LAUTERBACH
DEVELOPMENT TOOLS

Integration for Xilinx ISE

MANUAL

Integration for Xilinx ISE

Integration for Xilinx ISE

Version 06-Jun-2024

Introduction

In some scenarios it is necessary to use different tools via the same JTAG connector. One such scenario is using Xilinx iMPACT and ChipScope for configuration and error detection in an FPGA together with Lauterbach TRACE32 for debugging the software.

To avoid switching between different cables on the JTAG port, one can use the Lauterbach debugger hardware with Xilinx tools. This requires configuring the Lauterbach debugger and the Xilinx tools. This article describes the necessary configuration steps.

NOTE:	Currently the use of Xilinx tools via TRACE32 debugger hardware is only supported on Windows hosts. Please see section "**Compatibility**" for details regarding the supported versions of Xilinx ISE.

Getting Started

For using Xilinx tools via the Lauterbach TRACE32 debug interface some configuration of TRACE32 and the Xilinx tools is required. The **configuration** consists of the following steps:

- **Install the Microsoft Visual C++ 2008 SP1 Redistributable Package** which is available from Microsoft free of charge**.** It is important to use the **SP1** version, otherwise the Xilinx tools (e.g. iMPACT, ChipScope) will fail to load the plug-in's DLL.
- **Configure TRACE32** to allow 3rd-party tools to use it as JTAG back-end by adapting the `config.t32` file or changing the settings in the T32Start tool.
- **Configure Xilinx ISE** so that Lauterbach debug interface hardware is known to the Xilinx tools. This includes copying the plug-in DLL and an accompanying XML file to a predefined location.

For **starting a session** in which the Xilinx tools use the TRACE32 debug interface proceed as follows:

- **First** start an instance of **TRACE32 PowerView** and connect it to target. A running process of TRACE32 is required to service requests by the Xilinx tools to access the target's JTAG port.
- Start the Xilinx tool you want to use (e.g. iMPACT or ChipScope) and configure it by selecting the "Lauterbach cable" that was registered with the ISE installation.

Details on these steps are found in the following sections of this document.

For considerations on the use on 64bit hosts, refer to section "**Usage on 64 bit machines**".

For information about common problems, refer to section "**Troubleshooting**"

Usage on 64-bit Machines

When using TRACE32 as JTAG back-end, the **Xilinx tool's version** (32 bit vs. 64 bit) **must match that of the TRACE32 plug-in DLL** used. Therefore when using e.g. ChipScope 64bit, also the 64 bit version of the TRACE32 plug-in needs to be installed. The Xilinx tools will look for the plug-in in the location appropriate according to their version. Note that both versions of the TRACE32 plug-in can be installed in parallel because they reside in different locations.

Up to and including ISE13, Lauterbach only provides a 32 bit version of the plug-in. The 32 bit version is fully functional also on 64bit machines (e.g. Win7 64 bit) but needs to be used with 32 bit Xilinx tools. The warning printed by the 32 bit tools running on a 64 bit host can be safely ignored.

Starting with ISE14, Lauterbach provides both 32 bit and 64 bit versions of the plug-in DLL.

Note, however, that the **TRACE32 PowerView GUI** (e.g. `t32marm.exe`) is **not required** to match the plug-in version.

Configuring TRACE32

There are two distinct ways to configure TRACE32 as JTAG-back-end for 3rd-party tools. Both **activate** the so-called **TRACE32 Remote API,** i.e. the feature to invoke certain functions inside TRACE32 by remote processes that bind to TRACE32 via a socket interface. To enable the feature

- either configure the **T32Start** tool

```
core->interfaces->APIPort yes 20000
```

- or manually add the following block to the `config.t32` file:

```
                              <- mandatory blank line!
RCL=NETASSIST
PORT=20000
PACKLEN=1024
                              <- mandatory blank line!
```

The example assumes `PORT 20000` for communication with TRACE32 but any other free UDP port can be used as well. The same port number must be used in the configuration of TRACE32 and the Xilinx tools.

Configuring Xilinx ISE 11 and ISE 12

For using Lauterbach TRACE32 debugger hardware as back-end for iMPACT and ChipScope, it is necessary to configure Xilinx ISE so it recognizes the “Lauterbach cable”. When invoking a specific tool, it is also necessary to explicitly choose the “Lauterbach cable” for target access.

