Getting Started

FPGAView
Xilinx

July 14, 2008

First Silicon Solutions
a division of MIPS Technologies, Inc.

document 10587.000

Contact Info:

MIPS Technologies Inc.
First Silicon Solutions
1260 NW Waterhouse Ave., #100
Beaverton, OR 97006-5794

Ph. (503) 597-5091
Fax (503) 597-5098

http://www.fs2.com
info@fs2.com
support@fs2.com

Table of Contents

1. Introduction

FPGAView, combined with the Xilinx FPGA tools, and a Tektronix logic analyzer (TLA) or Tektronix Mixed Signal Oscilloscope (MSO), allows a designer to quickly and easily select one or more of several banks of internal FPGA signals for capture and analysis. A single hardware debug port can be used to view up to 64 different banks of internal FPGA signals, which are routed to selected FPGA output pins via 'FVX OCI' multiplexers incorporated into the FPGA under test via the Xilinx ISE. FPGAView is used to select which banks of signals to route to the hardware debug port(s) and simultaneously update the TLA or MSO with the selected signals' logical names.

The FS2 OCIGen utility is used by the designer to define and insert a debug core into the FPGA design. Then user then programs the FPGA and controls the debug core via the Xilinx Platform Cable USB JTAG cable. Currently, the Xilinx JTAG cable must be on the same machine on which FPGAView is running (PC or TLA.)

2. Setting Up the JTAG Interface

Prior to installing FPGAView, the Xilinx ISE version 8.2i or later must be installed on the same PC or the TLA which will be connected to the JTAG programming cable, and on which FPGAView will be run. If Xilinx ISE version 9.2i or later is installed, Chipscope Pro is not needed. If Xilinx ISE version 9.1i is installed, Xilinx Chipscope Pro ver 9.1i must also be installed.

Xilinx provides several options for FPGA programming and control via JTAG, but currently the only interface supported is the 'Xilinx Platform Cable USB' programming cable. Once you have installed the ISE, connect the Platform Cable USB JTAG interface to the board under test and to your development PC, or the TLA if you will be running FPGAView on the TLA itself. The Xilinx ISE contains the drivers required by the Platform Cable USB programming cable.

3. Installing the Software

FS2 software is supplied on a CDROM or via the Internet. To install the software, insert the CDROM and run the setup.exe program found in the root directory on the CDROM. Software updates are supplied electronically via the FS2 website to licensed users. To install an update, download the appropriate EXE file, and then execute it.

FPGAView may be installed on a PC or on a TLA. For an MSO it must be installed on a PC. Installing on the TLA is often preferable if you do not have a workstation PC near the board to be tested. You will also need the Xilinx Platform Cable USB JTAG interface for controlling the FPGA from FPGAView, and for reprogramming the FPGA as you make changes to your design.

If FPGAView is installed on the TLA, the Xilinx Platform Cable USB JTAG interface and ISE v.8.2i (or later) must also be installed on the TLA. If Xilinx ISE version 9.2i or later is installed, Chipscope Pro is not needed. If ISE v.9.1i or later is being used, Chipscope Pro v.9.1i or later MUST be installed on the same machine.

4. Setting Up the TLA

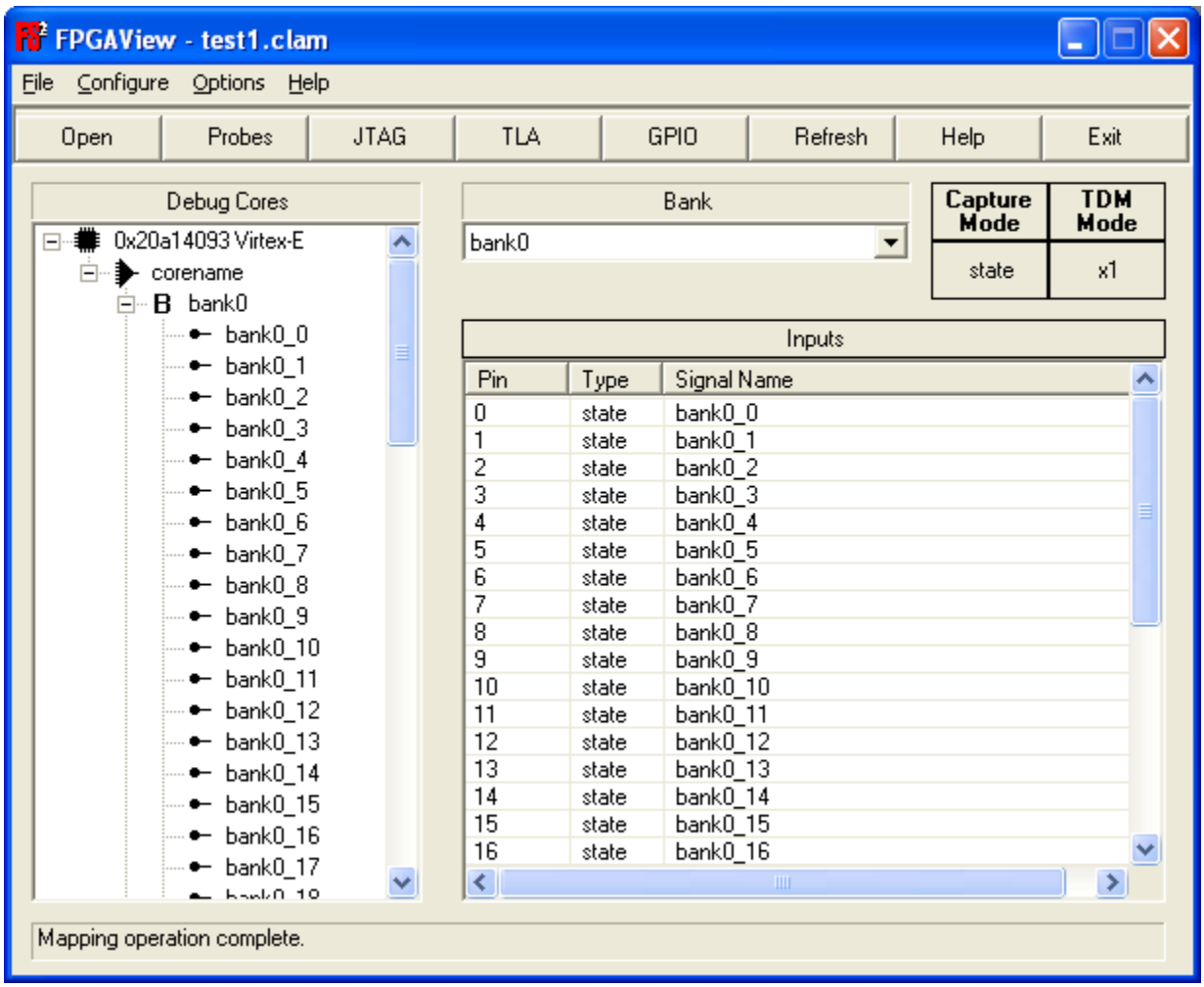
The TLA or MSO logic probes connect to the board under test via a debug port such as a Mictor38, connectorless board site, or a set of square pins. The debug port should be designed for easy connection to a logic analyzer. The signals available on the debug port are the outputs from the debug cores you create with the FS2 OCIGen utility. Once you decide which probe interfaces will be connected to the hardware debug port, you are ready to set up FPGAView.

