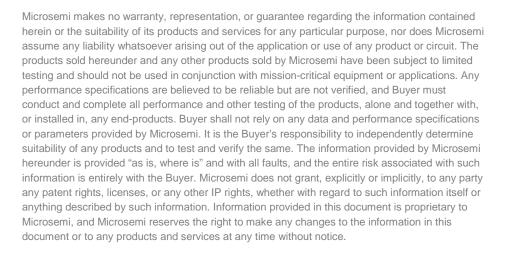
# **UG0773 User Guide PolarFire FPGA SmartDebug**

NOTE: PDF files are intended to be viewed on the printed page; links and cross-references in this PDF file may point to external files and generate an error when clicked. View the online help included with software to enable all linked content.







#### About Microsemi

Microsemi Corporation (NASDAQ: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions; security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, California, and has approximately 4,800 employees globally. Learn more at www.microsemi.com.

5-02-00773-3/12.17



Microsemi Corporate Headquarters One Enterprise, Aliso Viejo, CA 92656 USA Within the USA: +1 (800) 713-4113 Outside the USA: +1 (949) 380-6100 Fax: +1 (949) 215-4996 Email: sales.support@microsemi.com www.microsemi.com

©2017 Microsemi Corporation. All rights reserved. Microsemi and the Microsemi logo are registered trademarks of Microsemi Corporation. All other trademarks and service marks are the property of their respective owners.



# **Table of Contents**

Welcome to SmartDebug	6
Introduction to SmartDebug	6
Getting Started with SmartDebug	7
Using SmartDebug	7
Create Standalone SmartDebug Project	8
SmartDebug User Interface	
Standalone SmartDebug User Interface	
Programming Connectivity and Interface	
View Device Status	16
Debugging	19
Debug FPGA Array	
Hierarchical View	
Netlist View	
Live Probes	
Active Probes	
Probe Grouping (Active Probes Only)	
Memory Blocks	
Probe Insertion (Post-Layout)	
Debug sNVM	
Debug Transceiver	
Debug uPROM	
SmartDebug Tcl Commands	63
SmartDebug Tcl Support	63
add_probe_insertion_point	64
add_to_probe_group	64
create_probe_group	65
delete_active_probe	65
get_programmer_info	
load_active_probe_list	
loopback_mode	
move_to_probe_group	66
program_probe_insertion	67
read_active_probe	
read_lsram	
read_snvm_memory	
read_usram	
remove_from_probe_group	
remove_probe_insertion_point	70



save_active_probe_list	71
select_active_probe	71
set_live_probe	72
smartbert_test	72
static_pattern_transmit	74
ungroup	75
unset_live_probe	75
uprom_read_memory	75
write_active_probe	76
write_lsram	76
write_usram	77

## Frequently Asked Questions ......79

79
79
80
81
82
83



# Welcome to SmartDebug

## Introduction to SmartDebug

Design debug is a critical phase of FPGA design flow. Microsemi's SmartDebug tool complements design simulation by allowing verification and troubleshooting at the hardware level. SmartDebug provides access to non-volatile memory (sNVM), SRAM, transceiver, uPROM, and probe capabilities. Microsemi PolarFire FPGA devices have built-in probe logic that greatly enhance the ability to debug logic elements within the device. SmartDebug accesses the built-in probe points through the Active Probe and Live Probe features, which enables designers to check the state of inputs and outputs in real-time without re-layout of the design.

### **Use Models**

SmartDebug can be run in the following modes:

- Integrated mode from the Libero Design Flow
- Standalone mode
- Demo mode

### **Integrated Mode**

When run in integrated mode from Libero, SmartDebug can access all design and programming hardware information. No extra setup step is required. In addition, the Probe Insertion feature is available in Debug FPGA Array.

To open SmartDebug in the Libero Design Flow window, expand **Debug Design** and double-click **SmartDebug Design**.

### **Standalone Mode**

SmartDebug can be installed separately in the setup containing FlashPro, FlashPro Express, and Job Manager. This provides a lean installation that includes all the programming and debug tools to be installed in a lab environment for debug. In this mode, SmartDebug is launched outside of the Libero Design Flow. When launched in standalone mode, you must to go through SmartDebug project creation and import a Design Debug Data Container (DDC) file, exported from Libero, to access all debug features in the supported devices.

**Note**: In standalone mode, the Probe Insertion feature is not available in FPGA Array Debug, as it requires incremental routing to connect the user net to the specified I/O.

### **Demo Mode**

Demo mode allows you to experience SmartDebug features (Active Probe, Live Probe, Memory Blocks, SERDES) without connecting a board to the system running SmartDebug.

**Note:** SmartDebug demo mode is for demonstration purposes only, and does not provide the functionality of integrated mode or standalone mode.

Note: You cannot switch between demo mode and normal mode while SmartDebug is running.



### **Standalone Mode Use Model Overview**

In the main use model for standalone SmartDebug, the DDC file must be generated from Libero and imported into a SmartDebug project to obtain full access to the device debug features. Alternatively, SmartDebug can be used without a DDC file with a limited feature set.

### Supported Families, Programmers, and Operating Systems

Programming and Debug: PolarFire

Programmers: FlashPro3, FlashPro4, and FlashPro5

Operating Systems: Windows 7, Windows 10, RHEL 6.x, RHEL 7.x, Cent OS 6, and Cent OS 7

# Getting Started with SmartDebug

This topic introduces the basic elements and features of SmartDebug. If you are already familiar with the user interface, proceed to the Solutions to Common Issues Using SmartDebug or Frequently Asked Questions sections.

SmartDebug enables you to use JTAG to interrogate and view embedded silicon features and device status. SmartDebug is available as a part of the FlashPro programming tool.

See Using SmartDebug for an overview of the use flow.

You can use the debugger to Get device status and view diagnostics.

## Using SmartDebug

The most common flow for SmartDebug is:

- 1. Create your design. You must have a FlashPro programmer connected to use SmartDebug.
- 2. Expand **Debug Design** and double-click **Smart Debug Design** in the Design Flow window. SmartDebug opens for your target device.
- 3. Click **View Device Status** to view the device status report and check for issues.
- 4. Examine individual silicon features, such as FPGA debug.

### **Demo Mode**

The following example shows SmartDebug in demo mode.



SmartDebug (DEMO MODE)	- 0	×
<u>File View H</u> elp		
Device: MPF300TS_ES (MPF300TS_ES)	Programmer: simulation (simulation)	~
* SMARTDEBUG IS R	RUNNING IN DEMO MODE *	
ID code read from device: HARDWARE NOT CO	NNECTED	
View Device Status	Debug FPGA Array	
	Debug TRANSCEIVER	
Log		ē ×
🗐 Messages 🔞 Errors 🗼 Warnings 🌒 Info		1
		_

# Create Standalone SmartDebug Project

A standalone SmartDebug project can be configured in two ways:

- Import DDC files exported from Libero
- Construct Automatically

From the SmartDebug main window, click **Project** and choose **New Project**. The Create SmartDebug Project dialog box opens.



😟 Create	SmartDebug Project			×
Name:	sdebug1			
Location:	C:/Users			
	ct JTAG chain for the p ected programmers:	roject S201YQST1V	▼ Refresh	
		ı/negedgedk/2048_18_1024_36_v/srcs/R be imported with JTAG chain	RAM_Logical_View.ddc	
0	Construct Automatically	]		

Figure 1 · Create SmartDebug Project Dialog Box

### Import from DDC File (created from Libero)

When you select the **Import from DDC File** option in the Create SmartDebug Project dialog box, the Design Debug Data of the target device and all hardware and JTAG chain information present in the DDC file exported in Libero are automatically inherited by the SmartDebug project. The programming file information loaded onto other Microsemi devices in the chain is also transferred to the SmartDebug project.

Debug data is imported from the DDC file (created through Export SmartDebug Data in Libero) into the debug project, and the devices are configured using data from the DDC file.

### **Construct Automatically**

When you select the **Construct Automatically** option, a debug project is created with all the devices connected in the chain for the selected programmer. This is equivalent to Construct Chain Automatically in FlashPRO.

### **Configuring a Generic Device**

For Microsemi devices having the same JTAG IDCODE (i.e., multiple derivatives of the same Die), the device type must be configured for SmartDebug to enable relevant features for debug. The device can be configured by loading the programming file, by manually selecting the device using Configure Device, or by importing DDC files through Programming Connectivity and Interface. When the device is configured, all debug options are shown.

For debug projects created using Construct Automatically, you can use the following options to debug the devices:

- Load the programming file Right-click the device in Programming Connectivity and Interface.
- Import Debug Data from DDC file Right-click the device in Programming Connectivity and Interface.

The appropriate debug features of the targeted devices are enabled after the programming file or DDC file is imported.



## **Connected FlashPRO Programmers**

The drop-down lists all FlashPro programmers connected to the device. Select the programmer connected to the chain with the debug device. At least one programmer must be connected to create a standalone SmartDebug project.

Before a debugging session or after a design change, program the device through Programming Connectivity and Interface.

### See Also

Programming Connectivity and Interface View Device Status



# **SmartDebug User Interface**

## Standalone SmartDebug User Interface

You can start standalone SmartDebug from the Libero installation folder or from the FlashPRO installation folder.

### Windows:

<Libero Installation folder>/Designer/bin/sdebug.exe <FlashPRO Installation folder>/bin/sdebug.exe

### Linux:

<Libero Installation folder>/ bin/sdebug <FlashPRO Installation folder>/bin/sdebug

SmartDebug	
Project View Tools	
🕒 🚰 н 🕭	
SmartDebug Projects	
New	
Open	
Recent Projects	
C:\SmartDebug\test\test5 C:\SmartDebug\test\test3 C:\SmartDebug\test\test2 C:\SmartDebug\test\test1	
Log	đ ×
🔳 Messages 😵 Errors 🗼 Warnings 🁔 Info	

Figure 2 · Standalone SmartDebug Main Window

### **Project Menu**

The Project menu allows you to do the following:

- Create new SmartDebug projects (Project > New Project)
- Open existing debug projects (Project > Open Project)



- Execute SmartDebug-specific Tcl scripts (Project > Execute Script)
- Export SmartDebug-specific commands to a script file (Project > Export Script File)
- See a list of recent SmartDebug projects (Project > Recent Projects).

### Log Window

SmartDebug displays the Log window by default when it is invoked. To suppress the Log window display, click the View menu and toggle **View Log**.

The Log window has four tabs:

**Messages** – displays standard output messages

Errors – displays error messages

Warnings - displays warning messages

Info - displays general information

### **Tools Menu**

The Tools menu includes Programming Connectivity and Interface and Programmer Settings options, which are enabled after creating or opening a SmartDebug project.

## **Programming Connectivity and Interface**

To open the Programming Connectivity and Interface dialog box, from the standalone SmartDebug Tools menu, choose **Programming Connectivity and Interface**. The Programming Connectivity and Interface dialog box displays the physical chain from TDI to TDO.

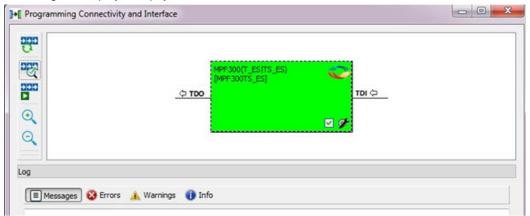
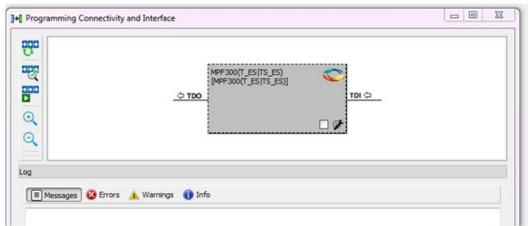


Figure 3 · Programming Connectivity and Interface Dialog Box – Project created using Import from DDC File

All devices in the chain are disabled by default when a standalone SmartDebug project is created using the **Construct Automatically** option in the Create SmartDebug Project dialog box.







The Programming Connectivity and Interface dialog box includes the following actions:

• Construct Chain Automatically - Automatically construct the physical chain.

Running Construct Chain Automatically in the Programming Connectivity and Interface removes all existing debug/programming data included using DDC/programming files. The project is the same as a new project created using the Construct Chain Automatically option.

- Scan and Check Chain Scan the physical chain connected to the programmer and check if it matches the chain constructed in the scan chain block diagram.
- Run Programming Action Option to program the device with the selected programming procedure.

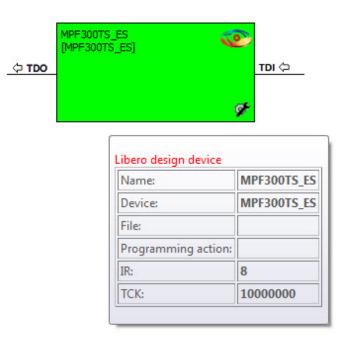
When two devices are connected in the chain, the programming actions are independent of the device.

- **Zoom In** Zoom into the scan chain block diagram.
- **Zoom Out** Zoom out of the scan chain block diagram.

### **Hover Information**

The device tooltip displays the following information if you hover your cursor over a device in the scan chain block diagram:

- Name: User-specified device name. This field indicates the unique name specified by the user in the Device Name field in Configure Device (right-click **Properties**).
- Device: Microsemi device name.
- Programming File: Programming file name.
- **Programming action:** The programming action selected for the device in the chain when a programming file is loaded.
- **IR:** Device instruction length.
- **TCK:** Maximum clock frequency in MHz to program a specific device; standalone SmartDebug uses this information to ensure that the programmer operates at a frequency lower than the slowest device in the chain.





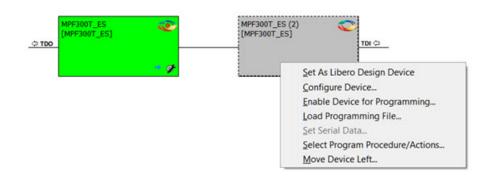
## **Device Chain Details**

The device within the chain has the following details:

- User-specified device name
- Device name
- Programming file name
- Programming action Select **Enable Device for Programming** to enable the device for programming. Enabled devices are green, and disabled devices are grayed out.

### **Right-click Properties**

The following options are available when you right-click a device in the Programming Connectivity and Interface dialog box.



Set as Libero Design Device - The user needs to set Libero design device when there are multiple identical Libero design devices in the chain.

Configure Device - Ability to reconfigure the device.

- Family and Die: The device can be explicitly configured from the Family, Die drop-down.
- Device Name: Editable field for providing user-specified name for the device.

**Enable Device for Programming** - Select to enable the device for programming. Enabled devices are shown in green, and disabled devices are grayed out.

Load Programming File - Load the programming file for the selected device.

Select Programming Procedure/Actions- Option to select programming action/procedures for the devices connected in the chain.

- Actions: List of programming actions for your device.
- **Procedures**: Advanced option; enables you to customize the list of recommended and optional procedures for the selected action.

Import Debug Data from DDC File - Option to import debug data information from the DDC file.

Note: This option is supported when SmartDebug is invoked in standalone mode.

The DDC file selected for import into device must be created for a compatible device. When the DDC file is imported successfully, all current device debug data is removed and replaced with debug data from the imported DDC file.

The JTAG Chain configuration from the imported DDC file is ignored in this option.

If a programming file is already loaded into the device prior to importing debug data from the DDC file, the programming file content is replaced with the content of the DDC file (if programming file information is included in the DDC file).



## **Debug Context Save**

Debug context refers to the user selections in debug options such as Debug FPGA Array, Debug Transceiver, and View Flash Memory Content. In standalone SmartDebug, the debug context of the current session is saved or reset depending on the user actions in Programming Connectivity and Interface.

The debug context of the current session is retained for the following actions in Programming Connectivity and Interface:

- Enable Device for Programming
- Select Programming Procedure/Actions
- Scan and Check Chain
- Run Programming Action

The debug context of the current session is reset for the following actions in Programming Connectivity and Interface:

- Auto Construct Clears all the existing debug data. You need to reimport the debug data from DDC file.
- Import Debug Data from DDC file
- Configure Device Renaming the device in the chain
- Configure Device Family/Die change
- Load Programming File

### **Selecting Devices for Debug**

Standalone SmartDebug provides an option to select the devices connected in the JTAG chain for debug. The device debug context is not saved when another debug device is selected.

Project View Tools Help	k_chain\g5_ddc_chain.dprj	and a state
🗋 🚰 н 🕭		
Device: MPF300(T_ES[TS_ES) (MPF300TS_ES) MPF300(T_ES[TS_ES) (MPF300TS_ES) M2S_M23QL010(T[S[TS]) (M2S_M23QL010(T[S[TS])) ID code read from device: IF8131CF	Programmer: S201YPV0TT (S201YPV0TT)	•
U code read from device: IP-0131CP View Device Status	Debug FPGA Array	]
Debug UPROM	Debug TRANSCEIVER	1
Log		8
log I Messages S Errors 🗼 Warnings 🚯 Info		ð



# **View Device Status**

Click **View Device Status** in the standalone SmartDebug main window to display the Device Status Report. The Device Status Report is a complete summary of IDCode, device certificate, design information, programming information, digest, and device security information. Use this dialog box to save or print your information for future reference.

evice: MPF300	TS_ES (MPF300	TS_ES) Programmer: S201YPV0	TT (S201YPV0TT)	Save	🗃 Print
Device Status:					
	IDCode (read	from the device) (HEX):	1F8131CF		
	Device Certifi	cate			
		Certificate is valid .			
	Design Inform	ation			
	besign intom	Design Name:	top		
		Design checksum (HEX):	6EBB		
		Design Version:	0		
	Digest Inform	ation			
	- georgeorgeorgeorgeorgeorgeorgeorgeorgeor	Fabric Digest (HEX):	cbb14c3b49b91c192b8ac	cf4075569e99	
			a036a15e10c2a332d4f5b	babdee39d169	
		SNVM Digest (HEX):	55b852781b9995a44c93	9b64e441ae27	
			24b96f99c8f4fb9a141cfd		
	Device Securi	hy Settings			
	Device Securi	Debug instructions are disable	ed.		
		Live probes are disabled.	2.00		
		User JTAG interface is disable			
		JTAG boundary scan is disable	ed.		
		IAP is disabled.			
		UFS UL segment is protected. Global key mode for factory in			
		Global key mode for zeroizatio			
		Global key mode for default k			
		Global key mode for authoriza			
		Global key mode for factory k			
		Global key mode for factory E			
		Global key mode for factory E		disabled.	
		Global key mode for user EC k Global key mode for user EC k		ablad	
		User Key1 write is protected.		abieu.	
		User Key2 write is protected.			
	Programming	Toformation			
	Programming	Cycle count:	16		
		*Algorithm Version:	1		
		* Programmer:	FlashPro 5		
		* Software Version:	FlashPro version not ava	ilable	
		* Programming Software:	FlashPro		
		* Programming Interface Prot * Programming File Type:	STAPL		
IOTE: * - The a	above Informat	tion is only relevant if the device	was programmed through JT/	AG or SPI Slave mode	

Figure 5 · Device Status Report



## IdCode

IDCode read from the device under debug.

### **Device Certificate**

Device certificate displays Family and Die information if device certificate is installed on the device. If the device certificate is not installed on the device, a message indicating that the device certificate may not have been installed is shown.

## **Design Information**

Design Information displays the following:

- Design Name
- Design Checksum
- Design Version

### **Digest Information**

Digest Information displays Fabric Digest, sNVM Digest (if applicable) computed from the device during programming. sNVM Digest is shown when sNVM is used in the design.

### **Device Security Settings**

Device Security Settings displays information about your security settings, including live probes, JTAG boundary scan, global key modes, and user keys.

### **Programming Information**

Programming Information displays the following:

- Cycle Count
- Algorithm Version
- Programmer
- Software Version
- Programming Software
- Programming Interface Protocol
- Programming File Type

### **Demo Mode**

The following figure shows an example of the Device Status Report in Demo Mode.



			N/S	
* SmartDel	oug is running in	Demo Mode. The device status	s seen below is an indicative example.	*
Device Status:				
	IDCode (read f	rom the device) (HEX):	Hardware not connected	
	Device Certifica	ate		
		Certificate is valid .		
	Design Informa	tion		
		Design Name:	sd	
		Design checksum (HEX):	F485	
		Design Version:	0	
	Digest Informa	tion		
		Fabric Digest (HEX):	6b0ee2f257cefa4c49a80942a12	37a9b
		85ff2e979d	7eb27d34c165de4c479631	
		SNVM Digest (HEX):	55b852781b9995a44c939b64e44	41ae27
		24b96f99c8	f4fb9a141cfc9842c4b0e3	
	Device Security	/ Settings		
	Programming Ir			
		Cycle count:	63	
		*Algorithm Version:	1	
		* Programmer:	FlashPro 5	
		* Software Version:	FlashPro version not available	
		* Programming Software:		
		* Programming Interface Pro		
		* Programming File Type:	STAPL	
OTE: * - The a	above Informatio	on is only relevant if the device	was programmed through JTAG or SP	I Slave mode.



# Debugging

## **Debug FPGA Array**

In the Debug FPGA Array dialog box, you can view your Live Probes, Active Probes, Memory Blocks, and Insert Probes (Probe Insertion).

The Debug FPGA Array dialog box includes the following four tabs:

- Live Probes
- <u>Active Probes</u>
- Memory Blocks
- Probe Insertion

## **Hierarchical View**

The Hierarchical View lets you view the instance level hierarchy of the design programmed on the device and select the signals to add to the Live Probes, Active Probes, and Probe Insertion tabs in the Debug FPGA Array dialog box. Logical and physical Memory Blocks can also be selected.

- Instance Displays the probe points available at the instance level.
- **Primitives** Displays the lowest level of probeable points in the hierarchy for the corresponding component —i.e., leaf cells (hard macros on the device).

You can expand the hierarchy tree to see lower level logic.

Signals with the same name are grouped automatically into a bus that is presented at instance level in the instance tree.

The probe points can be added by selecting any instance or the leaf level instance in the Hierarchical View. Adding an instance adds all the probe able points available in the instance to Live Probes, Active Probes, and Probe Insertion.



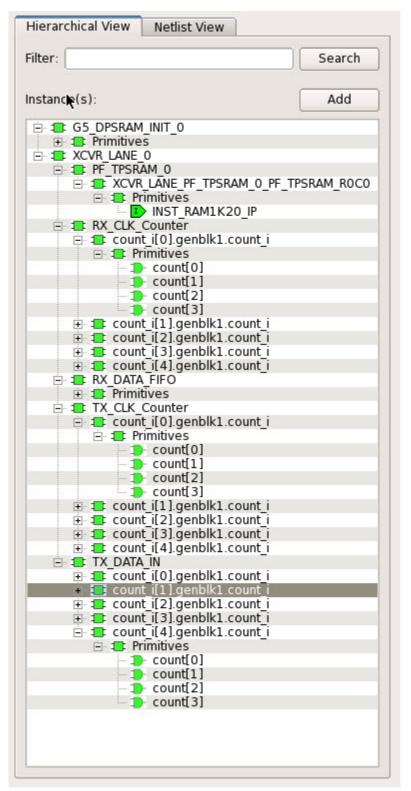


Figure 6 · Hierarchical View

Search

In Live Probes, Active Probes, Memory Blocks, and the Probe Insertion UI, a search option is available in the Hierarchical View. You can use wildcard characters such as \* or ? in the search column for wildcard matching.



Probe points of leaf level instances resulting from a search pattern can only be added to Live Probes, Active Probes, and the Probe Insertion UI. You cannot add instances of search results in the Hierarchical View.

## **Netlist View**

The Netlist View displays a flattened net view of all the probe-able points present in the design, along with the associated cell type.