For the purpose of this description, it is assumed that Xilinx ISE is installed on a MS Windows machine at `c:/Xilinx/12.1/`. Paths have to be adapted to your much machine. The configuration of ISE is done by

- Installing the **Microsoft Visual C++ 2008 SP1 Redistributable Package** which is available from Microsoft free of charge**.** It is important to use the **SP1 version**, otherwise the Xilinx tools will fail to load the plug-ins DLL.
- Downloading the plug-in from `https://www.lauterbach.com/support/static`**`/XilinxCsePlugin_11_12.zip`**
- Extracting the ZIP file to `c:/xilinx/12.1/ISE_DS/ISE/lib/nt/plugins/`

This will create the following directory structure:

```
$ cd c:/Xilinx/12.1/ISE_DS/ISE/lib/nt/plugins/
~/Xilinx/12.1/ISE_DS/ISE/lib/nt/plugins $ find .
./Lauterbach
./Lauterbach/lauterbacht32plugin
./Lauterbach/lauterbacht32plugin/lauterbacht32plugin.dll
./Lauterbach/lauterbacht32plugin/lauterbacht32plugin.xml
```

Test the installation by first invoking TRACE32, connecting it to a target board and powering it up. Then Invoke ChipScope and check if it correctly detects the target’s JTAG chain.

Configuring Xilinx ISE 13 and ISE 14

On a Windows machine with an installation of ISE13 or ISE14, the plug-in is installed by the following steps:

- Install the **Microsoft Visual C++ 2008 SP1 Redistributable Package,** which is available from Microsoft free of charge**.** It is important to use the **SP1 version**, otherwise the Xilinx tools will fail to load the plug-in's DLL.
- Download the plug-in from
  https://www.lauterbach.com/support/static/XilinxCsePlugin_13.zip or
  https://www.lauterbach.com/support/static/XilinxCsePlugin_14.zip
- Extract the ZIP-archive and copy the contents to one of the following locations, depending on whether it shall be available to all users or only to specific users:

```
%ALLUSERSPROFILE%\.cse\<platform>\<ise version>\plugins\
%USERPROFILE%\.cse\<platform>\<ise version>\plugins\
```

The parameter platform is set as follows
`<platform>` = `nt` for 32bit installations of Windows 7 or Windows XP
= `nt64` for 64bit installations of Windows 7 or Windows XP

For the first plug-in you install, you may have to create some of the directories.

Example: In an installation for ISE14.1, the DLL and the XML files are found at the following locations:

```
// 32bit
%PLUGINS32BIT%=C:\ProgramData\.cse\nt\14.1\plugins
%PLUGINS32BIT%\Lauterbach\Trace32CsePlugin_14\Trace32CsePlugin_14.dll
%PLUGINS32BIT%\Lauterbach\Trace32CsePlugin_14\Trace32CsePlugin_14.xml

// 64 bit
%PLUGINS64BIT%=C:\ProgramData\.cse\nt64\14.1\plugins
%PLUGINS64BIT%\Lauterbach\Trace32CsePlugin_14\Trace32CsePlugin_14.dll
%PLUGINS64BIT%\Lauterbach\Trace32CsePlugin_14\Trace32CsePlugin_14.xml
```

ATTENTION: On machines that are configured for non-english language, there may be multiple directories that are "tempting" to put the plug-ins into e.g.

```
c:\Documents and Settings\All Users\       (English)
c:\Dokumente und Einstellungen\All Users   (German)
```

Be sure to use one of the valid directories. These can by obtained by executing the following commands in a command shell:

```
echo %USERPROFILE%
echo %ALLUSERSPROFILE%
```

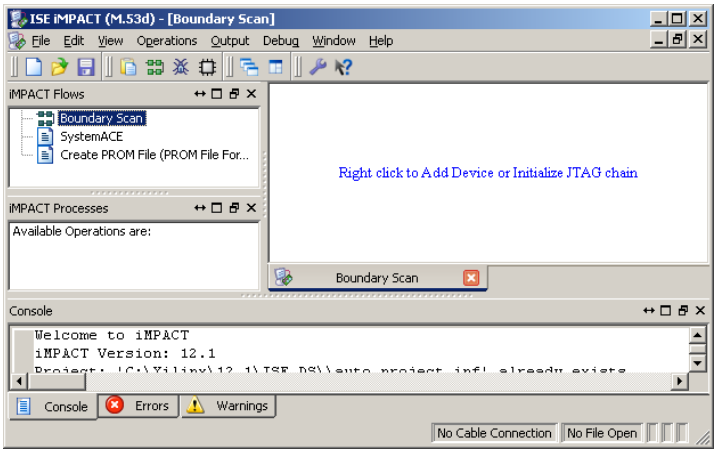
Using Xilinx iMPACT via the TRACE32 Debug Interface

Before using Xilinx iMPACT via the TRACE32 debug interface, be sure to configure TRACE32 and the ISE installation as described in the sections "**Configuring TRACE32**", "**Configuring Xilinx ISE 11 and ISE 12**" and "**Configuring Xilinx ISE 13 and ISE 14**".