For the case where FPGAView will be executed on a PC, it must be able to communicate with the TLA via a TCP/IP network connection.

For an MSO, the PC can connect up to it via network, USB port, or GPIB.

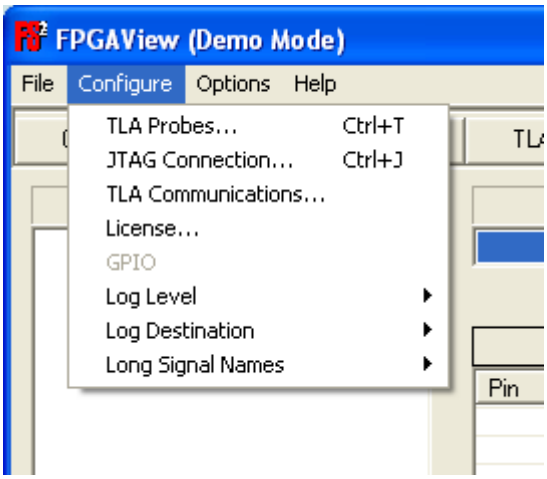
5. Setting up FPGAView

After starting FPGAView, you will see the main screen, which is both the main user interface for configuring the program, and for selecting which banks are to be routed to the TLA during development.

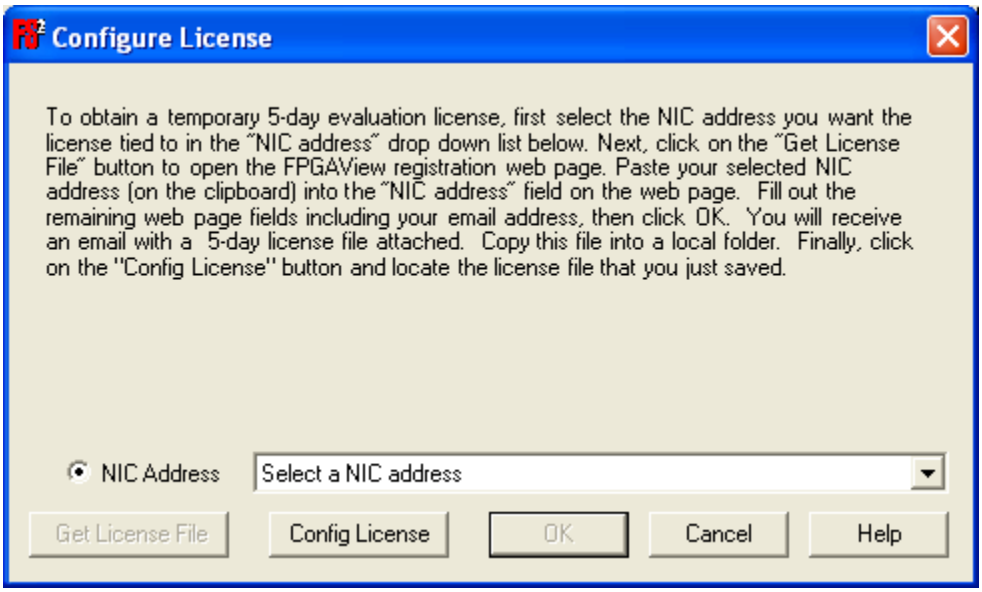


5.1. Licensing

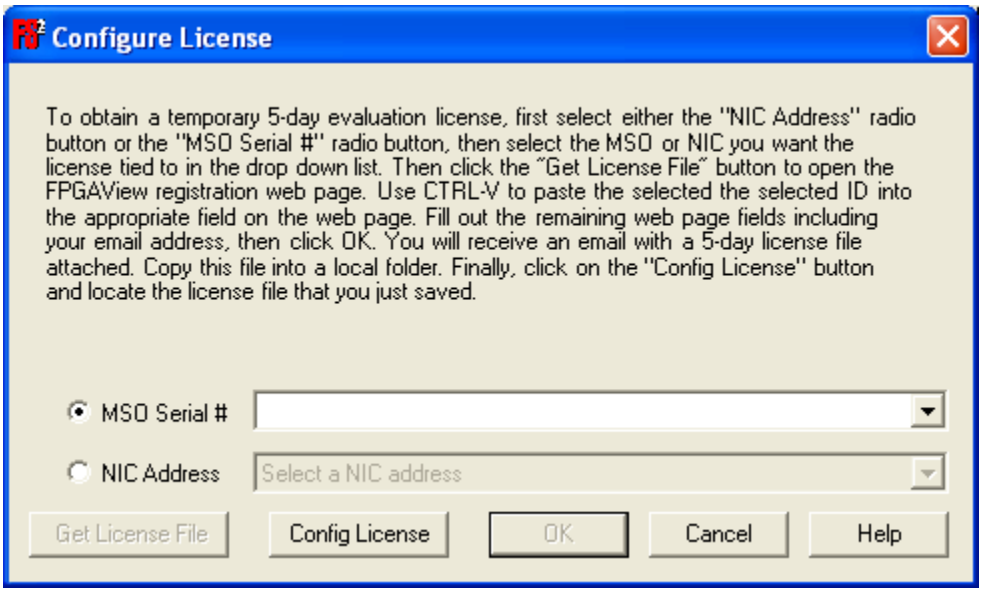
Prior to using FPGAView you must obtain a software license key. Select 'License…' from the 'Configure' menu entry.



If connecting to a TLA, the Configure License window displayed will be:



If connecting to a MSO, the Configure License window displayed will be:



The 'NIC address' field will contain one or more choices of NIC interfaces from which to choose. The NIC address is a 12 character value in the leftmost portion of the NIC address field.

The 'MSO Serial #' field will contain one or more choices of MSO instruments currently connected.

Click on the appropriate radio button to enable selection of the instrument you wish to associate with the license.

Clicking on the 'Get License File' button will open the FS2 FPGAView Evaluation Request webform. Put the cursor in the web form edit field labeled 'NIC Address' or 'MSO Serial #' and then select 'Paste' from the browser's Edit menu, or type Ctrl-v (hold the Ctrl key down and type the letter 'v'.) This will copy the NIC address or MSO serial number from the Configure License dialog into the web form. Repeat for the 'Confirm …' field. Complete all required entries in the form, then click on the 'Request Evaluation License' button at the bottom of the form. A license key file will be emailed to you.

