

Altera FPGA Dynamic Probe

Online Help



Notices

© Agilent Technologies, Inc. 2001-2009

No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Agilent Technologies, Inc. as governed by United States and international copyright laws.

Trademarks

Microsoft®, MS-DOS®, Windows®, Windows 2000®, and Windows XP® are U.S. registered trademarks of Microsoft Corporation.

Adobe®, Acrobat®, and the Acrobat Logo® are trademarks of Adobe Systems Incorporated.

Manual Part Number

Version 03.83.0000

Edition

October 12, 2009

Available in electronic format only

Agilent Technologies, Inc. 1900 Garden of the Gods Road Colorado Springs, CO 80907 USA

Warranty

The material contained in this document is provided "as is," and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this manual and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

Restricted Rights Legend

If software is for use in the performance of a U.S. Government prime contract or subcontract, Software is delivered and licensed as "Commercial computer software" as defined in DFAR 252.227-7014 (June 1995), or as a "commercial item" as defined in FAR 2.101(a) or as "Restricted computer software" as defined in FAR 52.227-19 (June 1987) or any equivalent agency regulation or contract clause. Use, duplication or disclosure of Software is subject to Agilent Technologies' standard commercial license terms, and non-DOD Departments and Agencies of the U.S. Government will receive no greater than Restricted Rights as defined in FAR 52.227-19(c)(1-2) (June 1987). U.S. Government users will receive no greater than Limited Rights as defined in FAR 52.227-14 (June 1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

Safety Notices

CAUTION

A **CAUTION** notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a **CAUTION** notice until the indicated conditions are fully understood and met.

WARNING

A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.

Using the Altera FPGA Dynamic Probe

When the Altera LAI (Logic Analyzer Interface) core has been inserted into an FPGA, the FPGA dynamic probe lets a logic analyzer capture data on signals internal to the FPGA.

The FPGA dynamic probe lets you change probe points without recompiling or affecting the timing of the design, and it lets you import internal signal names from your FPGA design tool.

- Overview (see page 9)
- Probing FPGA Debug Pins (see page 11)
- Installing and Licensing the FPGA Dynamic Probe for Altera FPGAs (see page 13)
- FPGA Design Steps (see page 15)
- Preparation Steps (see page 17)
 - Step 1. Install the JTAG cable programmer software (see page 18)
 - Step 2. Set up the JTAG cable using the programmer software (see page 19)
- Measurement Steps (see page 27)
 - Step 3. Establish connection between analyzer and JTAG cable (see page 28)
 - Step 4. Download configuration bits into FPGA (see page 30)
 - Step 5. Import signal names (see page 31)
 - Step 6. Map FPGA pins (see page 35)
 - Step 7. Make the measurement (see page 39)
- Troubleshooting (see page 41)
- Concepts (see page 47)
- Reference (see page 59)
 - FPGA Dynamic Probe Setup Dialog (see page 60)
 - FPGA Dynamic Probe Bank Selection Dialog (see page 67)
 - Specifications and Characteristics (see page 69)
- **See Also** Probe Control, COM Automation (see page 71)
 - Probe Setup, XML Format (see page 73)

Contents

Using the Altera FPGA Dynamic Probe 3

- **1** Overview
- 2 Probing FPGA Debug Pins
- **3** Installing and Licensing the FPGA Dynamic Probe
- 4 FPGA Design Steps

5 Preparation Steps

Step 1. Install the JTAG cable programmer software 18

Step 2. Set up the JTAG cable using the programmer software 19
To install the USB Blaster device driver 20
To set up a JTAG Server 21

6 Measurement Steps

Step 3. Establish connection between analyzer and JTAG cable 28

Step 4. Download configuration bits into FPGA 30

Step 5. Import signal names31To trim imported bus/signal names32To rename imported bus/signal names32To define additional FPGA bus/signal names

Step 6. Map FPGA pins 35

Step 7. Make the measurement 39

7 FPGA Dynamic Probe Troubleshooting

When I click the 'Cable Connection' button I get 'problem locating Altera software' dialog 42

33

When I click the 'Cable Connection' button I get 'script error' dialog 43

If you don't see activity in the logic analyzer 44

If state mode measurements don't work 45

8 FPGA Dynamic Probe Concepts

Example: Inserting an LAI Core 48 Creating a Logic Analyzer Interface 48 Configuring the Logic Analyzer Interface Core Parameters 49 Mapping the Clock Signal 50 Mapping the Logic Analyzer Interface File Pins to Available I/O Pins 51 Mapping Internal Signals to the Logic Analyzer Interface Banks 53 Saving the Logic Analyzer Interface 55 Enabling/Disabling the Logic Analyzer Interface 55 Compiling the Quartus II Project 56 FYI: Creating Multiple Logic Analyzer Interface Instances in One FPGA 57

Automated Logic Analyzer Set Up 58

9 FPGA Dynamic Probe Reference

FPGA Dynamic Probe Setup Dialog 60 **Cable Connection Dialog** 61 Pin Mapping Dialog 61 Pin Mapping Edit Dialog 62 Select FPGA Configuration File Dialog 64 Select Signal Import File Dialog 64 **Properties Dialog** 65 Core Details Dialog 65 FPGA Dynamic Probe Bank Selection Dialog 67 Trim Bus/Signal Names Dialog 67 **Specifications and Characteristics** 69

10 Probe Control, COM Automation

11 Probe Setup, XML Format

<Assignment> Element 74

<ATC_II> Element 75

<Bank> Element 76

<Banks> Element 80

<Core> Element 81

<Cores> Element 83

<DefinedProbes> Element 84

<Device> Element 85

<Devices> Element 87 <JTAG_Chain> Element 88 <Label> Element 89 <Label> Element (under NonATCLabels) 90 <Labels> Element 91 <NonATCLabels> Element 92 93 <PinMapping> Element <Pod> Element 95 <Pods> Element 96 97 <Probe> Element <Properties> Element 98 <Signal> Element (under Bank) 100 <Signal> Element (under Probe) 101 <Signals> Element (under Bank) 102 <Signals> Element (under Probe) 104 <SymbolInfo> Element 105 <TriggerInfo> Element 106 <WindowInfo> Element 107

Index



The FPGA dynamic probe lets you:

• View internal activity.

With a logic analyzer, you are normally limited to measuring signals at the periphery of the FPGA. With the FPGA dynamic probe, you can now access signals internal to the FPGA. You can measure multiple internal signals for each external pin dedicated to debug, unlocking visibility into your design that you never had before.

• Make multiple measurements in seconds.

Moving probe points internal to an FPGA used to be time consuming. Now, in less than a second you can easily measure a different set of internal signals – without design changes – and FPGA timing stays constant when you select new sets of internal signals for probing.

• Leverage the work you did in your design environment.

The FPGA dynamic probe is the industry's first tool that maps internal signal names from your FPGA design tool to your logic analyzer. Eliminate unintentional mistakes and save hours of time with this automatic setup of signal and bus names on your logic analyzer.



Create a time-saving FPGA measurement system. Insert an Altera LAI (Logic Analyzer Interface) core into your FPGA design. With the application running on your logic analyzer via JTAG, you control which group of internal signals to measure.



1 Overview



The Altera LAI core provides a multiplexer for selecting the signals that are output to FPGA debug pins. Core parameters determine the number of signals per bank and the number of banks.

Altera FPGA Dynamic Probe Online Help

2

Probing FPGA Debug Pins

The supported mechanisms for probing the FPGA debug pins with a logic analyzer are: soft touch (34-channel or 17 channel), Mictor, Samtec, and flying lead probes.

For more information on probing, see "Probing the Device Under Test" (in the online help).



2 Probing FPGA Debug Pins

Altera FPGA Dynamic Probe Online Help

3



Installing and Licensing the FPGA Dynamic Probe

Before you can use the Agilent B4656A FPGA dynamic probe for Altera FPGAs, you must install and license the software:

1 Install the FPGA dynamic probe software from the download web page at:

"http://www.agilent.com/find/la-sw-download"

2 Follow the instructions on your entitlement certificate to redeem and install the FPGA dynamic probe license.



Installing and Licensing the FPGA Dynamic Probe

Altera FPGA Dynamic Probe Online Help



Altera LAI cores are inserted into FPGAs using Altera's Quartus II design software. For an example, see:

• "Example: Inserting an LAI Core" on page 48

For more information, see Altera's web site at " www.altera.com".

Next • Preparation Steps (see page 17)



4 FPGA Design Steps



Before you can use the FPGA dynamic probe software with the *Agilent Logic Analyzer* application, you must take these preparation steps:

- 1 Install the JTAG cable programmer software (see page 18)
- 2 Set up the JTAG cable using the programmer software (see page 19)

Next • Measurement Steps (see page 27)



Step 1. Install the JTAG cable programmer software

Before you can use the FPGA dynamic probe for Altera FPGAs, you must download and install the free Altera Quartus II Programmer software on the same logic analysis system or PC that the *Agilent Logic Analyzer* application runs on.

NOTE The PC that has the JTAG cable connection to the device under test does not have to be the logic analysis system or PC that the *Agilent Logic Analyzer* application runs on. In this case, the logic analysis system communicates via LAN with the system that has the JTAG server (and cable connection to the DUT).

1 Download the Altera Quartus II Programmer software from:

"https://www.altera.com/support/software/download/programming/quart us2/dnl-quartus2_programmer.jsp"

- **2** Follow the installation instructions to install the software on the same logic analysis system or PC that the *Agilent Logic Analyzer* application runs on.
- **3** If the PC or logic analysis system that has the JTAG cable connection to the device under test is not the same computer that runs the *Agilent Logic Analyzer* application, and Quartus II design software is not already installed, install the Quartus II Programmer software there too.
- Next Step 2. Set up the JTAG cable using the programmer software (see page 19)

Step 2. Set up the JTAG cable using the programmer software

On the PC or logic analysis system that has the JTAG cable connection to the device under test:

- 1 If you are using the USB Blaster cable, driver files come with the Altera Quartus II Programmer software, but you must install them. See To install the USB Blaster device driver (see page 20).
- **2** Open the Altera Quartus II Programmer (or design) software and set up the JTAG cable connection:
 - a From the Windows Start menu, choose All Programs>Altera>Quartus II 7.2 Programmer and SignalTap II>Quartus II 7.2 Programmer.
 - b In the Quartus II Programmer window, click Hardware Setup....

₩ Q	🖗 Quartus II Programmer - [Chain1.cdf]						
Eile	⊑dit	Option	ns	Processing	Help		
ا 🔔	Hardwa	re Setup.		No Hardware			
Γ E	nable re	al-time IS	۶P	to allow backgro	und program	ming (for MAX II (devices)
	Start		Fi	le		Device	(
	Stop						

c In the Hardware Setup dialog, select the appropriate JTAG cable hardware.

Hardware Setup			
Hardware Settings JTAG Settin	gs		
Select a programming hardware hardware setup applies only to t	setup to use when p he current programme	rogramming device er window.	es. This programming
Currently selected hardware:	ByteBlaster [LPT1]	•
Available hardware items:	No Hardware ByteBlaster [LPT1]		
Hardware	USB-Blaster [USB-	0]	
ByteBlaster USB-Blaster	Local Local	LPT1 USB-0	Remove Hardware
			Close

You may have to click **Add Hardware...** to add the appropriate JTAG cable hardware.

- **d** After you have selected the appropriate JTAG cable hardware, click **Close** to close the Hardware Setup dialog.
- e Choose File>Exit to close the Quartus II Programmer software.

If the PC or logic analysis system that has the JTAG cable connection to the device under test is not the same computer that will run the *Agilent Logic Analyzer* application, you must set up a JTAG server. See "To set up a JTAG Server" on page 21.

• Step 3. Establish connection between analyzer and JTAG cable (see page 28)

To install the USB Blaster device driver

On the PC or logic analysis system that has the JTAG cable connection to the device under test:

- 1 Connect the USB Blaster device to your computer.
- 2 In the Found New Hardware Wizard dialog, select Install from a list or specific location; then, click Next >.



3 On the next page of the wizard, specify the **drivers\usb-blaster\x32** (for a 32-bit operating system) or **drivers\usb-blaster\x64** (for a 64-bit operating system) location underneath the Quartus II Programmer install directory; then, click **Next** >.



4 When the wizard has finished installing the driver, click Finish.

Found New Hardware Wizard				
	Completing the Found New Hardware Wizard The wizard has finished installing the software for: Altera USB-Blaster Click Finish to close the wizard.			
	< <u>B</u> ack Finish Cancel			

For more information on installing the USB Blaster driver, see "http://www.altera.com/support/software/drivers/usb-blaster/dri-usb-bla ster-xp.html" (or search for "USB Blaster driver" on the Altera web site).

To set up a JTAG Server

The Altera FPGA Dynamic Probe can connect to JTAG cables over a TCP/IP connection. This is useful when you want to remotely connect to a logic analysis system that also has the JTAG cable connection to the device under test. In this case, the *Agilent Logic Analyzer* application (and the Altera FPGA Dynamic Probe software), must communicate remotely with the logic analysis system as well as the JTAG cable server.

If your JTAG cable is connected between the device under test and the logic analysis system or PC that is running the *Agilent Logic Analyzer* application, you do not need to set up a JTAG server.

To set up the Altera Programmer software to allow client/server connections to a remote JTAG cable:

- 1 Make sure the Altera Quartus II Programmer software is installed on both:
 - The *client computer* where you will be running the *Agilent Logic Analyzer* application (the JTAG Client).
 - The *remote computer* (logic analysis system or PC) that is physically connected to the JTAG cable (the JTAG Server).

