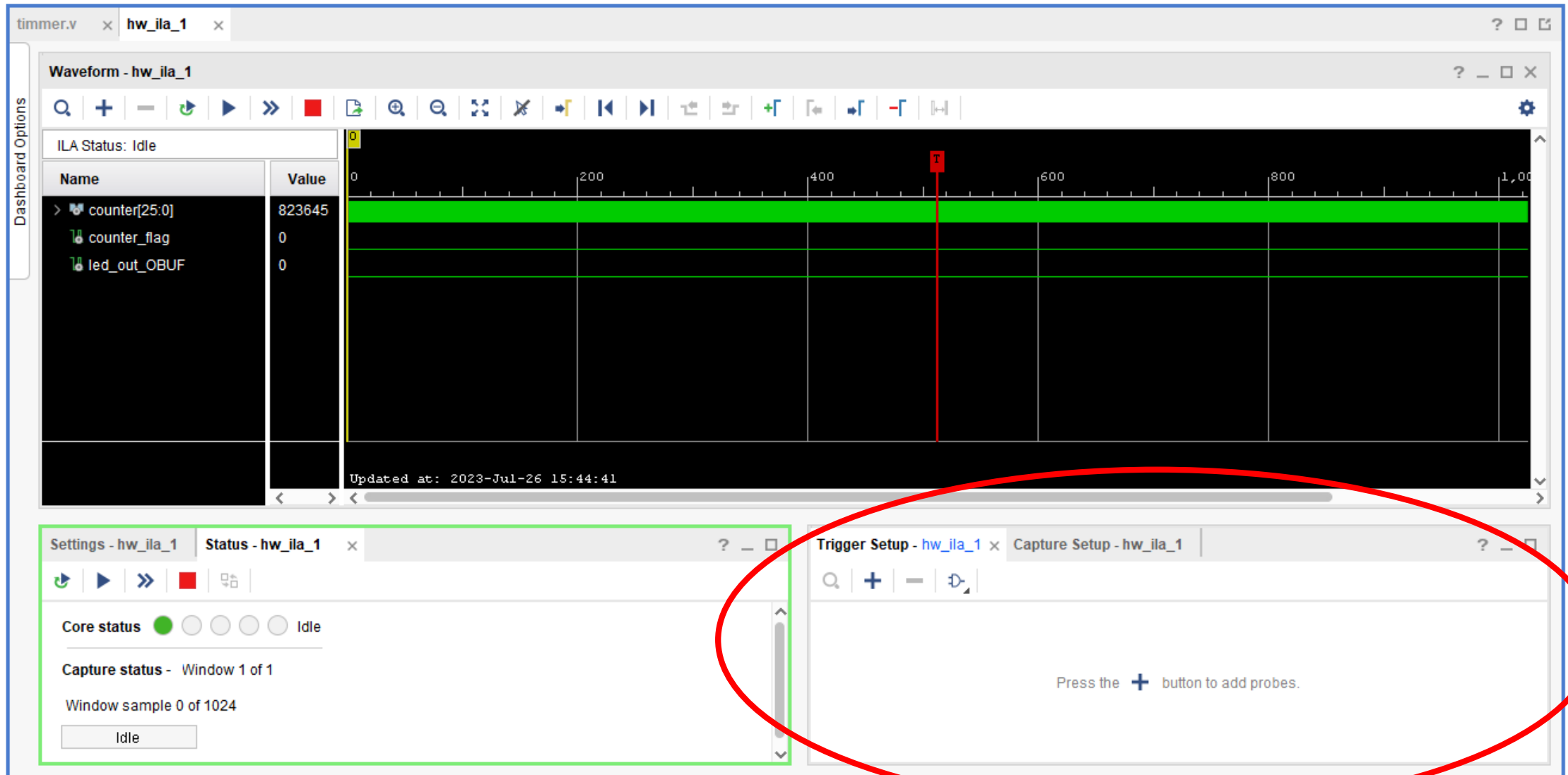
Hardware Devices (for unknown devices, specify the Instruction Register (IR) length)

Name	ID Code	IR Length
 xc7s50_0	0362F093	6

Hardware server: localhost:3121

 < Back Next > Finish Cancel

ILA debugging



ILA debugging

Trigger Setup - hw_ila_1 x Capture Setup - hw_ila_1

Search: Q-

Name	Operator	Radix	Value	Port	Comparator Usage
led_out_OBUF	==	[B] (Binary)	1	probe2[0]	1 of 1

counter[25:0]
counter_flag
led_out_OBUF

OK Cancel

ILA debugging

The screenshot displays the Xilinx ILA (Internal Logic Analyzer) interface for a target device named `hw_ila_1`. The main window shows a waveform titled "Waveform - hw_ila_1" with a time axis from 508 to 517. A red vertical line marks the current time at 512. The waveform shows a signal that is 0 at 512 and 513, and then transitions to 1 at 514. The signal is labeled `led_out_OBUF`.

On the left, the "Dashboard Options" panel shows the "ILA Status: Waiting For Trigger (512 out of 1024)". Below this, a table lists the variables being monitored:

Name	Value
<code>counter[25:0]</code>	0
<code>counter_flag</code>	0
<code>led_out_OBUF</code>	0

At the bottom left, the "Settings - hw_ila_1" and "Status - hw_ila_1" panels are visible. The "Status" panel shows the "Core status" as "Waiting for Trigger" with a green progress bar at 50%. The "Capture status" is "Window 1 of 1" with a sample rate of 512 of 1024.

At the bottom right, the "Trigger Setup - hw_ila_1" and "Capture Setup - hw_ila_1" panels are visible. The "Trigger Setup" panel shows a table with the following data:

Name	Operator	Radix	Value	Port	Condition
<code>led_out_OBUF</code>	<code>==</code>	[B]	1	<code>probe2[0]</code>	1

ILA debugging

The screenshot displays the Xilinx ILA (Internal Logic Analyzer) interface for a target device. The main window shows a waveform for the signal `counter[25:0]`, which is currently at value 1249489. The waveform is captured at a rate of 1024 samples per window. The status of the ILA is Idle.

The waveform data is as follows:

Sample	Value
508	1249997
509	1249998
510	1249999
511	1250000
512	0
513	1
514	2
515	3
516	4
517	5

The bottom left panel shows the **Settings - hw_ila_1** and **Status - hw_ila_1** tabs. The **Core status** is Idle. The **Capture status** is Window 1 of 1. The **Window sample 0 of 1024** is Idle. The bottom right panel shows the **Trigger Setup - hw_ila_1** and **Capture Setup - hw_ila_1** tabs. The **Trigger Setup** shows a trigger on `led_out_OBUF` with the operator `==` and radix `[B]`, with a value of 1. The **Capture Setup** shows a capture on `probe2[0]` with a value of 1.

ILA IP Setting

Re-customize IP

ILA (Integrated Logic Analyzer) (6.2)

Documentation IP Location Switch to Defaults

☐ Show disabled ports

Component Name

To configure more than 64 probe ports use Vivado Tcl Console

General Options Probe_Ports(0..2)

Monitor Type

☒ Native ☐ AXI

Number of Probes [1...512]

Sample Data Depth

☒ Same Number of Comparators for All Probe Ports

Number of Comparators

☒ Trigger Out Port

☒ Trigger In Port

Input Pipe Stages

Trigger And Storage Settings

☒ Capture Control

☒ Advanced Trigger

GUI configuration mode is limited to 64 probe ports.

Component Name

To configure more than 64 probe ports use Vivado Tcl Console

General Options Probe_Ports(0..2)

Probe Port	Probe Width [1..4096]	Number of Comparators	Probe Trigger or Data
PROBE0	<input type="text" value="26"/>	<input type="text" value="2"/>	DATA AND TRIGGER
PROBE1	<input type="text" value="1"/>	<input type="text" value="16"/>	DATA AND TRIGGER
PROBE2	<input type="text" value="1"/>	<input type="text" value="2"/>	DATA AND TRIGGER

OK Cancel

ILA IP Setting

The image displays two side-by-side screenshots of the Xilinx ILA (Integrated Logic Analyzer) tool, showing waveform captures and their corresponding settings.

Left Screenshot (Waveform - hw_ila_1):

- ILA Status: Idle
- Waveform capture shows a green bar representing the counter value, with a red vertical line indicating the trigger point at 2,000 clock cycles (circled in red).
- Updated at: 2023-Jul-26 16:33:14

Right Screenshot (Waveform - hw_ila_1):

- ILA Status: Idle
- Waveform capture shows a green bar representing the counter value, with a red vertical line indicating the trigger point at 8,000 clock cycles (circled in red).
- Updated at: 2023-Jul-26 16:41:23

Settings - hw_ila_1 (Left):

- Capture Mode Settings
- Capture mode: BASIC
- Number of windows: 1 [1 - 8192]
- Window data depth: 2048 [1 - 8192] (highlighted with a red arrow)
- Trigger position in window: 512 [0 - 2047]

Settings - hw_ila_1 (Right):

- Capture Mode Settings
- Capture mode: BASIC
- Number of windows: 1 [1 - 8192]
- Window data depth: 8192 [1 - 8192] (highlighted with a red arrow)
- Trigger position in window: 4096 [0 - 8191]

Trigger Panel (Right):

- Trigger Name: counter

ILA IP Setting

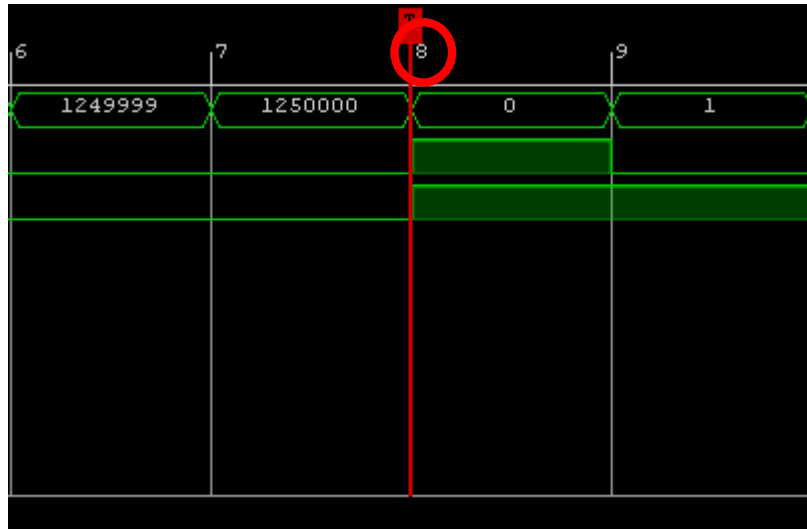
The screenshot displays the Xilinx ILA (Internal Logic Analyzer) interface. The top window, titled "Waveform - hw_ila_1", shows a digital waveform with three signals: `counter[25:0]`, `counter_flag`, and `led_out_OBUF`. The `counter[25:0]` signal is a green horizontal bar at a high level. The `counter_flag` signal is a green horizontal bar at a low level. The `led_out_OBUF` signal is a green horizontal bar at a high level. The waveform is updated at 2023-Jul-26 16:46:56. The left sidebar shows the "ILA Status: Idle" and a table of signal values:

Name	Value
<code>counter[25:0]</code>	1249490
<code>counter_flag</code>	0
<code>led_out_OBUF</code>	1

The bottom-left window, titled "Settings - hw_ila_1", shows the "Capture Mode Settings" tab. A red arrow points to the "Number of windows" field, which is set to 2. The "Window data depth" is set to 1024, and the "Trigger position in window" is set to 511. The bottom-right window, titled "Trigger Setup - hw_ila_1", shows the "Capture Setup" tab. It contains a table with the following data:

Name	Operator	Radix	Value	Port	Comparator U
<code>counter_flag</code>	<code>==</code>	[B]	1	<code>probe1[0]</code>	1 of 15

ILA IP Setting



Settings - hw_ila_1 × Status - hw_ila_1

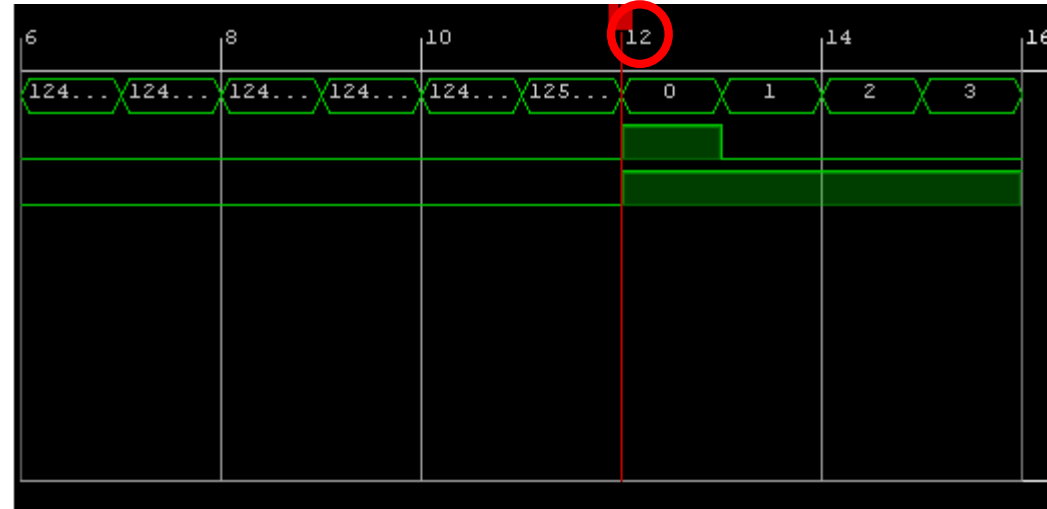
Number of windows: 1 [1 - 8192]

Window data depth: 16 [1 - 8192]

Trigger position in window: 8 [0 - 15]

General Settings

Refresh rate: 500 ms



Settings - hw_ila_1 × Status - hw_ila_1

Number of windows: 1 [1 - 8192]

Window data depth: 16 [1 - 8192]

Trigger position in window: 12 [0 - 15]

General Settings

Refresh rate: 500 ms

ILA IP Setting

Trigger Setup - hw_ila_1 x Capture Setup - hw_ila_1

Q + - ↻

Name	Operator	Radix	Value	Port	Comparator Usage
counter_flag	==	[B]	0	probe1[0]	2 of 15
counter_flag	==	[B]	1	probe1[0]	1 of 15
led_out_OBUF	==	[B]	1	probe2[0]	1 of 1
led_out_OBUF	==	[B]	R	probe2[0]	Out of comparators

Trigger Setup - hw_ila_1 x Capture Setup - hw_ila_1

Q + - ↻

Set Trigger Condition to Global AND'

✓ Set Trigger Condition to 'Global OR'

Set Trigger Condition to Global NAND'

Set Trigger Condition to 'Global NOR'

Name	Operator	Radix	Value	Port	Comparator Usage
counter_flag	==	[B]	0	probe1[0]	2 of 15
counter_flag	==	[B]	1	probe1[0]	1 of 15
led_out_OBUF	==	[B]	1	probe2[0]	1 of 1
led_out_OBUF	==	[B]	R	probe2[0]	Out of comparators

ILA IP Setting

Settings - hw_ila_1 x Status - hw_ila_1

Trigger Mode Settings

Trigger mode: BASIC_ONLY v

Capture Mode Settings

Capture mode: BASIC v

Number of windows: ALWAYS - 8192

Window data depth: 1024 [1 - 8192]

Trigger position in window: 512 [0 - 1023]

General Settings

Refresh rate: 500 ms

Trigger Setup - hw_ila_1 Capture Setup - hw_ila_1 x ? _ □

Q + - ↺

Press the + button to add probes.

Trigger Setup - hw_ila_1 Capture Setup - hw_ila_1 x

Q + - ↺

Name	Operator	Radix	Value	Port
counter_flag	== v	[B] v	1 v	probe1[0]

Add Probes x

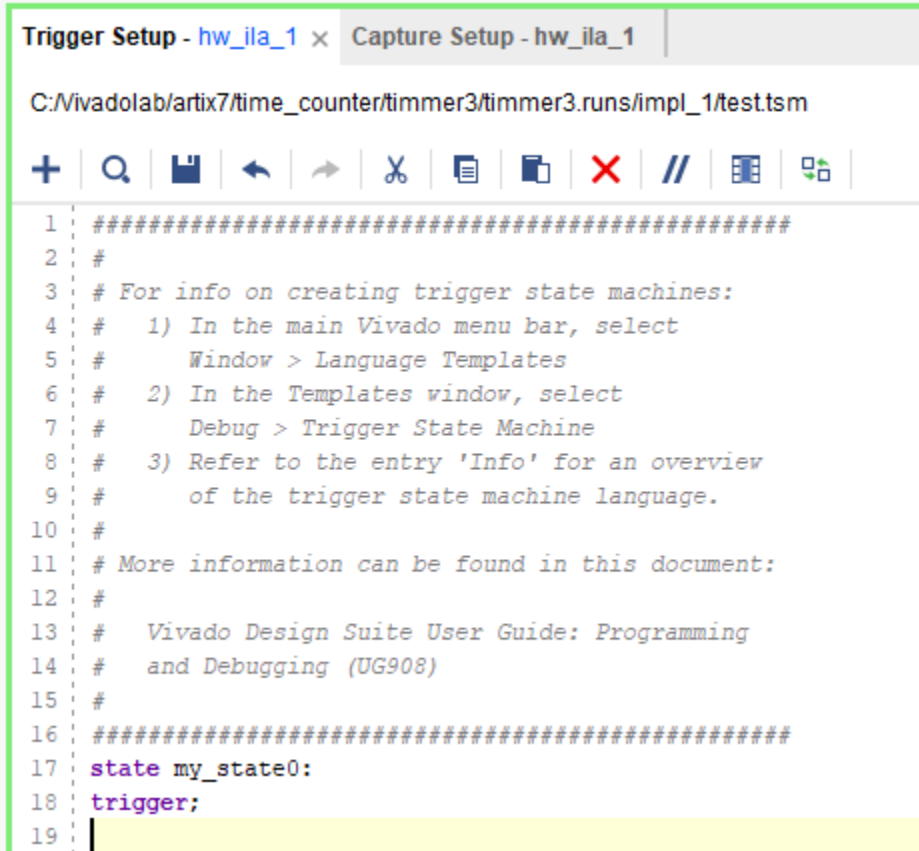
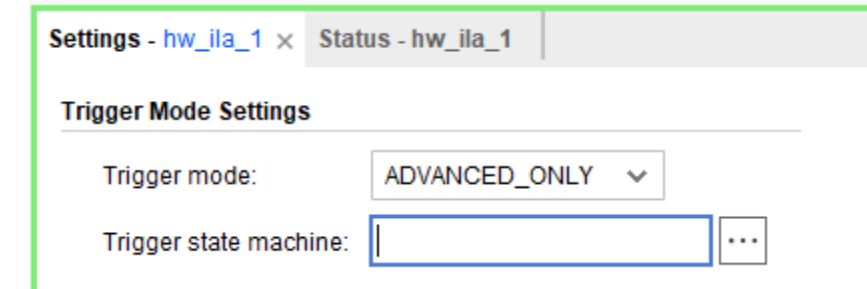
Search: Q-

counter[25:0]

led_out_OBUF

OK Cancel

ILA IP Setting

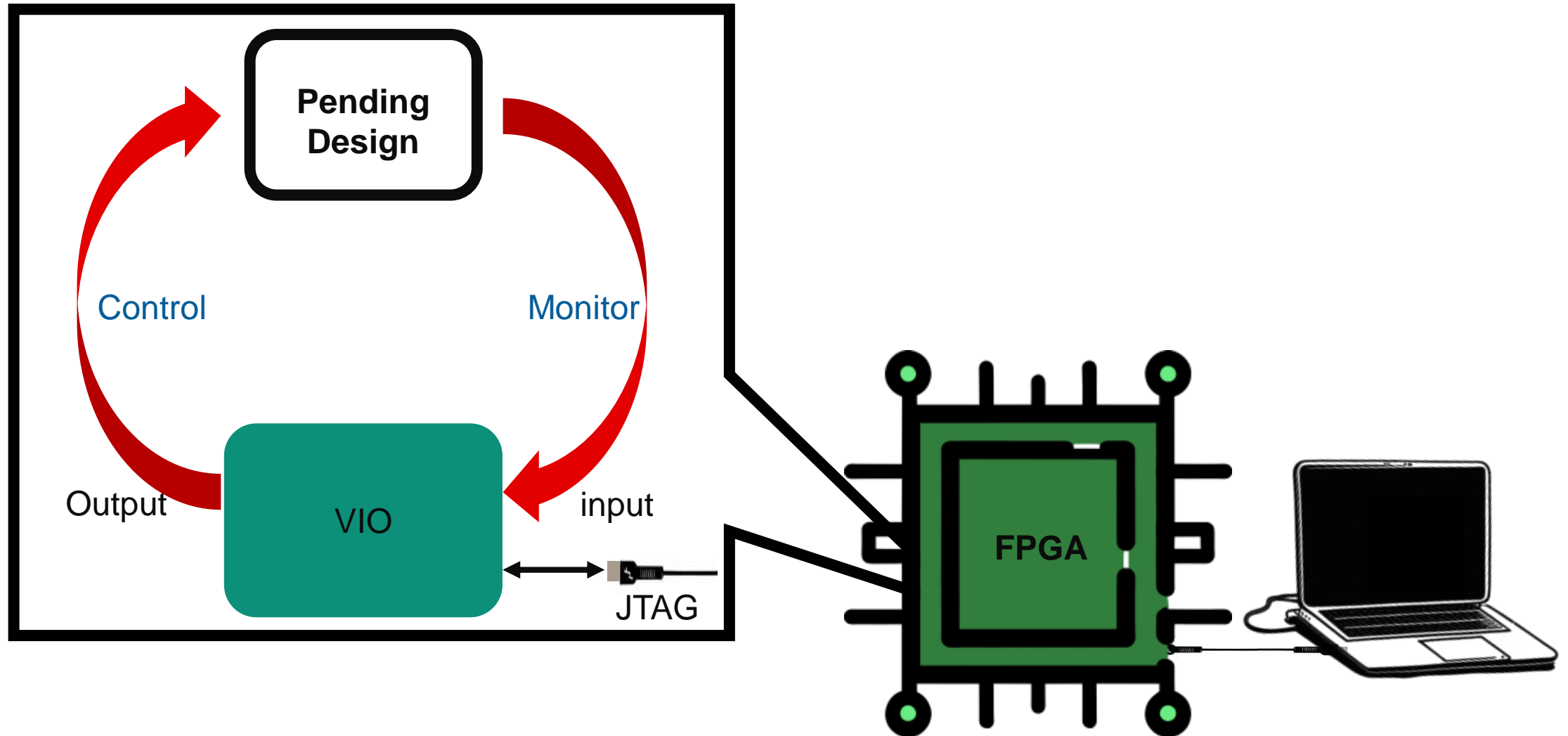




Virtual Input/Output

Course Agenda
2023

Virtual Input/Output



Add VIO IP Core

The screenshot displays the Vivado IDE interface with the 'Customize IP' window open for the 'VIO (Virtual Input/Output) (3.0)' core. The left pane shows the 'IP Catalog' with a search for 'VIO' and the 'VIO (Virtual Input/Output)' core selected. The right pane shows the configuration options for the VIO core.