If you want to purchase a longer term license, contact FS2 sales. A license key file will be provided to you.

Copy the .lic license key file you receive to a directory of your choice on the machine on which FPGAView will be run. Click the 'Config License' button and use the file open dialog window to locate the license key file you just installed. Finally, click the 'OK' button to configure the license key. The software will be fully functional for the license period. After that, the software will only operate in 'Demo Mode', which will perform exactly the same as the licensed version, except you will not be able to switch signal banks in any FPGAs.

5.2. Target JTAG Connection

Since FPGAView for Xilinx automatically uses the Xilinx Platform Cable USB programming cable, no configuration is required.

Clicking the 'JTAG' button in the main window causes FPGAView to close and reopen the Platform Cable USB cable and reacquire the device indentifiers.

The devices connected to the JTAG chain will be automatically scanned and displayed in the 'Debug Cores' treeview list. Since this is the first time you are running FPGAView, you must identify the .clam file which is associated with each device. The .clam file is created with the FS2 OCIGen utility.

Each device instance in the Debug Core list will be shown in red, indicating its associated .clam file is unknown. Click on each instance and a 'File Open' dialog box will be displayed to allow you to find the associated .clam file. If the .clam files are located on another machine, you must have network access to that machine, and the .clam files must be located in a shared folder or subfolder that is accessible to the user running FPGAView. Once the .clam file is located, FPGAView will remember the association so this operation will not have to be repeated. The device name and the debug cores contained in that device will now be displayed in black.

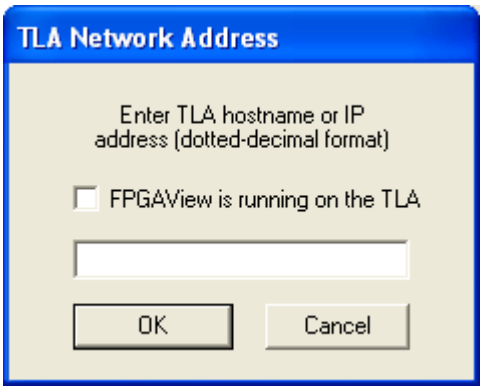
If you need to select a different .clam file for one of the device instances, click the device instance again and select a new .clam file.

Click on any Debug Core (symbol) to show the signals and banks it controls.

Click on any bank item ('B' symbol) to select that bank, or use the 'Bank' drop-down list box.

5.3. TLA Connection

Click on the 'TLA' button on the main window to configure the TLA connection. The following window will be displayed:



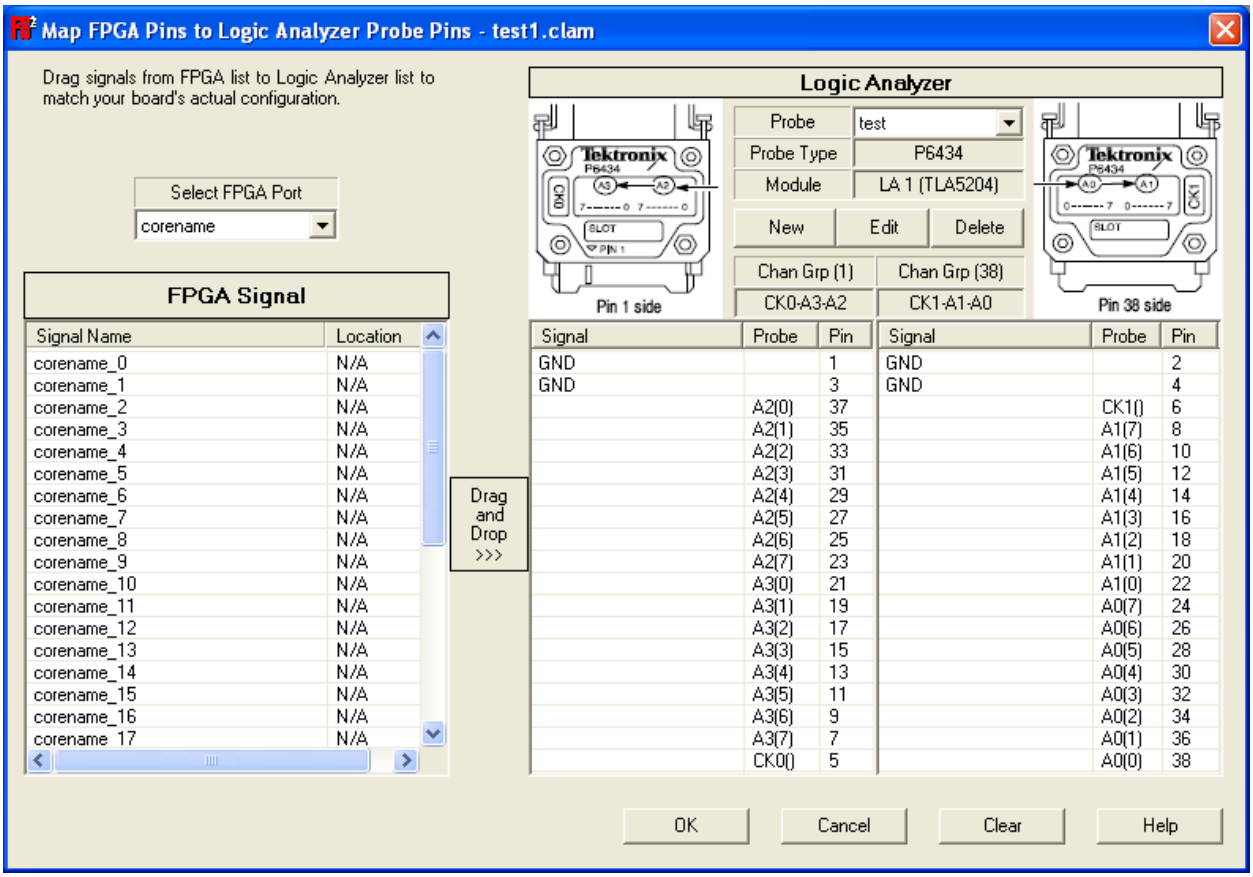
If FPGAView is running on the TLA itself, check the 'FPGAView is running on the TLA' checkbox. Otherwise, enter the IP address of the TLA in the edit window and click OK. Pressing the ESC key will cancel an ongoing attempt to connect to the TLA.

5.4. TLA Probe Configuration

Once you have at least one LAI file identified, you can configure the TLA probes. First create the signal groups on the TLA that you will use to monitor the multiplexed signals connected to the logic analyzer interface outputs. When setting up the TLA signal groups, keep in mind that each logic analyzer interface can be configured for timing mode or state mode, but not both simultaneously. Group your signals accordingly.

