EXOSTIV Dashboard Hands-on - MICA board

Rev. 1.0.4 - February 23, 2017



Table of Contents

EXOSTIV Dashboard – Hands-on	3
Introduction	3
EXOSTIV for Xilinx FPGA – Overview	3
The MICA board setup	5
Board Overview	5
Connecting the board to the EXOSTIV probe	5
Connecting the configuration cable to the MICA board	6
Quick Start – Run simple captures from the MICA board	7
Demonstration kit contents and files	7
Overview of the reference design used for the demo instrumented with EXOSTIV IP	8
Start EXOSTIV Dashboard and load the reference project file	9
EXOSTIV IP configuration review	
Switch to the EXOSTIV Dashboard Analyzer	
Connect the Probe	12
Run a simple capture	14
Run a burst capture with trigger	16
Creating a 'netlist flow' project with EXOSTIV Dashboard	19
Start Vivado and create a new netlist flow project	19
Using the Constant	
Using the Core Inserter	
Using the Core Inserter Program the target board	
-	35
Program the target board	35
Program the target board Creating a 'RTL flow' project with EXOSTIV Dashboard	35

Revision History

Revision	Modifications
1.0.0	Initial revision
1.0.1	Minor updates
1.0.2	 Update for EXOSTIV Dashboard v. 1.5.3 Added section describing the RTL flow.
1.0.3	Update for EXOSTIV Dashboard v. 1.5.4
1.0.4	General review and update



EXOSTIV Dashboard – Hands-on

Introduction

This session provides a documented practical example on how to use EXOSTIV Dashboard software for Xilinx FPGA.

In this session, we review the general flow used by EXOSTIV to instrument FPGA and extract debug data out of it.

Then we show how to use EXOSTIV Analyzer with a pre-configured demonstration board and configuration.

Finally, we describe how to use EXOSTIV Dashboard 'Core Inserter' to configure and insert a simple EXOSTIV IP core into a reference design.

EXOSTIV for Xilinx FPGA – Overview

RTL flow

Typically, the target FPGA design is instrumented with an 'EXOSTIV IP' core after synthesis or from the RTL source code. This IP is configured with the EXOSTIV Dashboard core inserter to reach FPGA internal nodes, sample them and send the sampled data to outside with gigabit transceivers.

Unlike JTAG-based solution, this approach does not require growing the FPGA memory resources with the size of the capture, as the recorded data is progressively extracted towards a memory located outside of the target FPGA. When the average required bandwidth does not exceed what the reserved transceivers are capable of, data can be extracted from the FPGA as a continuous flow or in bursts until the external memory is full.

Netlist flow

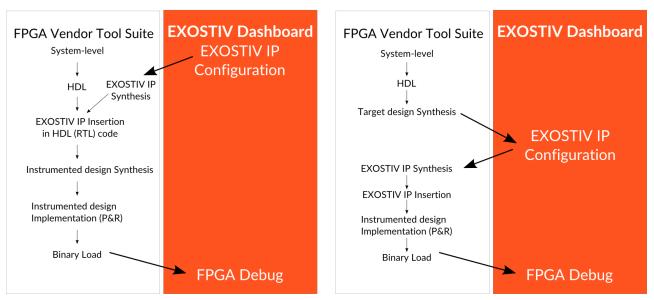


Figure 1: EXOSTIV for Xilinx FPGA – flows overview

- 2 alternate flows can be used with EXOSTIV: the 'RTL flow', and the 'netlist flow':
 The 'RTL flow' (or 'HDL flow') is used to insert the EXOSTIV IP in the RTL (VHDL or Verilog) source code. EXOSTIV Dashboard software is used to set up a generic IP provided as an output netlist and a top-level component (VHDL) or module (Verilog). This output IP has to be instantiated 'manually' in the source RTL code after which the synthesis and implementation of the design are to be run.
- The 'netlist flow' is used to insert the EXOSTIV IP into a synthesized target design (= in the target design netlist).
 EXOSTIV Dashboard is used to configure the EXOSTIV IP and to insert / connect it in the target design netlist. This flow is more automatically managed from the EXOSTIV Dashboard and does not require any manual insertion.
 From a flow point of view, it has the advantage to work from a synthesized design and hence to save the time needed to synthesize the target FPGA every time a new EXOSTIV IP must be inserted. Once the target design

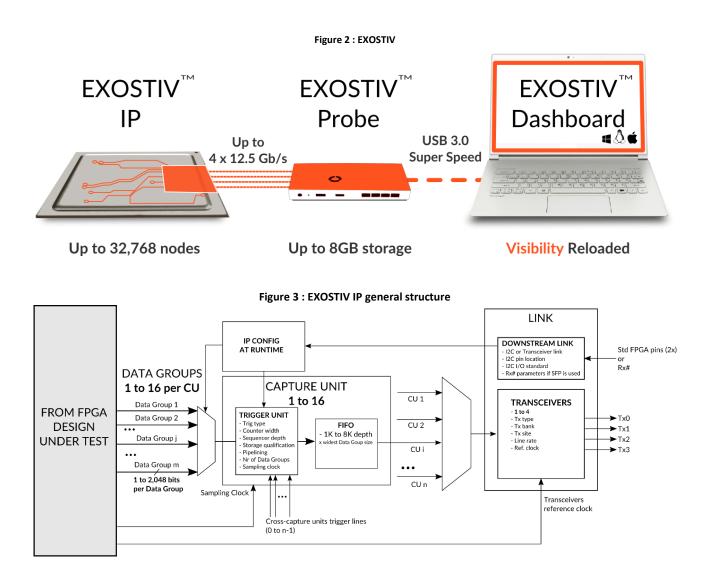


netlist is instrumented with EXOSTIV IP, it has to be implemented (place, route, bitstream generation). This flow is depicted at at

- Figure 1.

The general structure of EXOSTIV is depicted at **Figure 2**. With a large external 8GB memory, EXOSTIV provides up to 200,000 times more visibility than tradition embedded instrumentation solutions.

Figure 3 shows the general structure of the EXOSTIV IP core inserted into the FPGA.





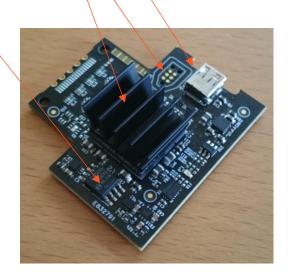
The MICA board setup

Board Overview

The 'MICA' board is a compact target board that can be used to get started with EXOSTIV and demo the system. It is pre-configured with an example FPGA design.

The 'MICA' board is a target board mounted with an Artix-7 FPGA. Features:

- Artix-7 FPGA xc7a35tcsg325-2
- Micro-HDMI type connector, with 3 GTP lines for use with EXOSTIV Dashboard. The board power is supplied through this connector by the EXOSTIV probe
- JTAG interface to reprogram the FPGA and the FPGA configuration EEPROM
- FPGA configuration EEPROM, pre-loaded with an example configuration for demo.



Connecting the board to the EXOSTIV probe

- 1. Place the HDMI to micro HDMI adapter on one end of the HDMI cable
- 2. Plug the HDMI end into the EXOSTIV probe HDMI connector and the micro-HDMI end into the MICA board



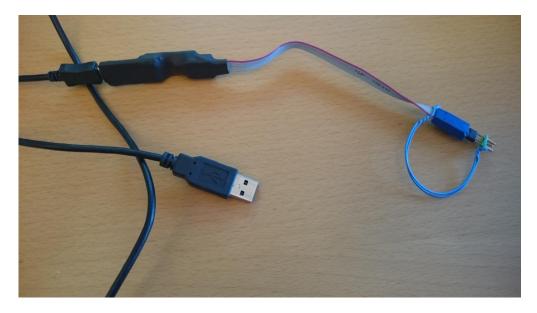


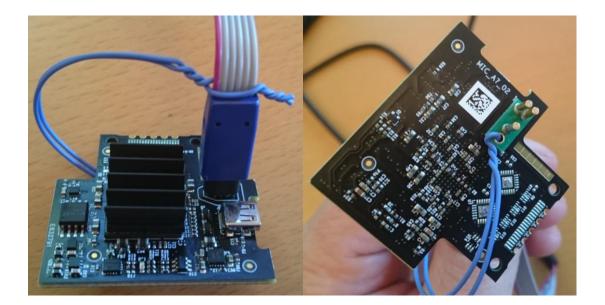
This setup can be used for demonstrating EXOSTIV: up to 3 transceivers are connected to the EXOSTIV probe through the HDMI connector and cables. The MICA board is powered through this cable too. Providing power through the HDMI cable requires using the proper option setting from EXOSTIV Dashboard (see Options menu).

Connecting the configuration cable to the MICA board

The demonstration kit includes a JTAG programming cable adapter for configuring the MICA board FPGA and its configuration EEPROM. This is only necessary when the FPGA configuration has to be changed.

It has to be connected to the PC with the provided micro USB cable. Please check the following article to know how to update the MICA board configuration: http://www.exostivlabs.com/knowledgebase/how-do-i-update-the-mica-board-configuration/







Quick Start – Run simple captures from the MICA board

This section of the tutorial shows how to use the MICA board setup with its standard configuration for capturing data with EXOSTIV Analyzer. This part loads a predefined project and does not use the EXOSTIV Core Inserter, which is described later in this document, from on page 19.

Demonstration kit contents and files

The demonstration kit includes:

- 1. 1x EXOSTIV probe with a power supply
- 2. 1x USB 3.0 cable for the EXOSTIV probe
- 3. 1x HDMI cable
- 4. 1x HDMI to mico-HDMI adapter
- 5. 1x MICA FPGA board
- 6. 1x programming kit for the MICA board

The following software and files are used with the demonstration kit:

Please go to <u>www.xilinx.com</u> to download the software. The Vivado Webpack (free) is ok for using this
demonstration kit. You'll need to register to download the software.
Please download from this page: <u>http://www.exostivlabs.com/support/downloads/</u>
Please contact <pre>support@exostivlabs.com</pre> to receive the latest download link and a license key.
Please check the <u>'UG501 - Getting started guide'</u> for installation instructions.
Please download from this page: http://www.exostivlabs.com/support/downloads/
File archive : Demo-MICA-3links_1.6.x.zip
Files: demo_mica702-3links-1.6.1.epf (or newer): EXOSTIV Dashboard project file. *.xml : EXOSTIV MYRIAD Waveform Viewer wave formatting files. Includes the MICA board binaries : demo_mica702-3links-1.6.1.bin and demo_mica702-3links- 1.6.1.bit – or newer.
The MICA board is pre-configured. Use these files if the board configuration was changed and you want to revert back to the original configuration.
Please download from this page: <u>http://www.exostivlabs.com/support/downloads/</u> File archive: DEMO_MIC_A7_02-export.zip

Overview of the reference design used for the demo instrumented with EXOSTIV IP.