Filter:	Sear	ch
Vet(s):	Add	ł
Name	Туре	•
count_0_q[0]:count_0/q[0]:Q	DFF	
count_0_q[10]:count_0/q[10]:Q	DFF	
count_0_q[11]:count_0/q[11]:Q	DFF	
count_0_q[12]:count_0/q[12]:Q	DFF	
count_0_q[13]:count_0/q[13]:Q	DFF	
count_0_q[14]:count_0/q[14]:Q	DFF	
count_0_q[15]:count_0/q[15]:Q	DFF	
count_0_q[16]:count_0/q[16]:Q	DFF	
count_0_q[17]:count_0/q[17]:Q	DFF	
count_0_q[18]:count_0/q[18]:Q	DFF	
count_0_q[19]:count_0/q[19]:Q	DFF	
count_0_q[1]:count_0/q[1]:Q	DFF	
count_0_q[2]:count_0/q[2]:Q	DFF	
count_0_q[3]:count_0/q[3]:Q	DFF	
count_0_q[4]:count_0/q[4]:Q	DFF	
count_0_q[5]:count_0/q[5]:Q	DFF	
count_0_q[6]:count_0/q[6]:Q	DFF	
count_0_q[7]:count_0/q[7]:Q	DFF	
count_0_q[8]:count_0/q[8]:Q	DFF	
count_0_q[9]:count_0/q[9]:Q	DFF	4

Figure 7 · Netlist View



### Search

A search option is available in the Netlist View for Live Probes, Active Probes, and Probe Insertion. You can use wildcard characters such as \* or ? in the search column for wildcard matching.

# Live Probes

Live Probes is a design debug option that uses non-intrusive real time scoping of up to two probe points with no design changes.

The Live Probes tab in the Debug FPGA Array dialog box displays a table with the probe names and pin types.

There are two channels, and Live Probe can be assigned/unassigned independently.

ive Probes Selection Ø X	PPGA Array debug data			
marchical Vew Netfat Vew	Live Probes Active Probes Memory Blocks Probe Intertion			
ser: Search		Delete	Delete All	
	Rate	Type		
terori(): Add	OH, PCR 35.0 (CREWINSHIM /F 0,U, OH, PCR 35.0 CREWINSHIM /F 0, WILSHINGHINGTE, 42-CH1, PCR 35.0 (CREWINSHIM /F 0,U, CH1, PCR 35.0 CREWINSHIM /F 0,U, CH1, PCR 35	Dete		
* B COREAHELSRAM PF 0	Ott, FCR, 55, S/CREARESIAN, JF, SUL, Ott, FCR, 55, CREARESIAN, JF, SJ, JRES/and/Jahour, yatek(S) Ott, FCR, 55, S/CREARESIAN, JF, SUL, Ott, FCR, 55, CREARESIAN, JF, SUL, Ott, FCR, 55, CREARESIAN, JF, SUL, Ott, FCR, 55, CREARESIAN, JF, SUL, Ott, FCR, 55, S/CREARESIAN, JF, S/L, S/L, S/L, S/L, S/L, S/L, S/L, S/L	DFF		
Student Control State	ON, ICE 35, SCREERESHM (F. S.), ON, ICE 35, CREERESHM (F. S. ARES-WITHERS y et al. 2-ON, ICE 35, SCREERESHM (F. S.), ON, ICE 35, CREERESHM (F. S.), ARES-WITHERS y et al. 3-ON	DFF		
S CONEXHETCAPE3_0 S CONEXHETCAPE3_1	OH, PCR, 55, 0/CREMBURM (P. O.). U, OH, PCR, 55, CREMBURM (P. O.). HIS smithurm, cover, ng(0)-OH, PCR, 55, 0/CREMBURM (P. O.). U, PCR, 55, 0/CREMBURM (P. O.). HIS smithurm, cover, ng(0)-OH	DPP		
B COREAVETOAPES_2 B CORECORTEMIL_1	OH, JCR, SS, SCORENBURN, M. SU, OH, JCR, SS, CORENBURN, M. P., JANSSWICKLAVER, GOH, JCR, SS, SCORENBURN, M. SU, OH, JCR, SS, CORENBURN, M. S. NESSWICKAVER, Q	Dee		
S CoreAHBUite_0     S CoreAD3HBnterconnect_0	OHLIPCIE, SS, SACREMELISIAM, M. SUL, OHLIPCIE, SS, COREMELISIAM, M. S., AMERICANI, M. S., SACREMELISIAM, M. SUL, OHLIPCIE, SS, COREMELISIAM, M. SUL, MERICANI, SS, AND	DFF		
> SCoreGPID_0 > SCoreGAITabb_0	OH, PCR, SS, SJCKREWELSKAW, M, SUL, OH, PCR, SS, COREWELSKAW, M, S, WELSHINGSON, PCR, SS, SJCKREWELSKAW, M, SUL, OH, PCR, SS, COREWELSKAW, M, S, WELSHINGSON, SS, SJCKREWELSKAW, M, S	DFF		
B PF_PCH_0 F S Ref_Design_0	OH, FCE, SS, ACOREMESSIAN (F, AU, OH, FCE, SS, COREMESSIAN (F, A, MESSIAN)(ISJ, OH, FCE, SS, ACOREMESSIAN (F, AU, OH, FCE, SS, COREMESSIAN (F, A, MESSIAN)(ISJ, OH, FCE, SS, ACOREMESSIAN (F, A, MESSIAN)(ISJ, OH, FCE, SS, ACOREMESSIAN (F, A, MESSIAN)(ISJ, OH, FCE, SS, ACOREMESSIAN (F, A), ARESIAN)(ISJ, OH, FCE, SS, ACOREMESSIAN)(ISJ, OH, FCE, SS, ACOREMESSIAN)(ISJ, OH, FCE, SS, ACOREMESSIAN)(ISJ, OH, FCE, SS, ACOREMESSIAN)(ISJ, ARESIAN)(ISJ, ARESIAN)(I		Data	
	OH, JCB, SS, DOREARSINH, JF, QU, DH, JCB, JS, DOREARSINH, JF, Q, ARSING(ISAND) OH, JCB, SS, DOREARSINH, JF, QU, DH, JCB, SS, DOREARSINH, J	DPP		
	OH, JYCE, SS, ACOREARESRAM, JP, ANNOUND, MOS[13], OH, JYCE, SS, ACOREARESRAM, JP, AU, OH, JYCE, SS, COREARESRAM, JP, ANNOUND, A[13], OH	Det.		
	OHL FCE, SS, ACOREMESRANUP, AMMONTH, SAMPLET, OHL FCE, SS, ACOREMESRANUP, AUL OHL FCE, SS, COREMESRANUP, A MEXAMINING SLIDIO	Dee		
	OH, JCIS, SS, ACOREARESRAY, JP, (Meaniner, and JLI), OH, JCIS, SS, COREARESRAY, JP, (M. JCIS, SS, COREARESRAY, JP, (M. ARSSAND/MCCR, JLI)) (2)	DFF		
	OH, JCD, SG, ACOREARESAN, JP, ANDRESS, ASS[15] OH, JCD, SG, ACOREARESAN, JP, AU, OH, JCD, SG, COREARESAN, JP, A, ARISSAND/MACOR, 4(15) O	Dete		
	ON 200 STOLEN AND A STOLEN ASTOLEN AND A STOLEN ASTOLEN AND A STOLEN ASTOLEN ASTOLEN ASTOLEN ASTOLEN ASTOLEN ASTOLEN AST			
	ONLINGS SCIONEARDSAM IF OWNERS AND INCOME INCOMESS ADDREAMS IF OU ONLINGS SCIONEARDSAM IF OU ONLINGS SCIONEARDSAM IF OU AND SMOTHACOL (DEC)	Det		
	OH, JCG, SS, OLOREWELSHAM, JP, ONNOVER, SAD (7), OH, JCG, SS, OLOREWELSHAM, JP, OU, OH, JCG, SS, OLOREWELSHAM, JP, O, MELSHANDAR, SJ7, Q	Dete		
	OTL/CE, 55,0008/HESRAF, P. 0HHVHH, MASH/OTL/CE, 55,0008/HESRAF, P. 0U. OTL/CE, 55,008/HESRAF, P. 0, HESRAF/HOR, (F. 0)	Dee		
	OHL FCR. 55.0 COREARESRAY IF OWNERS AND (SOLD FCR. 55.0 COREARESRAY IF OUL OHL FCR. 55. COREARESRAY IF O ARESING/HICOR. 5(5):0	Det		
	OH, JCIE 35 OKOREWESRAM JF OWNERS AND JCOL JCIE 35 OKOREWESRAM JF OU OH JCIE 35 OKRAMESRAM JF O WESSANDHORS, 460	Dete		
	OTLICE SE OCCERNISHER IF ONNOVER MEDIOTLICE SE OCCERNISHER IF OU OTLICE SE OCCERNISHER IF O ARSINGHOUSE (EE O	DFF		
	ONLING: \$5.00084485844 (P. 0)46444 (M. 0)2010 (C. 55.00084485844 (P. 0)) ONLING: \$5.0084485844 (P. 0) 4454401400, \$200	DIA		
	OHL PCIE 35 OKOREWESHAM IF OWNERS AND (DOHL PCIE 35 OKOREWESHAM IF OU OHL PCIE 35 OKOREWESHAM IF O WESHANDHOR (DO)	Dete		
	ONLINGE SS DICREMERSIAN IF DIREMEN AND DO IN INCE SS DICREMERSIAN IF DU UNLINGE SS COREMERSIAN IF D ARRENOTHED ADD	Dee		
	ONLIFE'S SUCCEARED SHAFF POINTERS SHILL ON LIFE'S SUCCEARED SHAFF POINTERS SUCCEARED SHAFF POINT	DAA		
			_	
	Assgn to Channel A -> OHL/Cle_SL3_COREARED.SAM /F _3.11_OHL/Cle_SL_COREARED.SAM /F _3.1485=milliound()[OHL/Cle_SL3_COREARED.SAM /F _3.11_OHL/Cle_SL3_COREARED.SAM /F _3.11_OHL/C		Chassi	
	Auge to Owned -> OHLPCR_SS_000RAHEURAH_PF_000bare_M69[10:OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_000RAHEURAH_PF_0U_OHLPCR_SS_00RAHEURAH_PF_0UAHAEURAH_PF_0UAHAEURAH_PF_0UAHAEURAH_PF_0UAHAEURAH_PF_0UAHAEURAH_PF_0UAHAEURAH_PF_0UAHAEURAH_PF_0UAHAEURAH_PF_0UAHAEURAHEURAH_PF_0UAHAEURAH_PF_0UAHAEURAH_PF_0UAHAEURAH_PF		Unassi	

Figure 8 · Live Probes Tab in SmartDebug FPGA Array Dialog Box

Two probe channels (ChannelA and ChannelB) are available. When a probe name is selected, it can be assigned to either ChannelA or ChannelB.

You can assign a probe to a channel by doing either of the following:

- Right-click a probe in the table and choose Assign to Channel A or Assign to Channel B.
- Click the **Assign to Channel A** or **Assign to Channel B** button to assign the probe selected in the table to the channel. The buttons are located below the table.

When the assignment is complete, the probe name appears to the right of the button for that channel, and SmartDebug configures the ChannelA and ChannelB I/Os to monitor the desired probe points. Because there are only two channels, a maximum of two internal signals can be probed simultaneously.

Click the **Unassign Channels** button to clear the live probe names to the right of the channel buttons and discontinue the live probe function during debug.

Note: Both probes can be assigned/unassigned independently.

### Live Probes in Demo Mode

You can assign and unassign Live Probes ChannelA and ChannelB. See the following example figure.



ctive Probes Selection	e ×	FPGA Array debug data
Hierarchical View   Netlist View	Search	Live Probes Active Probes Memory Blocks Probe Insertion Delete Delete All
Instance(s):	Add	Name
+         COREA+BTOAPB3_1           +         COREA+BTOAPB3_2           +         CORECORTEM1_1           +         CORECORTEM1_1           +         CORECORTEM1_1           +         CORECORTEM1_1           +         CORECORTEM1_1           +         CORECORTEM1_1           +         COREATED_0_0           +         COREATED_0_0           +         COREATED_0_0           +         FOR_0_0_0           +         Strate_0_0_0           +         Strate_0_0_0_0           +         Strate_0_0_0_0           +         Strate_0_0_0_0_0           +         Strate_0_0_0_0_0_0_0_0           +         Strate_0_0_0_0_0_0_0_0_0_0_0_0_0_0_0_0_0_0_0	1	CM1_PCIe_SS_0/ACVR_2_3_0/pattern_gen_dk_1_hstj_out[9]:CM1_PCIe_SS_0/ACVR_2_3_0/patter CM1_PCIe_SS_0/ACVR_2_3_0/pattern_gen_dk_1_hstj_out[8]:CM1_PCIe_SS_0/ACVR_2_3_0/patter CM1_PCIe_SS_0/ACVR_2_3_0/pattern_gen_dk_1_hstj_out[7]:CM1_PCIe_SS_0/ACVR_2_3_0/patter 4 Assign to Channel A -> CM1_PCIe_SS_0/ACVR_2_3_0/pattern_gen_dk_1_hstj_ou Assign to Channel B -> CM1_PCIe_SS_0/ACVR_2_3_0/pattern_gen_dk_1_rx_skp Unassign

## **Active Probes**

Active Probes is a design debug option to read and write to one or many probe points in the design through JTAG.

In the left pane of the Active Probes tab, all available Probe Points are listed in instance level hierarchy in the Hierarchical View. All Probe Names are listed with the Name and Type (which is the physical location of the flip-flop) in the Netlist View.

Select probe points from the Hierarchical View or Netlist View, right-click and choose **Add** to add them to the Active Probes UI. You can also add the selected probe points by clicking the **Add** button. The probes list can be filtered with the Filter box.

e/Active Probes Selection 🔅 🛞	FPGA Array debug data				
Herarchical View Netlist View	Live Probes Active Probes Memory Blocks				
Filter: Search	• = • •	Save	Load.	Delete	Delete Al
	Name	*	Type	Read Value	Strike Value
Instance(s) Add	* XCVR_LANE_Q(TX_DATA_IN_counter_out[19:0]		DFF	20hA8545	20h
B CS_DPSRAM_INT_0	* LANEO_TX_CLK_R_c[19:0]		DFF	20%CE0A6	20h
Initives     INT DONE	* LANED JX, DATA //FO_c[19:0]		DFF	20100000	20h
b blk b last addr	* R. DATA_c[19:0]		LSRAM	20100000	20h
x > ram state	GS_DPSRAM_INIT_O/6_bik_Z_GS_DPSRAM_INIT_O/6_bik-Q		DFF	0	-
<ul> <li>B read addr</li> <li>B read en</li> </ul>	GS DPSRAM INT Onlast addr 2 GS DPSRAM INT Onlast addr Q		DFF	1	-
x > write addr	INT_DONE_CISS_DPSRAM_INT_O/INT_DONE Q		DFF	1	-
<ul> <li>a write data</li> <li>a write en</li> </ul>	* G5 DPSRAM INT O/ram state[1 0]		DFF	213	28
H B XCVR LANE 0	a for the median formal and a		Der t	18.00	14.0
*         # 2000 LOB         TYSRAM (0 FF TYSRAM (0 ACC))           *         RY (C) Coulter         *           *         RY (C) Coulter         *					

Figure 9 · Active Probes Tab in SmartDebug FPGA Array Dialog Box

When you have selected the desired probe, points appear in the Active Probe Data chart and you can read and write multiple probes (as shown in the figure below).

You can use the following options in the Write Value column to modify the probe signal added to the UI:

- Drop-down menu with values '0' and '1' for individual probe signals
- Editable field to enter data in hex or binary for a probe group or a bus



+ = + +	Save	Load	Delete	Delete Al
lame	• h	ype	Read Value	Write Value
G5_DP5RAM_INIT_0/last_addr_Z_G5_DPSRAM_INIT_0/last_addr.Q				
G5_DPSRAM_INIT_0/b_blk_Z:G5_DPSRAM_INIT_0/b_blk:Q		DFF	0	
INIT_DONE_C:G5_DPSRAM_INIT_0/INIT_DONE:Q		DFF	1	0
XCVR_LANE_0/TX_DATA_IN_counter_out[19:0]		DFF	20hA8545	1
LANE0_TX_CLK_R_c[19:0]		DFF	20'hCE0A6	20hFFFFF
LANEO_RX_DATA_FIFO_c[19:0]		DFF	20%00000	20hFFFFF
R_DATA_c[19:0]	-	LSRAM	20'h00000	20hFFFFF
G5 DPSRAM INIT 0/ram_state[1:0]		DFF	2°h3	2'h

Figure 10 · Active Probes Tab - Write Value Column Options

### **Active Probes in Demo Mode**

In demo mode, a temporary probe data file with details of current and previous values of probes added in the active probes tab is created in the designer folder. The write values of probes are updated to this file, and the GUI is updated with values from this file when you click Write Active Probes. Data is read from this file when you click Read Active Probes. If there is no existing data for a probe in the file, the read value displays all 0s. The value is updated based on your changes.

See the following example figure.

Letive Probes Selection	e ×	FPGA Array debug data	Drohes   Mus	norv Blocks   Pr	who from the a		
Hierarchical View   Netlist View   Filter:	Search	+ - + +		Load		lete	Delete All
		Name	7	Type	Read Value	Write Val	ue 🔺
Instance(s):	Add	CM1_PCIe_SS_0	/counter[3]:Q	DFF	0		-
COREAHBTOAPB3_1     COREAHBTOAPB3_2     B CORECHBTOAPB3_2      B CORECHBTOAPB3_1      CORECHBUR_1      CoreAHBUR_0		CM1_PCIe_SS_0	k_0/rx_skip:Q	DFF	0		-
	CM1_PCIe_SS_0	/error_flag:Q	DFF	0		-	
	CM1_PCIe_SS_0	ror_flag_q1:Q	DFF	0		•	
CoreAXI4Interconnect_0     CoreGPIO_0	CoreAXI4Interconnect_0     CoreGRID_0		.ror_flag_q2:Q	DFF	1	1	-
CoreUARTapb 0		CM1_PCIe_SS_0	.ror_flag_q3:Q	DFF	1	1	-
PF PCIE_0     Ref_Design_0		CM1_PCIe_SS_0	ror_flag_q4:Q	DFF	0		•
<ul> <li>XCVR_2_3_0</li> <li>dvideby4_0</li> </ul>		CM1_PCIe_SS_0	/counter[0]:Q	DFF	0		-
B advideby4_1		CM1 PCIe SS 0	/counter[1]:0	DFF	lo		
<ul> <li># dvideby4_2</li> <li># dvideby4_3</li> </ul>	-	Read Act	ve Probes	Save Active Prob	es' Data V	Inte Active Prob	es

# Probe Grouping (Active Probes Only)

During the debug cycle of the design, designers often want to examine the different signals. In large designs, there can be many signals to manage. The Probe Grouping feature assists in comprehending multiple signals as a single entity. This feature is applicable to Active Probes only. Probe nets with the same name are automatically grouped in a bus when they are added to the Active Probes tab. Custom probe groups can also be created by manually selecting probe nets of a different name and adding them into the group.

The Active Probes tab provides the following options for probe points that are added from the Hierarchical View/Netlist View:

- Display bus name. An automatically generated bus name cannot be modified. Only custom bus names can be modified.
- Expand/collapse bus or probe group
- Move Up/Down the signal, bus, or probe group
- Save (Active Probes list)
- Load (already saved Active Probes list)



- Delete (applicable to a single probe point added to the Active Probes tab
- Delete All (deletes all probe points added to the Active Probes tab)
- In addition, the context (right-click) menu provides the following operations:
  - Create Group, Add/Move signals to Group, Remove signals from Group,
  - Ungroup
  - Reverse bit order, Change Radix for a bus or probe group
  - Read, Write, or Delete the signal or bus or probe group

ł	- + +	Save	Load	Delete	Delete All
Na	me	Туре	Read	Value Write	Value
	Q_0_c:DFN1_2/U0:Q		DFF 1		
	Q_1_c:DFN1_1/U0:Q		DFF 0	0	
	Q_2_c:DFN1_0/U0:Q		DFF 0	0	
4	q1_c[9:0]		DFF 10'	h395 10'h	
	q1_c[9]:count_0/q1[9]:Q		DFF 1		
	q1_c[8]:count_0/q1[8]:Q		DFF 1		
	q1_c[7]:count_0/q1[7]:Q		DFF 1		
	q1_c[6]:count_0/q1[6]:Q		DFF 0		
	q1_c[5]:count_0/q1[5]:Q		DFF 0		
	q1_c[4]:count_0/q1[4]:Q		DFF 1		
	q1_c[3]:count_0/q1[3]:Q		DFF 0		
	q1_c[2]:count_0/q1[2]:Q		DFF 1		
	q1_c[1]:count_0/q1[1]:Q		DFF 0		
	q1_c[0]:count_0/q1[0]:Q		DFF 1		
4	group1[1:0]		2'h:	1 2'h	
	q1_c[1]:count_0/q1[1]:Q		DFF 0		
	q1_c[0]:count_0/q1[0]:Q		DFF 1		
4	group2[2:0]		3'h	5 3'h	
	q1_c[4]:count_0/q1[4]:Q		DFF 1		
	q1_c[3]:count_0/q1[3]:Q		DFF 0		
	q1_c[2]:count_0/q1[2]:Q		DFF 1		

#### Figure 11 · Active Probes Tab

- Green entries in the "Write Value" column indicate that the operation was successful.
- Blue entries in the "Read Value" column indicate values that have changed since the last read.

### **Context Menu of Probe Points Added to the Active Probes UI**

When you right-click a signal or bus, you will see the following menu options:

For individual signals that are not part of a probe group or bus:

- Read
- Delete
- Poll
- Create Group

6



1_c[0]:count_0/q1[0]:Q		
	Read Delete	
	Poll Create Group	

For individual signals in a probe group:

- Read •
- Delete •
- Poll •
- Create Group ٠
- Add to Group
- Move to Group •
- Remove from Group

group1[1:0]			2'h1	2'h	
q1_c[1]:count_0/q1[1]:Q		DFF	0		
q1_c[0]:count_0/q1[0]:Q		DFF	1		
q1_c[9:0]	Read	DFF	10'h395	10 <sup>th</sup>	
q1_c[9]:count_0/q1[9]:Q	Delete	DFF	1		
q1_c[8]:count_0/q1[8]:Q		DFF	1		-
q1_c[7]:count_0/q1[7]:Q	Poll Create Group Add to Group	DFF	1		
q1_c[6]:count_0/q1[6]:Q		DFF	0		1
q1_c[5]:count_0/q1[5]:Q		DFF	0		1.1.1
q1_c[4]:count_0/q1[4]:Q		DFF	1		12
q1_c[3]:count_0/q1[3]:Q	Move to Group	DFF	0		-
q1_c[2]:count_0/q1[2]:Q	Remove from Group	DFF	1		
q1_c[1]:count_0/q1[1]:Q		DFF	0		- 0
q1_c[0]:count_0/q1[0]:Q		DFF	1		-

For individual signals in a bus:

- Read •
- Delete ٠
- Poll •
- Create Group
- Add to Group



q1_c[9]:count_0/q1[9]:O	
q1_c[8]:count_0/q1[8]:	Read
q1_c[7]:count_0/q1[7]:	Delete
q1_c[6]:count_0/q1[6]:	
q1_c[5]:count_0/q1[5]:	Poll
q1_c[4]:count_0/q1[4]:	Create Group
q1_c[3]:count_0/q1[3]:	
q1_c[2]:count_0/q1[2]:	Add to Group
q1_c[1]:count_0/q1[1]:Q	
q1_c[0]:count_0/q1[0]:Q	

For a bus:

- Delete
- Reverse Bit Order
- Change Radix to Binary
- Poll
- Create Group

c[9:0]	Delete
q1_c[9]:count_0/q1[9]:Q	Delete
q1_c[8]:count_0/q1[8]:Q	Reverse Bit Order
q1_c[7]:count_0/q1[7]:Q	Change Radix to Binary
q1_c[6]:count_0/q1[6]:Q	
q1_c[5]:count_0/q1[5]:Q	Poll
q1_c[4]:count_0/q1[4]:Q	Create Group
q1_c[3]:count_0/q1[3]:Q	
q1_c[2]:count_0/q1[2]:Q	
q1_c[1]:count_0/q1[1]:Q	
q1_c[0]:count_0/q1[0]:Q	

For a probe group:

- Delete
- Reverse Bit Order
- Change Radix to Binary
- Poll
- Create Group
- Ungroup



group2[4:0] q1_c[9]:count_0/q1[9	Delete
q1_c[9]:count_0/q1[9 q1_c[8]:count_0/q1[8 q1_c[7]:count_0/q1[7 q1_c[6]:count_0/q1[6 q1_c[9]:count_0/q1[9 q1_c[9]:count_0/q1[9 q1_c[8]:count_0/q1[8	Reverse Bit Order Change Radix to Binary Poll Create Group Ungroup
q1_c[7]:count_0/q1[7]:	9
q1_c[6]:count_0/q1[6]:	Q
q1_c[5]:count_0/q1[5]:	Q
q1_c[4]:count_0/q1[4]:	Q
q1_c[3]:count_0/q1[3]:	Q
q1_c[2]:count_0/q1[2]:	Q
q1_c[1]:count_0/q1[1]:	Q

### **Differences Between a Bus and a Probe Group**

A bus is created automatically by grouping selected probe nets with the same name into a bus. A bus *cannot* be ungrouped.

