

Simulating Visual IP Models with the NC-Verilog, Verilog-XL, VCS, or ModelSim (UNIX) Simulators

You can use the Visual IP software from Innoveda with Altera-provided models to simulate Altera intellectual property (IP) cores in third-party VHDL and Verilog HDL simulators. The following simulators support Visual IP models: Model Technology ModelSim, Cadence Leapfrog, Verilog-XL, NC-Verilog, and NC-VHDL, and Synopsys VCS and VSS. Altera distributes the Visual IP software for the end user and Visual IP models of Altera IP functions. This white paper describes how to use the Visual IP models with the NC-Verilog, Verilog-XL, VCS, and ModelSim simulators for UNIX workstations.

System Requirements

To run the Visual IP software, you need a UNIX workstation that meets the following system requirements:

- At least 32 MBytes of RAM
- Solaris 2.5.1 or higher
- At least 70 MBytes of free disk space
- At least 50 MBytes of swap space (more is required if your IP models are more complex)

Overview

Using the Visual IP models on UNIX workstations involves the following steps:

1. Download and install the Visual IP models you want to simulate.
2. Download and install the Visual IP End User software.
3. Set an environment variable pointing to the Visual IP models.
4. Run the Visual IP setup script.
5. Simulate with your simulation tool and the Visual IP models.

1. Download & Install the Visual IP Models You Want to Simulate

Altera provides Visual IP models in two ways: included in a MegaCore function installation package or as a separate file. You can download both the core and the models (if provided separately) from the same download page on the Altera web site. To download and install the core and/or models, perform the following steps:

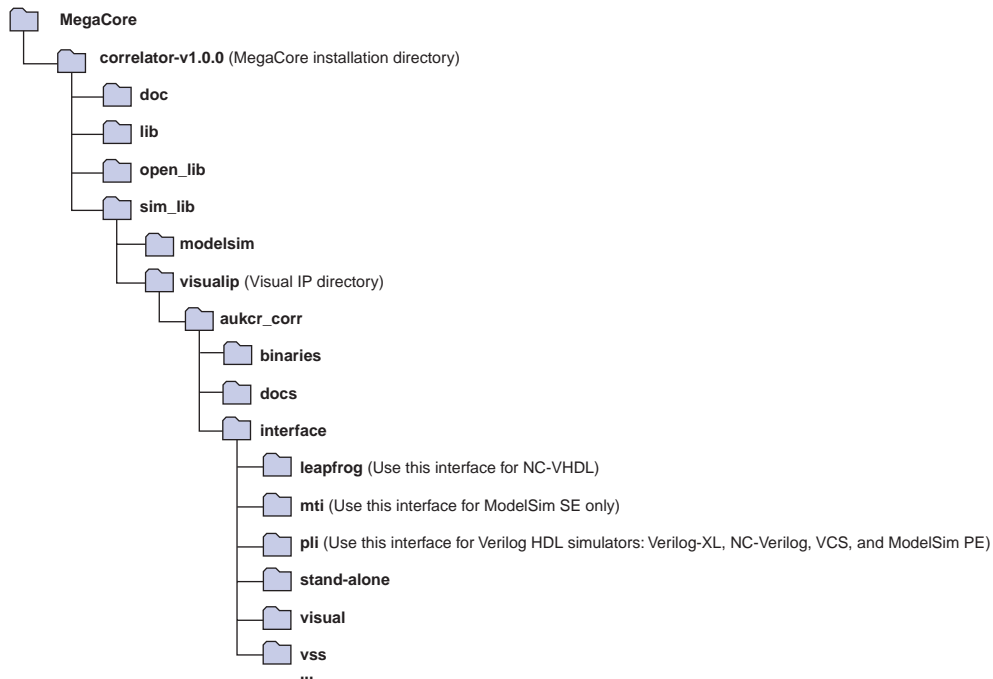
1. Point a web browser to <http://www.altera.com/IPmegastore>.
2. Enter the core name in the **Keyword Search** box of the IP MegaSearch area and click **Submit**.
3. Click the **Try** icon next to the MegaCore function you want to download.
4. Follow the on-line instructions to download the core and/or models and save them to your hard disk.
5. Refer to the core readme file and user guide for detailed installation instructions.



If the Visual IP models are provided as a separate file, a separate user guide is also provided. Refer to this user guide for instructions on how to install the models.

Figure 1 shows the directory structure of an example MegaCore function that is packaged with Visual IP models. The figure explains where to find the Visual IP models used for various simulators.

Figure 1. Location of Visual IP Models




2. Download & Install the Visual IP Software

To download and install the software, perform the following steps:

1. Point your web browser to http://www.altera.com/products/ip/altera/visual_ip.html.
2. Follow the on-line instructions to download the software.
3. The Visual IP software for UNIX has been compressed using the **gzip** utility. Move the **VIP_EndUser_<version><platform>.bin.gz** file to the location in which you want to install the software.
4. Unzip the file.
5. Run the **.bin** file.