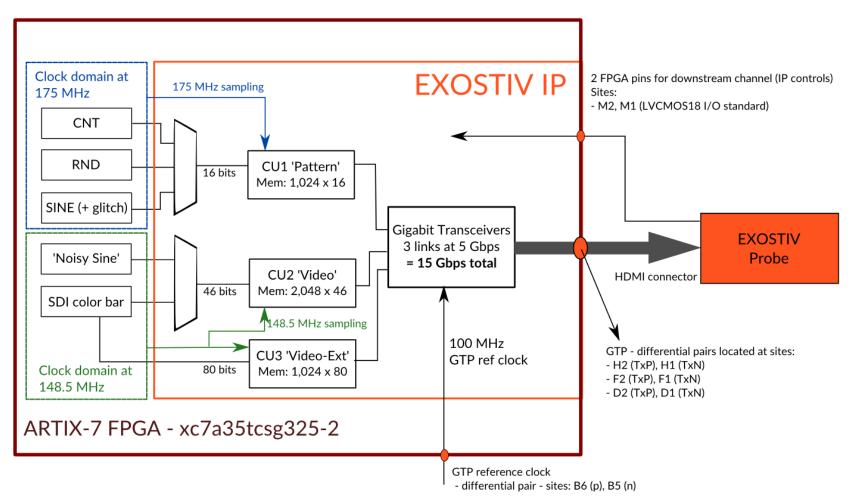


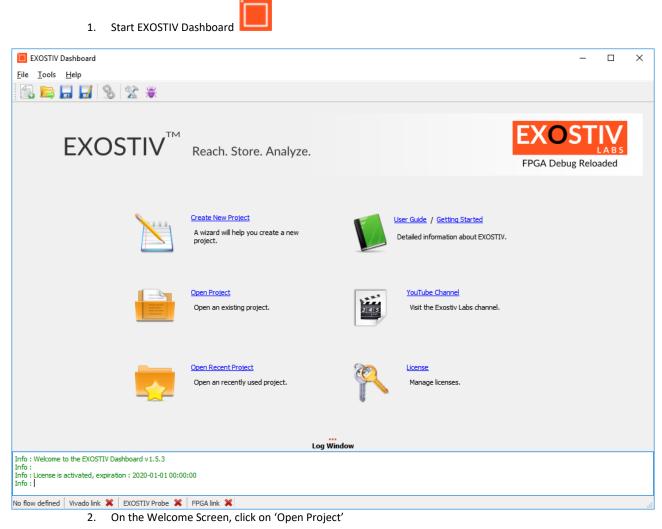
Figure 4 : Overview of the reference design instrumented with EXOSTIV IP.

REMARKS:

- 1) The original design provided in the DEMO_MIC_A7_02-export.zip Vivado project does not include the EXOSTIV IP inserted.
- 2) The binary files provided as demo_mica702-3links.bit and demo_mica702-3links.bin include the EXOSTIV IP . The MICA board is provided pre-configured with these files.

Start EXOSTIV Dashboard and load the reference project file

All project files used with the EXOSTIV Dashboard contains the settings of one EXOSTIV IP.



3. Locate and load 'demo_mica702-3links-1.6.1.epf'



Co	Link nfiguration	>>>	Capture Configuration	<u>>>></u>	Insert EXOSTIV IP	>>>	Debug Design	
PGA Type			Connector					
amily	Artix-7	•	Connector type	HDMI -				
ackage	csg325	•						
peed grade	-2	-						
art	xc7a35tcsg325-2	•						
pstream Lin	k		Downstream Link					
ransceiver	bank 216	•	• Use I2C link	O Use transceiver link				
IGT type	GTP		SCL package pin	M2 ~				
IGT_TxP0	H2		SDA package pin	M1 ~	-			
IGT_TxP1	F2		SCL I/O standard	LVCMOS18 -	Ī			
IGT_TxP2	D2		SDA I/O standard	LVCMOS18]			
/IGT_TxP3	B2		I/O voltages belo level shifter.	w 2.5V requires an external				
eference O	lock							
ransceiver	bank 216	•						
IGT_REFCLI	K_P0 D6							
IGT_REFCLI	K_P1 B6							
requency (I		2						
	Range : 60 MHz	to 660 MHz						
ine rate (Gb		•						
ink rate (Gb								
LL type use	ed QPLL ck.output							
XOSTIV do								
XOSTIV do								-
EXOSTIV do								_

The Dashboard opens on the Core Inserter 'Link Configuration' window. Please note that this example is based on the 'netlist flow'.

EXOSTIV IP configuration review

The 3 buttons on the top of the window show the flow for configuring EXOSTIV IP, run insertion and then use the EXOSTIV Dashboard analyzer.

The steps required to configure and insert an IP are detailed from section 'Creating a 'netlist flow' project with EXOSTIV' at page 19 below.



Clicking on the 'Link Configuration' and 'Capture Configuration' buttons switch the display and allow to check the EXOSTIV IP configuration as defined in the demonstration project and loaded into the MICA demonstration board.

The overall settings match the description of Figure 4.



EXOSTIV Dashboard - D:/Projects/	Xplorer/Hands-	On/Demo-MICA-3links/	demo_mica702-3links-1.6.1.epf			_		×
	*							
Link Configuration	>>>	Capture Configuration	>>>	Insert EXOSTIV IP	2.2	Debug Design		
Capture units (3 out of max	c. 16)			Pattern				
✓ Pattern	Т	riggering			Data			
Cnt	т	rigger unit type	Levels / Edges / Comparisons	-	Fifo depth	1024	•	
Sine Noise		it operations	X, 0, 1, R, F, B, N		Number of data groups	3 out of max. 16		
Double click to add Data Group		us operations Counter width	==, >, <, >=, <=, <>, in rar Disabled	ige, out of range	Number of data probes	16 out of max. 2048		
✓ Video	s	equencer Depth	Disabled	~				
SDI		torage qualification						
Noise Double click to add Data Group	N	lumber of pipes	Disabled	•				
✓ Video-Extended	S	ampling Clock						
Vid-Extended		Olk						
Double click to add Data Group Double click to add Capture Unit								
L			Log Window					
Info :			Log Wildow					^
Info : License is activated, expiration : 20. Info :				1				
Info : Project file "D:/Projects/Xplorer/Han	ias-On/Demo-MIC/	A-31111KS/demo_mica702-3lii	nks-1.6.1.ept ⁻ loaded successfully.					~
Netlist flow 🛛 Vivado link 💥 🗍 EXOSTIV P	robe 💥 🛛 FPGA	link X						



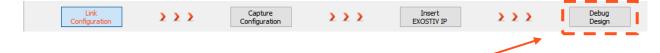
Switch to the EXOSTIV Dashboard Analyzer

The EXOSTIV Dashboard 'Analyzer' is used with the EXOSTIV Probe to sample and capture the internal nodes connected to the EXOSTIV IP.

Once you have loaded the target FPGA with the instrumented configuration, access the Analyzer by clicking on the 'bug icon' lets you access the EXOSTIV Analyzer.



Alternatively, the Analyzer can be accessed by clicking on 'Debug Design' in the top tool flow in the Core Inserter.



Connect the Probe

- 1. Connect the MICA board to the EXOSTIV probe
- 2. Power on the EXOSTIV Probe
- 3. Click on the 'connect' button in EXOSTIV Dashboard toolbar:
- 4. Click on 'Connect'

EXOSTIV Probe Connection	×
Probe found. Press 'Connect' to connect to the probe. Press 'Close' to continue without probe.	
searching for devices	Connect Close

Once connected to the probe, EXOSTIV automatically attempts to connect the probe to the IP that is in the target FPGA.

Info : Starting a link quality test for link S4 Info : Link quality test succeeded. Info : Info : Connected to EXOSTIV IP	
Netlist flow 🛛 Vivado link 💥 🗍 EXOSTIV Probe 🛩 🗍	FPGA link 🖌
EXOSTIV Dashboard is connected to the EXOSTIV Probe through USB	EXOSTIV Dashboard has found a valid EXOSTIV IP in the target design and there is A valid communication with it

Remarks:

- For the 'FPGA link' to be established, the EXOSTIV Probe must be able to:
- 1. properly communicate with an IP inserted in the target FPGA and:



- 2. check that the EXOSTIV IP inserted in the FPGA matches with the IP settings (CU, data groups, ...) as defined in the project file. For instance, if a project file is used, that does not match the EXOSTIV IP inserted in the FPGA, the communication will not be established and the EXOSTIV Dashboard Analyzer will not be usable. To check if the EXOSTIV IP inserted in the target FPGA matches with the project settings, a unique identifier (UUID) is programmed in the generated IP and read back with the EXOSTIV Probe. This UUID is also saved in the project file when a new EXOSTIV IP is generated.
- 3. The MICA board is powered through the HDMI link. The power line of the HDMI link is enabled through the EXOSTIV Dashboard menu: Tools > Menu

Options	×
Vivado	
Vivado Link Timeout	1 minute 🔹 💌
Capture Units Definition	
Confirm removing capture unit Confirm removing data group	\mathbb{N}
EXOSTIV IP Insertion	
Auto save project on IP insertion	
On Application Close	
Confirm close	
Save project file Save wave configuration files	
HDMI Cable Setup	
Enable HDMI output power	
Miscellaneous Settings	
Digit grouping	, 👻
Restore Defaults Cancel	OK

To power on the MICA board, the project file **must** be loaded into the EXOSTIV Dashboard before attempting to connect to the EXOSTIV Probe. When the EXOSTIV Dashboard connects to the EXOSTIV Probe through USB, it first checks in the project settings if the HDMI cable should be used – and if the output power option is selected. If it is the case, the power is enabled before the probe attempts to communicate with the EXOSTIV IP loaded in the target FPGA.



Run a simple capture

- 1. Select 'Pattern' capture unit tab
- 2. Select 'Sine' Data Group
- 3. Select 'Stream to probe' Transfer mode
- 4. Specify 1 as number of captures
- 5. Specify the following number of samples per capture: 1999872
- 6. Set trigger position to 1022
- 7. Click on 'Run immediately'.

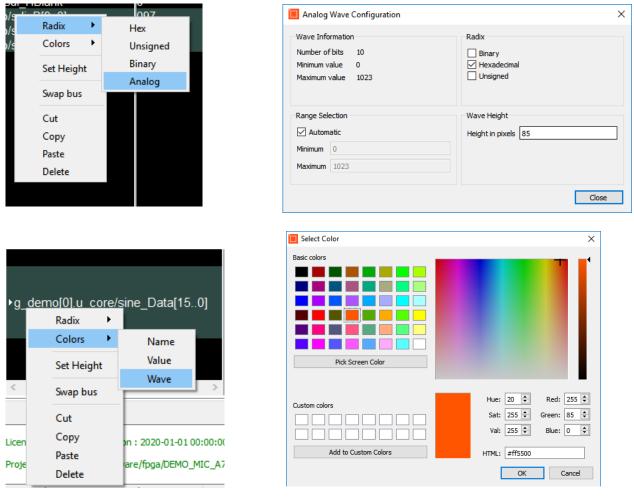
Pattern Video Vid	eo-Extended 1.	
	Data Group Selectio	n
Sine	2.	
>>>=>	Capture (Control
Data		
Transfer mode	Stream to Probe	
Transfer mode Number of captures	Stream to Probe	238
	1 1 to	238 36 1,024
Number of captures	1 1 to 1024 7 32 to	3. – 6
Number of captures Samples per capture	1 1 to 1024 7 32 to	3 6 1,024

The data is captured, uploaded to the PC and encoded as waves. They appear in the waveform viewer.