Project Summary | **timmer.v** | **IP Catalog** | **Customize IP**

Cores | **Interfaces**

Search: (2)

Name

- Vivado Repository
 - Debug & Verification
 - Debug
 - VIO (Virtual Input/Output)**
 - VIO (Virtual Input/Output with AXIS Interface)

Details

Name: **VIO (Virtual Input/Output)**
Version: 3.0 (Rev. 23)
Description: The Virtual Input/Output (VIO) core is a customizable in size to interface with the FPGA. The configuration options for your design are also applied to the component.
Status: **Production**
License: Included

VIO (Virtual Input/Output) (3.0)

☐ Show disabled ports

Component Name:

To configure more than 64 probe ports use Vivado Tcl Console

General Options | **PROBE_IN Ports(0..0)** | **PROBE_OUT Ports(0..0)**

Input Probe Count: [0 - 256]
Output Probe Count: [0 - 256]
☒ Enable Input Probe Activity Detectors

Diagram: A block diagram showing the VIO core with inputs 'ck' and 'probe_in[0:0]' and output 'probe_out[0:0]'.

Buttons: OK, Cancel

Add VIO IP Core

Component Name

To configure more than 64 probe ports use Vivado Tcl Console

General Options | PROBE_IN Ports(0..2) | PROBE_OUT Ports(0..0)

Probe Port	Probe Width [1 - 256]
PROBE_IN0	<input type="text" value="26"/>
PROBE_IN1	<input type="text" value="1"/>
PROBE_IN2	<input type="text" value="1"/>

Component Name

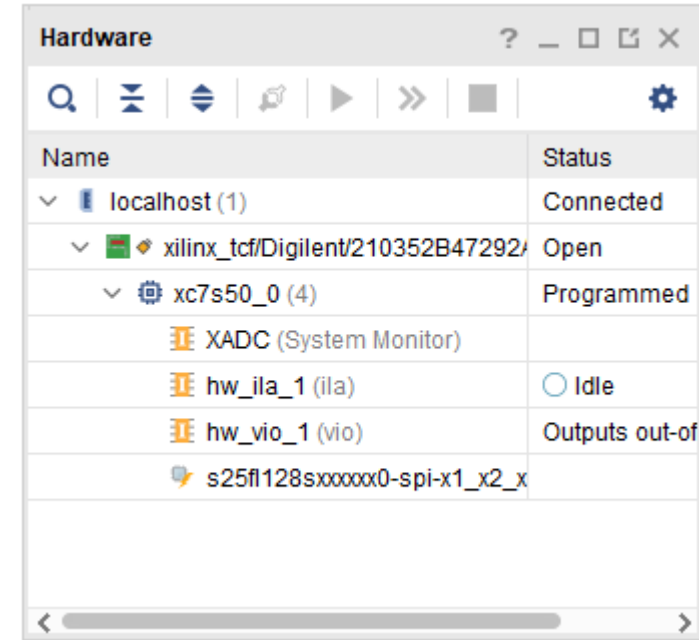
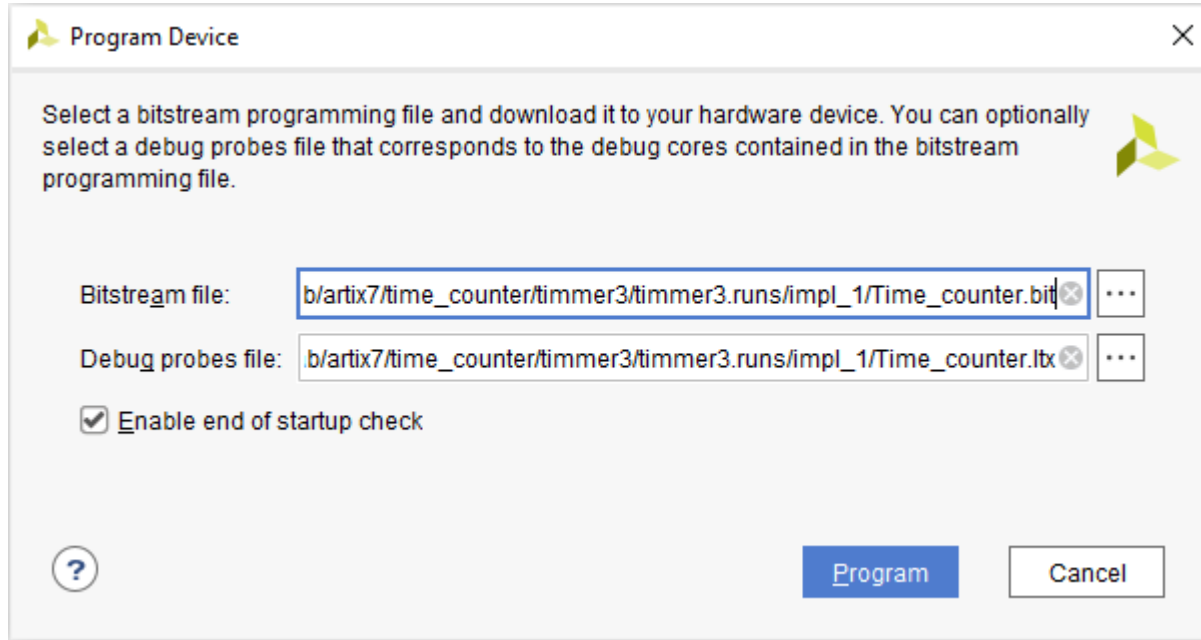
To configure more than 64 probe ports use Vivado Tcl Console

General Options | PROBE_IN Ports(0..2) | PROBE_OUT Ports(0..0)

Probe Port	Probe Width [1 - 256]	Initial Value (in hex)
PROBE_OUT0	<input type="text" value="1"/>	<input type="text" value="0x0"/>

```
vio_0 vio(  
    .clk(sys_clk),  
    .probe_out0(vbtn),  
    .probe_in0(counter),  
    .probe_in1(counter_flag),  
    .probe_in2(led_out)  
);
```


VIO Debugging



VIO Debugging

The screenshot displays the Vivado IDE interface for VIO debugging. The main window shows a table of VIO components for `hw_vio_1`. A red circle highlights the '+' button in the toolbar, indicating the action to add a new probe. An 'Add Probes' dialog is open, showing a search bar and a list of probes for `hw_vio_1`, including `led_out_OBUF`.

Name	Value	Activity	Direction	VIO
<code>vbtn</code>	<code>[B] 0</code>		Output	<code>hw_vio_1</code>
<code>> counter[25:0]</code>	<code>[U] 0</code>		Input	<code>hw_vio_1</code>
<code>counter_flag</code>			Input	<code>hw_vio_1</code>

Add Probes

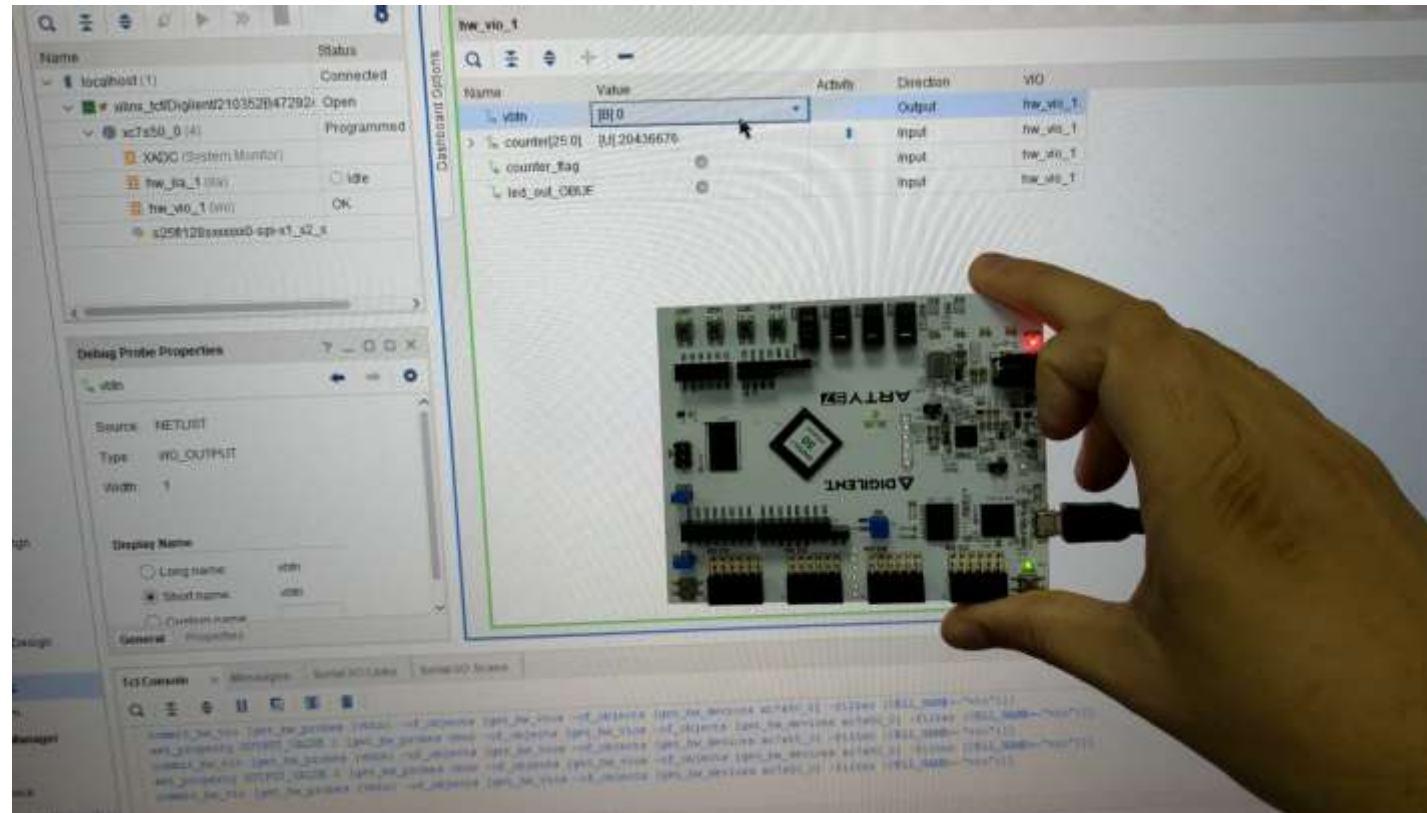
Search:

Probes for `hw_vio_1` (1)

- `hw_vio_1`
 - `led_out_OBUF`

OK Cancel

VIO Debugging



VIO Debugging

The screenshot displays the Xilinx Vivado IDE interface for VIO debugging. On the left, the 'hw_vio_1' component is shown with a table of variables. A context menu is open over the 'vbtn' variable, showing options like 'Toggle Button'. On the right, the 'dashboard_1' window is shown with a similar table, but with an additional 'VIO' column.

hw_vio_1 Table:

Name	Value	Activity	Direction
vbtn	[B] 0		Output
counter[25:0]	[U] 0		
counter_flag			
led_out_OBUF			

Context Menu Options:

- Debug Probe Properties...
- Text
- Active-High Button
- Active-Low Button
- Toggle Button**
- Radix
- Rename...
- Name
- Remove
- Export to Spreadsheet...

dashboard_1 Table:

Name	Value	Activity	Direction	VIO
vbtn	[B] 0		Output	hw_vio_1
counter[25:0]	[U] 0		Input	hw_vio_1
counter_flag			Input	hw_vio_1
led_out_OBUF			Input	hw_vio_1

VIO Debugging

Trigger Setup - hw_ila_1 hw_vio_1 x Capture Setup - hw_ila_1

Q + -

Name	Value	Activity	Direction	VIO
> counter[25:0]	[U] 866794		Input	hw_vio_1
long_flag	[B] 1		Input	hw_vio_1
short_flag	[B] 1		Input	hw_vio_1
led_on			Input	hw_vio_1
count			Input	hw_vio_1
vbtn			Output	hw_vio_1

Debug Probe Properties... Ctrl+E

- Text
- LED...
- Radix
- Rename...
- Name
- Remove Delete
- Export to Spreadsheet...

Select LED Colors

Low Value Color: Gray v

High Value Color: Green v

OK

Gray

Red

Green

Blue

Trigger Setup - hw_ila_1 hw_vio_1 x Capture Setup - hw_ila_1

Q + -

Name	Value	Activity	Direction	VIO
▼ counter[25:0]	[U] 409291		Input	hw_vio_1
└ counter[18]			Input	hw_vio_1
└ counter[17]			Input	hw_vio_1
└ counter[13]			Input	hw_vio_1
└ counter[12]			Input	hw_vio_1
└ counter[11]			Input	hw_vio_1
└ counter[10]			Input	hw_vio_1
└ counter[9]			Input	hw_vio_1
└ counter[7]			Input	hw_vio_1
└ counter[6]			Input	hw_vio_1
└ counter[3]			Input	hw_vio_1
└ counter[1]			Input	hw_vio_1
└ counter[0]			Input	hw_vio_1
└ counter[25]			Input	hw_vio_1
└ counter[24]			Input	hw_vio_1
└ counter[23]			Input	hw_vio_1
└ counter[22]			Input	hw_vio_1
└ counter[21]			Input	hw_vio_1
└ counter[20]			Input	hw_vio_1
└ counter[19]			Input	hw_vio_1
└ counter[16]			Input	hw_vio_1
└ counter[15]			Input	hw_vio_1

VIO Debugging

The image displays two screenshots of the Xilinx ILS (Integrated Logic Simulator) interface, specifically the 'Trigger Setup' and 'Capture Setup' windows for the 'hw_ila_1' target.

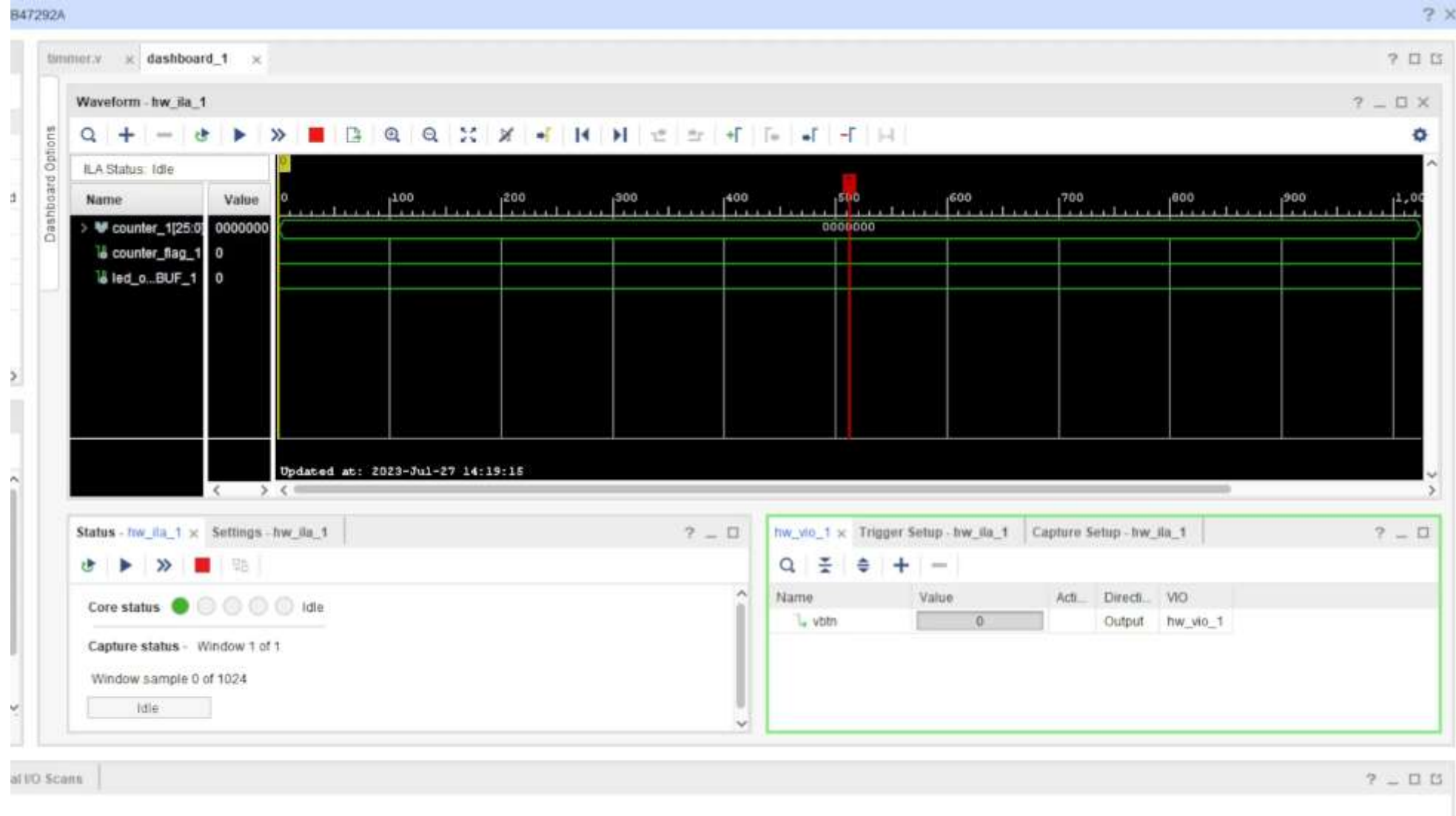
Left Screenshot (Trigger Setup - hw_ila_1):

- The 'hw_vio_1' tab is active.
- The 'Trigger Setup' window shows a list of signals: 'counter[25:0]', 'long_flag', 'led_out_OBUF', and 'vbtn'.
- The 'counter[25:0]' signal is selected, and a context menu is open.
- The context menu options include: 'Debug Probe Properties... (Ctrl+E)', 'Text', 'LED...', 'Radix', 'Activity Persistence', 'Rename...', 'Name', 'Remove (Delete)', and 'Export to Spreadsheet...'.
- The 'Activity Persistence' option is highlighted, and its sub-menu is open, showing: 'Infinite' (selected), 'Long (80 Samples)', and 'Short (8 Signals)'.

Right Screenshot (Capture Setup - hw_ila_1):

- The 'hw_vio_1' tab is active.
- The 'Capture Setup' window shows a list of signals: 'long_flag' and 'vbtn'.
- The 'vbtn' signal is selected, and its value is shown as 0.
- The 'vbtn' signal is configured as an 'Output' signal for 'hw_vio_1'.