A Probe Group is a custom group created by adding a group of signals in the Active Probes tab into the group. The members of a Probe Group are not associated by their names. A Probe Group *can* be ungrouped.

In addition, certain operations are also restricted to the member of a bus, whereas they are allowed in a probe group.

The following operations are not allowed in a bus:

- Move to Group: Moving a signal to a probe group
- Remove from Group: Removing a signal from a probe group

## **Memory Blocks**

The Memory Blocks tab in the Debug FPGA Array dialog box shows the hierarchical view of all memory blocks in the design. The depth and width of blocks shown in the logical view are determined by the user in SmartDesign, RTL, or IP cores using memory blocks.

#### Notes:

- RAM is not accessible to the user when SmartDebug is accessing RAM blocks.
- RAM is not accessible to the user during a read or write operation.
  - During a single location write, the RAM block is not accessible. If multiple locations are written, the RAM block is accessed and released for each write.
  - When each write is completed, access returns to the user, so the access time is a single write operation time.

The example figure that follows shows the hierarchical view of the Memory Blocks tab. You can view logical

blocks and physical blocks. Logical blocks are shown with an L (<sup>11</sup>), and physical blocks are shown with a P (<sup>12</sup>).



emory Blocks Selection	ð×	FPGA Array debug data
ilter:	Search	Live Probes Active Probes Memory Blocks Probe Insertion
femory Blocks:	Select	User Design Memory Block: Data Width:
Instance Tree		Port Used:
4 🎩 Fabric_Logic_0		
4 🕿 U2		
4 1 F_0_F0_U1		
<ul> <li># ramtmp_ramtmp_0_0</li> <li># Primitives</li> </ul>		
<ul> <li>INST_RAMIK18_IP</li> </ul>		
F_10_F1_U2		
F_11_F1_U2		
F 12 F1 U2		
F_13_F1_U2		
F_14_F1_U2		
F_15_F1_U2		
F_16_F1_U2		
F_17_F1_U2		Read Block Save Block Data Write Block
F_18_F1_U2 F_19_F1_U2		(
E 1 E1 U2	-	

Figure 12 · Memory Blocks Tab - Hierarchical View

You can only select one block at a time. You can select and add blocks in the following ways:

• Right-click the name of a memory block and click Add as shown in the following figure.

lemory Blocks Selection	₽×	FPGA Array debug data
Filter:	Search	Live Probes Active Probes Memory Blocks Probe Insertion
Memory Blocks:	Select	User Design Memory Block:
		Data Width:
Instance Tree	*	Port Used:
Fabric_Logic_0		For Cosed.
4 = U2		
4 1 F_0_F0_U1	E	
	Add	
4 P Printaves		
	RAM1K18_IP	
F_10_F1_U2		
F_11_F1_U2		
F_12_F1_U2		
F_13_F1_U2		
F_14_F1_U2		
F_15_F1_U2		
F_16_F1_U2		
F_17_F1_U2		
F_18_F1_U2		Read Block Save Block Data Write Block
F_19_F1_U2		
F_1_F1_U2	-	

- Click on a name in the list and then click Select .
- Select a name, drag it to the right, and drop it into the Memory Blocks tab.
- Enter a memory block name in the Filter box and click **Search** or press **Enter**. Wildcard search is supported.

Note: Only memory blocks with an L or P icon can be selected in the hierarchical view.

### **Memory Block Fields**

The following memory block fields appear in the Memory Blocks tab.

### **User Design Memory Block**

The selected block name appears on the right side. If the block selected is logical, the name from top of the block is shown.



### Data Width

If a block is logical, the width from each physical block is retrieved from each physical block, consolidated, and displayed. If the block is physical, the width depends on the standard that is chosen to best meet the requirement. The list of widths that can be seen for LSRAM blocks is 1-bit, 2-bits, 5-bits, 10-bits, 20-bits, and 40-bits. For uRAM blocks, a fixed width of 12-bits is the only configuration supported for a physical block.

### **Port Used**

This field is displayed only in the logical block view. Because configurators can have asymmetric ports, memory location can have different widths. The port shown can either be Port A or Port B. For TPSRAM, where both ports are used for reading, Port A is used. This field is hidden for physical blocks, as the values shown will be irrespective of read ports.

The following figure shows the Memory Blocks tab fields for a logical block view.

emory Blocks Selection	8 ×	FPGA Array debu	ig data			
ilter:	Search	Live Probes	Active Probes	Memory Blocks	Probe Insertion	
Memory Blocks:	Select	User Design M Data Width:		Fabric_Logic_0/U3/F_	0_F0_U1	
Instance Tree		Port Used:		Port A -		
4 E Fabric_Logic_0		Port Used:		PORTA		
4 🗱 U3						
4 1 F_0_F0_U1						
ramtmp_ramtmp_0_0						
Primitives						
INST_RAM64x18_IP						
F_10_F1_U2						
# 12 rambmp_rambmp_0_0						
<ul> <li>Primitives</li> </ul>						
INST_RAM64x18_IP						
# \$ F_11_F1_U2 # \$ ramtmp_ramtmp_0_0						
Reprinting of the second						
INST_RAM64x18_IP			2			
4 1 F_12_F1_U2			Rea	d Block Save Blo	ck Data Write Block	
# Tamba ramba ramba 0_0		2				
4 Drimitione	-					

Figure 13 · Memory Blocks Tab Fields for Logical Block View

The following figure shows the Memory Blocks tab fields for a physical block view.

nory Blocks Selection	8 ×	FPGA Array debug data
Filter:	Search	Live Probes Active Probes Memory Blocks Probe Insertion
Memory Blocks:	Select	User Design Memory Block: Fabric_Logic_0/U3/F_0_F0_U1/ramtmp_ramtmp_0_0/INST_RAM64x18_IP Data Width: 9-bit
Instance Tree	*	
<ul> <li># Febric Logic_0</li> <li># U3</li> <li># F_0_F0_U1</li> <li># Frantrop_ramtrop_0_0</li> <li># Primitives</li> <li># F_10_F1_U2</li> <li># F_10_F1_U2</li> <li># Frantrop_ramtrop_0_0</li> <li># Primitives</li> <li># B Primitives</li> <li># F_11_F1_U2</li> <li># # Frintrop_samtrop_0_0</li> <li># # Frintrop_samtrop_0_0</li> <li># # Primitives</li> </ul>	_	
<ul> <li>INST_RAM64x18_IP</li> <li>F_12_F1_U2</li> <li>ramtmp_ramtmp_0_0</li> <li>Primitives</li> </ul>		Read Block Save Block Data Write Block

Figure 14 · Memory Blocks Tab Fields for Physical Block View



### **Read Block**

Memory blocks can be read once they are selected. If the block name appears on the right-hand side, the Read Block button is enabled. Click **Read Block** to read the memory block.

### **Logical Block Read**

A logical block shows three fields. User Design Memory Block and Data Width are read only fields, and the Port Used field has options. If the design uses both ports, Port A and Port B are shown under options. If only one port is used, only that port is shown.

lemory Blocks Selection		5 ×	FPGA Arra	y debu	g data														
Filter:	Search		Live Pr	obes	Active	Probes	Me	mory Blo	ds	Probe	Insertio	n							
Memory Blocks:	Select		User De Data W	esign Me lidth:	mory B	ode	Fabric_ 18-bit	Logic_0)	U3/F_0	_F0_U1	L.								
Instance Tree			Port Us	ad-			Port A		-										
# E Fabric_Logic_0			- William				- alla	_	_										
4 🎩 U3		12		0	1	2	3	4	5	6	7	8	9	A	B	C	D	F	F
<ul> <li>F_0_F0_U1</li> <li>Famting_ramting_</li> </ul>						-		-					-	-		-		-	
ramonp_ramonp_     A      Primitives		0000	00A83	08809	09008	14500	00010	00381	12028	00040	12080	04000	20214	02000	11080	20040	1C220	0A020	
INST_RA	M64x18_IP		0010	02200	04451	0.0001	-	05000	22600	00120	00000	00080	00400	0.4010	10000	00050	00106	00C22	10059
4 🎩 F_10_F1_U2			0010	02700	04451	04001	08000	03000	32300	00120		00080	00420	04013	10000	00032	00100	000.22	10030
# 12 ramtmp_ramtmp_	0_0		0020	10400	00010	10000	14044	10040	0810E	39425	00990	10C14	00004	04001	10000	00100	00042	20100	08002
<ul> <li>Primitives</li> <li>INST RA</li> </ul>	102 4 10 10		1																C.C.C.C.C.
4 1 F_11_F1_U2	P104X10_IP		0030	0001B	000000	20808	0008A	001E0	28100	02883	00770	10020	04000	00000	00200	20004	22400	04006	0A090
A B ramtmp_ramtmp_	0.0																		
<ul> <li>Primitives</li> </ul>																			-
INST_RA	M64x18_IP							Read B	~	Caus	Block D		1844	te Block					
# # F_12_F1_U2 # # ramtmo_ramtmo_								Ned0 D	w.	Jaave	DRUCK DA	ara	4461	te plock					
A Drimitives	0.0	*																	

Figure 15 · Logical Block Read

The data shown is in Hexadecimal format. In the example figure above, data width is 18. Because each hexadecimal character has 4 bits of information, you can see 5 characters corresponding to 18 bits. Each row has 16 locations (shown in the column headers) which are numbered in hexadecimal from 0 to F.

**Note:** For all logical blocks that cannot be inferred from physical blocks, the corresponding icon does not contain a letter.

### **Physical Block Read**

When a Physical block is selected, only the User Design Memory Block and Data Width fields are shown.

emory Blocks Selection	6 ×	FPGA Arr	ay deb	ug dat	ta														
Filter:	Search	Live Pr	obes	Act	tive Pr	obes	M	mory	Blocks	F	robe	Insert	ion						
Memory Blocks:	Select	User D Data V		lemor	y Blod		Fabric, 9-bit	Logic	_0/U3	F_0_	F0_U1	/ramb	np_ra	mtmp	_0_0/	NST_	RAM6	4x18_	P
Instance Tree	•		1.00																
Fabric_Logic_0			0	1	2	3	- 4	5	6	7	8	9	Α	В	С	D	E	F	-
4 103 4 100 F0_F0_U1		0000	083	005	009	044	008	048	100	0A2	010	000	181	001	028	090	040	000	
A 2 rambnp_rambnp_0_0		1000																	18
A B Primitives		0010	080	090	000	020	014	101	000	010	080	088	040	100	020	0E1	020	050	Ξ
INST_RAM64x18_IP																			
4 1 F_10_F1_U2		0020	100	013	051	022	001	020	000	040	000	028	100	192	120	000	000	000	
ramtmp_ramtmp_0_0																			1.00
<ul> <li>Primitives</li> </ul>		0030	080	000	020	002	019	020	000	0E4	052	000	106	000	022	006	058	080	
INST_RAM64x18_IP		0040	000	000	010	000	000	000	044	010	0.40	050	100	0.40	0.75	104	100	060	
F_11_F1_U2		0040	000	004	0.10			000	011	040	040	UEU .	10C	040	023	104	790	000	
A Stramtmp_ramtmp_0_0		0050	014	086	004	000	001	020	000	080	100	000	042	000	100	100	002	040	-
<ul> <li>INST_RAM64x18_IP</li> </ul>																			
4 1 F_12_F1_U2						Read	Block		Save	Block (	Data.		Write	e Bloc	k				
A S ramtmp ramtmp 0 0																			
4 30 Primitions	-																		_

Figure 16 · Physical Block Read



## Write Block

### **Logical Block Write**

A memory block write can be done on each location individually. A logical block has each location of width that is displayed. The written format is hexadecimal numbers from 0 to F. Width is shown in bits, and values are shown in hexadecimal format. If an entered value exceeds the maximum value, SmartDebug displays a popup message showing the range of allowed values.

lemory Blocks Selection	ē×	FPGA Arr	av debu	g data														
Filter:	Search	Live Pr	obes	Active	Probes	Mer	nory Blo	da	Probe In	nsertion	1							
Memory Blocks:	Select	User D Data V	esign Me /idth:	smory Bl		Fabric_L 18-bit	.ogic_0/	U3/F_12	2_F1_U2									
Instance Tree		Port U	sed:			PortA	1	*										
Fabric_Logic_0     B U2			0	1	2		4	5	6	7	8	9	A	В	с	D	E	F
F_0_F0_U1		0000					01200	00824	00004	00304	00200	00600	0006A	20001	00060			00000
F_11_F1_U2 F_12_F1_U2	F_10_F1_U2 F_11_F1_U2			20410	20002	02101	00080	08016	020C0	0C200	000A0	00002	08000	10020	05004	00018	20008	08300
<ul> <li># S ramtmp_ramtmp</li> <li># Primitives</li> </ul>		0020	00200	00000	00000	00084	00080	02408	00001	02080	20000	00000	20000	00005	02000	02012	00C01	00454
F_13_F1_U2 F_14_F1_U2	LAM64x18_JP	0030	02400	10001	00001	04000	00400	00002	01201	00004	00020	01000	02040	10008	07242	18102	24041	02044
<ul> <li>F_15_F1_U2</li> <li>F_16_F1_U2</li> <li>F_16_F1_U2</li> <li>F_17_F1_U2</li> </ul>	_						Read B	ock	Save	Block Da	ta)	Writ	e Block					
F_18_F1_U2	-					-												

Figure 17 · Logical Block Write

### **Physical Block Write**

Physical blocks have variable widths based on the standard configuration. The maximum value that can be written in hexadecimal format depends on the width shown. If an entered value exceeds the limit, SmartDebug displays a popup message showing the range of values that can be entered.

emory Blocks Selection	đ×																		
emory bloods selection	0 A	FPGA Arra	ay deb	ug dat	ta														
Filter:	Search	Live Pr	obes	Act	tive Pr	obes	Me	emory	Blocks	P	robe	Insert	ion						
Memory Blocks:	Select	User De Data W		lemor	y Blod		Fabric, 9-bit	Logic	_0/U2/	/F_0_f	0_U1	l/ramb	np_ra	mtmp	_0_0/3	NST_	RAM1	K18_D	P
Instance Tree	*								_						_				-
Fabric_Logic_0			0	1	2	3	4	5	6	7	8	9	A	В	С	D	E	F	-
4 1 U2 4 1 F 0 F0 U1	E	0000	123	1FF	032	084	117	102	007	008	044	031	100	135	037	088	068	051	
4 38 rambmp_rambmp_0_0				-															
4 2 Primitives		0010	014	023	180	037	114	028	15C	003	00A	017	011	008	060	1D2	041	002	
INST_RAM1K18_IP																			
F_10_F1_U2		0020	024	159	15C	053	110	168	004	13C	18D	042	015	068	148	061	1DC	112	
F_11_F1_U2		0030	-																
F_12_F1_U2		0030	02F	068	058	02E	100	ICE	025	00	169	012	053	123	011	088	128	11C	
F_13_F1_U2		0040	040	147	052	102	044	1FF	145	OBE	010	029	OSE	049	106	1AC	011	104	
F_14_F1_U2				-	***												***		
F_15_F1_U2 F_16_F1_U2		0050	01F	18A	01C	040	1F5	044	165	018	OEO	117	033	003	110	OBC	0A4	068	-
F 17 F1 U2					-	1-1-2-1		-	-			-			-				
F_18_F1_U2						Read	Block		Save	Block (	Data.		Write	e Block	k.				
F_19_F1_U2					_			_	1		-				_				
5 R F 1 F1 112	7	1.1																	

Figure 18 · Physical Block Write



### **Unsupported Memory Blocks**

If RTL is used to configure memory blocks, it is recommended that you follow RAM block inference guidelines provided by Microsemi.

SmartDebug may or may not be able to support logical view for memory blocks that are inferred using RTL coding not specified in the above document.

## **Memory Blocks in Demo Mode**

A temporary memory data file is created in the designer folder for each type of RAM selected. All memory data of all instances of USRAM, LSRAM, and other RAM types is written to their respective data files. The default value of all memory locations is shown as 0s, and is updated based on your changes.

Both physical block view and logical block view are supported. See the following example figure.

	PGA Arta	ry debug	data															
Filter: Search	Live Pro	obes	Active P	robes	Memo	ry Blocks	Pro	be Inser	tion									
Memory Blocks: Select		sign Mer	nory Bloc		-	_SS_0/P	ef_Desi	n_0/m	Baxi_ref.	design	v0_0/ax	i4slv_ra	n/scramb	e_inst_	128×51	2/xr3axi	scramb	e_(
	Data W			-	4-bit fort A		-T											
Instance Tree	_			-			_							_				
CORECORTEXM1_1     CoreAXI4Interconnect_0		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	-
⊟ ■ Ref_Design_0 ⊕ ■ xr3axi_ref_design	0000	000012	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
B ■ axivisiv_ram B ■ scrambe_in	0010	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
⊡ 2 xr3axi ⊕ 28 xr3	0020	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
€ 25 xr3 € 28 xr3	0030	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
. ≇ xr3	0040	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
	0050	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
	0060	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
			1.000	1000000		120000				Constant of	100000		Canada a					
	0070												000000					
	0080	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
	0090	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
	00A0	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
	0080	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	000000	
			******			******												•
						Rea	Block	Sav	e Block (	ata	Whi	te Block						
						_		- 65		-	-							-

# Probe Insertion (Post-Layout)

### Introduction

Probe insertion is a post-layout debug process that enables internal nets in the FPGA design to be routed to unused I/Os. Nets are selected and assigned to probes using the Probe Insertion window in SmartDebug. The rerouted design can then be programmed into the FPGA, where an external logic analyzer or oscilloscope can be used to view the activity of the probed signal.

Note: This feature is not available in standalone mode because of the need to run incremental routing.



- 9 57

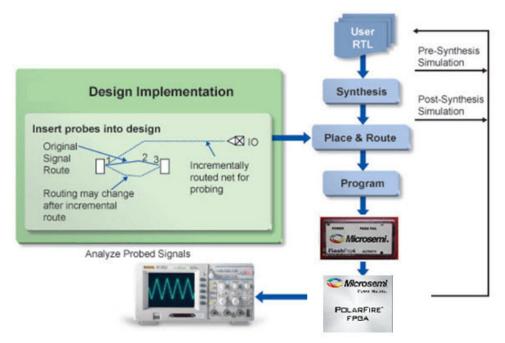


Figure 19 · Probe Insertion in the Design Process

The Probe Insertion debug feature is complementary to Live Probes and Active Probes. Live Probes and Active Probes use a special dedicated probe circuitry.

### **Probe Insertion**

-----

- 1. Double-click **SmartDebug Design** in the Design Flow window to open the SmartDebug main window.
- Note: FlashPro Programmer must be connected for SmartDebug.
- 2. Select **Debug FPGA Array** and then select the Probe Insertion tab.

be Insertion Data Selection	e ×	FPGA Array debu	ig data						
Probe Insertion Data Selection 8 × Hierarchical View Netilist View		Live Probes	Active Probes	Memory Blocks	Probe Insertion				
Filter:	Search							Delete	Delete All
Instance(s):	Add	Net		Driver		Package Pin	1	Port	Name
Instance Tree		AND2_0_Y	AND2_0/U	10:Y		Unassigned	-	Probe_Insert0	
>         S Primitives           >         S AND2_0           >         S Debuf           >         S FCCC_0           >         MUX_SEL           >         S Reset           >         S top           >         S top           >         UTTAG_0		D_c	UJTAG_0/I	INST_UJTAG_SYSRE	SET_FF_JP:UDRUPD	Unassigned	•	Probe_Insert1	
<ul> <li>Iser_CLK</li> <li>count_0</li> </ul>					Ins	ert probe(s) and	prog	ram the device	Run

#### Figure 20 · Probe Insertion Tab

In the left pane of the Probe Insertion tab, all available Probe Points are listed in instance level hierarchy in the Hierarchical View. All Probe Names are shown with the Name and Type in the Netlist View.

3. Select probe points from the Hierarchical View or Netlist View, right-click and choose **Add** to add them to the Active Probes UI. You can also add the selected probe points by clicking the **Add** button. The probes list can be filtered with the Filter box.



Each entry has a Net and Driver name which identifies that probe point.

The selected net(s) appear in the Probes table in the Probe Insertion tab, as shown in the figure below. SmartDebug automatically generates the Port Name for the probe. You can change the Port Name from the default if desired.

4. Assign a package pin to the probe using the drop-down list in the Package Pin column. You can assign the probe to any unused package pin (spare I/O).

be Insertion Data Selection	đ×	FPGA Array debug	data		
Hierarchical View Netlist Vie	2W	Live Probes	Active Probes Memory Blocks	Probe Insertion	
Filter:	Search				Delete Delete All
Instance(s):	Add	Net	Driver	Package Pin	Port Name
Instance Tree		q_c[0]	count_0/q[0]:Q	H5 -	Probe_Insert0
Primitives     AND2.0		q_c[1]	count_0/q[1]:Q	Н6 -	Test2
AND2_0 B D_ibuf B FCCC_0		q_c[3]	count_0/q[3]:Q	36 🕶	Probe_Insert2
MX2_0					
<ul> <li>▷ Stop</li> <li>▷ Stop</li> <li>▷ UJTAG_0</li> <li>▷ User_CLK</li> <li>▲ Scont_0</li> <li>▲ Primitives</li> <li>▷ D Q_RNO</li> <li>▷ D Q_RNO</li> <li>▷ D Q_s</li> </ul>					

Figure 21 · Debug FPGA Array > Probe Insertion > Add Probe

5. Click Run.

This triggers Place and Route in incremental mode, and the selected probe nets are routed to the selected package pin. After incremental Place and Route, Libero automatically reprograms the device with the added probes.

The log window shows the status of the Probe Insertion run.

### **Probe Deletion**

To delete a probe, select the probe and click **Delete**. To delete all probes, click **Delete All**.

**Note**: Deleting probes from the probes list without clicking **Run** does not automatically remove the probes from the design.

### **Reverting to the Original Design**

To revert to the original design after you have finished debugging:

- 1. In SmartDebug, click Delete All to delete all probes.
- 2. Click Run.
- 3. Wait until the action has completed by monitoring the activity indicator (spinning blue circle). Action is completed when the activity indicator disappears.
- 4. Close SmartDebug.