EXOSTIV Dashboard - C:/Projects/Xplorer/Hands-On/Demo-MICA-3links/demo_mica7	12-3links.epf	- 0 ×
File Jools Help		
Pattern Video Video-Extended		
Data Group Selection	AND Equation	Output Equation
sine •	Signal Name Operation	
4	8 1 □ g_demo[0].u_core/sine_Data[150] == * 0000 hex	
>i > S Capture Control	affic.	
Data	5	
Transfer mode Stream to Probe		
Number of captures 1 to 20,262		
Samples per capture 1024 * 32 to 1,024	Use other capture units	
Samples per capture 23552 1,024 to 477,218,560 Trigger		
	OR Equation Signal Name Operation	
Trigger position 8714 2 to 23,550	arginal reame Operation	
Status		
Status -		
Capture		
Capture sample		
»:		
Auto Export Captures	Use other capture units	
Enable auto export V		
	Wave Viewer	
📃 🗟 🗷 💁 🖋 🔍 🔍 🔍 🗶 🔏 🦾 🚳	ny transition 👻	
-8100 -720	-6300 -5400 -4500 -3600 -2700 -1800 -900 0 900 1800 2700 3800 4500 -	5400 6300 7200 8100 9000 9900 10800 11700 12800 13500 14400
•g_demo[0].u_core/sine_Data[15 EC22		· · · · · · · · · · · · · · · · · · ·
		· · · · · · · · · · · · · · · · · · ·
900 1	800 2700 3600 4500 5400 6300 7200 8100 9000 9900 10800 1170 12800 13500	14400 15300 16200 17100 18000 18000 18000 20700 21600 22500 2340
c > < > <	11/3	>
	Log Window	
Info : Starting a link quality test for link 54 Info : Link quality test succeeded.		· · · · · · · · · · · · · · · · · · ·
Info : Info : Connected to EXOSTIV IP		
Netlist flow Vivado link 🗱 EXIOSTIV Probe 🛩 FPGA link 🛩		



8. Right-click on the sine_data bus in the waveform window and change radix to 'Analog' to display the digital sine wave as analog signal. Then do it again and select Color > Wave to change the wave color. This is a simple example of how the wave can be formatted.



As a result the Sine Wave is displayed as analog format, and can be expanded bit by bit as well...





Run a burst capture with trigger

- 1. Select 'Video' tab
- Select 'SDI' data group
 Set up a capture with e.g. 100 captures of 3584 samples
 Position trigger at sample 2345

Pattern	Video	Video-	Extended			
			Data Gro	up Selec	tion	
₽₽	I					•
≥l ≥ Data	8			Captur	e Control	^
Transfer	mode		Stream to	Probe		•
Number	of captures	S	100		1 to 133,152	
Samples	per captur	e	2048		32 to 2,048	
Samples	per captur	e	3584		2,048 to 4,771,840	
Trigger						
Trigger p	osition		2345		2 to 3,582	
Status					-	
Status			-			
Capture						•

5. Go to the 'trigger controls' select the following pre-saved trigger equation:



		Signal Name		Operation
1		g_demo[0].u_core/sdi_Valid	== *	R b
	Use o	ther capture units 🖉 📉		
4			OR Equation	
		Signal Name		Operation
1	\checkmark	g_demo[0].u_core/sdi_SOF	== 🔻	R b
2		g_demo[0].u_core/sdi_HBlank	==	R b

6. Run the capture with trigger by clicking on the '>|' button:

												Wave	Viewe	r																	
mo[0] u_core/sdi_0 mo[0] u_core/sdi_0 mo[0] u_core/sdi_0 mo[0] u_core/sdi_10	🔁 🔤 ൽ 🛞 🔍 🖯	Q	1. 4	1 40	C					Iny tra	nsition	•																			
		11234					2222	121212	222	333	3333	333	444	444	4 4 4	455	555	sisisi	566	666	666	887	7 7 7	7 7 7	7 7 7	8 8 8	888	sistal	e e le le le	elele	99
	mol01.u core/sdi 90				A A A			111									ii i	i iii					i Hi	111	i iii						T
	no[0].u core/sdi V0	່ນມີມັນ	ιiii	i i i i	ti ti	ΠŪ	ШÚ	ίΟU	UUU	L III L	Ti Ti	ίιü	υŪU	1.III	ΠÚ	Шİ.	Шİ	Шİ.	ΞŪŪ	JUU	<u> I</u> III	uliii	Шİ	ЛÚ	ιü	ЛÚ		ШÜ	ίιu	UUU	Шİ
	no[0].u_core/sdi_H0																														
ma(0) u_core/sdi 208 mo(0) u_core/sdi 208 mo(0) u_core/sdi 204 mo(0) u_core/sdi 204	no[0].u_core/sdi_\1																														
	mo[0].u_core/sdi_030																														
	mo[0].u_core/sdi_298																<u> </u>						1 1 1	1111			10,014				
	moloj.u_core/sdi_000																														

- 7. To change the wave formatting, click on the 'load wave format' button :
- 8. Locate an select file 'sdi.xml', open, then zoom...

0



→ ↑ ↑	ardware > fpga > DEMO_MIC_A7_02 > Demo-N	/ICA-3links →	Search Der	mo-MICA-3links
nize 🔻 New folder				== -
.svn	^ Name ^	Date modified	Туре	Size
o board	demo_mica702-3links	8/18/2016 11:30 AM	File folder	
🔥 fpga	Cnt.xml	8/4/2016 5:19 PM	XML Document	2 KB
oc701_hfh_eeprom	noise.xml	8/4/2016 5:19 PM	XML Document	5 KB
o ac701_ibert	🗋 rnd.xml	8/4/2016 5:19 PM	XML Document	2 KB
o common	📄 sdi.xml	8/18/2016 4:28 PM	XML Document	5 KB
DEMO_01	sdi-ext.xml	8/17/2016 1:52 PM	XML Document	9 KB
DEMO_01_local160203				
DEMO_03				
DEMO_MIC_A7_01				
DEMO_MIC_A7_02				
Demo-MICA-3links				
- · · · · · · · ·	~			
File <u>n</u> ame: sdi.xml			 Configura 	tion Files (*.xml)

					Wave Vie	wer			
3 🗷 🗠 🕺 🖇 🤅	२ 🕂 व	<u> // // /</u>	🔏 🗲 I	🛤 🕅 🔽 関	Any transition 🔻				
I_core/sdi_SOF 0 I_core/sdi_VBlank 0 I_core/sdi_HBlank 0 I_core/sdi_Valid 1 I_core/sdi_LN[110] 48			7			10000 10000 10000 10000 10000 10000 10000 10000	10000 10000 10000 10000 10000 10000 10000 10000	10000 10000 10000 10000 10000 10000 10000 10000 10000	1231 10000 2 1231 10000 2 10000 2 10000 0 2 1000000 2 100000 2 10000
.u_core/sdi_R[90] 298			ÚI						
.u_core/sdi_G[90] 000									
.u_core/sdi_B[90] 048									
		350000	360000	370000 380000) 390000 4000	00 410000 420000	430000 440000	450000 460000	470000 480000



Creating a 'netlist flow' project with EXOSTIV Dashboard

In this section, we'll describe how to create a 'netlist flow' project from zero using EXOSTIV Dashboard. It shows how to configure an IP and insert it into the target design with EXOSITV Dashboard Core Inserter.

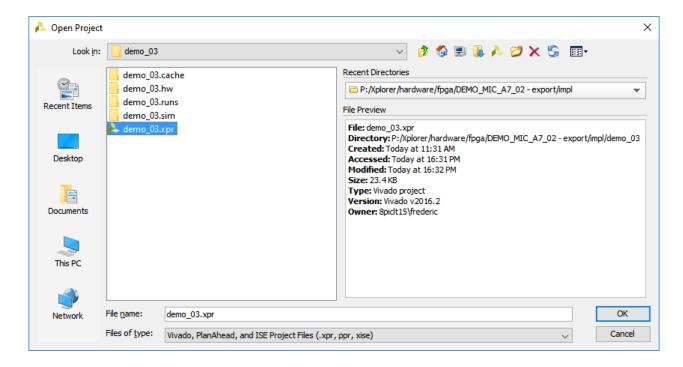
In this section, we'll use the MICA board reference design, Xilinx Vivado and the EXOSTIV Dashboard.

Start Vivado and create a new netlist flow project

Xilinx Vivado version 2015.4 or newer must be used. The free Webpack version is sufficient for this demonstration, as the Artix-7 FPGA device mounted on the MICA board is supported with this version.

1. Load the demonstration project 'DEMO03'

Vivado 2015.4	
le Flow <u>T</u> ools <u>W</u> indow <u>H</u> elp	
VIVADO. Productivity. Multiplied.	
Quick Start	
Create New Project Open Project Open Example Project	
Tasks	