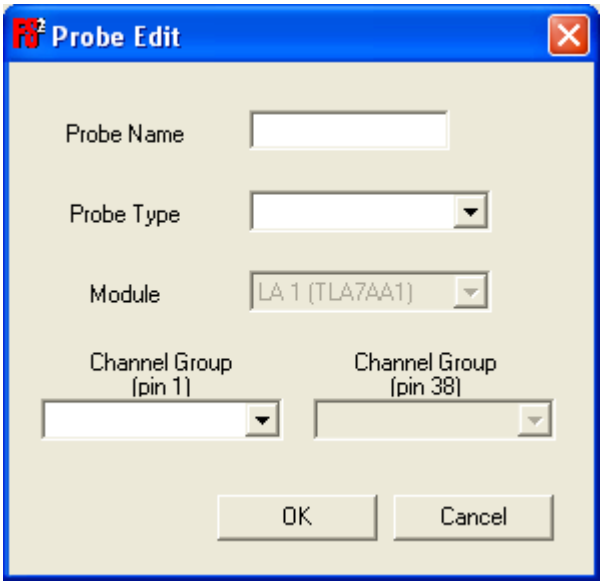
First select one of the FPGA devices in the 'Debug Cores' treeview on the main window by clicking on one of its Debug Cores.

Then, start the Probe Editor in FPGAView by clicking on the 'Probes' button on the main window. The following window will be displayed:



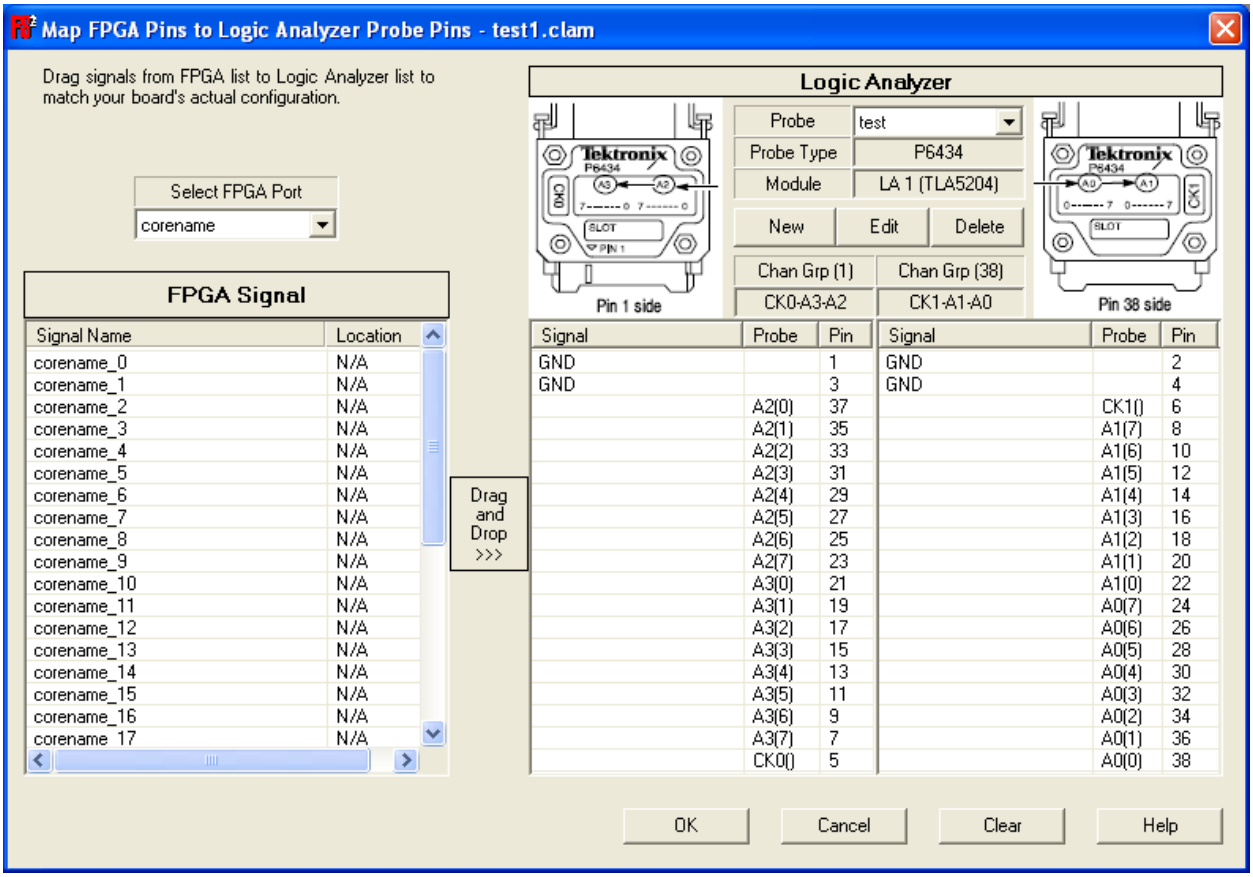
In the upper-left of this window is a combobox containing all the debug cores contained in the device you just selected.

Since you have not defined any probes yet, the following window will be displayed to allow you to define a new probe:



Enter a probe name of your choosing in the 'Probe Name' edit box. If you have multiple modules installed in the TLA, then select the module this probe is connected to from the 'Module' combobox. Then select the probe type from the 'Probe Type' list. Finally, choose the channel group that the probe is inserted into on the TLA from the list labeled 'Channel Group (pin 1)'. If the probe is a 34-signal probe with 2 distinct paddle connectors, also choose the second channel group from the 'Channel Group (pin 38)' list. If the probe is a single 34-pin paddle connector at the TLA end, the second channel group will be chosen automatically. Click OK to complete the probe definition.

After this operation you will see a Probe Mapping window resembling the following figure. Note that if you chose a 17-pin probe, the image and listbox on the right-hand side of the 'Logic Analyzer' section will not be displayed.



On the left side is a list called 'FPGA Signal', containing all the output signals connected to the Debug Core selected in the list box labeled 'Select FPGA Port'. These are the signals you connect to the probe pins. These connections must match the physical layout of the board under test. You may wish to consult the schematics for your board for this operation.

The right-hand side contains the two probe signal definition listboxes. The first column in each will contain the names of the signals you connect from the FPGA Signal list. The second column contains the probe signal hardware names. These names are set when you defined the probe and selected which channel groups the probe cables occupy. The third column contains the probe pin numbers. You can reorder the signals in these lists by clicking on the column headings. Clicking on the 'Probe' column heading will reorder the list so the probe signal names are displayed in ascending order, which will make signal assignment easier, assuming your output signals are in ascending order.

To assign a signal, click on it in the 'FPGA Signal' list, hold down the left mouse button, and drag it over to the Probe signal lists, releasing the mouse button when the cursor is on the correct probe signal line.