3. Set an Environment Variable Pointing to the Visual IP Models

Before you can use the Visual IP models, you must create an environment variable that points to the location of the files.

 The information presented here assumes that you are using the C shell. If not, you must use the appropriate syntax and procedures to set environment variables for your shell. If a command is split over two lines, you should still enter it as a single line following the system prompt.

For Models Included as Part of a MegaCore Package

If you downloaded and installed the Visual IP models as part of the MegaCore package, perform the following steps:

1. Refer to the directory structure information in the core user guide to determine in which directory the Visual IP models were installed.
2. Set an environment variable pointing to the directory you found in step 1 by typing the following command at a UNIX prompt:

```
setenv VIP_MODELS_DIR /<path>/sim_lib/visualip ↵
```

where *<path>* is where you installed the MegaCore function. For example:

```
setenv VIP_MODELS_DIR  
/home/altera_user/megacore/reed-solomon/sim_lib/visualip ↵
```

For Models Downloaded Separately

If you downloaded the Visual IP models as a separate file, perform the following steps:

1. Change to your home or working directory, for example:

```
cd /home/altera_user ↵
```

2. Create a directory for the Visual IP models:


```
mkdir vip ↵
```

3. Set an environment variable pointing to the directory you made in step 2:

```
setenv VIP_MODELS_DIR /home/altera_user/vip ↵
```

4. Run the Visual IP Setup Script

Innoveda provides a setup script for setting the environment variables needed by the Visual IP software. To run the script, perform the following steps.

 The information presented here assumes that you are using the C shell. If not, you must use the appropriate syntax and procedures to set environment variables for your shell.

- ✓ Enter the following command:


```
source /<path>/VIP_EndUser_<version>/<platform>/bin/vip.setup ↵
```

where *<path>* is the location in which you installed the Visual IP software, *<version>* is the Visual IP version, and *<platform>* is **SunOS5** or **hp10**. For example:

```
source /<path>/VIP_EndUser_43/SunOS5/bin/vip.setup ↵
```

5. Simulate the Visual IP Models

After you install the Visual IP software and models and set up your environment, you are ready to simulate. You can simulate using a single command or you can perform incremental compilation and simulation.

 The information presented here assumes that you are using the C shell. If not, you must use the appropriate syntax and procedures to set environment variables for your shell. If a command is split over two lines, you should still enter it as a single line following the system prompt.

This white paper provides instructions for the following simulators:

- NC-Verilog
- Verilog-XL
- VCS
- ModelSim


NC-Verilog

The following section describes how to simulate using the NC-Verilog simulator.

Simulate Using a Single Command

The command to simulate is:

```
ncverilog <path>/<model>.v <path>/<wrapper>.v <path>/<testbench>.v +debug +gui ↵
```

 The `+debug` option allows read access of the model signals. The `+gui` option launches the graphical interface for viewing waveforms.

Below is an example using the POS-PHY L2 L3 compiler models (which were installed separately into the `/home/altera_user/vip` directory) and the Altera-provided testbench and wrapper files (which were installed into the `/MegaCore/POSPHY` directory when the POS-PHY MegaCore function was installed).

`<path>/<model>.v` (Altera-provided Visual IP model):

```
/home/altera_user/vip/auk_pac_mrx_mw/interface/pli/auk_pac_mrx_mw.v
```

`<path>/<wrapper>.v` (either the `.v` file created by the MegaCore wizard or an Altera-supplied one):

```
/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_pl3_link.v
```

`<path>/<testbench>.v` (a testbench you have written or an Altera-supplied one):

```
/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_ref_tb.v
```

Example command:

```
ncverilog /home/altera_user/vip/auk_pac_mrx_mw/interface/pli/auk_pac_mrx_mw.v
          /MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_pl3_link.v
          /MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_ref_tb.v +debug +gui ↵
```

To view the simulation in the user interface, perform the following steps:

1. Choose **Navigator** (Tools menu).
2. Double-click on the testbench filename (**auk_pac_mrx_ref_tb.v** in the example above).
3. Select all of the nets and registers.
4. Select the waveform view.
5. Click **Run**.
6. Zoom out to view the waveform.

Perform Incremental Compilation & Simulation

For incremental compilation and simulation, perform the following steps:

1. Create a simulation directory, for example:

```
mkdir my_sim ↵
```

2. Change to the simulation directory:

```
cd my_sim ↵
```

3. Create a working library directory in your simulation directory. For example:

```
mkdir worklib ↵
```

4. Create a file named **cds.lib** in the **my_sim** directory that contains the following line:

```
DEFINE worklib ./worklib
```

5. Create a file named **hdl.var** in the **my_sim** directory that contains the following line:

```
DEFINE work worklib
```

6. Parse the Verilog HDL source code:

```
ncvlog <path>/<model>.v <path>/<wrapper>.v <path>/<testbench>.v ↵
```

For example:

```
ncvlog /home/altera_user/vip/auk_pac_mrx_mw/interface/pli/auk_pac_mtx_mw.v  
/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mtx_pl3_link.v  
/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mtx_ref_tb.v ↵
```

7. Elaborate the design (the `-rwc` option—read, write, connect—is required if you want to see all of the signals later):

```
ncelab worklib.auk_pac_mtx_ref -access +rwc -messages ↵
```


8. Simulate the design (the `-gui` option launches the graphical interface for viewing the waveforms):

```
ncsim worklib.auk_pac_mtx_ref -gui ↵
```

Verilog-XL

The command to simulate is:

```
verilog -s <path>/<model>.v <path>/<wrapper>.v <path>/<testbench>.v +debug +gui ←
```

 The `-s` option stops the simulator from completing simulation before you can open the waveform window to view the signal. The `+debug` option allows read access of the model signals. The `+gui` option launches the graphical interface for viewing waveforms.

