

# **Xilinx 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 03.82.0000

## Edition

April 10, 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 sub-contract, 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 Xilinx FPGA Dynamic Probe

When an Agilent ATC2 or MTC trace core has been inserted into an FPGA, the FPGA dynamic probe lets a logic analyzer capture data from signals inside the FPGA.

<b>ATC2</b>	The Agilent Trace Core (see <a href="#">page 9</a> ) lets you view a large number of internal FPGA signals using a small number of pins while consuming minimal FPGA resources (just over 1 slice required per input signal to ATC2). With ATC2 cores, the FPGA dynamic probe lets you change probe points without recompiling or affecting the timing of the design.
<b>MTC</b>	The MicroBlaze Trace Core (see <a href="#">page 11</a> ) lets you capture and decode the execution of a MicroBlaze soft processor core inside an FPGA. With 2x pin compression and the ability to look at just the significant signals, you can debug using fewer pins. The E9524A MicroBlaze trace toolset provides inverse assembly mnemonics in the Listing window, and you can view the high-level source code associated with captured execution in the Source window.

With either core, the FPGA dynamic probe's automatic pin-mapping and ability to import internal signal names from the FPGA design tools makes setting up the logic analyzer easy.

- Probing FPGA Debug Pins (see [page 13](#))
- Installing and Licensing the FPGA Dynamic Probe (see [page 15](#))
- ATC2 Design Steps (see [page 17](#))
  - ATC2 Design Step 1. Determine the ATC2 core parameters (see [page 18](#))
  - ATC2 Design Step 2. Create the ATC2 core (see [page 21](#))
  - ATC2 Design Step 3. Select groups of signals to probe (see [page 22](#))
- MTC Design Steps (see [page 23](#))
  - MTC Design Step 1. Create and instantiate an MTC core (see [page 24](#))
  - MTC Design Step 2. Implement your Xilinx FPGA design with MicroBlaze and MTC cores in place (see [page 25](#))
- Measurement Steps (see [page 27](#))
  - Measurement Step 1. Establish connection between analyzer and ATC2 or MTC core (see [page 28](#))
  - Measurement Step 2. Download configuration bits into FPGA (see [page 31](#))
  - Measurement Step 3. Map FPGA pins (see [page 32](#))
  - Measurement Step 4. Import signal names (see [page 37](#))
  - Measurement Step 5. Adjust sampling positions for state (synchronous) cores (see [page 41](#))

- Measurement Step 6. Make the measurement (see [page 44](#))
- Troubleshooting (see [page 49](#))
- Concepts (see [page 55](#))
- Reference (see [page 63](#))
  - FPGA Dynamic Probe Setup Dialog (see [page 64](#))
  - FPGA Dynamic Probe Bank Selection Dialog (see [page 71](#))
  - Specifications and Characteristics (see [page 73](#))

- See Also**
- Probe Control, COM Automation (see [page 77](#))
  - Probe Setup, XML Format (see [page 79](#))

# Contents

Using the Xilinx FPGA Dynamic Probe 3

## 1 Agilent Trace Core (ATC2) Overview

## 2 MicroBlaze Trace Core (MTC) Overview

## 3 Probing FPGA Debug Pins

## 4 Installing and Licensing the FPGA Dynamic Probe

## 5 ATC2 Design Steps

- ATC2 Design Step 1. Determine ATC2 core parameters 18
  - State (Synchronous) vs. Timing (Asynchronous) Cores 18
  - Number of FPGA Debug Pins, Number of Banks 19
  - FPGA Resource Consumption 20
- ATC2 Design Step 2. Create the ATC2 core 21
- ATC2 Design Step 3. Select groups of signals to probe 22

## 6 MTC Design Steps

- MTC Design Step 1. Create and instantiate an MTC core 24
- MTC Design Step 2. Implement your Xilinx FPGA design with MicroBlaze and MTC cores in place 25

## 7 Measurement Steps

- Measurement Step 1. Establish connection between analyzer and ATC2 or MTC core 28
- Measurement Step 2. Download configuration bits into FPGA 31
- Measurement Step 3. Map FPGA pins 32
  - To map pins for cores with auto pin-mapping 32
  - To map pins for cores without auto pin-mapping 34
- Measurement Step 4. Import signal names 37
  - To trim imported bus/signal names 38
  - To rename imported bus/signal names 38
  - To define additional FPGA bus/signal names 39
- Measurement Step 5. Adjust sampling positions for state (synchronous) cores 41

Measurement Step 6. Make the measurement	44
Making Measurements with ATC2 Cores	44
Making Measurements with MTC Cores	46

## 8 FPGA Dynamic Probe Troubleshooting

If you don't see activity in the logic analyzer	50
If state mode measurements don't work	51
If the ATC2 or MTC core isn't present in an FPGA after downloading configuration bits	52
If you get the "Can not open cable" error message	53

## 9 FPGA Dynamic Probe Concepts

How do ATC2 cores affect design timing and signal routing?	56
How do ATC2 cores affect FPGA performance?	57
Can I put multiple ATC2 cores in a single device?	58
Should I leave the ATC2 core in my design after validation is complete?	59
How does the optional 2X pin compression technology work?	60
Automated Logic Analyzer Set Up	61
How do I define buses/signals created in FPGA Editor?	62

## 10 FPGA Dynamic Probe Reference

FPGA Dynamic Probe Setup Dialog	64
Cable Connection Dialog	65
Pin Mapping Dialog	66
Pin Mapping Edit Dialog	67
Select FPGA Configuration File Dialog	68
Select Signal Import File Dialog	69
Properties Dialog	69
Core Details Dialog	70
FPGA Dynamic Probe Bank Selection Dialog	71
Trim Bus/Signal Names Dialog	71
FPGA Dynamic Probe Specifications and Characteristics	73

## 11 Probe Control, COM Automation

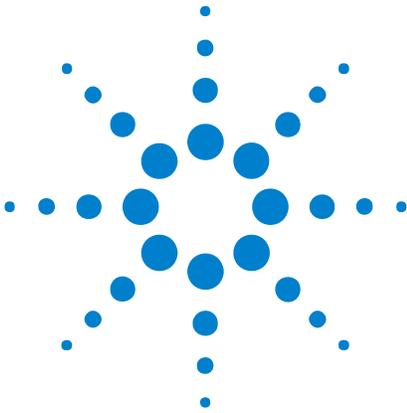
## 12 Probe Setup, XML Format

<Assignment> Element	80
<ATC_II> Element	81

<Bank> Element	82
<Banks> Element	86
<Cable> Element	87
<Core> Element	88
<Cores> Element	90
<DefinedProbes> Element	91
<Device> Element	92
<Devices> Element	94
<JTAG_Chain> Element	96
<Label> Element	97
<Label> Element (under NonATCLabels)	98
<Labels> Element	99
<NonATCLabels> Element	100
<PinMapping> Element	101
<Pod> Element	103
<Pods> Element	104
<Probe> Element	105
<Properties> Element	106
<Signal> Element (under Bank)	108
<Signal> Element (under Probe)	109
<Signals> Element (under Bank)	110
<Signals> Element (under Probe)	112
<SymbolInfo> Element	113
<TriggerInfo> Element	114
<WindowInfo> Element	115

## Index





# 1 Agilent Trace Core (ATC2) Overview

When used with Agilent ATC2 cores, the FPGA dynamic probe lets you:

- *View internal activity.*

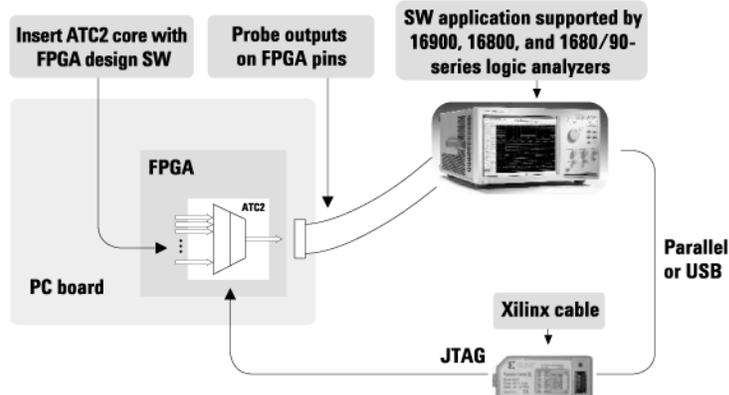
With a logic analyzer, you are normally limited to measuring signals at the periphery of the FPGA. With the FPGA dynamic probe, you can now access signals internal to the FPGA. You can measure up to 64 internal signals for each external pin dedicated to debug, unlocking visibility into your design that 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.

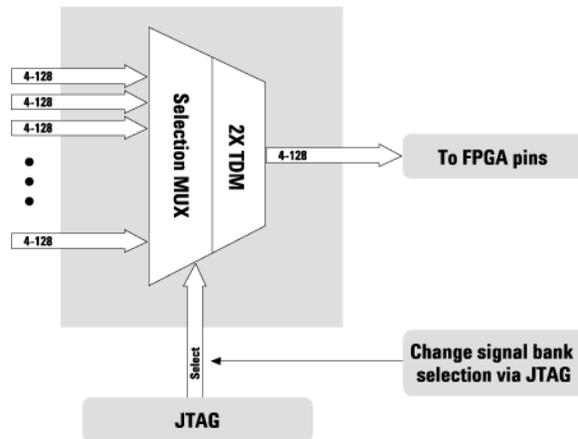
- *Leverage the work you did in your design environment.*

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.

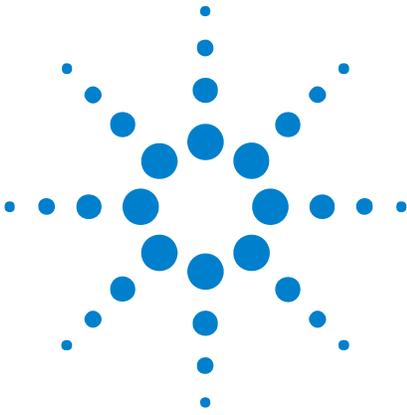


Create a time saving FPGA measurement system. Insert an ATC2 (Agilent Trace Core) core into your FPGA design. With the application running on your logic analyzer via JTAG you control which group of internal signals to measure.

## 1 Agilent Trace Core (ATC2) Overview



Access 64 internal signals for each debug pin. Select cores with 1, 2, 4, 8, 16, or 32 signal banks. Signal banks all have identical width (4 to 128 signals wide) determined by the number of pins you devote for debug. Each pin provides sequential access to 1 signal on every input bank. Using an optional 2X time division compression in state mode, each pin can simultaneously access 2 signals per bank.



## 2 MicroBlaze Trace Core (MTC) Overview

When used with Agilent MTC cores, the FPGA dynamic probe lets you:

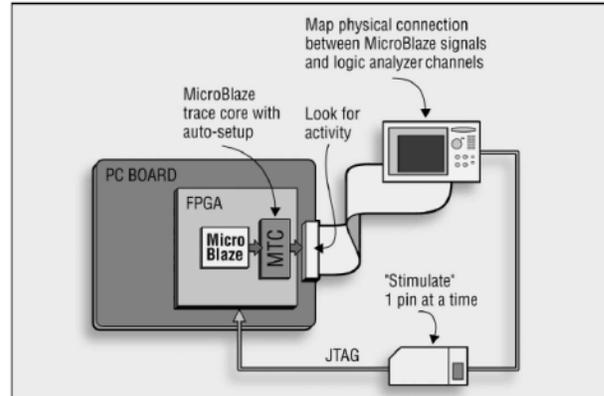
- *View internal MicroBlaze signals with minimal impact on the timing of your design.*

MTC cores are designed to be very small in terms of device resource consumption. An MTC core in a XC2V3000 device consumes roughly 1% of the LUTs and flops.

- *Use 2x pin compression to reduce the number of FPGA pins required.*

How does the optional 2X pin compression technology work? (see [page 60](#))

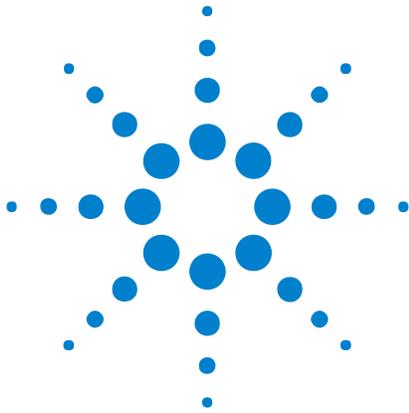
- *Use automatic pin-mapping to reduce logic analyzer set up time.*



Agilent's MTC reduces the set up time for an initial trace measurement. You can literally connect a logic analyzer to a connector with MTC core outputs routed to it, and within seconds, the logic analyzer becomes ready to take a MicroBlaze trace measurement.



## 2 MicroBlaze Trace Core (MTC) Overview



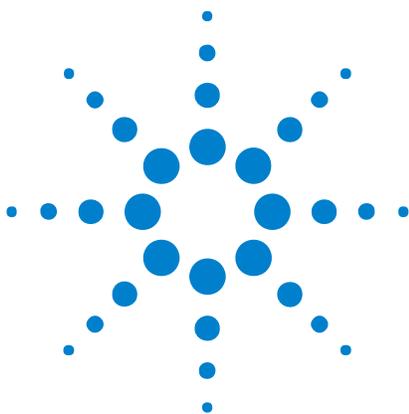
## 3 Probing FPGA Debug Pins

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).



### **3 Probing FPGA Debug Pins**



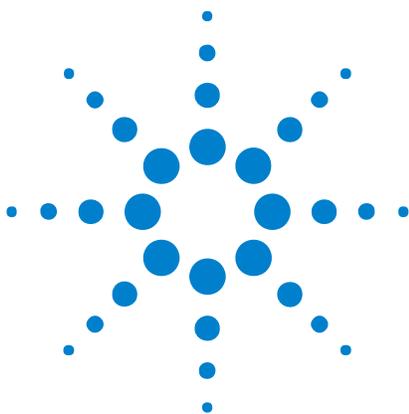
## 4 Installing and Licensing the FPGA Dynamic Probe

Before you can use the Agilent B4655A Xilinx FPGA dynamic probe or E9524A MicroBlaze trace toolset, you must install and license the software:

- 1 Install the FPGA dynamic probe software from the application software CD or from the Agilent web site at:  
["http://www.agilent.com/find/la-sw-download"](http://www.agilent.com/find/la-sw-download)
- 2 Follow the instructions on your entitlement certificate to redeem and install the software licenses.



## **4 Installing and Licensing the FPGA Dynamic Probe**



## 5 ATC2 Design Steps

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

- 1 Determine the ATC2 core parameters (see [page 18](#))
  - State (Synchronous) vs. Timing (Asynchronous) Cores (see [page 18](#))
  - Number of FPGA Debug Pins, Number of Banks (see [page 19](#))
  - FPGA Resource Consumption (see [page 20](#))
- 2 Create the ATC2 core (see [page 21](#))
- 3 Select groups of signals to probe (see [page 22](#))

- See Also**
- For more ATC2 design information, see: "<http://www.agilent.com/find/fpga>"
  - How do ATC2 cores affect design timing and signal routing? (see [page 56](#))
  - How do ATC2 cores affect FPGA performance? (see [page 57](#))
  - Can I put multiple ATC2 cores in a single device? (see [page 58](#))
  - Measurement Steps (see [page 27](#))



## ATC2 Design Step 1. Determine ATC2 core parameters

Before you use Xilinx ChipScope Pro to insert the ATC2 core and select groups of signals to probe, you need to determine the ATC2 core parameters you will use.