VIO Debugging



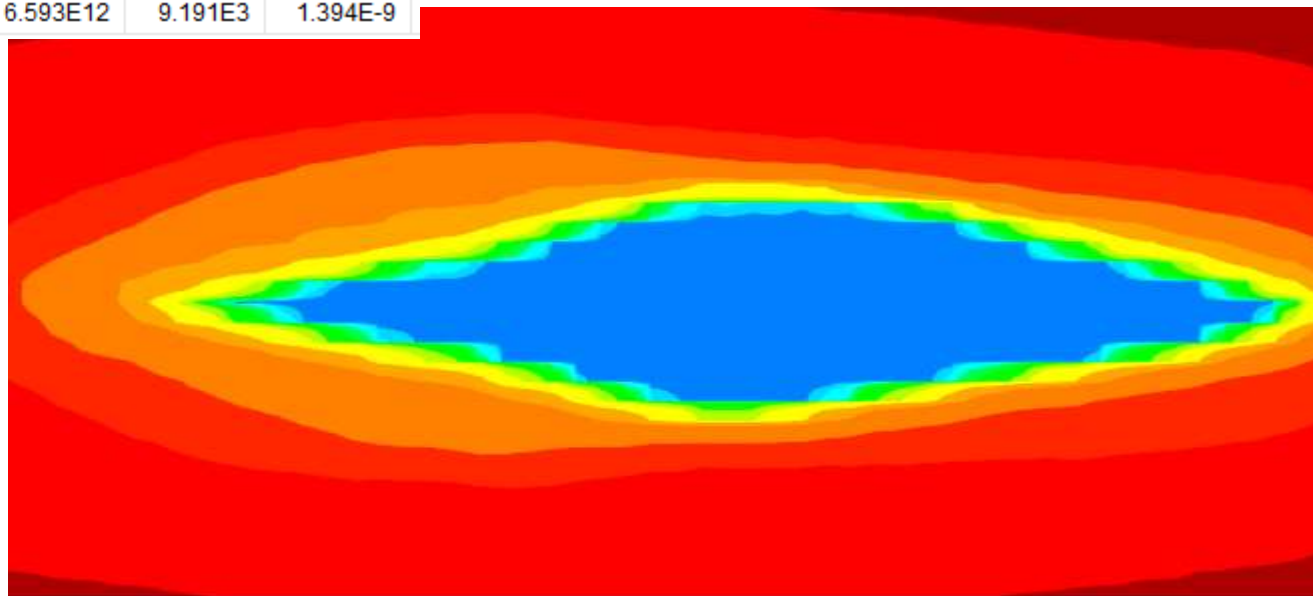


Integrated Bit Error Ratio Tester

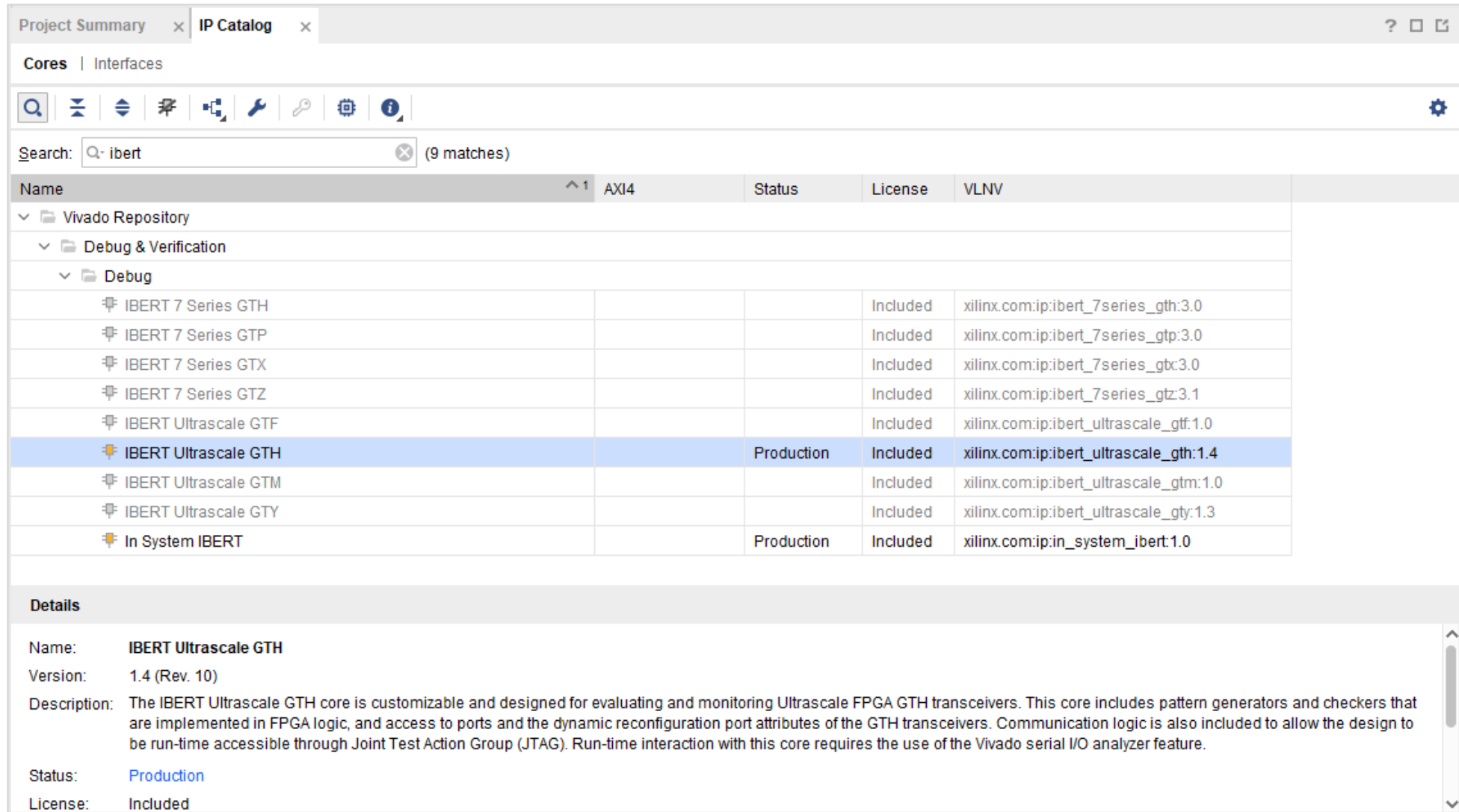
Course Agenda
2023

IBERT

Name	TX	RX	Status	Bits	Errors	BER
Link 12	MGT_X0Y12/TX	MGT_X0Y12/RX	15.625 Gbps	6.592E12	1.78E2	2.7E-11
Link 13	MGT_X0Y13/TX	MGT_X0Y13/RX	15.599 Gbps	6.592E12	2.595E4	3.936E-9
Link 14	MGT_X0Y14/TX	MGT_X0Y14/RX	15.625 Gbps	6.592E12	3.717E3	5.638E-10
Link 15	MGT_X0Y15/TX	MGT_X0Y15/RX	15.625 Gbps	6.592E12	3E4	4.551E-9
Link 16	MGT_X0Y16/TX	MGT_X0Y16/RX	15.646 Gbps	6.592E12	1.595E6	2.42E-7
Link 17	MGT_X0Y17/TX	MGT_X0Y17/RX	15.633 Gbps	3.681E12	3.007E11	8.168E-2
Link 18	MGT_X0Y18/TX	MGT_X0Y18/RX	15.625 Gbps	6.593E12	6.4E1	9.708E-12
Link 19	MGT_X0Y19/TX	MGT_X0Y19/RX	15.643 Gbps	6.593E12	9.191E3	1.394E-9



IBERT IP Setting



Project Summary x IP Catalog x

Cores | Interfaces

Search: (9 matches)

Name	AXI4	Status	License	VLNV
Vivado Repository				
Debug & Verification				
Debug				
IBERT 7 Series GTH			Included	xilinx.com:ip:ibert_7series_gth:3.0
IBERT 7 Series GTP			Included	xilinx.com:ip:ibert_7series_gtp:3.0
IBERT 7 Series GTX			Included	xilinx.com:ip:ibert_7series_gtx:3.0
IBERT 7 Series GTZ			Included	xilinx.com:ip:ibert_7series_gtz:3.1
IBERT Ultrascale GTF			Included	xilinx.com:ip:ibert_ultrascale_gtf:1.0
IBERT Ultrascale GTH		Production	Included	xilinx.com:ip:ibert_ultrascale_gth:1.4
IBERT Ultrascale GTM			Included	xilinx.com:ip:ibert_ultrascale_gtm:1.0
IBERT Ultrascale GTY			Included	xilinx.com:ip:ibert_ultrascale_gty:1.3
In System IBERT		Production	Included	xilinx.com:ip:in_system_ibert:1.0

Details

Name: **IBERT Ultrascale GTH**

Version: 1.4 (Rev. 10)

Description: The IBERT Ultrascale GTH core is customizable and designed for evaluating and monitoring Ultrascale FPGA GTH transceivers. This core includes pattern generators and checkers that are implemented in FPGA logic, and access to ports and the dynamic reconfiguration port attributes of the GTH transceivers. Communication logic is also included to allow the design to be run-time accessible through Joint Test Action Group (JTAG). Run-time interaction with this core requires the use of the Vivado serial I/O analyzer feature.

Status: **Production**

License: **Included**

IBERT IP Setting

Customize IP

IBERT 7 Series GTX (3.0)

Documentation IP Location Switch to Defaults

☐ Show disabled ports

Component Name: `ibert_7series_gtx_0`

Protocol Definition | Protocol Selection | Clock Settings | Summary

Silicon Version

Silicon Version: General ES/Production

The maximum number of quads available for this device is 4

Number of Protocols: 1

Protocol	LineRate(Gbps)	DataWidth	Refclk(MHz)	Quad Count	Quad PLL
Custom 1	3.125	32	125.000	1	<input checked="" type="checkbox"/>
cpri		32	125.000	1	
gigabit ethernet		32	156.250	2	
tenGBASE-R		32	187.500	3	
xlai		40	189.394	4	
Custom 1			195.313		
			234.375		
			250.000		
			284.091		

PLL

QPLL0

QPLL0

QPLL1

CPLL

OK Cancel

IBERT IP Setting

Customize IP

IBERT 7 Series GTX (3.0)

Documentation IP Location Switch to Defaults

Show disabled ports

Component Name `ibert_7series_gtx_0`

Protocol Definition Protocol Selection Clock Settings Summary

REFCLK Distribution

from QPLL

CPLL

TX Clock Dividers

TX PMA

TX PCS

RX Clock Dividers

RX PMA

RX PCS

UG476_c2_05_101810

Refclk(MHz)	Quad Count	Quad PLL
125.000	1	<input checked="" type="checkbox"/>
125.000	1	
156.250	2	
187.500	3	
189.394	4	
195.313		
234.375		
250.000		
284.091		

PLL

QPLL0

QPLL0

QPLL1

CPLL

OK Cancel

IBERT IP Setting

Component Name

Protocol Definition

Protocol Selection

Clock Settings

Summary

Please select Protocol-Quad combination

GTX Location	Protocol Selected	Refclk Selection	TXUSRCLK Source
QUAD_109	Custom 1 / 3.125 Gbps	MGTREFCLK0 109	Channel 0
QUAD_110	None	None	Channel 0
QUAD_111	None	None	Channel 0
QUAD_112	Custom 1 / 3.125 Gbps	MGTREFCLK0 112	Channel 0