To assign several signals at once, click on the first signal, hold down the Shift key and click on the last signal of the group you want to assign with the leff mouse button. Hold the mouse button down and drag the signals to the right side list, and release the mouse button when the cursor is on the first line of the group of probe signals to which this group is to be connected.

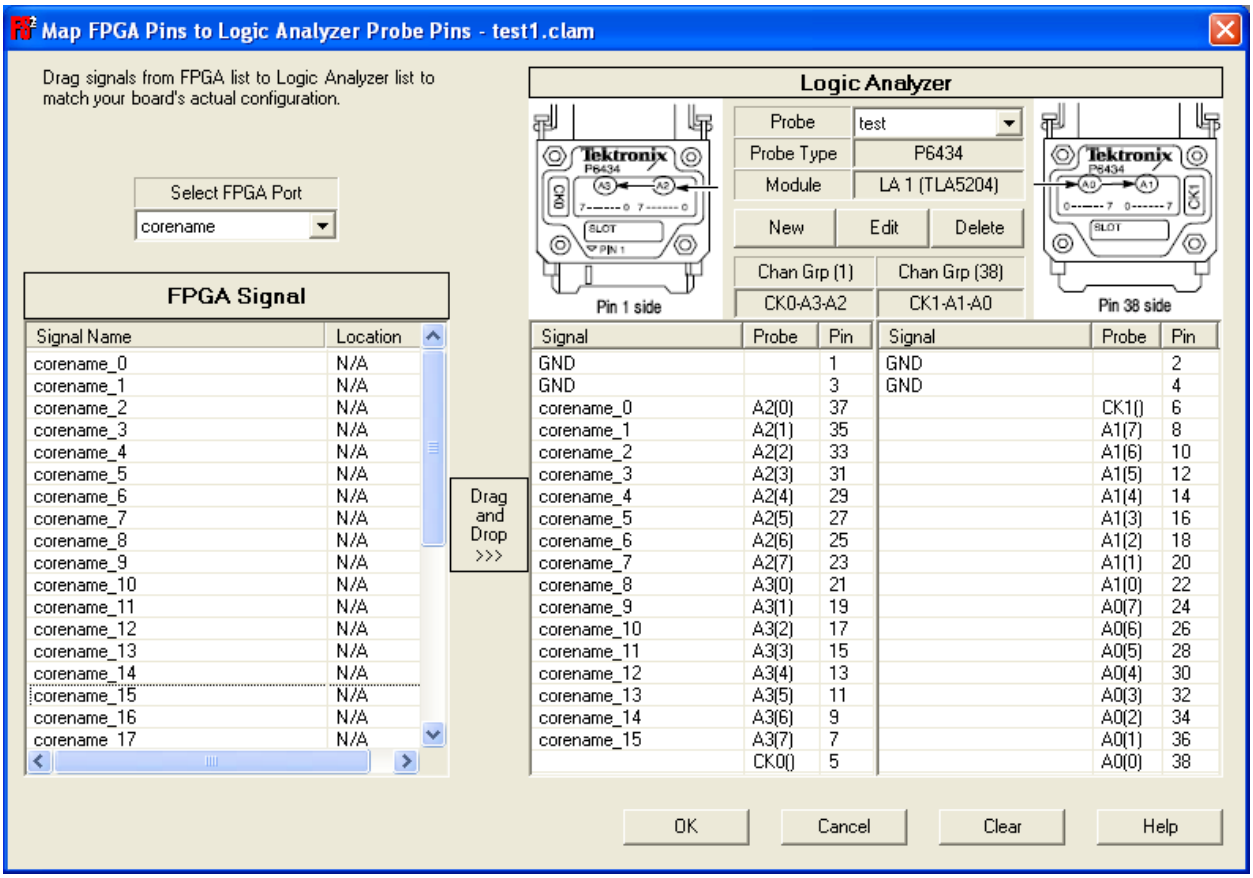
You can continue to map signals from the currently selected device until you are done. To map more devices, click OK to save the current mapping and return to the main window. Then select another FPGA device before returning to the Probe Mapping window to continue mapping more signals.

If you make a mistake during the drag-n-drop operation, you can select the signals which were placed incorrectly (click-then-shift-click) and type the DEL key, or click on the 'Clear' button, to delete those signals from the probe definition. Clicking on the Clear button when no signals are selected will prompt you to delete all signal definitions from the currently selected probe.

You can also drag-and-drop signals over existing signals, which replaces the previous signals with the new signals.

Click on the 'OK' button to save the signal mapping.

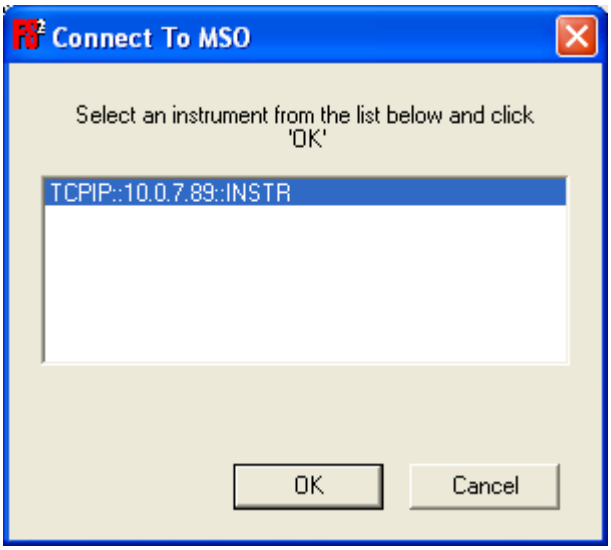
Below is an image of the form after a typical mapping operation has been completed.



5.5. MSO Connection

The MSO is configured and controlled via TekVISA, Tektronix's instrument communications interface. The TekVISA software needs to be installed and configured prior to attempting to use FPGAView with the MSO. Please refer to the TekVISA documentation for detailed instructions. The key point is that TekVISA must be able to locate and identify your MSO before FPGAView will be able to communicate with it.

