

# Board Level Simulation with

Board-level simulation capability has been added to the OrCAD VST simulator in the latest release, OrCAD VST 386+ v1.20. Users can now simulate a board-level design containing multiple Xilinx FPGAs and EPLDs.

For board-level simulation, create each Xilinx design in its own directory. Within each directory, route, back-annotate and simulate the design at the chip level. After each design has been verified, create a separate board-level directory and enter the board-level design. Copy the simulation files (.VST, .DBA) for the individual devices into the board-level directory. A board-level simulation can then be performed in the board directory. The procedures are as follows:

## Create the Simulation Model Library

In order to simulate designs in VST v1.20, each device source file (.DSF) must be compiled in the VST v1.20 format. If you plan to simulate designs from different Xilinx families together or include components from other vendors on a single board design, add those DSFs to the simulation model library using OrCAD VST's *Add*

*Device Model* command for each file. The DSFs for Xilinx are available on the Xilinx BBS as ORCSRC.ZIP. The source files are also available on the XACT 5.1 CD under XBBS\SWHELP\ORCSRC.ZIP.

## Prepare Individual Chip Designs

Each FPGA/EPLD design should be captured in OrCAD SDT 386+ using normal procedures. When labeling the signals connected to the I/O pads, use easy-to-identify, intuitive names to facilitate the process of creating a chip symbol to represent the FPGA/EPLD. For board-level simulation, these signal names provide the points of connectivity to the board-level wires. Create names of 14 characters or

fewer to prevent the software from creating shorter, random aliases.

After a design has been completed, use the following commands to generate the simulation files:

- 1: `xmake <design>`  
(for EPLD designs, replace `xmake` with `xemake`)
- 2: `xsimmake -f oft <design>`  
(for EPLD designs, replace `OFT` with `OET`)

Once the simulation files are created and the design verified, copy the simulation files (<design>.VST, <design>.DBA) to the board directory.

## Create a Library of Chip Symbols

Before preparing the board level schematic, create a library of symbols to represent each device on the board (the procedure is described on pages 3-8 through 3-10 of the *XACT OrCAD Interface User Guide*, April 1994). Make certain that the symbol's pin names match the underlying I/O signal names. In addition to the user I/O signals, pins for the global signals must be added. The names of the global signals are:

- For the XC2000 and XC3000 FPGA devices, the global reset signal is GR.
- For the XC4000 FPGA devices, global set/reset is GSR, and global tristate is GTS (if *STARTUP* is used, GSR and/or GTS are replaced with user signals).
- For the XC7000 EPLD devices, global set/reset is PRLD.

## Prepare the Board Level Design

Once the custom symbol library is in place, the board level schematic can be created using the normal OrCAD SDT procedures. In addition to the user-created library, the board design may also use libraries from other vendors if their DSF files have been included. Once the board design has been captured, run ANNOTATE on the schematic to update the reference designators for the symbols in the schematic. Then, from within SDT, edit the

CONTINUED

*“Once the custom symbol library is in place, the board level schematic can be created using the normal OrCAD SDT procedures.”*

SHEETPARTNAME for each Xilinx device symbol to add the following:

```
EXTERNALVIEW=<design>.SCH
```

where <design> is the root schematic worksheet for that particular chip. After adding the 'externalview' to all Xilinx devices, save the schematic, and then run INET on the board schematic to create the board-level INF simulation netlist.

### Simulate the Board Level Design

Prior to simulating the board-level design, set the SIMULATE local configura-

tion to *Use all delay annotation files (include separate dba files)*. Verify that the library prefix for the digital simulation tools points to the simulation model library created in the *Create the Simulation Model Library* section above. Simulate the board-level design using <board>.INF as the connectivity database.

A detailed application note with sample design files is available on the Xilinx BBS (Bulletin Board System) under the filename: VSTBSIM.ZIP. The application note is also available via the Xilinx XDOCS system (email xdocs@xilinx.com, document key 23012). ♦

## Configuration Checklist Available

A new, preliminary application note discussing FPGA configuration issues is now available on both the XFACTS and XDOCS technical support systems. The application note is intended to help determine if you have a configuration problem and, if so, suggest likely solutions. (It complements the already-existing application note, "FPGA Configuration Guidelines," document #0010229-01).

To get the current revision of this new document via Fax from XFACTS, please call 408-879-4400 from a touch-tone telephone and press "1" to get more information; this document is number 23021. If you are using the XDOCS E-mail system, this document can be ordered by sending the command "SEND 23021" as the subject line or in the body of a message addressed to xdocs@xilinx.com. (To learn the basics of XDOCS, please mail xdocs@xilinx.com with the word help in the subject line.)

This application note navigates you through a number of questions. Initially, you must determine if the problem is related to the configuration process or the functionality of the configured FPGA (Is there a problem with the functionality of a correctly-configured device, or has the configuration actually failed?)

There are several clues as to whether or not the FPGA has been successfully configured:

1. **INIT** — If configuring from power-up, does INIT go through a single positive transition from Low to High? If the FPGA is not working after it is reprogrammed, is there a single negative transition (High to Low) on INIT, followed by a single positive-going transition?
2. **DONE** — Does DONE go High?
3. **I/Os** — Are your I/Os at their active post-configuration levels? HDC is High during configuration, and LDC should be Low during configuration. Most other pins have weak pullups during configuration. If you configure the device to pass a clock signal in through an input and out through an output pad, does the output pin toggle at the end of configuration? If you configure one user I/O to drive Low at the end of configuration, and another to drive High, are both I/Os driving the proper logic levels at the end of configuration?
4. **Flip-Flops** — Are your flip-flops toggling?

If you can answer YES to these questions, you can be fairly certain that your FPGA has been configured properly, and that the problem involves the functionality of your design, not the configuration method.

Please note that this document is preliminary and feedback is very much appreciated; feedback can be sent by E-mail to hotline@xilinx.com or by FAX to 408-879-4442. ♦