This should have already been done in "Step 1. Install the JTAG cable programmer software" on page 18.

- 2 Set up the JTAG Server on the *remote computer*:
 - a Start the Quartus II Programmer application from the Windows Start menu by choosing All Programs>Altera>Quartus II 7.2 Programmer and SignalTap II>Quartus II 7.2 Programmer.
 - **b** In the Quartus II Programmer window, click Hardware Setup....

🏷 Q	uartus	; II Pro	gra	ammer - [Cha	ain1.cdf]		
Eile	<u>E</u> dit	Optior	ns	Processing	Help		
ا 🔔 ا	Hardwa	re Setup.		No Hardware			
🗐 Ei	nable re	al-time IS	βPι	to allow backgro	und program	ming (for MAX II devi	ces)
	Start		Fi	le		Device	
	Stop						

c In the Hardware Setup dialog's JTAG Settings tab, click **Configure** Local JTAG Server....

Hardware Setup		X
Hardware Settings JTAG Specify JTAG servers to remote clients to access	Settings add and remove from the JTAG Serv	vers list. Specify the password used by
JTAG Servers: Server Local	Connection Status OK	Add Server Remove Server
		Configure Local JTAG Server

d In the Configure local JTAG Server dialog, check **Enable remote clients to connect to the local JTAG server**, enter a password, and confirm the password.

Configure Local JTAG Server 🛛 🛛				
I Enable remote clients to connect to the local JTAG server.				
Password:				
Confirm password:				
OK Cancel				

Record this password as you will need it later. Then, click OK.

- e Click Close to close the Hardware Setup dialog.
- f Choose File>Exit to close the Quartus II Programmer software.
- **3** If the *remote computer* is running firewall software, allow access to the JTAG Server through the firewall. If the remote computer is not running firewall software, you can skip this step and proceed to step 4.

For example, if the *remote computer* is running Windows Firewall:

- a Make sure you are logged in as an administrator.
- **b** From the Windows Start menu, choose **Control Panel>Windows Firewall**.
- c In the Windows Firewall dialog's General tab, uncheck **Don't** allow exceptions.



If the firewall is off, it may be that the computer is using different firewall software. In this case, you must enable access to the Quartus II Programmer port using the different firewall software.

- d In the Exceptions tab, click Add Port....
- e In the Add a Port dialog:
 - i Enter the Name: Quartus II Programmer.
 - ii Enter the Port number: 1309.
 - iii Select TCP.
 - iv Click OK to close the Add a Port dialog.

Add a Port	
Use these settings number and protoc want to use.	to open a port through Windows Firewall. To find the port ol, consult the documentation for the program or service you
<u>N</u> ame:	Quartus II Programmer
<u>P</u> ort number:	1309
	⊙ICP OUDP
What are the risks	of opening a port?
Change scope	OK Cancel

e Click OK to close the Windows Firewall dialog.

- 4 Get the hostname of the *remote computer* (JTAG Server):
 - a From the Windows Start menu, choose Control Panel>System.
 - **b** Click the **Computer Name** tab.
 - **c** Write down the "Full computer name". This is the hostname of your remote JTAG server which you will need shortly.
 - d Click OK.
- **5** Set up the JTAG Client on the *client computer*:
 - a Start the Quartus II Programmer application from the Windows Start menu by choosing All Programs>Altera>Quartus II 7.2 Programmer and SignalTap II>Quartus II 7.2 Programmer.
 - **b** In the Quartus II Programmer window, click Hardware Setup....
 - c In the Hardware Setup dialog's JTAG Settings tab, click Add Server....
 - **d** In the Add Server dialog:
 - i Enter the full hostname of the remote JTAG Server you found in step 4.c as the **Server name**.
 - ii Enter the password you created in step 2.d as the Server password.
 - iii Click OK.

Add Server		X
Server name:	mtx33	
Server password:	NNNNNN	
	OK Cancel	

d Verify that the Hardware Setup dialog shows a Connection Status of "OK". If not, then there is a problem with the networking. Re-verify your firewall settings and server-name/server-password settings.

Hardware Setup	Settings	X
Specify JTAG servers to a remote clients to access t	add and remove from the JTAG Servers list. Spe he local JTAG server by configuring your local J	cify the password used by TAG server.
JIAG Servers Server Local mtx33	Connection Status OK OK	Add Server
	Configu	re Local JTAG Server
		Close

- e Click Close to close the Hardware Setup dialog.
- f Choose File>Exit to close the Quartus II Programmer software.

Now, when you establish the connection between the analyzer and the JTAG cable (see "Step 3. Establish connection between analyzer and JTAG cable" on page 28), the remote JTAG cable will be listed in the Cable Connection dialog.

Altera FPGA Dynamic Probe Online Help

Measurement Steps

After you have completed the FPGA Design Steps (see page 15) and Preparation Steps (see page 17), you are ready to take these measurement steps in the *Agilent Logic Analyzer* application:

- 1 Establish connection between analyzer and JTAG cable (see page 28)
- 2 Download configuration bits into FPGA (see page 30)
- **3** Import signal names (see page 31)
- 4 Map FPGA pins (see page 35)

6

5 Make the measurement (see page 39)



Step 3. Establish connection between analyzer and JTAG cable

The FPGA dynamic probe application establishes a connection between the logic analyzer and a Xilinx cable. It also determines what devices are on the JTAG scan chain and lets you pick which one you wish to communicate.

To establish a connection between the logic analyzer and the Altera LAI core:

1 Add a new FPGA Dynamic Probe set by choosing **Setup>(Logic Analyzer Module)>New Probe>FPGA Dynamic Probe**.



Or, in the Overview window, from a module's drop-down menu, choose New Probe>FPGA Dynamic Probe.

Modules	Windows
Slot A	
My Logic	Listing-1
× ⊡ ∰ ™	Show
🖸 🗹 New Probe	Altera FPGA Dynamic Probe
New Tool	 General Purpose Probe Set
New Window	
Setup	► × Show
Disable	
Rename	

2 In the FPGA Dynamic Probe Setup dialog (see page 60), click Cable Connection....

Press "Cable Connection" to read JTAG chain.	Cable Connection
	Configure Device
	Pin <u>M</u> apping

3 In the Cable Connection dialog (see page 61), select the type of cable and, if necessary, specify any cable parameters; then, click **OK**.

Cable Connection	×
• Existing cables	
USB-Blaster [USB-0] ByteBlaster [LPT1]	
No Cable (Demo M	ode)
	OK Cancel

When the connection has been established, you will see the devices on the JTAG chain, and you can select the desired device.

On De	vice Core	Imported Signals	Pin Mapping	Details	Cable Connection
@1:	: E No Cores			@1: EP1C20 (0x020840D.	Configure Device
<				>	Properties

Next • Step 4. Download configuration bits into FPGA (see page 30)

Step 4. Download configuration bits into FPGA

To download configuration bits into an FPGA:

1 In the FPGA Dynamic Probe Setup dialog (see page 60), select the FPGA device to which you wish to download configuration bits; then, click **Configure Device...**.

On Device Core Imported Signals Pin Mapping Details Cable Conr @1: E No Cores @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device @1: Device @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device @1: Device @1: EP1C20 (0x0208400 @1: EP1C20 (0x0208400 @1: Device	Đ					Р	Probe Setu	Dynamic	PGA I
<	nection 20 (0 Device	Cable Conne @1: EP1C2 Configure De	20 (0x020840D	Details @1: EP1C2	Pin Mapping	Imported Signals	Core No Cores	Device @1: E	On
Comparison of the second se	ping	Pin Mappi							
	ties	<u>P</u> ropertie							<

2 In the Select FPGA Configuration File dialog (see page 64), select the FPGA configuration file; then, click **Open**.

Next • Step 5. Import signal names (see page 31)

Step 5. Import signal names

The FPGA dynamic probe can automatically set up bus/signal names in the logic analyzer by reading a .lai file produced by Altera's Quartus II LAI (Logic Analyzer Interface) design software.

To import bus/signal names:

1 In the FPGA Dynamic Probe Setup dialog (see page 60), select the device whose bus/signal names you want to import; then, click Import Bus/Signals....

PGA Dynamic Probe Setup				
On Device Core	Imported Signals	Pin Mapping	Details	Cable Connection
@1: E No Cores		(@1: EP1C20 (0x020840D	@1: EP1C20 (0
				Configure Device
				Import Bus/Signals
				Pin <u>M</u> apping
<				Properties
			<u> </u>	ncel <u>H</u> elp

- 2 In the Select Signal Import File dialog (see page 64), select the signal import file; then, click **Open**.
- 3 In the import results dialog, view the bus/signal import information; then, click **OK**.

Import Successful!
Import successful! Imported the following bus/signals from Z:\Documents and Settings\pw5670\My Documents\Agilent Technologies\Logic Analyzer\Config Files\Altera_LAT\\ai1.lai small_ic20:inst[count[0] small_ic20:inst[count[1] small_ic20:inst[count[3] small_ic20:inst[count[4] small_ic20:inst[count[6] small_ic20:inst[count[6] small_ic20:inst[count[6] small_ic20:inst[count[9]
ОК

4 In the FPGA Dynamic Probe dialog, note that the defined cores now appear.

6 Measurement Steps



- **See Also** To trim imported bus/signal names (see page 32)
 - To rename imported bus/signal names (see page 32)
 - To define additional FPGA bus/signal names (see page 33)
 - **Next** Step 6. Map FPGA pins (see page 35)

To trim imported bus/signal names

NOTE

Trimming bus/signal names after initial bank selection may require manual insertion of bus/signals in Waveform and Listing display windows. We recommend that you trim bus/signal names before changing the bank selection.

- 1 In the FPGA Dynamic Probe Bank Selection dialog (see page 67), click Trim Bus/Signal Names....
- 2 In the Trim Bus/Signal Names dialog (see page 67), specify the bus/signal name characters to trim; then, click **OK**.

To rename imported bus/signal names

- 1 Open the Buses/Signals setup tab (see "Defining Buses and Signals" (in the online help)).
- 2 In the "FPGA Probe" bus/signal name folder, rename the bus/signal (see "To rename a bus or signal" (in the online help)).
- 3 Reopen the FPGA Dynamic Probe dialog by choosing Setup>(Logic Analyzer Module)>(FPGA Dynamic Probe Name)>Bank Selection....



Or, in the Overview window, click the FGPA dynamic probe's **Properties** button; then, choose **Bank Selection...**.



Note also that you can triple-click signal names in the FPGA Dynamic Probe dialog to rename them (without having to do global trimming).

To define additional FPGA bus/signal names

- 1 Open the Buses/Signals setup tab (see "Defining Buses and Signals" (in the online help)).
- 2 In the "FPGA Probe" bus/signal name folder, add a new bus/signal (see "To add a new bus or signal" (in the online help)).
- **3** Assign channels to the new bus/signal name (see "To assign channels in the default bit order" (in the online help) or "To assign channels, selecting the bit order" (in the online help)).
- 4 Reopen the FPGA Dynamic Probe dialog by choosing Setup>(Logic Analyzer Module)>(FPGA Dynamic Probe Name)>Bank Selection....

6 Measurement Steps



Or, in the Overview window, click the FGPA dynamic probe's **Properties** button; then, choose **Bank Selection...**.



Whenever buses/signals are added to the "FPGA Probe" folder, they are associated with a specific bank. If you select another bank, the added buses/signals do not appear.

If you want to define buses/signals that apply to all banks, create them outside of the "FPGA Probe" folder. That way, the buses/signals are not associated with a bank.

Step 6. Map FPGA pins

Quickly specify how the FPGA pins (the signal outputs of Altera LAI) are connected to your logic analyzer. Select your probe type and rapidly provide the information needed for the logic analyzer to automatically track names of signals routed through the Altera LAI core.

To map FPGA pins to logic analyzer probes:

1 In the FPGA Dynamic Probe Setup dialog (see page 60), select the Altera LAI core whose output pins you want to map; then, click **Pin Mapping...**.

On Device Core Imported Signals Pin Mapping Details ✓ @1: E @1: EP1C20 (0x0208400) @1: EP1C20 (0x0208400) ✓ " auto_lai_0 Altera_LAINai1.lai Incomplete Timing, 4 banks, 8 pins Cable Connection	FPGA Dynamic	: Probe Setup				
auto_lai_0 Pin <u>M</u> apping	0n Device ✓ @1: E ✓ "	Core	Imported Signals Altera_LAI\lai1.lai	Pin Mapping	Details @1: EP1C20 (0x020840D Timing, 4 banks, 8 pins	Cable Connection)
Properties	<					Properties

2 In the Pin Mapping dialog (see page 61), click Add Probe....

Pin Mapping				×
Probes used to connect to	your FPGA		ſ	Add Probe
Reference Designator	Probe Type	Logic Analyzer Pod	(s)	Jau/Edit Drobo
				<u>D</u> elete Probe
<				
		<u>o</u> k	Cancel	<u>H</u> elp

3 In the "Select the type of probe to add" dialog, select the type of probe that is used to connect to your FPGA; then, click **OK**.



If your probe doesn't appear in the list, you can "download the latest probe definitions from the web" (in the online help).

4 In the Pin Mapping Edit dialog (see page 62), select the FPGA pins (you can select multiple pins using Shift-click or Ctrl-click) and drag them on to the pin/pad map.