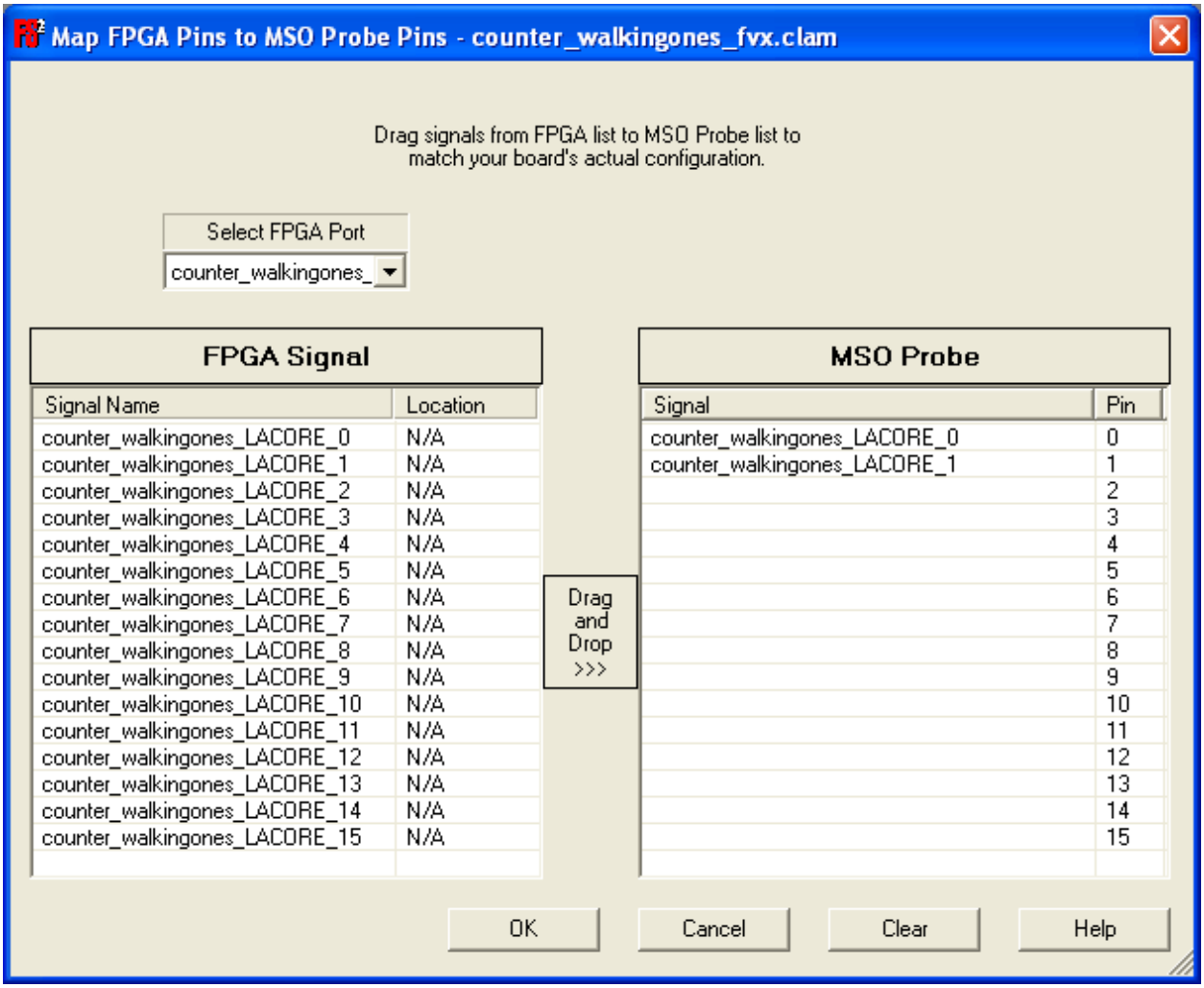
Once TekVISA is set up, start FPGAView with the FPGAView-Xilinx-MSO shortcut. FPGAView will automatically attempt to configure the available MSO's. You may also click the 'MSO' button. This will display the 'Connect to MSO' form.



Select the instrument to which you wish to connect, and click OK.

5.6. MSO Probe Configuration

Clicking on the 'Probes' button displays the 'Map FPGA Pins to MSO Probe Pins' form.



Map FPGA debug core output signals to MSO probe signals by selecting items from the FPGA Signal listbox and dragging them to the correct signals in the MSO Probe listbox, and click 'OK'.

The signals mapped to the MSO Probe can be removed by selecting them from the MSO Probe listbox and clicking the 'Clear' button, or by using the Delete key. Clicking 'Clear' when no signals are selected in the MSO Probe listbox will delete all the signal mappings from the MSO Probe listbox.

Click 'Cancel' if you do not want to keep the mappings. Click 'OK' to keep the mapping.

6. Using FPGAView

6.1. Switching Banks

Having survived the preceeding setup procedures, it is time to use FPGAView for normal operation.

In the main window, in the 'Debug Cores' tree all the FPGA devices which were found by the JTAG scan are displayed with the symbol . The first time a device is identified, it will be displayed in red, to indicate that no (.clam) file has been associated with the device. Click on any device line to display an 'open file' dialog window and navigate to the folder containing your design files. Choose a .clam file to be associated with the device.

Once a .clam file is associated with a device, the banks (B symbol) and signals (symbol) will be displayed automatically in the tree list each time the device is detected.

In the main window, click on any 'debug core' line (symbol) to list all defined banks in the 'Bank' list, and that bank's signals listed in the 'Inputs' list. This action will also enable the GPIO button and GPIO menu item, if the debug core contains GPIO registers (see below.)

Click on the arrow button at the right of the 'Bank' list to display all of the banks contained in the selected logic analyzer interface. Click on the bank to be connected to the TLA probe, and that bank will be selected, and the signal names will be sent to the TLA, which will display them next to the associated signals in any Waveform, Listing, or Setup window which contains the associated probe signal. You may also click on the bank line in the tree list to select the bank and send the signal names to the TLA.

6.2. Making Changes To Your Design

Any time a new design is programmed into the FPGA, you should click the 'Refresh' button in the main window. This will force the program to rescan the JTAG chain and detect all the devices and Debug Cores on the JTAG chain.

6.3. Making Changes to the TLA Signal Groups

Any time a change is made to the TLA signal grouping, FPGAView will automatically rescan the TLA for the new grouping configuration.