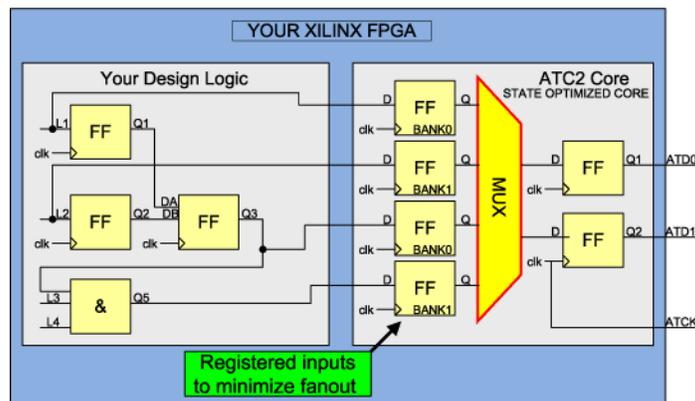
- State (Synchronous) vs. Timing (Asynchronous) Cores (see [page 18](#))
- Number of FPGA Debug Pins, Number of Banks (see [page 19](#))
- FPGA Resource Consumption (see [page 20](#))

**Next** • ATC2 Design Step 2. Create the ATC2 core (see [page 21](#))

### State (Synchronous) vs. Timing (Asynchronous) Cores

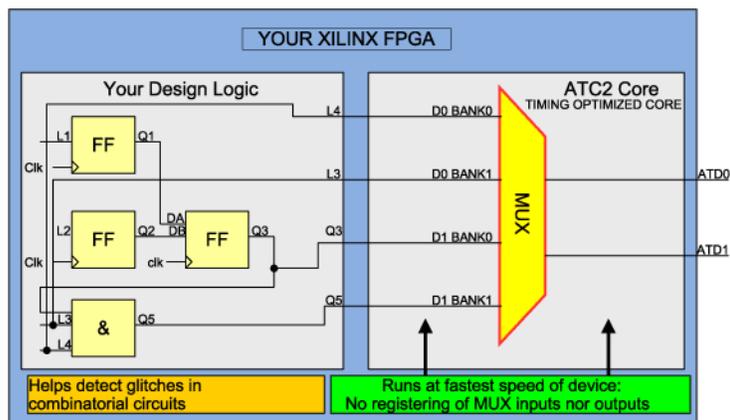
State cores:

- Provide most-accurate measurements for functional debug in one time domain.
- Register inputs to minimize fan out.
- Support time division multiplexing for 2X pin reduction (see [page 60](#)).
- Support calibration for more accurate measurements on buses with narrow data valid windows.



Timing cores:

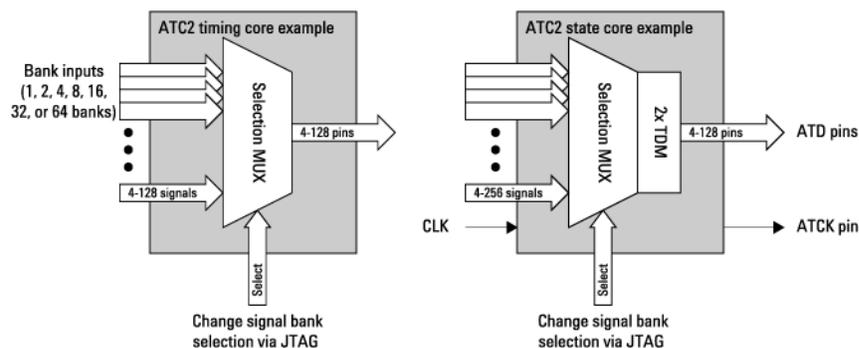
- Are best for measurements across multiple time domains.
- Run at fastest-possible speed of device, no registering of MUX inputs or outputs.
- Have minimal impact on design timing.



The FPGA dynamic probe will detect the type of ATC2 core and automatically set up the appropriate logic analyzer sampling mode (see page 61).

### Number of FPGA Debug Pins, Number of Banks

Using the FPGA dynamic probe, each pin provides access to up to 64 internal signals. The number of debug pins can range from 4 to 128 depending on your needs. When using state (synchronous) cores, one additional pin is used for the clock.



Number of debug pins <sup>1</sup> (4-128 in increments of 1)	Maximum number of internal FPGA signals that can be probed (w/2x TDM <sup>2</sup> ) per number of banks						
	1 bank	2 banks	4 banks	8 banks	16 banks	32 banks	64 banks
4	4 (8)	8 (16)	16 (32)	32 (64)	64 (128)	128 (256)	256 (512)
8	8 (16)	16 (32)	32 (64)	64 (128)	128 (256)	256 (512)	512 (1024)

16	16 (32)	32 (64)	64 (128)	128 (256)	256 (512)	512 (1024)	1024 (2048)
32	32 (64)	64 (128)	128 (256)	256 (512)	512 (1024)	1024 (2048)	2048 (4096)
64	64 (128)	128 (256)	256 (512)	512 (1024)	1024 (2048)	2048 (4096)	4096 (8192)
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
128	128 (256)	256 (512)	512 (1024)	1024 (2048)	2048 (4096)	4096 (8192)	8192 (16384)
<sup>1</sup> When using state (synchronous) cores, one extra pin is required for the clock. <sup>2</sup> The maximum number of signals per bank doubles when 2x TDM is selected. 2x TDM is only available for state cores.							

### FPGA Resource Consumption

Each input signal adds roughly 1 slice to the size of the ATC2 core. The ATC2 cores have been designed to be as small as possible. For example, an ATC2 state core configured with 8 signal banks and 80 signals per bank consumes about 94 slices, or less than 1 percent of the resources on a Xilinx XCV2000 device.

The actual core size depends on the parameters chosen, such as:

- 1 Core type: state, state with pin compression or timing.
- 2 Number of pins.
- 3 Number of signal banks.

Xilinx recommends measuring core sizes in terms of flops and LUTs because this is what the place and route tools work with. A calculator to determine LUT and flop resource utilization from the combination of the ATC2 core and ICON (JTAG controller) is available at "<http://www.agilent.com/find/fpga>" or in the Xilinx Core Inserter tool.

**Why do I see flip-flops in the timing core?**

The path from the probed signal to the output pad is not registered in the timing core. This path has only a combinatorial mux connected to output pads. The flip-flops in the timing core are used only for core control and status. An example of a core control signal is the bank mux select. These control and status flip-flops are registered by the JTAG TCK clock, not a design clock. Therefore, these flip-flops are part of the low-speed circuit used by the logic analyzer to control and poll the core.

## ATC2 Design Step 2. Create the ATC2 core

Use Xilinx ChipScope Pro or EDK (Embedded Development Kit) to create the ATC2 core and to merge it with your design. Using either of these tools, you can specify the parameters of the ATC2 core (number of pins, number of signal banks, state or timing measurement, and other ATC2 attributes), and you can specify which design signals go to the ATC2, making them available for real-time measurement.

ChipScope Pro includes Core Generator and Core Inserter.

**Core Inserter** Xilinx Core Inserter puts the core into your FPGA design post synthesis.

Core Inserter produces a .cdc file. This is a small file listing the signal inputs to the ATC2 core. This file is used to automatically synchronize design signal names with logic analysis bus and signal names.

Agilent recommends using Core Inserter so you can take advantage of signal-name mapping.

**Core Generator or EDK** If you use Core Generator or EDK, the tool instantiates your parameterized ATC2 as a black-box Verilog or VHDL unit. The synthesis tool puts the instantiated core into your design during the synthesis process. ATC2 cores produced by Core Generator are compatible with these synthesis tools:

- Exemplar Leonardo Spectrum
- Synopsys Design Compiler
- Synopsys Design Compiler II
- Synopsys FPGA Express
- Synplicity Synplify
- Xilinx XST 4

Xilinx has a stimulus core known as VIO. This core can only be created and placed in a design using Xilinx Core Generator. For a single design that contains both a VIO core and an ATC2, Core Generator must be used.

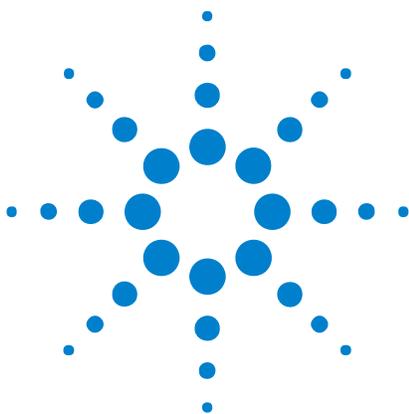
The Core Generator or EDK tools make a .cdc file that you can edit so that the bank input signal names reflect the signal path of the probed net. Otherwise, if no .cdc file is loaded, you can manually rename ATC signal names with the signal names from your design—the FPGA dynamic probe will remember these name changes.

**Next** • ATC2 Design Step 3. Select groups of signals to probe (see [page 22](#))

## ATC2 Design Step 3. Select groups of signals to probe

Specify banks of internal signals that are potential candidates for logic analysis measurements (using Xilinx Core Inserter or Xilinx embedded development kit (EDK)).

- Next**
- Measurement Step 1. Establish connection between analyzer and ATC2 core (see [page 28](#))



## 6 MTC Design Steps

Before you can use the FPGA dynamic probe software (and the *Agilent Logic Analyzer* application) with MTC cores, you must take these design steps:

- 1 Create and instantiate an MTC core (see [page 24](#))
- 2 Implement your Xilinx FPGA design with MicroBlaze and MTC cores in place (see [page 25](#))

- See Also**
- For more MTC design information, see: ["http://www.agilent.com/find/microblaze"](http://www.agilent.com/find/microblaze)
  - Measurement Steps (see [page 27](#))



## MTC Design Step 1. Create and instantiate an MTC core

Use Xilinx Platform Studio to select the MTC core and specify the parameters that best match your design needs. Parameters include: data compression, status signals, and location.

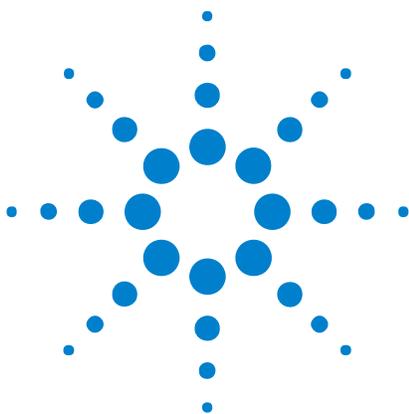
- Next**
- MTC Design Step 2. Implement your Xilinx FPGA design with MicroBlaze and MTC cores in place (see [page 25](#))

## MTC Design Step 2. Implement your Xilinx FPGA design with MicroBlaze and MTC cores in place

Once the design is ready for prototyping in the FPGA, use the Xilinx tools to implement the design (generate the bitstream) including the MicroBlaze and MTC cores. You are then ready to set up the logic analyzer for measurement.

- Next**
- Measurement Step 1. Establish connection between analyzer and ATC2 core (see [page 28](#))

## 6 MTC Design Steps



## 7 Measurement Steps

After you have completed the Design Steps (see [page 17](#)) of inserting the ATC2 core and selecting groups of signals to probe, you are ready to take these measurement steps in the *Agilent Logic Analyzer* application:

- 1 Establish connection between analyzer and ATC2 core (see [page 28](#))
- 2 Download configuration bits into FPGA (see [page 31](#))
- 3 Map FPGA pins (see [page 32](#))
- 4 Import signal names (see [page 37](#))
- 5 Adjust sampling positions for state (synchronous) cores (see [page 41](#))
- 6 Make the measurement (see [page 44](#))
  - Making Measurements with ATC2 Cores (see [page 44](#))
  - Making Measurements with MTC Cores (see [page 46](#))

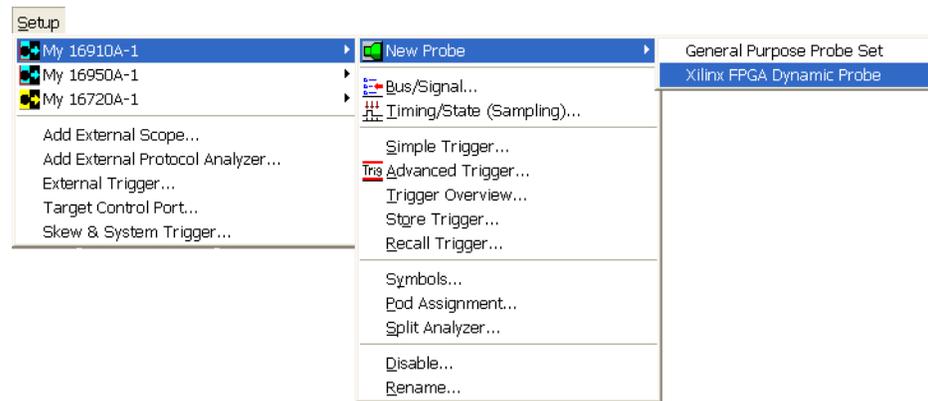


## Measurement Step 1. Establish connection between analyzer and ATC2 or MTC core

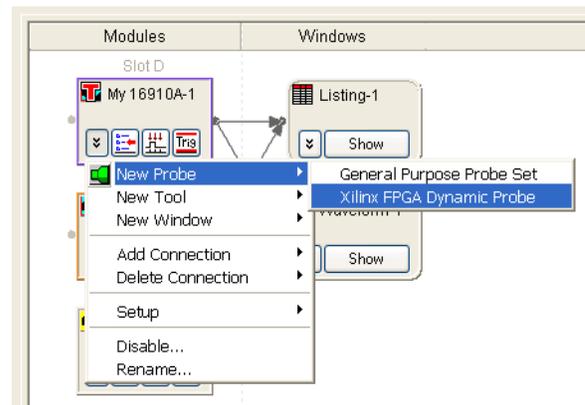
The FPGA dynamic probe application establishes a connection between the logic analyzer and a Xilinx cable. It also determines what devices are on the JTAG scan chain and lets you pick which one you wish to communicate.

To establish a connection between the logic analyzer and the ATC2 or MTC core:

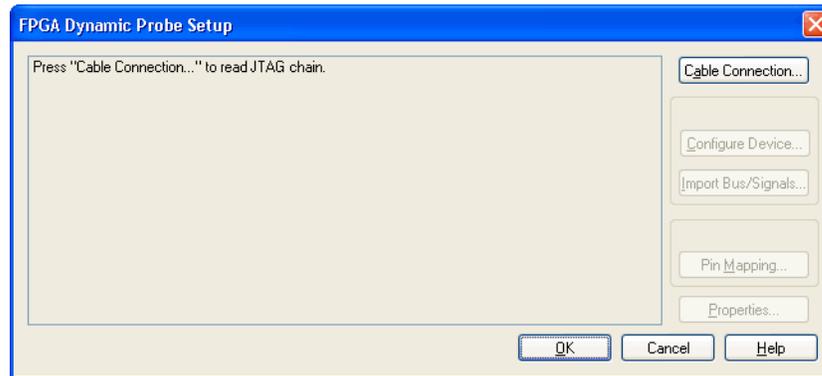
- 1 Add a new FPGA Dynamic Probe set by choosing **Setup>(Logic Analyzer Module);>New Probe>Xilinx FPGA Dynamic Probe.**



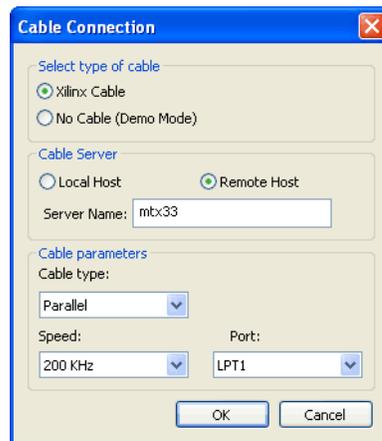
Or, in the Overview window, from a module's drop-down menu, choose **New Probe>Xilinx FPGA Dynamic Probe.**



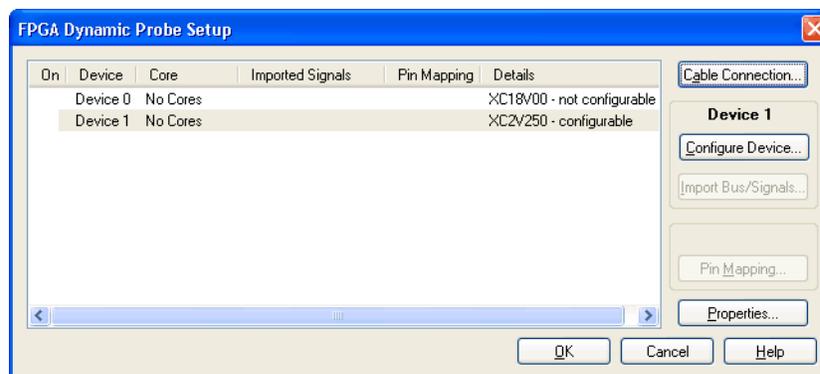
- 2 In the FPGA Dynamic Probe Setup dialog (see [page 64](#)), click **Cable Connection....**



- 3 In the Cable Connection dialog (see [page 65](#)), select the type of cable and, if necessary, specify any cable parameters; then, click **OK**.