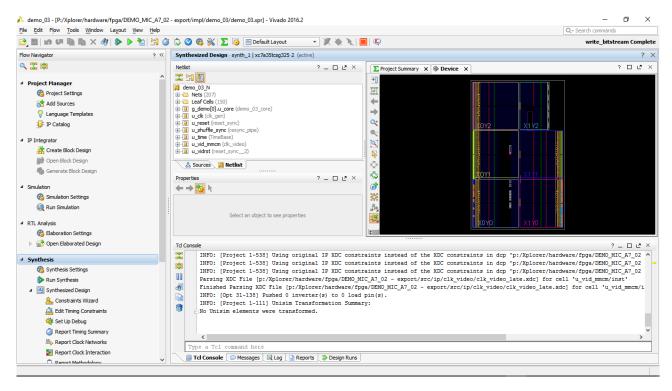




2. In Vivado, open 'Synthesized Design'

ile Edit Flow Tools <u>W</u> indow Layout													Q- Sea	rch comma	nds	
👌 🕼 🗸 🔷 🐂 📠 🗙 🖒	s 🕺 🗡 🔯	😬 Default Layout		🔭 🔲 🖏										write_bi	tstream (Complet
Flow Navigator	? «	Project Manager - de	mo_03													?)
2、 🔀 🖨		Sources		? = [🗆 🛃 × 🛛 🔽 Pro	ject Summ	nary X								? 🗆	l 🖻 🗵
4 Project Manager	^	🔍 🛣 🛱 🖬 💕			Projec	Project Settings										
🚳 Project Settings		Design Sources (name:	demo_0	3								
👌 Add Sources		🗄 🛅 Coefficient Fi			Project	location:	P:/Xplor	er/hardwar	e/fpga/D	EMO_MIC	A7_02 - export/	impl/dem	0_03			
Language Templates		Constraints (3) Gradient Source	e (E)		Product	family:	Artix-7									
IP Catalog			S (5)		Project	part:	<u>xc7a35</u>	csq325-2								
					Top mo	dule name:	demo C	<u>3 N</u>								
IP Integrator					: Target	anguage:	VHDL									
🎇 Create Block Design		Hierarchy IP Source	s Libraries Compil	e Order	Simulat	r language:	Mixed									
💕 Open Block Design		Properties		2 1	Synthe	sis					Tmol	ementa	tion			
🍓 Generate Block Design		Properties ? - C L* ×			Status:	√ Co	molata									
Simulation						es: (!) 649					Statu Mess			Complete 3 warning		
o Simulation Settings					-	un: synth						e run:		pl 1	8	
Run Simulation					Part:	xc7a35	itcsg325-2				Part:			7a35tcsq3	25-2	
-		S	elect an object to see	e properties	Strateg	y: <u>Vivado</u>	Synthesis	Defaults			Strat			vado Imple		Default
RTL Analysis											Incre	mental co	ompile: No			
Elaboration Settings					<											>
Open Elaborated Design		Design Runs													? _ □	L ² ×
Synthesis		Name		Constraints	Status	WNS	5 TNS	WHS	THS	TPWS	Failed Routes	LUT	FF B	RAM U		ISP St
🍪 Synthesis Settings		🔀 🖃 🗸 synth_1 (a	tive)	constrs_1	synth_design Comp	lete!						724	959	0	0	0
🗞 Run Synthesis				constrs_1	write_bitstream Comple	ete! 0.8	829 0.0	0.056	0.000	0.000		0 728	1079	21	0	0
Den Synthesized Design		Cut-of-Cont Dut-of-Cont Dut-of-Cont	w256_r8_synth_1	blk_mem_w256_r8	synth_design Complete	a							0	4	0	0
		∢		dk_gen	synth_design Complete							0	0	0	ŏ	Ő
Implementation		dk_video		dk_video	synth_design Complete							C		0	0	0
Maintain Settings		v rom_sine	synth_1 1021 synth 1	rom_sine rom_sine_1021	synth_design Complete synth design Complete							0		0	0	0
Run Implementation		📑 🗸 rom_sine	51b_T4111_synth_	1 rom_sine_51b_T411	1 synth_design Complete	8						24	53	11	0	ō
Open Implemented Design		····· V rom_sine	67b_T2039_synth_:	1 rom_sine_67b_T203	9 synth_design Complete	9						C	67	4	0	0
Program and Debug		z <														

After some time, the synthesized reference design is loaded into Vivado.



3. Start EXOSTIV Dashboard



In the welcome screen, click on 'Create New Proje

EXOSTIV Dashboard	/		. 🗆	×
File Tools Help				
🗟 🛤 🖬 🛃 🗞 🕱 🕷				
EXOSTIV [™]	Reach. Store. Analyze.	EXOS FPGA Debug R	LABS	
	·			
	Create New Project A wizard will help you create a new project. User Guide / Getting Started Detailed information about EXOSTIV.			
	Open Project YouTube Channel Open an existing project. Visit the Exostiv Labs channel.			
	Open Recent Project License Open an recently used project. Manage licenses.			
	Log Window			
Info : Welcome to the EXOSTIV Dashboard v1.5.3 Info : Info : License is activated, expiration : 2020-01-01 00:00 Info :				
No flow defined 🛛 Vivado link 💥 🗍 EXOSTIV Probe 💥	FPGA link 💥			

The 'Create New Project' window appears, prompting to select the desired flow (Netlist flow or RTL flow)

?	×
Browse.	
Create	•
	-

- Select 'Netlist IP Insertion' from the type drop-down box.

- Specify new project name and pick a location.

- For permission issues reasons please do not create project in Program Files (x86) or Program Files directory.



The 'Link Configuration window' of the core inserter appears.

	shboard - C:/Projects/my			sheer of				_	×
	<u>r</u> eip	k							
Cor	Link Infiguration	>>>	Capture Configuration	2	> >	Insert EXOSTIV IP	>>>	Debug Design	
FPGA Type			Connector						^
Family	Artix-7 - Defense-grade	•	Connector type	HDMI	•				
Package	cs325	-							
Speed grade	-11	-							
Part	xq7a50tcs325-1I	•							
- Upstream Lin	k		Downstream Link						
Transceiver b	bank 216	•	• Use I2C link	O Use transceiv	er link				
MGT type	GTP		SCL package pin	A9	~				
MGT_TxP0	H2		SDA package pin	A9	~				
MGT_TxP1	F2		SCL I/O standard	LVCMOS33	•				
MGT_TxP2	D2		SDA I/O standard	LVCMOS33	•				
MGT_TxP3	B2								
Reference Cl	ock								
Transceiver b	pank 216	•							
MGT_REFCLK	_P0 D6								
MGT_REFCLK	(_P1B6								
Frequency (N		2							
	Range : 60 MHz to	660 MHz							¥
				Log V	 Vindow				
Info : Info : License is a	tivated, expiration : 2020-	01-01 00:00:00							^
Info :	C:/Projects/mynewproject.		fully.						
									 ~
Vetlist flow Viva	do link 💢 🛛 EXOSTIV Prob	e X 🛛 FPGA link	×						

Using the Core Inserter

Overview

To access the 'Core Inserter', click on the following icon in the main toolbar:



Inserting the EXOSTIV IP requires 3 successive steps:

- Step 1 : Link Configuration
- Step 2 : Capture Configuration
- Step 3 : Run Insertion

These 3 steps are accessible through the top flow overview in the EXOSTIV Dashboard window

1	Link Configuration	<u>> > ></u>	Capture Configuration	>>>	Insert EXOSTIV IP	2.2.2	Debug Design
						_	



Step 1: Link Configuration

This step defines the characteristics of the target FPGA and of the interface between EXOSTIV IP and EXOSTIV Probe. The information required to complete this step depends on the target FPGA and the target FPGA board. For this hands-on session, we'll use the MICA board from Exostiv Labs. Please refer to the MICA 702 board user's guide for a details (<u>http://www.exostivlabs.com/files/documents/Demo%20Kit%20User's%20Guide%20-%20MICA702.pdf</u>).

Plea	ase set up as follows:
1.	FPGA Type

FPGA Type	
Family	Artix-7 👻
Package	csg325 👻
Speed grade	-2 🔻
Part	xc7a35tcsg325-2 🔻

2. Connector : HDMI, as we'll use the HDMI type of connection between the MICA board and EXOSTIV Probe

Connector	
Connector type	HDMI 🔻

3. Upstream link: this setting defines the location of the used gigabit transceivers on the FPGA package. In this case, we'll use the transceivers connected to the micro-HDMI on the MICA board. They are at sites H2, F2 and D2 of bank 216.

Upstream Link					
Transceiver bank	216 👻				
MGT type	GTP				
MGT_TxP0	H2 🗹				
MGT_TxP1	F2				
MGT_TxP2	D2				
MGT_TxP3	B2				

4. **Downstream link:** this setting defines the location of the 2 pins connected to the micro-HDMI connector used for the downstream 'I2C-like' link used to configure EXOSTIV IP at run time. SCL and SDA are respectively at sites M2 and M1 of the Artix-7 FPGA package. The I/O standard is LVCMOS18.

Downstream Link					
● Use I2C link ○ Use transceiver link					
SCL package pin	M2	\sim			
SDA package pin	M1	\sim			
SCL I/O standard	LVCMOS18	•			
SDA I/O standard	LVCMOS18	•			
I/O voltages below 2.5V require an external level shifter.					



5. Reference Clock: this defines the pin input of the reference clock used for the transceiver used for EXOSTIV, as well as its frequency (100 MHz). From the frequency, we can choose the link rate setting. We choose the maximum link rate available for this frequency / FPGA / EXOSTIV Probe model, that is 5 Gbps.

Reference Clock	
Transceiver bank	216 👻
MGT_REFCLK_P0	D6
MGT_REFCLK_P1	B6
Frequency (MHz)	100
	Range : 60 MHz to 660 MHz
Line rate (Gb/s)	5 🗸
Link rate (Gb/s)	15
EXOSTIV dock output	

Here is an overview of the 'Link Configuration' for this example:

Eile Iools Help								
	Capture Configuration Configuration	Debug						
Configuration FPGA Type Family Artix-7 Package csg325 Speed grade -2 Part xc7a3Stcsg325-2 Upstream Link Transceiver bank Transceiver bank 216 MGT type GTP MGT_TxP0 H2 MGT_TxP1 F2 MGT_TxP3 B2 Reference Clock E	Configuration EXOSTIV IP Connector Connector type Connector type HDMI • • Downstream Link • • Use I2C link O use transceiver link SCL package pin M2 ✓ SDA package pin M1 ✓ SCL I/O standard LVCMOS18 ▼ I/D voltages below 2.5V requires an external level shifter. •	Design		~				
Transceiver bank 216 MGT_REFCLK_P0 D6 MGT_REFCLK_P1 B6 Frequency (MHz) 100 Range : 60 MHz to 660 MHz Line rate (Gb/s) 5 Link rate (Gb/s) 5 PLL type used QPLL EXOSTIV dock output	Log Window			~				
Info : Info : Project file "D:/Projects/Xplorer/Hands-On/Demo-MICA-3links/ Netlist flow Vivado link 💥 EXOSTIV Probe 💥 FPGA link 💥				•				



✓ LINK CONFIGURATION: DONE !

To save the project:

Step 2 : Capture Configuration

This part of the flow defines the characteristics of the EXOSTIV IP core – namely:

- The capture units and their features:

- Trigger resources
- Enable / Disable storage qualification
- FIFO depth
- $\circ \quad \ \ {\rm The \ sampling \ clock \ for \ each \ capture \ unit}$
- o ...
- The data groups for each capture units and the signals from the target design that are part of each data group

EXOSTIV Dashboard - C:/Projects/mynewproject.epf			-	×
E 100 Ich				
Link Configuration	222	Insert >>>	Debug Design	
Capture units (1 out of max. 16)	Capture	e Unit 1		
Capture Unit 1		Data		
Data Group 1 Double click to add Data Group		▼ Fifo depth 1024	•	
Double click to add Data Gloup Bit operations Double click to add Capture Unit Bus operations	X, 0, 1, R, F, B, N ==	Number of data groups 1 out of max. 16		
Counter width	Disabled	Number of data probes 0 out of max. 2048		
Sequencer Depth	Disabled	*		
Storage qualification		_		
Number of pipes	Disabled	•		
ampling Clock				
· · · · · · · · · · · · · · · · · ·				
Select capture unit to see its main features				
-	Log Window			
Info : License is activated, expiration : 2020-01-01 00:00:00 Info :				^
Info : Project file "C:/Projects/Xplorer/hardware/fpga/DEMO_MIC_A7_02/test/Mica- Info : Project file "C:/Projects/mynewproject.epf" loaded successfully.	-3links-v1.5/demo_mica702-3links.epf"	oaded successfully.		
				~
Netlist flow 🛛 Vivado link 💢 🛛 EXOSTIV Probe 💥 🛛 FPGA link 💥				

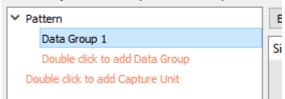


For 'Capture Unit 1', select the following:

- 1. Trigger unit type: Levels / Edges / Comparisons
- 2. Fifo Depth : 1024
- 3. Leave the other settings as they are.
- 4. Double-click on the Capture Unit name to change it. Change it to 'Pattern'.
- 5. Click on the data group name to define or see the connected nodes
- 6. Double-click on the Data Group name to change it. Change it to 'Sine'.