After you've mapped FPGA pins to the probe, you can hover the mouse pointer over a pin description field to view a tool tip describing the FPGA debug pin name, the pod connection, the channel number, and the signal type (single-ended or differential).
E53464	A 34-ch Mictor	single-ended	probe	
Reference Desi	gnator J1			
	Pin 38 D0	Pin 37 D0	lai0]
	Pin 36 D1	Pin 35 D1	lai1	1
	Pin 34 D2	Pin 33 D2	lai2 Slot A P	od 2 Channel 0
	Pin 32 D3	Pin 31 D3	lai3 Single-f	Inded Signal
	Pin 30 D4	Pin 29 D4	lai4	1
	Pin 28 D5	Pin 27 D5	lai5	
	Pin 26 D6	Pin 25 D6	lai6	Probe
	Pin 24 D7	Pin 23 D7		
•	Pin 22 D8	Pin 21 D8		
	Pin 20 D9	Pin 19 D9		
	Pin 18 D10	Pin 17 D10		

You can clear all FPGA pins that have been mapped to pins/pads by clicking **Clear**. You can clear individual pin mappings by dragging a pin from the pin/pad diagram back to the FPGA Pins list.

5 Select the logic analyzer pods that the probe is connected to.

When you wish to map multiple cores using different halves of the same probe, make sure you select **None** for the half that will be used by the other core.

For state (synchronous) cores, you need to make sure that the ATCK pin maps to one of the "Clk" pin/pad locations (which identify clock signal inputs) and that the associated logic analyzer pod is valid for clock inputs. (In the *General State Mode*, the clock lines on the first 4 pods of a logic analyzer can be used as clock inputs; in the *Turbo State Mode*, the clock line on the first pod can be used as a clock input.)

6 When you are done mapping FPGA pins, click **OK**. Note that your probe has been added to the list in the Pin Mapping dialog.

Pin Mapping			
Probes used to connect to	o your FPGA		Add Probe
Reference Designator	Probe Type E5346A 34-ch Mictor single-ended probe	Logic Analyzer Pod(s) Slot A Pod 1(Odd), Slot A	View/Edit Probe
			<u>D</u> elete Probe
<		>	
		<u>OK</u> Cancel	Help

7 Click OK to close the Pin Mapping dialog.

NOTE

In the FPGA Dynamic Probe Setup dialog, notice pin mapping is no longer "Incomplete".

i	PGA I	Dynamic	Probe Setu	p			×
		Device @1: E "	Core auto_lai_0	Imported Signals Altera_LAI\lai1.lai	Pin Mapping J1	Details @1: EP1C20 (0x020840D Timing, 4 banks, 8 pins	Cable Connection @1: EP1C20 (0 Configure Device) [Import Bus/Signals] auto_lai_0 Pin Mapping
	<					>	Properties
						<u> </u>	incel <u>H</u> elp

8 Click OK to close the FPGA Dynamic Probe Setup dialog.

The FPGA Dynamic Probe dialog opens automatically. If you expand a bank, you see the imported bus/signal names.

FPGA Dynamic Probe Bank Selection		×
Summary auto_lai_0		<u>Run Eye Finder</u> Irim Bus/Signal Names Properties Selected signal bank: Bank 0
Bank 4	∨ IK	Last selected in core: None Cancel Help

• Step 7. Make the measurement (see page 39)

Step 7. Make the measurement

At this point, you are ready to use the logic analyzer (as you would normally) to capture activity on internal FPGA signals.

You can tell the Altera LAI core to switch signal banks without affecting the timing of your design. When viewing the Probes toolbar (**View>Toolbars>Probes**), click do open the FPGA Dynamic Probe dialog. Then, select the signal bank to be routed to the logic analyzer and click **OK**. You can change signal banks as often as needed to make measurements throughout your FPGA.

FPGA Dynamic Probe Bank Selection	
auto_lai_0 auto_l	Run Eye Finder Irim Bus/Signal Names Properties Selected signal bank: Bank 1 Last selected in core: Bank 0
	OK Cancel Help

You can correlate internal FPGA activity with external measurements. With each new selection of a signal bank, the application updates new signal names from your design to the logic analyzer. View internal FPGA activity and time correlate internal FPGA measurements with external events in the surrounding system.

🔆 Agilent Logic Analyzer - Unna	med Configuration - [Waveform-1]	
Eile Edit <u>V</u> iew <u>S</u> etup <u>T</u> ools	Markers Run/Stop Waveform Window Help - 6	F ×
D 🚅 🖬 🎒 🖊 🕻 🙀 📕	TNQQ N ■ 時米頭 > ◇ ■ 図■ サッツッ 時米頭 № ▼ 参	
M1 to M2 = 40.96 us		
Scolo 20 uc/diu 🗏 Later		-1
		_
Bus/Signal Simple Trigger	0 us -80 us -60 us -40 us -20 us 0 s 20 us 40 us 60 us 80 us 100 · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · · ·	
Time	-101.25 us 100.628 us	^
	67 17 57 37 77 0F 4F 2F 6F 1F 5F 3F 7F 00 40 20 60 10 50 30	
small X ×	1 0	
Small_` X ×] 0	
‡small' X ¥	1 0	
-[] small_' X ×	0 1 0	=
[] small 1 ×		
‡small' 0 ×		
🖞 small_' 1 ×		
small_ X ×		-
Overview	Listing-1 Waveform-1	
For Help, press F1	Status Local	<u>نور</u> ا

NOTE

Timing zoom is automatically disabled when using the FPGA dynamic probe. You can re-enable timing zoom; however, because of the Altera LAI core, timing zoom does not provide an accurate representation of internal FPGA signals.

NOTE	 Captured data is invalidated whenever you: Select a different bank. Select a different core. Download configuration bits into an FPGA. Reopen a cable connection. Imported signal names. Trim imported bus/signal names. Change the FPGA pin mapping.
See Also	 "Capturing Data from the Device Under Test" (in the online help) "Analyzing the Captured Data" (in the online help)

Altera FPGA Dynamic Probe Online Help

7



FPGA Dynamic Probe Troubleshooting

- When I click the 'Cable Connection' button I get 'problem locating Altera software' dialog (see page 42)
- When I click the 'Cable Connection' button I get 'script error' dialog (see page 43)
- If you don't see activity in the logic analyzer (see page 44)
- If state mode measurements don't work (see page 45)



When I click the 'Cable Connection' button I get 'problem locating Altera software' dialog

Altera FPGA Dynamic Probe	
There was a problem locating the Altera software required to use this prod Quartus-II development environment or QProgrammer (no license required installed on the system running the FPGA Probe application. If the required software is installed, please locate the file "quartus_stp.exe", which should the "bin" folder underneath the Quartus install location. Otherwise, press I for instructions on how to install the Altera software.	uct. Either the) must be d Altera I be located in the Help button
\bin\quartus_stp.exe	Browse
Cancel	Help

Possible cause: Default install path was changed during Quartus II installation. Actions:

- 1 Follow the instructions displayed on the dialog. If the file 'quartus_stp.exe' does not exist anywhere on the machine, continue with step 2.
- 2 Close the error and 'Cable Connection' dialogs.
- **3** Use Windows Add/Remove Programs to "Modify" the Quartus II installation, checking both the QProgrammer and SignalTap boxes before confirming. Reboot.

When I click the 'Cable Connection' button I get 'script error' dialog

NOTE

The blue light on the USB Blaster should blink when it is being identified.

Possible causes:

- There are no connected cables. Actions:
 - a Close the error and 'Cable Connection' dialogs.
 - **b** Re-seat the connection between the oscilloscope or PC and the Altera cable.
 - c Retry the 'Cable Connection' button.
- Cable driver installation is incomplete. Actions:
 - a Close the error and 'Cable Connection' dialogs.
 - b Verify: Windows Device Manager should show a USB controller named "Altera USB-Blaster". If this controller has a red X, it is disabled. Right-click and select 'Enable'. If it has a yellow exclamation mark, then the Altera driver has not been fully installed. There is probably a New Hardware Found dialog hidden beneath the display. Complete the instructions on that dialog. See To install the USB Blaster device driver (see page 20)
 - c Retry the 'Cable Connection' button.
- Selected cable not connected to DUT (device under test). Actions:
 - a Close the error and 'Cable Connection' dialogs.
 - **b** Re-seat connection between Altera cable and DUT (ensure plug on board is not reversed).
 - c Retry the 'Cable Connection' button.
- DUT is not powered up. Actions:
 - a Close the error dialog and 'Cable Connection' dialog.
 - **b** Power up DUT.
 - c Retry the 'Cable Connection' button.

7 FPGA Dynamic Probe Troubleshooting

If you don't see activity in the logic analyzer

If you get a dynamic status from the core that says everything is enabled and ready, but you see no activity on your logic analyzer, check your connections to and from your device under test to the logic analyzer pod cables.

If state mode measurements don't work

If you are unable to capture data in state mode, either look at the clock activity indicators and select the appropriate clock for state measurements, or make a timing measurement to determine which clock is the master clock.

7 FPGA Dynamic Probe Troubleshooting

Altera FPGA Dynamic Probe Online Help



FPGA Dynamic Probe Concepts

- "Example: Inserting an LAI Core" on page 48
- "Automated Logic Analyzer Set Up" on page 58



Example: Inserting an LAI Core

In this example, to create designs with the Logic Analyzer Interface (LAI) core, you need to download the Altera Quartus II Web Edition from:

"https://www.altera.com/support/software/download/altera_design/quartus_ we/dnl-quartus_we.jsp"

The Quartus II software requires a free license that you can request from:

"https://www.altera.com/support/licensing/free_software/lic-q2web.jsp"

Here are the steps to insert a LAI core into your design:

- 1 "Creating a Logic Analyzer Interface" on page 48
- 2 "Configuring the Logic Analyzer Interface Core Parameters" on page 49
- **3** "Mapping the Clock Signal" on page 50
- 4 "Mapping the Logic Analyzer Interface File Pins to Available I/O Pins" on page 51
- 5 "Mapping Internal Signals to the Logic Analyzer Interface Banks" on page 53
- 6 "Saving the Logic Analyzer Interface" on page 55
- 7 "Enabling/Disabling the Logic Analyzer Interface" on page 55
- 8 "Compiling the Quartus II Project" on page 56
- **9** "FYI: Creating Multiple Logic Analyzer Interface Instances in One FPGA" on page 57

Creating a Logic Analyzer Interface

The Logic Analyzer Interface File (.lai) defines the interface between internal FPGA signals and the external logic analyzer. To define the Quartus II Logic Analyzer Interface, you can create a new LAI file or use an existing file. In this walkthrough, you open an existing design and add an LAI core.

To create a new Logic Analyzer Interface file, perform the following steps:

- 1 Double-click the Quartus II design file, small.qpf, located at C:\ S800_altera
- 2 In the Quartus II File menu, click New.
- 3 In the New dialog, click the **Other Files** tab and select **Logic Analyzer Interface File**.

De Ał BI	vice Design Files HDL Include File ock Symbol File	Other Files	
Ci H	nain Description F exadecimal (Intel-I	ile Format) File	
M Si Te Ve	emory Initializatior gnaTap II File 21 Script File xxt File xctor Waveform F	i File ile	

4 Click OK.

The Logic Analyzer Interface editor opens.

Next • "Configuring the Logic Analyzer Interface Core Parameters" on page 49

Configuring the Logic Analyzer Interface Core Parameters

After creating the LAI file, you must configure the Logic Analyzer Interface core parameters. To configure these parameters, from the **Setup View** list select Core Parameters.

📄 lai1.lai*				
Instance Manager: In	valid JTAG Config	uration	2	×
Instance	Status		Incremental Compilation	LEs: 128
🔳 auto_lai_0	Not connected			128 cells
Logical View: 3	× <u>S</u> etup View:	Core Pa	rameters	•
Core Parameters	Pin count:		8	
Ban Pins 0/8 8/8	Bank count		1	-
auto_lai_0	Output/Cap	ture mode:	Combinational/Timing	
	Clock:			
			,	
	Power-up s	tate:	Tri-stated	•
auto lai 0				

NOTE

Screen resolution might cause your screen to look different; please resize or scroll within the window to view the parameter.

The parameters can be set depending on the user's debug resources. In this example, the following parameters are used:

- **Pin count =** 13
- Bank count = 4
- Capture mode = State
- **Power-up state** = tristate

The Node Finder tool will be used in the next step to map the Clock signal.

Parameter	Description
Pin Count	Signifies the number of pins dedicated to the Logic Analyzer Interface. The external pins connected to the users' debug header on the board. The Pin Count parameter can range from 1 to 256 pins.
Bank Count	Signifies the number of internal signals mapped to each pin. A Bank Count of 8 implies you will connect eight internal signals to each pin. The Bank Count parameter can range from 1 to 256 banks.
Output/Capture Mode	 Signifies the type of acquisition the external logic analyzer will perform. There are two options to select from: Combinational/Timing – This acquisition uses the logic analyzer’s internal clock to determine when to sample data. Registered/State – This acquisition uses a signal from the system under test to determine when to sample. Registered/State acquisition samples data synchronously providing a functional view of the FGPA.
Clock	The clock parameter is available when the Output/Capture Mode is set to Registered State. The sample clock can be any signal in the design.
Power-Up State	The Power-Up State parameter specifies the power-up state of the Logic Analyzer Interface pins. The two options are tri-stated for all pins or selection a particular bank.

Next • "Mapping the Clock Signal" on page 50

Mapping the Clock Signal

To assign the Clock, which will synchronize the state capture, click the ... button. This will launch the Node Finder tool.