When the connection has been established, you will see the devices on the JTAG chain, and you can select the desired ATC2 or MTC core.



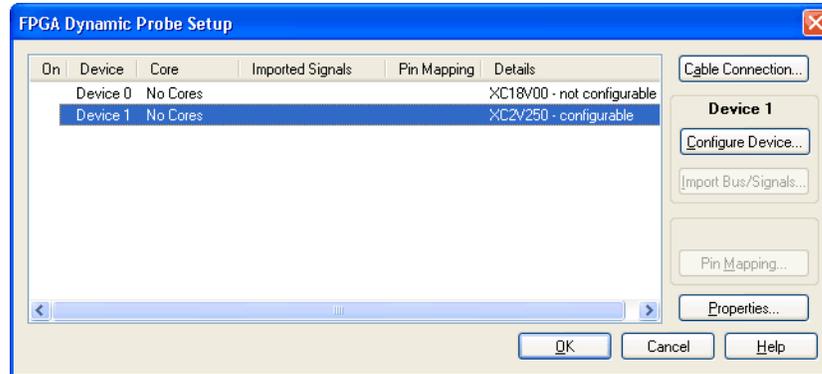
## 7 Measurement Steps

- Next**
- Measurement Step 2. Download configuration bits into FPGA (see [page 31](#))

## Measurement Step 2. Download configuration bits into FPGA

To download configuration bits into an FPGA:

- 1 In the FPGA Dynamic Probe Setup dialog (see [page 64](#)), select the FPGA device to which you wish to download configuration bits; then, click **Configure Device...**



- 2 In the Select FPGA Configuration File dialog (see [page 68](#)), select the FPGA configuration file; then, click **Open**.

**See Also**

- If the ATC2 or MTC core isn't present in an FPGA after downloading configuration bits (see [page 52](#))

**Next**

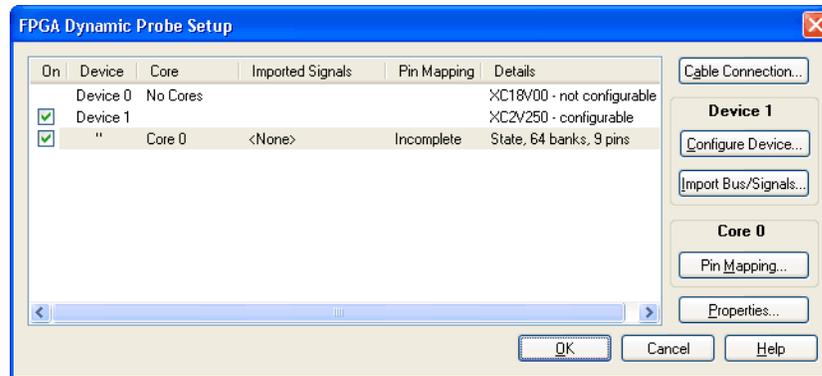
- Measurement Step 3. Map FPGA pins (see [page 32](#))

## Measurement Step 3. Map FPGA pins

Quickly specify how the FPGA pins (the signal outputs of ATC2 or MTC) 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 ATC2 or MTC core.

To map FPGA pins to logic analyzer probes:

- 1 In the FPGA Dynamic Probe Setup dialog (see [page 64](#)), select the ATC2 or MTC core whose output pins you want to map; then, click **Pin Mapping...**

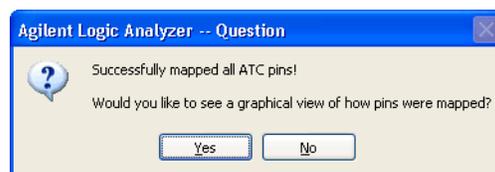


Your next steps depend on whether the core has auto pin-mapping capability:

- To map pins for cores with auto pin-mapping (see [page 32](#))
- To map pins for cores without auto pin-mapping (see [page 34](#))

### To map pins for cores with auto pin-mapping

- 1 If you have a core with auto pin-mapping (like the MTC or an ATC2 core with the auto setup parameter enabled), pin mapping happens automatically after you click **Pin Mapping...** (in the FPGA Dynamic Probe Setup dialog (see [page 64](#))). When the automatic pin mapping completes, you see:



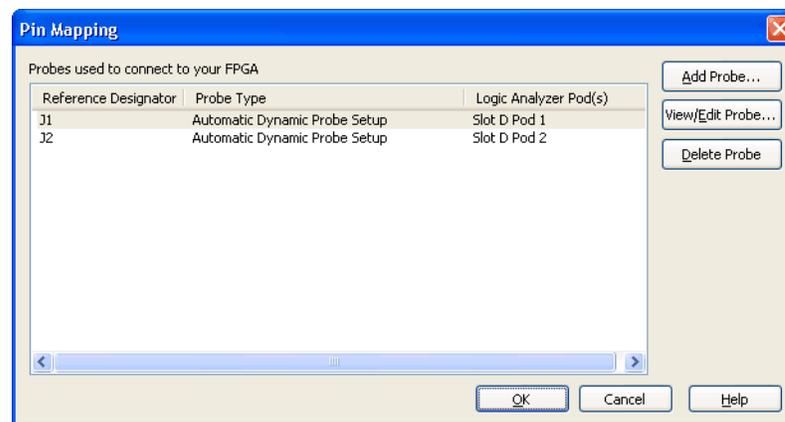
**NOTE**

The FPGA Dynamic Probe does not know the correct reference designators; it simply uses J1, J2, etc.

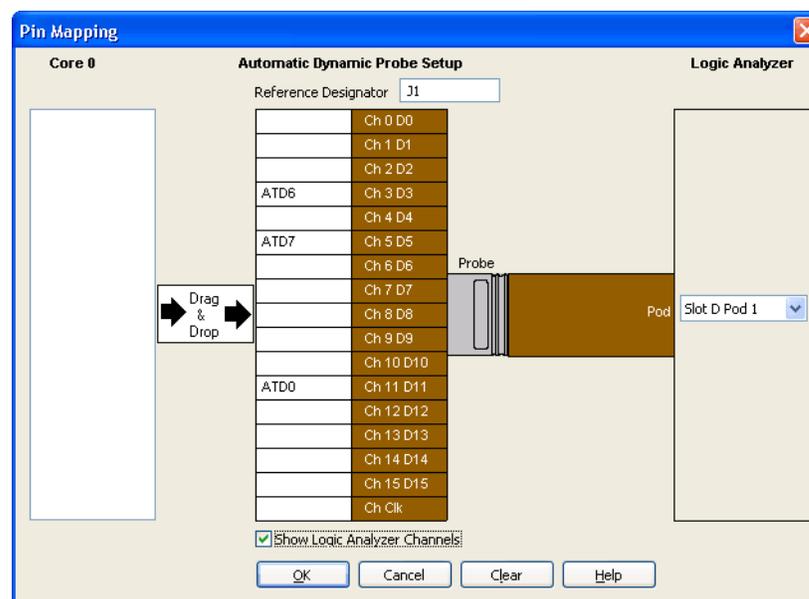
Make sure that the reference designators are correct by looking at the graphical view and editing the probes that were automatically set up. (This is especially important if you will use the "Probe Summary tab" (in the online help) in the System Summary dialog for reconnecting the probes later.)

2 To see how pins were mapped, click **Yes**.

In the Pin Mapping dialog, you will see "Automatic Dynamic Probe Setup" probe types.



3 Select one of the automatically set up probes, and click **Edit Probe...**



- 4 Click **OK** to close the Pin Mapping dialogs.

### To use graphical pin mapping

- 1 If you want to manually map pins using graphical pin mapping, click **Pin Mapping...** again.

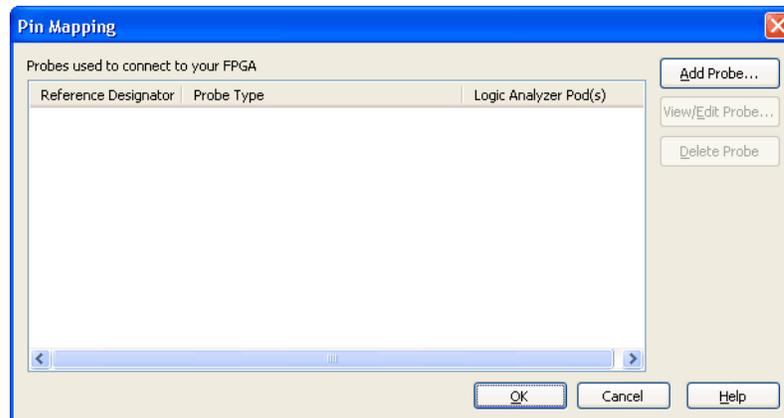


- 2 In the Choose Pin Mapping Method dialog, click **Graphical Pin Mapping...**
- 3 In the Pin Mapping dialog, you can delete the automatically set up probes, and map pins as you would if the core was not plug-n-play (see To map pins for cores without auto pin-mapping (see [page 34](#))).

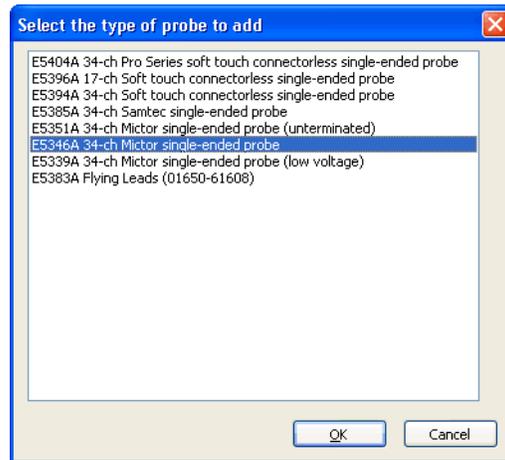
**Next** • Measurement Step 4. Import signal names (see [page 37](#))

### To map pins for cores without auto pin-mapping

- 1 In the Pin Mapping dialog (see [page 66](#)), click **Add Probe...**

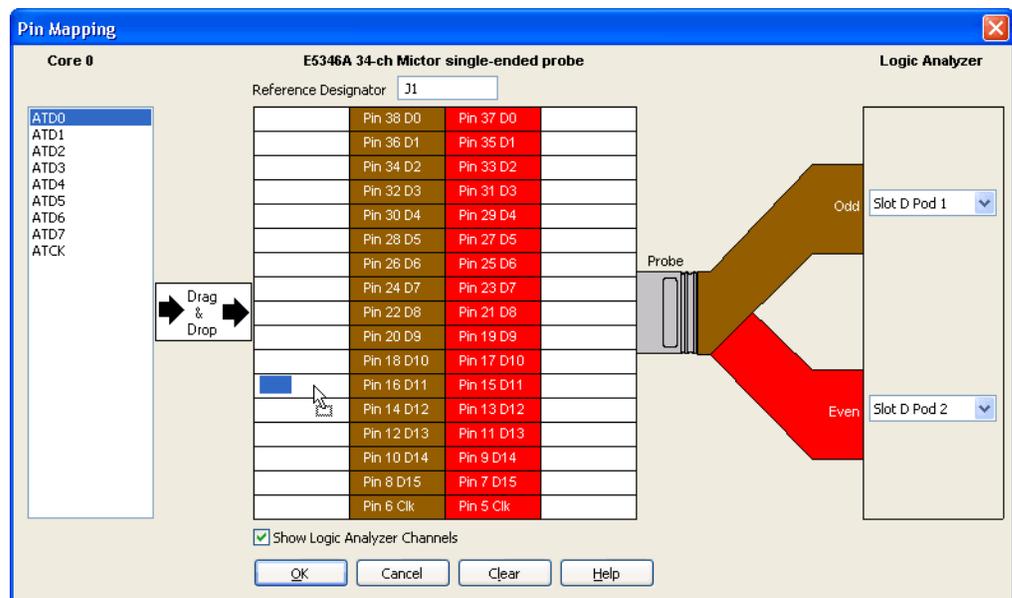


- 2 In the "Select the type of probe to add" dialog, select the type of probe that is used to connect to your FPGA; then, click **OK**.



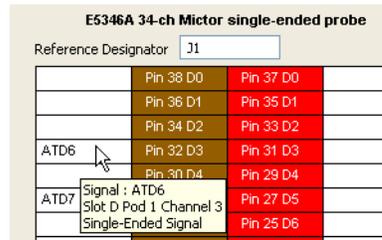
If your probe doesn't appear in the list, you can "download the latest probe definitions from the web" (in the online help).

- 3 In the Pin Mapping Edit dialog (see [page 67](#)), 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).

## 7 Measurement Steps



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.

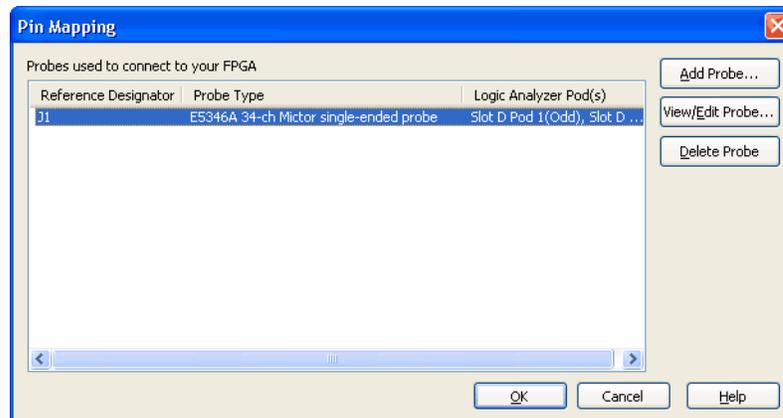
- 4 Select the logic analyzer pods that the probe is connected to.

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.

### NOTE

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.)

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



Click **OK** to close the Pin Mapping dialog.

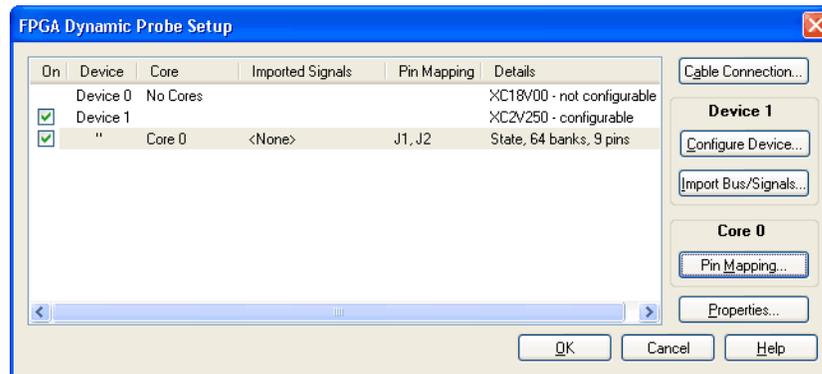
- Next** • Measurement Step 4. Import signal names (see [page 37](#))

## Measurement Step 4. Import signal names

The FPGA dynamic probe can automatically set up bus/signal names in the logic analyzer by reading a .cdc file produced by Xilinx Core Inserter.