EXOSTIV Dashboard - C:/Projects/mynewproject.epf <u>File Iools H</u> elp	_	
Link Capture Insert Configuration Configuration Configuration	Debug Design	
Capture units (1 out of max. 16) Data Group 1		
Capture Unit 1 Edit Probes		
Data Group 1	Data	Trigger
Double click to add Capture Unit	Data	ngger
Select data group to define / see connected nodes		
C		>
Log Window		
Info : License is activated, expiration : 2020-01-01 00:00:00 Info :		^
Info : Project file "C:/Projects/Xplorer/hardware/fpga/DEMO_MIC_A7_02/test/Mica-3links-v1.5/demo_mica702-3links.epf" loaded successfully. Info : Project file "C:/Projects/mynewproject.epf" loaded successfully.		
		~
Netlist flow 🛛 Vivado link 💥 🗋 EXOSTIV Probe 💥 🗍 FPGA link 💥		

Capture units (1 out of max. 16)





EXOSTIV Dashboard - C:/Users/frederic/new_projec	t.epf				-	
Eile Iools Help						
Link Configuration	Capture Configuration	>>>	Insert EXOSTIV IP	2 2 2	Debug Design	
Capture units (1 out of max. 16)			Data Group 1			
✓ Pattern	Edit Probes					
Sine Double click to add Data Group Double click to add Capture Unit	Signal Names				Data	Trigger
Info : Info : Probe disconnected. Info : Project file "C:/Projects/Xplorer/Hands-On/Demo-MICA Info : Project file "C:/Users/frederic/new_project.epf" writter Netlist flow Vivado link	-3links/demo_mica702-3links.epf ^e written successfully. n successfully.]	og Window				>

7. We now have to connect EXOSTIV Dashboard to Vivado, in order to select the nodes to be observed. Switch back to Vivado and click on 'EXOSTIV Dashboard' shortcut in the main toolbar.

🍌 demo_03 - [P:/Xplorer/hardware/fpga/DEMO_MIC_A7_02 -	- export/impl/demo_03/demo_03.xpr] - Vivado 2016.2	
<u>File Edit Flow Tools Window Layout View Help</u>		
😂 🖩 100 UF 🐚 🛍 🗙 🛷 🗞 🏷 📸 🚰 🥥	🗘 🥝 🍪 💥 ∑ 🎼 📴 Default Layout 🔹) 🗶 🔌 🔭 🔲 😜
Flow Navigator ? «	Synthesized Design - synth_1 xc7a35tcsg325-2 (active)	
	NF-10-1	

8. In the 'EXOSTIV Dashboard Launcher' window, select the running instance of the EXOSTIV Dashboard and click on 'Link to Vivado'. This establishes a link between Vivado and EXOSTIV Dashboard.

EXOSTIV Dashboard Launcher -							×
Link to Vivado Bring To Front New Dashboard Location Network 🔻					Quer	у	
Host Name 8piclt15	IP Address 192.168.0.8		Pid 7968	Project File P:/Xplorer/hardware/fpga/DEMO_MIC_4	\7_02/te	est/demo)_mi



Info : Replied to Vivado query packet. Info : Received link to Vivado request. Info : Successfully connected to Vivado. Info : Acknowledged link to Vivado request.
Netlist flow 🛛 Vivado link 🛩 🗍 EXOSTIV Probe 💥 🗍 FPGA link 💥

9. In EXOSTIV Dashboard, select the 'Sine' data group and click on 'Edit Probes'

EXOSTIV Dashboard - P:/Xplorer/hardware/fpg	a/DEMO_MIC_A7_02/test/demo_mica702-3links.epf	-		×
<u>F</u> ile <u>T</u> ools <u>H</u> elp				
📇 🔚 🛃 🗞 😤 🕷				
Link Configuration	Capture 2 2 2 Run 2 2 2	Debug Design		
Capture units (1 out of max. 16)	Sine			
✓ Pattern	Edit Probes			
Sine Double click to add Data Group	Signal Names	Data	Trigg	ger
Double click to add Capture Unit				

10. From the 'Connect Probes' window, browse the design and select the signal 'u_demo/sine_Data[15:0]

Connect Probes		
Design Hierachy		
✓ demo_03_N		^
g_demo[0].u_core		
> u_color		
> u_noisy		
u_noisy_2		
u_period		
> u_ram		
u_rnd		
> u_rom1		
> u_rom2		
u_sample		
u_shuffle_sync		
> u_sine		<u> </u>
u wr svnc		•••
Filter 🔍 sine_d		
Found Signals		Data Signals
Found Signals sine_Data[3]_i_1_n_0	^	Data Signals g_demo[0].u_core/sine_Data[150]
_	^	-
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0	^	-
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0	^	g_demo[0].u_core/sine_Data[150]
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0	^	-
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0 sine_Data[8]_i_1_n_0	^	g_demo[0].u_core/sine_Data[150]
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0 sine_Data[8]_i_1_n_0 sine_Data[9]_i_1_n_0	^	g_demo[0].u_core/sine_Data[150]
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0 sine_Data[8]_i_1_n_0 sine_Data[9]_i_1_n_0 sine_Data[10]_i_1_n_0	^	g_demo[0].u_core/sine_Data[150]
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0 sine_Data[8]_i_1_n_0 sine_Data[9]_i_1_n_0 sine_Data[10]_i_1_n_0 sine_Data[11]_i_1_n_0	^	g_demo[0].u_core/sine_Data[150] Click on arrow to place signal in the 'Data Signals list'
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0 sine_Data[8]_i_1_n_0 sine_Data[9]_i_1_n_0 sine_Data[10]_i_1_n_0 sine_Data[11]_i_1_n_0 sine_Data[12]_i_1_n_0	Select	g_demo[0].u_core/sine_Data[150]
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0 sine_Data[8]_i_1_n_0 sine_Data[9]_i_1_n_0 sine_Data[10]_i_1_n_0 sine_Data[11]_i_1_n_0 sine_Data[12]_i_1_n_0 sine_Data[13]_i_1_n_0		g_demo[0].u_core/sine_Data[150] Click on arrow to place signal in the 'Data Signals list'
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0 sine_Data[8]_i_1_n_0 sine_Data[9]_i_1_n_0 sine_Data[10]_i_1_n_0 sine_Data[11]_i_1_n_0 sine_Data[12]_i_1_n_0 sine_Data[13]_i_1_n_0 sine_Data[13]_i_1_n_0		g_demo[0].u_core/sine_Data[150] Click on arrow to place signal in the 'Data Signals list'
<pre>sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0 sine_Data[8]_i_1_n_0 sine_Data[9]_i_1_n_0 sine_Data[10]_i_1_n_0 sine_Data[11]_i_1_n_0 sine_Data[12]_i_1_n_0 sine_Data[13]_i_1_n_0 sine_Data[13]_i_0 sine</pre>	Select	g_demo[0].u_core/sine_Data[150] Click on arrow to place signal in the 'Data Signals list'
sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0 sine_Data[8]_i_1_n_0 sine_Data[9]_i_1_n_0 sine_Data[10]_i_1_n_0 sine_Data[11]_i_1_n_0 sine_Data[12]_i_1_n_0 sine_Data[13]_i_1_n_0 sine_Data[13]_i_1_n_0		g_demo[0].u_core/sine_Data[150] Click on arrow to place signal in the 'Data Signals list'
<pre>sine_Data[3]_i_1_n_0 sine_Data[4]_i_1_n_0 sine_Data[5]_i_1_n_0 sine_Data[6]_i_1_n_0 sine_Data[7]_i_1_n_0 sine_Data[8]_i_1_n_0 sine_Data[9]_i_1_n_0 sine_Data[10]_i_1_n_0 sine_Data[11]_i_1_n_0 sine_Data[12]_i_1_n_0 sine_Data[13]_i_1_n_0 sine_Data[13]_i_0 sine</pre>	Select	g_demo[0].u_core/sine_Data[150] Click on arrow to place signal in the 'Data Signals list'

11. If necessary, change the signal filter by clicking on the magnifier icon:



Connect Probes			
Design Hierachy			
✓ demo_03_N			^
g_demo[0].u_core			
> u_color			
> u_noisy			
u_noisy_2			
u_period			
> u_ram			
u_rnd			
> u_rom1			
> u_rom2			
u_sample u_shuffle_pmc			
> u_sing			
u_site			~
		•••	
Found 🖌 All signals		Data Signals	
sit MARK_DEBUG signals	^	g_demo[0].u_core/sine_Dat	a[150]
sii 🗸 Plain text			
si			
siı Plain text with wildcards			
sii Regular expression		>	
sii Case sensitive		>>	
sine_Data[10]_i_1_n_0 sine_Data[11]_i_1_n_0		**	
sine_Data[12]_i_1_n_0		<	
sine_Data[13]_i_1_n_0			
sine_Data[14]_i_1_n_0			
> sine_Data[150]			
sine_Data[15]_i_2_n_0	~		
Number of probes : 16		L	Cancel Done

12. Select 'Pattern' capture unit and click on the '...' button to define the capture unit's sampling clock:

EXOSTIV Dashboard - C:/Projects/mynewpro	ject.epf			– 🗆 X	
<u>F</u> ile <u>T</u> ools <u>H</u> elp					
🖺 🛤 🖬 🖌 🗞 💥 🕷					
Link Configuration	Capture Configuration		Insert OSTIV IP	Debug Design	
Capture units (1 out of max. 16)		Patte	ern		
Y Pattern	Triggering		Data		
Sine Double click to add Data Group	Trigger unit type	Levels / Edges / Comparisons	▼ Fifo depth 1024	-	
Double click to add Data Group Double click to add Capture Unit	Bit operations Bus operations	X, 0, 1, R, F, B, N ==, >, <, >=, <=, <>, in range, out o	Number of data groups 1 out	fmax. 16	
	Counter width	Disabled	Number of data probes 0 out o	f.nax. 2048	
	Sequencer Depth	Disabled	~		
	Storage qualification				
	Number of pipes	Disabled	▼		
	Sampling Clock			. –	
	•				
		Log Window			
Info : Replied to Vivado query packet. Info : Received link to Vivado request.				,	^
Info : Successfully connected to Vivado. Info : Acknowledged link to Vivado request.					
					¥
Netlist flow 🛛 Vivado link 🖋 🛛 EXOSTIV Probe 💥 🗌	FPGA link 💢				



13. Select 'demo_03_N/Clk', bring it to the 'Clock signal' box with the '>' and click on 'Done'.

Connect Probes	
Design Hierachy	
 demo_03_N g_demo[0].u_core u_clk u_reset u_shuffle_sync u_time u_vidrst u_vid_mmcm 	
Filter	
Found Signals	Clock Signal
Clk SysClk vid_Clk	
	Cancel Done

14. Set up 2 additional capture units, with the parameters of the table below (or load the reference project file 'demo_mica702-3links.epf' if you want to skip this step)

	Name	Trigger unit type	Storage qualification	Number of pipes	Fifo depth	Number of data groups	Number of data probes	Sampling Clock
CU1	Pattern	Levels/Edges/Comparisons	NO	Disabled	1024	3	16	Clk
CU2	Video	Levels/Edges/Comparisons	NO	Disabled	2048	2	46	vid_Clk
CU3	Video- Extended	Levels/Edges	NO	Disabled	1024	1	80	vid_Clk