X-----
MGTREFCLK0 111
MGTREFCLK1 111
MGTREFCLK0 112
MGTREFCLK1 112

Protocol Definition

Protocol Selection

Clock Settings

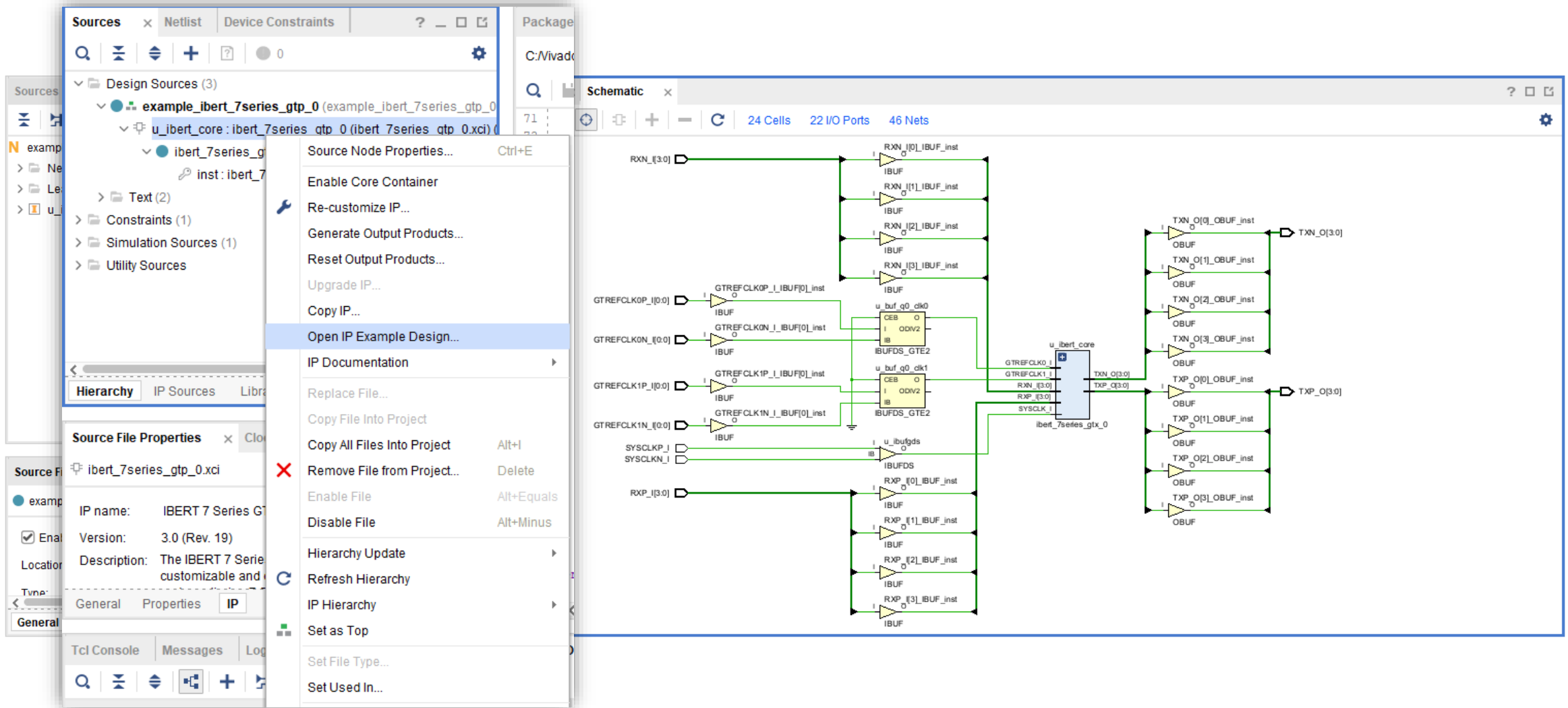
Summary

RXOUTCLK Probe

☒ Add RXOUTCLK Probes

Clock Type	Source	I/O Standard	P Package Pin	N Package Pin	Frequency(MHz)
RXOUTCLK	QUAD 109	DIFF SSTL15	UNASSIGNED	UNASSIGNED	97.65625
System Clock	QUAD109 0	DIFF SSTL15	UNASSIGNED	UNASSIGNED	125.000

IBERT Example Design



IBERT Example Design

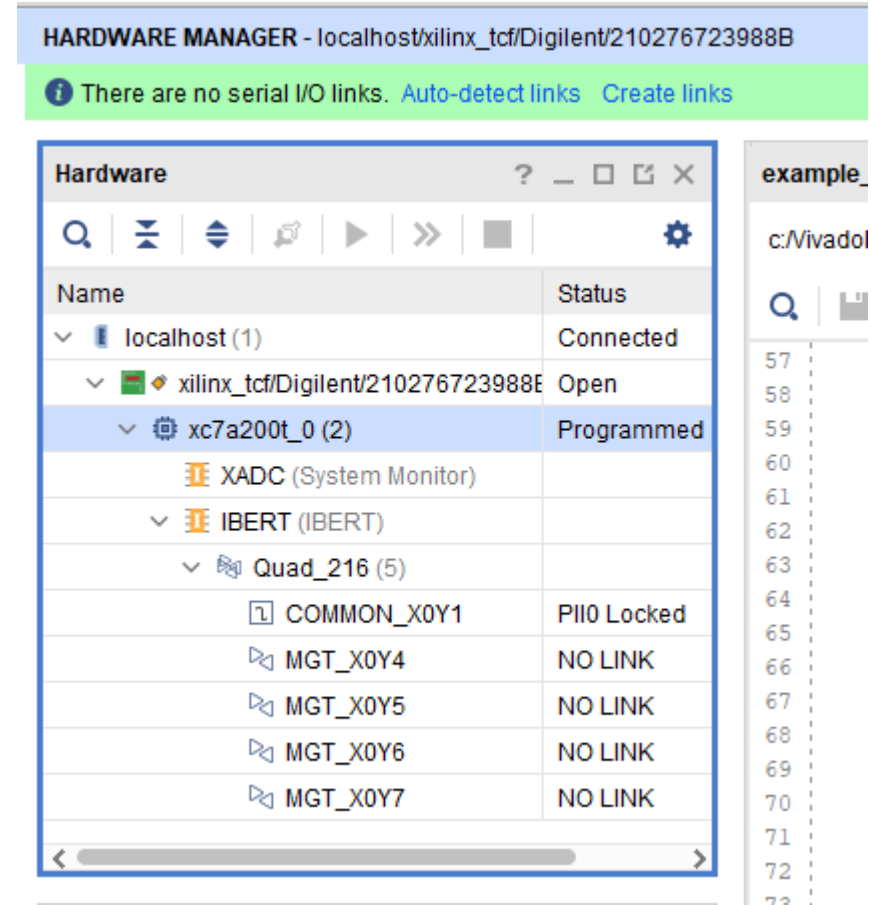
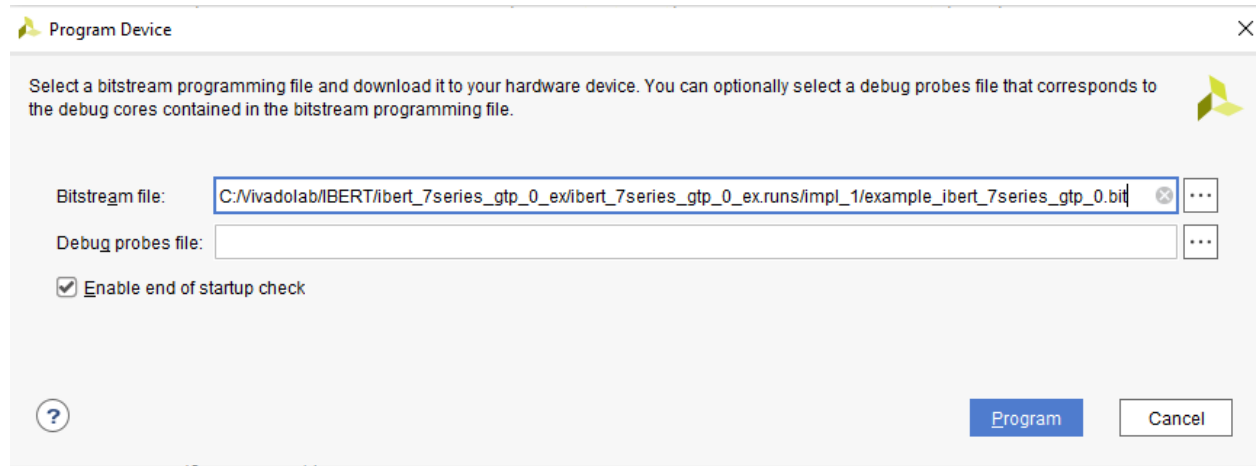
Port Descriptions

The core ports are shown in [Table 2-1](#).

Table 2-1: Core Ports

Signal Name	I/O	Description
SYSCLOCK_I	I	Clock that clocks all communication logic. This port is present only when an external clock is selected in the generator.
TXN_O[7:0]	O	Transmit differential pairs for each of the n GTP transceivers used.
TXP_O[7:0]		
RXN_O[7:0]	I	Receive differential pairs for each of the n GTP transceivers used.
RXP_O[7:0]		
GTREFCLK0_I[1:0]	I	GTP transceiver reference clocks used. The number of MGTRFCLK ports can be equal to or less than the number of transmit and receive ports because some GTP transceivers can share clock inputs.
GTREFCLK1_I[1:0]		
RXOUTCLK_O	O	Quad based RX output clock.

IBERT Debugging



IBERT Debugging

HARDWARE MANAGER - localhost/xilinx_tcf/Digilent/210276723988B

There are no serial I/O links. [Auto-detect links](#) [Create links](#)

Hardware

Name	Status
localhost (1)	Connected
xilinx_tcf/Digilent/210276723988B	Open
xc7a200t_0 (2)	Programmed
XADC (System Monitor)	
IBERT (IBERT)	
Quad_216 (5)	
COMMON_X0Y1	PII0 Locked
MGT_X0Y4	NO LINK
MGT_X0Y5	NO LINK
MGT_X0Y6	NO LINK
MGT_X0Y7	NO LINK

Serial I/O Links

[Auto-detect links](#) or [Create links](#)

to add serial I/O links to this window.

example_ibert_7series_gtp_0.v

Create Links

To create a new link select a TX GT and/or an RX GT, then click the Add button on the New Links toolbar.

TX GTs

Search: Q-

- Quad_216/MGT_X0Y4/TX (xc7a200t_0)
- Quad_216/MGT_X0Y5/TX (xc7a200t_0)
- Quad_216/MGT_X0Y6/TX (xc7a200t_0)
- Quad_216/MGT_X0Y7/TX (xc7a200t_0)

RX GTs

Search: Q-

- Quad_216/MGT_X0Y4/RX (xc7a200t_0)
- Quad_216/MGT_X0Y5/RX (xc7a200t_0)
- Quad_216/MGT_X0Y6/RX (xc7a200t_0)
- Quad_216/MGT_X0Y7/RX (xc7a200t_0)