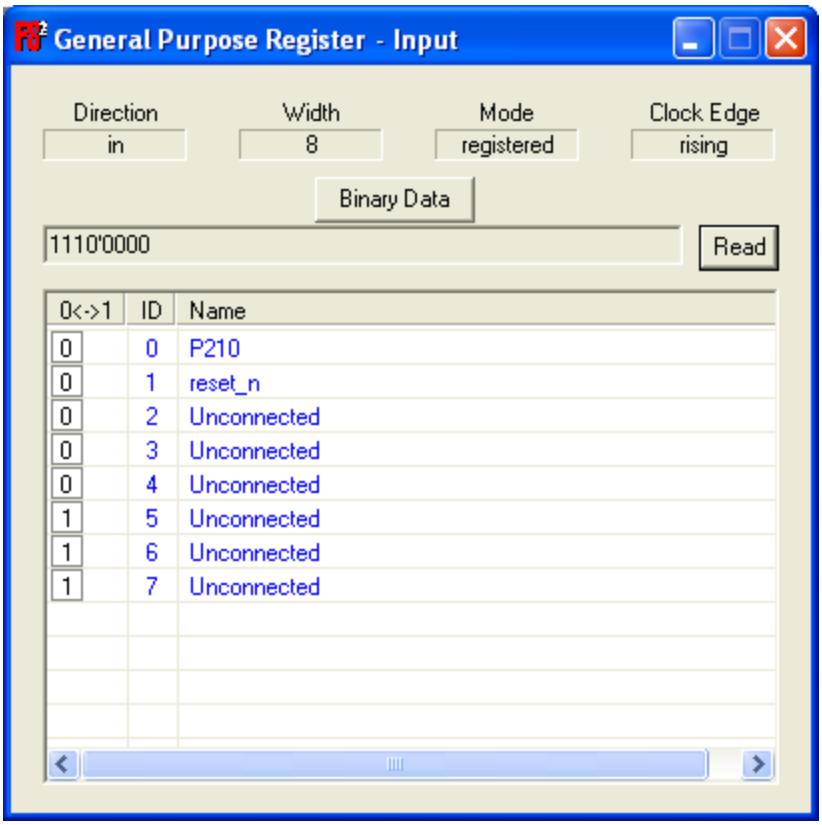
6.4. Associating a New .Clam File

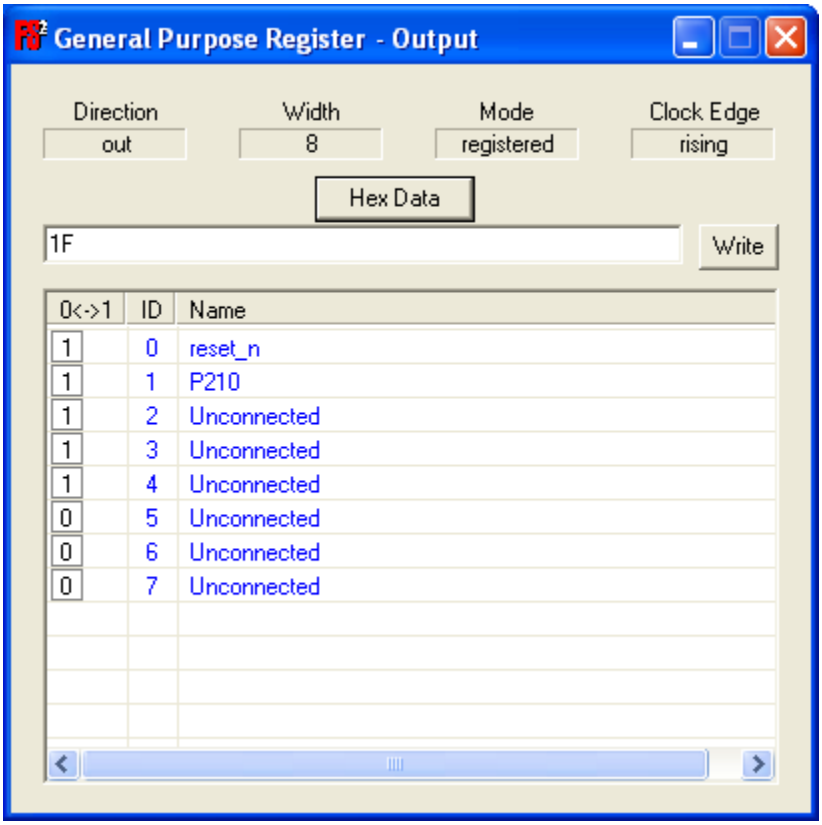
You may associate a new .clam file with a device by clicking on the device in the tree list. This will display a 'open file' dialog which you use to locate and select a new .clam file to associate with the device. If the FPGA device has not been programmed with the .clam file, you will be warned when you try to switch banks.

6.5. General Purpose Registers

Each debug core instance may include one general-purpose input register and one general-purpose output register. Each register may contain 8, 16, 32, or 64 bits of data. When a debug core (symbol) is selected which contains GP registers, the GPIO menu item and GPIO button on the main form become activated. Selecting the GPIO menu item or clicking the the GPIO button will display the GP Input and GP Output forms, show below.

Please note: While either or both GPIO forms are visible, access to the JTAG cable will be locked, and it willl not be possible to use the Xilinx ISE tools to probe or reprogram the FPGA.


General Purpose Register - Input
Direction
in
Width
8
Mode
registered
Clock Edge
rising
Binary Data
1110'0000
Read
0<->1
ID
Name
0 0 P210
0 1 reset_n
0 2 Unconnected
0 3 Unconnected
0 4 Unconnected
1 5 Unconnected
1 6 Unconnected
1 7 Unconnected


General Purpose Register - Output
Direction
out
Width
8
Mode
registered
Clock Edge
rising
Hex Data
1F
Write
0<->1
ID
Name
1 0 reset_n
1 1 P210
1 2 Unconnected
1 3 Unconnected
1 4 Unconnected
0 5 Unconnected
0 6 Unconnected
0 7 Unconnected

Each GP Register form shows the data direction, 'in' or 'out', followed by the register 'width', or number of data bits in the register, the mode, 'registered' or 'unregistered', and the clock edge on which the data is latched ('rising' or 'falling'), for registered mode. These items are all read-only.

Below the attributes is a button used to change the data display format. Clicking this button changes the display format from 'Hex Data', to 'Decimal Data', and finally to 'Binary Data'. These settings are preserved between sessions.

Below the data format button is a textbox containing the data to be written, in the case of the GP output register, or the data read from the device, in the case of the GP input register. This textbox is editable for the GP output register, and typing the Enter key while the cursor is in this textbox causes the value to be immediately written to the device, reflected in the bit listbox, and saved to the fs2.ini file. The output data value will be automatically restored and written to the hardware output register when the GP Output Register form is reopened.

To the right of the data textbox is a 'Write' button, for the GP Output Register form, or a 'Read' button, for the GP Input Register form. Clicking on this button causes the value in the data textbox to be written to the device, for the 'Write' button, or causes the textbox value and bit values to be updated from the device, for the 'Read' button.

At the bottom of the GP register form is the bit listbox. Each bit in the data is displayed as a '0' or '1', inside boxes. Next to the numeric bit value is the bit position 'ID' value, followed by the name of the signal connected to that data bit, displayed in blue text. In the GP Output form, clicking the boxes causes that data bit to change to the opposite value, and changes the text color to red, to indicate that the bit value has changed, but the changed data has not been sent to the hardware output register. Clicking on the 'Write' button updates the hardware output register with the new data, after which the text is displayed in blue again.

The GP Input form is very similar to the GP Output form, except that the bits cannot be changed by clicking on them. Clicking on the 'Read' button causes the bits and textbox to be updated with the value read from the hardware.

7. Saving and Restoring Settings

The menu selections under 'File' include 'Save' and 'Restore' selections. These menu items allow you to save probe definitions and debug core file associations you created during the current session, or restore these settings from a previously saved file. The settings are saved in files with the extension '.fvd' in the folder or your choice.