Capture	Data	Nodes	Data	Trigger	
Unit	group				
CU1	Cnt	g_demo[0].u_core/Cnt[150]	YES	YES	
'Pattern'	Sine	g_demo[0].u_core/sine_Data[150]	YES	YES	
	Noise	g_demo[0].u_core/Rnd[150]	YES		
CU2 'Video'	SDI	g_demo[0].u_core/sdi_SOF	YES	YES	
		g_demo[0].u_core/sdi_VBlank	YES	YES	
		g_demo[0].u_core/sdi_HBlank	YES	YES	
		g_demo[0].u_core/sdi_Valid	YES	YES	
		g_demo[0].u_core/sdi_LN[110]	YES	YES	
		g_demo[0].u_core/sdi_R[90]	YES	NO	
		g_demo[0].u_core/sdi_G[90]	YES	NO	
		g_demo[0].u_core/sdi_B[90]	YES	NO	
	Noise	g_demo[0].u_core/vid_Sine[150]	YES	YES	
		g_demo[0].u_core/vid_Noise[170]	YES	NO	
		g_demo[0].u_core/vid_Addr[90]	YES	NO	
CU3 'Video-	Vid-	g_demo[0].u_core/sdi_SOF	YES	YES	
Extended'	Extended	g_demo[0].u_core/sdi_VBlank		YES	
		g_demo[0].u_core/sdi_HBlank		YES	
		g_demo[0].u_core/sdi_Valid		YES	
		g_demo[0].u_core/sdi_LN[110]		YES	
		g_demo[0].u_core/sdi_R[90]		NO	
		g_demo[0].u_core/sdi_G[90]		NO	



g_demo[0].u_core/sdi_B[90]	NO
g_demo[0].u_core/vid_Sine[150]	NO
g_demo[0].u_core/vid_Noise[170]	NO

✓ CAPTURE CONFIGURATION: DONE !

(! Don't forget to save your project !)



Step 3: Run Insertion

This part of the flow executes the following:

- IP configuration checks
- EXOSTIV IP synthesis with Vivado
- EXOSTIV IP insertion in the target FPGA netlist
- Additional constraints generation
- Instrumented design implementation and bitstream generation.
- 1. Click on 'Insert EXOSTIV IP'

EXOSTIV Dashboard - C:/Projects/	mynewproject.epf							_		
<u>T</u> ools <u>H</u> elp										
2 & 🖬 🖬 🗧	*									
Link Configuration	> > >	Capture Configuration	> >	2	Insert EXOSTIV IP	2	2 2	Debug Design		
Insert EXOSTIV IP	(/ivado/2016.2/bin	Use this but	ton to star	Output	neration and	l insertior) proces	5		
Jse IP cache Manage		~				1				
rogress										
Checking configuration	The path to	the Vivado ex	ecutable n	nust be spe	cified	-				
Starting Vivado shell		ing the IP gen				- 1				
Creating debug core project		0				-				
Configuring debug core					Constant and	-				
					Shocity th		the EXC	STIV IP inst	ance	
Generating memories					Speenyth	e name of				
Synthesising debug core					Speenyth	- Hanne Of				
-					Speeny in	-				
Synthesising debug core					Specify in	- - -				
Synthesising debug core Inserting debug core					Specify th	- - - -				
Synthesising debug core Inserting debug core Connecting probes					Specify th	- - - - - -				
Synthesising debug core Inserting debug core Connecting probes Generating constraints	1	-			Specify th	- - - - - -				
Synthesising debug core Inserting debug core Connecting probes Generating constraints Design rules check	1	•			Specify th	- - - - - - -				
Synthesising debug core Inserting debug core Connecting probes Generating constraints Design rules check Implement design impl	1	•	Log Win	idow						
Synthesising debug core Inserting debug core Connecting probes Generating constraints Design rules check Implement design Generate bitstream Replied to Vivado query packet.	1	•		ıdow						
Synthesising debug core Inserting debug core Connecting probes Generating constraints Design rules check Implement design Impl Generate bitstream	_1	•		ıdow						

- 2. Enable Implementation and select 'impl_1' from the drop down list: Select 'Implement Design' and 'Generate bitstream'.
- 3. Click on 'Insert EXOSTIV IP' button.

Insertion running ... and completed.



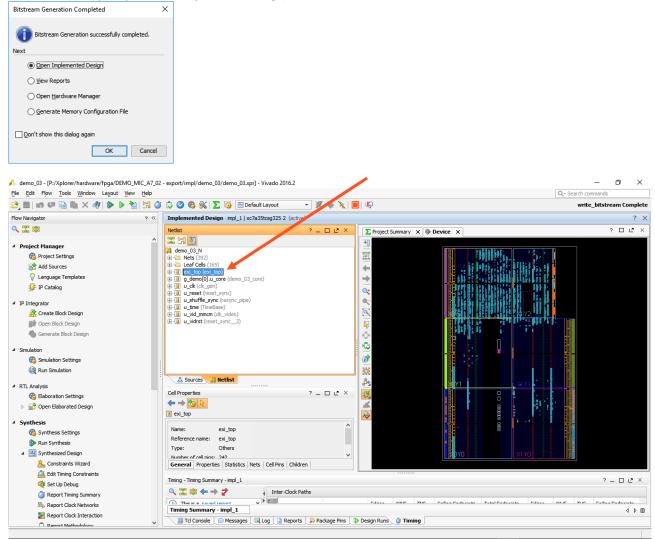
	CA-3links/demo_mica702-3links.epf			- 🗆
<u>T</u> ools <u>H</u> elp				
, 🔄 🖬 😼 💲 💥				
Link Configuration	Capture Configuration	Insert EXOSTIV IP	>>>	Debug Design
Cancel Insertion				
onfiguration		Output		
vado installation folder C:/cad/Xilinx/Vivado/2016.2/bin		EXOSTIV IP instance name exi_top		
se IP cache 🗹 Manage Cache				
ogress				
Checking configuration			Done (0:00:02)	
Starting Vivado shell			Done (0:00:07)	
Creating debug core project Configuring debug core			Done (0:00:01) Done (0:00:00)	
Generating memories			0:00:00	
Synthesising debug core			-	
nserting debug core				
Connecting probes				
Generating constraints			-	
Design rules check			-	
Implement design impl_1			-	
Generate bitstream				
		•••		
NFO: [IP_Flow 19-234] Refreshing IP repositories	Log	Window		
NFO: [IP_Flow 19-1704] No user IP repositories specified				
ini 0, [Ir_1 low Is-1704] no user Ir repositories specified				
flow 🛛 Vivado link 🛩 🗍 EXOSTIV Probe 💥 🗍 FPGA link 💥				
<u>T</u> ools <u>H</u> elp				—
📴 🖬 🛃 🗞 😤 💥	Capture) Insert	>>>	Debug
🛤 🖬 📓 💲 🛣 🕷	Capture Configuration	Insert EXOSTIV IP	>>>	
E S S S	Capture Configuration	Insert EXOSTIV IP	>>>	Debug
Image: Second second	Capture Configuration	Dutput	> > >	Debug
Insert EXOSTIV IP	Configuration	Output	> > >	Debug
Insert EXOSTIV IP	Capture Configuration		> > >	Debug
Image: Second system Image: Second system Insert EXOSTIV IP Infiguration ado installation folder C://cad/Nilmx/Vivado/2016.2/bin	Configuration	Output	> > >	Debug
Image Cache	Configuration	Output	> > >	Debug
Image Cache Image Cache	Configuration	Output	Done (0:00:02)	Debug
Image Cache Insert EXOSTIV IP	Configuration	Output EXOSTIV IP instance name exi_top		Debug
Image Cache Insert EXOSTIV IP Insert EXOSTIV IP Infiguration rado installation folder C:/cad/Xilmx/Nvado/2016.2/bin e IP cache Image Cache orgeess Checking configuration itarting Vivado shell	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02)	Debug
Image Cache Insert EXOSTIV IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert IP Insert IP Insert IP Insert IP Inseret IP Insert IP	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07)	Debug
Image Cache Insert EXOSTIV IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE IP Insert EXOSTIVE	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:01)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:01) Done (0:00:00)	Debug
Image Cache Insert EXOSTIV IP Insert EXOSTIVE Insert EXOSTIV	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:01) Done (0:00:00) Done (0:00:03)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert ISSUE Insert ISSUE ISSU	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:01) Done (0:00:00) Done (0:00:03) Done (0:03:59)	Debug
Image: Second	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:01) Done (0:00:00) Done (0:00:03) Done (0:00:359) Done (0:00:13)	Debug
Image: Configuration Image: Configuration Insert EXOSTIV IP Infiguration radio installation folder c:/cad/Nilmx/Ivvado/2016.2/bin e IP cache Image: Cache agress Checking configuration starting Vivado shell Creating debug core project Configuring debug core Gonguering debug core Synthesising debug core Configuring lebug core Configuring debug core Configuring debug core Configuring debug core Configuring debug core Conserting probes Generating constraints	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:01) Done (0:00:00) Done (0:00:03) Done (0:00:359) Done (0:00:13) Done (0:00:05)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIVE Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Inser	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:01) Done (0:00:00) Done (0:00:03) Done (0:00:359) Done (0:00:13) Done (0:00:05) Done (0:00:19)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP Insert Inserting debug core Inserting debug core Inserting debug core Inserting debug core Inserting debug core Inserting debug core Inserting rules check	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:07) Done (0:00:00) Done (0:00:03) Done (0:00:39) Done (0:00:13) Done (0:00:13) Done (0:00:05) Done (0:00:19) Done (0:00:03)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP In	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:07) Done (0:00:00) Done (0:00:03) Done (0:00:39) Done (0:00:13) Done (0:00:13) Done (0:00:05) Done (0:00:19) Done (0:00:03)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert I EXOSTIV I I I I I I I I I I I I I I I I I I	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:07) Done (0:00:00) Done (0:00:03) Done (0:00:39) Done (0:00:13) Done (0:00:13) Done (0:00:05) Done (0:00:19) Done (0:00:03)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert IS Inse	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:07) Done (0:00:00) Done (0:00:03) Done (0:00:39) Done (0:00:13) Done (0:00:13) Done (0:00:05) Done (0:00:19) Done (0:00:03)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert IS Inse	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:07) Done (0:00:00) Done (0:00:03) Done (0:00:39) Done (0:00:13) Done (0:00:13) Done (0:00:05) Done (0:00:19) Done (0:00:03)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert IS Inse	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:07) Done (0:00:00) Done (0:00:03) Done (0:00:39) Done (0:00:13) Done (0:00:13) Done (0:00:05) Done (0:00:19) Done (0:00:03)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert ISO Installation folder Iso Installation folder Iso Installation folder Iso Inserting Configuration Iso Inserting debug core Iso Inserting debug core Iso Inserting debug core Iso Inserting debug core Iso Inserting debug core Iso Inserting debug core Iso Inserting debug core Iso Inserting debug core Iso Inserting debug core Iso Inserting Iso Iso Iso Iso Iso Iso Iso Iso Iso Iso	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:07) Done (0:00:00) Done (0:00:03) Done (0:00:39) Done (0:00:13) Done (0:00:13) Done (0:00:05) Done (0:00:19) Done (0:00:03)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert EXOSTIV IP Insert ISO Installation folder Iso Inserting Configuration Istarting Vivado shell Iso Istarting Vivado shell Iso Istarting Vivado shell Iso Istarting Vivado shell Iso Istarting Vivado shell Iso Istarting Vivado shell Iso Istarting Vivado shell Iso Istarting Vivado Istallation Istarting Vivado Istall Iso Istarting Vivado Istall Iso Istarting Vivado Istall Iso Istarting Vivado Istall Iso Istallation Istarting Vivado Istall Iso Istallation Istallation Is	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:07) Done (0:00:00) Done (0:00:03) Done (0:00:39) Done (0:00:13) Done (0:00:13) Done (0:00:05) Done (0:00:19) Done (0:00:03)	Debug
Insert EXOSTIV IP Insert EXOSTIV IP In	Configuration	Output EXOSTIV IP instance name exi_top	Done (0:00:02) Done (0:00:07) Done (0:00:07) Done (0:00:00) Done (0:00:03) Done (0:00:39) Done (0:00:13) Done (0:00:13) Done (0:00:05) Done (0:00:19) Done (0:00:03)	Debug



Click on 'OK' at prompt to save the project.