🛋 lai1.lai*						×
Instance Manager: Ir	ivalio	JTAG Configura	ation	2		×
Instance SI		Status	Incremental Compilation		LE s: 12	28
auto_lai_0	١	Not connected			128 ce	lls
Logical View:	×	Setup View:	Core Pa	ameters		•
Core Parameters BanPins 0/8 8/8 auto lai 0		Pin count: Bank count:		8 1	: : :	~
	1	Output/Captu Clock:	re mode:	Combinational/Timing	•	
		Power-up stat	e:	Tri-stated	•	~
🔳 auto_lai_0						

Click **List** to display the signals from the synthesized design. Search through signal names and locate the clock signal, small_1s10:inst|clk. When the signal is selected, click the **≥**; button to make the core assignment. Click **OK** to continue.

Node Finder		×
Named: ×	Eilter: SignalTap II: pre-synthesis Customize	
Loo <u>k</u> in: Ismall	Cancel	
Nodes Foun <u>d</u> :	Selected Nodes:	
Name	Assignments 🔨 Name Assignments T	
<pre> out_port_from_the_led_pio out_port_from_the_led_pio[0] out_port_from_the_led_pio[1] out_port_from_the_led_pio[2] out_port_from_the_led_pio[3] out_port_from_the_led_pio[5] out_port_from_the_led_pio[5] out_port_from_the_led_pio[6] out_port_from_the_led_pio[7] PLD_CLEAR_N PLD_CLOCKINPUT small_1s10:inst[count1] small_1s10:inst[count1[1] small_1s10:inst[count1[2] small_1s10:inst[count1[2] </pre>	Unassigned PIN_H27 PIN_H28 PIN_L23 PIN_L24 PIN_J25 PIN_J26 PIN_J26 PIN_J26 PIN_L20 PIN_L20 PIN_L20 PIN_L23 PIN_K17 Unassigned Unassigned Unassigned Unassigned Unassigned Unassigned Unassigned	
small_1s10:inst[count1[4] small_1s10:inst[count1[5]	Unassigned	

• "Mapping the Logic Analyzer Interface File Pins to Available I/O Pins" on page 51

Mapping the Logic Analyzer Interface File Pins to Available I/O Pins

To configure the LAI I/O pins, from the Setup View list select Pins.

8 FPGA Dynamic Probe Concepts

NOTE

🖹 lai1.lai*						
Instance Manag	ger: <mark>Invalid</mark>	JTAG Co	onfiguration) [2]		×
Instance	St	tatus		Incremental Compilation LE	s: 157	
🔳 auto_lai_0	N	ot conne	cted	L 15	7 cells	
Logical View:	×	<u>S</u> etup \	view: Pi	ns		-
				Pin		1/0
Core Para	meters	Туре	Index	Name	Location	Standard
0/12	1	•	0	altera_reserved_lai_0_0	PIN_H26	LVTTL 🔺
Bank 1		•	1	altera_reserved_lai_0_1	PIN_N25	LVTTL
0/12	Pins		2	altera_reserved_lai_0_2	PIN_N26	LVTTL
Bank 2	13/13		3	altera_reserved_lai_0_3	PIN_L26	LVTTL
Bank 3		•	4	altera_reserved_lai_0_4	PIN_L25	LVTTL
0/12	i I Banks		5	altera_reserved_lai_0_5	PIN_M19	LVTTL 👻
auto_l*	0/48	0	6	[•	-	•
auto_lai_0						

To assign pin locations double-click inside the **Location** column next to the reserved pins in the **Name** column. This action opens the Pin Planner tool.

Please resize the window as necessary.

You can assign pins to the LAI by double-clicking in the **Location** column within Pin Planner. The pin assignment is typed in as PIN_XYZ or selected from the drop-down list.

The pins are assigned according to the following table.

Node Name	Location	I/O Standard
altera_reserved_lai_0_0	PIN_H26	LVTTL
altera_reserved_lai_0_1	PIN_N25	LVTTL
altera_reserved_lai_0_2	PIN_N26	LVTTL
altera_reserved_lai_0_3	PIN_L26	LVTTL
altera_reserved_lai_0_4	PIN_L25	LVTTL
altera_reserved_lai_0_5	PIN_M19	LVTTL
altera_reserved_lai_0_6	PIN_M20	LVTTL
altera_reserved_lai_0_7	PIN_K28	LVTTL
altera_reserved_lai_0_8	PIN_K27	LVTTL
altera_reserved_lai_0_9	PIN_N21	LVTTL
altera_reserved_lai_0_10	PIN_N22	LVTTL

Node Name	Location	I/O Standard	
altera_reserved_lai_0_11	PIN_M26	LVTTL	
altera_reserved_lai_0_12	PIN_M25	LVTTL	



Once all LAI nodes have been assigned pin locations, close the Pin Planner window. The pin assignments will be reflected in the Setup View.

Next • "Mapping Internal Signals to the Logic Analyzer Interface Banks" on page 53

Mapping Internal Signals to the Logic Analyzer Interface Banks

Having specified the number of banks to use in the Core Parameters, internal signals must be assigned for each bank in the Logic Analyzer Interface. Click the **Setup View** arrow and select **Bank n** to assign signals a bank at a time or select **All Banks**.

8 FPGA Dynamic Probe Concepts

🛋 lai1.lai							×
Instance Manager: Invalid JTAG Configuration					2		x
Instance	Stat	us		Increm	ental Compilati	on LEs: 157	
auto_lai_0 Not connected						157 cells	
Logical View:	×	<u>S</u> etup Vie	ew: All E	lanks			•
Cons Domenton		Bank	Pin			Node	
Bank 0		Name	Index	Туре	Alias	Name	
12/12			0		State Clock	small_1s10:instjclk	^
Bank 1			1	Ð		small_1s10:inst count1[0]	
12/12 Pins	-		2	•		small_1s10:inst count1[1]	
12/12			3	•		small_1s10:inst count1[2]	
Bank 3			4	Ð		small_1s10:inst count1[3]	
12/12 All Banks			5	E		small_1s10:inst count1[4]	1
auto_lai_0 48/48		Bank 0	6	•		small_1s10:inst count1[5]	
			7	•		small_1s10:inst count1[6]	1
			8	•		small_1s10:inst count1[7]	
			9	•		small_1s10:inst count1[8]	1
			10	•		small_1s10:inst count1[9]	1
			11	•		small_1s10:inst count1[10]	1
			12	•		small_1s10:inst count1[11]	1.
■: auto_lai_0					A.I. A.I.		

NOTE

If using a state core, bit 0 of each bank will be reserved for the clock.

To begin assigning signals, double-click in the **Name** column to launch Node Finder. Within Node Finder click **List** to display all the design signals. Find the signals of interest, and select the signals from the Node Finder dialog box. As signals are assigned in Node Finder, the LAI schematic in the Logical View begins to reflect the assignments.

Jamed:	▼ Filter: SignalTar	n II: pre-si	inthesis	List	ПК
		p II. pie sj			
Look in: Jismall			Include subentities	Stop	Lance
Nodes Found:			Selected Nodes:		
Name	Assignments 🔺		Name	Assignments T	r i
🐼 small_1s10:inst count1	Unassigned 📥		Ismallsmall_1s10:inst[count1[0]	Unassigned F	3
small_1s10:inst[count1[0]	Unassigned		Ismallsmall_1s10:inst[count1[1]	Unassigned F	3
small_1s10:inst[count1[1]	Unassigned	>	Ismallsmall_1s10:inst[count1[2]	Unassigned F	3
🐵 small_1s10:inst count1[2]	Unassigned	Jan	st count1[3]	Unassigned F	3
🐵 small_1s10:inst count1[3]	Unassigned	_ <u>{Cob</u> }	smansman_rsro.nst count1[4]	Unassigned F	3
🐵 small_1s10:inst count1[4]	Unassigned	<	Ismallsmall_1s10:inst[count1[5]	Unassigned F	3
🐵 small_1s10:inst count1[5]	Unassigned		Ismallsmall_1s10:inst[count1[6]	Unassigned F	3
🐵 small_1s10:inst count1[6]	Unassigned	<<	Ismallsmall_1s10:inst[count1[7]	Unassigned F	3
small_1s10:inst[count1[7]	Unassigned		Ismallsmall_1s10:inst[count1[8]	Unassigned F	3
small 1s10tinstlcount1[8]	Unassigned 🔛				
<	>		<	>	

The design contains four counters; count1 and count3 are up counters and count2 and count4 are down counters. Assign a counter output per bank, for example, count1 to Bank0, count2 to Bank1, count3 to Bank2, and count4 to Bank3.

Next • "Saving the Logic Analyzer Interface" on page 55

Saving the Logic Analyzer Interface

To save the LAI file perform the following steps:

- 1 In the File menu, click Save As, the save dialog box opens.
- 2 In the File name box, enter your desired file name. Click Save.
- **3** When prompted, click **Yes** to enable the Logic Analyzer Interface file for the current project.

Quartus	s II. 🛛 🔀
⚠	Do you want to enable Logic Analyzer Interface File "lai1.lai" for the current project?
	<u>Yes</u> <u>No</u> Cancel

Next • "Enabling/Disabling the Logic Analyzer Interface" on page 55

Enabling/Disabling the Logic Analyzer Interface

The Logic Analyzer Interface can be enabled and disabled to include and remove the core from the design. This step can be preformed as follows:

- 1 On the Assignments menu click Settings. The Settings dialog box opens.
- 2 Under Category, click Logic Analyzer Interface. The Logic Analyzer Interface displays.
- **3** Make sure **Enable Logic Analyzer Interface** is checked to include the core.
- **4** The **Logic Analyzer Interface file name** displays the full path name of the LAI file.



Click **OK** to continue.

Next • "Compiling the Quartus II Project" on page 56

Compiling the Quartus II Project

The next step is to compile the project. On the **Processing** menu click **Start Compilation**.

NOTE

Warnings during compilation are okay.

To ensure the Logic Analyzer Interface is properly compiled with the project, expand the entity hierarchy in the Project Navigator. If the Logic Analyzer Interface is compiled with the design, the *sld_hub* and *sld_multitap* entities will be shown in the project navigator.

Project Navigator	- ×
Entity	Logic Cel
Stratix: EP1S10F780C6	
🖻 💦 small	260 (0)
Home sld_multitap:auto_lai_0	54 (23)
<u>abd</u> small_1s10:inst	103 (103)
	103 (28)

 "FYI: Creating Multiple Logic Analyzer Interface Instances in One FPGA" on page 57

FYI: Creating Multiple Logic Analyzer Interface Instances in One FPGA

The Logic Analyzer Interface supports multiple interfaces in a single FPGA. This feature is particularly useful when you want to build LAI configurations that contain different settings. For example, build one LAI instance to perform Registered/State analysis and build another instance that performs Combinational/Timing analysis on the same set of signals. Another example would be to perform Registered/State analysis on portions of the design that are in different clock domains.

To create multiple Logic Analyzer Interfaces, right-click in the **Instance Manger** window and select **Create Instance**.

📸 small.bdf				📄 📄 lai1	l.lai*
Instance Manager:	Invalid JT	AG Configuration		2	
Instance	S	Create Instance		ompilation	LE s: 285
💷 auto_lai_0	N	<u>C</u> redie Instance	Del		157 cells
🔳 auto_lai_1	N	D <u>e</u> lete Instance	Del		128 cells
		Rena <u>m</u> e Instance	F2	I	
		Instance Status <u>H</u> el	р		

Automated Logic Analyzer Set Up

The FPGA dynamic probe automatically sets up the logic analyzer for the type of Altera LAI core it connects to.

For timing (asynchronous) cores, the FPGA dynamic probe automatically sets up:

• Timing mode.

For state (synchronous) cores, the FPGA dynamic probe automatically sets up:

- State mode.
- Master clock mode.
- Clock signal and edge.

For all types of cores, the FPGA dynamic probe automatically sets up:

• Bus/signal names for the selected bank.

In the logic analyzer module's "Analyzer Setup dialog" (in the online help), you can rename buses/signals (see page 32) and define additional bus/signal names (see page 33), but changing any of the other settings made by the FPGA dynamic probe will interfere with its operation.

You are free to change settings that are untouched by the FPGA dynamic probe (like memory depth, trigger position, or sampling positions); they will not affect the FPGA dynamic probe.

Altera FPGA Dynamic Probe Online Help

9 FPGA Dynamic Probe Reference

- FPGA Dynamic Probe Setup Dialog (see page 60)
 - Cable Connection Dialog (see page 61)
 - Pin Mapping Dialog (see page 61)
 - Pin Mapping Edit Dialog (see page 62)
 - Select FPGA Configuration File Dialog (see page 64)
 - Select Signal Import File Dialog (see page 64)
 - Properties Dialog (see page 65)
 - Core Details Dialog (see page 65)
- FPGA Dynamic Probe Bank Selection Dialog (see page 67)
 - Trim Bus/Signal Names Dialog (see page 67)
- Specifications and Characteristics (see page 69)



FPGA Dynamic Probe Setup Dialog

The FPGA Dynamic Probe (see page 9) dialog lets you:

- Establish a connection between the logic analyzer and an FPGA with one or more Altera LAI cores.
- Configure the FPGA with a new design file.
- Map FPGA pins to probe pins/pads.
- Import signal names from the FPGA design tool.