# Debug sNVM

The sNVM block stores User data and UIC data. This data is stored as clients and can be configured in the Libero design. The USK (User Secret Key) security key secures pages within the memory. Authenticated data can be plain text or encrypted text, and non-authenticated data is plain text. SmartDebug helps the user read the page content of the sNVM block.

The sNVM Debug window has two tabs - Client View and Page View.

### **Client View**

When you open the sNVM window, two tabs are visible. Client information appears in the Client View tab when it is configured in the Libero design. Select a client to expand the table and see pages and page status inside the client. Click the **Read From Device** button to view the memory content.

You can select only one client at a time. Pages inside the client cannot be selected.

Start Page, End Page, and Number of Bytes are displayed for the selected client.

Click the **View All Page Status** button to see information for all pages in the client. See the following example figures.

Client List	Start Page	End Page	Number of Bytes	Write cycles	Page Type	Used as ROM	USK status	
INIT_STAGE_2_SNVM_C	CLIENT 0	0				No		
Page 0	0		252	44	Plain Text	No	N/A	
INIT_STAGE_1_SNVM_C		220				No	1	
Page 220	220		252	33	Plain Text	No	N/A	
Page 219	219	10	252	43	Plain Text	No	N/A	

Figure 22 · Client View - expanded list

M Debug	V Page Vew																
Refresh CA	ient Details																
Client List		C Stat Page	End Page	Number of Bytes	Write cycles	Page Type	Used as ROM	USK status									
INT S	TAGE 2 SNIM	URNT 0	0	252	44	Plain Text	No	N/A									
	TAGE 1, SNIM, C	1.0NT 219	220		13			N/A									
10	nge 220	229		292 192	43	Plain Test Plain Test	1945 1946	N/A N/A									
ed from De	evice																
est Conto	ent Retrieved fro	on Device:															Wed Sep 06 10:21:5
leved Con	tent: Clent "INET,	STAGE L SWIT CLE	NT'.														
									Ven Al Pag	e Stetue							
- 2	0	1	2	3	4	5		6	7		. 5	A	1	с	D	t.	
060	-02	90	06	0	00	00		47	00	42	62	00	80	4C	0	00	68
070	42	00	06	CD	04	04		04	04	02	62	00	80	48	AD	00	08
080	62	00	06	0	00	00		47	00	62	62	00	80	4C	Al	00	68
090	00	00	06	0	00	00		00	00	03	00	00	10	00	00	00	00
040	00	00	00	00	00	00		00	00	00	00	00	00	00	00	00	00
080	00	00	00	00	00	00		80	00	00	00	00	00	00	00	00	00
000	00	00	00	00	00	00		00	00	00	00	00	00	00	00	00	00
000	00	00	00	00	00	00		00	00	00	00	00	00	00	00	00	00
CEO	00	-00	00	00	00	00		00	00	00	00	00	00	00	00	00	00
OFO	00	00	00	00	00	00		00	00	00	00	00	00	00	00	00	00
100	00	00	00	00	00	.00		00	00	00	00	00	Page 219, A	ddress - Di00FB, Vali		00	80
110		20		00	00	00		00	00	-00	00	00	00		20	60	00
120	00	00			00			8	00				00		-		00
130	00	00			00	00					- 00		00	-	10		00
140	00	00	00	00	00	00			00				00		00	00	00
	00				00										00		
150		00		00		-00		00	00		00	00	00			00	00
1160	00	00	- 00	00	00	00		00	00	60	00	00	00	00	00	00	00
170	00	00	00	00	00	00		00	00	00	00	00	00	00	00	00	00

Figure 23 · Client View - Memory



### **Page View**

Page View is used to read a range of pages where start and end page have been specified.

If a page is secured, the default USK is used by SmartDebug to get the page status. If successful, the USK automatically reads the page. If a different USK has been set using system services, use the option to enter the USK, as shown in the example figure below.

Debug						- 0
ent View V P						
erewew y -	all ren (					
Hart page:	0	Check Page S	tatus			
Ind Page:		(11Pages)				
and Pages	1.0	(II) ages /				
Page List	V Number of Bytes	Page Type	Write Cycles	Used as ROM	USK Status	
Page 0	252	Plain Text	44	No	N/A	
Page 1 Page 2 Page 3 Page 4	252	Plain Text	142	No	N/A	
Page 2		Plain Text	11	No	N/A	
Page 1	252	Plain Text	4	No	N/A	
Page 4	252		4	No	N/A	
		Plain Text	4	No	N/A	
Page 6 Page 7	252		3	No	N/A	1
Page 7		Plain Text	3	No	N/A.	
Page 8	252	Plain Text	3	No	N/A	
Page 9	252	Plain Test	3	No	N/A	
Page 10	212	Blank Text	0	140	Enter USCkey (24 hexadedmal characters )	
t Content R	letrieved from Device:					Wed Sep 06 10:21:50 2
eved Contents						Wed Sep 06 30:21:50 20
						Wed Sep 06 30:21:00 2

Figure 24 · Page View - Enter USK highlighted

M Debug	Page																	
Startpage	1	9	Check Page St	table .														
End Pages	ſ	2ů	(11 Pages)															
Page List	7	Number of Bytes	Page Type	Write Cycles	Used as ROM	USK Status												-
- (Page	0	252	Plain Test	44	No	IN/A												_
Page		252		14	No	N/A												
- Page		252	Plain Text	11	No	N/A												
2020	1	252	Plain Test	4	No	N/A												
Page		252		4	No	N/A												
- Page	2	28	Plain Test	4	No	N/A												_
2000	-	252	Plain Text Plain Text	-	No	N/A N/A												-
Tage .		1252	Plain Test Plain Test	1	No	N/A N/A												-
Page	-	104	Plan Test	1	No	N/A												-
Page	10	282	Blank Text	0	No	14												-
		rved from Device: Page 0 to Page 31															Wed Sep 06 10:31	214
	nt for	Page 0 to Page 3							Vex Al Pag									1.4
		Page 0 to Page 3		2	3	4	5	6	Ven Al Pag	e Statue] 8	9	A	1	c	D	E.	F	1.4
inved Canta	0 0	Page 0 to Page 11	1	00	00	00	00	00	7	8	00	00	00	00	00	£ 00	F 00	1.4
aved Canta (A10) (A20)	0 00 00	Page 0 to Page 11	1	00 00	00 00	00 00	00 00	00 00	7 00 00	8 00 00	00	00	00 00	00 00	00	E 00 00	1 00 00	1.4
ano (a) (a) (a) (a) (a) (a) (a) (a) (a) (a)	0 00 00 00	Page 0 to Page 11	1/ 1/ 0 0 0	00 00 00	00 00 00	00 40 60	00 00 00	00 00 00	7 00 00 00	8 50 50 50	00 00 00	00 00 00	00 00 00	00 00 00	00 00 00	E 00 00 00	7 08 00	1.4
A10 A20 A40	0 00 00 00 00	Page 8 to Page 31	1	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	7 00 00 00 00	8 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	00 00 00 00	E 00 00 00 00	F 00 00 00	1.4
inved Control (A.10) (A.20) (A.30) (A.30) (A.30) (A.30)	0 00 00 00 00 00	Page 0 to Page 11	1 0 0 0 0 0	00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	7 00 00 00 00 00	8 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00 00 00 00	00 00 00 00 00	E 00 00 00 00 00	F 00 00 00 00 00	1.4
5410 5420 5430 5450 5460	0 00 00 00 00 00 00	Page 0 to Page 11	1	00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00 00	00 00 00 00 00 00	7 00 00 00 00 00 00 00 00	8 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00 00	00 00 00 00 00 00	00 00 00 00 00 00	00 00 00 00 00 00	1 00 00 00 00 00 00	F 00 00 00 00 00 00	1.4
0410 0420 0430 0440 0450 0450 0470	0 00 00 00 00 00 00 00	Page 0 to Page 1		00 00 00 00 00 00 00	00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	7 00 00 00 00 00 00 00 00	8 00 00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	1 00 00 00 00 00 00 00 00 00	F 00 00 00 00 00 00 00 00	1.40
iaved Conte (A.10) (A.2	0 00 00 00 00 00 00 00 00 00	Page 0 to Page 1	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	7 00 00 00 00 00 00 00 00 00	8 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	8 00 00 00 00 00 00 00 00	1 00 00 00 00 00 00 00 00 00	1.40
aved Conte (4.10) (4.20) (4.30) (4.30) (4.30) (4.30) (4.30) (4.30) (4.30)	0 00 00 00 00 00 00 00	Page 0 to Page 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	00 00 00 00 00 00 00	00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	7 00 00 00 00 00 00 00 00	8 00 00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	00 00 00 00 00 00 00	1 00 00 00 00 00 00 00 00 00	F 00 00 00 00 00 00 00 00	1.4
A10 A20 A30 A30 A30 A30 A30 A30 A30 A30 A30 A3	ent from 0 00 00 00 00 00 00 00 00 00 00	Page 0 to Page 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00	7 00 00 00 00 00 00 00 00 00 00 00	8 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	8 00 00 00 00 00 00 00 00 00	7 00 00 00 00 00 00 00 00 00 00	1.40
410 410 420 430 440 430 440 430 440 440 440 440 44	ent for 0 00 00 00 00 00 00 00 00 00 00 00	Page 8 to Page 1 0 0 0 0 0 0 0 0 0 0 0 0 0	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	00 00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00	7 00 00 00 00 00 00 00 00 00 00	8 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00 00	1 00 00 00 00 00 00 00 00 00 00 00 00	F 00 00 00 00 00 00 00 00 00 00 00 00 00	1.40
	0 0 00 00 00 00 00 00 00 00 00 00 00 00	Page 6 to Page 1 0 0 0 0 0 0 0 0 0 0 0 0 0		00 00 00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00	7 00 00 00 00 00 00 00 00 00 00 00	8 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00 00 00	3 00 00 00 00 00 00 00 00 00 00 00 00 00	7 00 00 00 00 00 00 00 00 00 00 00 00 00	

Figure 25 · Memory View - Page Range



Secured NVM Details			? X
		Save	Print
and Sheh Hannes Cashard Connector			
ecured Flash Memory Content [ SNVM Pages sNVM Page #0:	0		
Page Status:			
Write Cycle Count:	44		
Page Type:	Plaintext		
Use as ROM: Data Read Status:	Off Success		
sNVM Page #1:	SUCCESS		
Page Status:			
Write Cycle Count:	14		
Page Type:	Plaintext		
Use as ROM:	Off		
Data Read Status:	Success		
sNVM Page #2: Page Status:			
Write Cycle Count:	11		
Page Type:	Plaintext		
Use as ROM:	Off		
Data Read Status:	Success		
sNVM Page #3:			
Page Status:	4		
Write Cycle Count: Page Type:	Plaintext		
Use as ROM:	Off		
Data Read Status:	Success		
sNVM Page #4:			
Page Status:			
Write Cycle Count:	4		
Page Type: Use as ROM:	Plaintext		
Data Read Status:	Success		
sNVM Page #5:			
Page Status:			
Write Cycle Count:	4		
Page Type:	Plaintext		
Use as ROM: Data Read Status:	Off Success		
sNVM Page #6:	JULLESS		
Page Status:			
Write Cycle Count:	3		
Page Type:	Plaintext		
Use as ROM:	Off		
Data Read Status: sNVM Page #7:	Success		
Page Status:			
Write Cycle Count:	3		
Page Type:	Plaintext		
Use as ROM:	Off		
Data Read Status:	Success		
sNVM Page #8: Page Status:			
Write Cycle Count:	3		
Page Type:	Plaintext		
Use as ROM:	Off		
Data Read Status:	Success		
sNVM Page #9:			
Page Status: Write Cyde Count:	3		
Page Type:	Plaintext		
Use as ROM:	Off		
Data Read Status:	Success		
sNVM Page #10:			
Page Status:			-
Help			Close

Figure 26  $\cdot$  View All Page Status



### **Read Operation**

#### **Client View**

The Client View displays all the clients that are configured in the design. When a client is expanded, a table listing all pages is displayed.

When a client is selected, the Read from Device button is enabled. Click **Read from Device** to read the content of the client. A client can have one or more pages. Refresh Client Details option is given to the user to refresh the table. Click **Refresh Client Details** to update the information in SmartDebug and refresh the table. This is helpful when a client configuration is changed using system services.

#### **Page View**

When valid parameters are entered and **Check Page Status** is clicked, a table of all pages is shown with page status information. Pages in the table are read-only and cannot be selected. The page range included in Start Page and End Page is validated, and the Read from Device button is enabled. Click **Read from Device** to read the content.

#### **Runtime Operations**

After a design is programmed into the device, you can do the following:

- Change the content of a page
- Authenticate a page
- Change the security key of each configured page

The above operations are not possible if the page is used as ROM.

You can refresh page status in SmartDebug:

- Click the **Refresh Client Details** button in the Client View tab to refresh the client view table and update it with the latest changes.
- Click the Check Page Status button in the Page View tab to refresh the pages in the table.

If the security key has been changed, SmartDebug prompts you to enter the USK manually.

Enter the USK in the USK Status column (Client View tab and Page View tab).

By default, the USK entered in the configurator as the USK client is used to authenticate the page.

## **Debug Transceiver**

### **Debug Transceiver**

The Debug Transceiver feature in SmartDebug checks the lane functionality and health for different settings of the lane parameters.

To access the Debug Transceiver feature in SmartDebug, click **Debug Transceiver** in the main SmartDebug window.



rogrammer: S201YQST1V (S201YQST1V)	*
Debug FPGA Array	
Debug TRANSCEIVER	
	8
	Debug FPGA Array Debug TRANSCEIVER

This opens the Debug TRANSCEIVER dialog box, which is shown in the following example.

onfiguration Report Smar	BERT LION	hack Modes	Static Patte	Transmit Eye Monitor	
anes		PF_XCVR_1			
Physical Location	Q0_LANE0	Q3_LANE3	Q1_LANES	Q2_LANE2	
Tx PMA Ready	•	•	•	•	
Rx PMA Ready	•	•	•	•	
TX PLL	•	•	•	•	
RX PLL	•	•	•	•	
Data Width	40 bit	8 bit	0 bit	0 bit	
A LANEL					
Physical Location	Q0_LANE1	Q3_LANE2	Q1_LANE1	Q2_LANE0	
Tx PMA Ready	•	•	•	•	
Rx PMA Ready	•	•	•	•	
TX PLL	•	•	•	•	
RX PLL	•	•	•	•	
Data Width	40 bit	8 bit	0 bit	0 bit	
LANE2					
Physical Location	Q0_LANE2	Q3_LANE1	Q1_LANE2	Q2_LANES	
Tx PMA Ready	•	•	•	•	
Rx PMA Ready	•	•	•	•	
TX PLL	•	•	•	•	
RX PLL	•	•	•	•	
Data Width	40 bit	8 bit	0 bit	0 bit	
A LANES					
Physical Location	Q0_LANES	Q3_LANE0	Q1_LANE0	Q2_LANE1	
Tx PMA Ready	•	•	•	•	
Rx PMA Ready		•	•	•	
TX PLL	•	•	•	•	
RX PLL	•	•	•	•	
Data Width	40 bit	8 bit	0 bit	0 bit	
					Refres
Help					

Figure 27 · Debug Transceiver Dialog Box

Debug Transceiver has five distinct debug features, which are represented by tabs in the Debug TRANSCEIVER dialog box:

- <u>Configuration Report</u> (shown by default when the dialog box opens)
- SmartBERT
- Loopback Modes
- Static Pattern Transmit
- Eye Monitor



### **Demo Mode**

The following example shows Debug Tansceiver in demo mode.

TANKO         Construction of the construle of the construction of the construction of the con	nes	CM1 PCTe SS O/PE PCTE 0	CM1_PCIe_SS_0/XCVR_2_3_0/PF_XCVR_0	
Physical location         Q0_LANE2           Tr PNA Ready         CO_LANE2           Tr PNA Ready         CO_LANE2           TX PLL         CO_LANE3           Physical location         Q0_LANE1           Physical location         Q0_LANE3           The PLA Ready         CO_LANE3           The PLA Ready         CO_LANE3           The PLA Ready         CO_LANE3           TA PLA Ready         CO_LANE3           THE PLA Ready         CO_LANE3           TA PLA READY         CO_LANE3		and an point of an or		
To PUA Ready         PARA Ready           TX, PUA Ready         PARA Ready           TX, PUL         PARA Ready           Data Worth         40 bit           Data Worth         40 bit           Data Worth         40 bit           Data Worth         40 bit           Data Worth         00 LANEI           TS PUA Ready         Para Ready           TA Ready         PARA Ready		O0 LANE0	OD LANE2	
Rc PAA Ready         Image: Control of the contro	Tx PMA Ready	•		
TX PLL         PR PLL           Deta Wideh         40 bit           Deta Wideh         40 bit           Deta Wideh         40 bit           Presida Ilocation         Q0 LANE1           Dep Nataway         0           By Nataway         0           RX PLL         0	Rx PMA Ready		ě.	
Data Wideh 40 bit 40 bit LANET Physical Continn OQ_LANET OQ_LANES To PMA Ready To PMA Ready TA PUL TX PUL	- TX PLL			
LANE1         PDpsical Location         C0_LANE1         C0_LANE3           To FMLA Randy         Image: Control Contro Control Contron Control Control Contron Control Control Control C	RX PLL	•	•	
Physical location         Q0_LAMEI         Q0_LAMEI           To FMAR.Backy         C         C           Sc FMAR.Backy         C         C           Sc FMAR.Backy         C         C	Data Width	40 bit	40 bit	
Physical Location     Q0_LANE     Q1_ANE     Q1_Q1_ANE     Q1_ANE     Q1				
Te MA Rady Ro MA Rady TX PLL RX PLL		001111171		
- Rz PMA Ready - TX PLL - RX PLL	Physical Location		QU_LANES	
	IX PMA Keady	-		
RX PLL	TX OIL			
Usta Wildth 40 bit 40 bit	Port It of the			
	Data width	40 DR	40 DK	
Facto				n.6

#### **Configuration Report**

The Configuration Report is the first tab in the Debug TRANSCEIVER dialog box, and is shown by default when the dialog box opens. The Configuration Report shows the lane status/health properties of all lanes of Quads in the design.

Click the **Refresh** button to refresh the information.

Note: The report refreshes automatically when you navigate from another tab.

State         PF_XCR_0         PF_XCR_1         PF_XCR_2         PF_XCR_0         PF_XCR_0 <t< th=""><th>onfiguration Report S</th><th>mart BERT Loop</th><th>pback Modes</th><th>Static Patter</th><th>Transmit Eye Monitor</th><th></th></t<>	onfiguration Report S	mart BERT Loop	pback Modes	Static Patter	Transmit Eye Monitor	
Tr. PMA Ready	anes	PF_XCVR_0	PF_XCVR_1	PF_XCVR_2	PF_XCVR_3	
Resdy       V       V       V         TX PLL       V       V       V         RVPL       V       V       V         Data Width       40 bit       8 bit       0 bit         / LMH2       V       V       V         Physical Location       00 LANEI       02 LANEI       02 LANEI         TreMA Ready       V       V       V         Re PMA Ready       V       V       V         Physical Location       00 LANEI       02 LANEI       V         Re PMA Ready       V       V       V         Re PMA Ready       V       V       V         Physical Location       00 LANE2       02 LANEI       V         Re PMA Ready       V       V       V       V         Re PMA Ready       V	Physical Location	Q0_LANE0	Q3_LANE3	Q1_LANES	Q2_LANE2	
TXPLL       • <td>Tx PMA Ready</td> <td>•</td> <td>•</td> <td>•</td> <td></td> <td></td>	Tx PMA Ready	•	•	•		
KPUL         O         O         O           Data Width         40 bit         8 bit         0 bit         0 bit           / LMNE1         Physical Location         00 LANE1         01 LANE1         02 LANE1         02 LANE1           Physical Location         00 LANE1         01 LANE2         02 LANE1         02 LANE1         02 LANE1           TX PML         0         0         0 bit         0 bit         0 bit           VERMA Ready         0         0         0 bit         0 bit         0 bit           Physical Location         00 LANE1         0 LANE2         0 LANE1         0 bit         0 bit           Physical Location         00 LANE2         02 LANE2         02 LANE2         02 LANE3         0 bit           TX PML         0         0         0 bit         0 bit         0 bit         0 bit           TX PML         0         0         0 bit         0 bit         0 bit         0 bit         0 bit           TX PML         0         0         0 bit         0 bit         0 bit         0 bit         0 bit           TX PML         0         0 bit         0 bit         0 bit         0 bit         0 bit           TX PML	Rx PMA Ready	•	•	•	•	
Data Width         40 bit         8 bit         0 bit           2 Matti         Physical location         00 LANEI         02 LANEI         02 LANEI           Physical location         00 LANEI         02 LANEI         02 LANEI           Tre PhA Ready         0         0         0           Re PhA Aready         0         0         0           Physical location         02 LANEI         02 LANEI         0 LaNEI           Physical location         02 LANEI         0 LaNEI         0 LaNEI           Physical location         02 LANEI         0 LaNEI         0 Linei           RK PLL         0         0         0         0           TX PL         0         0         0         0           RK PL         0         0         0         0           TX PL         0         0         0         0           Physical location         02 LANEI         0         0		•	•	•		
I MREI           Physical Location         00 LANE         02 LANE         02 LANE         02 LANE           To PAA Ready         •         •         •         •           Physical Location         00 LANE         02 LANE         02 LANE         •           To PAA Ready         •         •         •         •           TX PUL         •         •         •         •           Physical Location         02 LANE2         02 LANE3         •         •           TX PUL         •         •         •         •         •           Physical Location         02 LANE2         02 LANE3         •         •         •           TX PUL         •         •         •         •         •         •           Physical Location         02 LANE3         •         •         •         •         •           TX PUL         •         •         •         •         •         •         •           TX PUL         •         •         •         •         •         •         •           TX PUL         •         •         •         •         •         •         •         •         • <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>						
Physical Location         00 LANEI         02 LANEI <td>Data Width</td> <td>40 bit</td> <td>8 bit</td> <td>0 bit</td> <td>) bit</td> <td></td>	Data Width	40 bit	8 bit	0 bit	) bit	
Physical Location         00 LANE         02 LANE         02 LANE           Tr VMA Ready         Image: Control of the control of t	4 LANEL					
Tir PMA Ready       Image: Constraint of the second s		00 LANE1	03 LANE2	O1 LANE1	02 LANF0	
Re. PMA Ready       Image: Control of the second seco						
TXPLL       • <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>						
RVPLL       O <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>						
Data Width         40 bit         8 bit         0 bit         0 bit                IARUZ               Physical location               QLANE2               QLANE2               QLANE2               QLANE2               QLANE2               QLANE2               QLANE2               QLANE2             QLANE3               QLANE3               QLANE3               QLANE3               QLANE3               QLANE3               Data               Data               Data               QLANE3               QLANE3               QLANE3               Data						
Physical Location         00 (LANE2         02 (LANE1         02 (LANE3           Tic PMA Ready         Image: Control of the state of the st		40 bit	8 bit	0 bit	) bit	
Physical Location         00 (LANE2         02 (LANE1         02 (LANE3           Tic PMA Ready         Image: Control of the state of the st						
Tir DMA Ready       Image: Constraint of the second s	4 LANE2					
Re, PMA Ready     Image: Constraint of the second sec	Physical Location	Q0_LANE2	Q3_LANE1	Q1_LANE2	Q2_LANE3	
TXPLL       • <td>Tx PMA Ready</td> <td></td> <td>•</td> <td>•</td> <td>•</td> <td></td>	Tx PMA Ready		•	•	•	
RV PLL Data Width 40 bit 8 bit 0 bit 0 bit ANNES Physical Location 00 LANES 03 LANE0 01 LANE0 02 LANE1 To PMA Ready Rc PMA Ready Rc PMA Ready Data Width 40 bit 8 bit 0 bit 0 bit CR PLL Bit 0 bit 0 bit	Rx PMA Ready	•	•	•	•	
Data Width         40 bit         8 bit         0 bit         0 bit           / LANE3         Physical Location         C0_LANE0         Q1_LANE0         Q2_LANE1           Tk_PMA Ready         0         0         0         0         0           TX_PUL         0         0         0         0         0           Data Width         40 bit         8 bit         0 bit         0         0		•	•	•	•	
LANE3     Physical Location     Q0 LANE3     Q3 LANE0     Q1 LANE0     Q2 LANE1     Tr.PMA Ready     Rc PMA Ready     Rc PMA Ready     Data Width     40 bit     8 bit     0 bit     0 bit     Refeath     Refeath	RX PLL	•	•	•	•	
Physical Location OQ LANE3 OZ LANE0 OZ LANE1 Tr PAA Ready Ro PAA Ready TX FPIL Data Width 40 bit 8 bit 0 bit 0 bit	Data Width	40 bit	8 bit	0 bit	) bit	
Physical Location OQ LANE3 OZ LANE0 OZ LANE1 Tr PAA Ready Ro PAA Ready TX FPIL Data Width 40 bit 8 bit 0 bit 0 bit	4 LANES					
Tir PMA Ready Ric PMA Ready TX PLL RX PLL Data Width 40 bit 8 bit 0 bit 0 bit Referance		00 LANES	O3 LANED	O1 LANED	D2 LANE1	
Ry PMA Ready TX PLL • • • • RX PLL • • • Data Width 40 bit 8 bit 0 bit 0 bit			Contractor Contractor	AL CHINED	defenses	
TX PLL PL						
RX PLL		ě				
Data Width 40 bit 8 bit 0 bit Refeat						
						Refresh

Figure 28 · Debug TRANSCEIVER Dialog Box - Configuration Report

The Configuration Report shows the physical location, status/health, and data width for all lanes of all the quads enabled in the system controller.

