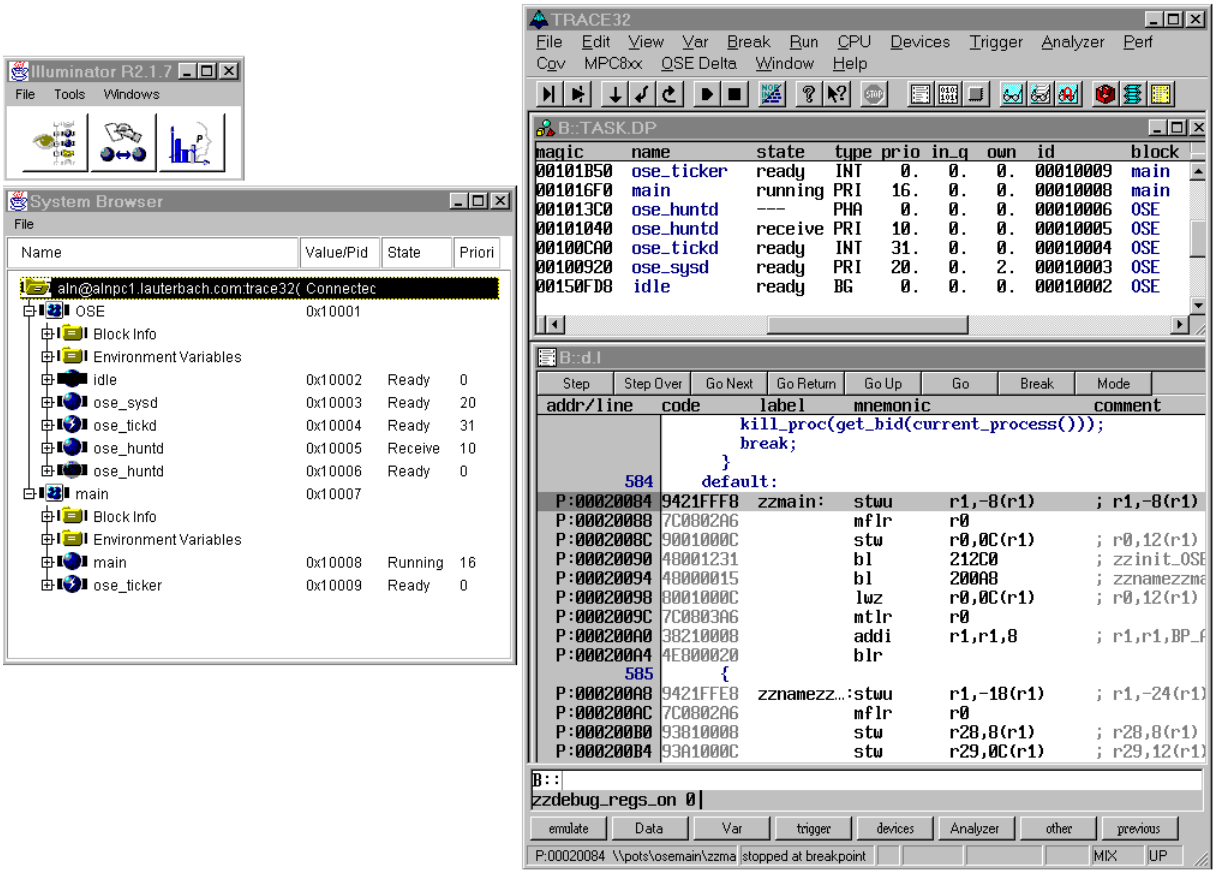
Integration with OSE Illuminator (deprecated)

Integration with OSE Illuminator (deprecated)

Version 18-Nov-2020

06-Jul-20 This manual will be removed with the date of publication from DVD 02/2021.
The manufacturer of Illuminator stopped support a long time ago and
Lauterbach will stop support for the integration with DVD 02/2021.



Requirements

This interface integrates the OSE Illuminator (Enea OSE Systems AB) and TRACE32. Current supported is:

OSE Illuminator version 2.1.7 or later

CPUs: PowerPC 860

RTOS: OSE Delta

Hosts: Windows95, Windows NT, UNIX

Brief Overview of Documents for New Users

Architecture-independent information:

- **"Debugger Basics - Training"** (training_debugger.pdf): Get familiar with the basic features of a TRACE32 debugger.
- **"T32Start"** (app_t32start.pdf): T32Start assists you in starting TRACE32 PowerView instances for different configurations of the debugger. T32Start is only available for Windows.
- **"General Commands"** (general_ref_*<x>*.pdf): Alphabetic list of debug commands.