FPGA I	Dynamic	Probe Setup				
	Device @1: E "	Core auto_lai_0	Imported Signals	Pin Mapping	Details @1: EP1C20 (0x020840D Timing, 4 banks, 8 pins	Cable Connection) Cable Connection) Configure Device Import Bus/Signals auto_lai_0 Pin Mapping
<						Properties
					<u> </u>	incel <u>H</u> elp

On (in FPGA device list)	The check boxes in this column let you enable or disable a core for use. If you do not want to use a particular core (typically in the multiple core case), you can uncheck its box, and the outputs of the core will be disabled. If a core's "always on" mode is enabled (see the core Details (see page 65) within its Properties dialog (see page 65)), the core is always enabled and can be probed at powerup (bank 0 will be the selected bank). In this case, the core cannot be disabled, and the check box cannot be unchecked.
Cable Connection	Opens the Cable Connection dialog (see page 61) for establishing a connection between the logic analyzer and the Altera LAI core.
Configure Device	Opens the Select FPGA Configuration File dialog (see page 64) for downloading a design into the selected FPGA device.
Import Bus/Signals	Opens the Select Signal Import File dialog (see page 64) for importing internal FPGA bus/signal names.

Pin Mapping	Opens the Pin Mapping dialog (see page 61) for defining the logic analyzer probes that are used to connect to the FPGA and setting up the pin mapping (see Step 6. Map FPGA pins (see page 35)).
Properties	Opens the Properties dialog (see page 65) which lets you rename devices and cores as well as display information about the selected Altera LAI core.

See Also • Measurement Steps (see page 27)

Cable Connection Dialog

The Cable Connection dialog lets you specify the the type of cable used to connect the logic analyzer to the device under test JTAG port.

Cable Connection	X
• Existing cables	
USB-Blaster [USB-0] ByteBlaster [LPT1]	
No Cable (Demo Mode)	
	<u> </u>

Existing cables	Lets you select a cable that has been set up using the Altera Quartus II Programmer (or design) software.
No Cable (Demo Mode)	Simulates a cable connection for demonstration purposes.

- **See Also** Step 2. Set up the JTAG cable using the programmer software (see page 19)
 - Step 3. Establish connection between analyzer and JTAG cable (see page 28)

Pin Mapping Dialog

The Pin Mapping dialog lets you define the logic analyzer probes that are used to connect to the FPGA, and it lets you set up the FPGA pin to probe pin/pad mapping.

9 FPGA Dynamic Probe Reference



Add Probe	Opens the "Select the type of probe to add" dialog; when you click OK , the Pin Mapping Edit dialog (see page 62) is opened for mapping the FPGA output pins to the probe pins/pads. If your probe doesn't appear in the list of probe types, you can "download the latest probe definitions from the web" (in the online help).
Edit Probe	For the selected probe, opens the Pin Mapping Edit dialog (see page 62) for editing the FPGA output pins to probe pin/pad mapping.
Delete Probe	Deletes the selected probe.

See Also • Step 6. Map FPGA pins (see page 35)

Pin Mapping Edit Dialog

The Pin Mapping Edit dialog lets you map the FPGA output pins to the logic analyzer probe pins/pads.



Reference Designator	Identifies the reference designator (in the device under test) of the probe connector, the connectorless probe retention module, or pins probed by flying leads.
FPGA Pins	Lists the FPGA pins used for the Altera LAI core outputs. When dragging these pins onto the pin/pad map, you can select multiple pins using Shift-click or Ctrl-click.
Probe Pin/Pad Diagram	Diagrams probe pins/pads, flying-lead channels, or termination adapter pins, and provides fields for dropping FPGA pin numbers.
Logic Analyzer Slot, Pod	Lets you select the logic analyzer module slots/pods to which the probe, flying leads, or termination adapter is connected.
Show Logic Analyzer Channels	When checked, the logic analyzer pod channel numbers are displayed in the probe pin/pad diagram next to the pin/pad numbers.
Clear	Clears all FPGA pins that have been mapped to pins/pads. You can clear individual pin mappings by dragging a pin from the pin/pad diagram back to the FPGA Pins list.

See Also • Step 7. Map FPGA pins (see page 39)

Select FPGA Configuration File Dialog

The Select FPGA Configuration File dialog lets you select a design file for downloading into an FPGA device on the JTAG chain.

Select FPGA Co	onfiguration File)			? 🛛
Look in:	😂 Altera_LAI	×	G 🦻	بي 🥙	
My Recent Documents	2cores.sof small.sof				
Desktop					
My Documents					
My Computer					
	File <u>n</u> ame:	small.sof		~	<u>O</u> pen
My Network	Files of type:	FPGA Configuration files (*.sof)		~	Cancel

Select Signal Import File Dialog

The Select Signal Import File dialog lets you select the file (from the FPGA design tool) that contains the names of the internal buses/signals that appear on Altera LAI core inputs.



See Also • Step 5. Import signal names (see page 31)

Properties Dialog

The Properties dialog lets you change device, core, and bank names as well as view detailed information about the core.

Properties	×	
Device Name		
@1: EP1C20 (0×020840DD)		
(@1. EF1C20 (0x0200+05D)		
Core Name		
auto_lai_0		
Test Bank Data		
	~	
Details		
OK Cancel		

Properties 🛛 🔀
Core Name
auto_lai_0
Bank Name
Bank 1
Test Bank Data
×
Details
QK Cancel

Device Name	Lets you rename the selected device.	
Core Name	Lets you rename the selected core.	
Bank Name	Lets you rename the selected bank.	
Test Bank Data	Not available with Altera LAI cores.	
Details	Opens the Core Details dialog (see page 65) which displays information about the selected Altera LAI core.	

Core Details Dialog

The Core Details dialog displays information about the selected Altera LAI core.

9 FPGA Dynamic Probe Reference



FPGA Dynamic Probe Bank Selection Dialog

The FPGA Dynamic Probe (see page 9) Bank Selection dialog lets you:

- Select a different bank of internal signals to probe.
- Rename individual signals (without having to do global trimming) by triple-clicking the signal name.

FPGA Dynamic Probe Bank Selection		×
Summary auto_lai_0 auto_lai_0 Bank 0 small_1s10:instcount1(0) small_1s10:instcount1(2) small_1s10:instcount1(3) small_1s10:instcount1(5) Bank 1 Bank 2 Bank 4 Bank 4 Bank 4 Bank 5	<u>Run Eye Finder</u> <u>Irim Bus/Signal Names</u> <u>Properties</u> Selected signal bank: Bank 0 Last selected in core: None	
	OK Cancel Help	

Run Eye Finder	Opens the "Thresholds and Sample Positions dialog" (in the online help) for running <i>eye finder</i> to automatically adjust the state mode sampling positions. This button is available when probing only state (synchronous) cores. If there are any timing (asynchronous) cores, the logic analyzer is set up in timing mode, and <i>eye finder</i> is not available.
Trim Bus/Signal Names	Opens the Trim Bus/Signal Names dialog (see page 67) for shortening imported FPGA internal bus/signal names.
Properties	Opens the Properties dialog (see page 65) which lets you rename cores and banks as well as display information about the selected Altera LAI core.

See Also • Measurement Steps (see page 27)

Trim Bus/Signal Names Dialog

The Trim Bus/Signal Names dialog lets you specify how imported bus/signal names should be shortened.



If the Signal Name is at least	Only bus/signal names longer than this value will be trimmed.
Remove left most	The number of characters to remove from the beginning of the names.
Remove right most	The number of characters to remove from the end of the names.
Remove all except right most	The number of characters to leave at the end of the names.
Locate right most Nth occurrence of the string	The string before which all characters from the names are stripped.
Restore original Signal names	Undoes the bus/signal name trimming.

See Also • Step 5. Import signal names (see page 31)

Specifications and Characteristics

The FPGA dynamic probe for Altera FPGAs has these specifications and characteristics:

- Supported Logic Analyzers (see page 69)
- FPGA Dynamic Probe Software Application (see page 69)

Supported Logic Analyzers	Standalone logic analyzers:	1680 Series, 1690 Series, 16800 Series
	Modular logic analysis systems:	 16900A, 16901A, 16902A, 16902B, 16903A with one or more of the following cards: 16740A, 16741A, 16742A. 16750A, 16751A, 16752A, 16753A, 16754A, 16755A, 16756A. 16760A. 16910A, 16911A, 16950A, 16950B, 16951B. A single node-locked FPGA dynamic probe license will enable all modules within a 16900 Series system.
	Triggering capabilities:	Determined by logic analyzer.
	Supported probing mechanisms:	Soft touch (34-channel and 17-channel), Mictor, Samtec, Flying lead

FPGA Dynamic Probe Software Application

Maximum number of devices supported on a JTAG scan chain:	256
Maximum number of Altera LAI cores supported per FPGA device:	15

9 FPGA Dynamic Probe Reference

Altera FPGA Dynamic Probe Online Help



Probe Control, COM Automation

The *Agilent Logic Analyzer* application includes the COM Automation Server. This software lets you write programs that control the *Agilent Logic Analyzer* application from remote computers on the Local Area Network (LAN).

In a COM automation program, you can configure a probe by:

- Loading a configuration file (which configures the complete logic analyzer setup).
- Using the "Probe" (in the online help) object's "DoCommands" (in the online help) method with an XML-format string parameter (see Probe Setup, XML Format (see page 73)).

You can get information about a probe's configuration using the Probe object's "QueryCommand" (in the online help) method. Queries supported by the FPGA dynamic probe are listed below.

For more information about logic analyzer COM automation and probe objects in general, see "COM Automation" (in the online help).

XML-Based The FPGA dynamic probe supports the following XML-based queries (made with the "Probe" (in the online help) object's "QueryCommand" (in the online help) method).

Query	Description
GetAllSetup	Returns the current setup, using the full tag set, used for writing generic configuration files (see the XML format <properties> element (see page 98)).</properties>
GetProperties	Returns the current setup, using the full tag set, equivalent to "GetAllSetup" (see the XML format <properties> element (see page 98)).</properties>

See Also

- "COM Automation" (in the online help)
 - Probe Setup, XML Format (see page 73)



10 Probe Control, COM Automation
Altera FPGA Dynamic Probe Online Help

11

Probe Setup, XML Format

When you save logic analyzer configurations to XML format files, setup information for the FPGA dynamic probe is included.

This XML format setup information is also used when writing COM automation programs to control the logic analyzer from a remote computer.

XML elements for the FPGA dynamic probe have the following hierarchy:

```
<Properties> (see page 98)
   <ATC_II> (see page 75)
   <JTAG_Chain> (see page 88)
   <Devices> (see page 87)
      <Device> (see page 85)
         <Cores> (see page 83)
            <Core> (see page 81)
               <Banks> (see page 80)
                  <Bank> (see page 76)
                     <Signals> (see page 102)
                        <Signal> (see page 100)
                           <Labels> (see page 91)
                              <Label> (see page 89)
                     <NonATCLabels> (see page 92)
                        <Label> (see page 90)
                           <Assignment> (see page 74)
                     <WindowInfo> (see page 107)
                     <SymbolInfo> (see page 105)
                     <TriggerInfo> (see page 106)
               <PinMapping> (see page 93)
                  <DefinedProbes> (see page 84)
                     <Probe> (see page 97)
                        <Pods> (see page 96)
                           <Pod> (see page 95)
                        <Signals> (see page 104)
                            <Signal> (see page 101)
```

See Also • "XML Format" (in the online help)

• Probe Control, COM Automation (see page 71)



<Assignment> Element

The <Assignment> element describes the logic analyzer pod and channel assignments for a Label element (under NonATCLabels).

Attributes

Name	Description
Channel	'number'
Pod	'number'

Parents This element can have the following parents: <Label> (see page 90).

```
Example <NonATCLabels>
```

```
<Label Name='My Bus 2' Comment='' Handle='131'>

<Assignment Pod='1' Channel='0' />

<Assignment Pod='1' Channel='1' />

<Assignment Pod='1' Channel='2' />

<Assignment Pod='1' Channel='3' />

<Assignment Pod='1' Channel='4' />

<Assignment Pod='1' Channel='5' />

<Assignment Pod='1' Channel='6' />

<Assignment Pod='1' Channel='6' />

</Label>

</NonATCLabels>
```

<ATC_II> Element

The $<ATC_{II}>$ element describes the selected device, the selected core, and the cable type.

Attributes

Name	Description
CableType	'number'
ParallelCablePort	'number'
ParallelCableSpeed	'number'
ParallelCableType	'number'
SelectedCore	'number'
SelectedDevice	'number'
USBCableSpeed	'number'

- **Parents** This element can have the following parents: <Properties> (see page 98).

<Bank> Element

The <Bank> element describes a bank within an Altera LAI core.

Attributes

Name	Description
CalibrationBank	'F' (false) or 'エ' (true)
Name	'string'
NumDataPins	'number'
State2X	'F' (false) or 'エ' (true)

Children This element can have the following children: <Signals> (see page 102), <NonATCLabels> (see page 92), <WindowInfo> (see page 107), <SymbolInfo> (see page 105), <TriggerInfo> (see page 106).