Parameter information is shown in table format, with lane numbers as rows and transceiver instance names as columns.

The lane parameters are as follows:

Physical Location - Physical block and lane location in the system controller.

Tx PMA Ready - Indicates if the Tx of the lane is powered up and ready for transactions.

**Rx PMA Ready** - Indicates if the Rx of the lane is powered up and ready for transactions.



TX PLL - Indicates if the lane is locked onto TX PLL.

RX PLL - Indicates if the lane is locked onto RX PLL.

Data Width - Data Width of the Lane.

For the parameters above, green indicates true and red indicates false.

Notes:

Click the **Refresh** button to update the lane status.

The report refreshes automatically when you navigate from another tab.

### **Transceiver Hierarchy**

Transceiver Hierarchy is a lane hierarchy with all the lanes instantiated in the design shown with respect to top level instance.

Transceiver Hierarchy is shown in the following tabs: "SmartBERT" on page 43, "Loopback Modes" on page 47, "Static Pattern Transmit" on page 49, and "Eye Monitor" on page 50.

In the SmartBERT, Loopback Modes, and Static Pattern Transmit pages, check boxes allow multiple lanes to be selected for debug, as shown in the following example.

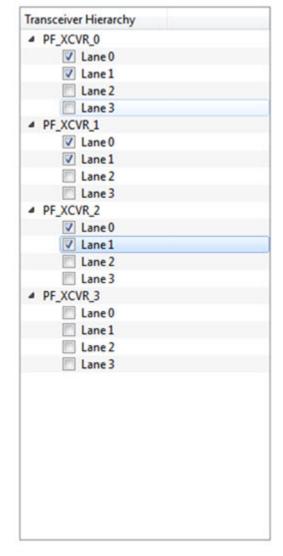


Figure 29 · Transceiver Hierarchy Lane Selection Example - SmartBERT, Loopback Modes, and Static Pattern Transmit Pages



In the Eye Monitor page, eye monitoring is done one lane at a time, as shown in the following example.

▲ PF_XCVR_0 Lane 0 Lane 1 Lane 2 Lane 3 ▲ PF_XCVR_1 Lane 0 Lane 1 Lane 2 Lane 3 ▲ PF_XCVR_22 Lane 0 Lane 1 Lane 2 Lane 3 ▲ PF_XCVR_3	Transo	eiver		
Lane 1 Lane 2 Lane 3 PF_XCVR_1 Lane 0 Lane 1 Lane 2 Lane 3 PF_XCVR_2 Lane 0 Lane 1 Lane 2 Lane 0 Lane 2 Lane 3	⊿ PF	_XCVR_0		
Lane 2 Lane 3 PF_XCVR_1 Lane 0 Lane 1 Lane 2 Lane 3 PF_XCVR_2 Lane 0 Lane 1 Lane 2 Lane 2 Lane 3		Lane 0		
Lane 3  PF_XCVR_1  Lane 0  Lane 1  Lane 2  Lane 3  PF_XCVR_2  Lane 0  Lane 1  Lane 2  Lane 2  Lane 3		Lane 1		
<ul> <li>PF_XCVR_1</li> <li>Lane 0</li> <li>Lane 1</li> <li>Lane 2</li> <li>Lane 3</li> <li>PF_XCVR_2</li> <li>Lane 0</li> <li>Lane 1</li> <li>Lane 2</li> <li>Lane 3</li> </ul>		Lane 2		
Lane 0 Lane 1 Lane 2 Lane 3 PF_XCVR_2 Lane 0 Lane 1 Lane 2 Lane 3		Lane 3		
Lane 1 Lane 2 Lane 3 PF_XCVR_2 Lane 0 Lane 1 Lane 2 Lane 3	▲ PF	XCVR_1		
Lane 2 Lane 3 PF_XCVR_2 Lane 0 Lane 1 Lane 2 Lane 3		Lane 0		
Lane 3 PF_XCVR_2 Lane 0 Lane 1 Lane 2 Lane 3		Lane 1		
<ul> <li>PF_XCVR_2</li> <li>Lane 0</li> <li>Lane 1</li> <li>Lane 2</li> <li>Lane 3</li> </ul>		Lane 2		
Lane 0 Lane 1 Lane 2 Lane 3		Lane 3		
Lane 1 Lane 2 Lane 3	₫ PF	_XCVR_2		
Lane 2 Lane 3		Lane 0		
Lane 3		Lane 1		
		Lane 2		
▷ PF_XCVR_3		Lane 3		
	▷ PF	XCVR_3		

Figure 30 · Transceiver Hierarchy Lane Selection Example - Eye Monitor Page

### **SmartBERT**

In the SmartBERT page of the Debug TRANSCEIVER dialog box, you can select lanes in the Transceiver Hierarchy and use debug options to run Smart BERT tests.

Click the SmartBERT tab in the Debug TRANSCEIVER dialog box to open the SmartBERT page.



onfiguration Report	Smart BERT	Loopback Modes	Static Pattern Transmit	Eye Monitor									
Transceiver Hierarchy	Physical Loca	ation 4		Pattern	EQ-NearEnd	TX PLL	RX PLL	Lock to Data	Cumulative Error Count	Data Rate (Gbps)	BER	Error Counter	
PF_XCVR_0													
✓ LANEO	Q0_LANE0		PF_XCVR_0/LANE0	PRBS7 *	Enable	•	•	•	0	5	1.54e-11	Reset	
LANE1	Q0_LANE1												
✓ LANE2	Q0_LANE2			-	Dent	-	-	-					
LANE3	Q0_LANE3		PF_XCVR_0/LANE2	PRBS15 *	Enable	•	•	•	NA	5	NA	Reset	
PF_XCVR_1													
✓ LANE0	Q3_LANE3		PF_XCVR_1/LANE0	PRBS9 +	Enable				0	5	1.54e-11	Reset	
✓ LANE1	Q3_LANE2		PF_ACVICI/David	11000		-	-	-	0	1	1746-11	Medee	
LANE2	Q3_LANE1												
LANE3	Q3_LANE0		PF_XCVR_1/LANEL	PRBS23 *	Enable	•	•	•	0	5	1.54e-1	Reset	
PF_XCVR_2													
LANE0	Q1_LANE3												
LANE1	Q1_LANE1		PF_XCVR_2/LANE3	PRBS15 *	Enable	•	•	•	0	5	1.54e-1	Reset	
✓ LANE2	Q1_LANE2												
✓ LANE3	Q1_LANE0				-		1	1					
+ PF_XCVR_3	03110103		PF_XCVR_2/LANE2	PRBS31 -	🗸 Enable	•	•	•	0	5	1.54e-11	Reset	
LANE0	Q2_LANE2 Q2_LANE0												
LANE2	Q2_LANE3		PF_XCVR_3/LANEL	PRBS9 -	Enable				NA	5	NA	Reset	
✓ LANE3	Q2_LANES		PT_ACTICS/DAVID		C) choose	-	-	-	190	1	194	NCACE.	
U LAINES	Q2_LAINEL												
			PF_XCVR 3/LANE3	PRBS15 *	✓ Enable	•	•	•	0	5	1.54e-11	Reset	
			Constraint Services										
Phy Reset			Start										Stop
			_										

Figure 31 · Debug TRANSCEIVER Dialog Box - SmartBERT Page

The following input options and outputs are represented as columns:

**Pattern** – Input option. Select a PRBS pattern type from the drop-down list: PRBS7, PRBS9, PRBS15, PRBS23, or PRBS31. The default is PRBS7.

EQ-NearEnd – Input option. When checked, enables EQ-NearEnd loopback from Lane Tx to Lane Rx.

TX PLL – Indicates if lane is locked onto TX PLL when the SmartBERT test is in progress.

Gray - Indicates test is not in progress

Green - Indicates lane is locked onto TX PLL

Red – Indicates lane is not locked onto TX PLL

RX PLL - Indicates if lane is locked onto RX PLL when the SmartBERT test is in progress.

Gray - Indicates test is not in progress

Green - Indicates lane is locked onto TX PLL

Red – Indicates lane is not locked onto TX PLL

Lock to Data – Indicates if lane is locked onto incoming data / RX CDR PLL when the SmartBERT test is in progress.

Gray - Indicates test is not in progress

Green - Indicates lane is locked onto TX PLL

Red - Indicates lane is not locked onto TX PLL

.Cumulative Error Count – Displays the error count when the SmartBERT test is in progress.

Data Rate - Indicates the data rate (in Gbps) configured in the lane.

**BER** – Calculates the Bit Error Rate (BER) from the cumulative error count and data rate and displays it in the column.

Error Counter Reset - Resets the error counter and BER of the lane. A reset can be done at any time.

All output parameters are updated approximately once per second, with their values retrieved from the device.

To add lanes, in the Transceiver Hierarchy, check the boxes next to the lanes to be added. To remove lanes, uncheck the boxes next to the lanes to be removed.

Select the desired options and click Start to start the Smart BERT test on all selected lanes.

**Note:** A popup message appears if a test cannot be started on one lane, multiple lanes, or all lanes. Tests will start normally on all unaffected lanes.

Click the **Phy Reset** button to do a Phy reset on all checked lanes in the Transceiver Hierarchy. This button is disabled when a PRBS test is in progress.



**Note:** You can navigate to other tabs when a SmartBERT test is in progress, but you cannot perform any debug activity except to use Plot Eye for any lane on the Eye Monitor page.

**Note:** You cannot close the SmartBERT window when a test is in progress. Attempting to do so will result in the following message:

Trar	nsceiver Debug	X
	Cannot close, debug activity is in p	rogress.
		ок

Click the **Stop** button to stop the SmartBERT test on all lanes simultaneously.

#### **SmartBERT IP**

**Important Note:** There is currently an issue in which JTAG writes to some Probe Points that are not working as expected. A software workaround has been provided to determine which Probe Points are working as expected by probing them from SmartDebug.

See TU0804: PolarFire FPGA: Debugging FPGA Design Using SmartDebug Tutorial for details.

The CoreSmartBERT core provides a broad-based evaluation and demonstration platform for PolarFire transceivers (PF\_XCVR). Parameterizable to use different transceivers and clocking topologies, the SmartBERT core can also be customized to use different line rates and reference clock rates. Data pattern generators and checkers are included for each PF\_XCVR, giving several different Pseudo-random binary sequences PRBS (27,223, 215 and 231).

Each SmartBERT IP can have four lanes configured. Each Lane can have the pattern type PRBS7, PRBS9, PRBS23, or PRBS31 configured.

SmartDebug identifies the lanes that are used by the SmartBERT IP and distinguishes them by adding "\_IP" to the SmartBERT IP instance name in the Transceiver Hierarchy. See the following example.

You can expand a SmartBERT IP instance to see all the lanes. Check the checkbox next to a lane to add it to the SmartBERT IP page and include the lane in a PRBS test. If the box is unchecked, it will not be added. See the following example.

101 De



	-												
Constitution	Physical 4		Pattern	EQ-NearEnd	TOPL	RICPUL L	Lock to Deta	Cumulative Error Count	Data Rate (Gps)	BOR	Error Counter	Error Injection	4
Z LANEI Z LANEI	Q2_LAN Q2_LAN	Smart80RT_L4_0(Smart80RT IP)/LANE0	PE8531(Sear6887.3P) +	Drable	•	• •	•	2	\$.25	4.)6e-11	Aeset	Inject Error	
C LANE2	Q2,LAN Q2,LAN	SmartBERT_L4_D(SmartBERT IP)/LANE1	PRESZI(SeartBERT IP) *	Druble	•	• •	•	4	6.25	145e-11	Reset	Inject Error	
		SmartBERT_L4_0(SmartBERT IP)/LANE2	PRESIGNATION +	🗆 trable	•	• •	•	2	6.25	3.69e-11	Reset	Inject Error	
		SmartBERT_L4_NSmartBERT P3/LANE)	PHBS7(SeardBR11P) +	D trable	•	• •	•	1	625	114e-11	Reset	Inject Error	

You can select patterns for the added lane(s) from a drop-down list. See the following example.

iver Herarchy	Physical Loca 4		Pattern	EQ-NewEnd	TX PLL F	RX PLL L	ock to Data	Cumulative Error Count	Data Rate (Gbps)	BER E	nor Counter	Error Injection		4	
SmartBERT_L4_0(SmartBERT)	C2_LANE0	SmartBERT_L4_0(SmartBERT IP)/LANEO	PR8523Qinert8ERT IP)	IF trable			0	6		2.49e-11	Reset	Prject fire	<i>y</i>		-
E LANEZ LANEZ	Q2_LANE2 Q2_LANE3	SmartBERT_LA_O(SmartBERT IP)/LANE1	PR8523(Smart8ERT IP)	P thebie	•	•	•	1	625	3.20e-11	Reset	Insect Dro	97 I		
		SmartBERT_L4_0(SmartBERT IP)/LANE2	P8859 .	IF trable		0	0	0	6.25	1.56e-12	Reset	Inject fire	y .		
		SmartBERT_L4_0(SmartBERT IP)/LANES	PR8515	IF trable	•	•	•	0	6.25	3.56e-12	Reset	briect Dro	×		
			PRES31(SmartBERT IP) PRES7(SmartBERT IP) PRES9(SmartBERT IP) PRES7												
			PR059 PR059 PR0515 PR0523												
			#82531												
		I												Step	
Reset		Start													

After the lane(s) have been added and the patterns(s) selected, click **Start** to enable the transmitter and receiver for the added lanes and patterns.

#### **Error Injection**

When SmartBERT IP lanes are added, you will see the Error Injection column and Error Inject button. Errors can be injected by clicking the **Error Inject** button when a PRBS test is running. This feature tests whether the error is identified by the pattern checker.

**Note:** This column does not appear for non-SmartBERT IP lanes, or if a non-configured PRBS pattern has been selected.

#### **Error Count**

Error Count is shown when a lane is added and a PRBS pattern is run. The error count can be cleared by clicking the **Reset** button under the Error Counter column.

The following example shows the Reset and Inject Error buttons.



	Physical Loca 4		Pattern	EQ-NearEnd	TXPLL	RX PLL	Lock to Data	Cumulative Error Count	Ceta Rate (Gbps)	BER I	Error Counter	Error Injection		•
CANED	Q2_LANE0 Q2_LANE1 Q2_LANE2	SmartBERT_L4_0(SmartBERT IP)/LANE0	PRESZUCIANE REPORT (P) +	P trable	•	•	•	6	6.25	1.39e-11	Reset	inject Bry	w l	_
Z LAND Z LAND	Q2_LANE Q2_LANE	SmartBERT_L4_0(SmartBERT IP)/LANE1	PR0523(Imar 009/7 IP)	🗗 truble	•	•	•		6.25	436e-11	Reset	Inject for	-	
		SmartBERT_L4_0(SmartBERT IP)/LANE2	PR855 +	P trable	•	•	•	0	6.25	4.85e-12	Reset	Inject Dr	w.	
		SmartBERT_L4_0(SmartBERT IP)/LANE3	-	P trubie	•	•	•	0	6.25	4.85e-12	Reset	byea by	w I	
	1													

### **Demo Mode**

The following example shows the SmartBERT	Γ page in demo mode.
---	----------------------

nsceiver Hierarchy Physical Location	4		Pa	ttern	EQ-NearEnd	TX PLL F	X PLL Lock	to Data	Cumulative Error Count	Data Rate (Gbps)	BER	Error Counter	Error Injection	4
CM1_PCIe_SS_0 = PF_PCIE_0 - PI_LANE0 Q0_LANE0		CM1_PCIe_SS_0/PF_PCIE_0/	LANEO	R857 ¥	F Enable	•	• •		0	5	5.00e-11	Reset		
- CANEL QOLANEL CM1_PCIe_SS_0		CM1_PCIe_SS_0/PF_PCIE_0/		RBS7 -	F Enable	•	• •		NA	5	NA	Reset		
														-1

### **Loopback Modes**

The Loopback Modes page in the Debug TRANSCEIVER dialog box allows you to select lanes from the Transceiver Hierarchy and use Loopback Mode debug options.

Click the **Loopback Modes** tab in the Debug TRANSCEIVER dialog box.



nfiguration Report	Smart BERT Loopbac	k Modes	Static Pattern Transmit	Eye Monitor					
ansceiver Hierarchy	Physical Location	4	PF_XCVR_1/LANE0	C EQ-NEAR END	C EQ-FAR END	COR FAR END	No Loopback		
PF_XCVR_0			PF_XCVR_1/LANE2	C EO-NEAR END	. EO-FAR END	COR FAR END	No Loopback		
LANE0	Q0_LANE0		PF_XCVR_0/LANE2			and the second second second			
LANE1	Q0_LANE1								
LANE2	Q0_LANE2		PF_XCVR_0/LANE0	EQ-NEAR END	C EQ-FAR END	COR FAR END	No Loopback		
LANE3	Q0_LANE3								
PF_XCVR_1									
LANEO	Q3_LANE3								
LANE1	Q3_LANE2								
LANE2	Q3_LANE1								
LANE3	Q3_LANE0								
PF_XCVR_2									
LANE0	Q1_LANE3								
LANE1	Q1_LANE1								
LANE2	Q1_LANE2								
LANE3	Q1_LANE0								
PF_XCVR_3									
LANE0	Q2_LANE2								
LANE1	Q2_LANE0								
LANE2	Q2_LANE3	1 I							
LANE3	Q2_LANE1								
		-							
hy Reset									Apply

Figure 32 · Debug TRANSCEIVER Dialog Box - Loopback Modes Page

You can select the desired loopback type (EQ-NEAREND, EQ-FAREND, CDRFAREND, or No Loopback) for each lane.

**EQ-NEAR END** – Set EQ-Near End loopback from Lane Tx to Lane Rx. This loopback mode is supported up to 10.3125 Gbps.

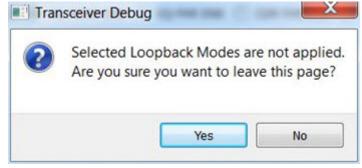
**EQ-FAR END** – Set EQ-Far End loopback from Lane Tx to Lane Rx.

CDR FAR END – Set CDR Far End loopback from Lane Rx to Lane Tx.

**No Loopback** – Set this option to have no loopback between Lane Tx and Lane Rx. (For external loopback using PCB backplane or High Speed Loopback cables.)

When you have selected the desired options, click **Apply** to enable the selected loopback mode on the lane(s).

**Note:** If you proceed to another tab without applying your changes to loopback modes, the following popup message appears:



Click **Yes** to ignore the changed selections and move to another selected page. Click **No** to remain on the current page.

### **Demo Mode**

The following example shows the Loopback Modes page in demo mode.



A second second second	pback Modes Static Pattern Transmit Eye Monitor	
ansceiver Hierarchy Physical Locatio	d         - CMT_PCIe_SS_0PF_PCE_0LANE         C EQ4EAR EDD         C EQ4AR EDD         C CMT_PCIe_SS_0PF_PCE_0LANE         C EQ4EAR EDD         C CMT_PCIE_SS_0PF_PCE_SS_0PF_P	
CM1_PCIe_SS_0		
Hy Reset		Apply

### **Static Pattern Transmit**

In the Static Pattern Transmit page of the Debug TRANSCEIVER dialog box, you can select lanes from the Transceiver Hierarchy and use Static Pattern Transmit debug options.

Click the **Static Pattern Transmit** tab in the Debug TRANSCEIVER dialog box to open the Static Pattern Transmit page.

nfiguration Report	Smart BERT	Loopback Modes	Static Pattern Transmit	Eye Monitor						
ansceiver Hierarchy	Physical Loca	tion 4		Pattern	Value	Mode	TX PLL	RX PLL	DataWidth	
PF_XCVR_0	Q0_LANE0		PF_XCVR_3/LANE2	Fixed Pattern	• 101010	HEX	•	•	0 bit	
LANE1	Q0_LANE1 Q0_LANE2 Q0_LANE3		PF_XCVR_3/LANE0	Fixed Pattern	• 101010	HEX	•	•	0 bit	
PF_XCVR_1	Q3_LANE3 Q3_LANE2		PF_XCVR_2/LANE2	Fixed Pattern	•	HEX	0	•	0 bit	
LANE2     LANE3     PF_XCVR_2	Q3_LANE1 Q3_LANE0		PF_XCVR_2/LANE0	Fixed Pattern	• 10 10 10	HEX	•	•	0 bit	
LANE0 LANE1 LANE2	Q1_LANE3 Q1_LANE1 Q1_LANE2		PF_XCVR_0/LANE2	Fixed Pattern	• 10 10 10	HEX	•	•	40 bit	
PF_XCVR_3	Q1_LANE0 Q2_LANE2		PF_XCVR_0/LANE3	Fixed Pattern	• 101010	HEX	•	•	40 bit	
LANE1	Q2_LANE0 Q2_LANE3 Q2_LANE1		PF_XCVR_1/LANE1	Fixed Pattern	• 101010	HEX	•	•	8 bit	
			PF_XCVR_1/LANE2	Fixed Pattern	• 10 10 10	HEX	٠	٠	8 bit	
Phy Reset			Start							Stop

Figure 33 · Debug TRANSCEIVER Dialog Box – Static Pattern Transmit Page

When a lane is added from the Transceiver Hierarchy, the following debugging options can be selected: **Pattern** – Fixed Pattern, Max Run Length Pattern, and User Pattern can be selected from the drop-down list.

- Fixed Pattern is a 10101010... pattern. Length is equal to the data width of the Tx Lane.
- Max Run Length Pattern is a 1111000... pattern. Length is equal to the data width of the Tx Lane, with half 1s and half 0s.
- User Pattern is a user defined pattern in the value column. Length is equal to the data width.

Value – Editor available only with the User Pattern pattern type. For other pattern type selections, it is disabled.

- Takes the input pattern to transmit from the Lane Tx of selected lanes.
- Pattern type should be Hex numbers, and not larger than the data width selected.
- Internal validators dynamically check the pattern and indicate when an incorrect pattern is given as input.



Mode – Currently, HEX mode is supported for pattern type.