New Links

+ -

No content

☒ Create link group

Link group description: Link Group 0

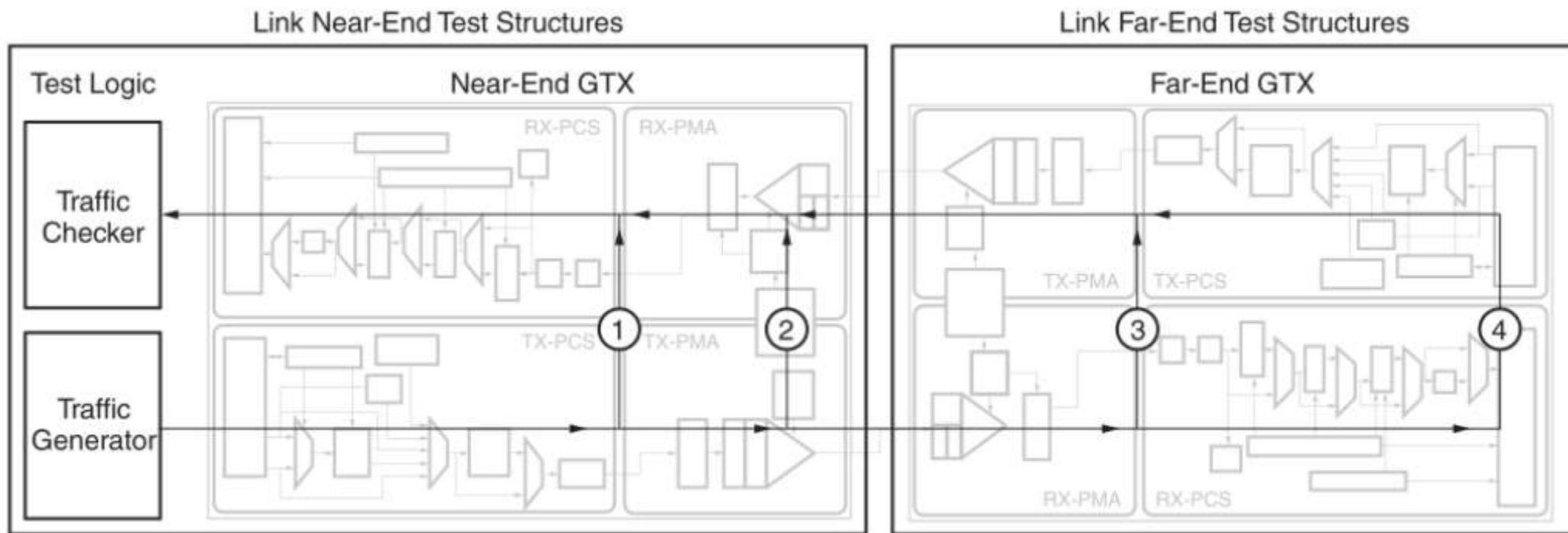
☒ Open Serial I/O Analyzer layout

OK Cancel

IBERT Debugging

Serial I/O Links																		
Name	TX	RX	Status	Bits	Errors	BER	IBERT Reset	TX Pattern	RX Pattern	TX Pre-Cursor	TX Post-Cursor	TX Diff Swing	Inject Error	TX Reset	RX Reset	RX PLL Status	TX PLL Status	Loopback Mode
Ungrouped Links (0)																		
Link Group 0 (4)							Reset	PRBS 7-bit	PRBS 7-bit	0.00 dB (001)	0.00 dB (000)	959 mV (1100)	Inject	Reset	Reset			None
Link 0	Qua...	Quad...	NO LINK	1.967E12	3.745E11	1.904E-1	Reset	PRBS 7-bit	PRBS 7-bit	0.00 dB (001)	0.00 dB (000)	959 mV (1100)	Inject	Reset	Reset	Locked	Locked	None
Link 1	Qua...	Quad...	NO LINK	1.959E12	1.029E12	5.252E-1	Reset	PRBS 7-bit	PRBS 7-bit	0.00 dB (001)	0.00 dB (000)	959 mV (1100)	Inject	Reset	Reset	Locked	Locked	None
Link 2	Qua...	Quad...	NO LINK	1.967E12	9.153E11	4.654E-1	Reset	PRBS 7-bit	PRBS 7-bit	0.00 dB (001)	0.00 dB (000)	959 mV (1100)	Inject	Reset	Reset	Locked	Locked	None
Link 3	Qua...	Quad...	NO LINK	1.967E12	9.672E11	4.917E-1	Reset	PRBS 7-bit	PRBS 7-bit	0.00 dB (001)	0.00 dB (000)	959 mV (1100)	Inject	Reset	Reset	Locked	Locked	None

IBERT Debugging



UG476_c2_20_101810

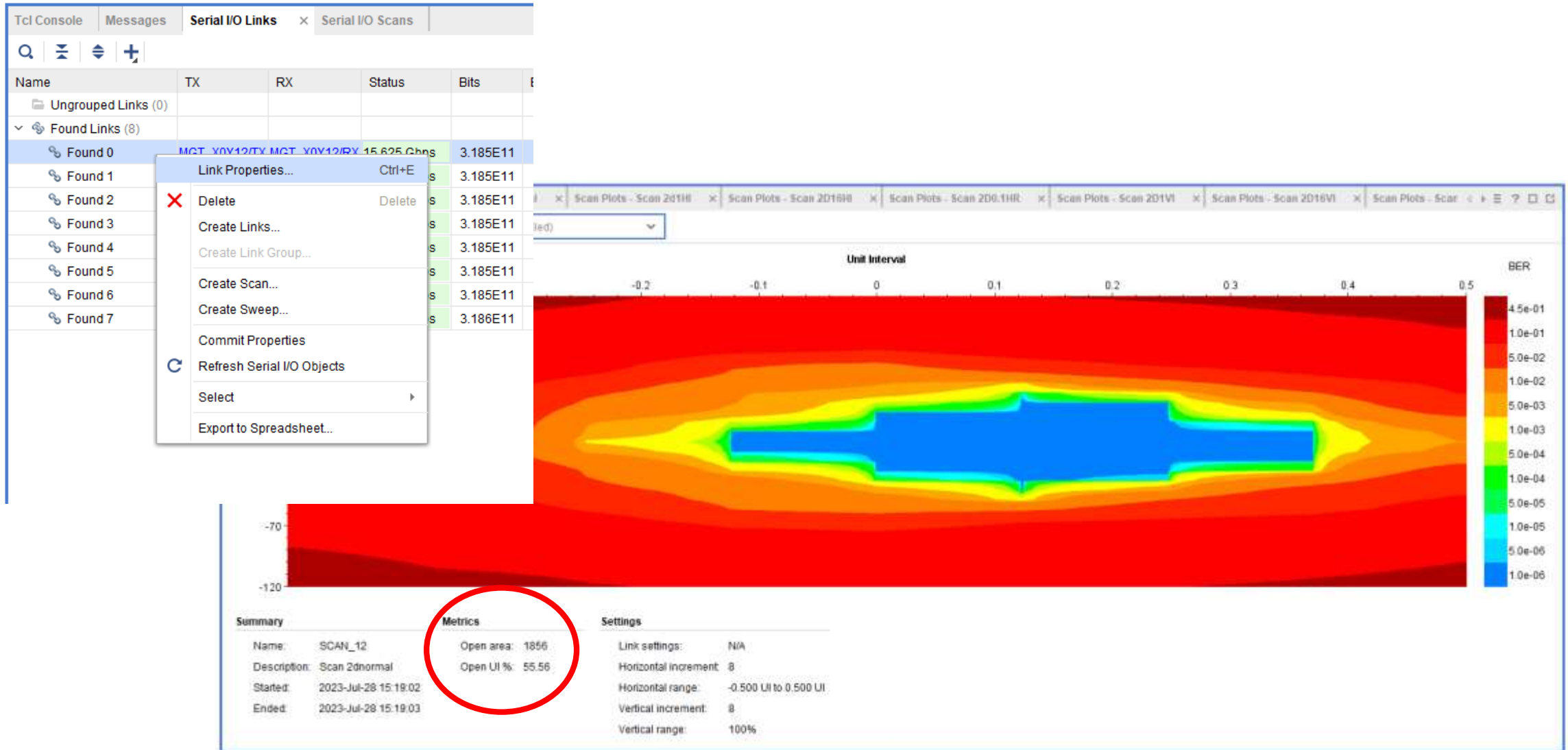
IBERT Debugging

Name	TX	RX	Status	Bits	Errors	BER	BERT Reset	RX PLL Status	TX PLL Status	Loopback Mode
Ungrouped Links (0)																		
Found Links (2)							Reset											Near-End PCS
Auto detected link 0	Quad_216...	Quad_216...	3.131 Gbps	1.574E12	1.241E10	7.881E-3	Reset									Locked	Locked	Near-End PCS
Auto detected link 1	Quad_216...	Quad_216...	3.118 Gbps	1.568E12	1.264E10	8.064E-3	Reset									Locked	Locked	Near-End PCS



TX Pattern	RX Pattern	TX Pre-Cursor	TX Post-Cursor	TX Diff Swing	Inject Error	TX Reset	RX Reset
PRBS 7-bit	PRBS 7-bit	0.00 dB (00000)	0.00 dB (00000)	959 mV (1100)	Inject	Reset	Reset
PRBS 7-bit	PRBS 7-bit	0.00 dB (00000)	0.00 dB (00000)	959 mV (1100)	Inject	Reset	Reset
PRBS 15-bit	PRBS 7-bit	0.00 dB (00000)	0.00 dB (00000)	959 mV (1100)	Inject	Reset	Reset
PRBS 23-bit	PRBS 7-bit	0.00 dB (00000)	0.00 dB (00000)	959 mV (1100)	Inject	Reset	Reset
PRBS 31-bit							
Fast Clk							
Slow Clk							

Eye Pattern



Eye Pattern

Create Sweep

Select the sweep properties and values to create and optionally run a set of scans on the selected link.

Link: Found 0 (MGT_X0Y12/TX, MGT_X0Y12/RX)

Description: Sweep 0

Scan Properties

1

Scan type:

2D Full Eyescan

2

Horizontal increment:

8

3

Horizontal range:

-0.500 UI to 0.500 UI

Vertical increment:

8

Vertical range:

100%

Dwell

4

☒ BER:

1e-5

☐ Time:

0

Sweep Properties

Sweep mode: Semi Custom

For each property select values to be swept. The sweep will cover all combinations of property values.

Set Properties & Values

Preview 81 Scans

Name	RXTERM	TXDIFFSWING	TXPOST	TXPRE
Sweep 0 - Scan 31	260 mV	170 mV (0000)	0.00 dB (00000)	0.00 dB (00000)
Sweep 0 - Scan 32	260 mV	170 mV (0000)	0.00 dB (00000)	4.08 dB (01111)
Sweep 0 - Scan 33	260 mV	170 mV (0000)	0.00 dB (00000)	6.02 dB (11111)
Sweep 0 - Scan 34	260 mV	170 mV (0000)	4.08 dB (01111)	0.00 dB (00000)
Sweep 0 - Scan 35	260 mV	170 mV (0000)	4.08 dB (01111)	4.08 dB (01111)

☐ Reset RX after applying Settings for each scan

☒ Run sweep

Sweep Properties

Sweep mode: Semi Custom

For each property select values to be swept. The sweep will cover all combinations of property values.