Architecture-specific information:

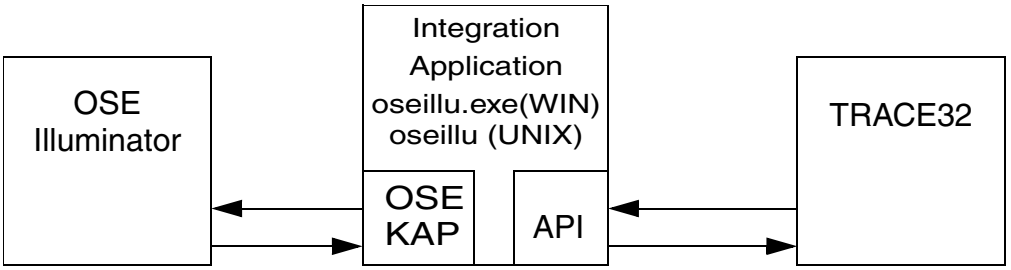
- **"Processor Architecture Manuals"**: These manuals describe commands that are specific for the processor architecture supported by your debug cable. To access the manual for your processor architecture, proceed as follows:

  - Choose **Help** menu > **Processor Architecture Manual**.

- **"OS Awareness Manuals"** (rtos_*<os>*.pdf): TRACE32 PowerView can be extended for operating system-aware debugging. The appropriate OS Awareness manual informs you how to enable the OS-aware debugging.

Operation Theory

The interface consists of an additional application. It translates the communication between the TRACE32 API Interface and the OSE KAP Interface which communicates with the OSE Illuminator. Requests to TRACE32 are sent via sockets. The OSE Illuminator also sends and reseives its requests via sockets. Therefore WINSOCK or BSD socket (UNIX) has to be installed.



NOTE:	This integration uses internally the **TRACE32 Remote API**. The Remote API has **restrictions** if TRACE32 runs in demo mode. Please see there for further details.

Restrictions to the functionality of OSE Illuminator

When using the integration to the OSE Illuminator debugging of the system is only possible in freeze mode. **Freeze mode** means that interaction between TRACE32 and the OSE Illuminator can only take place when the system is completely halted. The OSE Illuminator handles the freeze mode target as it handles an ordinary run mode target and you are free to mix run mode targets with freeze mode targets. But there are some limitations when running the OSE Illuminator in freeze mode:

- The event-action system is not supported.
- CPU profiling is not supported.
- Processes or Blocks can not be started, killed or stopped.

Installation

Follow these steps to configure the OSE Illuminator extension for TRACE32:

1. **For TRACE32:**

   Close TRACE32.

   Edit your "**config.t32**" file: Add between two empty lines the line "**RCL=NETASSIST**", if not already there.

   Start TRACE32 again.

2. **For OSE Illuminator:**

   Start OSE Illuminator.

   In the OSE Illuminator select **File > Preferences** and open the **Connection** tab.

   Check **Broadcast Enable**.

   Set **Broadcast Port** to 50000 (default).

   Set **Interface IP Address** to Default.

3. Copy the Integration Application oseillu.exe(WIN) or oseillu (UNIX) into the system directory of TRACE32.

Startup Sequence

Keep to the following startup sequence to secure the correct setup of the OSE Illuminator integration:

1. Switch on power on TRACE32 and your target.
2. Start the TRACE32 display driver.
3. Load your application into TRACE32.
4. Run until the OSE Delta Kernel is initialized.
5. Start the OSE Illuminator.
6. Open a System Browser window in OSE Illuminator.
7. Start the Integration Application and a node should pop up in the System Browser window of the OSE Illuminator.
8. Connect to TRACE32 by expanding the node of target
   *<user_name>***@***<pc_name>***.***<domain_name>***:trace32(***<id>***)**
   in OSE Illuminator.
9. After the node has expanded (Connected) try to open the block node to see the tasks.
10. Do run and break in TRACE32 and update the OSE Illuminator window to see what has changed.