Parents This element can have the following parents: <Banks> (see page 80).

```
Example
         <Bank Name='Bank-0' CalibrationBank='F' NumDataPins='16' State2X='F'>
             <Signals>
               <Signal>
                   <Labels>
                      <Label Name='/s2mon/tid' Bit='0' />
                   </Labels>
                </Signal>
                <Signal>
                   <Labels>
                      <Label Name='/s2mon/tid' Bit='1' />
                   </Labels>
                </Signal>
                <Signal>
                   <Labels>
                      <Label Name='/s2mon/tid' Bit='2' />
                  </Labels>
                </Signal>
                <Signal>
                   <Labels>
                      <Label Name='/s2mon/tid' Bit='3' />
                   </Labels>
                </Signal>
                <Signal>
                   <Labels>
                     <Label Name='/s2mon/tid' Bit='4' />
                   </Labels>
                </Signal>
                <Signal>
                   <Labels>
                      <Label Name='/s2mon/tid' Bit='5' />
                   </Labels>
                </Signal>
                <Signal>
```

```
<Labels>
        <Label Name='/s2mon/tid' Bit='6' />
      </Labels>
   </Signal>
   <Signal>
     <Labels>
        <Label Name='/s2mon/tid' Bit='7' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
        <Label Name='/s2mon/lastackid' Bit='0' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
         <Label Name='/s2mon/lastackid' Bit='1' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
        <Label Name='/s2mon/lastackid' Bit='2' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
        <Label Name='/s2mon/lastackid' Bit='3' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
        <Label Name='/s2mon/lastackid' Bit='4' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
        <Label Name='/s2mon/lastackid' Bit='5' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
        <Label Name='/s2mon/lastackid' Bit='6' />
      </Labels>
   </Signal>
   <Signal>
      <Labels />
   </Signal>
</Signals>
<NonATCLabels />
<WindowInfo WindowHandle='1' WindowSettings='
<Setup&gt;
   <Sampling PerDivision=&apos;5 ns&apos;
        Delay='0 s'/>
   <BusSignals&gt;
     <Clear/&gt;
      <BusSignal Module=&apos;My 1682D-1&apos;
           Name='My Bus 1' DefaultBase='Hex'
```

```
Color=' hFFFFFF' Height=' 30' />
    <BusSignal Module=&apos;My 1682D-1&apos;
         Name='/s2mon/tid' DefaultBase='Hex'
         Color=' hFFFFFF' Height=' 30' />
    <BusSignal Module=&apos;My 1682D-1&apos;
         Name='/s2mon/s2mstate'
         DefaultBase='Hex' Color='hFFFFF'
         Height='30'/>
    <BusSignal Name=&apos;Time&apos; Color=&apos;hFFFFFF&apos;
         Height='30'/>
  </BusSignals&gt;
</Setup&gt;
'/>
<WindowInfo WindowHandle='2' WindowSettings='
<BusSignals&gt;
  <Clear/&gt;
  <BusSignal Module=&apos;My 1682D-1&apos;
       Name='Sample Number' Color='hFFFFF'
       Alignment='Right' Width='112'/>
  <BusSignal Module=&apos;My 1682D-1&apos;
       Name='My Bus 1' DefaultBase='Hex'
       Color=' hFFFFFF' Alignment=' Right'
       Width='113'/>
  <BusSignal Name=&apos;Time&apos;
       DefaultBase='Absolute' Color='hFFFFF'
       Alignment='Right' Width='152'/>
  <BusSignal Module=&apos;My 1682D-1&apos;
       Name='/s2mon/tid' DefaultBase='Hex'
       Color='hFFFFFF' Alignment='Right'
       Width='113'/>
  <BusSignal Module=&apos;My 1682D-1&apos;
       Name='/s2mon/s2mstate'
       DefaultBase='Hex' Color='hFFFFFF'
       Alignment='Right' Width='120'/>
</BusSignals&gt;
'/>
<SymbolInfo ModuleHandle='1' SymbolSettings='
<Module&gt;
  <BusSignalSetup&gt;
    <BusSignals&gt;
       <BusSignal Name=&apos;My Bus 1&apos;/&gt;
       <Folder Name=&apos;Core 0 FPGA Probe&apos;
           Comment='Created by FPGA Dynamic Probe-1'>
         <BusSignal Name=&apos;/s2mon/tid&apos;/&gt;
         <BusSignal Name=&apos;/s2mon/s2mstate&apos;/&gt;
       </Folder&gt;
    </BusSignals&gt;
    <NetlistImport/&gt;
  </BusSignalSetup&gt;
  <Config TimeOfTrigger=&apos;1.110240661 Gs&apos;
       CorrelatedTriggerTime='0 s'
       UserSkewTime='0 s'
       SystemTrigger='T'/>
</Module&gt;
'/>
<TriggerInfo ModuleHandle='1' TriggerSettings='
<Module&gt;
```

```
<Trigger Mode=&apos;State&apos; Type=&apos;Normal&apos;&gt;
       <StoreQual Mode=&apos;Custom&apos;&gt;
          <Event ParensNeeded=&apos;F&apos;&gt;
            <Anything/&gt;
          </Event&gt;
       </StoreQual&gt;
       <Step Number=&apos;1&apos;&gt;
          <If&gt;
            <Event ParensNeeded=&apos;F&apos;&gt;
               <BusSignal Name=&apos;/s2mon/tid&apos;
                    Bit='All' Operator='Equals'
                    Value='h3F'/>
            </Event&gt;
            <Occurrence Value=&apos;1&apos;
                 Mode='Eventual'/>
            <Action&gt;
               <TriggerAction Operator=&apos;Fill Memory&apos;&gt;
                 <StoreQual Mode=&apos;Custom&apos;&gt;
                    <Event ParensNeeded=&apos;F&apos;&gt;
                      <DefaultStore/&gt;
                    </Event&gt;
                 </StoreQual&gt;
               </TriggerAction&gt;
            </Action&gt;
          </If&gt;
       </Step&gt;
    </Trigger&gt;
    <Config TimeOfTrigger=&apos;1.110240661 Gs&apos;
          CorrelatedTriggerTime='0 s'
          UserSkewTime='0 s'
          SystemTrigger='T'/>
  </Module&gt;
  '/>
</Bank>
```

<Banks> Element

The <Banks> element contains descriptions of banks within an Altera LAI core. Children This element can have the following children: <Bank> (see page 76). Parents This element can have the following parents: <Core> (see page 81). Example <Banks> <Bank Name='Bank 0' CalibrationBank='F' NumDataPins='16' State2X='F'> . . . </Bank> <Bank Name='Bank 1' CalibrationBank='F' NumDataPins='16' State2X='F'> . . . </Bank> <Bank Name='Bank 2' CalibrationBank='F' NumDataPins='16' State2X='F'> . . . </Bank> <Bank Name='Bank 3' CalibrationBank='F' NumDataPins='16' State2X='F'> . . . </Bank> <Bank Name='Test Bank' CalibrationBank='T' NumDataPins='16' State2X='F'> . . . </Bank> </Banks>

<Core> Element

The <Core> element describes a core within a device on the JTAG chain.

Attributes

Name	Description
ClockPodIndex	'number'
CoreCannotBeMaster	'F' (false) or 'エ' (true)
CoreIsMaster	'F' (false) or 'エ' (true)
Latency	'number'
MinimumPeriod	'number'
Name	'string'
NumBanks	'number'
NumPins	'number'
NumSignals	'number'
SelectedBank	'number'
SelectedForUse	'F' (false) or 'エ' (true)
SelectedSignal	'number'
StateMode	'F' (false) or 'エ' (true)
TDM_1X	'F' (false) or 'エ' (true)
TestBankAvailable	'F' (false) or 'エ' (true)
TestBankMode	'number'
Туре	'number'
ThresholdCode	'number'

Children This element can have the following children: <Banks> (see page 80), <PinMapping> (see page 93).

Parents This element can have the following parents: <Cores> (see page 83).

```
State2X='F'>
         . . .
      </Bank>
      <Bank Name='Bank 2' CalibrationBank='F' NumDataPins='16'
           State2X='F'>
         . . .
      </Bank>
      <Bank Name='Bank 3' CalibrationBank='F' NumDataPins='16'
          State2X='F'>
        . . .
      </Bank>
      <Bank Name='Test Bank' CalibrationBank='T' NumDataPins='16'
          State2X='F'>
         . . .
      </Bank>
   </Banks>
   <PinMapping Attached='T' ModuleHandle='1'>
      <DefinedProbes>
         <Probe Name='J1' Type='E5346A 34-ch Mictor single-ended probe'>
            <Pods>
               <Pod Index='0' />
               <Pod Index='1' />
            </Pods>
            <Signals>
               . . .
            </Signals>
         </Probe>
      </DefinedProbes>
   </PinMapping>
</Core>
```

<Cores> Element

The *<*Cores*>* element contains descriptions of cores within a device on the JTAG chain. Children This element can have the following children: <Core> (see page 81). Parents This element can have the following parents: <Device> (see page 85). Example <Cores> <Core Name='Core 0' Type='0' SelectedBank='0' SelectedSignal='-1'</pre> NumBanks='5' NumSignals='16' NumPins='9' StateMode='T' TDM_1X='F' ThresholdCode='92' TestBankAvailable='T' ClockPodIndex='0' SelectedForUse='T' CoreIsMaster='F' CoreCannotBeMaster='F' MinimumPeriod='-1' TestBankMode='2' Latency='4'> <Banks> . . . </Banks> <PinMapping Attached='T' ModuleHandle='1'> . . . </PinMapping> </Core> <Core Name='Core 1' Type='0' SelectedBank='0' SelectedSignal='-1'</pre> NumBanks='2' NumSignals='9' NumPins='9' StateMode='F' TDM_1X='T' ThresholdCode='92' TestBankAvailable='F' ClockPodIndex='2' SelectedForUse='T' CoreIsMaster='T' CoreCannotBeMaster='F' MinimumPeriod='-1' TestBankMode='0'> <Banks> . . . </Banks> <PinMapping Attached='T' ModuleHandle='1'> . . . </PinMapping> </Core> </Cores>

<DefinedProbes> Element

The <DefinedProbes> element contains defined probes.

Children This element can have the following children: <Probe> (see page 97).
Parents This element can have the following parents: <PinMapping> (see
 page 93).

```
Example
         <DefinedProbes>
            <Probe Name='J1' Type='E5346A 34-ch Mictor single-ended probe'>
               <Pods>
                   <Pod Index='0'/>
                   <Pod Index='1'/>
               </Pods>
               <Signals>
                   <Signal Name='ATD0' Pin='38' PinMapIndex='0'
                         ClockChannel='F' />
                   <Signal Name='ATD1' Pin='36' PinMapIndex='1'
                        ClockChannel='F' />
                   <Signal Name='ATD2' Pin='34' PinMapIndex='2'
                        ClockChannel='F' />
                   <Signal Name='ATD3' Pin='32' PinMapIndex='3'
                        ClockChannel='F' />
                   <Signal Name='ATD4' Pin='30' PinMapIndex='4'
                        ClockChannel='F' />
                   <Signal Name='ATD5' Pin='28' PinMapIndex='5'
                        ClockChannel='F' />
                   <Signal Name='ATD6' Pin='26' PinMapIndex='6'
                         ClockChannel='F' />
                   <Signal Name='ATD7' Pin='24' PinMapIndex='7'
                         ClockChannel='F' />
                   <Signal Name='ATD8' Pin='22' PinMapIndex='8'
                         ClockChannel='F' />
                   <Signal Name='ATD9' Pin='20' PinMapIndex='9'
                         ClockChannel='F' />
                   <Signal Name='ATD10' Pin='18' PinMapIndex='10'
                         ClockChannel='F' />
                   <Signal Name='ATD11' Pin='16' PinMapIndex='11'
                         ClockChannel='F' />
                   <Signal Name='ATD12' Pin='14' PinMapIndex='12'
                         ClockChannel='F' />
                   <Signal Name='ATD13' Pin='12' PinMapIndex='13'
                         ClockChannel='F' />
                   <Signal Name='ATD14' Pin='10' PinMapIndex='14'
                         ClockChannel='F' />
                   <Signal Name='ATCK' Pin='6' PinMapIndex='16'
                        ClockChannel='T' />
               </Signals>
            </Probe>
         </DefinedProbes>
```

<Device> Element

The <Device> element describes a device on the JTAG chain.

Attributes

Name	Description
CDCFilename	'string'
Configurable	'F' (false) or 'T' (true)
FPGAFilename	'string'
IRLength	'number'
Name	'string'
NumCores	'number'
SelectedForUse	'F' (false) or 'エ' (true)
Туре	'number'
UserRegNum	'number'

Children This element can have the following children: <Cores> (see page 83).

Parents	This element can	have the	following	parents:	<devices></devices>	(see	page	87).
---------	------------------	----------	-----------	----------	---------------------	------	------	----	----

Example <Device Name='Device 1' Type='0' NumCores='2' UserRegNum='1'</pre> Configurable='T' IRLength='6' CDCFilename='C:\Documents and Settings\user\My Documents\Agilent Technologies\Logic Analyzer\Config Files\demo\Altera_demo_V8.cdc' FPGAFilename='C:\Documents and Settings\user\My Documents\Agilent Technologies\Logic Analyzer\Config Files\demo\Altera_demo_V8.bit' SelectedForUse='F'> <Cores> <Core Name='Core 0' Type='0' SelectedBank='0' SelectedSignal='-1'</pre> NumBanks='5' NumSignals='16' NumPins='9' StateMode='T' TDM_1X='F' ThresholdCode='92' TestBankAvailable='T' ClockPodIndex='0' SelectedForUse='T' CoreIsMaster='F' CoreCannotBeMaster='F' MinimumPeriod='-1' TestBankMode='2' Latency='4'> <Banks> . . . </Banks> <PinMapping Attached='T' ModuleHandle='1'> . . . </PinMapping> </Core> <Core Name='Core 1' Type='0' SelectedBank='0' SelectedSignal='-1'

NumBanks='2' NumSignals='9' NumPins='9' StateMode='F'
TDM_1X='T' ThresholdCode='92' TestBankAvailable='F'
ClockPodIndex='2' SelectedForUse='T' CoreIsMaster='T'
CoreCannotBeMaster='F' MinimumPeriod='-1' TestBankMode='0'>
<Banks>

. . .

```
</Banks>
<PinMapping Attached='T' ModuleHandle='1'>
...
</PinMapping>
</Core>
</Cores>
```

<Devices> Element

The <Devices> element contains descriptions of the devices on the JTAG chain.