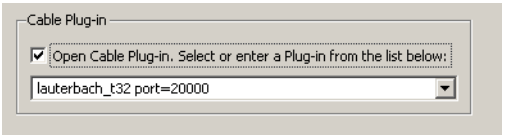
Then perform the following steps in this order:

1. **Start TRACE32** because a running instance of TRACE32 is required for target access.

2. Start iMPACT. Cancel the "Welcome to iMPACT" dialog and **double-click on "Boundary Scan"**. **Only after doing this**, the menu **Output** will show the menu items required for selecting the TRACE32 debug interface.



3. Choose **Output** menu -> **Cable Setup** and configure the "Cable Plug-in" box with `lauterbach_t32 port=20000` and click **OK**.



4. Test the plug-in by doing a scan of the target's scan chain (by right-clicking into the window).

NOTE:	The port number 20000 must coincide with the configuration of the TRACE32 remote API interface.

5. Test iMPACT with the command File -> Initialize Chain. iMPACT should detect the JTAG chain of the target.

In case of problems, please refer to section "**Troubleshooting**".

Using Xilinx ChipScope via the TRACE32 Debug Interface

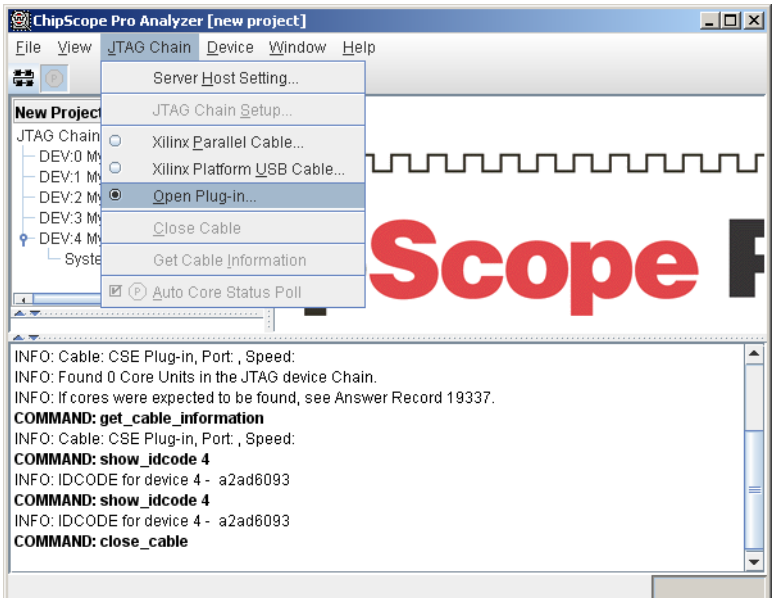
Before using Xilinx ChipScope via the TRACE32 debug interface, be sure to configure TRACE32 and Xilinx ISE, so it recognizes TRACE32 as a valid option for a "debug cable". For details, see sections "**Configuring TRACE32**" and "**Configuring Xilinx ISE 11 and ISE 12**" and "**Configuring Xilinx ISE 13 and ISE 14**".

Then perform the following steps in this order:

1. **Start TRACE32** because a running instance of TRACE32 is required for target access.
2. **Start ChipScope**.
3. Enable the use of a Chipscope plug-in for connecting to a custom JTAG cable through the menu item `JTAG Chain -> Open Plug-in`. Select the TRACE32 debug interface with the plug-in parameters

   ```
   lauterbach_t32 port=20000.
   ```

   This assumes, that TRACE32 also was configured for port 20000.



After doing this configuration, ChipScope should auto-detect the scan chain of the target.

In case of problems, please refer to section "**Troubleshooting**".

Compatibility

As of 12/2012 the use of TRACE32 as JTAG back-end for Xilinx tools is **only supported for the Windows platform** (Windows XP, Windows 7).

All versions of TRACE32 from 2011 or later are compatible with this feature.

Xilinx ISE 11 is the first version to introduce the plug-in mechanism for using 3rd-party JTAG hardware.

- Xilinx ISE 11 and Xilinx ISE12 use the same plug-in.
- Xilinx ISE13 requires a specific plug-in and the location for installing the plug-in changed.
- Xilinx ISE14 requires a specific plug-in. Lauterbach provides both 32 bit and 64 bit versions of the CSE plug-in.