Below is an example using the POS-PHY L2 L3 compiler models (which were installed separately into the `/home/altera_user/vip` directory) and the Altera-provided testbench and wrapper files (which were installed into the `/MegaCore/POSPHY` directory when the POS-PHY MegaCore function was installed).

<path>/<model>.v (Altera-provided Visual IP model):

```
/home/altera_user/vip/auk_pac_mrx_mw/interface/pli/auk_pac_mrx_mw.v
```

<path>/<wrapper>.v (either the .v file created by the MegaCore wizard or an Altera-supplied one):

```
/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_pl3_link.v
```

<path>/<testbench>.v (a testbench you have written or an Altera-supplied one):

```
/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_ref_tb.v
```

Example command:

```
verilog -s /home/altera_user/vip/auk_pac_mrx_mw/interface/pli/auk_pac_mrx_mw.v
/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_pl3_link.v
/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_ref_tb.v +debug +gui ←
```

VCS

You can simulate using a single command and you need to link to the **pli.tab** and **libvip_pli.a** files provided with the Visual IP software. To simulate, use the following command:

```
vcs -P <path>/pli.tab <path>/libvip_pli.a <path>/<model>.v <path>/<wrapper>.v
<path>/<testbench>.v -RI ←
```

 The `-RI` option launches the interactive interface.

Below is an example using the POS-PHY L2 L3 compiler models (which were installed separately into the `/home/altera_user/vip` directory) and the Altera-provided testbench and wrapper files (which were installed into the `/MegaCore/POSPHY` directory when the POS-PHY MegaCore function was installed).

<path>/pli.tab:

```
<installation path>/VIP_EndUser_43/SunOS5/verilog_src/pli.tab
```

<path>/libvip_pli.a:

```
<installation path>/VIP_EndUser_43/SunOS5/verilog_src/libvip_pli.a
```

`<path>/<model>.v` (Altera-provided Visual IP model):

`/home/altera_user/vip/auk_pac_mrx_mw/interface/pli/auk_pac_mrx_mw.v`

`<path>/<wrapper>.v` (either the .v file created by the MegaCore wizard or an Altera-supplied one):

`/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_pl3_link.v`

`<path>/<testbench>.v` (a testbench you have written or an Altera-supplied one):


`/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_ref_tb.v`

Example command:

```
vcs -P <installation path>/VIP_EndUser_43/SunOS5/verilog_src/pli.tab
<installation path>/VIP_EndUser_43/SunOS5/verilog_src/libvip_pli.a
/home/altera_user/vip/auk_pac_mrx_mw/interface/pli/auk_pac_mrx_mw.v
/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_pl3_link.v
/MegaCore/POSPHY/sim_lib/testbench/verilog/auk_pac_mrx_ref_tb.v ←
```

ModelSim

The PLI interface is used to instantiate the Visual IP models using Verilog HDL. To use the PLI interface, perform the following steps. If you want to instantiate the Visual IP models using VHDL, skip to [step 6](#).

 You must set a variable in the **modelsim.ini** file to use the ModelSim PLI interface. These instructions describe how to update your local copy of the **modelsim.ini** file in your working directory.

1. Create a local copy of the **modelsim.ini** file by performing the following steps:
 - a. Create a new directory.
 - b. Run the ModelSim software.
 - c. Change to the new directory you created.
 - d. Perform some action in the ModelSim software, e.g., change the compile options. ModelSim automatically creates a new **modelsim.ini** file in your working directory.
 - e. Close ModelSim so that you do not inadvertently overwrite the **modelsim.ini** file.
2. Open the **modelsim.ini** file with a text editor.
3. Find the line that has the text string `veriuser`.
4. Change the line to point to the Visual IP **libplimtivip** file. For example:

```
veriuser = <installation path>/VIP_EndUser_43/SunOS5/lib/libplimtivip
```
5. Save the **modelsim.ini** file.
6. Compile the model, wrapper, and testbench (your own or an Altera-provided one).
7. Load the testbench.
8. Open the signal and waveform windows.

9. Select signals you wish to monitor.

10. Run the simulation.



Once the Visual IP model has been loaded, you must close ModelSim and then reopen it to restart the simulation or to load a different model.



101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
<http://www.altera.com>

Copyright © 2001 Altera Corporation. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services. All rights reserved.