Children This element can have the following children: <Device> (see page 85).

```
Parents This element can have the following parents: <Properties> (see page 98).
```

```
Example
         <Devices>
             <Device Name='Device 0' Type='0' NumCores='0' UserReqNum='0'</pre>
                   Configurable='F' IRLength='8' CDCFilename='' FPGAFilename=''
                   SelectedForUse='F'>
                <Cores/>
             </Device>
             <Device Name='Device 1' Type='0' NumCores='2' UserRegNum='1'</pre>
                   Configurable='T' IRLength='6' CDCFilename='C:\Documents and
                   Settings\user\My Documents\Agilent Technologies\Logic
                   Analyzer\Config Files\demo\Altera_demo_V8.cdc'
                   FPGAFilename='C:\Documents and Settings\user\My
                   Documents\Agilent Technologies\Logic Analyzer\Config
                   Files\demo\Altera_demo_V8.bit' SelectedForUse='F'>
                <Cores>
                   <Core Name='Core 0' Type='0' SelectedBank='0'
                         SelectedSignal='-1' NumBanks='5' NumSignals='16'
                         NumPins='9' StateMode='T' TDM_1X='F' ThresholdCode='92'
                         TestBankAvailable='T' ClockPodIndex='0'
                         SelectedForUse='T' CoreIsMaster='F'
                         CoreCannotBeMaster='F' MinimumPeriod='-1'
                         TestBankMode='2' Latency='4'>
                      <Banks>
                         . . .
                      </Banks>
                      <PinMapping Attached='T' ModuleHandle='1'>
                         . . .
                      </PinMapping>
                   </Core>
                   <Core Name='Core 1' Type='0' SelectedBank='0'
                         SelectedSignal='-1' NumBanks='2' NumSignals='9'
                         NumPins='9' StateMode='F' TDM_1X='T' ThresholdCode='92'
                         TestBankAvailable='F' ClockPodIndex='2'
                         SelectedForUse='T' CoreIsMaster='T'
                         CoreCannotBeMaster='F' MinimumPeriod='-1'
                         TestBankMode='0'>
                      <Banks>
                         . . .
                      </Banks>
                      <PinMapping Attached='T' ModuleHandle='1'>
                         . . .
                      </PinMapping>
                   </Core>
                </Cores>
             </Device>
          </Devices>
```

<JTAG_Chain> Element

The $\langle JTAG_Chain \rangle$ element describes the number of devices and the number of Altera LAI cores on the JTAG chain.

Attributes

Name	Description
NumATC_II	'number'
NumDevices	'number'

Parents This element can have the following parents: <Properties> (see page 98).

Example <JTAG_Chain NumDevices='3' NumATC_II='2' />

<Label> Element

The <Label> element describes a label within a signal.

Attributes

Name	Description
Bit	'number'
Name	'string'

Parents This element can have the following parents: <Labels> (see page 91).

Example <Label Name='/s2mon/tid' Bit='0' />

<Label> Element (under NonATCLabels)

The <Label> element describes a bus/signal name within the NonATCLabels element.

Attributes

Name	Description
Comment	'string'
Handle	'number'
Name	'string'

Children This element can have the following children: <Assignment> (see page 74).

Parents This element can have the following parents: <NonATCLabels> (see page 92).

Example <NonATCLabels>

<Labels> Element

	The <labels> element contains descriptions of labels within a signal.</labels>
Children	This element can have the following children: <label> (see page 89).</label>
Parents	This element can have the following parents: <signal> (see page 100).</signal>
Example	<labels> <label bit="0" name="/s2mon/tid"></label> </labels>

<NonATCLabels> Element

The <NonATCLabels> element contains a bank's bus/signal names that were not imported from a .cdc file or were renamed (for example, the Calibration Bus or any other renamed or additionally defined bus/signal names).

- Children This element can have the following children: <Label> (see page 90).
- **Parents** This element can have the following parents: <Bank> (see page 76).
- Example <NonATCLabels>

```
<Label Name='My Bus 2' Comment='' Handle='131'>

<Assignment Pod='1' Channel='0' />

<Assignment Pod='1' Channel='1' />

<Assignment Pod='1' Channel='2' />

<Assignment Pod='1' Channel='3' />

<Assignment Pod='1' Channel='4' />

<Assignment Pod='1' Channel='5' />

<Assignment Pod='1' Channel='6' />

<Assignment Pod='1' Channel='6' />

<Assignment Pod='1' Channel='7' />

</Label>

</NonATCLabels>
```

<PinMapping> Element

The <PinMapping> element contains descriptions of pin mapping within an Altera LAI core.

Attributes

Name	Description
Attached	'F' (false) or 'エ' (true)
ModuleHandle	'number'

Children	This element can have the following children: <definedprobes> (see page 84).</definedprobes>
Parents	This element can have the following parents: <core> (see page 81).</core>
Example	<pinmapping attached="T" modulehandle="1"> <definedprobes> <probe name="J1" type="E5346A 34-ch Mictor single-ended probe"> <pods> <pod index="0"></pod> <pod index="1"></pod> </pods></probe></definedprobes></pinmapping>
	<signal <="" name="ATD0" pin="38" pinmapindex="0" th=""></signal>
	ClockChannel='F' /> <signal <br="" name="ATD1" pin="36" pinmapindex="1">ClockChannel='F' /></signal>
	<pre><signal clockchannel="F" name="ATD2" pin="34" pinmapindex="2"></signal></pre>
	<signal <br="" name="ATD3" pin="32" pinmapindex="3">ClockChannel='F' /></signal>
	<signal <br="" name="ATD4" pin="30" pinmapindex="4">ClockChannel='F' /></signal>
	<signal <br="" name="ATD5" pin="28" pinmapindex="5">ClockChannel='F' /></signal>
	<signal <br="" name="ATD6" pin="26" pinmapindex="6">ClockChannel='F' /></signal>
	<signal <br="" name="ATD7" pin="24" pinmapindex="7">ClockChannel='F' /></signal>
	<signal <br="" name="ATD8" pin="22" pinmapindex="8">ClockChannel='F' /></signal>
	<signal <br="" name="ATD9" pin="20" pinmapindex="9">ClockChannel='F' /></signal>
	<signal <br="" name="ATD10" pin="18" pinmapindex="10">ClockChannel='F' /></signal>
	<pre><signal clockchannel="E" name="ATD11" pin="16" pinmapindex="11"></signal></pre>
	<pre><signal clockchannel="F" name="ATD12" pin="14" pinmapindex="12"></signal></pre>
	<pre><signal clockchannel="F" name="ATD13" pin="12" pinmapindex="13"></signal></pre>
	<signal <="" name="ATD14" pin="10" pinmapindex="14" th=""></signal>

```
ClockChannel='F' />
<Signal Name='ATCK' Pin='6' PinMapIndex='16'
ClockChannel='T' />
</Signals>
</Probe>
</DefinedProbes>
</PinMapping>
```

<Pod> Element

The <Pod> element describes the pod index used within a defined probe.

Attributes

Name	Description
Index	'number'

Parents This element can have the following parents: <Pods> (see page 96).

Example <Pod Index='0'/>

<Pods> Element

The <Pods> element contains the pods used by a defined probe.

Children This element can have the following children: <Pod> (see page 95).

Parents This element can have the following parents: <Probe> (see page 97).

```
Example <Pods>
<Pod Index='0'/>
<Pod Index='1'/>
</Pods>
```

<Probe> Element

The <Probe> element describes a defined probe.

Attributes

Name	Description
Name	'string' (name of connector in device under test)
Туре	'string' (name of probe)

- **Children** This element can have the following children: <Pods> (see page 96), <Signals> (see page 104).
- **Parents** This element can have the following parents: <DefinedProbes> (see page 84).

```
Example
         <Probe Name='J1' Type='E5346A 34-ch Mictor single-ended probe'>
            <Pods>
               <Pod Index='0'/>
               <Pod Index='1'/>
            </Pods>
            <Signals>
               <Signal Name='ATD0' Pin='38' PinMapIndex='0' ClockChannel='F' />
               <Signal Name='ATD1' Pin='36' PinMapIndex='1' ClockChannel='F' />
               <Signal Name='ATD2' Pin='34' PinMapIndex='2' ClockChannel='F' />
               <Signal Name='ATD3' Pin='32' PinMapIndex='3' ClockChannel='F' />
               <Signal Name='ATD4' Pin='30' PinMapIndex='4' ClockChannel='F' />
               <Signal Name='ATD5' Pin='28' PinMapIndex='5' ClockChannel='F' />
               <Signal Name='ATD6' Pin='26' PinMapIndex='6' ClockChannel='F' />
               <Signal Name='ATD7' Pin='24' PinMapIndex='7' ClockChannel='F' />
               <Signal Name='ATD8' Pin='22' PinMapIndex='8' ClockChannel='F' />
               <Signal Name='ATD9' Pin='20' PinMapIndex='9' ClockChannel='F' />
               <Signal Name='ATD10' Pin='18' PinMapIndex='10' ClockChannel='F' />
               <Signal Name='ATD11' Pin='16' PinMapIndex='11' ClockChannel='F' />
               <Signal Name='ATD12' Pin='14' PinMapIndex='12' ClockChannel='F' />
               <Signal Name='ATD13' Pin='12' PinMapIndex='13' ClockChannel='F' />
               <Signal Name='ATD14' Pin='10' PinMapIndex='14' ClockChannel='F' />
               <Signal Name='ATCK' Pin='6' PinMapIndex='16' ClockChannel='T' />
            </Signals>
         </Probe>
```

<Properties> Element

The <Properties> element contains setup information for the FPGA dynamic probe.

- **Children** This element can have the following children: <ATC_II> (see page 75), <JTAG_Chain> (see page 88), <Devices> (see page 87).
- **Parents** This element can have the following parents: "<Probe>" (in the online help).

When used in COM automation, this element is returned by the "QueryCommand method" (in the online help)'s GetAllSetup and GetProperties queries. You can also use this element string as an XMLCommand with the "DoCommands method" (in the online help) to configure the FPGA dynamic probe.

```
Example
         <Properties>
             <ATC_II SelectedDevice='1' SelectedCore='1' CableType='0'
                   ParallelCableType='0' ParallelCableSpeed='2'
                   ParallelCablePort='0' USBCableSpeed='0'/>
             <JTAG_Chain NumDevices='2' NumATC_II='2'/>
             <Devices>
                <Device Name='Device 0' Type='0' NumCores='0' UserRegNum='0'</pre>
                      Configurable='F' IRLength='8' CDCFilename=''
                      FPGAFilename='' SelectedForUse='F'>
                   <Cores/>
                </Device>
                <Device Name='Device 1' Type='0' NumCores='2' UserRegNum='1'</pre>
                      Configurable='T' IRLength='6' CDCFilename='C:\Documents
                      and Settings\user\My Documents\Agilent Technologies\Logic
                      Analyzer\Config Files\demo\Altera_demo_V8.cdc'
                      FPGAFilename='C:\Documents and Settings\user\My
                      Documents\Agilent Technologies\Logic Analyzer\Config
                      Files\demo\Altera_demo_V8.bit' SelectedForUse='F'>
                   <Cores>
                      <Core Name='Core 0' Type='0' SelectedBank='0'
                            SelectedSignal='-1' NumBanks='5' NumSignals='16'
                            NumPins='9' StateMode='T' TDM_1X='F'
                            ThresholdCode='92' TestBankAvailable='T'
                            ClockPodIndex='0' SelectedForUse='T'
                            CoreIsMaster='F' CoreCannotBeMaster='F'
                            MinimumPeriod='-1' TestBankMode='2' Latency='4'>
                         <Banks>
                            . . .
                         </Banks>
                         <PinMapping Attached='T' ModuleHandle='1'>
                            . . .
                         </PinMapping>
                      </Core>
                      <Core Name='Core 1' Type='0' SelectedBank='0'
                            SelectedSignal='-1' NumBanks='2' NumSignals='9'
                            NumPins='9' StateMode='F' TDM 1X='T'
                            ThresholdCode='92' TestBankAvailable='F'
```

```
ClockPodIndex='2' SelectedForUse='T' CoreIsMaster='T'
CoreCannotBeMaster='F' MinimumPeriod='-1'
TestBankMode='0'>
<Banks>
...
</Banks>
<PinMapping Attached='T' ModuleHandle='1'>
...
</PinMapping>
</Core>
</Core>
</Device>
</Device>
</Properties>
```

<Signal> Element (under Bank)

The <Signal> element describes a signal within a bank.

```
Children This element can have the following children: <Labels> (see page 91).
```

Parents This element can have the following parents: <Signals> (see page 102).

<Signal> Element (under Probe)

The <Signal> element describes a signal within a defined probe.

Attributes

Name	Description
ClockChannel	'F' (false) or 'エ' (true)
Name	'string'
PinMapIndex	'number'

Parents This element can have the following parents: <Signals> (see page 104).

Example <Signal Name='ATD0' Pin='38' PinMapIndex='0' ClockChannel='F' />

<Signals> Element (under Bank)

The <Signals> element contains descriptions of signals within a bank.