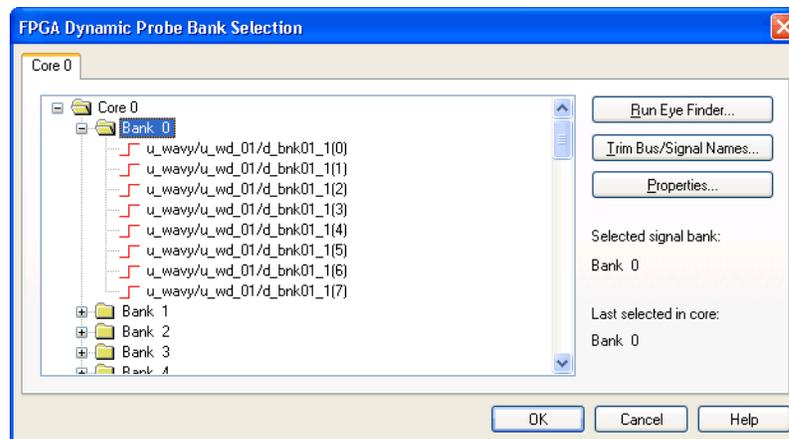
To import bus/signal names:

- 1 In the FPGA Dynamic Probe Setup dialog (see [page 64](#)), select the device whose bus/signal names you want to import; then, click **Import Bus/Signals....**



- 2 In the Select Signal Import File dialog (see [page 69](#)), select the signal import file; then, click **Open**.
- 3 In the Import Results dialog, view the bus/signal import information; then, click **OK**.
- 4 Click **OK** to close the FPGA Dynamic Probe Setup dialog.

In the FPGA Dynamic Probe dialog, note that the imported bus/signal names appear.



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

- See Also**
- To trim imported bus/signal names (see [page 38](#))
  - To rename imported bus/signal names (see [page 38](#))
  - To define additional FPGA bus/signal names (see [page 39](#))

**Next** If you are using an MTC core or you have selected the state capture mode when inserting the ATC2 core (see [page 21](#)), see:

- Measurement Step 5. Adjust sampling positions for state (synchronous) cores (see [page 41](#))

If you selected the timing capture mode when inserting the ATC2 core, see:

- Measurement Step 6. Make the measurement (see [page 44](#))

### 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.

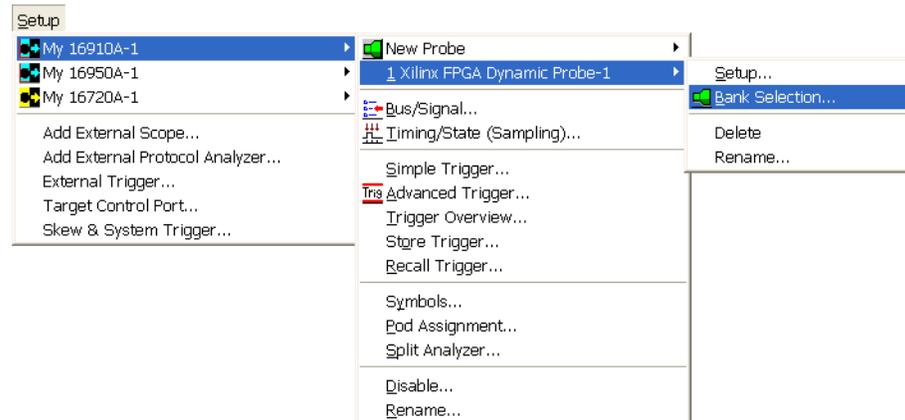
#### NOTE

Because the MicroBlaze inverse assembler relies on particular bus/signal names, do not trim bus/signal names from an MTC core.

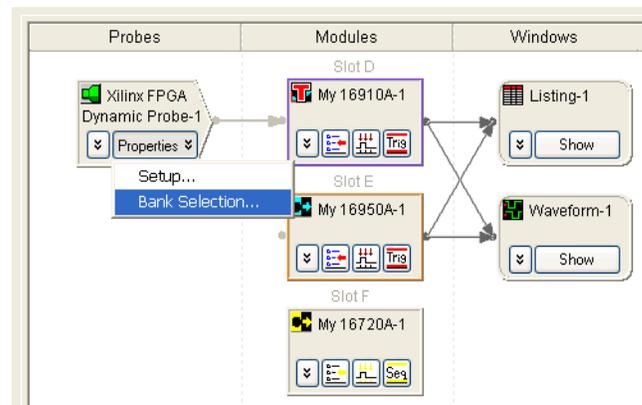
- 1 In the FPGA Dynamic Probe Bank Selection dialog (see [page 71](#)), click **Trim Bus/Signal Names...**
- 2 In the Trim Bus/Signal Names dialog (see [page 71](#)), specify the bus/signal name characters to trim; then, click **OK**.

### 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...**



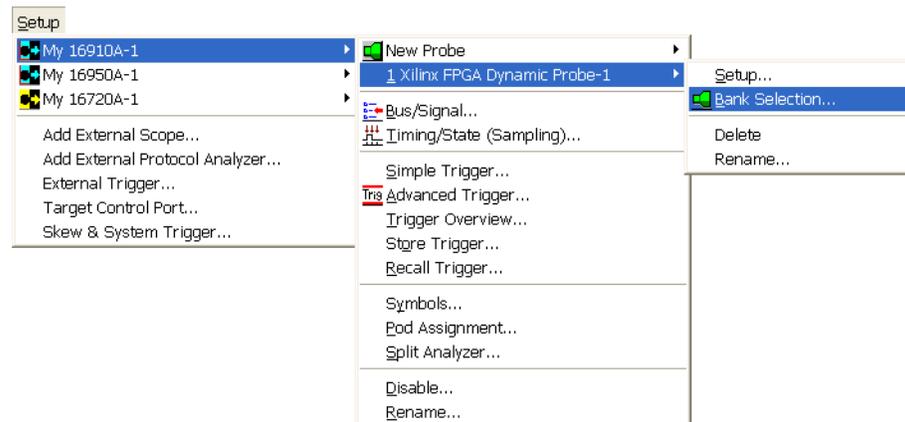
Or, in the Overview window, click the FPGA dynamic probe's **Properties** button; then, choose **Bank Selection...**



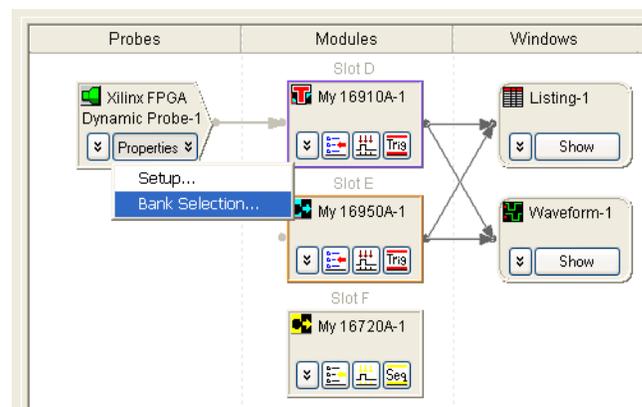
## To define additional FPGA bus/signal names

- 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...**

## 7 Measurement Steps



Or, in the Overview window, click the FPGA 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.

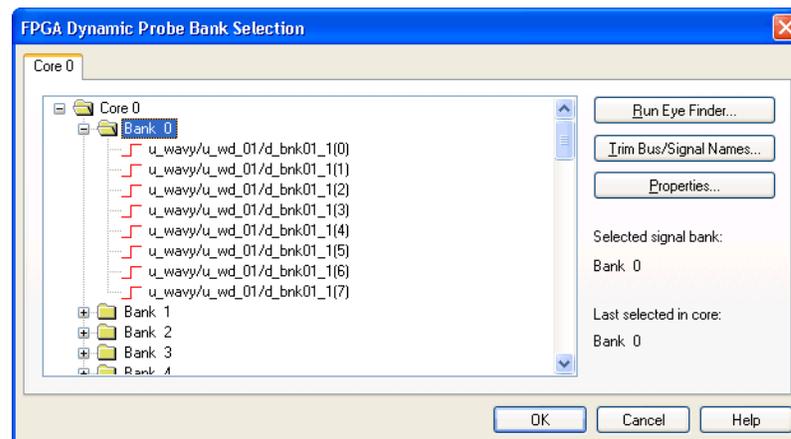
## Measurement Step 5. Adjust sampling positions for state (synchronous) cores

When using MTC or ATC2 state (synchronous) cores, you can automatically adjust the logic analyzer's sampling positions to compensate for variances in signal paths. This results in extremely accurate measurements even on high-speed buses with narrow data valid windows. Agilent strongly recommends that you perform this step when using cores with 2X pin compression.

Agilent trace cores contain test generation circuitry to stimulate the signal banks and output pins. With this active data running, the logic analyzer invokes the *eye finder* feature to automatically adjust for variances in path delays through the core to the acquisition system on the logic analyzer.

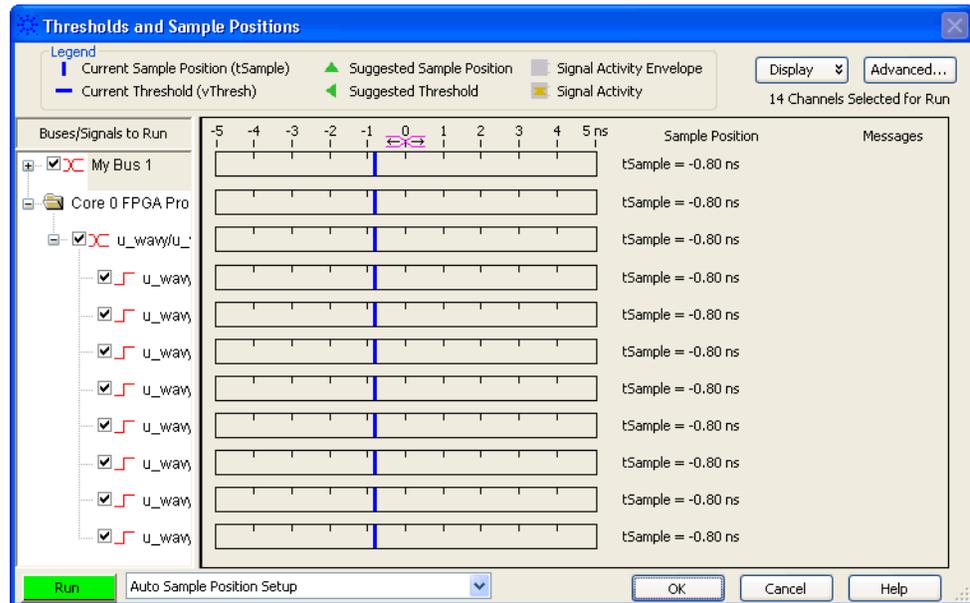
To automatically adjust state mode sampling positions:

- 1 In the FPGA Dynamic Probe Bank Selection dialog (see [page 71](#)), select one of the banks probing internal FPGA signals.

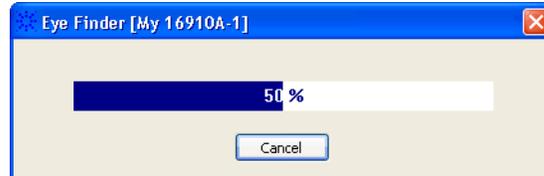


- 2 Click **Run Eye Finder...**
- 3 In the "Thresholds and Sample Positions dialog" (in the online help), click **Run**.

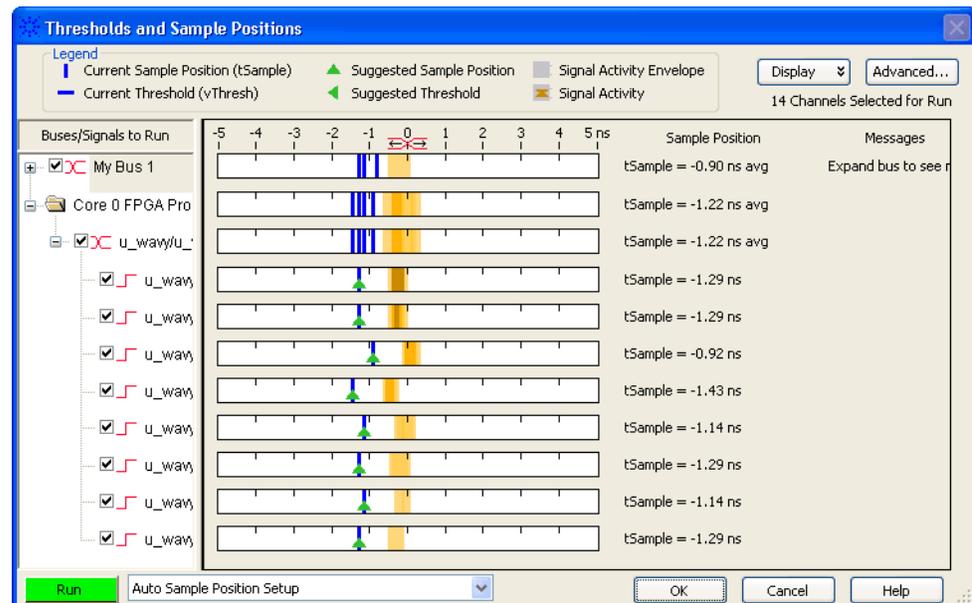
## 7 Measurement Steps



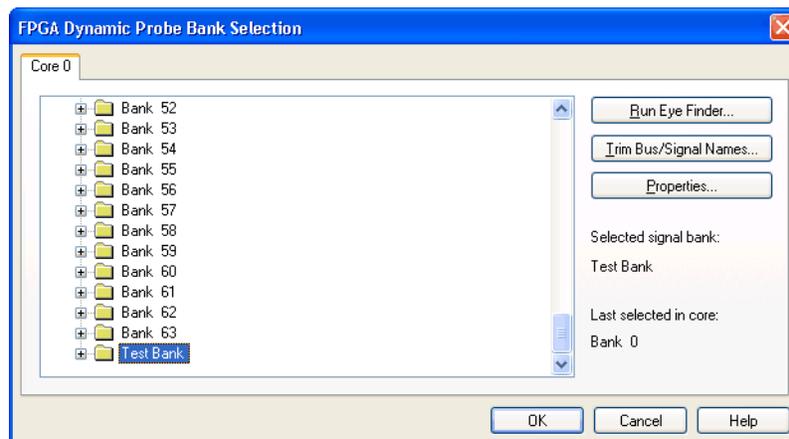
The Eye Finder dialog shows you the progress of the automatic sample positions adjustment.



When the automatic sample positions adjustment is complete, you can view the results in the Sample Positions dialog.



**Test Bank** The Test Bank contains signals internally generated by the ATC2 core. This bank can be used as a first-pass method for adjusting state mode sampling positions, for example, when the internal FPGA signals do not generate a sufficient number of transitions or perhaps during early stages of debug if signals are not being generated properly.



**Next** • Measurement Step 6. Make the measurement (see [page 44](#))

## Measurement Step 6. 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.

- Making Measurements with ATC2 Cores (see [page 44](#))
- Making Measurements with MTC Cores (see [page 46](#))

### NOTE

Timing zoom is automatically disabled when using the FPGA dynamic probe. You can re-enable timing zoom; however, because of the ATC2 core, timing zoom does not provide an accurate representation of internal FPGA signals.