Set Properties

Full Custom

Exhaustive

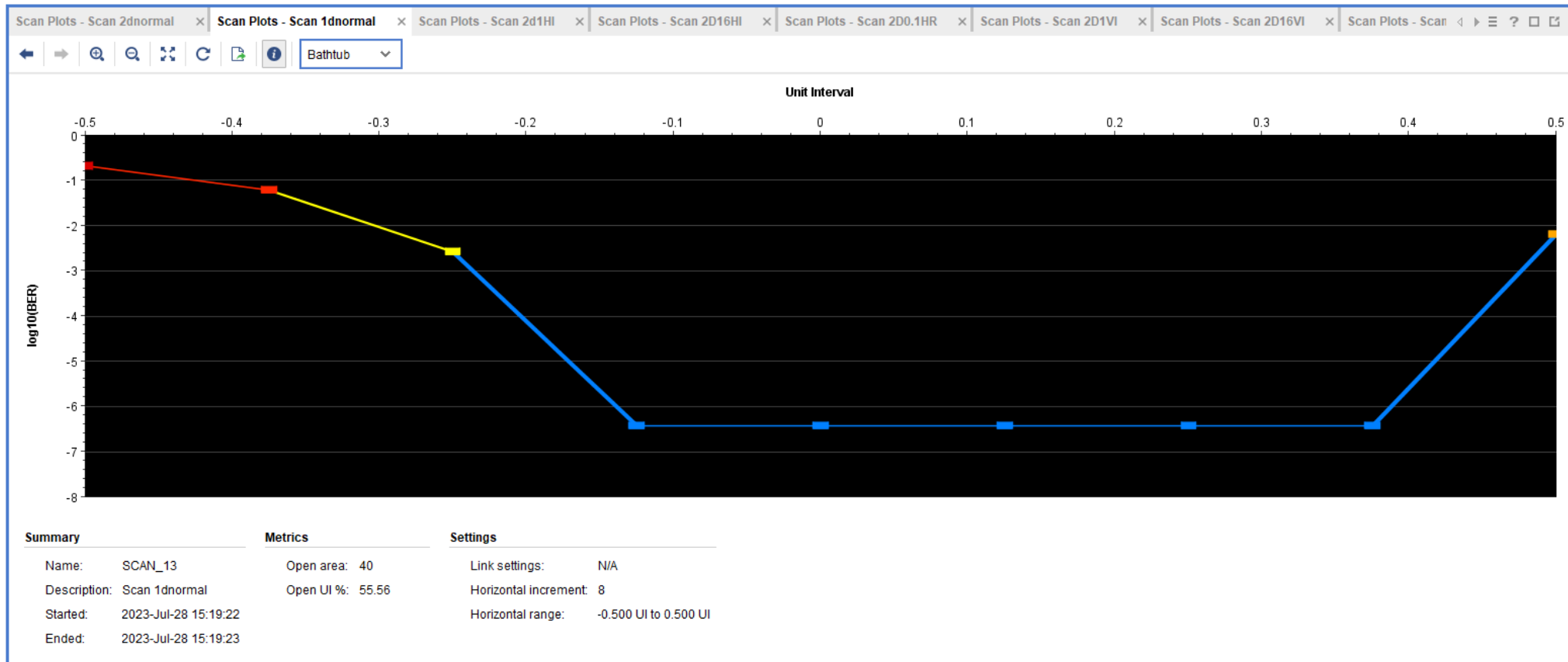
Sweep mode

Order	Property Name	Values to Sweep	# of Values
1	RXTERM	260 mV, 563 mV, 1030 mV	3
2	TXDIFFSWING	170 mV (0000), 660 mV (0111), 1080 mV (1111)	3
3	TXPOST	0.00 dB (00000), 4.08 dB (01111), 12.96 dB (11111)	3

☐ Reset RX after applying Settings for each scan

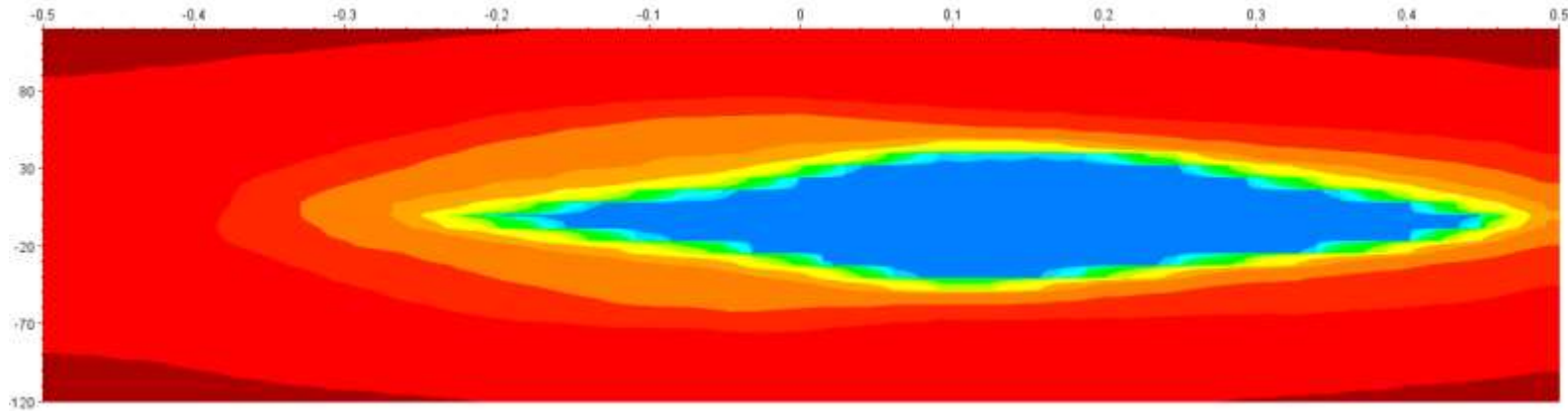
☒ Run sweep

Eye Pattern

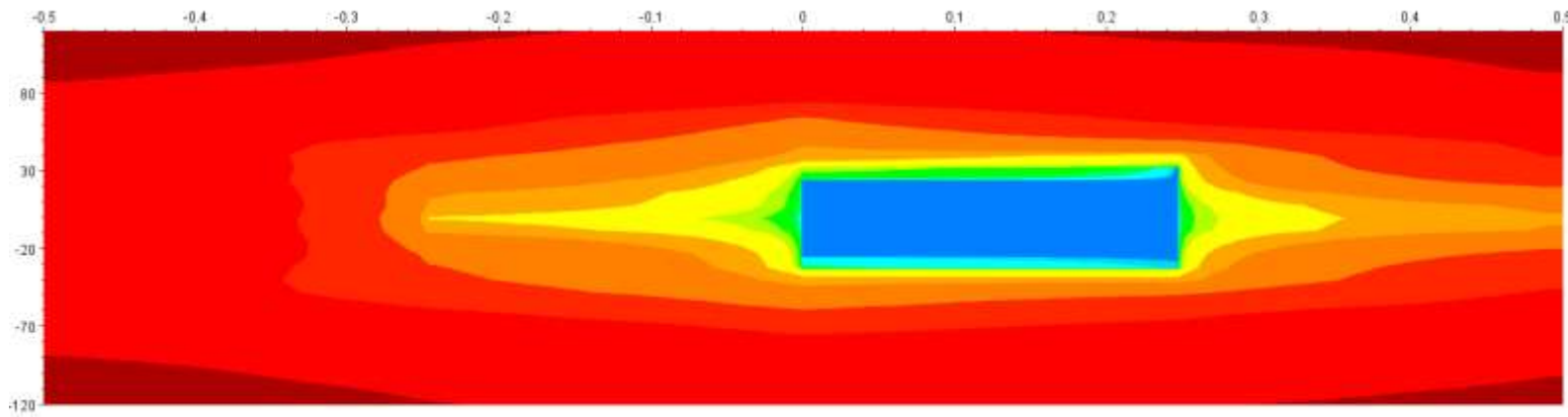


Eye Pattern

Horizontal incremental (1)

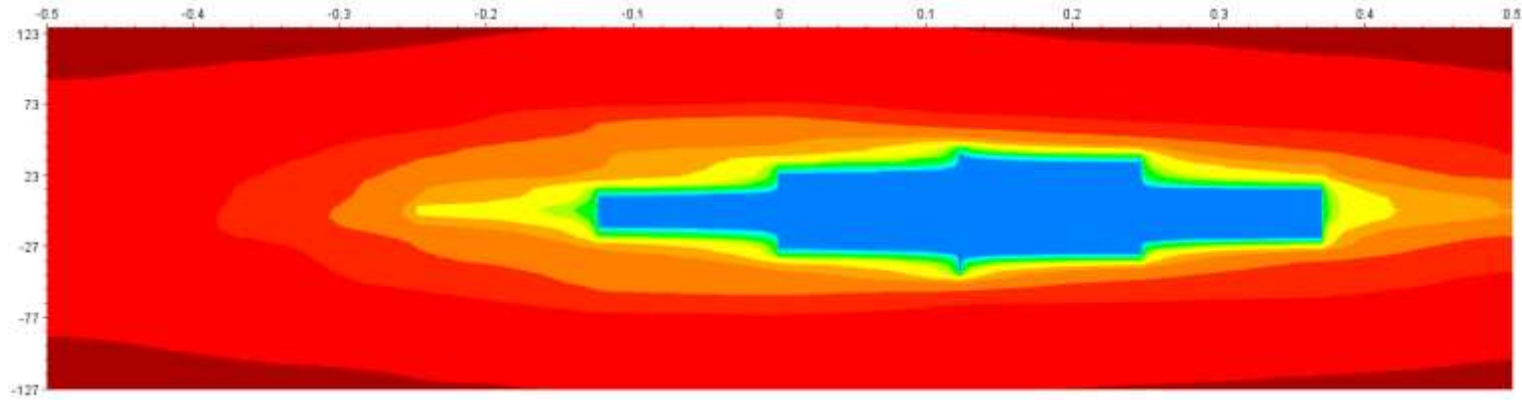


Horizontal incremental (16)

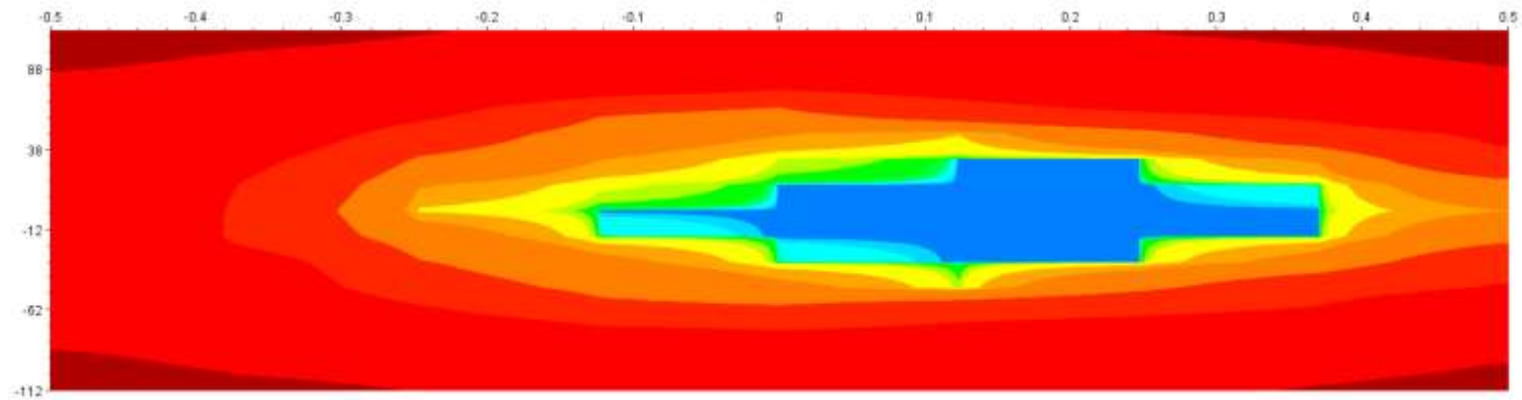


Eye Pattern

Vertical incremental (1)



Vertical incremental (16)





Thank you very much for your attention!

Course Agenda
2023