Auto Save Project	\times
The project file must be saved when EXOSTIV IP is inserted. This ensures a project file is uniquely linked to the inserted IP.	
Automatically save the project file.	
Cancel OK	

We can check that 'exi_top' was inserted into the design if you open the 'design implementation' in Vivado (Don't save when prompted to save synthesized design):





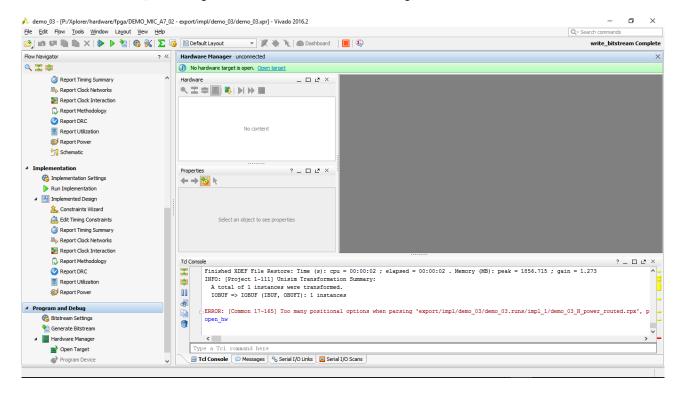
demo_03 - [P:/Xplorer/hardware/fpga/DEMO_MIC_A7_02 - export/impl/demo_03/demo_03.xpr] - Vivado 2016.2
 File Ent Flow Tools Window Lawout View Help

🖻 🗎 🗠 🕫 🐚 🦍 X 🛷 🕽	🎦 🖄	🗘 🙆 🆓 📡 👩 😬 Default Layout 💿 🗴 🕷 🔭 🚺		write_bitstream Complet
Flow Navigator	? «	Implemented Design - impl_1 xc7a35tcsg325-2 (active)		
Q 🔀 🖨				
		Netlist ? _ 🗆 🗠 ×	∑ Project Summary × ⊗ Device × 🖻 Schematic × 🖉 Schematic (2) ×	? 🗆 🖻 ×
Project Manager	^	X 🗄 🖪	<u> <u> </u> <u> <u>171 Cells</u> <u>641/O Ports</u> <u>392 Nets</u> </u></u>	
Project Hanager		M demo_03_N	time_tick_out_reg	1
······				
dd Sources		exi_top (exi_top)	CE	_time_tick_out_OBUF_inst
💡 Language Templates		iii. iii. Nets (239)		
IP Catalog				OBUF
		B - 3 u_dk (dk_gen)	FDRE	
4 IP Integrator		u_reset (reset_sync)	exi top	
🎇 Create Block Design		u_shuffle_sync (resync_pipe)		
📑 Open Block Design		B-B u_time (TimeBase) B-B u vid mmcm (clk video)		
🍓 Generate Block Design				
			cu0_ing0_i15:0	
4 Simulation			cu0 Trig[2][15:0]	
🍪 Simulation Settings			cu1_Data[0][29:0]	
🔍 Run Simulation			cu1_Data[1][29:0] SDA	_exi_top_gt_TX_1_obuf_n
		👃 Sources 🕅 Netlist		
 A RTL Analysis 		Cell Properties ? _ D L ^a ×		OBUF exi top gt_TX_0_obuf_n
Elaboration Settings		← → ¹ / ₁	cu2_Trig0115:01	
Open Elaborated Design			at. RX. n(2:0)	OBUF
A Combinete		exi_top	at RX p[2:0]	exi_top_gt_TX_2_obuf_n
4 Synthesis		Name: exi top	ot RefCik p	
🔞 Synthesis Settings		Reference name: exi top	exi top	OBUF exi_top_gt_TX_0_obuf_p
📎 Run Synthesis		Type: Others	ex_top	
4 🔛 Synthesized Design		Number of cell pips: 242	T	OBUF
🚵 Constraints Wizard		General Properties Statistics Nets Cell Pins Children		
🚵 Edit Timing Constraints				
🐺 Set Up Debug		Timing - Timing Summary - impl_1		? _ 🗆 🖻 ×
🖉 Report Timing Summary		🔍 🛣 🚔 🔶 🥇 🕴 🕴 Inter-Clock Paths		
Report Clock Networks		(i) This is a caved report	edite unio mio estis-estatis estatestatis edita unio	man matter made to a
		Timing Summary - impl_1		4 0 0
Report Clock Interaction		🔚 Tcl Console 📁 Messages 🖾 Log 📄 Reports 🔎 Package Pins 📑	Design Runs di Timing	
Penort Methodology	v			

Program the target board

We now need to use the generated binary to program the target board.

- Return to Vivado and open the 'Hardware Manager'.
- Connect the programming cable to the MICA board
- Turn on the MICA board by enabling the output HDMI power and connecting to the EXOSTIV Probe.
- Once connected, right-click on the target FPGA and select 'Program'.
- Then, select the generated bit file and load the FPGA configuration.





Creating a 'RTL flow' project with EXOSTIV Dashboard

The 'RTL flow' enables configuring and synthesizing EXOSTIV IP for insertion in RTL source code. The 'RTL flow' is somewhat simpler than the 'netlist flow' as it does not involve user interactions with Vivado. EXOSTIV IP is inserted and connected to the target FPGA EXOSTIV design nodes by instantiating IP in the RTI code 'manual' process. _ а

1. From the Welcome screen, click on 'Create New Project' The 'Create New Project' window appears, prompting to select the desired flow (Netlist flow or RTL flow). Specify a project name and pick a location. The 'Link Configuration window' of the core inserter appears. EXOSTIV Dashboard × <u>File Tools H</u>elp 🗟 📄 🖬 🛃 🔌 😤 🗰 EXOSTIV[™] Reach. Store. Analyze. FPGA Debug Reloaded Create New Project User Guide / Getting Started A wizard will help you create a new Detailed information about EXOSTIV. project Open Project Knowledge Base Open an existing project. Online knowledge base. Open Recent Project Open an recently used project. lanage licenses. 🔲 New Project ? × Create New Project **RTL IP insertion** • Type The EXOSTIV IP is configured and synthesized based on the number of nodes to be connected from the target FPGA design. EXOSTIV IP is synthesized and provided as a netlist and a RTL component / module. The target RTL code must be manually edited to insert and connect the EXOSTIV IP. Name new_project Create in C:/Users/frederic Browse... Cancel Create



	board - C:/Users/frede/new_	project.epf						_	×
File Tools Hel	p								
	ink yuration	Σ	Capture Configuration	2.2	2 2	Generate EXOSTIV IP	>>>	Debug Design	
FPGA Type			Connector						
Family A	rtix-7 - Defense-grade	•	Connector type	HDMI	•				
Package c	\$325	-							
Speed grade -:	11	•							
Part x	q7a50tcs325-1I	-							
Upstream Link			Downstream Link						
Transceiver bar	ık 216	-	• Use I2C link	O Use transceive	er link				
MGT type	GTP		SCL package pin	A9	~				
MGT_TxP0	H2		SDA package pin	A9	~				
MGT_TxP1	F2		SCL I/O standard	LVCMOS33	•				
MGT_TxP2	D2		SDA I/O standard	LVCMOS33	•				
MGT_TxP3	B2								
Reference Clock	۲								
Transceiver bar	k 216	-							
MGT_REFCLK_P	0 D6								
MGT_REFCLK_P	1 B6								
Frequency (MH:	z) 100	2							
	Range : 60 MHz to 660 №	Hz							
Line rate (Gb/s)		-							
Link rate (Gb/s)									
PLL type used EXOSTIV clock of	QPLL putput								
				Log W	indow				
Info : Project file "D:	/Projects/Xplorer/Hands-On/De /Users/frede/new_project.epf	mo-MICA-3lin	ks/demo_mica702-3link	-					^
Info : Project file "D:	/Projects/Xplorer/Hands-On/De	mo-MICA-3link	ks/demo_mica702-3link	s-1.6.1.epf" loade	d successfully.				
Into : Project file "C:	/Users/frede/new_project.epf"	written succe	esstully.						~
TI flow	k 🕱 🛛 EXOSTIV Probe 🕱 🗍	EDCA liek 👻							-

Using the IP generator (RTL flow)

In RTL flow, the EXOSTIV Dashboard allows to **generate EXOSTIV IP.** Unlike the 'netlist flow', RTL flow won't automatically insert EXOSTIV IP into the target design. This has to be done 'manually', in the RTL code (VHDL / Verilog).

Generating EXOSTIV IP in RTL flow requires 3 steps:

- Step 1 : Link Configuration
- Step 2 : Capture Configuration
- Step 3 : Generate EXOSTIV IP

These 3 steps are accessible through the top flow overview in the EXOSTIV Dashboard window

_							
	Link Configuration	<u>>>></u>	Capture Configuration	> > >	Generate EXOSTIV IP	2.2.2	Debug Design
						-	



Step 1: Link Configuration

Tools <u>H</u>	lelp	· ¥							
Con	Link figuration	<u>></u> >>	Capture Configuration	> >	Σ	Insert EXOSTIV IP	>>>	Debug Design	
PGA Type			Connector						
amily	Artix-7	•	Connector type	HDMI	-				
ackage	csg325	•							
peed grade	-2	•							
Part	xc7a35tcsg325-2	•							
pstream Link			Downstream Link						
ransceiver b	ank 216	-	• Use I2C link	O Use transceiver	link				
IGT type	GTP		SCL package pin	M2	~				
IGT_TxP0	H2		SDA package pin		~				
IGT_TxP1	F2		SCL I/O standard		•				
IGT_TxP2	D2		SDA I/O standard		-				
4GT_TxP3	B2			w 2.5V requires an e	xternal				
Reference Clo Fransceiver b	ank 216	•							
IGT_REFCLK	_								
requency (M	-								
requency (n		Hz to 660 MHz							
ine rate (Gb/	/s) 5	-							
ink rate (Gb/	s) 15								
LL type used XOSTIV dock									
				Log Win					
Project file "I	D:/Projects/Xplorer/Ha	nds-On/Demo-MICA-3lin roject.epf" written succe	ks/demo_mica702-3link	s-1.6.1.epf" loaded	successfully.				
Designation of the Pro-	D:/Projects/Xplorer/Ha	Lo b arros at	1 (1 · · · · · · · · · · · · · · · · · ·		C H L				

This step in RTL flow does not differ from the same step in netlist flow. Please refer to Step 1: Link Configuration, page 23 for an illustrated example on how to use it.