---

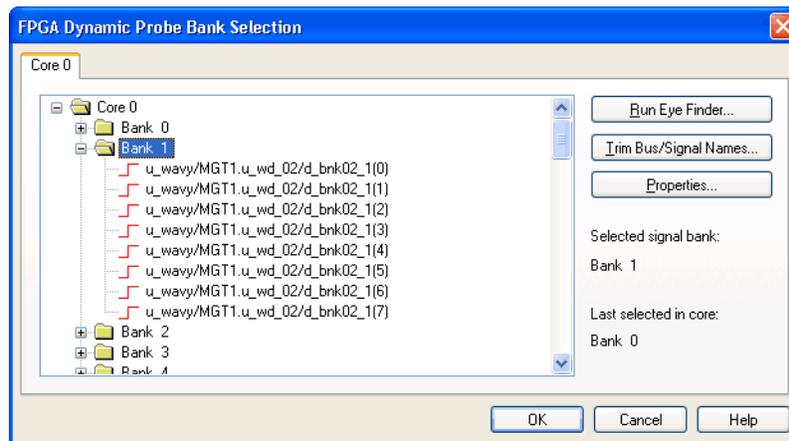
### NOTE

Captured data is invalidated whenever you:

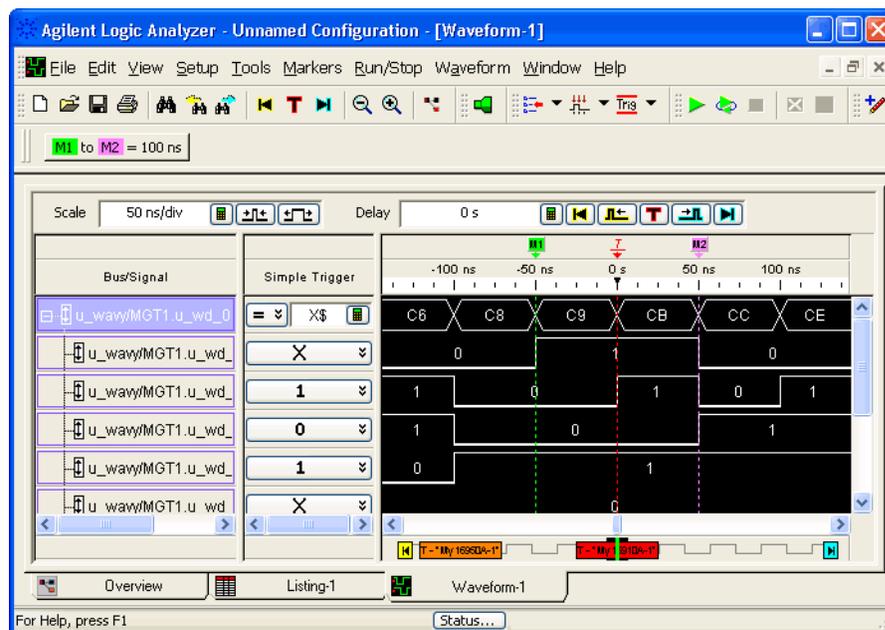
- Select a different bank.
  - Select a different core.
  - Download configuration bits into an FPGA.
  - Reopen a cable connection.
  - Import signal names.
  - Trim imported bus/signal names.
  - Change the FPGA pin mapping.
- 

## Making Measurements with ATC2 Cores

You can tell the ATC2 core to switch signal banks without affecting the timing of your design. When viewing the Probes toolbar (**View>Toolbars>Probes**), click  to 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.



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.



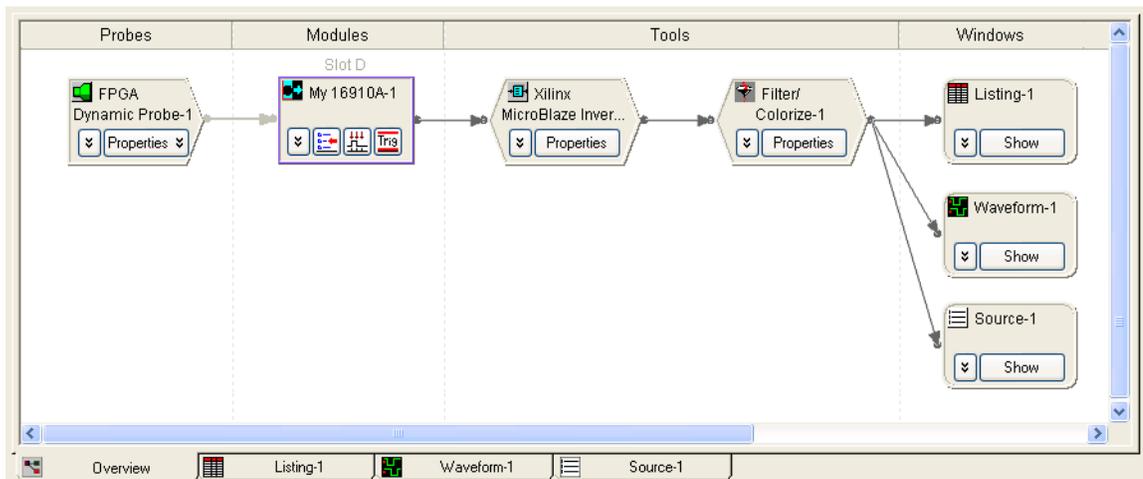
- See Also**
- "Capturing Data from the Device Under Test" (in the online help)
  - "Analyzing the Captured Data" (in the online help)

## Making Measurements with MTC Cores

You can add the MicroBlaze inverse assembler tool to display captured data as assembly language mnemonics in the Listing window.

You can add a Filter/Colorize tool, for example, to hide wait states or add color for certain types of bus cycles.

You can add a Source window to display the high-level language code associated with the captured data.



For example, here is a Source window that shows captured MicroBlaze processor execution:

Agilent Logic Analyzer - [...\SPA\mtx33.ala] - [Source-1]

Displayed File: Z:\Documents and Settings\...\Core5\SPA\SPA.c

```

12 static int OuterExit;
13 OuterEntry=0x21;
14 int j;
15 int numTimes = 0;
16 for ( j =0; j<10; j++)
17 {
18     numTimes ++;
19 };
20 Inner();

```

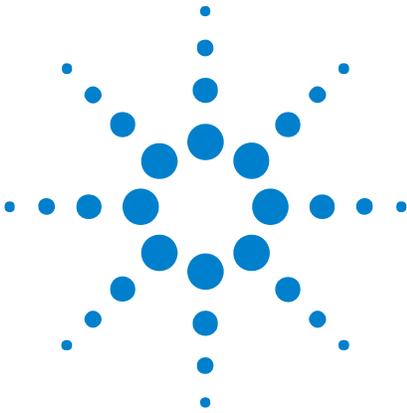
Sample Number	Address	MicroBlaze Inverse Assembly	Cycle Type	Time	ATC
-24	.ini+1C	Write Data 0x9	Write Data	-480 ns	
-19	.te+18C	lwi r3,r19,0x001C	Instructi..	-380 ns	
-18	.ini+18	Read Data 0x8	Read Data	-360 ns	
-11	.te+190	addik r3,r3,0x0001	Instructi..	-220 ns	
-10		Register r3 = 0x9	Write Reg..	-200 ns	
-6	.te+194	swi r3,r19,0x001C	Instructi..	-120 ns	
-5	.ini+18	Write Data 0x9	Write Data	-100 ns	
0	.te+198	bri 0x000001e0	Instructi..	0 s	
9	.te+170	lwi r3,r19,0x001C	Instructi..	180 ns	
10	.ini+18	Read Data 0x9	Read Data	200 ns	
16	.te+174	addik r18,r0,0x0009	Instructi..	320 ns	
17		Register r18 = 0x9	Write Reg..	340 ns	
21	.te+178	cmp r18,r3,r18	Instructi..	420 ns	
22		Register r18 = 0x0	Write Reg..	440 ns	
25	.te+17C	blti r18,0x000001ec	Instructi..	500 ns	

Overview Listing-1 Waveform-1 Source-1

For Help, press F1 Status... Local

- See Also**
- "Xilinx MicroBlaze Inverse Assembler" (in the online help)
  - "Using the Filter/Colorize Tool" (in the online help)
  - "Capturing Data from the Device Under Test" (in the online help)
  - "Analyzing the Captured Data" (in the online help)

## **7 Measurement Steps**



## 8 FPGA Dynamic Probe Troubleshooting

- If you don't see activity in the logic analyzer (see [page 50](#))
- If state mode measurements don't work (see [page 51](#))
- If the ATC2 or MTC core isn't present in an FPGA after downloading configuration bits (see [page 52](#))
- If you get the "Can not open cable" error message (see [page 53](#))



## **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.

## If the ATC2 or MTC core isn't present in an FPGA after downloading configuration bits

FPGA configuration can fail for the following reasons:

- The speed of the cable connection is too high.

In this case, modify the cable connection, set the speed to the lowest setting, 200KHz, and re-attempt to configure the FPGA.

- The FPGA is not properly initialized.

In this case, re-attempt to configure the FPGA (the re-attempt can fix initialization problems).

- The FPGA is in a state where re-configuration does not fix the problem.

In this case, cycle power on the FPGA. After cycling power, use the FPGA dynamic probe to configure the FPGA.

If you are still unable to program the FPGA, refer to the following web site for tips on FPGA configuration:

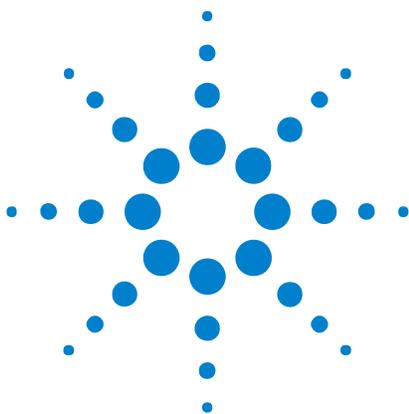
["http://www.support.xilinx.com/support/troubleshoot/psolvers.htm"](http://www.support.xilinx.com/support/troubleshoot/psolvers.htm)

## If you get the "Can not open cable" error message

This message occurs when you open the Cable Connection dialog (see [page 65](#)) while there is a problem with the parallel cable connection.

Check the parallel cable connection.





## 9 FPGA Dynamic Probe Concepts

- How do ATC2 cores affect design timing and signal routing? (see [page 56](#))
- How do ATC2 cores affect FPGA performance? (see [page 57](#))
- Can I put multiple ATC2 cores in a single device? (see [page 58](#))
- Should I leave the ATC2 core in my design after validation is complete? (see [page 59](#))
- How does the optional 2X pin compression technology work? (see [page 60](#))
- Automated Logic Analyzer Set Up (see [page 61](#))
- How do I define buses/signals created in FPGA Editor? (see [page 62](#))

**See Also**

- The Frequently Asked Questions document at:  
["http://www.agilent.com/find/fpga"](http://www.agilent.com/find/fpga)



## How do ATC2 cores affect design timing and signal routing?

The ATC2 core is a firm core. The core will not affect the synthesis of the design when you create and insert the design using Core Inserter. The inclusion of ATC2 core in the design may affect the placement and routing of the overall FPGA design. In typical applications, the effect should be small. The effect on design timing will depend on how the core is configured (for example, the number of signals you choose per bank, and the total number of banks) and the number of resources available on the particular FPGA device being used.

**State core:** The ATC2 state core includes a flop that registers data. The ATC2 core adds one additional load (flop) on each signal probed. This load will be factored into routing to meet user-specified timing constraints. This flop helps the tools meet timing constraints for most types of designs.

**Timing core:** The ATC2 timing core adds a wire load for each signal probed. This wire load is a false path that is ignored during place and route of the design. Hence, the timing core has minimal-to-no effect on design timing. You can determine the skew between signals by viewing the delay file generated by the design tools to estimate this difference.

Once the ATC2 core (either state or timing core) is in the design, switching from signal bank to signal bank does not change design timing, as all connections have been already made.

## How do ATC2 cores affect FPGA performance?

The core will have some effect on performance, but it should not be significant. The affect on performance is directly related to timing constraints. The timing core is capable of running at the fastest internal FPGA speed of the device you are using. For the timing core, the sample rate on the logic analyzer will determine how often data is captured. For state cores, the maximum sample rate will be the lower of the speed of the FPGA time domain being measured or the maximum state speed of the logic analyzer. All supported logic analyzers have state speeds of 200 MHz and greater.

## Can I put multiple ATC2 cores in a single device?

Yes. The FPGA dynamic probe application has been architected to support multiple cores in a single FPGA device. If the cores are state cores, each will need a clock from the design it is measuring.

## Should I leave the ATC2 core in my design after validation is complete?

This decision is up to you. Many designers prefer leaving debug cores in the design because removing the cores from the design will change design timing. The core also can be useful for debugging once units are in the field, if needed. The ATC2 core is by default "turned off", and it wakes when an Agilent logic analyzer communicates with it. This feature saves power when you leave the core in a finished design.

## How does the optional 2X pin compression technology work?

All signal inputs to the ATC2 state core or MTC core are registered and clocked on the edge you specify. Internal to the core, the registered value of signal 1 is passed to the logic analyzer on the rising edge of the clock, and the value of signal 2 is passed to the logic analyzer on the falling edge of the clock. The logic analyzer uses a demultiplexing state sampling clock mode to decompress the information and preserve all triggering and analysis capabilities.

### NOTE

The FPGA dynamic probe detects the type of trace core and automatically sets up the appropriate logic analyzer sampling mode. Because 2X pin compression requires the demultiplex sampling clock mode, you must not change the logic analyzer's sampling option to "*Turbo State Mode*" (in the online help) (which restricts clocking). The 16760A logic analyzer does not support 2X pin compression because it does not have a demultiplex sampling clock mode.

**If I have trace core signal banks with 64-bit width, how many pins and logic analyzer channels does it take with 2X pin compression turned on?**

Simultaneous measurement of 64 signals requires 32 FPGA pins and 64 logic analysis channels. Only 32 of the logic analysis channels are physically connected to pins of the FPGA. The other 32 channels provide their acquisition memory and triggering resources for the measurement.

## Automated Logic Analyzer Set Up

The FPGA dynamic probe automatically sets up the logic analyzer for the type of ATC2 core it connects to.

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

- Timing mode.

For state 1x cores, the FPGA dynamic probe automatically sets up:

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

For state 2x cores, the FPGA dynamic probe automatically sets up:

- State mode.
- Master/Slave/Demux clock mode.
- Clock signals and edges.
- Demultiplex for the pods used (which makes the other pods in the pod pairs unavailable).

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

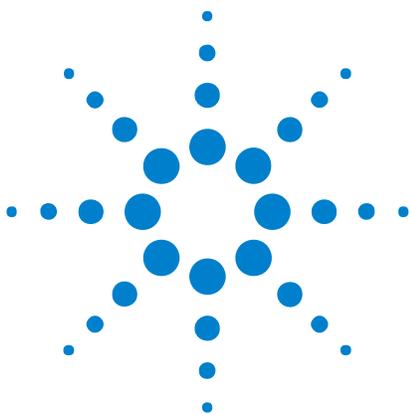
- Bus/signal names for the selected bank.

In the logic analyzer module's "Analyzer Setup dialog" (in the online help), you can rename buses/signals (see [page 38](#)) and define additional bus/signal names (see [page 39](#)), 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.

## How do I define buses/signals created in FPGA Editor?

For buses/signals created in the FPGA Editor (which are not in the .cdc file), you can manually define additional FPGA bus/signal names (see [page 39](#)) in the logic analyzer's Bus/Signal Setup dialog.