Note on 64 bit machines:	The version of the plug-in (32 bit vs. 64 bit) must match the version of the tools (Chipscope, Xilinx) used. Up to and including ISE13, Lauterbach only provides a 32bit version of the plug-in. However, the 32bit version is fully functional also on 64bit machines (e.g. Win7 64bit) but needs to be used with 32bit Xilinx tools. The warning printed by the tools can be safely ignored. Starting with ISE14, Lauterbach provides both 32bit and 64bit versions of the plug-in DLL.

The following table gives detailed information on the versions of ISE and the required plug-ins:

ISE version	Plug-in Name	Status
14	Trace32CsePlugin_14	The plug-in is known to work up to ISE14.6 32bit and 64bit versions are available.
13	XilinxCsePlugin13	The plug-in is known to work up to ISE 13.1.
12	XilinxCsePlugin11_12	The plug-in is known to work up to ISE 12.4.
11	XilinxCsePlugin11_12	ChipScope Pro is known to work with all ISE 11.x versions. iMPACT does not support the plug-in mechanism.
<= 10		No plug-in mechanism for 3rd-party debug interfaces.

- Some typical status and error messages given by **ChipScope** when attempting to open TRACE32 as debug cable:

  - **The TRACE32 PowerView GUI was not running**:

```
COMMAND: open_target lauterbach_t32 port=20000
INFO: Started ChipScope host (localhost:50001)
INFO: Successfully opened connection to server: localhost:50001
(localhost/127.0.0.1)
ERROR: Failed to open lauterbach t32. See message(s) above.
```

  - The TRACE32 PowerView GUI was running, but the **dongle was not connected to a target** or **the target was powered off**:

```
COMMAND: open_target lauterbach_t32 port=20000
INFO: Started ChipScope host (localhost:50001)
INFO: Successfully opened connection to server: localhost:50001
(localhost/127.0.0.1)
ERROR: No devices detected while scanning the JTAG chain
ERROR: Failed detecting JTAG device chain
ERROR: Opened lauterbach_t32 but failed to detect JTAG Chain.
...
```

  - Successfully opened TRACE32 as debug cable:

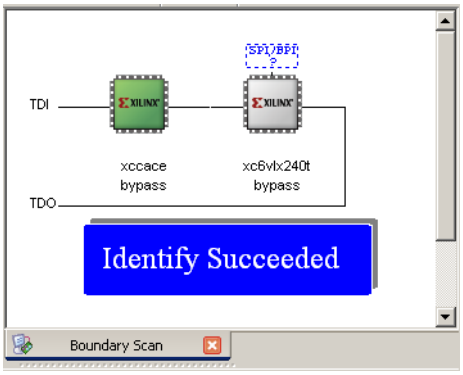
```
COMMAND: open_target lauterbach_t32 port=20000
INFO: Started ChipScope host (localhost:50001)
INFO: Successfully opened connection to server: localhost:50001
(localhost/127.0.0.1)
INFO: Successfully opened lauterbach t32
INFO: Cable: CSE Plug-in, Port: , Speed:
INFO: Found 0 Core Units in the JTAG device Chain.
INFO: If cores were expected to be found, see Answer Record 19337.
```

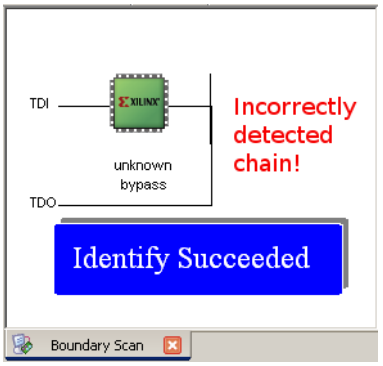
- Be sure you installed the Microsoft Visual C++ 2008 SP1 Redistributable Package which is available from Microsoft free of charge. It is important to use the SP1 version, otherwise iMPACT will fail to load the plug-ins DLL.
- **iMPACT reports "WARNING: iMPACT:923 - Can not find cable, check cable setup!"**:

  Be sure to correctly configure TRACE32 **and** ISE as described in this document.
  Be sure that an instance of **TRACE32 is already running**, when starting the Xilinx tools.
  When using multiple installations of ISE or TRACE32 be sure to configure the correct installation.
- **iMPACT does not detect error conditions**:

  When working via the TRACE32 debug interface, iMPACT may ignore error conditions (like no power on the target). Also see "iMPACT does not correctly recognize the scan chain".
- **iMPACT does not correctly recognize the scan chain**:

Because iMPACT ignores error conditions when working over the TRACE32 debug interface, it may detect the scan incorrectly as shown in the example. The left picture shows a valid scan chain detection (the target was powered up). The right picture shows the result of a scan chain detection on the same target, but with the target powered down.