7.1. Saving Settings

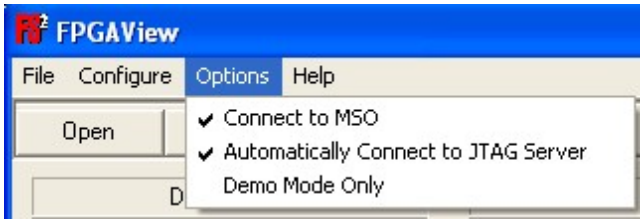
Save your current settings by selecting 'Save' from the 'File' menu. A dialog box will be opened to allow you to choose the folder and file in that folder in which to save the current session's settings. You may type a new name to save the settings in a new file (file extension is not required.)

7.2. Restoring Settings

You may restore previous settings by selecting 'Restore' from the 'File' menu. A dialog box will be opened to allow you to choose the folder and file in that folder from which the saved settings will be obtained.

8. Options

Under the menu item 'Options' are the following items:



8.1. Connect to TLA or MSO

This option, when checked, will cause FPGAView to attempt to connect to the TLA or MSO at startup, once the TLA or MSO connection information has been configured. Pressing the ESC key during the connection attempt will cancel the connection attempt. You may click the 'TLA' or 'MSO' button on the main form, or click the menu item Configure > TLA Communications… or Configure > MSO Communications… to configure the TLA network address or select the MSO instrument and attempt to reconnect.

8.2. Automatically Connect to JTAG Server

This option, when checked, will cause FPGAView to attempt to connect to the JTAG server at startup, once the JTAG server information has been configured.

8.3. Demo Mode Only

This option, when checked, will operate FPGAView in demo mode. In Demo Mode, no attempt will be made to connect to the JTAG server, and therefore no bank switching will occur. The signal names will be sent to the TLA or MSO though, so this mode is useful for demonstrating the program when no FPGA hardware is available or the JTAG connection is not set up.

Deselecting Demo Mode, by unchecking the Demo Mode menu item, will revert the program back to normal mode, and will cause the JTAG connection to be reestablished immediately. If a valid license is not found, the program will require that a valid license key be obtained and entered before Demo mode is turned off.

If a JTAG connection cannot be established, the program will revert to Demo mode once again, and it will be necessary to turn Demo mode off manually before the program will try to reestablish the JTAG connection.

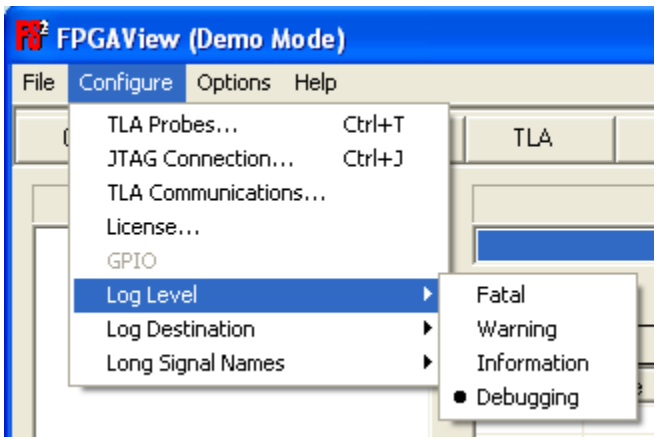
9. Error Logging

FPGAView logs significant events to the Application Event Log. To control the amount and type of messages logged, use the menu 'Configure…Log Level'. The four logging levels are:

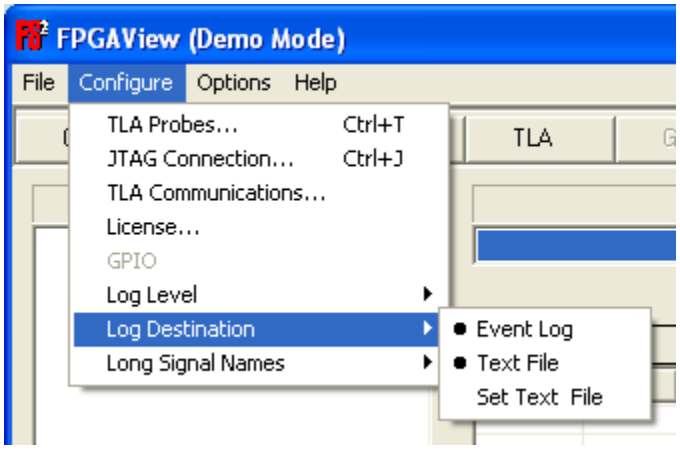
- Fatal – Only those errors that prevent FPGAView from continuing to operate are logged.
- Warning – Both fatal errors and those errors that may indicate a problem with FPGAView are logged.
- Information – Fatal, Warning and informative messages are logged.

- Debugging – Fatal, Warning, Informative and messages intended for detailed debugging are logged. Use this selection to obtain information for use by FS2/MIPS technical support personnel.

Normally, you should set the logging level to Fatal or Warning, to prevent the event log from filling too quickly.

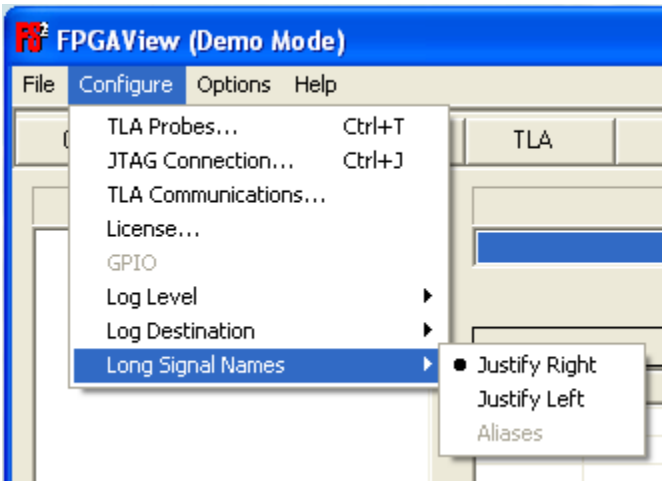


Use the 'Log Destination' to enable logging to the Application Event Log (use Administrative Tools Event Viewer to view these,) or to a text file, or both. Use the 'Set Text File' menu item to change the file to which the log messages are written.



10. Long Signal Names

The Configure menu contains a submenu to configure the display of long signal names. Some signal names generated by the ISE may be too long to display in a meaningful way on certain instruments. For instance, the MSO can only display 32 characters for a signal name. In such situations, it is sometimes better to display the trailing part of a signal name. The 'Long Signal Names' configuration menu allows you to choose either left-justified signal name display (default) with the first character in the name always visible, or right-justified signal name display, with the last character in the name always visible.



11. Help

The Help button and the Help menu item ‘FPGAView Help’ display this document.

The Help menu item ‘About’ displays the software version, and a web link to FS2.com.