## 10 FPGA Dynamic Probe Reference

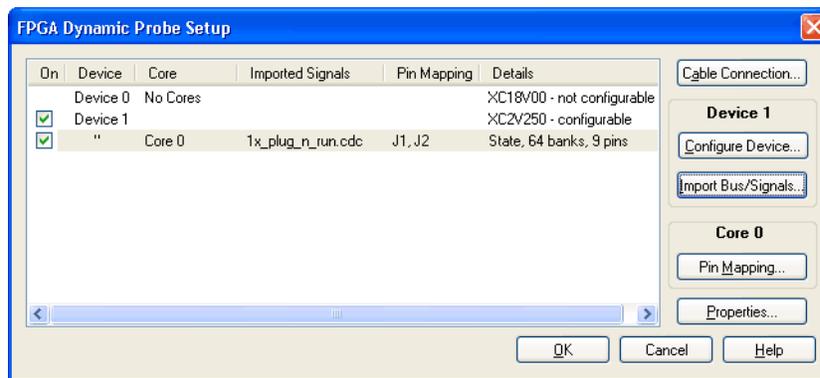
- FPGA Dynamic Probe Setup Dialog (see [page 64](#))
  - Cable Connection Dialog (see [page 65](#))
  - Pin Mapping Dialog (see [page 66](#))
  - Pin Mapping Edit Dialog (see [page 67](#))
  - Select FPGA Configuration File Dialog (see [page 68](#))
  - Select Signal Import File Dialog (see [page 69](#))
  - Properties Dialog (see [page 69](#))
  - Core Details Dialog (see [page 70](#))
- FPGA Dynamic Probe Bank Selection Dialog (see [page 71](#))
  - Trim Bus/Signal Names Dialog (see [page 71](#))
- Specifications and Characteristics (see [page 73](#))



## 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 ATC2 or MTC cores.
- Configure the FPGA with a new design file.
- Map FPGA pins to probe pins/pads.
- Import signal names from the FPGA design tool.



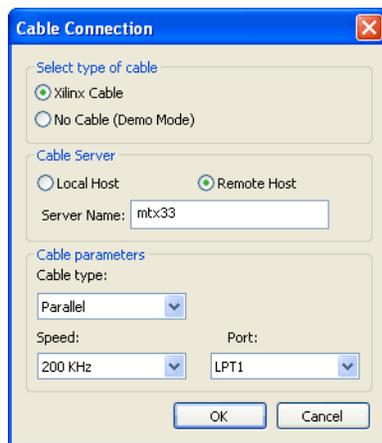
On (in FPGA device list)	The check boxes in this column let you enable or disable a core for use. If you do not want to use a particular core (typically in the multiple core case), you can uncheck its box, and the outputs of the core will be disabled. If a core's "always on" mode is enabled (see the core Details (see <a href="#">page 70</a> ) within its Properties dialog (see <a href="#">page 69</a> )), 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 <a href="#">page 65</a> ) for establishing a connection between the logic analyzer and the ATC2 or MTC core.
Configure Device...	Opens the Select FPGA Configuration File dialog (see <a href="#">page 68</a> ) for downloading a design into the selected FPGA device.
Import Bus/Signals...	Opens the Select Signal Import File dialog (see <a href="#">page 69</a> ) for importing internal FPGA bus/signal names.

Pin Mapping...	<p>Depending on the type of core:</p> <ul style="list-style-type: none"> <li>• If the core is an MTC core or and ATC2 core with the auto setup parameter enabled, automatic pin mapping occurs (see To map pins for cores with auto pin-mapping (see <a href="#">page 32</a>)).</li> <li>• If the core is an older ATC2 core without auto pin-mapping capability or an ATC2 core with the auto setup parameter disabled, this button opens the Pin Mapping dialog (see <a href="#">page 66</a>) for defining the logic analyzer probes that are used to connect to the FPGA and setting up the pin mapping (see To map pins for cores without auto pin-mapping (see <a href="#">page 34</a>)).</li> </ul>
Properties...	<p>Opens the Properties dialog (see <a href="#">page 69</a>) which lets you rename devices and cores as well as display information about the selected ATC2 core.</p>

**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.



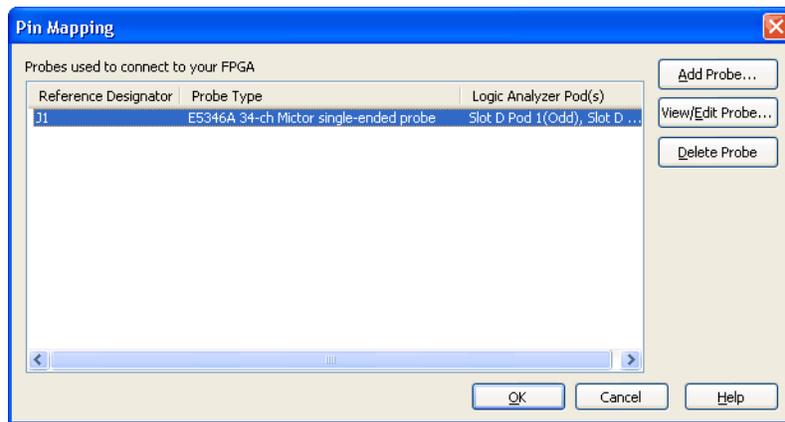
Xilinx Cable	Normally selected.
No Cable (Demo Mode)	Simulates a cable connection for demonstration purposes.
Local Host	Select this when the JTAG cable is connected to the same logic analysis system or PC that runs the <i>Agilent Logic Analyzer</i> application.

Remote Host	Select this when the JTAG cable is connected to a different logic analysis system or PC than the one running the <i>Agilent Logic Analyzer</i> application; then, enter the hostname of the system connected to the JTAG cable (and running the server software).
Cable type	Selects either a Parallel (Xilinx Parallel III and IV MultiLINX cables) or Platform USB cable. The Platform USB cable is supported by version A.03.20 or greater of the <i>Agilent Logic Analyzer</i> application.
Speed	Selects the speed supported by the cable.
Port	When a parallel cable is selected, this selects the port that the cable is connected to.

- See Also**
- Measurement Step 1. Establish connection between analyzer and ATC2 or MTC core (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.

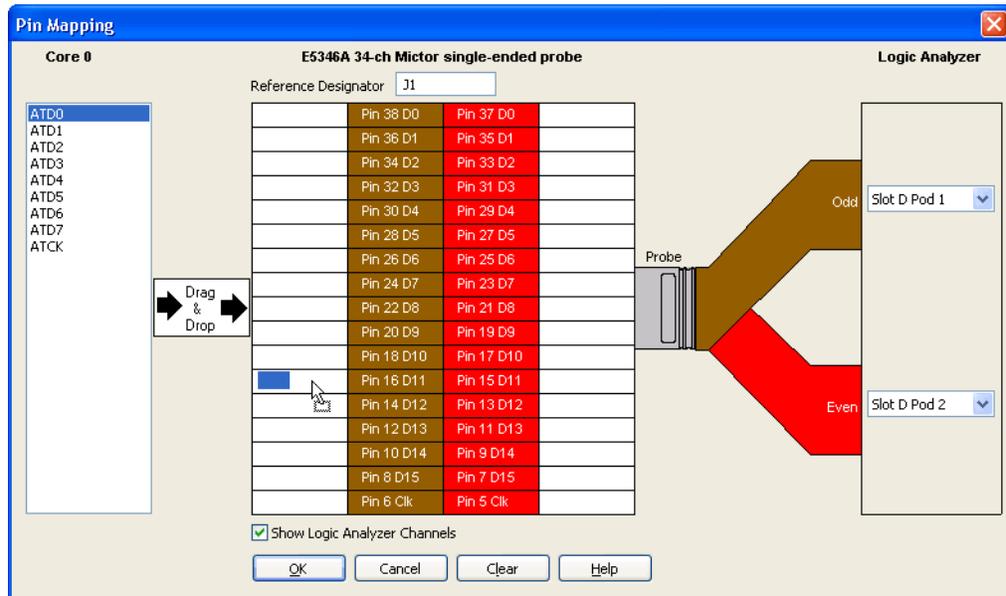


Add Probe...	Opens the "Select the type of probe to add" dialog; when you click <b>OK</b> , the Pin Mapping Edit dialog (see <a href="#">page 67</a> ) is opened for mapping the FPGA output pins to the probe pins/pads. If your probe doesn't appear in the list of probe types, you can "download the latest probe definitions from the web" (in the online help).
Edit Probe...	For the selected probe, opens the Pin Mapping Edit dialog (see <a href="#">page 67</a> ) for editing the FPGA output pins to probe pin/pad mapping.
Delete Probe	Deletes the selected probe.

**See Also** • Measurement Step 3. Map FPGA pins (see [page 32](#))

### Pin Mapping Edit Dialog

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

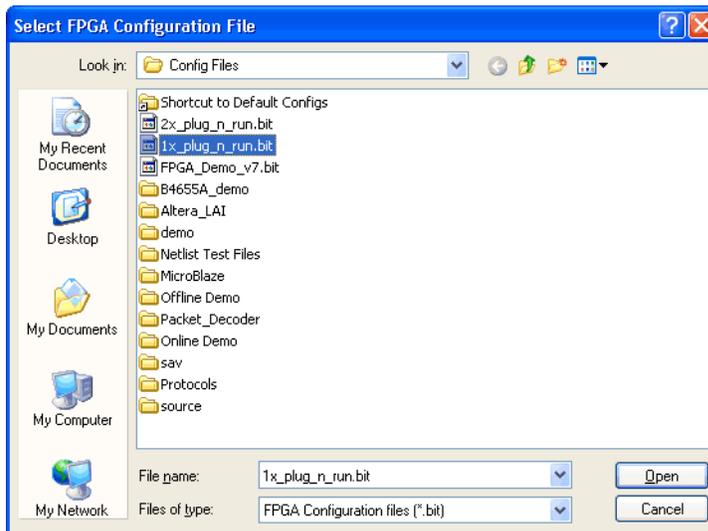


Reference Designator	Identifies the reference designator (in the device under test) of the probe connector, the connectorless probe retention module, or pins probed by flying leads.
FPGA Pins	Lists the FPGA pins used for the ATC2 or MTC core outputs. When dragging these pins onto the pin/pad map, you 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** • Measurement Step 3. Map FPGA pins (see [page 32](#))

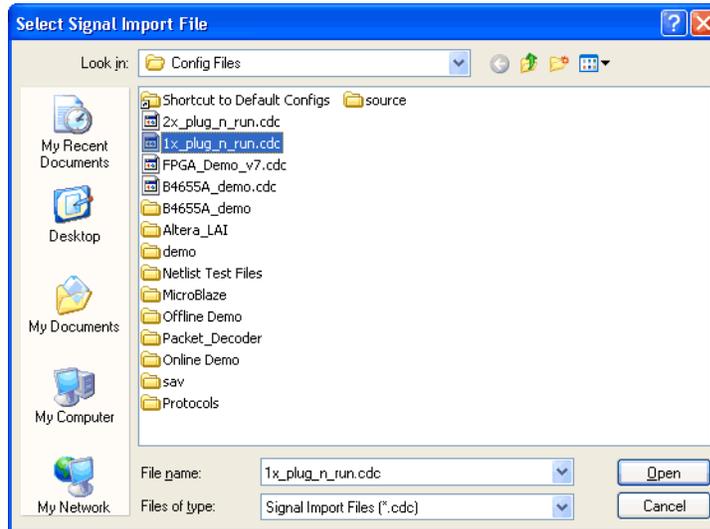
### Select FPGA Configuration File Dialog

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



## Select Signal Import File Dialog

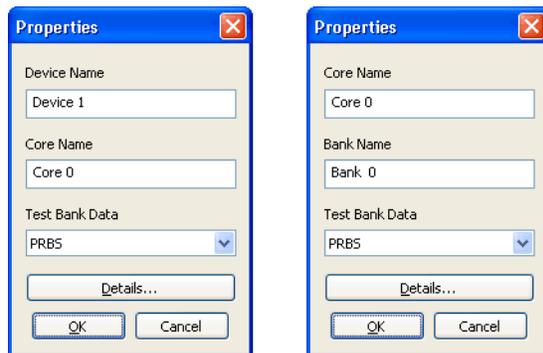
The Select Signal Import File dialog lets you select the file (from the FPGA design tool) that contains the names of the internal buses/signals that appear on ATC2 or MTC core inputs.



**See Also** • Measurement Step 4. Import signal names (see [page 37](#))

## Properties Dialog

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

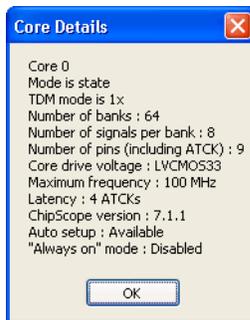


Device Name	Lets you rename the selected device.
Core Name	Lets you rename the selected core.
Bank Name	Lets you rename the selected bank.

Test Bank Data	Lets you select the type of data that appears on the test bank. The types available depend on the options used when the ATC2 or MTC core was created.
Details...	Opens the Core Details dialog (see <a href="#">page 70</a> ) which displays information about the selected ATC2 or MTC core.

## Core Details Dialog

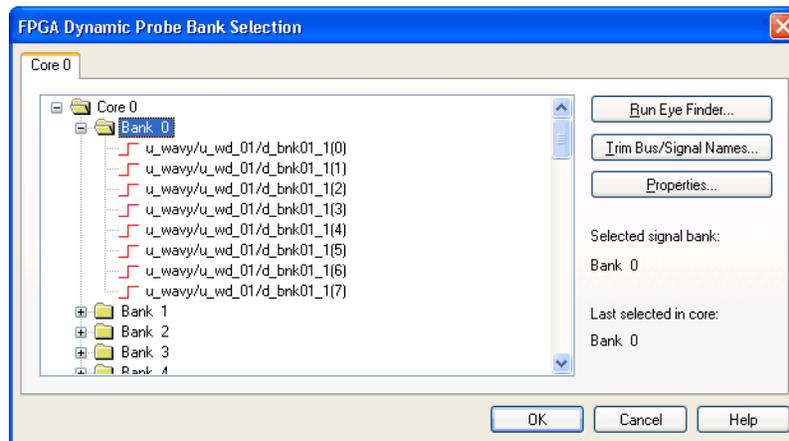
The Core Details dialog displays information about the selected ATC2 core.



## FPGA Dynamic Probe Bank Selection Dialog

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

- Adjust sampling positions (when using MTC cores or ATC2 cores with state mode capture selected).
- Select a different bank of internal signals to probe.
- Rename individual signals (without having to do global trimming) by triple-clicking the signal name.



Run Eyefinder...	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.
Trim Bus/Signal Names...	Opens the Trim Bus/Signal Names dialog (see <a href="#">page 71</a> ) for shortening imported FPGA internal bus/signal names.
Properties...	Opens the Properties dialog (see <a href="#">page 69</a> ) which lets you rename cores and banks as well as display information about the selected ATC2 or MTC core.

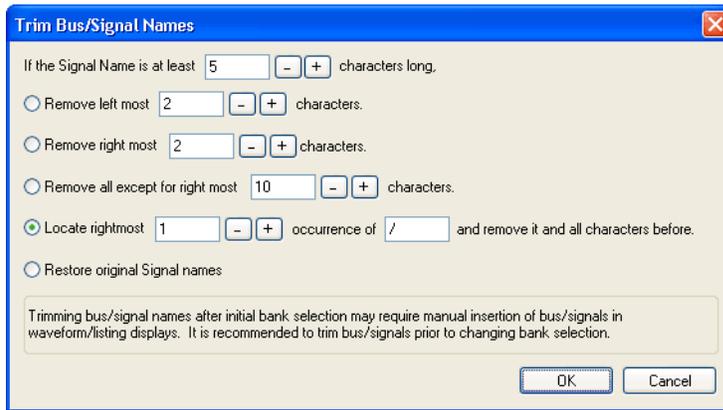
**See Also** • [Measurement Steps](#) (see [page 27](#))

### Trim Bus/Signal Names Dialog

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

#### NOTE

Because the MicroBlaze inverse assembler relies on particular bus/signal names, do not trim bus/signal names from an MTC core.