TX PLL - Indicates Lane lock onto TX PLL when Static Pattern Transmit is in progress

- Gray Test is not in progress
- Green Lane is locked onto TXPLL
- Red Lane is not locked onto TXPLL

RX PLL - Indicates Lane lock onto RX PLL when Static Pattern Transmit is in progress

- Gray Test is not in progress
- Green Lane is locked onto RXPLL
- Red Lane is not locked onto RXPLL

**Data Width** – Displays the data width of the configured lane. Can be used as reference when giving the user pattern.

Click Start to start Static Pattern Transmit on selected lanes.

Click Stop to stop Static Pattern Transmit test on selected lanes.

#### **Demo Mode**

Debug TRANSCEIVER	(DEMO MODE)										-	
onfiguration Report	Smart BERT   Loopba	k Modes	Static Pattern Transmit   Eye Monitor	1								
Transceiver Hierarchy	Physical Location	•[		Pattern		Value	Mode	TX PLL	RX PLL	DataWidth		•
	Q0_LANE0 Q0_LANE1		CM1_PCIe_SS_0/PF_PCIE_0/LANE0	Fixed Pattern	•	101010	HEX	•	•	40 bit		
E CM1_PCIe_SS_0	QULANEI		CM1_PCIe_SS_0/PF_PCIE_0/LANE1	Fixed Pattern	-	101010	HEX	•	•	40 bit		

#### **Eye Monitor**

You can determine signal integrity with the Eye Monitor feature. It allows you to create an eye diagram to measure signal quality. Eye Monitoring estimates the horizontal and vertical eye-opening at the receiver serial data sampling point and helps you select an optimum data sampling point at the receiver.

To use the Eye Monitor feature, do the following:

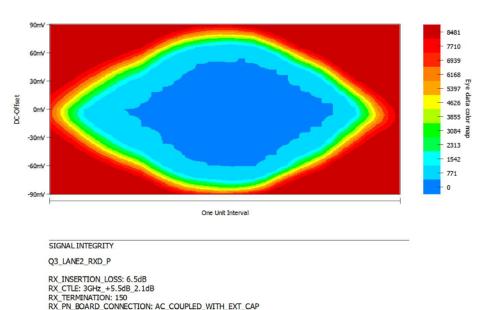
- 1. Invoke SmartDebug from Libero.
- 2. Click the Eye Monitor tab in the Debug TRANSCEIVER dialog box.

In the Eye Monitor page, you can select a lane and click **Plot Eye** to start eye monitoring for that lane. The eye diagram displays, as shown in the following example.

Note: Ensure data transmission on Lane Rx for successful monitoring.



LaneName: SmartBERT\_L4\_0(SmartBERT IP)/LANE2



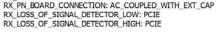
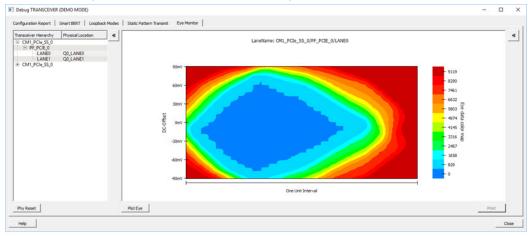


Figure 34 · Eye Monitor Example

You can move to the SmartBERT or Static Pattern Transmit page and start a SmartBERT test or Static Pattern Transmit (using external high speed board cables), respectively, which sends traffic in Lane Rx. You can then return to the Eye Monitor page and click **Plot Eye**.

### **Demo Mode**



The following example shows the Eye Monitor page in demo mode.

### **Signal Integrity**

The Signal Integrity feature in SmartDebug works with Signal Integrity in the I/O Editor, allowing the import and export of .pdc files.



The Signal Integrity pane appears in the following SmartDebug pages:

- SmartBERT
- Loopback Modes
- Static Pattern Transmit
- Eye Monitor

When you open Debug Transceiver in SmartDebug and click the SmartBERT, Loopback Modes, Static Pattern Transmit, or Eye Monitor tab, all parameters in the Signal Integrity pane are shown as Undefined. Only the Export All Lanes and Import All Lanes buttons are enabled. See the following example.

sceiver Hierarchy	Physical Location		Pattern	EQ-NearEnd TX P ()	Signal Integrity: TX	RX
CMT_PCIe_SS_0 CMT_PCIe_SS_0 BXCVR_2_3_0 BPFXC_					TX Emphasis Amplitude	RX Insertion Loss
B D PF XC.						25.068
L L LA	Q0_LANE2 Q0_LANE3				TX Impedance ( ohms )	N/A.
	do contra					- N/A
					TX Transmit Common Mode Adjustment (% of VD	AA) RX CTLE
						<ul> <li>No_Peak_+10.7d8</li> </ul>
					Polarity (P,N reversal)	RX Termination (ohms)
					Inverted	150
						RX P/N Board Connection
						DC_COUPLED
						RX Loss of Signal Detector - Low
						1
						RX Loss of Signal Detector - High
						1
		1000				Washington 1997
		•		<u></u>	Export Import	Design Defaults Apply
y Reset		Start		Stop	Export Al Lanes Import Al Lanes	

When a lane is selected in the SmartBERT, Loopback Modes, Static Pattern Transmit, or Eye Monitor pages, the corresponding Signal Integrity parameters (configured in the I/O Editor or changed in SmartDebug) are enabled and shown in the Signal Integrity pane. See the following example.

ansceiver Hierarchy	Physical Location	4		Pattern		D	Signal Integrity: CM1_PCIe_SS_0/XCVR_2_3_0/PF_	CVR_0/LANEO
CM1_PCIe_SS_0 CM1_PCIe_SS_0 E   XCVR_2.3_0			CM1_PCH_SS_0/XCVR_2_3_0/PF_XCVR_0/LANE0	PR857	•	_	LANED_TXD_P/N TX Emphasis Amplitude 200mV_with2.5d0	LANEO_RXD_P/N RX Insertion Loss 25.0dB
E PRIC PLA PLA	Q0_LANE2 Q0_LANE3		CM1_PCIe_SS_0/XCVR_2_3_0/PF_XCVR_0/LANE1	PR857	•		TX Impedance ( ohms ) 85 TX Transmit Common Mode Adjustment (% of VDDA	The transceiver data rate is set to 5000Mbps for th The current settings will configure this port in CDR RX CTLE
							70 rolarity (P/N reversal )	No_Peak_+10.7d8 RX Termination (ohms)
							[Inverted _	300 RX P/N Board Connection
								AC_COUPLED_WITH_EXT_CAP RX Loss of Signal Detector - Low
								4 RX Loss of Signal Detector - High  6
			<u>د</u>		Ŀ	1	Export Import	Design Defaults Accily
try Reset			Start	1	Stop .		Export Al Lanes Import Al Lanes	

The selected lane instance name is displayed in the Signal Integrity group box, and the Export, Import, and Design Defaults buttons are enabled.



You can select options for each parameter from the drop-down for that parameter. Click **Apply** to set the selected transceiver instance with the selected options.

Note: The Apply button is enabled when you make a selection for any parameter.

ansoeiver Hierarchy	Physical Location	] 4 [	Sec	Pattern	EQ-NearEnd	TX PLL RX	F 10	Signal Integrity: PF_XCVR_1/LANE0			
PF_XCVR_1	Q3_LANE0		PF_XCVR_1/LANEO	PR857 -	□ Enable		_	LANE0_TXD_P/N TX Emphasis Amplitude		LANED_FOID_P/N RX Insertion Loss	
				1				400mV_with3.5d8		25.048	
								TX Impedance (ohms)		The transceiver data rate is set to 10137.6H	
								150	-	The current settings will configure this port is	n DFE mode
								TX Transmit Common Mode Adjustment (*	% of VDDA )	RX CTLE	
								50		N/A	
								Polarity (P/N reversal)		RX Termination (phms)	
								Normal	*	100	
										RX P/N Board Connection	
										AC_COUPLED_WITH_EXT_CAP	
										RX Loss of Signal Detector - Low	
										011	
										RX Loss of Signal Detector - High	
										0ff	
hy Reset			•	J		Stop	10 C	Export Import Export Al Lanes		Design Defaults	Apply

If you change parameter options and click another lane, move to another tab, or click Import, Import All, or Design Defaults without applying the changes, you will see the following message:

CM1_PCIe_SS_0	Physical Location	4			Pattern E D	Signal Integrity: CM1_PCIe_S5_0/XC/R LANEO_TXD_P/N	23.004.30	IR_O,LANEO LANEO,RXD_P/N	
CM1 PCIe SS (	)			XCVR_0/LANEO	PR#57 •	TX Emphasis Amplitude		RX Insertion Loss	
B D XCVR 2.3 (						200mV_w/th2.5d8	*	17.0d0	2
E LA	Q0_LANE2 Q0_LANE3		CM1_PCIe_SS_0/XCVR_2_3_0/PF	XCVR_0/LANET	PR857 •	TX Impedance ( ohms )	110	The transceiver data rate is set to 5000Mbp	
e u	Conjunes.					85	•	The current settings will configure this port	In CDR mod
						TX Transmit Common Mode Adjustment		RX CTLE	
						80	-	No_Peak_+10.7d8	
						Polarity (P/N reversal)		RX Termination (ohms)	
						Inverted	•	100	
								RX P/N Board Connection	_
								AC_COUPLED_WITH_EXT_CAP	
				Con a second		and the second se		RX Loss of Signal Detector - Low	
				Transceiver	rDebug			4	
				2 Ser	e Signal Integrity options do you wish to continue	are modified.		RX Loss of Signal Detector - High 6	_
			L						
						Export Import		Design Defaults	Αρρίγ

Click Apply to apply the changes or Discard to discard the changes.

### **Design Defaults**

Clicking the **Design Defaults** button loads the Signal Integrity parameter options for the selected lane instance. These are the signal integrity settings that were selected in the Libero design flow run and reside in the STAPL file. Design Default parameter options are applied to the device and updated in Modified Constraints.



**Note:** Modified Constraints is a list of IO constraints set on the TXP/N and RXP/N lane ports. For a selected lane, this set is created in the SmartDebug session and is updated when a Signal Integrity parameter option is modified and applied or an external PDC file is imported.

ansoeiver Hierarchy	Physical Location	- 41		Pattern	1	D	Signal Integrity: CM1_PCIe_SS_0/XCVR_2_3_0/PF_XC	
© Conf. Pole. SJ. ⇒ Conf. Pole. SJ. ⇒ Di Kong JJ. ⇒ Di Kong JJ	0		- CM1,PCH_SS_0YXCVR_2,3_07F_XCVR_01_ANE0	PR857	-		UNE_IDD_PN TS Enghains Anghlude DDM*, why	UNE (JOD, PA EX Insertion-Less [17:08] The transceiver data rate is set to 5000Hops for this put the current settings will configure this port in CDR and RX CTLE [96, Press, 4: 92.78] [100]
Phy Reset		L	start		1		Export Import Export Al Lanes	Design Defaults Apply

### Export

Clicking the **Export** button exports the current selected parameter options and other physical information for the selected lane instance to an external PDC file. A popup box prompts you to choose the location where you want the .pdc file to be exported.

ransceiver Hierarchy	Physical Location	-		Pattern	0 1	Signal Integrity: CM1_PCIe_SS_0/XCVR_2_3_0/PF_XX	
CM1_PCIe_SS_0			CM1_PCIe_SS_0/XCVR_2_3_0/PF_XCVR_0/LANEO	PR857 -		LANE 1_TXD_P/N TX Emphasis Amplitude	LANE 1_RXD_P/N RX Insertion Loss
B ♥ XCVR.2.3.0 B ♥ PF_XC				-	1	200mV_with6.0d8	17.0d0
- PI LA	Q0_LANE2 Q0_LANE3		CM1_PCIe_SS_0/XCVR_2_3_0/PF_XCVR_0/LANE1	PR857 *		TX Impedance ( ohms )	The transceiver data rate is set to 5000Mbps for this po
- MIA	Q0_LANE3				•	100 -	The current settings will configure this port in CDR mode
						TX Transmit Common Mode Adjustment (% of VDDA )	RX CTLE
						70 💌	No_Peak_+10.7d8
						Polarity (P/N reversal)	RX Termination (ohms)
						Inverted 💌	100
							RX P/N Board Connection
							JOC_COUPLED
							RX Loss of Signal Detector - Low
							3
							RX Loss of Signal Detector - High
							4
						Depart Import	Design Defaults Acoly
Phy Reset		L	Start	Stop		Export Allanes Import Allanes	Design Detauts ACDY

The exported content will be in the form of two set\_io commands, one for the TXP port and one for the RXP port of the selected lane instance.



#### **Export All Lanes**

Clicking the **Export All Lanes** button exports the current selected parameter options and other physical information for all lane instances in the design to an external PDC file. A popup box prompts you to choose the location where you want the .pdc file to be exported.

ansceiver Hierarchy	Physical Location	4		Pattern	0.1	Signal Integrity: CM1_PCIe_SS_0/XCVR_2_3_0/PF_3	CVR_OLANE1
CMI_POle_SS_0		-	CM1_PCIe_SS_0/XCVR_2_3_0/PF_XCVR_0/LANE0		•	LANE1_TXD_P/N TX Emphasis Amplitude	LANE 1_ROD_P/N RX Insertion Loss
E PF_XC	Q0_LANE2 Q0_LANE3		CM1_PCIe_SS_0XXCVR_2,3_0/PF_XCVR_0/LANE1	PRIIS7		200mV_wtb6.0d8       TX Impedance ( ohms )       500       TX Transmit Common Mode Adjustment (% of VDDA)	The transceiver data rate is set to 5000Mbps for this p The current settings will configure this port in CDR mod
						70   Polarity (P/N reversal)	[No_Peak_+10.7d8
						Diverted v	300
							RX P/N Board Connection DC_COUPLED
							RX Loss of Signal Detector - Low
							RX Loss of Signal Detector - High
Yry Reset			Start	510	<u>ا</u>	Export Import Export Al Lanes	Design DefaultsApply

### Import

Clicking the **Import** button imports Signal Integrity parameter options and other physical information for the selected lane from an external PDC file.

ceiver Hierarchy	Physical Location	4		Pattern	E 10	Signal Integritys CM1_PCIe_S5_0/XCIR_2_3_0/PF_3X	VR_0/LANE1	
CMI_PCle_SS	)		- CM1_PCIe_SS_0/XCVR_2_3_0/PF_XCVR_0/LANE0	PR#57	-	LANE 1_TXD_P/N TX Emphasis Amplitude	LANE 1_ROD_P/N RX Insertion Loss	
B Ø XCVR 2.3) B Ø P#_XC.						200mV_with6.0d8	17.0d0	-
E LA	Q0_LANE2 Q0_LANE3		CM1_PCIe_SS_0/XCVR_2_3_0/PF_XCVR_0/LANE1	PR#57	-	TX Impedance ( ohms )	The transceiver data rate is set to \$000	
e a	QD_LANES					100 -	The current settings will configure this po	ort in CDR mor
						TX Transmit Common Mode Adjustment (% of VDDA )	RX CTLE	
						70 💌	No_Peak_+10.7d8	
						Polarity (P/N reversal)	RX Termination (ohms)	
						Driverted _	100	
							RX P/N Board Connection	
							DC_COUPLED	
							RX Loss of Signal Detector - Low	
							3	
							RX Loss of Signal Detector - High	
							4	
Reset		L	Start		200	Export Import Export Al Lanes	Design Defaults	Apply

The Signal Integrity parameter options are applied to the device and updated in Modified Constraints.



#### **Import All Lanes**

Clicking the Import All Lanes button imports Signal Integrity parameter options and other physical information for all lanes from an external PDC file.

ansoeiver Hierarchy	Physical Location	] .		Pattern	e I	n 11	Signal Integrity: CM1_PCIe_SS_0/XCVR_2_3_0/PF_XC	VR_0/LANE1
CM1_PCIe_SS_0			CM1_PCIe_S5_0/XCVR_2_3_0/PF_XCVR_0/LANE0	PRES7		-	LANE 1_TXD_P/N TX Emphasis Amplitude	LANE 1_RXD_P/N RX Insertion Loss
B D XCVR.2.30 B D PF_JC	Q0_LANE2 Q0_LANE3		CM1_PCIe_SS_0/XCVR_2_3_0/PF_XCVR_0/LANE1	PR857	-		200mV_with6.0d8  TX Impedance ( ohms )	17.0dB The transceiver data rate is set to 5000Mbps for th
Elw.	Children .				-		TX Transmit Common Mode Adjustment (% of VDDA )	The current settings will configure this port in CDR is RX CTLE
							70   Polarity (P/N reversal)	No_Peak_+10.7d8 RX Termination (ohms)
							Driverted	200
								RX P/N Board Connection
								RX Loss of Signal Detector - Low
								3 RX Loss of Signal Detector - High
								4
Phy Reset		l	•	5	• ***		Export Import Export Al Lanes	Design Defaults Apply

The Signal Integrity parameter options are applied to the device and updated in Modified Constraints.

#### **Demo Mode**

Signal Integrity in Demo Mode lets users experience and understand the debug activities that can be performed with the Signal Integrity feature in SmartDebug. All debug activities except Apply are available in Demo Mode. Fixed data is used to populate the eye plot to allow you to experience the feature.

ansoeiver Hierarchy	Physical Location	4		Pattern	EQ-NearEnd	TX PLL RM	F D		PF_XCVR_1/LANED			
PF_XCVR_1	Q3_LANE0		PF_XCVR_1/LANEO	PR#57 •	Enable			LANEO_TXD_P/N TX Emphasis Amp			LANED_FOID_P/N FOI Insertion Loss	
								400mi, unity, -3 Ti Empediance (- 1156 TX Transmitt Cam 50 Folianity (PArtre- Normal	ohms ) mon Mode Adjustment (% of	* * * *	25.080         「25.080           126.120         「26.000           The transmission defaults in set to 30         The current settings will configure the RR CPLE           19/14         RR CPLE           100         RR CPLE           100         RR CPLE           RAC ROUND WITH LET CAP           RR Lises of Signal Detector - Low           Off	
ry Reset			4	J		Stop	J.	Export	Import		Design Defaults	Apply



### **Optimize DFE Coefficients**

The Optimize DFE Coefficients function allows you to optimize the DFE coefficients for the selected lanes. The optimize function runs through an algorithm for each lane and programs the best available coefficients for each selected lane for the current temperature, voltage, and data pattern conditions. After the optimization is complete, the transceiver lanes are programmed with these coefficients for the user to continue debugging.

**Note:** Optimize DFE Coefficients is only supported for data rates >= 5Gbps.

For information about how to use the optimized coefficients without SmartDebug, see the PolarFire Transceiver User Guide.

sceiver Hierarchy	Physical Loc	•1[		Þ	Signal Integrity: SmartBERT_L4_0(SmartB	BERT IP) AAN	E1	
SmartBERT_L4_0(SmartBERT LANE0 LANE1	Q3_LANE0				Q3_LANE1_TXD_P/N TX Emphasis Amplitude		Q3_LANE1_RXD_P/N RX Insertion Loss	
LANE2 LANE3	Q3_LANE2 Q3_LANE3				400mV_with3.5dB TX Impedance ( ohms )	•	17.0d8 The transceiver data rate is set to 10000Mbps	s for this p
					150 TX Transmit Common Mode Adjustment (	•	The current settings will configure this port in	OFE mode
					50	(SECTION)	N/A	-
					Polarity (P/N reversal)	•	RX Termination (ohms)	
							RX P/N Board Connection	
							AC_COUPLED_WITH_EXT_CAP RX Loss of Signal Detector - Low	
							Off	
							RX Loss of Signal Detector - High	
Reset			Plot Eye	Export	Export Import Export Al Lanes Import Al Lanes		Optimize DFE Design Defaults	Apply

Figure 35 · Debug TRANSCEIVER - Optimize DFE

Click **Optimize DFE** to open the Optimize DFE dialog. This dialog shows the lanes that are configured in the design. See the following example.



Optimize	e DFE	?	×
Select lanes	to Optimize DFE		
Quad 3	▼ Lane 0 ▼ Lane 1  Lane 2	Lane 3	
		_	
	Optimize DFE on Selected Lanes		
Help	1	Clos	e
	_		

Figure 36 · Optimize DFE Dialog

Select the lanes on which to run Optimize DFE and click Optimize DFE on Selected Lanes.

A Tcl script runs on the selected lanes to optimize the DFE coefficients.

To use these optimized coefficients directly in the STAPL file from Libero, you need to use a generated file. After the TCL script completes, an override file (<Design\_Name>\_SD\_DFE.txt) is generated. This file contains all the registers that were updated with the new coefficients.

The override file has the format <Reg. Address> <Reg. Value> and is saved in integrated flow in the designer folder. For standalone SmartDebug, the file is saved in the directory where the contents of ddc were extracted to when the standalone project was created.

In Libero, select the Configure Design Initialization Data and Memories tool, select the generated override file as the override file, and generate a new STAPL file.

**Note:** Additional design work must be done to use the DFE feature. See the PolarFire Transceiver User Guide for more information.

## Debug uPROM

You can debug clients configured in a design and debug  $\mu$ PROM memory address information with the Debug  $\mu$ PROM feature.

In the main SmartDebug window, click Debug uPROM.



⊗ SmartDebug <u>F</u> ile <u>V</u> iew <u>H</u> elp		
Device: MPF300TS_ES (MPF300TS_ES)	Programmer: S201YQST1V (S201YQST1V)	Ψ
ID code read from device: 2F8131CF		
View Device Status	Debug FPGA Array	
Debug UPROM	Debug TRANSCEIVER	7
Log		ē ×
Log I Messages 😵 Errors 🗼 Warnings 🌒 Info		8 ×
Log Thessages Strors A Warnings () Info		& ×
		8 ×

Figure 37 · SmartDebug Window - Debug uPROM

If a µPROM memory block is used in the Libero design, the µPROM Debug dialog box appears.

<b>)</b> (	JPROM Debug				? <mark>×</mark>
UP	ROM Debug				
1	Jser Design View 🔪	Direct Address View			
	Client List	Start Address		Number of words	
1	client_test	0	2000		
2	client2	200	500		
	ad from Device				
	Help			[	Close

Figure 38  $\cdot$   $\mu PROM$  Debug Dialog Box

### **User Design View**

The User Design View tab in the  $\mu$ PROM Debug dialog box lists all clients configured in the design. Selecting a client in the list enables the **Read from Device** button.

Clicking the **Read from Device** button displays a table showing the data in the location at the selected client address. See the following example.



_	Client L		Direct Add		T T			N	mber o	f words							
1 use		-	0.0	uuress	52224			m	inder o	i words							
-																	
lead fr	om Devic	e															
_	0	1	2	3	4	5	6	7	8	9	A	в	с	D	E	F	
0000	000	001	002	003	004	005	006	007	008	009	00A	008	00C	000	OOE	OOF	L
010	010	011	012	013	014	015	016	017	018	019	01A	01B	01C	01D	01E	01F	
020	020	021	022	023	024	025	026	027	028	029	02A	02B	02C	02D	02E	02F	
030	030	031	032	033	034	035	036	037	038	039	03A	038	03C	03D	03E	03F	
040	040	041	042	043	044	045	046	047	048	049	04A	048	04C	04D	04E	04	

The Client address is associated with *Start Address* and *Number of 9-bit words*. Therefore, the table will contain as many locations as the number of 9-bit words.

In the example above, Number of 9-bit words is 52224, so 52224 words will be shown in the table.

Column headers are numbered 0 to F in hexadecimal format, representing 16 words in a row.

Row addresses begin with a word address associated with *Start Address*. For example, if the *Start Address* is 0x15 (hex), the starting row has an address of 0x0010.