- **Children** This element can have the following children: <Signal> (see page 100).
- **Parents** This element can have the following parents: <Bank> (see page 76).

```
Example
         <Signals>
             <Signal>
                <Labels>
                   <Label Name='/s2mon/tid' Bit='0' />
                </Labels>
             </Signal>
             <Signal>
                <Labels>
                   <Label Name='/s2mon/tid' Bit='1' />
                </Labels>
             </Signal>
             <Signal>
                <Labels>
                   <Label Name='/s2mon/tid' Bit='2' />
                </Labels>
             </Signal>
             <Signal>
                <Labels>
                   <Label Name='/s2mon/tid' Bit='3' />
               </Labels>
             </Signal>
             <Signal>
                <Labels>
                   <Label Name='/s2mon/tid' Bit='4' />
                </Labels>
             </Signal>
             <Signal>
                <Labels>
                   <Label Name='/s2mon/tid' Bit='5' />
                </Labels>
             </Signal>
             <Signal>
                <Labels>
                   <Label Name='/s2mon/tid' Bit='6' />
                </Labels>
             </Signal>
             <Signal>
                <Labels>
                   <Label Name='/s2mon/tid' Bit='7' />
                </Labels>
             </Signal>
             <Signal>
                <Labels>
                   <Label Name='/s2mon/lastackid' Bit='0' />
               </Labels>
             </Signal>
```

```
<Signal>
      <Labels>
         <Label Name='/s2mon/lastackid' Bit='1' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
         <Label Name='/s2mon/lastackid' Bit='2' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
         <Label Name='/s2mon/lastackid' Bit='3' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
         <Label Name='/s2mon/lastackid' Bit='4' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
         <Label Name='/s2mon/lastackid' Bit='5' />
      </Labels>
   </Signal>
   <Signal>
      <Labels>
         <Label Name='/s2mon/lastackid' Bit='6' />
      </Labels>
   </Signal>
   <Signal>
      <Labels />
   </Signal>
</Signals>
```

<Signals> Element (under Probe)

The <Signals> element contains the signals used by a defined probe.

- **Children** This element can have the following children: <Signal> (see page 101).
- Parents This element can have the following parents: <Probe> (see page 97).

```
Example
         <Signals>
            <Signal Name='ATD0' Pin='38' PinMapIndex='0' ClockChannel='F' />
            <Signal Name='ATD1' Pin='36' PinMapIndex='1' ClockChannel='F' />
            <Signal Name='ATD2' Pin='34' PinMapIndex='2' ClockChannel='F' />
            <Signal Name='ATD3' Pin='32' PinMapIndex='3' ClockChannel='F' />
            <Signal Name='ATD4' Pin='30' PinMapIndex='4' ClockChannel='F' />
            <Signal Name='ATD5' Pin='28' PinMapIndex='5' ClockChannel='F' />
            <Signal Name='ATD6' Pin='26' PinMapIndex='6' ClockChannel='F' />
            <Signal Name='ATD7' Pin='24' PinMapIndex='7' ClockChannel='F' />
            <Signal Name='ATD8' Pin='22' PinMapIndex='8' ClockChannel='F' />
            <Signal Name='ATD9' Pin='20' PinMapIndex='9' ClockChannel='F' />
            <Signal Name='ATD10' Pin='18' PinMapIndex='10' ClockChannel='F' />
            <Signal Name='ATD11' Pin='16' PinMapIndex='11' ClockChannel='F' />
            <Signal Name='ATD12' Pin='14' PinMapIndex='12' ClockChannel='F' />
            <Signal Name='ATD13' Pin='12' PinMapIndex='13' ClockChannel='F' />
            <Signal Name='ATD14' Pin='10' PinMapIndex='14' ClockChannel='F' />
            <Signal Name='ATCK' Pin='6' PinMapIndex='16' ClockChannel='T' />
         </Signals>
```

<SymbolInfo> Element

The <SymbolInfo> elements contain the module bus/signal symbol settings associated with a particular bank.

Attributes

Name	Description
ModuleHandle	'number'
SymbolSettings	'string'

Parents This element can have the following parents: <Bank> (see page 76).

```
Example
        <SymbolInfo ModuleHandle='1' SymbolSettings='
        <Module&gt;
           <BusSignalSetup&gt;
             <BusSignals&gt;
                <BusSignal Name=&apos;My Bus 1&apos;/&gt;
                <Folder Name=&apos;Core 0 FPGA Probe&apos;
                     Comment='Created by FPGA Dynamic Probe-1'>
                  <BusSignal Name=&apos;/s2mon/tid&apos;/&gt;
                  <BusSignal Name=&apos;/s2mon/s2mstate&apos;/&gt;
                </Folder&gt;
             </BusSignals&gt;
             <NetlistImport/&gt;
           </BusSignalSetup&gt;
           <Config TimeOfTrigger=&apos;1.110240661 Gs&apos;
                CorrelatedTriggerTime='0 s'
                UserSkewTime='0 s' SystemTrigger='T'/>
        </Module&gt;
        '/>
```

See Also • "<Module> Element (under Configuration Setup)" (in the online help)

<TriggerInfo> Element

The <TriggerInfo> elements contain the module trigger settings associated with a particular bank.

Attributes

Parents

Name	Description
ModuleHandle	'number'
TriggerSettings	'string'

This element can have the following parents: <Bank> (see page 76).

Example <TriggerInfo ModuleHandle='1' TriggerSettings=' <Module> <Trigger Mode='State' Type='Normal'> <StoreQual Mode='Custom'> <Event ParensNeeded='F'> <Anything/> </Event> </StoreQual> <Step Number='1'> <If> <Event ParensNeeded='F'> <BusSignal Name='/s2mon/tid' Bit='All' Operator='Equals' Value='h3F'/> </Event> <Occurrence Value='1' Mode='Eventual'/> <Action> <TriggerAction Operator='Fill Memory'> <StoreQual Mode='Custom'> <Event ParensNeeded='F'> <DefaultStore/> </Event> </StoreQual> </TriggerAction> </Action> </If> </Step> </Trigger> <Config TimeOfTrigger='1.110240661 Gs' CorrelatedTriggerTime='0 s' UserSkewTime='0 s' SystemTrigger='T'/> </Module> '/>

See Also •

• "<Module> Element (under Configuration Setup)" (in the online help)

<WindowInfo> Element

The <WindowInfo> elements contain the display window settings associated with a particular bank.

Attributes

Name	Description
WindowHandle	'number'
WindowSettings	'string'

Parents	This element can have the following parents: <bank> (see page 76).</bank>
Example	<pre><windowinfo <br="" defaultbase="Hex" s2mon="" tid'="" windowhandle="1" windowsettings=" <Setup> <Sampling PerDivision='5 ns' Delay='0 s'/> <BusSignals> <Clear/> <BusSignal Module='My 1682D-1' Name='My Bus 1' DefaultBase='Hex' Color='hFFFFF' Height='30'/></pre></th></tr><tr><th></th><th><pre><BusSignal Module='My 1682D-1'
Name=">Color='hFFFFFF' Height='30'/> <bussignal <br="" module="My 1682D-1">Name='/s2mon/s2mstate' DefaultBase='Hex' Color='hFFFFFF' Height='30'/></bussignal></windowinfo></pre>
	<pre><BusSignal Name='Time' Color='hFFFFFF' Height='30'/> '/></pre>
	<pre></pre> </th
	Name='/s2mon/s2mstate' DefaultBase='Hex' Color='hFFFFF' Alignment='Right' Width='120'/>

```
</BusSignals&gt;
'/>
```

See Also • "<Window> Element (under Configuration Setup)" (in the online help)
Index

Symbols

.bit FPGA configuration file, **64** .lai FPGA signal import file, **64**

A

activity not on FPGA dynamic probe, 44 Altera FPGA dynamic probe, 3 Altera JTAG cable programmer software, 18 Altera LAI core, selecting, 28 Altera LAI core, selecting signal banks, 39 Altera LAI cores, enabling/disabling, 60 Altera software, problem locating, 42 Assignment, XML element, 74 ATC_II, XML element, 75 automated logic analyzer set up (FPGA dynamic probe), 58

B

B4656A FPGA dynamic probe for Altera FPGAs, 13 Bank, XML element, 76 banks (Altera LAI core), selecting, 39 Banks, XML element, 80 bus/signal names, adding FPGA, 33 bus/signal names, renaming FPGA, 32 bus/signal names, trimming FPGA, 32

C

Cable Connection dialog, 61 cable connection, selecting type of, 28 captured data, invalidation of, 39 client, JTAG, 21 Clk pin/pad locations, 35 clock activity indicators, 45 COM automation, FPGA dynamic probe, 71 concepts, FPGA dynamic probe for Altera FPGAs, 47 configuration file, FPGA, 64 configure FPGA device, 30 connection to Altera LAI core, establishing, 28 Core Details dialog, 65 core name, 65 Core, XML element, 81 cores (Altera LAI), enabling/disabling, 60 cores, multiple, mapping to two halves of the same probe, 35 Cores, XML element, 83

D

data, invalidation of, 39 debug pins (FPGA), mapping to probes and pods, 35 debug pins (FPGA), probing, 11 debug pins, FPGA, 62 DefinedProbes, XML element, 84 demo mode, FPGA dynamic probe, 61 design (FPGA) tools, 69 design steps for FPGA dynamic probe, 15 device name, 65 Device, XML element, 85 Devices, XML element, 87 disabling/enabling Altera LAI cores, 60

Ε

enabling/disabling Altera LAI cores, 60 error, script, 43

F

firewall, Quartus II Programmer port number, 24 FPGA bus/signal names, adding, 33 FPGA configuration file, 64 FPGA debug pins. 62 FPGA debug pins, mapping to probes and pods, 35 FPGA debug pins, probing, 11 FPGA Dynamic Probe Bank Selection dialog, 67 FPGA dynamic probe characteristics, 69 FPGA Dynamic Probe Setup dialog, 60 FPGA dynamic probe specifications, 69 FPGA dynamic probe, automated logic analyzer set up. 58 FPGA dynamic probe, design steps, 15 FPGA dynamic probe, installing and licensing, 13 FPGA dynamic probe, measurement steps, 27 FPGA dynamic probe, no activity, 44 FPGA dynamic probe, state mode measurements, 45 FPGA imported bus/signal names, renaming, 32 FPGA imported bus/signal names, trimming, 32 FPGA Probe, bus/signal name folder, 33 FPGA signal names, importing from LAI file, 31 FPGA, downloading configuration bits, 30

Η

hostname of JTAG server, 25

imported FPGA bus/signal names, renaming, 32
imported FPGA bus/signal names, trimming, 32
importing FPGA signal names from LAI file, 31
installing the FPGA dynamic probe, 13
invalidation of captured data, 39

J

JTAG cable programmer software, Altera, 18 JTAG cable, establishing a connection, 28 JTAG cable, setting up, 19 JTAG scan chain, devices on, 28 JTAG scan chain, maximum number of devices, 69 JTAG server, 21 JTAG_Chain, XML element, 88

L

Label, XML element, 89 Label, XML element (under NonATCLabels), 90 Labels, XML element, 91 LAI core, compiling the project (example), 56 LAI core, configuring parameters (example), 49 LAI core, creating (example), 48 LAI core, enabling/disabling (example), 55 LAI core, inserting (example), 48 LAI core, mapping internal signals to banks (example), 53 LAI core, mapping the clock signal (example), 50 LAI core, mapping to I/O pins (example), 51 LAI core, multiple instances (example), 57 LAI core, saving (example), 55 LAI file, importing FPGA signal names from, 31 licensing the FPGA dynamic probe, 13 local JTAG server, 21 logic analyzer probing of FPGA debug pins, 11 logic analyzer set up, automated by FPGA dynamic probe, 58

Μ

mapping FPGA debug pins to probes and pods, 35

Index

mapping multiple cores to two halves of the same probe, 35
measurement steps for FPGA dynamic probe, 27
measurements on FPGA internal signals, 39
multiple cores, mapping to two halves of the same probe, 35

Ν

new (FPGA dynamic) probe, 28 no activity on FPGA dynamic probe, 44 NonATCLabels, XML element, 92 notices, 2

0

On column in FPGA Dynamic Probe Setup dialog, 60 overview, FPGA dynamic probe, 9

P

Pin Mapping dialog, 61 Pin Mapping Edit dialog, 62 PinMapping, XML element, 93 Pod, XML element, 95 pods, logic analyzer, 62 pods, mapping FPGA debug pins to, 35 Pods, XML element, 96 preparation steps, 17 probe, mapping multiple cores to two halves of the same, 35 Probe, XML element, 97 Probes toolbar, 39 probes, adding/editing, 61 probes, mapping FPGA debug pins to, 35 probing of FPGA debug pins, 11 Programmer software, Altera Quartus II, 18 Properties dialog, 65 Properties, XML element, 98

0

Quartus II Programmer software, Altera, 18

R

reference designator, 62 reference, FPGA dynamic probe, 59 remote JTAG server, 21 rename individual signals, 35, 67 renaming imported FPGA bus/signal names, 32

S

script error, 43 Select FPGA Configuration File dialog, 64 Select Signal Import File dialog, 64 server, JTAG, 21 signal banks (Altera LAI core), selecting, 39 signal import file, FPGA, 64 signal names (FPGA), importing from LAI file, 31 Signal, XML element, 100, 101 signal/bus names, adding FPGA, 33 signal/bus names, renaming FPGA, 32 signal/bus names, trimming FPGA, 32 Signals, XML element, 102, 104 state mode measurements in FPGA dynamic probe, 45 SymbolInfo, XML element, 105

Т

timing zoom disabled by FPGA dynamic probe, **39** tool tip, mapping FPGA debug pins, trademarks, TriggerInfo, XML element, Trim Bus/Signal Names dialog, trimming imported FPGA bus/signal names, triple-click to rename individual signals, **31**, troubleshooting, FPGA dynamic probe,

U

USB Blaster device driver, 20

W

WindowInfo, XML element, 107

X

XML format, FPGA dynamic probe, 73