Menu Commands (WINDOWS)

The oseillu.exe is normally just a backround application for translating the messages from OSE Illuminator to TRACE32 commands. For the handling of the TCP/IP connection to OSE Illuminator there is also a menu Control Illuminator. This chapter describes the menu items.

Connect to Illuminator

This command connects the TRACE32 to the OSE Illuminator.

Normally the OSE Illuminator connection is established when starting the oseillu.exe. But for some reason it could be possible that during the work the connection gets lost. Connect again to the OSE Illuminator by using this menu item and the …trace32(*<id>*) target will pop up again in the OSE Illuminator. If not please see **Troubleshooting**.

Disconnect to Illuminator

This function terminates the connection to the OSE Illuminator.

When closing the Integration Application this is done automatically. If you want to load a different application into TRACE32 you have to disconnect the old application from the OSE Illuminator first. Afterwards you will connect to OSE Illuminator again.

Configuration

Set the port number for the connection to the OSE Illuminator.

If the default port number (50000) is already used by a different application for communication you have to select a different port number for the Illuminator connection. In this example we use 60000 as new port number:

For OSE Illuminator:
In the OSE Illuminator disconnect all targets, select **File > Preferences** and open the **Connection** tab.
Set **Broadcast Port** to 60000.
Press the **Apply** button and close the window.

For the Integration Application:
Select **Configuration** and set **Broadcast Port** to 60000.
Press the **OK** button and do **Connect to Illuminator** to reconnect to OSE Illuminator.

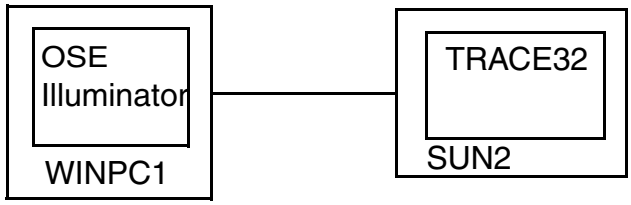
The oseillu is just a backround application for translating the messages from OSE Illuminator to TRACE32 commands. For the handling of the TCP/IP connection to OSE Illuminator there are some options to configure the connection correctly.

Format: **oseillu** [**-t32d 1**] [**-t32n** *<node>*] [**-t32i** *<port_number>*]

Use the options the following way:

-t32d 1 Switch on the debug output.

-t32n *<node>* Configure the host node where TRACE32 application is running.
e. g. oseillu -t32n sun2



If you don't use this option the default ist **localhost**. That means TRACE32 application is running on the same machine as the OSE Illuminator.

-t32i *<port_number>* Configure the port on the host where OSE Illuminator expects a connection.
The default port is 50000.

Working with the OSE Illuminator extension

OSE Illuminator integration tracks the current situation of the emulator. It does not recognize any action done in the display driver. That means: If you perform actions in the display driver OSE Illuminator will show you the situation when you last synchronized the view. Performing an **Update** in OSE Illuminator will get you back to the current view.

If you are resetting and reloading your target in TRACE32 you have to do an **Update** to see the correct current situation of the emulator in OSE Illuminator.

If you quit the Integration Application the target will automatically get to “Not Connected” state. After a while it will also disappear in the symbol browser window of OSE Illuminator.

Use OSE Illuminator and TRACE32 as usual.

Troubleshooting

Integration application can not connect to TRACE32

Does the config.t32 file hold the line **RCL=NETASSIST**?
Is Winsocket installed?
Is TRACE32 open?

New Target does not pop up in OSE Illuminator

Does the OSE Illuminator use the same port number as the Integration Application?
Have you set **Broadcast enable** in OSE Illuminator?
Is another target with the same name of your user already open in OSE Illuminator?

Known Bugs

If for any reason the connection from OSE Illuminator to the Integration Application is broken the target stays in the OSE Illuminator with state “Not Connected”. It takes a while till the OSE Illuminator updates its display and takes this target out of its target list. Within the same session of the Integration Application the **Connect to Illuminator** will not work until the old target is out of the list. Speed up the taking out of the target throught choosing **Update** or **Disconnect Target** in the OSE Illuminator.

Sometimes **Update** or **Disconnect Target** does not work for taking a “Not Connected” target out of the list in OSE Illuminator. To refresh the target list close OSE Illuminator and start it again.