	Client		Direct A	t Addre						umber							-
				( Addre					N	umber	or word	15					
1 che	ent_test		0		20	000										_	
2 clie	ent2		200		50	0											
Dand 6	rom Dev	ice															
cead m	om Dev	ice				_											
	0	1	2	3	4	5	6	7	8	9	A	В	С	D	E	F	-
0200	008	00A	00C	00E	010	012	014	016	018	01A	062	064	066	074	074	074	[
0210	07A	07A	07A	080	080	080	086	086	086	08C	08C	08C	092	092	092	098	
0220	098	098	09E	09E	09E	0A4	0A4	0A4	0AA	0AA	0AA	0B0	0B0	0B0	0B6	0B6	
0230	086	OBC	0BC	0BC	0C2	0C2	0C2	0C8	0C8	0C8	OCE	0CE	OCE	0D4	0D4	0D4	
0240	0DA	0DA	0DA	0E0	0E0	0E0	0E6	0E6	0E6	0EC	0EC	0EC	0F2	0F2	0F2	0F8	
0250	0E8	0E8	OFF	OFF	OFF	104	104	104	004	004	004	010	010	010	016	016	

You can hover over a cell to see its address and value, as shown in the following example.



user De	esign View	V Dir	ect Acdre	ss View	1											
0	lient List		Start Ad	dress				Num	ber of w	vords						
useri		0.			52224											
-																
ead fr	om Devic	_	2	2		r	6	7	ô	0		P	6		5	E
ead fr	om Devic	2e	2	3	4	5	6	7	8	9	A	В	с	D	E	F
ead fr	0	_	2 NR	3 003	4	5 005	6 006	7 007	8 008	9 009	A 00A	B 00B	с 00С	D 00D	E 00E	F 00F

### **Direct Address View**

The Direct Address View tab in the  $\mu$ PROM Debug dialog box provides access to  $\mu$ PROM memory. You can read a part of a client or more than one client by specifying the *Start Address* and *Number of 9-bit words*.

Start Address - hexadecimal value (0 -9, A-F, upper/lower case)

Values are validated and errors are indicated by a red "STOP" icon (). The error message displays when you hover over the icon.

Number of 9-bit words - positive integer value

Values are validated and errors are indicated by a red "STOP" icon (<sup>100</sup>). The error message displays when you hover over the icon.

Read from Device - Disabled until valid values are entered in the fields.

Invalid or blank values are indicated by a red "STOP" icon (<sup>()</sup>). The error message displays when you hover over the icon.

**Note:** If the word falls within the 16 words that are placed in a row, the start location and the end location are highlighted in the row to show the starting point of the data. All preceding locations show 'NR' (Not Read). See the following example.



	Debug Design	View V	Direct	Address	View											
	Start address: 0x 5 Number of 9-bit words: 20					Decin	nal									
Read f	rom Dev	vice 1	2	3	4	5	6	7	8	9	A	в	с	D	E	F
Read f			2 NR	3 NR	4 NR	5	6 006	7 007	8 008	9 009	A 00A	B 00B	C 00C	D 00D	E 00E	F 00F

#### Notes:

When one field is entered, both fields are validated to enable the Read from Device button.

If fields change after enabling Read from Device, values are validated again and Read from Device may be disabled if invalid values are entered.

If the  $\mu$ PROM Debug dialog box is closed and reopened, the session is retained. The  $\mu$ PROM Debug session is lost only if the main SmartDebug window is closed.



# **SmartDebug Tcl Commands**

## SmartDebug Tcl Support

The following table lists the Tcl commands related to SmartDebug forPolarFire. Click the command to view more information.

Table 1 · SmartDebug 1	Tcl Commands
------------------------	--------------

Command	Action
	Probe
add_probe_insertion_point	Adds probe points to be connected to user-specified I/Os for probe insertion flow.
add to probe group	Adds the specified probe points to the specified probe group
<u>create probe group</u>	Creates a new probe group.
delete_active_probe	Deletes either all or the selected active probes.
load_active_probe_list	Loads the list of probes from the file.
move to probe group	Moves the specified probe points to the specified probe group.
program_probe_insertion	Runs the probe insertion flow on the selected nets.
remove probe insertion point	Deletes an added probe from the probe insertion UI.
set live probe	Set Live probe channels A and/or B to the specified probe point (or points).
select active probe	Manages the current selection of active probe points to be used by active probe READ operations.
read active probe	Reads active probe values from the device.
remove_from_probe_group	Move out the specified probe points from the group.
save_active_probe_list	Saves the list of active probes to a file.
select active probe	Manages the current selection of active probe points to be used by active probe READ operations.
<u>ungroup</u>	Disassociates the probes as group.
unset live probe	Discontinues the debug function and clears live probe channels.
write_active_probe	Sets the target probe point on the device to the specified value.
	LSRAM
read Isram	Reads a specified block of large SRAM from the device.
write_Isram	Writes a seven bit word into the specified large SRAM location.



Command	Action				
	Probe				
	uSRAM				
read_usram	Reads a uSRAM block from the device.				
write_usram	Writes a seven bit word into the specified uSRAM location.				
Transceiver					
loopback_mode	Applies loopback to a specified lane.				
smartbert_test	Starts and stops a Smart BERT test and resets error counter.				
static_pattern_transmit	Starts and stops a Static Pattern Transmit.				
	Additional Commands				
get_programmer_info	Lists the IDs of all FlashPRO programmers connected to the machine.				
uprom read memory	Reads uPROM memory block from the device.				

## add\_probe\_insertion\_point

This Tcl command adds probe points to be connected to user-specified I/Os for probe insertion flow.

add\_probe\_insertion\_point -net net\_name -driver driver -pin package\_pin\_name -port port name

### **Arguments**

-net net\_name
Name of the net used for probe insertion.
-driver driver
Driver of the net.
-pin package\_pin\_name
Package pin name (i.e. I/O to which the net will be routed during probe insertion).
-port port\_name
User-specified name for the probe insertion point.

### Example

add\_probe\_insertion\_point -net {count\_c[0]} -driver {Counter\_8bit\_0\_count\_out[0]:Q} pin {H5} -port {Probe\_Insert0}

## add\_to\_probe\_group

Tcl command; adds the specified probe points to the specified probe group.

add\_to\_probe\_group -name probe\_name -group group\_name



### Arguments

-name probe\_name Specifies one or more probes to add. -group group\_name Specifies name of the probe group.

#### Example

## create\_probe\_group

Tcl command; creates a new probe group.

create\_probe\_group -name group\_name

#### **Arguments**

-name group\_name

Specifies the name of the new probe group.

#### Example

create\_probe\_group -name my\_new\_grp

## delete\_active\_probe

Tcl command; deletes either all or the selected active probes. **Note**: You cannot delete an individual probe from the Probe Bus.

delete\_active\_probe -all | -name probe\_name

### Arguments

-all Deletes all active probe names. -name probe\_name Deletes the selected probe names.

#### **Example**

```
delete -all <- deletes all active probe names
delete -name out[5]:out[5]:Q \
        -name my_grpl.out[1]:out[1]:Q #deletes the selected probe names
delete -name my_grpl \
        -name my_bus #deletes the group, bus and their members.</pre>
```

## get\_programmer\_info

This Tcl command lists the IDs of all FlashPRO programmers connected to the machine.

get\_programmer\_info



This command takes no arguments.

#### **Example**

set a [get\_programmer\_info]

## load\_active\_probe\_list

Tcl command; loads the list of probes from the file.

load\_active\_probe\_list -file file\_path

### Arguments

-file *file\_path* The input file location.

#### **Example**

load\_active\_probe\_list -file "./my\_probes.txt"

## loopback\_mode

This Tcl command applies loopback to a specified lane.

loopback\_mode -lane {Physical\_Location} -apply -type {loopback\_type}

#### Arguments

-lane {Physical\_Location}
Specify the physical location of the lane.
-apply
Apply specified loopback to specified lane.
-type {loopback\_type}
Specify the loopback type to apply.

### **Examples**

```
loopback_mode -lane {Q3_LANE2} -apply -type {EQ-NearEnd}
loopback_mode -lane {Q3_LANE0} -apply -type {EQ-FarEnd}
loopback_mode -lane {Q0_LANE0} -apply -type {CDRFarEnd}
loopback_mode -lane {Q0_LANE1} -apply -type {NoLpbk}
loopback_mode -lane {Q1_LANE2} -apply -type {EQ-FarEnd}
loopback_mode -lane {Q1_LANE0} -apply -type {NoLpbk}
loopback_mode -lane {Q2_LANE2} -apply -type {EQ-NearEnd}
loopback_mode -lane {Q2_LANE3} -apply -type {CDRFarEnd}
```

### move\_to\_probe\_group

Tcl command; moves the specified probe points to the specified probe group. **Note**: Probe points related to a bus cannot be moved to another group.

move\_to\_probe\_group -name probe\_name -group group\_name



#### **Arguments**

-name probe\_name Specifies one or more probes to move. -group group\_name Specifies name of the probe group.

#### Example

## program\_probe\_insertion

This Tcl command runs the probe insertion flow on the selected nets.

program\_probe\_insertion

This command takes no arguments.

## read\_active\_probe

Tcl command; reads active probe values from the device. The target probe points are selected by the <u>select active probe</u> command.

read\_active\_probe [-deviceName device\_name] [-name probe\_name] [-group\_name bus\_name|group\_name][value\_type b|h][-file file\_path]

### Arguments

-deviceName device\_name

Parameter is optional if only one device is available in the current configuration.

-name probe\_name

Instead of all probes, read only the probes specified. The probe name should be prefixed with bus or group name if the probe is in the bus or group.

-group\_name bus\_name | group\_name

Instead of all probes, reads only the specified buses or groups specified here.

-value\_type b | h

Optional parameter, used when the read value is stored into a variable as a string.

b = binary

h = hex

-file file\_path

Optional. If specified, redirects output with probe point values read from the device to the specified file.

**Note**: When the user tries to read at least one signal from the bus/group, the complete bus or group is read. The user is presented with the latest value for all the signals in the bus/group.

#### Example

```
read_active_probe -group_name {bus1}
read_active_probe -group_name {group1}
To save into variable:
set a [read_active_probe -group_name {bus_name} -value_type h] #save read data in
hex string
```



If read values are stored into a variable without specifying value\_type parameter, it saves values as a binary string by default.

#### Example

set a [read\_active\_probe ] #sets variable a as binary string of read values after read\_active\_probe command.

## read\_lsram

Tcl command; reads a specified block of large SRAM from the device.

### **Physical block**

read\_lsram -name block\_name -fileName file\_name

#### Arguments

-name block\_name

Specifies the name for the target block.

-fileName file\_name

Optional; specifies the output file name for the data read from the device.

#### Exceptions

- Array must be programmed and active
- Security locks may disable this function

#### Example

Reads the LSRAM Block Fabric\_Logic\_0/U2/F\_0\_F0\_U1/ramtmp\_ramtmp\_0\_0/INST\_RAM1K20\_IP from the PolarFire device and writes it to the file output.txt.

read\_lsram -name {Fabric\_Logic\_0/U2/F\_0\_F0\_U1/ramtmp\_ramtmp\_0\_0/INST\_RAM1K20\_IP} fileName {output.txt}

### Logical block

read\_lsram -logicalBlockName block\_name -port port\_name

#### Arguments

-logicalBlockName block\_name

Specifies the name for the user defined memory block.

-port port\_name

Specifies the port for the memory block selected. Can be either Port A or Port B.

### Example

read\_lsram -logicalBlockName {Fabric\_Logic\_0/U2/F\_0\_F0\_U1} -port {Port A}



### read\_snvm\_memory

This Tcl command reads pages present in sector 1 of sNVM memory. 221 pages can be accessed through SmartDebug tool. This command can also read encrypted and authenticated pages.

read\_snvm\_memory [-deviceName device\_name] [-client client\_name] -startpage integer\_value endpage integer\_value [-fileName snvm\_data\_file\_name] -uskKey usk\_key

#### **Arguments**

-deviceName device\_name

Specifies the device to which a sNVM read is requested. This parameter is optional.

-client *client\_name* 

Specifies the client name that was configured in the design. This parameter is optional. This will be recorded if client view is requested.

-startpage integer\_value

Specifies the page value that is the beginning of the range of pages to be read.

-endpage integer\_value

Specifies the page value that is the end of the pages to be read.

-fileName snvm\_data\_file\_name

Specifies the file name to which the output will be redirected. This parameter is optional.

-uskKey usk\_key

Specifies the user secret key for each page. This field can have multiple keys that will be equal to the number of pages requested. The key must be 24 hexadecimal characters or 0. Multiple keys can be entered by separating them with a colon (:).

If the page is plaintext, the USK value should be set to 0.

### **Examples**

#### Single page read from client:

read\_snvm\_memory -deviceName {MPF300TS\_ES} -client {client\_auth\_PT} -startpage {1} -uskKey {0123456789ABCDEF01234567} –fileName {output.log}

#### Multiple pages read from client:

read\_snvm\_memory -deviceName {MPF300TS\_ES} -client {client\_PlainText} -startpage {0} -endpage {1} - uskKey {0:0} -fileName {output.log}

#### Page range read without client:

read\_snvm\_memory -deviceName {MPF300TS\_ES} -startpage {0} -endpage {4} -uskKey {0:0:0:0: 0123456789ABCDEF01234567} -fileName {output.log}

### read\_usram

Tcl command; reads a uSRAM block from the device.

### **Physical block**

read\_usram [-name block\_name] -fileName file\_name

#### Arguments

-name block\_name

Specifies the name for the target block.

-fileName file\_name

Optional; specifies the output file name for the data read from the device.



### Exceptions

- Array must be programmed and active
- Security locks may disable this function

#### Example

Reads the uSRAM Block Fabric\_Logic\_0/U3/F\_0\_F0\_U1/ramtmp\_ramtmp\_0\_0/INST\_RAM64x12\_IP from the PolarFire device and writes it to the file sram\_block\_output.txt.

read\_usram -name {Fabric\_Logic\_0/U3/F\_0\_F0\_U1/ramtmp\_ramtmp\_0\_0/INST\_RAM64x12\_IP} fileName {output.txt}

### Logical block

read\_usram -logicalBlockName block\_name -port port\_name

#### Arguments

-logicalBlockName *block\_name* 

Specifies the name of the user defined memory block.

-port port\_name

Specifies the port of the memory block selected. Can be either Port A or Port B.

#### Example

read\_usram -logicalBlockName {Fabric\_Logic\_0/U3/F\_0\_F0\_U1} -port {Port A}

## remove\_from\_probe\_group

Tcl command; removes the specified probe points from the group. That is, the removed probe points won't be associated with any probe group.

Note: Probes cannot be removed from the bus.

```
remove_from_probe_group -name probe_name
```

### **Arguments**

-name probe\_name

Specifies one or more probe points to remove from the probe group.

### **Example**

The following command removes two probes from my\_grp2.

## remove\_probe\_insertion\_point

This Tcl command deletes an added probe from the probe insertion UI.

remove\_probe\_insertion\_point -net net\_name -driver driver



### Arguments

#### -net net\_name

Name of the existing net which is added using the add\_probe\_insertion\_point command. -driver *driver* Driver of the net.

#### **Example**

```
remove_probe_insertion_point -net {count_out_c[0]} -driver
{Counter_8bit_0_count_out[0]:Q}
```

### save\_active\_probe\_list

Tcl command; saves the list of active probes to a file.

```
save_active_probe_list -file file_path
```

#### Arguments

-file file\_path

The output file location.

#### Example

save\_active\_probe\_list -file "./my\_probes.txt"

## select\_active\_probe

Tcl command; manages the current selection of active probe points to be used by active probe READ operations. This command extends or replaces your current selection with the probe points found using the search pattern.

select\_active\_probe [-deviceName device\_name] [-name probe\_name\_pattern] [-reset true/false]

### Arguments

-deviceName device\_name

Parameter is optional if only one device is available in the current configuration..

-name probe\_name\_pattern

Specifies the name of the probe. Optionally, search pattern string can specify one or multiple probe points. The pattern search characters "\*" and "?" also can be specified to filter out the probe names.

#### -reset true | false

Optional parameter; resets all previously selected probe points. If name is not specified, empties out current selection.

### Example

The following command selects three probes. In the below example, "grp1" is a group and "out" is a bus..



## set\_live\_probe

Tcl command; set\_live\_probe channels A and/or B to the specified probe point(s). At least one probe point must be specified. Only exact probe name is allowed (i.e. no search pattern that may return multiple points).

set\_live\_probe [-deviceName device\_name] [-probeA probe\_name] [-probeB probe\_name]

### **Arguments**

-deviceName device\_name

Parameter is optional if only one device is available in the current configuration or set for debug .

-probeA probe\_ name

Specifies target probe point for the probe channel A.

-probeB probe\_ name

Specifies target probe point for the probe channel B.

### **Exceptions**

- The array must be programmed and active
- Active probe read or write operation will affect current settings of Live probe since they use same probe circuitry inside the device
- Setting only one Live probe channel affects the other one, so if both channels need to be set, they must be set from the same call to set\_live\_probe
- Security locks may disable this function
- In order to be available for Live probe, ProbeA and ProbeB I/O's must be reserved for Live probe respectively

#### **Example**

Sets the Live probe channel A to the probe point A12 on device MPF300TS\_ES. set\_live\_probe [-deviceName MPF300TS\_ES] [-probeA A12]

## smartbert\_test

This Tcl command is used for the following:

- Start a Smart BERT test
- Stop a Smart BERT test
- Reset error count

#### smartbert\_test -start

This Tcl command starts a Smart BERT test with a specified pattern on a specified lane.

smartbert\_test -start -pattern {pattern\_type} -lane {Physical\_Location}

#### Arguments

-start
Start the Smart BERT test.
pattern {pattern\_type}
Specify the pattern type of the Smart BERT test.
-lane{Physical\_Location}
Specify the physical location of the lane.
-EQ-NearEndLoopback



Enable EQ-Near End Loopback on specified lane.

#### Examples

```
smartbert_test -start -pattern {prbs9} -lane {Q0_LANE3}
smartbert_test -start -pattern {prbs23} -lane {Q3_LANE2}
smartbert_test -start -pattern {prbs7} -lane {Q3_LANE1}
smartbert_test -start -pattern {prbs31} -lane {Q1_LANE2} -EQ-NearEndLoopback
smartbert_test -start -pattern {prbs9} -lane {Q2_LANE2} -EQ-NearEndLoopback
smartbert_test -start -pattern {prbs15} -lane {Q2_LANE3} -EQ-NearEndLoopback
```

#### smartbert\_test -stop

This Tcl command stops a Smart BERT test on a specified lane.

smartbert\_test -stop -lane {Physical\_Location}

#### Arguments

-stop Stop the smart BERT test. -lane {*Physical\_Location*} Specify the physical location of the lane.

#### **Examples**

```
smartbert_test -stop -lane {Q0_LANE0}
smartbert_test -stop -lane {Q0_LANE3}
smartbert_test -stop -lane {Q3_LANE2}
smartbert_test -stop -lane {Q3_LANE1}
smartbert_test -stop -lane {Q1_LANE2}
smartbert_test -stop -lane {Q2_LANE2}
smartbert_test -stop -lane {Q2_LANE3}
```

#### smartbert\_test -reset\_counter

This Tcl command resets a lane error counter.

smartbert\_test -reset\_counter -lane {Physical\_Location}

#### Arguments

-reset\_counter

Reset lane error counter on hardware and cumulative error count on the UI. -lane {Physical\_Location} Specify the physical location of the lane.

#### Examples

```
smartbert_test -reset_counter -lane {Q0_LANE0}
smartbert_test -reset_counter -lane {Q3_LANE2}
smartbert_test -reset_counter -lane {Q2_LANE3}
smartbert_test -reset_counter -lane {Q2_LANE2}
```



```
smartbert_test -reset_counter -lane {Q1_LANE2}
smartbert_test -reset_counter -lane {Q3_LANE1}
```

## static\_pattern\_transmit

This Tcl command starts and stops a Static Pattern Transmit.

#### static\_pattern\_transmit -start

```
static_pattern_transmit -start -lane {Physical_Location} -pattern {pattern_type} -value
{user_pattern_value}
```

### **Parameters**

-start
Start the Static Pattern Transmit.
-lane {Physical\_Location}
Specify physical location of lane.
-pattern {pattern\_type}
Specify pattern\_type of Static Pattern Transmit.
-value {user\_pattern\_value
Specify user\_pattern\_value in hex if pattern\_type selected is custom.

#### Examples

```
static_pattern_transmit -start -lane {Q0_LANE0} -pattern {fixed}
static_pattern_transmit -start -lane {Q0_LANE2} -pattern {maxrunlength} -value {}
static_pattern_transmit -start -lane {Q3_LANE2} -pattern {custom} -value {df}
static_pattern_transmit -start -lane {Q3_LANE0} -pattern {fixed} -value {}
static_pattern_transmit -start -lane {Q1_LANE1} -pattern {custom} -value {4578}
static_pattern_transmit -start -lane {Q1_LANE2} -pattern {fixed} -value {}
static_pattern_transmit -start -lane {Q1_LANE2} -pattern {fixed} -value {}
static_pattern_transmit -start -lane {Q2_LANE2} -pattern {fixed} -value {}
static_pattern_transmit -start -lane {Q2_LANE2} -pattern {maxrunlength} -value {}
static_pattern_transmit -start -lane {Q2_LANE2} -pattern {maxrunlength} -value {}
static_pattern_transmit -start -lane {Q2_LANE1} -pattern {custom} -value {abcdef56}
```

#### static\_pattern\_transmit -stop

static\_pattern\_transmit -stop -lane {Physical\_Location}

#### **Parameters**

-stop Stop the Static Pattern Transmit. -lane {Physical\_Location} Specify physical location of lane.

#### Examples

```
static_pattern_transmit -stop -lane {Q0_LANE0}
static_pattern_transmit -stop -lane {Q0_LANE2}
static_pattern_transmit -stop -lane {Q3_LANE2}
static_pattern_transmit -stop -lane {Q3_LANE0}
```



static_pattern_transmit	-stop	-lane	$\{Q1\_LANE1\}$
static_pattern_transmit	-stop	-lane	$\{Q1\_LANE2\}$
static_pattern_transmit	-stop	-lane	$\{Q2\_LANE2\}$
static_pattern_transmit	-stop	-lane	$\{Q2\_LANE1\}$

### ungroup

Tcl command; disassociates the probes as a group.

nngroup -name group\_name

### Arguments

-name group\_name Name of the group.

#### **Example**

ungroup -name my\_grp4

## unset\_live\_probe

Tcl command; discontinues the debug function and clears live probe A, live probe B, or both probes (Channel A/Channel B). An all zeros value is shown in the oscilloscope.

unset\_live\_probe -probeA 1 -probeB 1 [-deviceName device\_name]

### **Arguments**

-probeA Live probe Channel A. -probeB

Live probe Channel B.

-deviceName device\_name

Parameter is optional if only one device is available in the current configuration or set for debug (see the SmartDebug User's Guide for details).

### **Exceptions**

- The array must be programmed and active.
- Active probe read or write operation affects current of Live Probe settings, because they use the same probe circuitry inside the device.
- Security locks may disable this function.

#### Example

The following example unsets live probe Channel A from the device MPF300TS\_ES. unset\_live\_probe -probeA 1[-deviceName MPF300TS\_ES]

### uprom\_read\_memory

This Tcl command reads a uPROM memory block from the device.

read\_uprom\_memory -startAddress {hex\_value} -words {integer\_value}



#### Arguments

-startAddress *hex\_value* Specifies the start address of the uPROM memory block. -words *integer\_value* Specifies the number of 9-bit words.