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** • Measurement Step 4. Import signal names (see [page 37](#))

## FPGA Dynamic Probe Specifications and Characteristics

- Supported Logic Analyzers (see [page 73](#))
- FPGA Dynamic Probe Software Application (see [page 73](#))
- Agilent Trace Core (ATC2) Characteristics (see [page 73](#))
- ATC2 Compatible Design Tools (see [page 74](#))
- MicroBlaze Trace Core (MTC) Characteristics (see [page 75](#))
- MTC Compatible Design Tools (see [page 75](#))

### 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: <ul style="list-style-type: none"> <li>• 16740A, 16741A, 16742A.</li> <li>• 16750A, 16751A, 16752A, 16753A, 16754A, 16755A, 16756A.</li> <li>• 16760A (2X pin compression is not supported because there is no demultiplex sampling clock mode).</li> <li>• 16910A, 16911A, 16950A, 16950B, 16951B.</li> </ul> A single node-locked FPGA dynamic probe license will enable all modules within a 16900 Series system.
Triggering capabilities:	Determined by logic analyzer.
Supported Xilinx FPGA families:	Virtex-5 series, Virtex-4 series, Virtex-II Pro series (including QPro Virtex2 Military/Hi-Rel and QPro Virtex2 Rad Tolerant), Virtex-II series, Spartan-3 series
Supported Xilinx cables (required):	Parallel 3 and 4, Platform Cable USB
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 ATC2 cores supported per FPGA device:	15

### Agilent Trace Core (ATC2) Characteristics

Number of output signals:	User definable: Clock line plus 4 to 128 signals in increments of 1 signal
Signal banks:	User definable: 1, 2, 4, 8, 16, 32, or 64
Modes:	State (synchronous) or timing (asynchronous) mode

Compression:	Optional 2X compression in state mode via time division multiplexing. Logic analyzer decompresses the data stream to allow for full triggering and measurement capability.
FPGA Resource consumption:	Approximately 1 slice required per input signal to ATC2 core. Consumes no BUFGs, DCM or Block RAM resources. See resource calculator at: <a href="http://www.agilent.com/find/fpga">"www.agilent.com/find/fpga"</a> .

**ATC2 Compatible Design Tools**

ChipScope Pro Version	1680, 1690, 16800, 16900 Series SW Version	Primary New Features
6.2i, 6.3i	A.02.50 or higher	Mouse-click bank select, graphical pin mapping, .cdc signal name import.
6.2i, 6.3i	A.03.00 or higher	Support for Virtex-4 devices, improved JTAG drivers, single-session multi-core support, user-definable naming.
7.1i	A.03.20 or higher	Plug & run (auto pin-mapping), ATC2 "always on" option, ATC2 width + 64 banks, Platform Cable USB support, PRBS stimulus on test bank.
8.2i	A.03.50 or higher	Support for Virtex-5, 16800 Series logic analyzers.

EDK (Embedded Development Kit)	1680, 1690, 16800, 16900 Series SW Version	Primary New Features
8.2i SP2	A.03.20 or higher	Support for ATC2 core using EDK flow.
Synthesis:		<p>Core Inserter produces ATC2 cores post-synthesis (pre-place and route) making the cores synthesis independent. ATC2 cores produced by Core Generator are compatible with:</p> <ul style="list-style-type: none"> <li>• Exemplar Leonardo Spectrum</li> <li>• Synopsys Design Compiler</li> <li>• Synopsys Design Compiler II</li> <li>• Synopsys FPGA Express</li> <li>• Synplicity Synplify</li> <li>• Xilinx XST 4</li> </ul>

**MicroBlaze Trace  
Core (MTC)  
Characteristics**

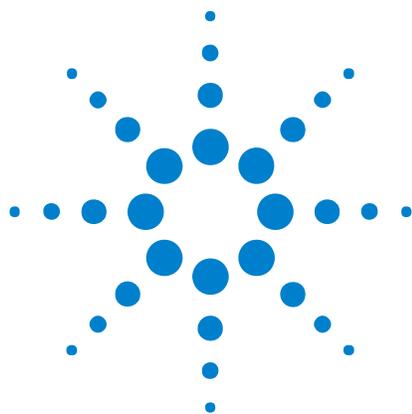
Number of output signals:	User definable: Clock line plus 4 to 128 signals in increments of 1 signal
Inputs:	User definable: Program Counter, Trace Data Address, Control Signals
Compression:	Optional 2X compression via time division multiplexing. Logic analyzer decompresses the data stream to allow for full triggering and measurement capability.
FPGA Resource consumption:	Approximately 1 slice required per input signal to ATC2 core. Consumes no BUFGs, DCM or Block RAM resources. See resource calculator at: " <a href="http://www.agilent.com/find/MicroBlaze">www.agilent.com/find/MicroBlaze</a> ".

**MTC Compatible  
Design Tools**

EDK (Embedded Development Kit)	1680, 1690, 16800, 16900 Series SW Version	Primary New Features
8.2i	A.03.50 or higher	Support for MTC core using EDK flow.

Additional information available via the Internet ("[www.agilent.com/find/FPGA](http://www.agilent.com/find/FPGA)" and "[www.agilent.com/find/fpga\\_FAQ](http://www.agilent.com/find/fpga_FAQ)").





# 11 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 79](#))).

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 Queries Supported

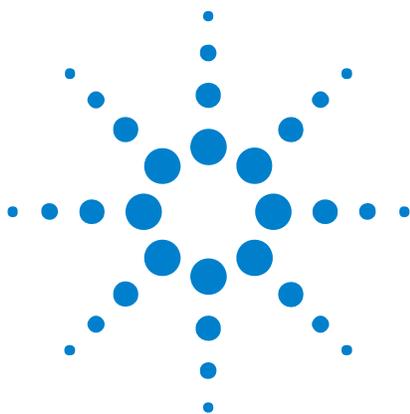
The FPGA dynamic probe supports the following XML-based queries (made with the "Probe" (in the online help) object's "QueryCommand" (in the online help) method).

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 <a href="#">page 106</a> )).
GetProperties	Returns the current setup, using the full tag set, equivalent to "GetAllSetup" (see the XML format <Properties> element (see <a href="#">page 106</a> )).

- See Also**
- "COM Automation" (in the online help)
  - Probe Setup, XML Format (see [page 79](#))







## 12 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 106)
  <ATC_II> (see page 81)
  <JTAG_Chain> (see page 96)
  <Cable> (see page 87)
  <Devices> (see page 94)
    <Device> (see page 92)
      <Cores> (see page 90)
        <Core> (see page 88)
          <Banks> (see page 86)
            <Bank> (see page 82)
              <Signals> (see page 110)
                <Signal> (see page 108)
                  <Labels> (see page 99)
                    <Label> (see page 97)
                  <NonATCLabels> (see page 100)
                    <Label> (see page 98)
                    <Assignment> (see page 80)
                  <WindowInfo> (see page 115)
                  <SymbolInfo> (see page 113)
                  <TriggerInfo> (see page 114)
              <PinMapping> (see page 101)
            <DefinedProbes> (see page 91)
              <Probe> (see page 105)
                <Pods> (see page 104)
                  <Pod> (see page 103)
                <Signals> (see page 112)
                  <Signal> (see page 109)
```

- See Also**
- "XML Format" (in the online help)
  - Probe Control, COM Automation (see [page 77](#))

## <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 98](#)).

### 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='7' />
  </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 106](#)).

**Example**

```
<ATC_II SelectedDevice='1' SelectedCore='0' CableType='0'
ParallelCableType='0' ParallelCableSpeed='2' ParallelCablePort='0'
USBCableSpeed='0' />
```

## <Bank> Element

The <Bank> element describes a bank within an ATC2 core.

### Attributes

Name	Description
CalibrationBank	'F' (false) or 'T' (true)
Name	'string'
NumDataPins	'number'
State2X	'F' (false) or 'T' (true)

**Children** This element can have the following children: <Signals> (see [page 110](#)), <NonATCLabels> (see [page 100](#)), <WindowInfo> (see [page 115](#)), <SymbolInfo> (see [page 113](#)), <TriggerInfo> (see [page 114](#)).

**Parents** This element can have the following parents: <Banks> (see [page 86](#)).

**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>
  </Signals>
</Bank>
```

```

        <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>
    <Labels />
</Signals>
<NonATCLabels />
<WindowInfo WindowHandle='1' WindowSettings='
&lt;Setup&gt;
    &lt;Sampling PerDivision=&apos;5 ns&apos;
        Delay=&apos;0 s&apos;/&gt;
    &lt;BusSignals&gt;
        &lt;Clear/&gt;
        &lt;BusSignal Module=&apos;My 1682D-1&apos;
            Name=&apos;My Bus 1&apos; DefaultBase=&apos;Hex&apos;

```

```

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

```

```

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

```

## <Banks> Element

The <Banks> element contains descriptions of banks within an ATC2 core.

**Children** This element can have the following children: <Bank> (see [page 82](#)).

**Parents** This element can have the following parents: <Core> (see [page 88](#)).

**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>
```

## <Cable> Element

The <Cable> element describes the JTAG cable.

### Attributes

Name	Description
CableHostName	'string' (hostname of logic analyzer or PC with the JTAG cable)
CableHostType	'number' (0 = Local Host, 1 = Remote Host)
CableType	'number' (0 = Xilinx Cable, 1 = No Cable (Demo Mode))
ParallelCablePort	'number' (0 = LPT1, 1 = LPT2)
ParallelCableSpeed	'number' (0 = 5 MHz, 1 = 2.5 MHz, 2 = 200 kHz)
ParallelCableType	'number' (0 = Platform USB, 1 = Parallel)
USBCableSpeed	'number' (0 = 24 MHz, 1 = 12 MHz, 2 = 6 MHz, 3 = 3 MHz, 4 = 1.5 MHz, 5 = 750 kHz)

**Parents** This element can have the following parents: <Properties> (see [page 106](#)).

**Example**

```
<Cable CableType='0' ParallelCableType='0' ParallelCableSpeed='2'
ParallelCablePort='0' USBCableSpeed='2' CableHostType='1'
CableHostName='mtx33' />
```

## <Core> Element

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

### Attributes

Name	Description
ClockPodIndex	'number'
CoreCannotBeMaster	'F' (false) or 'T' (true)
CoreIsMaster	'F' (false) or 'T' (true)
Latency	'number'
MinimumPeriod	'number'
Name	'string'
NumBanks	'number'
NumPins	'number'
NumSignals	'number'
SelectedBank	'number'
SelectedForUse	'F' (false) or 'T' (true)
SelectedSignal	'number'
StateMode	'F' (false) or 'T' (true)
TDM_1X	'F' (false) or 'T' (true)
TestBankAvailable	'F' (false) or 'T' (true)
TestBankMode	'number'
Type	'number'
ThresholdCode	'number'

**Children** This element can have the following children: <Banks> (see [page 86](#)), <PinMapping> (see [page 101](#)).

**Parents** This element can have the following parents: <Cores> (see [page 90](#)).

**Example**

```
<Core Name='Core-1' Type='0' SelectedBank='0' SelectedSignal='-1'
  NumBanks='5' NumSignals='16' NumPins='17' StateMode='T' TDM_1X='T'
  ThresholdCode='92' TestBankAvailable='T' ClockPodIndex='0'
  SelectedForUse='T' CoreIsMaster='T' CoreCannotBeMaster='F'
  MinimumPeriod='-1' TestBankMode='2' Latency='4'>
  <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>
<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 88](#)).

**Parents** This element can have the following parents: <Device> (see [page 92](#)).

**Example**

```
<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>
```

## <DefinedProbes> Element

The <DefinedProbes> element contains defined probes.

**Children** This element can have the following children: <Probe> (see [page 105](#)).

**Parents** This element can have the following parents: <PinMapping> (see [page 101](#)).

**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 'T' (true)
Type	'number'
UserRegNum	'number'

**Children** This element can have the following children: <Cores> (see [page 90](#)).

**Parents** This element can have the following parents: <Devices> (see [page 94](#)).

**Example**

```
<Device Name='Device 1' Type='0' NumCores='2' UserRegNum='1'
  Configurable='T' IRLength='6'
  CDCFilename='C:\Documents and Settings\user\My Documents\Agilent
  Technologies\Logic Analyzer\Config Files\demo\B4655A_demo_V8.cdc'
  FPGAFilename='C:\Documents and Settings\user\My Documents\Agilent
  Technologies\Logic Analyzer\Config Files\demo\B4655A_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>
    </Core>
  </Cores>
</Device>
```

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

## <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 92](#)).

**Parents** This element can have the following parents: <Properties> (see [page 106](#)).

**Example**

```
<Devices>
  <Device Name='Device 0' Type='0' NumCores='0' UserRegNum='0'
    Configurable='F' IRLength='8' CDCFilename='' FPGAFilename=''
    SelectedForUse='F'>
    <Cores/>
  </Device>
  <Device Name='Device 1' Type='0' NumCores='2' UserRegNum='1'
    Configurable='T' IRLength='6'
    CDCFilename='C:\Documents and Settings\user\My
    Documents\Agilent Technologies\Logic Analyzer\Config
    Files\demo\B4655A_demo_V8.cdc'
    FPGAFilename='C:\Documents and Settings\user\My
    Documents\Agilent Technologies\Logic Analyzer\Config
    Files\demo\B4655A_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>
```

```
    </Device>  
</Devices>
```

## <JTAG\_Chain> Element

The <JTAG\_Chain> element describes the number of devices and the number of ATC2 cores on the JTAG chain.

### Attributes

Name	Description
NumATC_II	'number'
NumDevices	'number'

**Parents** This element can have the following parents: <Properties> (see [page 106](#)).

**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 99](#)).

**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 80](#)).

**Parents** This element can have the following parents: <NonATCLabels> (see [page 100](#)).

### 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='7' />
  </Label>
</NonATCLabels>
```

## <Labels> Element

The <Labels> element contains descriptions of labels within a signal.

**Children** This element can have the following children: <Label> (see [page 97](#)).

**Parents** This element can have the following parents: <Signal> (see [page 108](#)).

**Example**

```
<Labels>
  <Label Name='/s2mon/tid' Bit='0' />
</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 98](#)).

**Parents** This element can have the following parents: <Bank> (see [page 82](#)).

**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='7' />
  </Label>
</NonATCLabels>
```

## <PinMapping> Element

The <PinMapping> element contains descriptions of pin mapping within an ATC2 core.

### Attributes

Name	Description
Attached	'F' (false) or 'T' (true)
ModuleHandle	'number'

**Children** This element can have the following children: <DefinedProbes> (see [page 91](#)).

**Parents** This element can have the following parents: <Core> (see [page 88](#)).

**Example**

```
<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>
        <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'
```

## 12 Probe Setup, XML Format

```
        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 104](#)).

**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 103](#)).

**Parents** This element can have the following parents: <Probe> (see [page 105](#)).

**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)
Type	'string' (name of probe)

**Children** This element can have the following children: <Pods> (see [page 104](#)), <Signals> (see [page 112](#)).

**Parents** This element can have the following parents: <DefinedProbes> (see [page 91](#)).

**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 81](#)), <JTAG\_Chain> (see [page 96](#)), <Cable> (see [page 87](#)), <Devices> (see [page 94](#)).