- **iMPACT always crashes on startup**:

  With EDK12.1 sometimes ISE creates an invalid project file. If you enabled the option

  Edit -> Preferences -> “Automatically load the most recent project file when iMPACT starts”,

  iMPACT will crash at each startup, attempting to interpret the damaged file. The only way to get iMPACT to work again is to delete the ISE project file. It has the extension “`.ipf`” and should be located at `$TOOLDIR/ISE_DS/auto_project.ipf`.

  After starting iMPACT successfully, disable the option to auto-load the previous project file.

Contacting Support

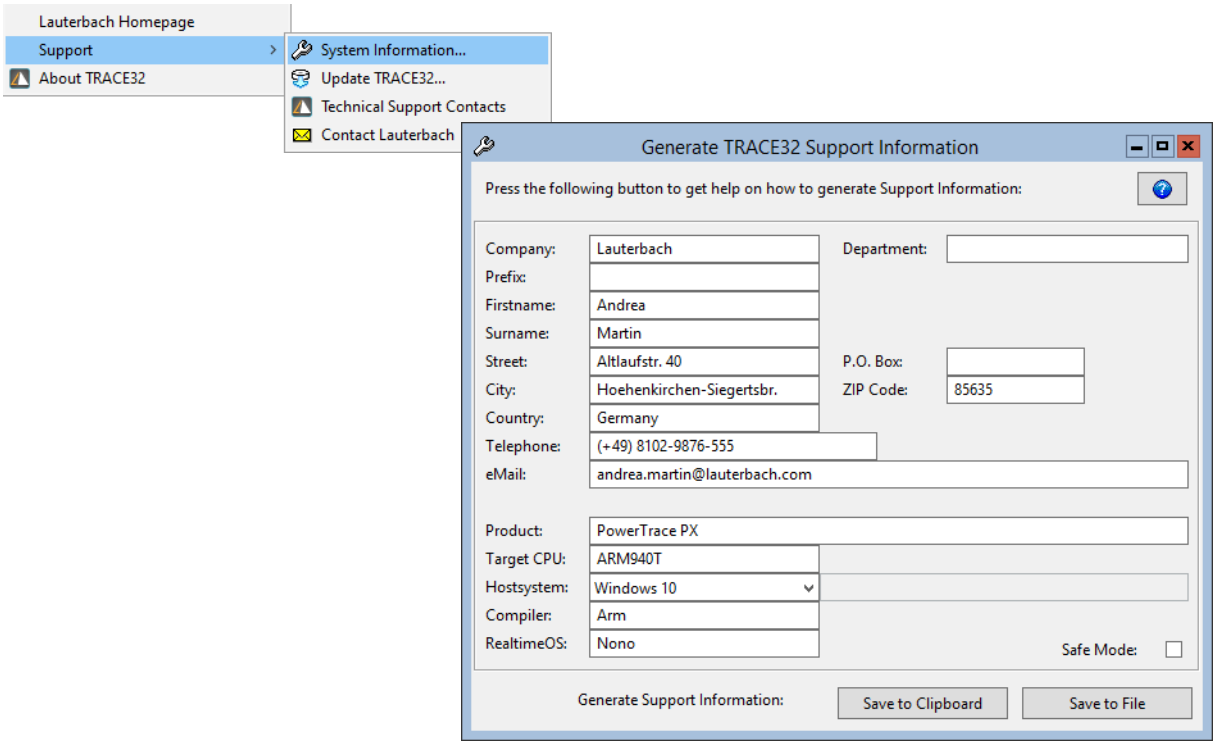
Use the Lauterbach Support Center: https://support.lauterbach.com

- To contact your local TRACE32 support team directly.
- To register and submit a support ticket to the TRACE32 global center.
- To log in and manage your support tickets.
- To benefit from the TRACE32 knowledgebase (FAQs, technical articles, tutorial videos) and our tips & tricks around debugging.

Or send an email in the traditional way to support@lauterbach.com.

Be sure to include detailed system information about your TRACE32 configuration.

1. To generate a system information report, choose **TRACE32** > **Help** > **Support** > **Systeminfo**.



NOTE:	Please help to speed up processing of your support request. By filling out the system information form completely and with correct data, you minimize the number of additional questions and clarification request e-mails we need to resolve your problem.

2. Preferred: click **Save to File**, and send the system information as an attachment to your e-mail.
3. Click **Save to Clipboard**, and then paste the system information into your e-mail.