#### Example

read\_uprom\_memory -startAddress {0xA} -words {100}

## write\_active\_probe

Tcl command; sets the target probe point on the device to the specified value. The target probe point name must be specified.

write\_active\_probe [-deviceName device\_name] -name probe\_name -value true/false
-group\_name group\_bus\_name -group\_value "hex-value" | "binary-value"

### Arguments

-deviceName device\_name

Parameter is optional if only one device is available in the current configuration. -name probe\_name Specifies the name for the target probe point. Cannot be a search pattern. -value true | false hex-value | binary-value Specifies values to be written. True = High False = Low -group\_name group\_bus\_name Specify the group or bus name to write to complete group or bus. -group\_value "hex-value" | "binary-value" Specify the value for the complete group or bus.

Hex-value format : " <size>'h<value>"

Binary-value format: " <size>'b<value>"

### Example

### write\_lsram

Tcl command; writes a word into the specified large SRAM location.



### **Physical block**

write\_lsram -name block\_name] -offset offset\_value -value integer\_value

#### Arguments

-name block\_name

Specifies the name for the target block.

-offset offset\_value

Offset (address) of the target word within the memory block.

-value integer\_value

Word to be written to the target location. Depending on the configuration of memory blocks, the width can be 1, 2, 5, 10, or 20 bits.

#### Exceptions

- Array must be programmed and active
- The maximum value that can be written depends on the configuration of memory blocks
- Security locks may disable this function

#### Example

```
write_lsram -name {Fabric_Logic_0/U2/F_0_F0_U1/ramtmp_ramtmp_0_0/INST_RAM1K20_IP} -offset
0 -value 291
```

#### Logical block

```
write_lsram -logicalBlockName block_name -port port_name -offset 1 offset_value -logicalValue
hexadecimal_value
```

#### Arguments

-logicalBlockName block\_name

Specifies the name of the user defined memory block.

-port *port\_name* 

Specifies the port of the memory block selected. Can be either Port A or Port B.

-offset offset\_value

Offset (address) of the target word within the memory block.

-logicalValue hexadecimal\_value

Specifies the hexadecimal value to be written to the memory block. Size of the value is equal to the width of the output port selected.

#### Example

```
write_lsram -logicalBlockName {Fabric_Logic_0/U2/F_0_F0_U1} -port {Port A} -offset 1 -
logicalValue {00FFF}
```

## write\_usram

Tcl command; writes a 12-bit word into the specified uSRAM location.



### **Physical block**

write\_usram -name block\_name] -offset offset\_value -value integer\_value

#### Arguments

-name block\_name

Specifies the name for the target block.

-offset offset\_value

Offset (address) of the target word within the memory block.

-value integer\_value

12-bit value to be written.

#### Exceptions

- Array must be programmed and active
- The maximum value that can be written is 0x1FF
- Security locks may disable this function

#### Example

Writes a value of 0x291 to the device PolarFire in the Fabric\_Logic\_0/U3/F\_0\_F0\_U1/ramtmp\_ramtmp\_0\_0/INST\_RAM64x12\_IP with an offset of 0. write\_lsram -name {Fabric\_Logic\_0/U3/F\_0\_F0\_U1/ramtmp\_ramtmp\_0\_0/INST\_RAM64x12\_IP} offset 0 -value 291

### Logical block

write\_usram -logicalBlockName block\_name -port port\_name -offset offset\_value -logicalValue
hexadecimal\_value

### Arguments

-logicalBlockName block\_name

Specifies the name of the user defined memory block.

```
-port port_name
```

Specifies the port of the memory block selected. Can be either Port A or Port B.

-offset offset\_value

Offset (address) of the target word within the memory block.

-logicalValue hexadecimal\_value

Specifies the hexadecimal value to be written to the memory block. Size of the value is equal to the width of the output port selected.

#### Example

```
write_usram -logicalBlockName {Fabric_Logic_0/U3/F_0_F0_U1} -port {Port A} -offset 1 -logicalValue {00FFF}
```



# **Frequently Asked Questions**

## How do I monitor a static or pseudo-static signal?

To monitor a static or pseudo-static signal:

- 1. Add the signal to the Active Probes tab.
- 2. Select the signal in the Active Probes tab, right-click, and choose Poll....

Live Probes Active Probes Memor	ry Blocks Probe	Insertion	
+ - ★ ★ Save	Load	Delete	Delete All
Name	Туре	Read Value	Write Value
Shift_Reg_0/shft_reg[13:0]	DFF	14h0001	14h
D FF 0/a 0:D FF 0/a:O	DFF	0	-
		Read	
		Delete	
		Delete	
		Poll	
		Create Group	
			27
Read Active Probes Sa	ve Active Probes' Da	Write Act	ive Probes
		Contraction of the second seco	Contraction and the second second
	Image: Name     Save       Name     > Shift_Reg_0(shft_reg[13:0]       > JF_0/q_0:0_JF_0/q;Q	★         Save         Load           Name         Type           >         Shift_Reg_0/shft_reg[13:0]         DFF           D_FF_0/q_0:D_FF_0/q:Q         DFF	Image: Save     Load     Delete       Name     Type     Read Value       > Skift_Reg_0/shft_reg[13:0]     DFF     14h0001       D FF_0/q_0:D_FF_0/q;Q     DFF     0       Read     Delete     Poll       Create Group

3. In the Pseudo-static Signal Polling dialog box, choose a value in Polling Setup and click Start Polling.

Pseudo-static signal polling		? <mark>×</mark>
ignal : D_FF_0/q_0:D_FF_0/q:Q		
Polling Setup		
Poll for 0	Poll for 1	
	once per second. It should be used for pseudo-static signals to pseudo-static signal polling, click the Help button.	hat do not change frequently.

For more information, refer to the SmartDebug for PolarFire User Guide.

## How do I force a signal to a new value?

To force a signal to a new value:

- 1. In the SmartDebug window, click Debug FPGA Array.
- 2. Click the Active Probes tab.



3. Select the signal from the selection panel and add it to Active Probes tab.

	₽×	FPGA Array debug data			
ierarchical View Netlist View		Live Probes Active Probes 1	Memory Blocks Probe I	nsertion	
lter:	Search	+ - + + Save	Load	Delete	elete All
et(s):	Add	Name	Туре	Read Value Write V	alue
lame	Type ^				
<ul> <li>B_DOUT_1_c[6:0]</li> <li>B_DOUT_2_c[7:0]</li> </ul>	RAM64x18 RAM64x18				
B DOUT c[5:0]	RAM64x18				
DFN1_0_Q:DFN1_0:Q Add	DFF				
DFN1_1_Z:DFN1_1:Q					
	0:0] RAM64x18				
count_6_0_q[5:0]	DFF				
count_6_0_q[5:0] count_6_2_0_q[7:0]	DFF DFF DFF				
<pre>&gt; count_6_0_q[5:0] &gt; count_6_2_0_q[7:0] &gt; count_7_0_q[6:0]</pre>	DFF				
<ul> <li>▷ IRAM_0/sd_URAM_0_RACO/8_ADDR_net[9</li> <li>▷ cont.(5,0,0[5:0])</li> <li>▷ cont.(5,2,0,q[7:0])</li> <li>▷ cont.7,2,0,q[6:0]</li> <li>▷ cont.7,2,0,q[8:0]</li> </ul>	DFF DFF	Read Active Probes	Save Active Probes' Da	ta) Write Active Probes	

- 1. Click Read Active Probe to read the value.
- 2. In the Write Value column, enter the value to write to the signal and then click Write Active Probes.

		₽×	FPGA An	ray debug da	ta				
Hierarchical View Netlist View			Live P	robes Ad	tive Probe	s Memor	y Blocks Probe I	nsertion	
Filter:	Search		+	- +	+	Save	Load	Delete	Delete All
Net(s):	Add		Nam	e			Туре	Read Value	Write Value
westal.	Add		0	FN1_0_Q:D	N1_0:Q		DFF	1	0 -
Name	Туре	*	ÞB	DOUT_c[5:	0]		RAM64x18	6'h0E	6'h9
<ul> <li>B_DOUT_(5:0)</li> <li>DFN1_0_Q:DFN1_0:Q</li> <li>DFN1_1:Q</li> <li>DFN1_1:Q</li> <li>URAM_0Vsd_URAM_QURAM_R0C0/A_ADDR_net(9:0)</li> <li>URAM_0Vsd_URAM_0_URAM_R0C0/B_ADDR_net(9:0)</li> <li>count_6_0_q(5:0)</li> <li>count_6_2_0_q(7:0)</li> <li>count_6_2_0_q(6:0)</li> </ul>	RAM64x18 DFF DFF DFF	•							
<pre>&gt; count_7_0_q[8:0] &lt;</pre>	1.22.23						ve Active Probes' Da		

## How do I perform simple SmartBERT tests?

You can perform SmartBERT tests using the Debug Transceiver option in SmartDebug.

To perform a SmartBERT test, in the SmartBERT page of the Debug Transceiver dialog box, select to run a PRBS test on-die or off-die with EQ-NEAREND checked or unchecked. For more information, see "SmartBERT" on page 43.

To perform a SmartBERT test, in the Smart BERT page of the Debug Transceiver dialog box, select your options and click **Start** to run a Smart BERT test on-die or off-die with EQ-NEAREND checked or unchecked. For more information, see "SmartBERT" on page 43.



## How do I read LSRAM or USRAM content?

To read RAM content:

- 1. In the Debug FPGA Array dialog box, click the **Memory Blocks** tab.
- 2. Select the memory block to be read from the selection panel on the left of the window.

lemory Blocks Selection	8 ×	FPGA Array debug data
Filter:	Search	Live Probes Active Probes Memory Blocks Probe Insertion
Memory Blocks:	Select	User Design Memory Block:
		Data Width:
Instance Tree	-	Port Used:
4 🎩 Fabric_Logic_0		
▲ 1 U2 ▲ 1 F_0_F0_U1	E	
4 12 ramth A	dd	
4 Primuves		
	RAM1K18_IP	
F_10_F1_U2		
F_11_F1_U2 F_12_F1_U2		
▶ I F_13_F1_U2		
F_14_F1_U2		
F_15_F1_U2		
F_16_F1_U2		
▷ IL F_17_F1_U2 ▷ IL F 18 F1 U2		
↓ ↓ F_10_F1_02		Read Block Save Block Data Write Block
▶ 1 F 1 F1 U2	-	

An "L" in the icon next to the block name indicates that it is a logical block, and a "P" in the icon indicates that it is a physical block. A logical block displays three fields in the Memory Blocks tab: User Design Memory Blocks, Data Width, and Port Used. A physical block displays two fields in the Memory Blocks tab: User Design Memory Block and Data Width.

- 3. Add the block in one of the following ways:
  - Click Select.
  - Right-click and choose Add.
  - Drag the block to the Memory Blocks tab.
- 4. Click **Read Block** to read the content of the block.

ory Blocks Selection 🗗 🗙		FPGA Arra	ay debug	g data														
Filter: Searc	h	Live Pr	obes	Active	Probes	Me	mory Bl	ocks	Probe	Insertio	n							
Vemory Blocks: Select	t	User De Data W	esign Me idth:	mory Bl	ock:	Fabric_ 18-bit	Logic_0	/U3/F_0	_F0_U1									
Instance Tree	^	Port Us	ed:			Port A	5	-										
Fabric_Logic_0								_										
4 1 U3 4 1 F_0_F0_U1	-		0	1	2	3	4	5	6	7	8	9	Α	В	С	D	E	F
Terramtmp_ramtmp_0_0     Terramtmp_ramtmp_0_0		0000	00A83	08809	09008	14500	00010	00381	12028	00040	12080	04000	20214	02000	11080	20040	1C220	0A020
INST_RAM64x18_IP INST_RAM64x18_IP		0010	02700	04451	04001	08000	05000	32500	00120	00000	00080	00420	04019	1C800	00052	00106	00C22	10058
<ul> <li>Image: a contract of the second second</li></ul>		0020	10400	00010	10000	14044	1C040	0810E	39425	0D990	10C14	00004	04001	10000	00100	00042	20100	08002
INST_RAM64x18_IP INST_RAM64x18_IP		0030	000 1B	00000	20808	0008A	00 1E0	28100	02883	00770	10020	04000	00000	00200	20004	22400	04006	0A090
<ul> <li>Image: a state of the state of</li></ul>																		
<ul> <li>INST_RAM64x18_IP</li> <li>F_12_F1_U2</li> </ul>							Read B	lock	Save	Block D	ata	Wri	te Block					
Image: A state of the state	-							_	Sec.4				_					





SmartDebug for PolarFire User Guide

## How do I change the content of LSRAM or USRAM?

To change the content of LSRAM or USRAM:

- 1. In the SmartDebug window, click **Debug FPGA Array**.
- 2. Click the Memory Blocks tab.
- 3. Select the memory block from the selection panel on the left of the window.

lemory Blocks Selection	₽×	FPGA Array debug data
Filter:	Search	Live Probes Active Probes Memory Blocks Probe Insertion
Memory Blocks:	Select	User Design Memory Block:
		Data Width:
Instance Tree	*	Port Used:
Fabric_Logic_0		
4 🎩 U2	E	
▲ ■ F_0_F0_U1 ▲ ■ ramtn Ad		
a IP: Promoves		
D INST_R	AM1K18_IP	
▷ I F_10_F1_U2		
F_11_F1_U2		
F_12_F1_U2		
F_13_F1_U2 F_14_F1_U2		
▶ ₩ F_15_F1_U2		
▷ : F_16_F1_U2		
▷ 🎩 F_17_F1_U2		
F_18_F1_U2		Read Block Save Block Data Write Block
F_19_F1_U2		
F_1_F1_U2	-	

An "L" in the icon next to the block name indicates that it is a logical block, and a "P" in the icon indicates that it is a physical block. A logical block displays three fields in the Memory Blocks tab: User Design Memory Blocks, Data Width, and Port Used. A physical block displays two fields in the Memory Blocks tab: User Design Memory Block and Data Width.

- 4. Add the memory block in one of the following ways:
  - Click Select.
  - Right-click and choose Add.
  - Drag the block to the Memory Blocks tab.
- 5. Click Read Block. The memory content matrix is displayed.
- 6. Select the memory cell value that you want to change and update the value.
- 7. Click **Write Block** to write to the device.



lemory Blocks Selection	5 ×	FPGA Arr	ay debu	g data														
Filter:	Search	Live Pr	obes	Active	Probes	Men	nory Blo	dks	Probe In	nsertion								
Memory Blocks: Select		User Design Memory Block: Fabric_Logic_0/U3/F_12_F1_U2 Data Width: 18-bit																
Instance Tree	*	Port Us	ed.			Port A		-										
4 🎩 Fabric_Logic_0																		
▷ 1 U2	E		0	1	2	3	4	5	6	7	8	9	A	В	с	D	E	F
✓ ■ U3 ▷ ■ F_0_F0_U1				-	-						-	-		-		-		
F 10 F1 U2		0000	00083	3FFFF	00102	00088	01200	00824	00004	00304	00200	00E00	0006A	20001	00060	00050	00300	00000
F_11_F1_U2																		
4 1 F_12_F1_U2		0010	00000	20410	20002	02101	00080	08016	020C0	0C200	000A0	00002	08000	10020	05004	00018	20008	08300
Image: A start and the star		0020	00200	00000	00000	00084	00080	02408	00001	02080	20000	00000	20000	00005	02000	02012	00C01	00454
4 😰 Primi																		
INST_RAM64x18_IP		0030	02400	10001	00001	04000	00400	00002	01201	00004	00020	01C00	02040	10008	07242	18102	24041	02044
F_13_F1_U2 F_14_F1_U2																		
F_14_F1_02 F_15_F1_02																		
▷ I F 16 F1 U2						_								_				
F_17_F1_U2							Read B	lock	Save	Block Da	ta	Writ	e Block					
F 18 F1 U2																		
> I F 19 F1 112																		

#### See Also

"Memory Blocks " on page 28 SmartDebug for PolarFire User Guide

## How do I read the health check of the Transceiver?

You can read the transceiver health check using the following Debug Transceiver options:

 Review the Configuration Report, which returns Tx PMA Ready, Rx PMA Ready, TxPLL status, and RxPLL status. For the transceiver to function correctly, all four should be green. The Configuration Report can be found in the Debug TRANSCEIVER dialog box under Configuration Report. See "Debug Transceiver" on page 39.

Lanes       PF_JCRR_0       PF_JCRR_1       PF_JCRR_2       PF_JCRR_0         Projecial Location       C0_LANB0       Q2_LANB3       Q2_LANB3       Q2_LANB3       Q2_LANB3         TX PUL       Data Width       40 bit       8 bit       0 bit       0 bit         * LANE1       Projecial Location       C0_LANB1       Q2_LANB2       Q2_LANB3         TX PUL       Data Width       40 bit       8 bit       0 bit       0 bit         * LANE1       Projecial Location       C0_LANB1       Q2_LANB2       Q2_LANB3         TX PUL       Data Width       40 bit       8 bit       0 bit       0 bit         V LANE2       VLANB2       Q2_LANB3       0 bit       0 bit         TX PUL       Data Width       40 bit       8 bit       0 bit       0 bit         V LANE2       Projecial Location       C0_LANB2       Q2_LANB3       0 bit         TX PUL       Data Width       40 bit       8 bit       0 bit       0 bit         V LANE2       Projecial Location       C0_LANB2       Q2_LANB3       0 bit         TX PUL       Data Width       40 bit       8 bit       0 bit       0 bit         VANB       Data Width       40 bit       8 bit<	onfiguration Report Smart	BERT Loop	back Modes	Static Patter	Transmit Eye Monitor	
Tri MA Ready         S <t< th=""><th>anes</th><th>PF_XCVR_0</th><th>PF_XCVR_1</th><th>PF_XCVR_2</th><th>¥_XCVR_3</th><th></th></t<>	anes	PF_XCVR_0	PF_XCVR_1	PF_XCVR_2	¥_XCVR_3	
R. P. DA Ready       Image: Control of the state of the	Physical Location	Q0_LANE0	Q3_LANES	Q1_LANES	22_LANE2	
TypeIL         C <td>Tx PMA Ready</td> <td>•</td> <td>•</td> <td>•</td> <td></td> <td></td>	Tx PMA Ready	•	•	•		
RX PLL         O         O         O           Data Width         40 bit         8 bit         0 bit         0 bit           Data Width         40 bit         8 bit         0 bit         0 bit           Data Width         0 LANEI         0 LANEI         0 LANEI         0 LANEI           Priprical Location         0 LANEI         0 LANEI         0 LANEI         0 LANEI           TA Macky         0         0         0         0         0           TA Macky         0         0         0         0         0           TA Width         0 bit         0 bit         0         0         0           Data Width         0 bit         0 bit         0         0         0           Physical Location         0 LANEI         0 LANEI         0 LANEI         0 LANEI         0 LANEI           TA MA Resky         0         0         0         0         0         0           TA Width         0 bit         0 bit         0         0         0         0           TA Physical Location         0 LANEI           TA Physical Location <td>Rx PMA Ready</td> <td>•</td> <td>•</td> <td>•</td> <td></td> <td></td>	Rx PMA Ready	•	•	•		
Data         Width         Q bat         Q bat         Q bat           Data         Mode         Q LANE         Q LANE         Q LANE           Projectione         Q LANE         Q LANE <t< td=""><td></td><td>•</td><td>•</td><td>•</td><td></td><td></td></t<>		•	•	•		
VMRI         Pojicial Location         QU LANEL         Q2 LANE2         QL LANEL         Q2 LANE3           Tr PMA Ready         C         C         C         C         C           TX PML         C         C         C         C         C           Physical Location         QU LANE2         QL LANE3         C         C         C           TX PML         C         C         C         C         C         C           TX PML         C         C         C         C         C         C         C           TX PML         C						
Physical Location         00, LANEI         02, LANEI         04, LANEI	Data Width	40 bit	8 bit	0 bit	bit	
Physical Location         00 LANEI         02 LANEI         00 H           TX PUL         0 <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>						
Tr. VMA Ready         Solution						
Rv PMA Resdy         V         V         V           TX PLL         V         V         V           RV PL         V         V         V           Data Width         40 bit         8 bit         0 bit         0 bit            LAME2         Physical Location         Q0_LANE2         Q3_LANE3         Q2_LANE3           RV PMA Resdy         V         V         V         V         V           RV PMA Resdy         V         V         V         V           Data Width         V         V         V         V         V           Physical Location         Q1_LANE3         Q1_LANE3         Q1_LANE3         Q1_LANE3         V           RV PMA Resdy         V         V         V         V         V         V           RV PMA Resdy         V         V         V         V         V         V			Q3_LANE2	Q1_LANE1	Q2_LANE0	
TXPLL         SXPLL         SXPLL <th< td=""><td>Tx PMA Ready</td><td></td><td>•</td><td>•</td><td></td><td></td></th<>	Tx PMA Ready		•	•		
KV PLL         Out         Out           Physical Location         Q2, LANE2         Q3, LANE2         Q3, LANE3         Q3, LANE3         Q4, LANE3 <td></td> <td></td> <td></td> <td>•</td> <td></td> <td></td>				•		
Data Width         40 bit         8 bit         0 bit         0 bit           z Materia         Z						
4 LARE2         Op_LARE2         Op_LARE2         Op_LARE2         Op_LARE2         Op_LARE2         Op_LARE3						
Physical Location         Q0 LANE2         Q1 LANE2         Q2 LANE3           Tir YMA Ready              Ra PMA Ready              TX YUL              Data Width         40 bit         8 bit         0 bit           Physical Location         Q0 LANE3         Q1 LANE3           Physical Location         Q0 LANE3         Q1 LANE3           Physical Location         Q0 LANE3         Q1 LANE3           TX FMA Ready             TX FMA Ready             TX FMA Ready             TX FUL             Physical Location         Q0 LANE3         Q1 LANE3           TX FMA Ready             TX FUL             TX FUL             TX FUL             TX FUL             Data Width         40 bit         8 bit         0 bit	Data Width	40 bit	8 bit	0 bit	) bit	
Tir PMA Ready         Image: Control of the second of	A LANE2					
Rv PMA Ready         Image: Control of the second seco	Physical Location	Q0_LANE2	Q3_LANE1	Q1_LANE2	Q2_LANE3	
Rv PMA Ready         Image: Control of the second seco	Tx PMA Ready	•	•	•		
KV PLL         Outs Width         Obit         Obit           Jata Width         40 bit         8 bit         0 bit         0 bit	Rx PMA Ready	•	•	•		
Data Width     40 bit     8 bit     0 bit       # LNE=       Physical location     00 LANS     03 LANS     02 LANS       To FMA Ready     0     0     0	TX PLL			•		
LANE3     Physical Location Q0_LANE3 Q3_LANE0 Q1_LANE0 Q2_LANE1     Tr PMA Ready     For PMA Ready     TX PLL     TX PLL     TX PLL     Data Width     40 bit     8 bit     0 bit     0 bit	RX PLL	•	•	•		
Physical Location 00 LANE3 03 LANE0 01 LANE0 02 LANE1 Tr PMA Ready Kr PMA Ready TX PUL Data Width 40 bit 8 bit 0 bit 0 bit	Data Width	40 bit	8 bit	0 bit	) bit	
Physical Location 00 LANE3 03 LANE0 01 LANE0 02 LANE1 Tx PMA Ready 0 0 0 0 LANE 0 0 LANE0 02 LANE1 Tx PMA Ready 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	A LANES					
Tr (PMA Ready Control of the second of the s		O0 LANES	O3 LANEO	O1 LANEO	22 LANEL	
R: PMA Ready F F F F F F F F F F F F F F F F F F F			•	•		
TX PLL C C C C C C C C C C C C C C C C C C						
RX PLL Obit 8 bit 0 bit 0 bit						
Data Width 40 bit 8 bit 0 bit 0 bit						
Refer	Data Width	40 bit	8 bit	0 bit		
Reffes						Defeath
						Refresh

 Run the SmartBERT Test, with EQ-NEAR END checked or with external loopback connection from Tx to Rx on selected lanes. This should result in 0 errors in the Cumulative Error Count column. See "SmartBERT" on page 43.