**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' />
  <Cable CableType='0' ParallelCableType='0' ParallelCableSpeed='2'
    ParallelCablePort='0' USBCableSpeed='2' CableHostType='1'
    CableHostName='mtx33' />
  <Devices>
    <Device Name='Device 0' Type='0' NumCores='0' UserRegNum='0'
      Configurable='F' IRLength='8' CDCFilename=''
      FPGAFilename='' SelectedForUse='F'>
      <Cores />
    </Device>
    <Device Name='Device 1' Type='0' NumCores='2' UserRegNum='1'
      Configurable='T' IRLength='6'
      CDCFilename='C:\Documents and Settings\user\My
      Documents\Agilent Technologies\Logic Analyzer\Config
      Files\demo\B4655A_demo_V8.cdc'
      FPGAFilename='C:\Documents and Settings\user\My
      Documents\Agilent Technologies\Logic Analyzer\Config
      Files\demo\B4655A_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'>
            ...
          </Core>
        </Cores>
      </Device>
  </Devices>
</Properties>
```

```

    </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>
</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 99](#)).

**Parents** This element can have the following parents: <Signals> (see [page 110](#)).

**Example**

```
<Signal>
  <Labels>
    <Label Name='/s2mon/tid' Bit='0' />
  </Labels>
</Signal>
```

## <Signal> Element (under Probe)

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

### Attributes

Name	Description
ClockChannel	'F' (false) or 'T' (true)
Name	'string'
PinMapIndex	'number'

**Parents** This element can have the following parents: <Signals> (see [page 112](#)).

**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 108](#)).
- Parents** This element can have the following parents: <Bank> (see [page 82](#)).

**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>
</Signals>
```

```
<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 109](#)).
- Parents** This element can have the following parents: <Probe> (see [page 105](#)).

**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 82](#)).

**Example**

```
<SymbolInfo ModuleHandle='1' SymbolSettings='
  <Module>
    <BusSignalSetup>
      <BusSignals>
        <BusSignal Name='My Bus 1'>
          <Folder Name='Core 0 FPGA Probe'
            Comment='Created by FPGA Dynamic Probe-1'>
            <BusSignal Name='s2mon/tid'>
              <BusSignal Name='s2mon/s2mstate'>
            </Folder>
          </BusSignals>
        </ModuleImport>
      </BusSignalSetup>
      <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)

## <TriggerInfo> Element

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

### Attributes

Name	Description
ModuleHandle	'number'
TriggerSettings	'string'

**Parents** This element can have the following parents: <Bank> (see [page 82](#)).

**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 82](#)).

### Example

```
<WindowInfo WindowHandle='1' WindowSettings='
<lt;Setup>
  <lt;Sampling PerDivision=&apos;5 ns&apos; Delay=&apos;0 s&apos; /&gt;
  <lt;BusSignals>
    <lt;Clear /&gt;
    <lt;BusSignal Module=&apos;My 1682D-1&apos;
      Name=&apos;My Bus 1&apos; DefaultBase=&apos;Hex&apos;
      Color=&apos;hFFFFFF&apos; Height=&apos;30&apos; /&gt;
    <lt;BusSignal Module=&apos;My 1682D-1&apos;
      Name=&apos;/s2mon/tid&apos; DefaultBase=&apos;Hex&apos;
      Color=&apos;hFFFFFF&apos; Height=&apos;30&apos; /&gt;
    <lt;BusSignal Module=&apos;My 1682D-1&apos;
      Name=&apos;/s2mon/s2mstate&apos;
      DefaultBase=&apos;Hex&apos; Color=&apos;hFFFFFF&apos;
      Height=&apos;30&apos; /&gt;
    <lt;BusSignal Name=&apos;Time&apos; Color=&apos;hFFFFFF&apos;
      Height=&apos;30&apos; /&gt;
  <lt;/BusSignals>
<lt;/Setup>
' />
<WindowInfo WindowHandle='2' WindowSettings='
<lt;BusSignals>
  <lt;Clear /&gt;
  <lt;BusSignal Module=&apos;My 1682D-1&apos;
    Name=&apos;Sample Number&apos; Color=&apos;hFFFFFF&apos;
    Alignment=&apos;Right&apos; Width=&apos;112&apos; /&gt;
  <lt;BusSignal Module=&apos;My 1682D-1&apos;
    Name=&apos;My Bus 1&apos; DefaultBase=&apos;Hex&apos;
    Color=&apos;hFFFFFF&apos; Alignment=&apos;Right&apos;
    Width=&apos;113&apos; /&gt;
  <lt;BusSignal Name=&apos;Time&apos; DefaultBase=&apos;Absolute&apos;
    Color=&apos;hFFFFFF&apos; Alignment=&apos;Right&apos;
    Width=&apos;152&apos; /&gt;
  <lt;BusSignal Module=&apos;My 1682D-1&apos;
    Name=&apos;/s2mon/tid&apos; DefaultBase=&apos;Hex&apos;
    Color=&apos;hFFFFFF&apos; Alignment=&apos;Right&apos;
    Width=&apos;113&apos; /&gt;
  <lt;BusSignal Module=&apos;My 1682D-1&apos;
    Name=&apos;/s2mon/s2mstate&apos; DefaultBase=&apos;Hex&apos;
    Color=&apos;hFFFFFF&apos; Alignment=&apos;Right&apos;
    Width=&apos;120&apos; /&gt;
<lt;/BusSignals>
```

## 12 Probe Setup, XML Format

```
</BusSignals>  
' />
```

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

# Index

## Symbols

.bit FPGA configuration file, [68](#)  
.cdc FPGA signal import file, [69](#)

## Numerics

2x pin compression technology, [60](#)

## A

activity not on FPGA dynamic probe, [50](#)  
Agilent Trace Core overview, [9](#)  
always on ATC core mode, [64](#)  
Assignment, XML element, [80](#)  
ATC cores, enabling/disabling, [64](#)  
ATC\_II, XML element, [81](#)  
ATC2 (Agilent Trace Core) characteristics, [73](#)  
ATC2 core not present after FPGA configuration, [52](#)  
ATC2 core parameters, [18](#)  
ATC2 core, creating, [21](#)  
ATC2 core, establishing a connection, [28](#)  
ATC2 core, FPGA timing and signal routing effects, [56](#)  
ATC2 core, leaving in design, [59](#)  
ATC2 core, selecting, [28](#)  
ATC2 core, selecting signal banks, [44](#)  
ATC2 core, state vs. timing, [18](#)  
ATC2 cores, auto pin-mapping, [32](#)  
ATC2 cores, FPGA performance, [57](#)  
ATC2 cores, multiple, [58](#)  
ATC2 overview, [9](#)  
ATC2 state cores, sampling positions, [41](#)  
automated logic analyzer set up (FPGA dynamic probe), [61](#)  
automatic pin mapping, [32](#)

## B

B4655A Xilinx FPGA dynamic probe, [15](#)  
Bank, XML element, [82](#)  
banks (ATC2 core), number of, [19](#)  
banks (ATC2 core), selecting, [44](#)  
Banks, XML element, [86](#)  
bus/signal names, adding FPGA, [39](#)  
bus/signal names, renaming FPGA, [38](#)  
bus/signal names, trimming FPGA, [38](#)

## C

Cable Connection dialog, [65](#)

cable connection, selecting type of, [28](#)  
Cable, XML element, [87](#)  
calculator, FPGA resource, [20](#)  
can not open cable, [53](#)  
captured data, invalidation of, [44](#)  
CDC file, importing FPGA signal names from, [37](#)  
CDC file, produced by Xilinx Core Inserter, [21](#)  
Clk pin/pad locations, [34](#)  
clock activity indicators, [51](#)  
clock pin required for ATC2 state core, [19](#)  
COM automation, FPGA dynamic probe, [77](#)  
concepts, FPGA dynamic probe, [55](#)  
configuration (FPGA), trace core not present after, [52](#)  
configuration file, FPGA, [68](#)  
configure FPGA device, [31](#)  
connection to ATC2 core, establishing, [28](#)  
consumption, FPGA resource, [20](#)  
core (ATC2 or MTC) not present after FPGA configuration, [52](#)  
Core Details dialog, [70](#)  
Core Generator (Xilinx), [21](#)  
Core Inserter (Xilinx), [21](#)  
core name, [69](#)  
Core, XML element, [88](#)  
cores (ATC), enabling/disabling, [64](#)  
cores, auto pin-mapping, [32](#)  
cores, multiple, mapping to two halves of the same probe, [34](#)  
Cores, XML element, [90](#)

## D

data, invalidation of, [44](#)  
debug pins (FPGA), mapping to probes and pods, [32](#)  
debug pins (FPGA), number of, [19](#)  
debug pins (FPGA), probing, [13](#)  
debug pins, FPGA, [67](#)  
DefinedProbes, XML element, [91](#)  
demo mode, FPGA dynamic probe, [65](#)  
design (FPGA) tools, [73](#)  
design steps for ATC2, [17](#)  
design steps for MTC, [23](#)  
device name, [69](#)  
Device, XML element, [92](#)  
Devices, XML element, [94](#)  
disabling/enabling ATC cores, [64](#)

## E

enabling/disabling ATC cores, [64](#)

eye finder, ATC2 state core sampling positions, [41](#)

## F

flip-flop (FPGA resource), [20](#)  
flop (FPGA resource), [20](#)  
FPGA bus/signal names, adding, [39](#)  
FPGA configuration file, [68](#)  
FPGA configuration, trace core not present after, [52](#)  
FPGA debug pins, [67](#)  
FPGA debug pins, mapping to probes and pods, [32](#)  
FPGA debug pins, probing, [13](#)  
FPGA design tools, [73](#)  
FPGA dynamic probe characteristics, [73](#)  
FPGA Dynamic Probe dialog, [71](#)  
FPGA Dynamic Probe Setup dialog, [64](#)  
FPGA dynamic probe specifications, [73](#)  
FPGA dynamic probe, ATC2 design steps, [17](#)  
FPGA dynamic probe, automated logic analyzer set up, [61](#)  
FPGA dynamic probe, measurement steps, [27](#)  
FPGA dynamic probe, MTC design steps, [23](#)  
FPGA dynamic probe, no activity, [50](#)  
FPGA dynamic probe, state mode measurements, [51](#)  
FPGA Editor, buses/signals created by, [62](#)  
FPGA imported bus/signal names, renaming, [38](#)  
FPGA imported bus/signal names, trimming, [38](#)  
FPGA internal signals, selecting which to probe, [22](#)  
FPGA performance, how ATC2 cores affect, [57](#)  
FPGA Probe, bus/signal name folder, [39](#)  
FPGA resource consumption, [20](#)  
FPGA signal names, importing from CDC file, [37](#)  
FPGA, downloading configuration bits, [31](#)

## G

graphical pin mapping, [32](#)

## I

imported FPGA bus/signal names, renaming, [38](#)  
imported FPGA bus/signal names, trimming, [38](#)  
importing FPGA signal names from CDC file, [37](#)  
installing the Xilinx FPGA dynamic probe, [15](#)

## Index

internal FPGA signals, selecting which to probe, [22](#)  
invalidation of captured data, [44](#)

## J

JTAG scan chain, devices on, [28](#)  
JTAG scan chain, maximum number of devices, [73](#)  
JTAG\_Chain, XML element, [96](#)

## L

Label, XML element, [97](#)  
Label, XML element (under NonATCLabels), [98](#)  
Labels, XML element, [99](#)  
licensing the Xilinx FPGA dynamic probe, [15](#)  
logic analyzer probing of FPGA debug pins, [13](#)  
logic analyzer set up, automated by FPGA dynamic probe, [61](#)  
LUT (FPGA resource), [20](#)

## M

mapping FPGA debug pins to probes and pods, [32](#)  
mapping multiple cores to two halves of the same probe, [34](#)  
measurement steps for FPGA dynamic probe, [27](#)  
measurements on FPGA internal signals, [44](#)  
MicroBlaze inverse assembler, [46](#)  
MicroBlaze Trace Core overview, [11](#)  
MTC (MicroBlaze Trace Core) characteristics, [73](#)  
MTC core not present after FPGA configuration, [52](#)  
MTC core, creating, [24](#)  
MTC core, implementing, [25](#)  
MTC core, measurements with, [46](#)  
MTC cores, sampling positions, [41](#)  
MTC overview, [11](#)  
multiple ATC2 cores, [58](#)  
multiple cores, mapping to two halves of the same probe, [34](#)

## N

new (FPGA dynamic) probe, [28](#)  
no activity on FPGA dynamic probe, [50](#)  
NonATCLabels, XML element, [100](#)  
notices, [2](#)

## O

On column in FPGA Dynamic Probe Setup dialog, [64](#)  
overview, ATC2, [9](#)  
overview, MTC, [11](#)

## P

parallel cable connection problem, [53](#)  
parallel cable, Xilinx, [65](#)  
performance (FPGA), how ATC2 cores affect, [57](#)  
Pin Mapping dialog, [66](#)  
Pin Mapping Edit dialog, [67](#)  
PinMapping, XML element, [101](#)  
plug-n-run cores, automatic pin mapping, [32](#)  
Pod, XML element, [103](#)  
pods, logic analyzer, [67](#)  
pods, mapping FPGA debug pins to, [32](#)  
Pods, XML element, [104](#)  
PRBS test bank data, [69](#)  
probe, mapping multiple cores to two halves of the same, [34](#)  
Probe, XML element, [105](#)  
Probes toolbar, [44](#)  
probes, adding/editing, [66](#)  
probes, mapping FPGA debug pins to, [32](#)  
probing internal FPGA signals, [22](#)  
probing of FPGA debug pins, [13](#)  
Properties dialog, [69](#)  
Properties, XML element, [106](#)

## R

reference designator, [67](#)  
reference designators with automatic pin mapping, [32](#)  
reference, FPGA dynamic probe, [63](#)  
rename individual signals, [32, 71](#)  
renaming imported FPGA bus/signal names, [38](#)  
resource consumption, FPGA, [20](#)

## S

sampling positions for ATC2 state cores, [41](#)  
Select FPGA Configuration File dialog, [68](#)  
Select Signal Import File dialog, [69](#)  
selecting FPGA signals to probe, [22](#)  
signal banks (ATC2 core), selecting, [44](#)  
signal import file, FPGA, [69](#)  
signal names (FPGA), importing from CDC file, [37](#)  
signal routing (FPGA), how ATC2 cores affect, [56](#)  
Signal, XML element, [108, 109](#)  
signal/bus names, adding FPGA, [39](#)  
signal/bus names, renaming FPGA, [38](#)  
signal/bus names, trimming FPGA, [38](#)  
signals per bank (ATC2 core), number of, [19](#)  
Signals, XML element, [110, 112](#)  
slice (FPGA resource), [20](#)  
state ATC2 core, [18](#)  
state ATC2 cores, sampling positions, [41](#)  
state mode measurements in FPGA dynamic probe, [51](#)  
SymbolInfo, XML element, [113](#)  
synthesis (FPGA) tools, [21, 73](#)

## T

test bank, [41](#)  
test bank data, [69](#)  
timing (FPGA), how ATC2 cores affect, [56](#)  
timing ATC2 core, [18](#)  
timing zoom disabled by FPGA dynamic probe, [44](#)  
tool tip, mapping FPGA debug pins, [34](#)  
trace core not present after FPGA configuration, [52](#)  
trademarks, [2](#)  
TriggerInfo, XML element, [114](#)  
Trim Bus/Signal Names dialog, [71](#)  
trimming imported FPGA bus/signal names, [38](#)  
triple-click to rename individual signals, [37, 71](#)  
troubleshooting, FPGA dynamic probe, [49](#)

## V

VIO stimulus core, Xilinx Core Generator required, [21](#)

## W

WindowInfo, XML element, [115](#)

## X

Xilinx ChipScope Pro, [21](#)  
Xilinx Core Generator, [21](#)  
Xilinx FPGA dynamic probe, [3](#)  
Xilinx FPGA dynamic probe, installing and licensing, [15](#)  
Xilinx FPGA families supported by dynamic probe, [73](#)  
Xilinx parallel cable, [65](#)  
Xilinx Platform Studio, [24](#)  
XML format, FPGA dynamic probe, [79](#)