Step 2 : Capture Configuration

During this step, the characteristics of EXOSTIV IP are defined:

- Capture units
 - Number and names
 - Trigger options
 - \circ Size of the memory buffer

- Data groups

0

- Number and names
 - As opposed to the 'netlist flow' described at page (-), data group definition is limited to:
 - The number of bits in each data group
 - The quality of the inputs of each data group: the number of 'data / trigger' nodes and the number of 'data only' nodes.

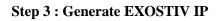


Capture units (3 out of max. 16)		Pattern	l i i i i i i i i i i i i i i i i i i i	
Pattern Sine	Triggering Trigger unit type	Levels / Edges / Comparisons	Data Fifo depth	1024
Cnt Rnd	Bit operations Bus operations	X, 0, 1, R, F, B, N ==, >, <, >=, <=, <>, in range, out of range	Number of data groups	3 out of max. 16
Double click to add Data Group Video	Counter width Sequencer Depth	Disabled V	Number of data probes	16 out of max. 2048 0 out of 16
SDI Noisy Sine	Storage qualification			
Double click to add Data Group	Number of pipes	Disabled 🔻		
Video-ext Data Group 1 Double click to add Data Group				
Double click to add Capture Unit	1			

Use this panel to define the Capture units and the data groups.

EXOSTIV Dashboard - C:/Projects/Xplore Eile <u>I</u> ools <u>H</u> elp	er/Hands-On/RTLIP/DebugCoreTo	op.epf			- 0	×				
🎒 🔚 🔚 😼 😤 🗰										
Link Configuration	Capture Configuration	<u> </u>	Generate EXOSTIV IP	2 2 2	Debug Design					
Capture units (3 out of max. 16)			Pattern							
 Pattern Sine Cnt Rnd Double click to add Data Group Video SDI Noisy Sine Double click to add Data Group Video-ext Data Group 1 Double click to add Data Group Double click to add Data Group 	•	Levels / Edges / Comparisons X, 0, 1, R, F, B, N ==, >, <, >=, <=, <>, in range Disabled Disabled Disabled Disabled Disabled	* *	Number of data probes 16 out of nax.						
Use this panel to define the capture units and the data groups.	the data	groups.	For each - - -	capture unit, define The size of the mer The number of data The number of data Optionally, the nur that won't be used	nory buffer in th a groups a probes nber of data pro	obes				
<						>				
		Log Window								
Info : Project file "C:/Users/frederic/new_project Info : Project file "C:/Projects/Xplorer/Hands-On/ Info : Project file "C:/Users/frederic/new_project Info : Project file "C:/Projects/Xplorer/Hands-On/	Demo-MICA-3links/demo_mica702-3 .epf" written successfully.	links.epf" loaded successfully.				~				
RTL flow 🛛 Vivado link X 🗍 EXOSTIV Probe X	FPGA link 样					.:				





Tools Help	*											
Link Configuration	>>>	Capture Configuration	> >	2	Genera EXOSTI		>	> >		Debug Design		
Generate EXOSTIV IP	Click here to	o start the gene	ration	orocess								
onfiguration				Output					_		_	
vado installation folder C:/cad/Xilinx	/Vivado/2016.2/bin			EXOSTIV IP ins	tance name	DebugC	oreTop					_
se IP cache 🗹 Manage	Cache			Output folder	4	C:/Proje	cts/Xplorer/Ha	ands-On/RTLI	P/DebugCore	Top-Files		
ogress												•
Checking configuration	/				/		-					-
Starting Vivado shell					/		-					
Creating debug core project		Specify	the n	ame pref	ix and	the	location	where	the			
Configuring debug core		generate	ed IP fil	es should l	be store	d	-					
Generating memories							-					
Synthesising debug core							-					
Generating constraints	Path to Vivado	executable (use	d to sv	nthesize E		IP)	-					
Exporting files		executione (use	.u to sy	Inthe Size E	00110	,	-					
							1	low pro	gress			
			Log W	indow				_				
Project file "C:/Users/frederic/new_p	roject.epf" written success	fully.	_									
Project file "C:/Projects/Xplorer/Hand Project file "C:/Users/frederic/new_p	is-On/Demo-MICA-3links/de roject.epf" written success	emo_mica702-31inks.ept*1 fully:	oaded succ	esstully.								



EXOSTIV Dashboard -	C:/Projects/Xplor	er/Hands-On/F	TLIP/DebugCore	Top.epf									_		Х
			-												
🗟 📄 📕 🖌	8 😒 🕷														
Link Configuration	2	2.2	Capture Configuratio	'n	2	2 2	Genera EXOSTI			> >	>		Debug Design		
Cancel Gene	eration														
Configuration						Output									
Configuration Output Vivado installation folder C:/cad/Xilinx/Vivado/2016.2/bin EXOSTIV IP instance name DebugCoreTop															
Use IP cache	Manage Cach					Output folder				r/Hands-O	n/RTLIP	/DebugCo	oreTop-Files		
Progress															
	20								Don	e (0:00:01)					
Checking configuration Starting Vivado shell	/11									e (0:00:01) e (0:00:07)					
Creating debug core	project									e (0:00:07) e (0:00:00)					
Configuring debug co										e (0:00:00)					
Generating memories										e (0:00:15)					
Synthesising debug c									C	0:03:05					
Generating constraint	s									-					
Exporting files										-					
exi_mem_read.vhd C:/Users/ Users/frederic/AppData/Local Temp/ExostivDashboard-bE7 rrc/exi_pointer_fifo_wrapp E2T#d/src/exi_pointer_fifo_ exi_rx_decoder.vhd C:/Users exi_rx_msg_ram_wrapper_xii w_i_shift_rear_d.vhd C:/Users RTL flow Vivado link X 1	/Temp/ExostivDasl Fd/src/exi_pkg.vh er_xil.vhd C:/User !_wrapper_xil.vhd /frederic/AppData, .vhd C:/Users/fred s/frederic/ApnData	board-bE7tFd/s d C:/Users/frede s/frederic/AppDa C:/Users/frederic Local/Temp/Exos eric/AppData/Lo	rc/exi_mux161.vh ric/AppData/Local, ta/Local/Temp/Exi c/AppData/Local/T stivDashboard-bE7 cal/Temp/ExostivD istivDashboard-bE	d C:/Users/ Temp/Exos ostivDashbo emp/Exostiv 'tFd/src/exi ashboard-b	mem_write. frederic/App tivDashboar bard-bE7tFd vDashboard _rx_msg_bu E7tFd/src/e	Data/Local/Tem rd-bE7tFd/src/ex /src/exi_pointer -bE7tFd/src/exi_ uffer.vhd C:/Use exi_shift_reg.vhc	np/ExostivDask xi_pkt_encode _fifo_1_wrapp _registers.vhd ers/frederic/Ap d C:/Users/fre	hboard-bE7 r.vhd C:/U ber_xil.vhd l C:/Users/t ppData/Loc deric/AppD	7tFd/src/e sers/fred C:/Users frederic// al/Temp/l ata/Loca	exi_mux81. leric/AppDa :/frederic/A AppData/Lo ExostivDas I/Temp/Exo	.vhd C:, ata/Loca AppData ocal/Ten hboard ostivDas	/Users/fre al/Temp/E> a/Local/Ter np/Exostiv -bE7tFd/sr shboard-bi	deric/AppDa xostivDashb mp/ExostivE Dashboard- rc/ E7tFd/src/	ata/Loca ioard-bE iashboar ibE7tFd/	l/ 7tFd/ rd- src/
Ince started, the I	P generatio	n progress	is displaye	d in the	e windo	w. Messag	ges from	the ru	nning	sessio	n of '	Vivado	o appea	r in t	he
og Window. Wait		ocess is ov	er.									_			
Configuring debug co	re								Done	(0:00:00)					
Generating memories									Done	(0:00:15)					
Synthesising debug co	ore		EXOST	'IV Dashb	oard			×	Done	(0:07:32)					
Generating constraint	5								Done	(0:00:00)					
Exporting files EXOSTIV IP generation completed successfully. Done (0								(0:00:00)							
								_							



Overview of the files generated with the RTL flow

A series of files are generated by the EXOSTIV Dashboard RTL flow IP generation process. These files must be used with the target FPGA design in order to instrument the target FPGA and use EXOSTIV to debug the FPGA.

All the files names are prefixed with the EXOSTIV IP instance name specified in the project. Here is the list of generated files and their usage:

File name	Usage	Add to Vivado project?							
<instance name="">.edf</instance>	EXOSTIV IP synthesized netlist.	YES							
<instance name="">.vhd</instance>	Example template on how to instantiate the EXOSTIV IP in the target design.	NO this is an example template.							
<instance name="">_pinout.xdc</instance>	Constraint file containing the pinout required by the EXOSTIV IP (like transceiver location).	YES							
<instance name="">_pkg.vhd</instance>	VHDL Package file containing types used for the EXOSTIV IP instantiation.	YES							
<instance name="">_timing.xdc</instance>	Constraint file containing the timing constraints relative to EXOSTIV IP.	NO, should be 'sourced' in Vivado, not just added.							

RTL flow: inserting EXOSTIV IP and implementing the design

From here, the RTL code of the target design has to be modified manually. An instance of EXOSTIV IP has to be created. The connection with the design internal nodes should be done at this step, from the RTL code.

Then the required files should be added to the Vivado project and the synthesis / implementation / bitstream generation of the instrumented design can be run from the Vivado interface.

Finally, the bitstream has to be used with the target board, and EXOSTIV analyzer can be used for extracting trace data (refer to page 12).



Copyright

© Byte Paradigm sprl 2017, Exostiv Labs™, the Exostiv Labs logo, EXOSTIV™ and MYRIAD™ are trade names and/or trademarks of Byte Paradigm sprl. All rights reserved. Other brands and names mentioned in this document may be the trademarks of their respective owners.

Byte Paradigm sprl is a company registered in Belgium, 18 Avenue Molière, 1300 Wavre. VAT / REG nr: BE0873.279.914.

Disclaimer

THIS DOCUMENT IS PROVIDED "AS IS". EXOSTIV LABS PROVIDES NO REPRESENTATIONS AND NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NON-INFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO THE DOCUMENT. For the avoidance of doubt, EXOSTIV LABS makes no representation with respect to, and has undertaken no analysis to identify or understand the scope and content of, third party patents, copyrights, trade secrets, or other rights.

This document may include technical inaccuracies or typographical errors.

The contents of this document are subject to change without notice. This document may contain information on a Exostiv Labs product under development by Exostiv Labs. Exostiv Labs reserves the right to change or discontinue work on any product without notice.

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL EXOSTIV LABS BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF EXOSTIV LABS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Exostiv Labs products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Exostiv Labs products in such critical applications.

http://www.exostivlabs.com